

# MANUAL LOGO!

FUNCIONES ESPECIALES

# ÍNDICE

- 1      *Retardo en la conexión.*
- 2      *Retardo a la desconexión.*
- 3      *Retardo conexión/desconexión.*
- 4      *Retardo a la conexión con memoria.*
- 5      *Relé de barrido (Salida de impulsos).*
- 6      *Relé de barrido disparado por flancos.*
- 7      *Generador de impulsos asíncronos.*
- 8      *Generador aleatorio.*
- 9      *Interruptor de alumbrado para escalera.*
- 10     *Interruptor bifuncional.*
- 11     *Temporizador semanal.*
- 12     *Temporizador anual.*
- 13     *Reloj astronómico.*
- 14     *Cronómetro.*
- 15     *Contador adelante/atrás.*
- 16     *Contador de horas de funcionamiento.*
- 17     *Selector de umbral.*
- 18     *Instrucción aritmética.*
- 19     *Comparador analógico.*
- 20     *Conmutador analógico de valor de umbral.*
- 21     *Amplificador analógico.*
- 22     *Vigilancia del valor analógico.*
- 23     *Conmutador analógico de valor umbral diferencial.*
- 24     *MUX analógico.*
- 25     *Rampa analógica.*
- 26     *Regulador PI.*
- 27     *PWM.*
- 28     *Filtro analógico.*
- 29     *Max/Min.*
- 30     *Valor medido.*
- 31     *Relé autoenclavador.*
- 32     *Relé de impulsos.*
- 33     *Texto de aviso.*
- 34     *Interruptor de software.*
- 35     *Registro de desplazamiento.*
- 36     *Detección de error de la instrucción aritmética.*
- 37     *Convertidor flotante/entero.*
- 38     *Convertidor entero/flotante.*

## ***Introducción***

El presente manual ha sido elaborado con el propósito de ofrecer una guía clara, estructurada y confiable para el uso, comprensión y aplicación de los diferentes bloques funcionales y herramientas involucradas en el desarrollo de proyectos de automatización. En un entorno industrial cada vez más dinámico, la correcta interpretación y utilización de estos elementos resulta fundamental para garantizar sistemas eficientes, seguros y capaces de responder a las exigencias actuales de control.

A lo largo del documento se presentan descripciones detalladas, ejemplos prácticos y explicaciones precisas que permitirán al lector adquirir una visión amplia del funcionamiento de cada bloque, así como de sus parámetros, modos de operación y aplicaciones más comunes. La información contenida ha sido organizada de forma progresiva, con el fin de facilitar el aprendizaje tanto a usuarios principiantes como a estudiantes de ingeniería, técnicos en formación o profesionales que desean reforzar sus conocimientos.

Este manual también busca servir como una herramienta de consulta rápida durante el diseño, la implementación y la puesta en marcha de sistemas basados en controladores lógicos programables. Por ello, cada sección se ha redactado con un enfoque práctico, orientado a resolver dudas habituales y a brindar al usuario la capacidad de identificar, configurar y adaptar cada bloque a las necesidades específicas de su proyecto.

Asimismo, se han incorporado explicaciones conceptuales que permiten comprender el fundamento teórico detrás de cada función. Esto no solo ayuda a evitar errores comunes, sino que también fomenta el análisis crítico y el desarrollo de soluciones más robustas, confiables y eficientes.

En resumen, este manual pretende convertirse en un recurso integral que acompañe al usuario en cada etapa de su proceso de aprendizaje y trabajo, proporcionando información precisa, ejemplos útiles y un lenguaje accesible. Su objetivo es fortalecer las competencias técnicas del lector y facilitar la creación de sistemas de automatización modernos, sólidos y orientados a una operación segura y estable.

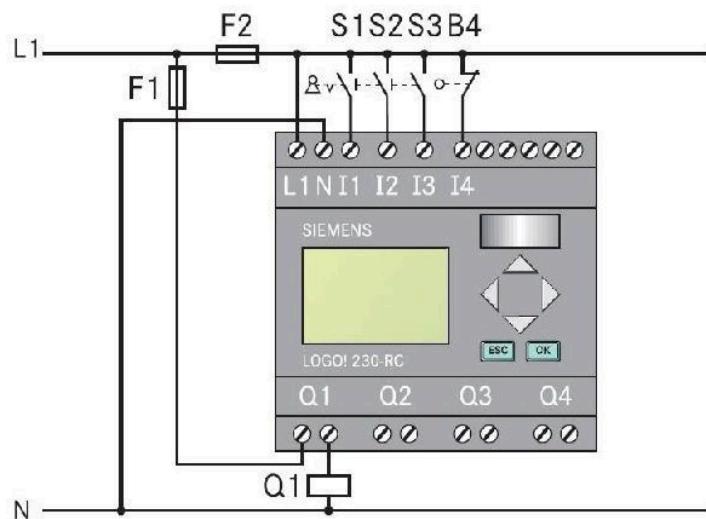
## *¿Qué es un plc?*

un plc (controlador lógico programable, por sus siglas en inglés programmable logic controller) es un dispositivo electrónico diseñado para automatizar procesos industriales mediante la ejecución de instrucciones lógicas previamente programadas. su función principal es recibir señales de entrada provenientes de sensores, interruptores o dispositivos externos, procesarlas de acuerdo con un programa interno y generar señales de salida que controlan actuadores como motores, válvulas, relevadores o sistemas de iluminación.

los plc se caracterizan por su alta confiabilidad, robustez y capacidad para operar de manera continua en entornos industriales exigentes, donde existen vibraciones, cambios de temperatura, ruido eléctrico y condiciones que harían inadecuados a equipos informáticos convencionales. además, permiten modificaciones rápidas en la lógica de control sin necesidad de alterar físicamente el cableado, lo que facilita el mantenimiento, la expansión y la adaptabilidad del sistema.

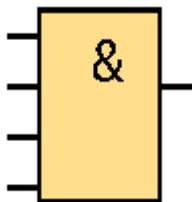
un plc está compuesto por módulos esenciales como la unidad central de procesamiento (cpu), las memorias, y módulos de entradas y salidas digitales o analógicas. a través de estos componentes, el controlador interpreta datos del proceso, toma decisiones en tiempo real y asegura que los equipos funcionen de acuerdo con los parámetros establecidos.

en la actualidad, los plc forman parte fundamental de la automatización moderna, al integrarse con redes industriales, sistemas scada y otros dispositivos inteligentes. gracias a su versatilidad y facilidad de programación, se utilizan en líneas de producción, sistemas de transporte, control de maquinaria, monitoreo energético, procesos continuos y prácticamente cualquier aplicación donde se requiera un control preciso, seguro y eficiente.



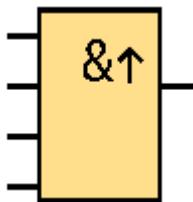
## *Compuertas lógicas en LOGO;*

### **AND**



El bloque AND es una función lógica que produce una salida activa únicamente cuando todas sus entradas están activadas simultáneamente. Si al menos una de las entradas se encuentra en estado desactivado, la salida permanecerá apagada. Este bloque se utiliza para condiciones donde deben cumplirse varios requisitos al mismo tiempo para permitir el accionamiento de un proceso o dispositivo.

### **AND (FLANCO)**

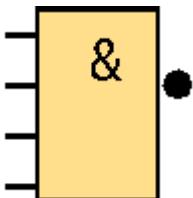


La salida de una función AND con evaluación de flancos solo adopta el estado 1 si todas las entradas tienen el estado 1 y por lo menos una de ellas tenía el estado 0 en el ciclo anterior.  
La salida permanece puesta a 1 durante exactamente un ciclo, debiéndose poner de nuevo a 0 durante un ciclo como mínimo antes de poder adoptar otra vez el estado 1.

Si no se utiliza una entrada de este bloque (x), se le asigna el valor x = 1.

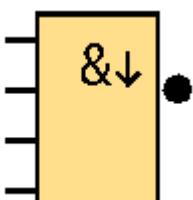
Cronograma de la función AND con evaluación de flancos:

### **NAND**



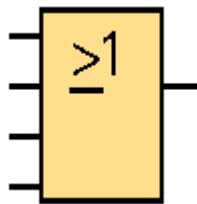
La salida de la función NAND solo adopta el estado 0 si todas las entradas tienen el estado 1, es decir, si están cerradas.  
Si no se utiliza una entrada de este bloque (x), se le asigna el valor x = 1.

### **NAND (FLANCO)**



La salida permanece puesta a 1 durante exactamente un ciclo, debiéndose poner de nuevo a 0 durante un ciclo como mínimo antes de poder adoptar otra vez el estado 1.  
Si no se utiliza una entrada de este bloque (x), se le asigna el valor x = 1.

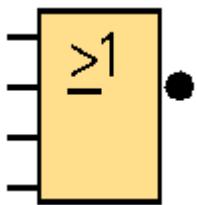
## ***OR***



La salida de una función OR adopta el estado 1 si por lo menos una entrada tiene el estado 1, es decir, si está cerrada.

Si no se utiliza una entrada de este bloque (x), se le asigna el valor x = 0.

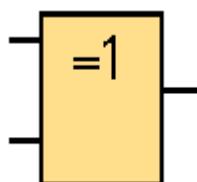
## ***NOR***



La salida de una función NOR (NOT OR) solo adopta el estado 1 si todas las entradas tienen el estado 0, es decir, si están abiertas. Tan pronto como se activa una de las entradas (es decir, cuando adopta el estado 1), se desactiva la salida.

Si no se utiliza una entrada de este bloque (x), se le asigna el valor x = 0.

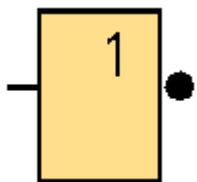
## ***XOR***



La salida de una función XOR (O-exclusiva) adopta el estado 1 si las entradas tienen diferentes estados lógicos.

Si no se utiliza una entrada de este bloque (x), se le asigna el valor x = 0.

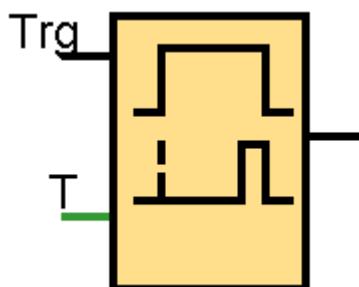
## ***NOT***



La salida de una función NOT adopta el estado 1 si la entrada tiene el estado 0. El bloque NOT invierte el estado de la entrada.

La ventaja de NOT es p. ej. que para LOGO! ya no se necesitan contactos normalmente cerrados. Solo tiene que utilizar un contacto normalmente abierto y, mediante el bloque NOT, convertirlo en un contacto normalmente cerrado.

## **RETARDO EN LA CONEXIÓN**



Bloque temporizador que retrasa la activación de una salida durante un tiempo determinado después de recibir una señal de entrada. Si la entrada se desactiva antes de finalizar el tiempo programado, el conteo se reinicia. Se utiliza para evitar activaciones inmediatas o para secuenciar operaciones.

### ***Pinout y descripción de conexiones***

#### ***Entrada (Trg):***

Cuando este pin pasa de 0 a 1 (nivel bajo a alto), comienza a contar el tiempo de retardo programado. Si el pin Trg vuelve a 0 antes de que termine el tiempo, el conteo se reinicia y la salida no se activa.

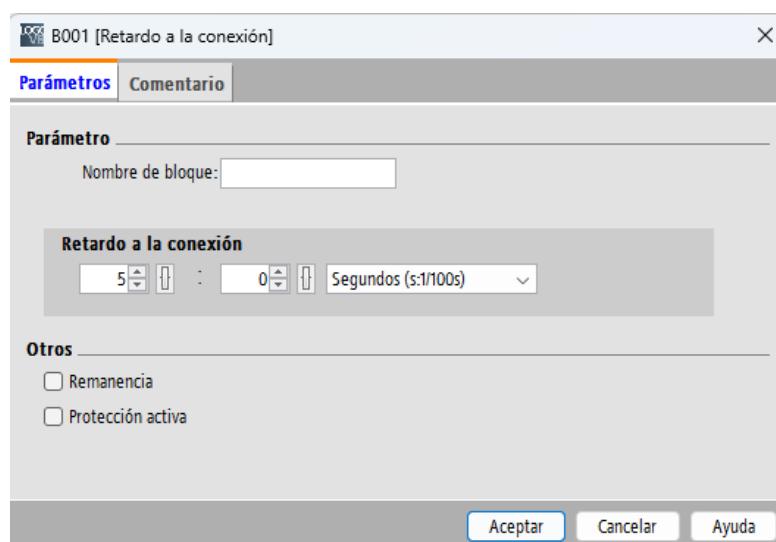
#### ***Time (T):***

Valor de tiempo programable que determina cuánto debe esperar el bloque antes de activar su salida después de recibir la señal de disparo (Trg).

#### ***Salida (Q):***

Salida que se activa una vez transcurrido el tiempo de retardo programado después de que la entrada Trg permanece en nivel alto. Si la entrada se desactiva antes de completar el tiempo, la salida no se activa.

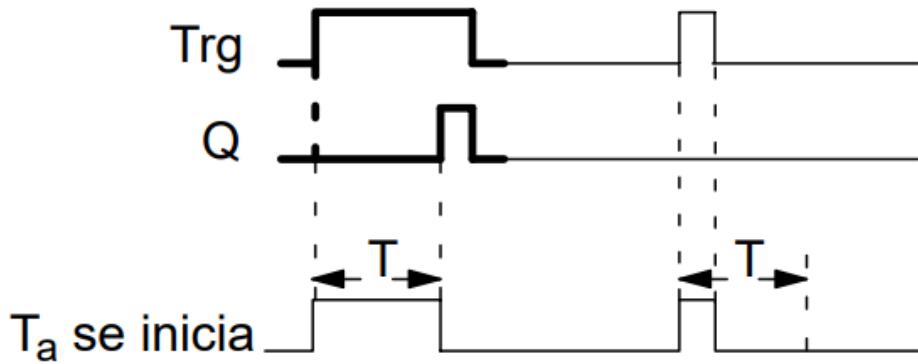
## **Parámetros**



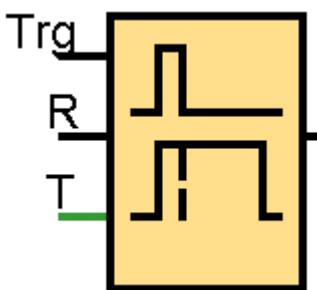
#### **Retardo a la conexión**

Es el tiempo que espera un dispositivo antes de activar o energizar una carga después de haber recibido la señal de encendido.

## Cronograma



## RETARDO A LA DESCONEXIÓN



El retardo a la desconexión en LOGO! es una función especial que mantiene la salida activada durante un tiempo determinado después de que la señal de entrada se desactiva.

En otras palabras, cuando la entrada pasa de “1” a “0”, el bloque no apaga inmediatamente la salida, sino que espera el tiempo programado antes de hacerlo.

## Pinout y descripción de conexiones

### Entrada (Trg):

Recibe la señal de control. Cuando la entrada pasa de nivel alto (“1”) a nivel bajo (“0”), se inicia el conteo del tiempo de retardo a la desconexión. Mientras Trg esté en “1”, la salida permanece activada.

### Reset (R):

Fuerza la desactivación inmediata de la salida Q y la anulación del tiempo de retardo, sin importar el estado de Trg ni del temporizador.

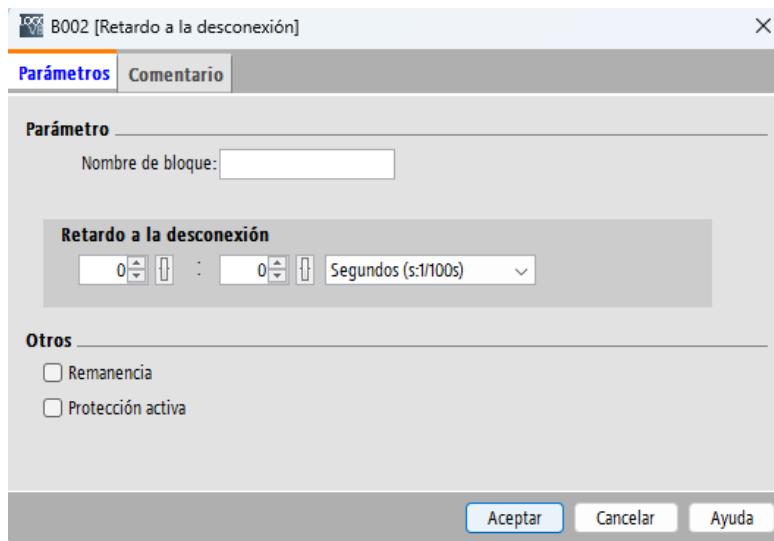
### Time (T):

Define el intervalo de retardo a la desconexión, es decir, el tiempo durante el cual la salida permanece activa después de que Trg pasa a “0”. Se configura en segundos o milisegundos según la necesidad.

### Salida (Q):

Permanece activada mientras Trg esté en nivel alto. Una vez que Trg pasa a nivel bajo, Q continúa activa durante el tiempo definido por T, y luego se desactiva automáticamente.

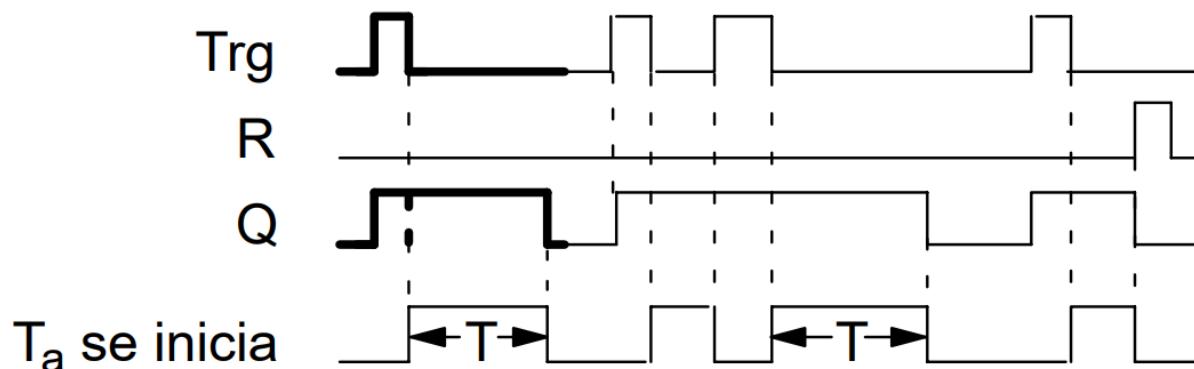
## Parámetros



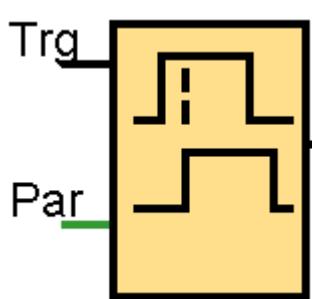
### Retardo a la desconexión

Es el tiempo que el dispositivo mantiene la salida activada después de recibir la orden de apagado o después de finalizar la condición de activación.

## Cronograma



## RETARDO A LA CONEXIÓN/DESCONEXIÓN



El retardo a la conexión/desconexión en LOGO! es una función temporizadora combinada que retrasa tanto la activación como la desactivación de la salida en función del estado de la señal de entrada.

De forma técnica, cuando la entrada Trg pasa de “0” a “1”, la salida Q se activa únicamente después de transcurrido el tiempo de conexión (TON) configurado. A su vez, cuando la entrada vuelve de “1” a “0”, la salida permanece activa durante el tiempo de desconexión (TOF) antes de desactivarse.

## **Pinout y descripción de conexiones**

### **Entrada (Trg):**

Recibe la señal de control que activa o desactiva el temporizador. Al detectar un flanco ascendente (de “0” a “1”), se inicia el tiempo de conexión; al detectar un flanco descendente (de “1” a “0”), comienza el tiempo de desconexión.

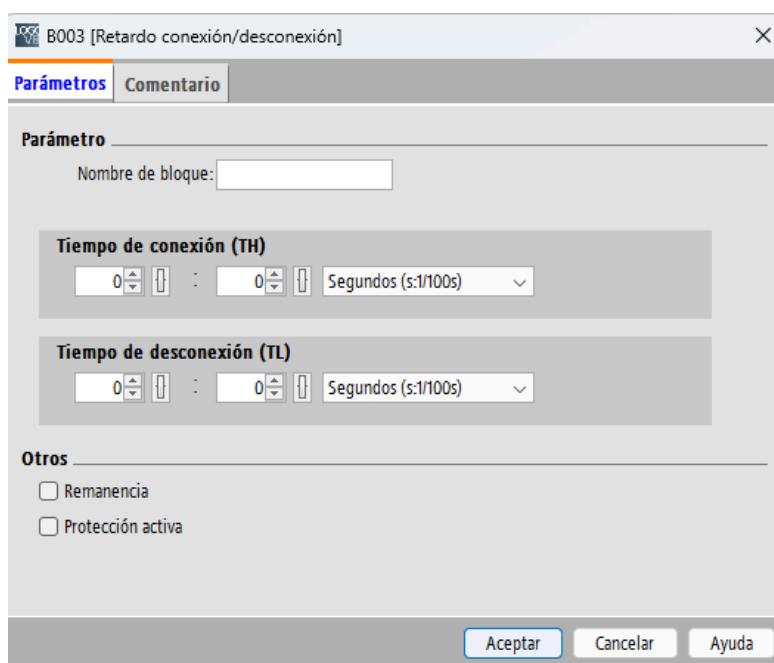
### **Parametro (Par):**

Define los tiempos de conexión y desconexión del bloque. Permite ajustar cuánto debe esperar la salida Q antes de activarse o desactivarse tras el cambio de estado en Trg.

### **Salida (Q):**

Representa la salida controlada por el temporizador. Se activa después del tiempo de conexión y se desactiva una vez transcurrido el tiempo de desconexión.

## **Párametros**



### **Tiempo de conexión (TH)**

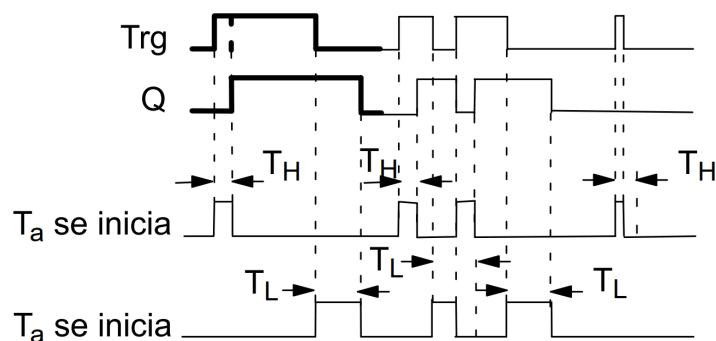
Es el periodo durante el cual la salida permanece activada.

Comienza una vez que la función se inicia o cuando recibe una señal de arranque.

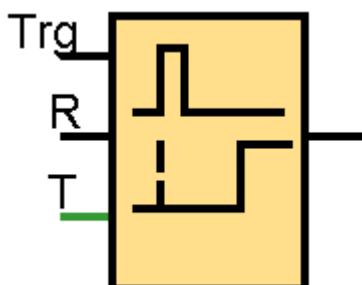
### **Tiempo de desconexión (TL)**

Es el periodo durante el cual la salida permanece desactivada antes de volver a activarse nuevamente, formando un ciclo completo con el tiempo de conexión.

## **Cronograma**



## **RETARDO A LA CONEXIÓN CON MEMORIA**



El retardo a la conexión con memoria en LOGO! es una función temporizadora que retrasa la activación de la salida tras la aparición de una señal de entrada, pero mantiene el tiempo acumulado aunque la entrada se desactive momentáneamente.

Técnicamente, cuando la entrada Trg pasa a nivel alto (“1”), el temporizador comienza a contar el tiempo configurado. Si la señal se interrumpe antes de que el tiempo finalice, el conteo no se reinicia, sino que continúa desde el valor alcanzado cuando la señal vuelve a activarse. Una vez que el tiempo total programado se cumple, la salida Q se activa.

### ***Pinout y descripción de conexiones***

#### ***Entrada (Trg):***

Recibe la señal de activación. Cuando pasa de nivel bajo (“0”) a nivel alto (“1”), inicia el conteo del tiempo de conexión. Si la señal se interrumpe antes de completarse el tiempo, el conteo se mantiene en memoria y continúa al reactivarse la entrada.

#### ***Reset (R):***

Restablece el bloque a su estado inicial. Al activarse, borra el tiempo acumulado y fuerza la desactivación inmediata de la salida Q, sin importar el estado de Trg.

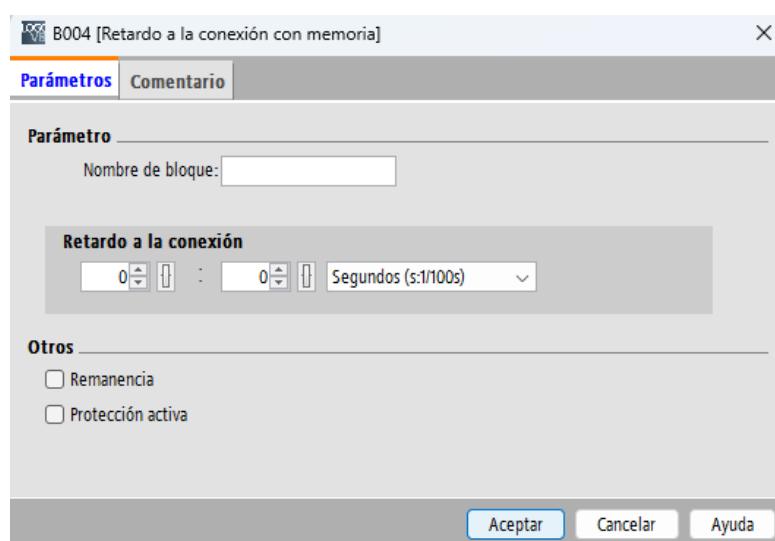
#### ***Time (T):***

Define el tiempo de retardo a la conexión, es decir, el período que debe transcurrir desde la activación de Trg hasta que la salida Q se energiza.

#### ***Salida (Q):***

Permanece desactivada hasta que el tiempo configurado T se cumple. Una vez alcanzado el tiempo total, Q pasa a nivel alto (“1”) y permanece así mientras la entrada Trg esté activa.

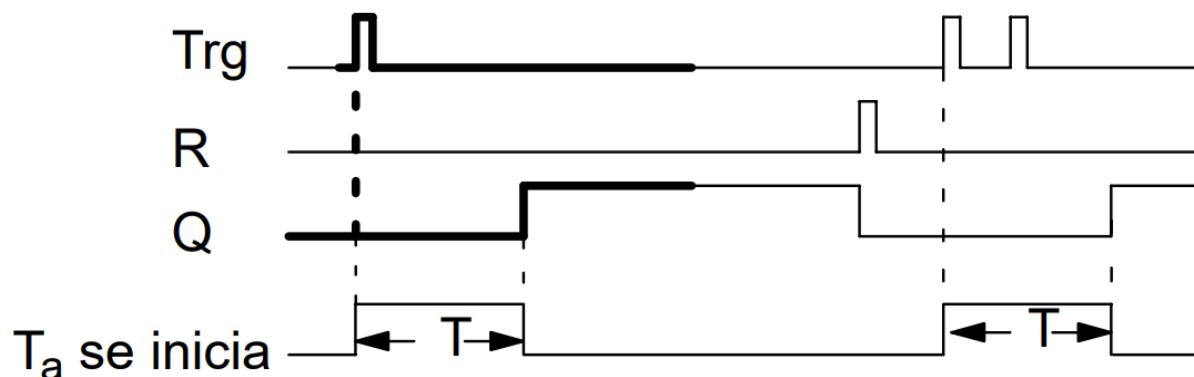
## Párametros



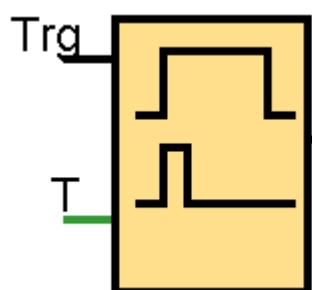
## Retardo a la conexión

Este parámetro define el tiempo que debe transcurrir antes de activar la salida desde el momento en que se recibe la señal de encendido.

## Cronograma



## RELÉ DE BARRIDO (SALIDA DE IMPULSOS).



El relé de barrido (salida de impulsos) en LOGO! es una función que genera impulsos de salida de duración fija cada vez que se activa la señal de entrada.

Técnicamente, al detectar un flanco ascendente en la entrada Trg, el bloque activa la salida Q durante un tiempo determinado (configurado en el parámetro T), tras el cual la salida vuelve automáticamente a nivel bajo, independientemente del estado de la entrada.

## **Pinout y descripción de conexiones.**

### **Entrada (Trg):**

Recibe la señal de activación. Al detectar un flanco ascendente (de “0” a “1”), inicia la generación del impulso en la salida Q, independientemente de la duración de la señal de entrada.

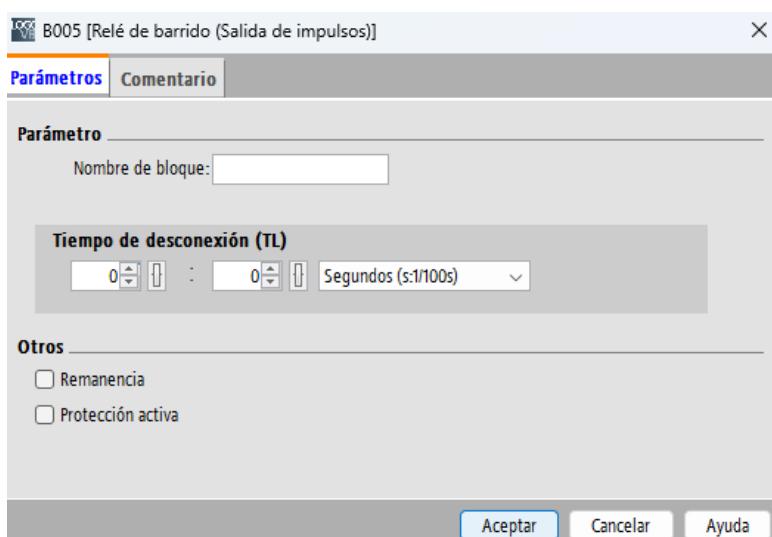
### **Time (T):**

Define la duración del impulso de salida. Es el intervalo durante el cual la salida Q permanece activa tras recibir la señal de disparo en Trg.

### **Salida (Q):**

Emite un impulso de duración fija al activarse Trg. Una vez transcurrido el tiempo configurado en T, la salida retorna automáticamente a nivel bajo (“0”).

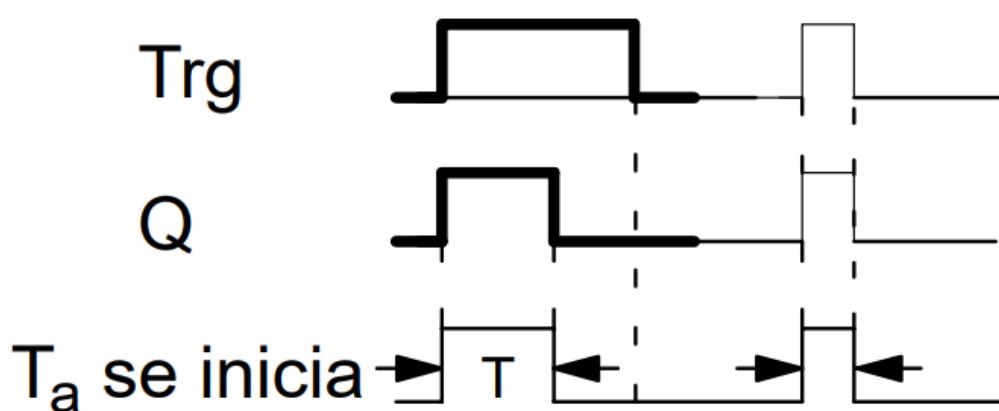
## **Párametros**



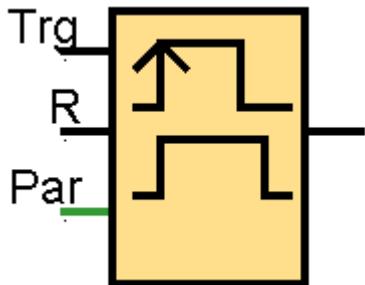
### **Tiempo de desconexión (TL)**

Es el intervalo durante el cual la salida del relé permanece desactivada antes de iniciar un nuevo ciclo de activación.

## **Cronograma**



## **Relé de barrido disparado por flancos**



El Relé de barrido disparado por flancos genera una secuencia de activaciones y desactivaciones cíclicas, pero únicamente cuando se detecta un flanco de disparo en su entrada.

Cada vez que ocurre un flanco ascendente o descendente (según la configuración), el relé inicia un ciclo compuesto por un tiempo de conexión y un tiempo de desconexión. Durante este ciclo, la salida alterna entre estado activo e inactivo hasta que se completa el barrido.

El bloque se activa solo al recibir el evento de flanco, evitando repeticiones continuas mientras la señal permanezca estable.

### **Pinout y descripción de conexiones**

#### *Entrada (Trg):*

Entrada que detecta el flanco configurado (ascendente o descendente). Cuando se produce el flanco, el bloque inicia el ciclo de barrido (tiempo de conexión + tiempo de desconexión). No responde a nivel, solo a cambios.

#### *Reset (R):*

Detiene el ciclo en curso y apaga la salida inmediatamente. Al activar el reset, el temporizador se reinicia y espera un nuevo flanco en la entrada Trigger para comenzar otro ciclo.

#### *Párametro (Par):*

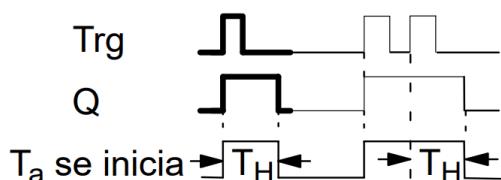
Entrada opcional para modificar valores del bloque externamente, como tiempos o habilitación del barrido. Si no se utiliza, los tiempos se toman de los parámetros configurados en el bloque.

#### *Salida (Q):*

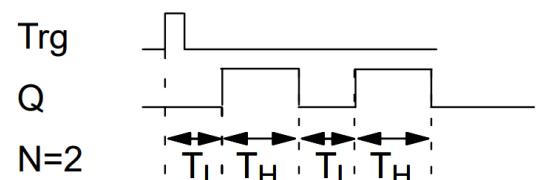
Señal de salida del barrido. Se activa durante el tiempo de conexión y se desactiva durante el tiempo de desconexión, completando un ciclo cada vez que se detecta el flanco válido.

### **Cronograma**

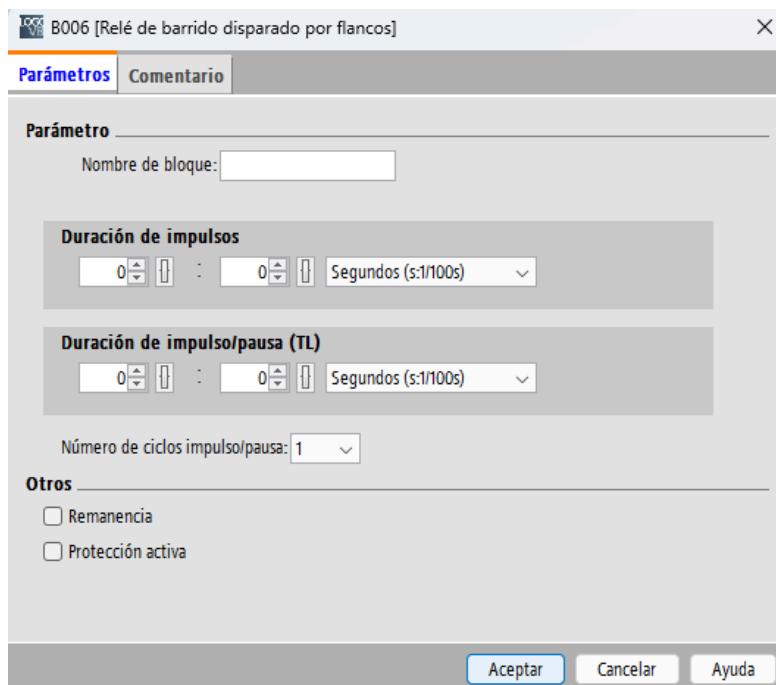
#### **Cronograma A**



#### **Cronograma B**



## Párametros



### Duración de impulsos

Define el tiempo durante el cual la salida permanece activada cada vez que se inicia un barrido.

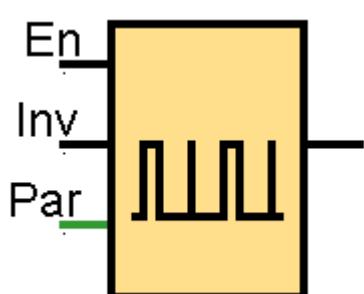
Cuando ocurre el flanco de disparo, la salida se enciende y se mantiene en estado activo por el valor configurado en este parámetro.

### Duración de impulso/pausa (TL)

Determina el ciclo completo del barrido, combinando el tiempo de activación y el tiempo de desactivación en una sola programación.

La salida se activa por la duración del impulso y luego se desactiva durante la pausa, completando así un ciclo. Una vez terminado, la salida queda desactivada hasta que ocurra un nuevo flanco.

## Generador de impulsos asíncrono



El Generador de impulsos asíncrono produce una señal de salida periódica que alterna entre encendido y apagado de forma continua, sin depender de una entrada de disparo.

Su funcionamiento se basa en dos tiempos programables: duración de impulso (tiempo encendido) y duración de pausa (tiempo apagado). Una vez habilitado, el bloque genera pulsos de manera indefinida, formando una onda cuadrada temporizada.

Este generador opera de forma asíncrona, lo que significa que continúa trabajando independientemente del ciclo del programa o de otras funciones del controlador, manteniendo la frecuencia y el duty cycle establecidos.

## Pinout y descripción de conexiones

**Entrada (En):**

Habilita o inicia el generador de impulsos. Cuando esta entrada está activa, el bloque comienza a generar la secuencia de impulsos (encendido y apagado) de forma continua según los parámetros configurados. Si la entrada se desactiva, la salida se apaga y el generador se detiene.

#### *Entrada invertida (Inv):*

Funciona como habilitación por lógica inversa. El generador se activa cuando esta entrada está en nivel bajo y se detiene cuando está en nivel alto. Se utiliza cuando se desea controlar el generador con señal negada sin necesidad de un bloque adicional de inversión.

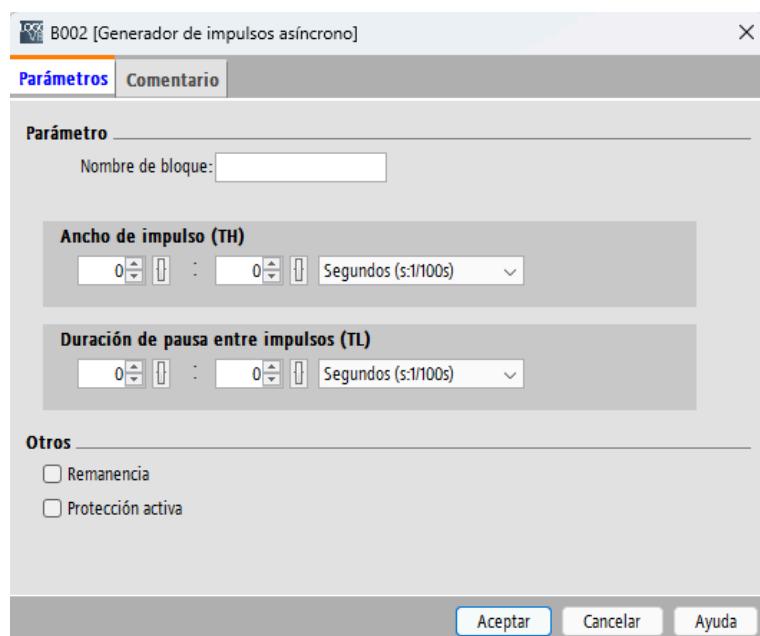
#### *Pámetro (Par):*

Permite ajustar externamente los valores de tiempo del bloque (ancho de impulso y pausa) desde otra parte del programa. Si no se utiliza, el bloque trabaja con los valores definidos dentro de su configuración.

#### *Salida (Q):*

Señal pulsante generada por el bloque. Alterna entre encendido (durante el ancho de impulso) y apagado (durante la pausa) mientras la entrada habilitadora esté activa.

## **Pámetros**



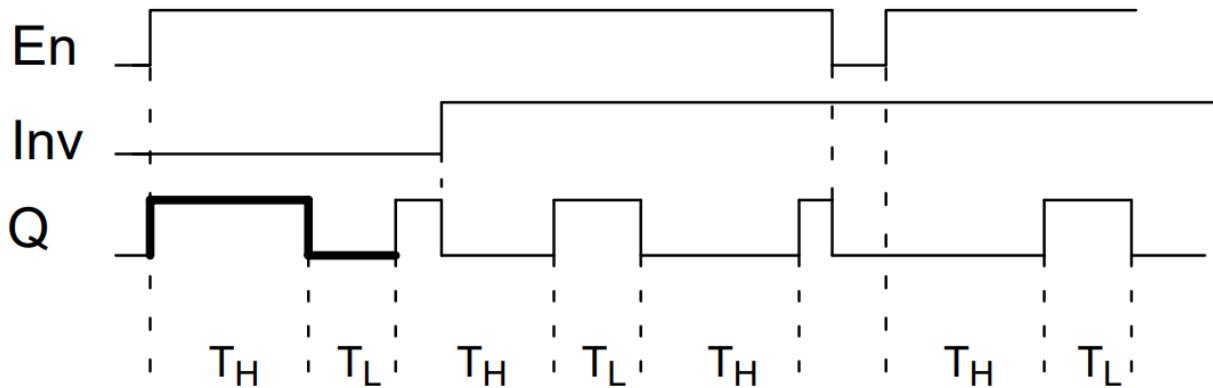
#### **Ancho de impulso (TH)**

Tiempo que la salida permanece encendida en cada pulso.

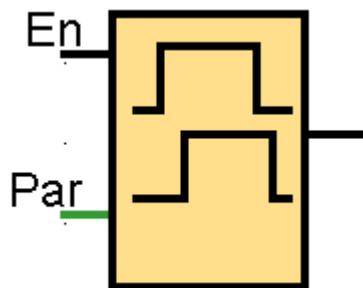
#### **Pausa entre impulsos (TL)**

Tiempo que la salida permanece apagada entre un pulso y el siguiente.

## **Cronograma**



## *Generador aleatorio*



El Generador aleatorio produce una señal de salida que cambia su estado en intervalos variables e impredecibles dentro de un rango de tiempo configurable.

A diferencia de un generador de impulsos convencional, este bloque no utiliza tiempos fijos, sino que selecciona aleatoriamente la duración del impulso y/o la duración de la pausa dentro de los límites establecidos por el usuario.

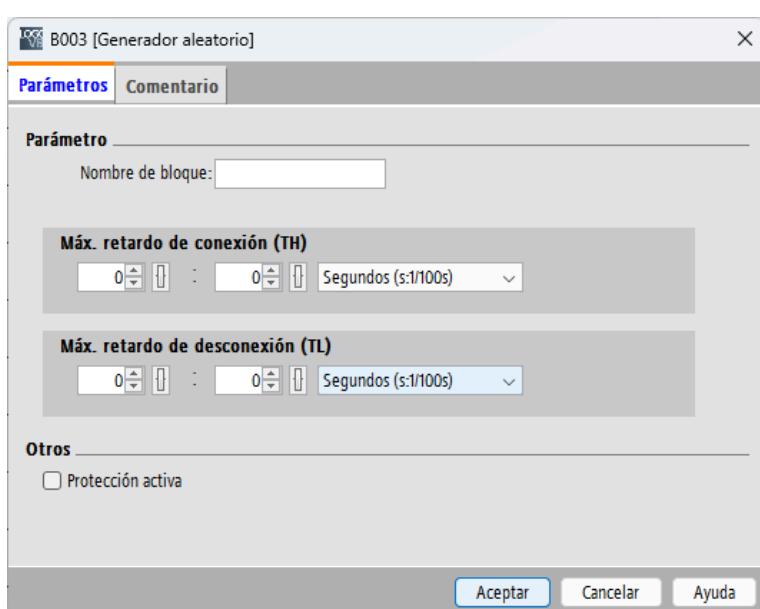
Esto permite simular comportamientos no repetitivos, útiles para pruebas, iluminación decorativa, efectos intermitentes, control de cargas con operación irregular o simulación de fallas/eventos aleatorios.

## *Pinout y descripción de conexiones*

<b>Entrada (En)</b>	Habilita el generador. Cuando esta entrada está activa, el bloque comienza a producir impulsos con tiempos aleatorios dentro de los rangos configurados. Si la entrada se desactiva, la salida se apaga y el generador se detiene.
<b>Pámetro (Par)</b>	Permite modificar externamente los valores mínimos y máximos de duración del impulso y pausa. Si no se utiliza, el bloque trabaja con los parámetros establecidos internamente en su configuración.
<b>Salida (Q)</b>	Señal de salida del generador aleatorio. Se activa y desactiva con tiempos impredecibles, seleccionados dentro de los

rangos establecidos para los impulsos y pausas mientras la entrada esté habilitada.

## Párametros



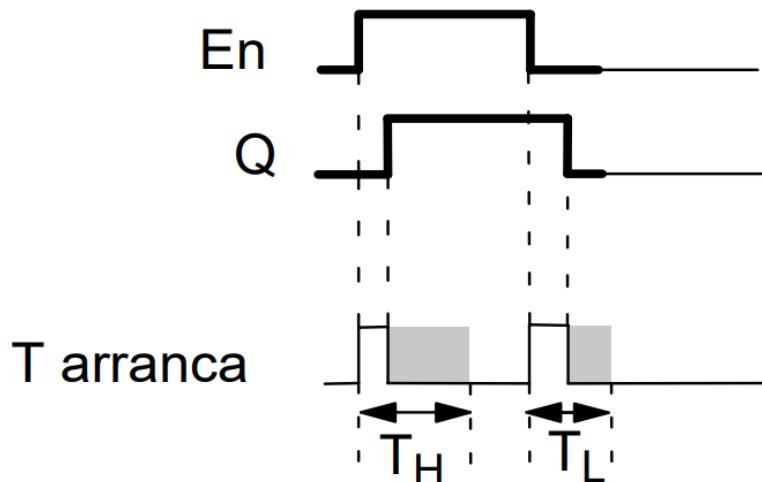
### Máx. retardo de conexión (TH)

Define el tiempo máximo que la salida puede permanecer encendida dentro del ciclo aleatorio.

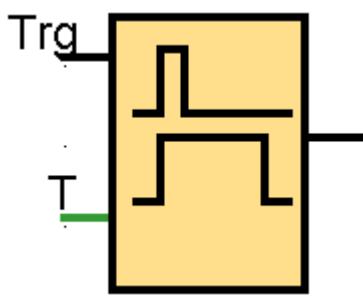
### Máx. retardo de desconexión (TL)

Define el tiempo máximo que la salida puede permanecer apagada entre un impulso y el siguiente.

## Cronograma



## **Interruptor de alumbrado para escalera**



El Interruptor de alumbrado para escalera es un temporizador diseñado para controlar la iluminación durante un tiempo limitado después de haber sido activado.

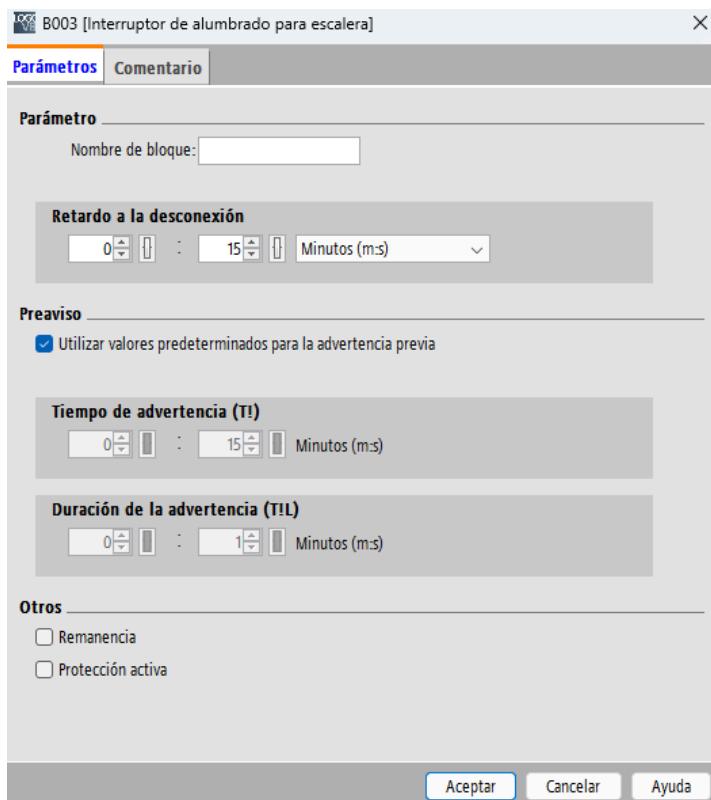
Al recibir un pulso en la entrada (por ejemplo, desde un pulsador), el bloque enciende la salida y mantiene la luz encendida durante el tiempo programado. Una vez transcurrido ese tiempo, la salida se apaga automáticamente, aun si no se vuelve a presionar el botón.

Este bloque permite encender y apagar luces desde varios puntos sin enclavamientos ni interruptores físicos de dos o tres vías, lo que lo hace ideal para alumbrado de pasillos, escaleras o zonas de tránsito.

### **Pinout y descripción de conexiones**

<b>Trigger (TRG)</b>	Es la entrada de activación. Cada vez que esta entrada recibe un pulso (generalmente desde un pulsador), el temporizador se activa y enciende la salida Q durante el tiempo programado. Se puede conectar a un botón NA (normal abierto).
<b>Tiempo (T)</b>	Es la entrada de parámetro, donde se ajusta el tiempo de iluminación. Dependiendo del módulo puede ser un potenciómetro, selector o ajuste digital. Determina cuánto tiempo quedará activa la salida después del disparo.
<b>Salida (Q)</b>	Es la salida de control. Mientras el tiempo esté corriendo, Q estará activa (encendiendo el foco, contactor o relé). Cuando el tiempo expira, Q se desactiva automáticamente apagando la luz.

## Párametros y preaviso



### Retardo a la desconexión

Define el tiempo total que la salida permanece encendida después de recibir un pulso en la entrada de disparo.

Una vez que este tiempo termina, la salida se apaga automáticamente.

Este valor determina el periodo de iluminación disponible para el usuario al accionar el pulsador.

### Tiempo de advertencia

Establece el instante previo al apagado en el que comenzará el aviso al usuario.

El bloque inicia señales de advertencia (como parpadeos o pulsos) cuando el tiempo de funcionamiento está por

terminar.

Sirve para que el usuario se dé cuenta de que la luz está por apagarse.

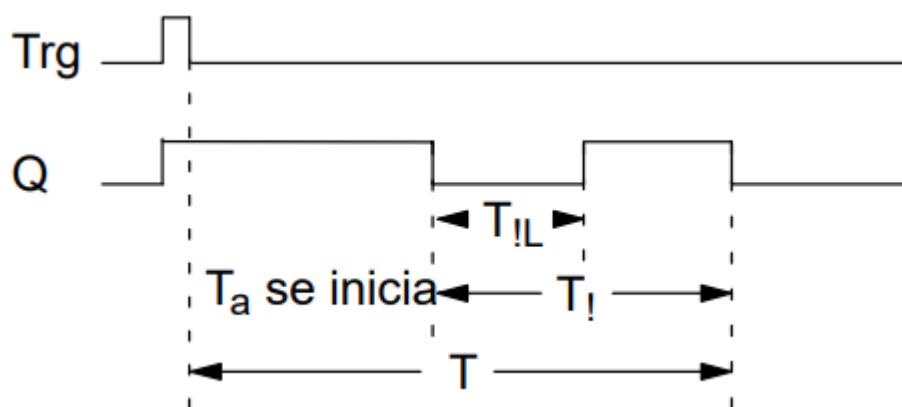
### Duración de la advertencia

Indica cuánto dura visualmente el aviso antes del apagado definitivo.

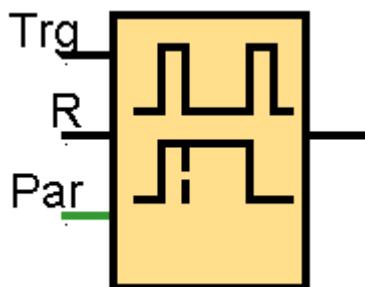
Durante este periodo la salida continúa activa, pero con señal de advertencia (parpadeo o destello).

Al finalizar este tiempo, la salida se desactiva completamente.

### Cronograma



## **Interruptor bifuncional**



El interruptor bifuncional es un bloque que combina dos modos de operación: función de pulsador y función de interruptor. Permite controlar una salida de dos maneras distintas:

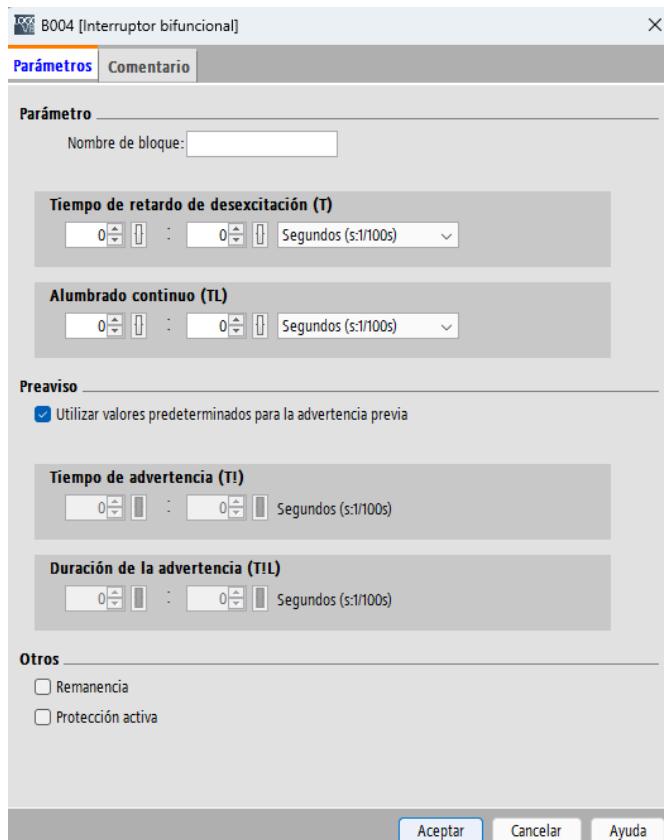
- Al recibir un pulso corto, la salida puede alternar su estado (encendido/apagado).
- Al mantener la señal activa durante un tiempo determinado, la salida puede funcionar como un interruptor temporizado, apagándose automáticamente después del tiempo programado.

Este bloque es útil cuando se desea que una misma señal de entrada funcione tanto para activar manualmente una carga, como para mantenerla encendida por un tiempo sin necesidad de otro comando. Ofrece mayor flexibilidad en aplicaciones de iluminación, control de motores o maniobras donde se requieran ambas funciones en un mismo control.

## **Pinout y descripción de conexiones**

<b>Entrada (TRG)</b>	Una señal en la entrada Trg (Trigger) activa la salida Q (iluminación permanente) o desactiva Q con retardo a la desconexión. Si la salida Q está activada, se puede volver a poner a cero con una señal en Trg.
<b>Entrada (R)</b>	Una señal en la entrada R desactiva el tiempo actual Ta y la salida.
<b>Pámetro (Par)</b>	Configuraciones internas del bloque
<b>Salida (Q)</b>	La salida Q se activa con una señal en la entrada Trg y se vuelve a desactivar al cabo de un tiempo configurable y en función del ancho de impulso en Trg, o bien se desactiva por medio de una nueva señal en Trg.

## Párametros



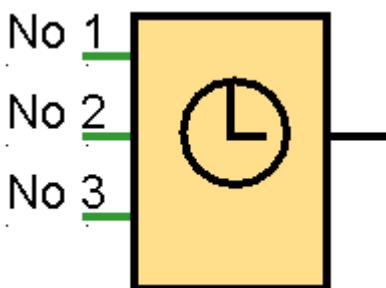
$T$  : determina el tiempo de retardo a la desconexión. la salida se desactiva (transición de 1 a 0) cuando expira el tiempo  $T$ .

$T_L$  : determina el período durante el que debe estar activada la entrada para habilitar la función de iluminación permanente.

$T_{!L}$  : determina el retardo a la conexión del tiempo de preaviso.

$T_{!L}$  : determina la duración del tiempo de preaviso.

Remanencia activada = el estado se guarda de forma remanente.



El temporizador semanal es un bloque de programación horaria que permite activar o desactivar una salida según días específicos de la semana y horarios definidos por el usuario. Con este bloque es posible automatizar encendidos y apagados sin intervención manual, siguiendo un calendario recurrente de 7 días.

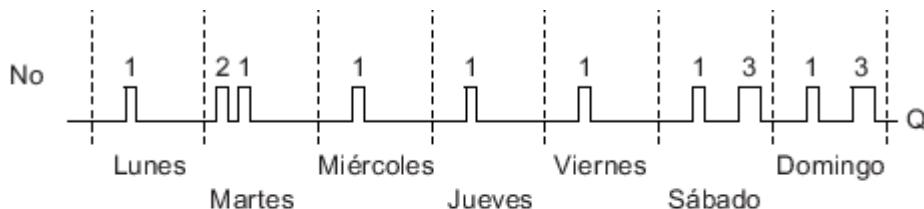
Permite seleccionar qué días operará, establecer horas de inicio y fin, e incluso programar múltiples intervalos dentro de la misma jornada. Es ideal para control de iluminación, sistemas de riego, ventilación, alarmas, y cualquier automatismo que requiera funcionamiento cíclico basado en fecha y hora.

## Pinout y descripción de conexiones

No 1	Leva No 1
No 2	Leva No 2

No 3	Leva No 3
Salida (Q)	Q se activa cuando se acciona la leva configurada.

## Cronograma



No 1:	Diariamente:	06:30 a 8:00 horas
No 2:	Martes:	03:10 a 04:15 horas
No 3:	Sábado y domingo:	16:30 a 23:10 horas

## Parámetros

B005 [Temporizador semanal]

**General** Comentario

Parámetro \_\_\_\_\_

Nombre de bloque: \_\_\_\_\_

B005 [Temporizador semanal]

**General** Comentario

Parámetro \_\_\_\_\_

Nombre de bloque: \_\_\_\_\_

**Leva 1**

Lunes     Martes     Miércoles     Jueves  
 Viernes     Sábado     Domingo

Momento de conexión:  :  Inactiva  
Momento de desconexión:  :  Inactiva

**Leva 2**

Lunes     Martes     Miércoles     Jueves  
 Viernes     Sábado     Domingo

Momento de conexión:  :  Inactiva  
Momento de desconexión:  :  Inactiva

**Leva 3**

Lunes     Martes     Miércoles     Jueves  
 Viernes     Sábado     Domingo

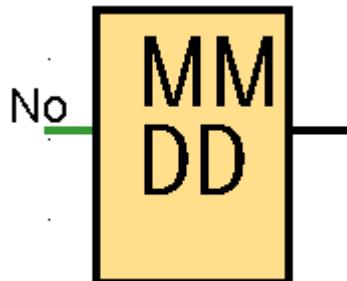
Momento de conexión:  :  Inactiva  
Momento de desconexión:  :  Inactiva

**Otros**

Protección activa

A través de los parámetros No1, No2, No3 se ajustan los momentos de conexión y desconexión de cada una de las levas ("Nocken" en alemán) del temporizador semanal. Para cada leva se especifican el día de la semana y la hora de conexión y desconexión

## **Temporizador anual**



El temporizador anual es un bloque de programación de calendario que permite activar o desactivar una salida en fechas específicas dentro del año. A diferencia del temporizador semanal, este bloque no trabaja por días repetitivos, sino por eventos puntuales basados en día, mes y hora.

Se pueden programar intervalos únicos o repetitivos cada año, permitiendo automatizar funciones que solo ocurren en fechas determinadas, como encendido de iluminación en temporadas, activación de sistemas especiales, recordatorios, o funcionamiento condicionado en períodos festivos o de mantenimiento.

Este bloque elimina la necesidad de intervención manual, garantizando que la salida actúe únicamente en el periodo definido y permanezca inactiva el resto del año.

## **Pinout y diagrama de conexión**

<b>No</b>	En el parámetro No (leva) se configuran el modo del temporizador, los momentos de conexión y desconexión del mismo, así como si la salida es una salida de impulso.
<b>Salida (Q)</b>	Q se activa si está conectada alguna de las levas configuradas.

## **Párametros**

B006 [Temporizador anual]

Parámetros Comentario

Parámetro

Nombre de bloque:

Patrón de reaparición

Anual

Mensual

## **Patrón de reaparición**

Define la forma en que el evento programado volverá a repetirse con el paso del tiempo.

Permite seleccionar si la activación será única o si el evento se repetirá periódicamente cada cierto número de años.

### ***Intervalo de reaparición (año)***

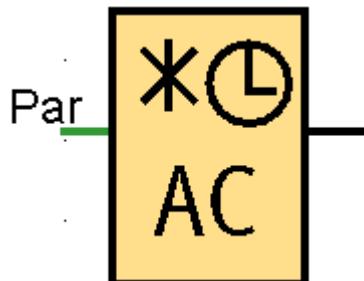
Especifica cada cuántos años volverá a ejecutarse el evento programado.

### ***Fecha***

Indica el día y el mes en que ocurrirá la activación o desactivación de la salida.

La acción programada solo se ejecutará cuando el reloj interno del sistema coincida con la fecha configurada.

### ***Reloj astronómico***

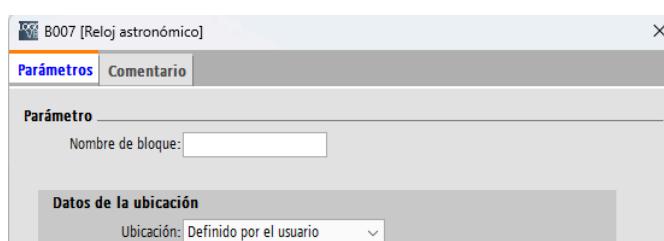


La función especial "Reloj astronómico" se utiliza para activar una salida entre la salida y la puesta del sol con base en la hora local en la ubicación geográfica de los dispositivos 0BA7 o 0BA8. El estado de la salida de este bloque de función también depende de la configuración del cambio de horario de verano/invierno.

### ***Pinout y diagrama de conexión***

<b>Pámetro (Par)</b>	Los datos de la ubicación incluyendo la longitud, la latitud, la zona horaria y el desfase entre el amanecer (TR) y el anochecer (TS).
<b>Salida (Q)</b>	Q se pone en "hi" cuando se alcanza la hora de salida del sol y conserva este estado hasta alcanzar la hora de puesta del sol.

### ***Pámetros***



### **Datos de la ubicación:**

Son los parámetros sobre el lugar utilizado para la señal, en el cual esta ubicación, longitud, latitud, huso horario, nombre.

se puede seleccionar la ubicación del dispositivo LOGO! entre una de las siguientes zonas horarias predefinidas:

Pekín, Berlín, Londres, Roma, Moscú, Tokio, Washington, Ankara, Madrid, Ámsterdam

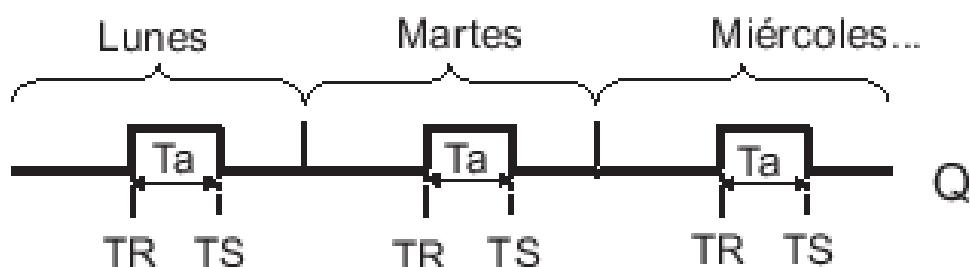
Si elige una de estas ubicaciones, LOGO!Soft Comfort aplica la latitud, longitud y la zona horaria correspondiente a la selección.

Otra posibilidad es configurar la latitud, longitud y la zona horaria específica de su ubicación y asignarle un nombre personalizado.

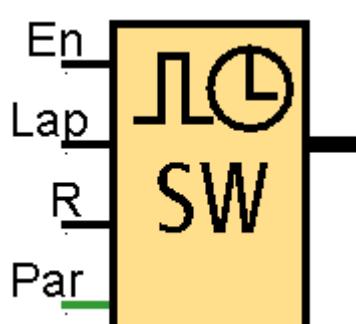
### **Diferencia horaria**

Es el parámetro ajustable sobre la diferencia horaria respecto al anochecer o amanecer

### **Cronograma**



### **Cronometro**



El bloque Cronómetro permite medir y acumular el tiempo durante el cual una señal de entrada permanece activa. Cada vez que la entrada se energiza, el cronómetro inicia la medición y continúa contando mientras la señal siga presente. Al desactivarse la entrada, el conteo se detiene y el valor queda almacenado hasta que se reinicie o se active la función de puesta a cero.

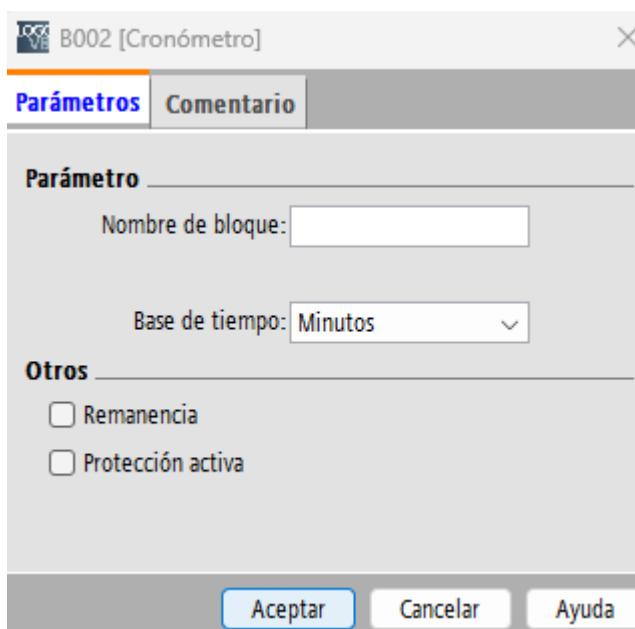
Este bloque es útil para registrar tiempos de operación de máquinas, horas de uso, tiempos productivos, mantenimiento preventivo o cualquier aplicación donde se requiera llevar control de la duración de una señal o proceso. El valor contado puede visualizarse y utilizarse dentro de la lógica del programa, permitiendo tomar decisiones basadas en el tiempo acumulado.

### ***Pinout y descripción de conexión***

<b><i>Entrada (En)</i></b>	En (Enable) es la entrada de vigilancia. LOGO! pone a 0 el tiempo transcurrido actualmente y comienza a contabilizar el tiempo transcurrido tan pronto En pasa de 0 a 1. Cuando En pasa de 1 a 0, el tiempo transcurrido se congela.
<b><i>Entrada (Lap)</i></b>	Un flanco ascendente (cambio de 0 a 1) en la entrada Lap detiene el cronómetro y aplica en la salida el tiempo de pausa. Un flanco descendente (cambio de 1 a 0) en la entrada Lap reanuda el cronómetro y aplica en la salida el tiempo transcurrido actualmente.
<b><i>Entrada (R)</i></b>	Una señal en la entrada R (Reset) borra el tiempo transcurrido actualmente y el tiempo de pausa.
<b><i>Parámetro (Par)</i></b>	Base para el tiempo transcurrido, se pueden ajustar horas, minutos, segundos o centésimas de segundo.
<b><i>Salida (AQ)</i></b>	La salida AQ emite el valor del tiempo transcurrido actualmente cuando se registra un flanco descendente (cambio de 1 a 0) en la entrada Lap y emite el valor del tiempo de pausa cuando se registra un flanco ascendente (cambio de 0 a 1) en la entrada Lap.

	Un flanco ascendente (cambio de 0 a 1) pone a 0 el valor de la salida AQ.
--	---

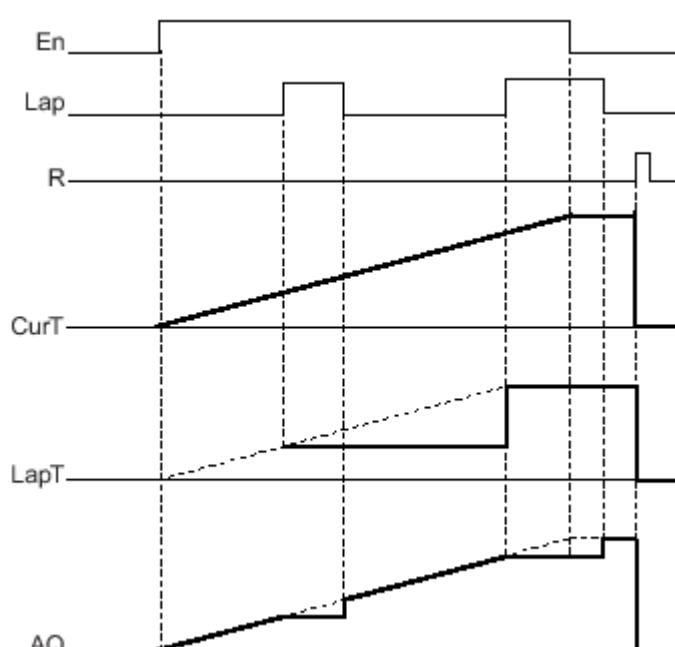
## Párametros



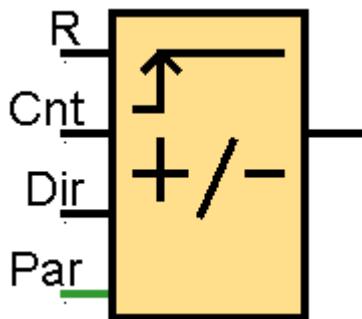
## Base de tiempo:

Permite establecer la resolución del tiempo

## Cronograma



## **Contador adelante/atrás**



El bloque contador adelante/atrás permite incrementar o decrementar un valor interno de conteo en función de los impulsos recibidos. En cada flanco de la señal de entrada, el contador suma o resta una unidad según el estado de la entrada Dir: cuando esta está activa, el conteo avanza (incrementa); cuando está inactiva, el conteo retrocede (decrementa).

El bloque además compara continuamente el valor actual del contador con un umbral configurado por el usuario. Cuando dicho valor es alcanzado o superado en el sentido de conteo correspondiente, la salida del bloque cambia de estado (se activa o se desactiva), permitiendo generar una señal de control basada en el valor contado.

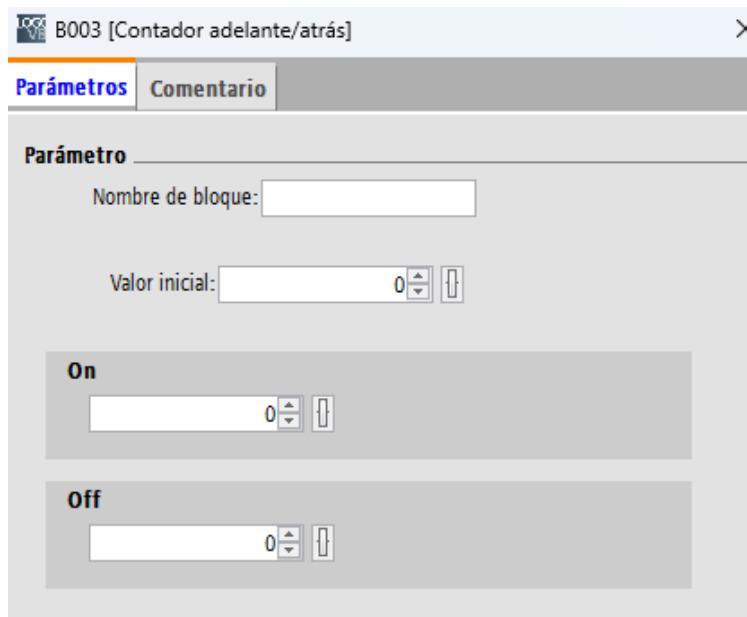
Este tipo de bloque es útil para aplicaciones donde se requiera supervisar secuencias cíclicas, conteos de objetos, movimientos de actuadores o cualquier sistema que necesite conocer la cantidad de eventos producidos con capacidad de avance y retroceso en el conteo.

### **Pinout y descripción de conexión**

<b>Entrada (R)</b>	Con una señal en la entrada R (Reset), el valor de conteo interno y la salida se ajustan al valor inicial (StartVal).
--------------------	---

<b>Entrada (Cnt)</b>	<p>La función cuenta en la entrada Cnt los cambios de estado de 0 a 1. Los cambios de estado de 1 a 0 no se cuentan.</p> <ul style="list-style-type: none"> <li>• Utilice las entradas I3, I4, I5 e I6 para contadores rápidos (LOGO! 12/24RCE/RCEo, LOGO! 24CE/24CEO): máx. 5 kHz, si la entrada rápida está conectada directamente al bloque de función contador adelante/atrás</li> <li>• Utilice cualquier otra entrada o un elemento del circuito para contajes lentos (típ. 4 Hz).</li> </ul>
<b>Entrada (Dir)</b>	<p>La entrada Dir (Direction) determina el sentido de conteo:</p> <p>Dir = 0: adelante Dir = 1: atrás</p>
<b>Parámetro (Par)</b>	<p>On: umbral de conexión / Rango de valores: 0 a 999999 Off: umbral de desconexión / Rango de valores 0 a 999999 Valor inicial: valor inicial a partir del cual se cuenta adelante o atrás. Remanencia activada = el estado se guarda de forma remanente.</p>
<b>Salida (Q)</b>	<p>Q se activa o desactiva en función del valor real Cnt y de los umbrales ajustados.</p>

## Párametros



### On:

valor en el que la salida se activará. Puede configurarse entre 0 y 999999.

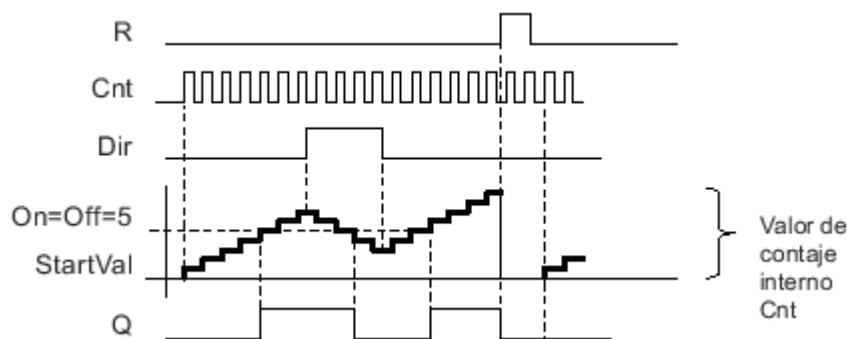
### Off:

valor en el que la salida se desactivará. También admite valores desde 0 hasta 999999.

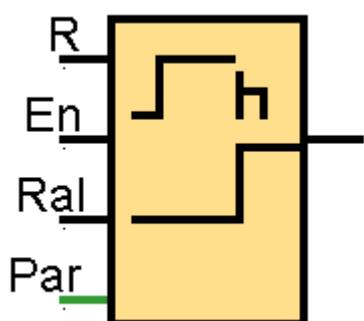
### **Valor inicial:**

número desde el cual inicia el conteo, ya sea ascendente o descendente.

### **Cronograma**



### **Contador de horas de funcionamiento**



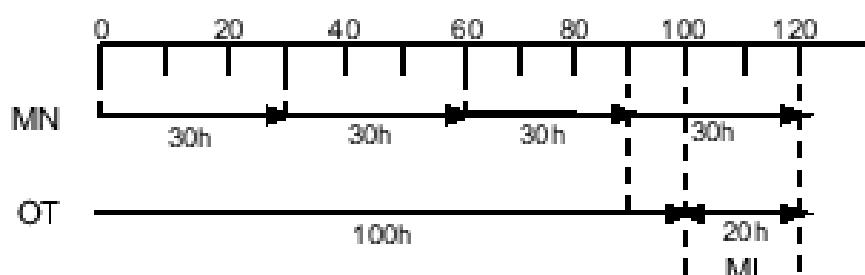
El contador de horas de funcionamiento registra el tiempo total durante el cual un equipo o proceso permanece en operación. Cada vez que la entrada de habilitación está activa, el bloque comienza a acumular tiempo de trabajo y lo almacena de forma interna. Este valor puede visualizarse, compararse con un umbral o utilizarse para generar acciones de mantenimiento preventivo.

Su función principal es llevar un registro confiable de las horas reales de uso, permitiendo programar servicios, detectar desgastes, o gestionar ciclos de operación en maquinaria e instalaciones industriales.

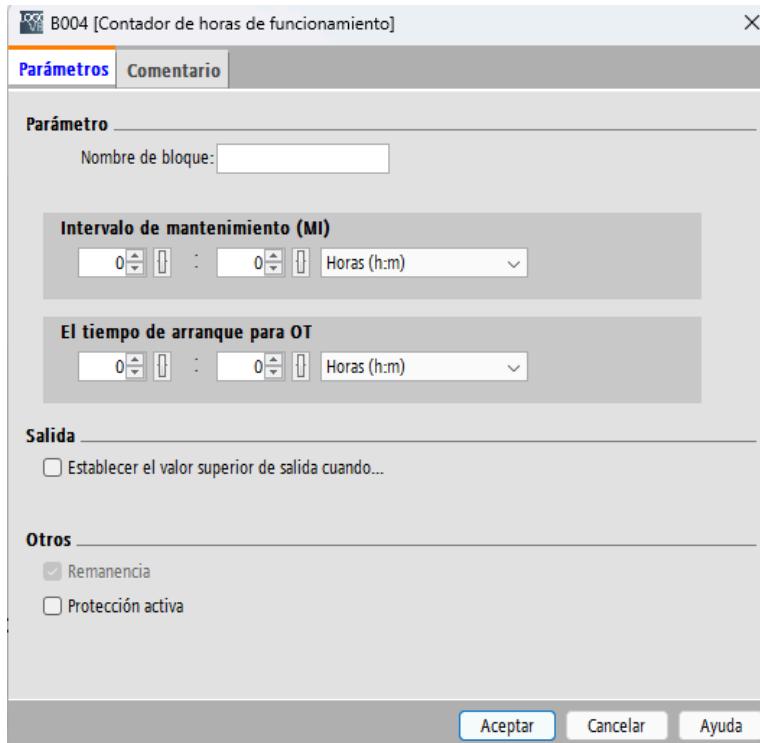
### **Pinout y descripción de conexión**

<b>Entrada (R)</b>	Un flanco ascendente (cambio de 0 a 1) en la entrada R desactiva la salida Q y pone el contador al valor configurado MI durante el tiempo restante (MN).
<b>Entrada (En)</b>	En es la entrada de vigilancia. LOGO! mide el tiempo durante el que está activada esta entrada.
<b>Entrada (Ral)</b>	Un flanco ascendente en la entrada Ral (Reset all) desactiva el contador de horas de funcionamiento (OT) y la salida. Además, el valor del tiempo restante (MN) se ajusta al intervalo de mantenimiento configurado (MI): <ul style="list-style-type: none"> <li>• Salida Q = 0</li> <li>• Horas de funcionamiento medidas OT = 0</li> <li>• Tiempo restante del intervalo de mantenimiento MN = MI</li> </ul>
<b>Parámetro (Par)</b>	Configuración interna del bloque
<b>Salida (Q)</b>	Cuando el tiempo restante MN = 0 se activa la salida. La salida se desactiva: <ul style="list-style-type: none"> <li>• Cuando "Q → 0:R+En" si R = 1 o Ral = 1 o En = 0</li> <li>• Cuando "Q → 0:R" si R = 1 o Ral = 1.</li> </ul>

### Cronograma



## Párametros



Intervalo de mantenimiento (MI):

Intervalo de mantenimiento que debe especificarse en unidades de horas y minutos

Rango de valores: 0000 h a 9999 h, 0 m a 59 m

El tiempo de arranque para OT:

Tiempo de funcionamiento total acumulado. Se puede determinar un tiempo de inicio de offset en horas y minutos.

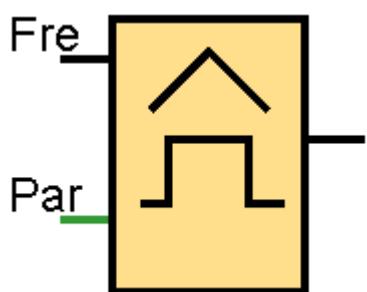
minutos.

Rango de valores: 00000 h a 99999 h, 0 m a 59 m

$Q \rightarrow 0$ :

- Cuando "R" está seleccionado:  
 $Q = 1$  si  $MN = 0$ ;  
 $Q = 0$  si  $R = 1$  o  $Ral = 1$
- Cuando "R+En" está seleccionado:  
 $Q = 1$  si  $MN = 0$ ;  
 $Q = 0$  si  $R = 1$  o  $Ral = 1$  o  $En = 0$ .

## **Selector de umbral**

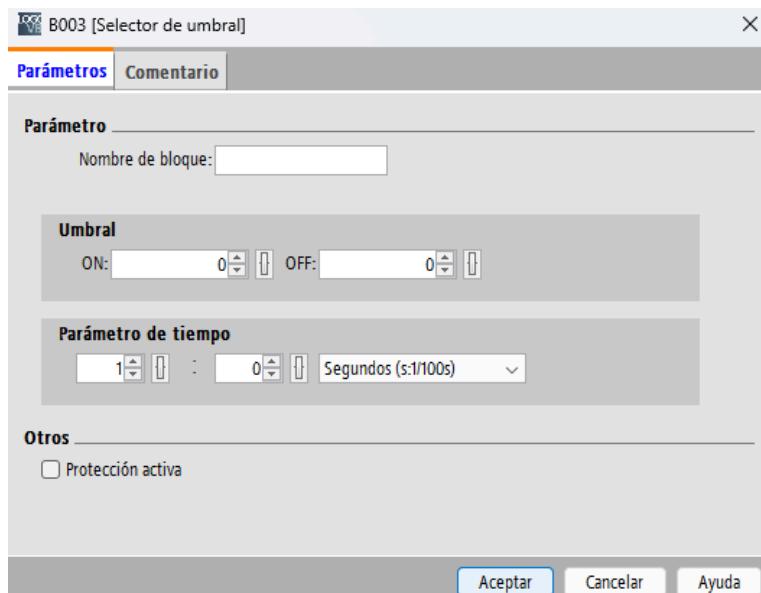


El bloque selector de umbral compara una señal de entrada analógica con un valor límite configurado. Cuando el valor de entrada supera o desciende por debajo del umbral, la salida cambia de estado (se activa o se desactiva) según la condición establecida. Este bloque permite generar una señal digital a partir de una magnitud analógica, facilitando la detección de niveles, alarmas o condiciones de control basadas en valores medidos.

## **Pinout y descripción de conexión**

<b>Entrada (Fre)</b>	<p><i>La función cuenta en la entrada Fre los cambios de estado de 0 a 1. Los cambios de estado de 1 a 0 no se cuentan.</i></p> <ul style="list-style-type: none"><li>• Utilice las entradas I3, I4, I5 e I6 para contadores rápidos (LOGO! 12/24RCE/RCEo, LOGO! 24CE/24CEO): máx. 5 kHz, si la entrada rápida está conectada directamente al bloque de función Selector de umbral</li><li>• Utilice cualquier otra entrada o un elemento del circuito para frecuencias bajas (típ. 4 Hz).</li></ul>
<b>Parámetro (Par)</b>	<p><i>Configuración interna del bloque</i></p>
<b>Salida (Q)</b>	<p><i>Q se activa o desactiva en función de los valores umbral.</i></p>

## Párametros



Umbral On:

umbral de conexión Rango de valores de 0000 a 9999

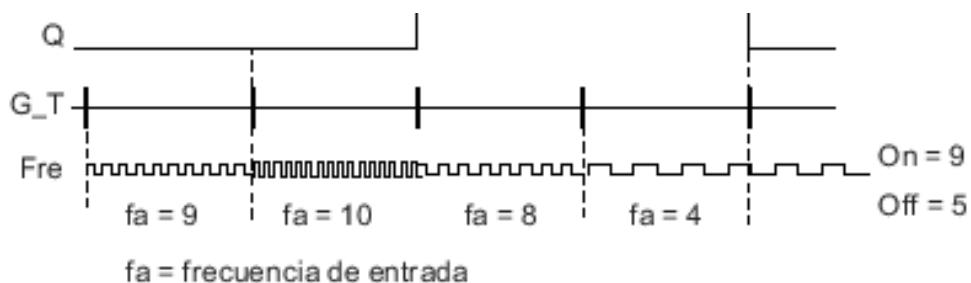
Umbral Off:

umbral de desconexión Rango de valores de 0000 a 9999

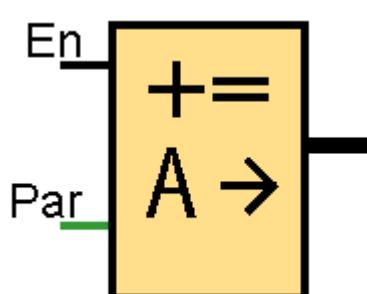
Parámetro de tiempo:

intervalo de tiempo (o "tiempo de puerta") durante el cual se miden los impulsos de entrada. Rango de valores: de 00:00 s a 99:99 s

## Cronometro



## Instrucción aritmética



El bloque de instrucción aritmética realiza operaciones matemáticas entre señales numéricas dentro del programa. Permite sumar, restar, multiplicar o dividir valores provenientes de entradas analógicas, constantes o resultados de otros bloques.

El valor resultante se entrega como salida para su uso en controles, comparaciones o cálculos posteriores.

Este bloque es útil cuando se requiere procesar magnitudes, convertir unidades, aplicar factores de escala o generar valores derivados de mediciones.

### **Pinout y descripción de conexión**

<b>Entrada (En)</b>	Un flanco ascendente en la entrada En habilita el bloque de función "Instrucción aritmética"
<b>Parámetro (Par)</b>	Ajustes internos de los parámetros del bloque
<b>Salida (AQ)</b>	La salida AQ es el resultado de la ecuación de valores de operandos y operadores. AQ se pone a 32767 si ocurre una división por 0 o un rebase por exceso, o bien a -32768 si ocurre un rebase por defecto.

### **Párametros**

**Parámetros**

**Parámetro**  
Nombre del bloque:

**Instrucción**

V1 <input type="text"/> 0 <input type="button"/> Operador 1: + Prioridad 1: H
V2 <input type="text"/> 0 <input type="button"/> Operador 2: + Prioridad 2: M
V3 <input type="text"/> 0 <input type="button"/> Operador 3: + Prioridad 3: L
V4 <input type="text"/> 0 <input type="button"/>

**Posiciones decimales**  
Decimales en el texto de aviso:  0  +12345

**Salida**  
Si En="0", la salida es:  
 0       Último valor

**Otros**  
 Protección activa

**V1:** valor 1: primer operando

**V2:** valor 2: segundo operando

**V3:** valor 3: tercer operando

**V4:** valor 4: cuarto operando

Rango de valores: de -32768 a 32767

Operador 1: primer operador

Operador 2: segundo operador

Operador 3: tercer operador

Prioridad 1: prioridad de la primera operación

Prioridad 2: prioridad de la segunda operación

Prioridad 3: prioridad de la tercera operación

p: número de decimales

Ajustes posibles: 0, 1, 2, 3

### **Posiciones decimales:**

Resolución de decimales mostrada en el aviso

### **Salida:**

Permite seleccionar si cuando está en 0 la salida muestre 0 o el último valor

### ***Errores posibles: división por cero y desbordamiento***

Si la ejecución del bloque de función de la instrucción aritmética resulta en una división por cero o un desbordamiento, se activan bits internos que indican el tipo de error que ha ocurrido. Es posible programar un bloque de función de detección de error de la instrucción aritmética para detectar estos errores y controlar la reacción del programa de la manera deseada. Para cada bloque de función de la instrucción aritmética se programa un bloque de función de detección de error de la instrucción aritmética.

### ***Ejemplos***

<i>V1</i>	<i>Operator1 (prioridad 1)</i>	<i>V2</i>	<i>Operator2 (prioridad 2)</i>	<i>V3</i>	<i>Operator3 (prioridad 3)</i>	<i>V4</i>
12	+ ( <i>M</i> )	6	/ ( <i>H</i> )	3	- ( <i>L</i> )	1

*Ecuación: (12 + (6 / 3)) - 1*

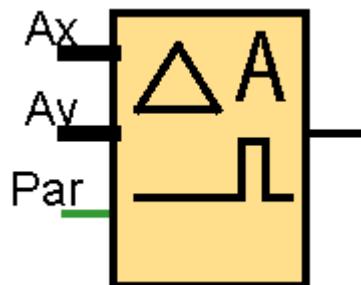
*Resultado: 13*

<i>V1</i>	<i>Operator1</i> <i>(prioridad 1)</i>	<i>V2</i>	<i>Operator2</i> <i>(prioridad 2)</i>	<i>V3</i>	<i>Operator3</i> <i>(prioridad 3)</i>	<i>V4</i>
2	$+ (L)$	3	$* (M)$	1	$+ (H)$	4

*Ecuación:*  $2 + (3 * (1 + 4))$

*Resultado:* 17

## Comparador analógico



El comparador analógico evalúa una señal de entrada y la compara con un valor umbral configurado. Según el resultado de la comparación, el bloque activa o desactiva su salida digital. Puede trabajar con condiciones como mayor que, menor que o igual a un valor determinado.

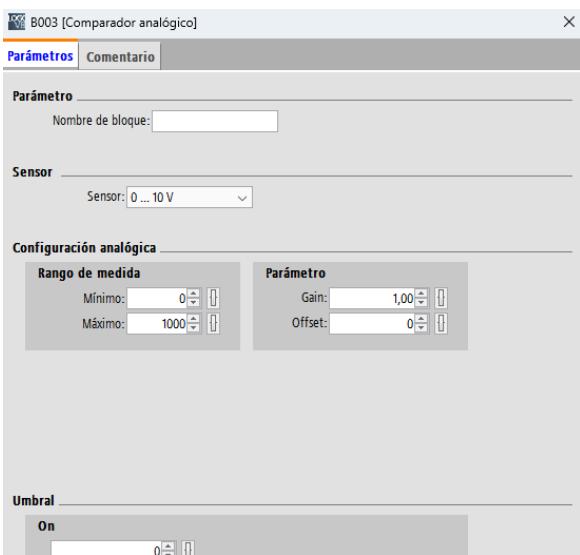
Se utiliza para convertir una medición analógica en una señal lógica, permitiendo detectar niveles, activar alarmas, controlar procesos en función de límites o supervisar variaciones en sensores.

## Párametros

<b>Entradas</b> <i>Ax, Ay</i>	<p><i>La entradas Ax y Ay son dos señales analógicas de las siguientes:</i></p> <ul style="list-style-type: none"><li>• <i>AI1 a AI8 (*)</i></li><li>• <i>AM1 a AM6 (para 0BA6), AM1 a AM16 (para 0BA7) o AM1 a AM64 (para 0BA8)</i></li><li>• <i>NAI1 a NAI32 (para 0BA7 o 0BA8)</i></li><li>• <i>AQ1 y AQ2 (para 0BA7) o AQ1 a AQ8 (para 0BA8)</i></li><li>• <i>NAQ1 a NAQ16 (para 0BA7 o 0BA8)</i></li><li>• <i>El número de bloque de una función con salida analógica</i></li></ul>
<b>Parámetro</b>	<p><i>Configuracion interna de los parámetros del bloque</i></p>
<b>Salida Q</b>	<p><i>Q se activa o desactiva en función de los valores umbral ajustados.</i></p>

\* *AI1 a AI8: 0 a 10 V es proporcional a 0 a 1000 (valor interno).*

## Párametros



### Sensores:

Selección de los valores de salida de los sensores

### Ganancia:

Rango de valores: de -10,00 a 10,00

**Offset:**

Rango de valores: de -10000 a 10000

**On:**

Umbral de conexión

Rango de valores: de -20000 a 20000

**Off:**

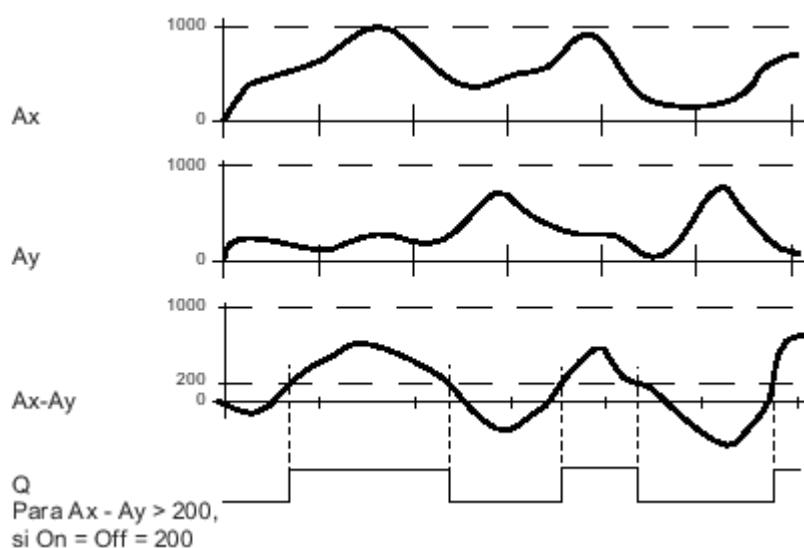
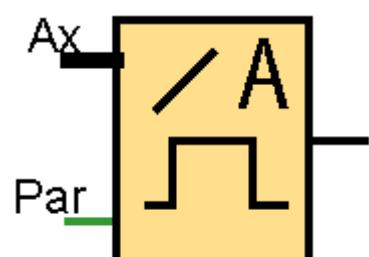
Umbral de desconexión

Rango de valores: de -20000 a 20000

**p:**

número de decimales

Rango de valores: 0, 1, 2, 3

**Cronograma****Conmutador analógico de valor umbral**

El conmutador analógico de valor umbral compara una señal de entrada con un valor límite establecido. Cuando la señal analógica alcanza o supera el umbral configurado, el bloque conmuta su salida y mantiene ese estado hasta que la señal vuelve a caer por debajo del límite definido.

Este bloque permite generar una señal digital a partir de una magnitud analógica, siendo útil para activar alarmas, indicadores o acciones de control cuando una variable supera un nivel determinado.

### ***Pinout y descripción de conexión***

<b><i>Entrada (Ax)</i></b>	La entrada Ax es una de las siguientes señales analógicas: <ul style="list-style-type: none"><li>• AI1 a AI8 (*)</li><li>• AM1 a AM6 (para 0BA6), AM1 a AM16 (para 0BA7) o AM1 a AM64 (para 0BA8)</li><li>• NAI1 a NAI32 (para 0BA7 o 0BA8)</li><li>• AQ1 y AQ2 (para 0BA7) o AQ1 a AQ8 (para 0BA8)</li><li>• NAQ1 a NAQ16 (para 0BA7 o 0BA8)</li><li>• El número de bloque de una función con salida analógica</li></ul>
<b><i>Parámetro (Par)</i></b>	Configuración de parámetros internos del bloque

<b>Salida (Q)</b>	Q se activa o desactiva en función de los valores umbral ajustados.
-------------------	---

\* AI1 a AI8: 0 a 10 V es proporcional a 0 a 1000 (valor interno).

## Párametros

B004 [Comutador analógico de valor umbral]

**Parámetros** Comentario

**Parámetro**

Nombre de bloque:

**Sensor**

Sensor: 0 ... 10 V

**Configuración analógica**

<b>Rango de medida</b>	<b>Parámetro</b>
Mínimo: 0 <input type="button" value="▲"/> <input type="button" value="▼"/>	Gain: 1,00 <input type="button" value="▲"/> <input type="button" value="▼"/>
Máximo: 1000 <input type="button" value="▲"/> <input type="button" value="▼"/>	Offset: 0 <input type="button" value="▲"/> <input type="button" value="▼"/>

**Valor umbral**

<b>ON</b>	0 <input type="button" value="▲"/> <input type="button" value="▼"/>
<b>OFF</b>	0 <input type="button" value="▲"/> <input type="button" value="▼"/>

**Posiciones decimales**

Decimales en el texto de aviso: 0   +12345

**Otros**

Protección activa

Aceptar Cancelar Ayuda

### Sensores:

Selección de los valores de salida de los sensores

### Ganancia:

Rango de valores: de -10,00 a 10,00

### Offset:

Rango de valores: de -10000 a 10000

### On:

Umbral de conexión

Rango de valores: de -20000 a 20000

### Off:

Umbral de desconexión

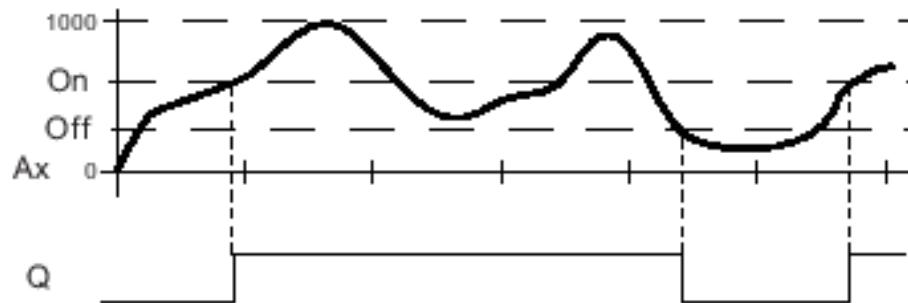
Rango de valores: de -20000 a 20000

p:

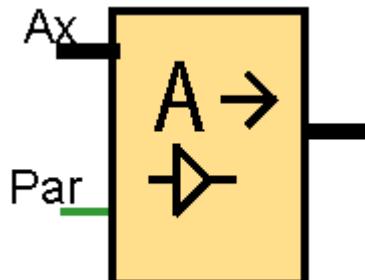
número de decimales

Rango de valores: 0, 1, 2, 3

## Cronograma



## Amplificador analógico



El bloque Amplificador Analógico tiene la función de aumentar la amplitud de una señal analógica de entrada sin alterar su forma original. Opera tomando una señal débil y entregando una versión amplificada que conserva las características esenciales como frecuencia y forma de onda.

Este bloque suele emplearse cuando es necesario acondicionar señales provenientes de sensores, transductores u otros dispositivos que entregan niveles de voltaje muy bajos. El amplificador proporciona ganancia ajustable y puede configurarse para operar con diferentes rangos de entrada y salida, manteniendo la linealidad y reduciendo el ruido.

En aplicaciones de control y automatización, este bloque se utiliza para asegurar que la señal alcance niveles adecuados para su procesamiento, comparación o conversión posterior.

## Pinout y descripción de conexión

<b>Entrada (Ax)</b>	La entrada Ax es una de las siguientes señales analógicas:
	<ul style="list-style-type: none"> <li>• AI1 a AI8 (*)</li> <li>• AM1 a AM6 (para 0BA6), AM1 a AM16 (para 0BA7) o AM1 a AM64 (para 0BA8)</li> <li>• NAI1 a NAI32 (para 0BA7 o 0BA8)</li> <li>• AQ1 y AQ2 (para 0BA7) o AQ1 a AQ8 (para 0BA8)</li> <li>• NAQ1 a NAQ16 (para 0BA7 o 0BA8)</li> <li>• El número de bloque de una función con salida analógica</li> </ul>
<b>Parámetro (Par)</b>	Configuración de parámetros interna del bloque
<b>Salida (AQ)</b>	Rango de valores para AQ: -32768 a +32767

\* *AI1 a AI8: 0 a 10 V es proporcional a 0 a 1000 (valor interno).*

## Párametros



## Ganancia

La función lee el valor de una señal analógica aplicada en la entrada analógica Ax.

Este valor se multiplica por el parámetro A (ganancia). El parámetro B (decalaje) se suma al producto del siguiente modo:

$$(Ax * \text{ganancia}) + \text{decalaje} = \text{valor real Ax.}$$

El valor real Ax se devuelve en la salida AQ.

Rango de valores: -10,00 a 10,00

Decalaje

Rango de valores: -10000 a 10000

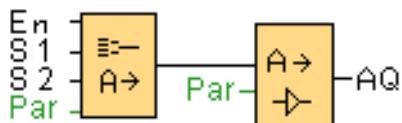
p: número de decimales

Ajustes posibles: 0, 1, 2, 3

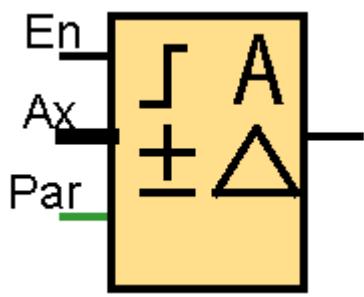
## ***Salida analógica***

Si conecta esta función especial con una salida analógica real, recuerde que la salida analógica solo puede procesar valores comprendidos entre 0 y 1000. Para hacerlo, conecte un amplificador adicional entre la salida analógica de la función especial y la salida analógica real. Con el amplificador se normaliza el rango de salida de la función especial a un rango de valores de 0 a 1000.

Ejemplo: Amplificador adicional detrás de un multiplexor analógico.



## ***Vigilancia del valor analógico***



El bloque Vigilancia del Valor Analógico permite supervisar continuamente el valor de una señal analógica con el fin de detectar variaciones significativas respecto a un valor previamente registrado. Esta función almacena la variable de proceso proveniente de una entrada analógica y genera una salida cuando la señal actual supera o desciende por debajo del valor guardado, considerando un margen u offset configurable por el usuario.

Este bloque es útil en sistemas donde se requiere detectar incrementos o decrementos bruscos, monitorear niveles críticos, o activar alarmas y acciones de control cuando la señal analógica varía más allá de un límite dinámico. Gracias al offset ajustable, el usuario puede definir el rango de tolerancia adecuado para su aplicación.

## ***Pinout y descripción de conexión***

<b>Entrada (En)</b>	Con el flanco ascendente (cambio de 0 a 1) en la entrada de habilitación En se guarda el valor analógico de la entrada Ax ("Aen") y se inicia la vigilancia del rango de valores analógicos Aen +- Delta.
<b>Entrada (Ax)</b>	<p>La entrada Ax es una de las siguientes señales analógicas:</p> <ul style="list-style-type: none"> <li>• AI1 a AI8 (*)</li> <li>• AM1 a AM6 (para 0BA6), AM1 a AM16 (para 0BA7) o AM1 a AM64 (para 0BA8)</li> <li>• NAI1 a NAI32 (para 0BA7 o 0BA8)</li> <li>• AQ1 y AQ2 (para 0BA7) o AQ1 a AQ8 (para 0BA8)</li> <li>• NAQ1 a NAQ16 (para 0BA7 o 0BA8)</li> <li>• El número de bloque de una función con salida analógica</li> </ul>
<b>Parámetro (Par)</b>	<p>Ganancia Rango de valores: de -10,00 a 10,00</p> <p>Offset Rango de valores: de -10000 a 10000</p> <p>Umbral 1: Valor de diferencia superior a Aen: umbral de conexión/desconexión Rango de valores: de 0 a 20000</p> <p>Umbral 2: valor de diferencia inferior a Aen: umbral de conexión/desconexión Rango de valores: de 0 a 20000</p> <p>p: número de decimales Ajustes posibles: 0, 1, 2, 3</p> <p>Remanencia activada = el estado se guarda de forma remanente.</p>

<b>Salida (Q)</b>	Q se activa/desactiva en función del valor analógico guardado y del offset.
<b>* AI1 a AI8: 0 a 10 V es proporcional a 0 a 1000 (valor interno).</b>	

## Párametros

B022 [Vigilancia del valor analógico]

**Parámetros** Comentario

**Parámetro**

Nombre de bloque:

**Sensor**

Sensor: 0 ... 10 V

**Configuración analógica**

Rango de medida

Mínimo: 0	Gain: 1,00
Máximo: 1000	Offset: 0

Umbral

Umbral 1 (superior: +) 0

Umbral 2 (inferior: -) 0

Posiciones decimales

Decimales en el texto de aviso: 0 +12345

Otros

Remanencia

Protección activa

### Ganancia

Rango de valores: de -10,00 a 10,00

### Offset

Rango de valores: de -10000 a 10000

### Umbral 1:

Valor de diferencia superior a Aen: umbral de conexión/desconexión

Rango de valores: de 0 a 20000

### Umbral 2:

valor de diferencia inferior a Aen: umbral de conexión/desconexión

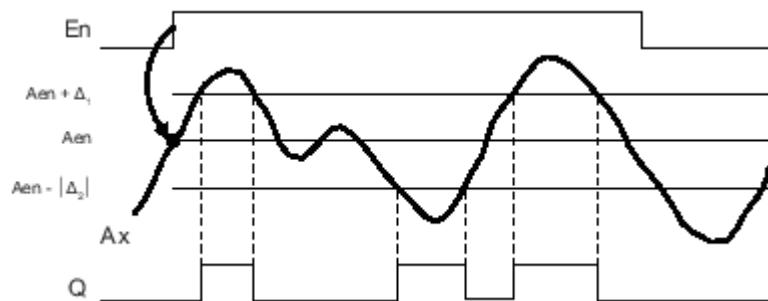
Rango de valores: de 0 a 20000

p:

número de decimales

Ajustes posibles: 0, 1, 2, 3

## Cronograma



## Descripción de la función

Si el estado de la entrada En cambia de 0 a 1, se guarda el valor de la señal en la entrada analógica Ax. Esta variable de proceso guardada se denomina "Aen".

Los valores analógicos reales Ax y Aen se multiplican por el valor del parámetro A (ganancia), respectivamente. El parámetro B (offset) se suma luego al producto del siguiente modo:

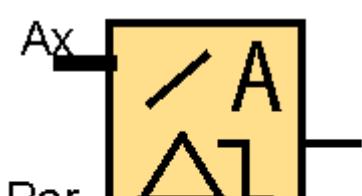
$(Ax * \text{ganancia}) + \text{offset} = \text{valor real Aen}$ , si la entrada En cambia de 0 a 1, o bien

$(Ax * \text{ganancia}) + \text{offset} = \text{valor real Ax}$ .

La salida Q se activa si la entrada En = 1 y el valor real en la entrada Ax se encuentra fuera del rango Aen + Threshold 1 / Aen - Threshold 2.

La salida Q se desactiva si el valor real en la entrada Ax se encuentra dentro del rango Aen + Threshold 1 / Aen - Threshold 2, o bien si la entrada En cambia a cero.

## Conmutador analógico de valor umbral diferencial



El bloque Comutador Analógico de Valor Umbral Diferencial compara continuamente una señal analógica de entrada con dos valores de referencia: un umbral superior y un umbral inferior. La salida del bloque cambia de estado únicamente cuando la señal atraviesa alguno de estos límites, creando así una ventana de operación estable.

A diferencia de un conmutador analógico con un solo umbral, el tipo diferencial evita oscilaciones indeseadas cuando la señal se mantiene cercana al punto de conmutación. Gracias al rango diferencial definido por los dos umbrales, la lógica de salida solo se activa al superar el umbral superior, y no se desactiva hasta que la señal cae por debajo del umbral inferior (o viceversa).

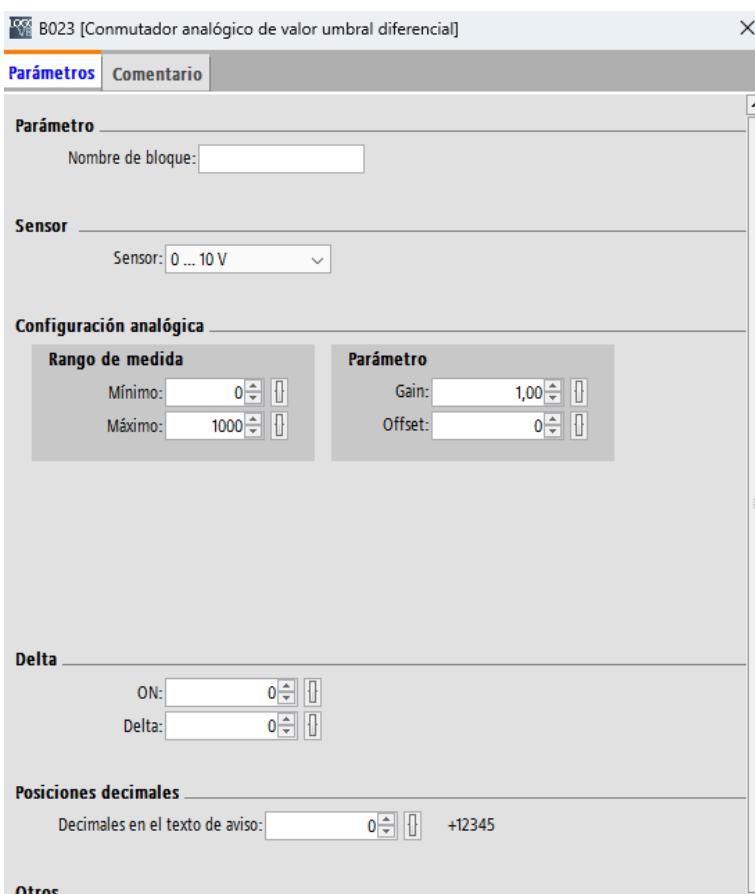
Este comportamiento es útil para sistemas que requieren estabilidad, filtrado de ruido, e histeresis analógica, evitando múltiples cambios de estado por pequeñas variaciones o interferencias en la señal.

### ***Pinout y descripción de conexión***

<b><i>Entrada (Ax)</i></b>	<p>La entrada Ax es una de las siguientes señales analógicas:</p> <ul style="list-style-type: none"><li>• AI1 a AI8 (*)</li><li>• AM1 a AM6 (para 0BA6), AM1 a AM16 (para 0BA7) o AM1 a AM64 (para 0BA8)</li><li>• NAI1 a NAI32 (para 0BA7 o 0BA8)</li><li>• AQ1 y AQ2 (para 0BA7) o AQ1 a AQ8 (para 0BA8)</li><li>• NAQ1 a NAQ16 (para 0BA7 o 0BA8)</li><li>• El número de bloque de una función con salida analógica</li></ul>
----------------------------	--

<b>Parámetro (Par)</b>	<p>Ganancia</p> <p>Rango de valores: -10,00 a 10,00</p> <p>Decalaje</p> <p>Rango de valores: -10000 a 10000</p> <p>On: umbral de conexión/desconexión</p> <p>Rango de valores: -20000 a 20000</p> <p><math>\Delta</math>: valor diferencial para calcular el parámetro Off</p> <p>Rango de valores: -20000 a 20000</p> <p>p: número de decimales</p> <p>Rango de valores: 0, 1, 2, 3</p>
<b>Salida (Q)</b>	Q se activa o desactiva en función del valor umbral y diferencial.
<p><b>* AI1 a AI8: 0 a 10 V es proporcional a 0 a 1000 (valor interno).</b></p>	

## Párametros



Ganancia  
Rango de valores: -10,00 a 10,00

Decalaje  
Rango de valores: -10000 a 10000

On: umbral de conexión/desconexión  
Rango de valores: -20000 a 20000

$\Delta$ : valor diferencial para calcular el parámetro Off  
Rango de valores: -20000 a 20000

p: número de decimales  
Rango de valores: 0, 1, 2, 3

El parámetro p solo es aplicable para visualizar los valores On, Off y Ax en un texto de aviso.

El parámetro p no es válido para la comparación con valores On y Off. (El separador decimal representado se ignora en la comparación.)

## Descripción de la función

La función lee la señal analógica en la entrada Ax.

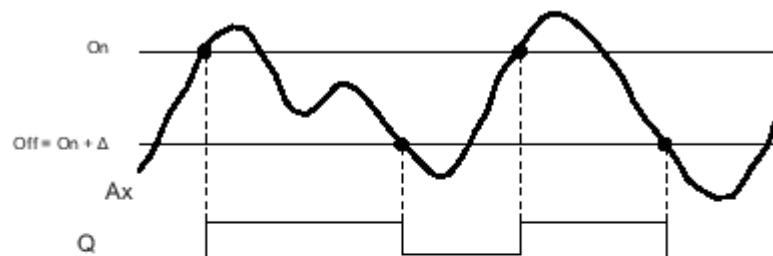
Ax se multiplica por el valor del parámetro A (ganancia). El valor del parámetro B (decalaje) se suma al producto, es decir:

$$(Ax * \text{ganancia}) + \text{decalaje} = \text{valor real Ax.}$$

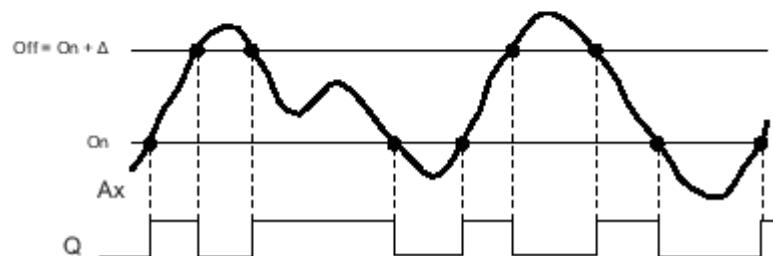
La salida Q se activa o desactiva en función del umbral de conexión (On) ajustado y del valor diferencial (Delta). La función calcula el parámetro Off automáticamente:  $\text{Off} = \text{On} + \Delta$ , pudiendo ser Delta un valor positivo o negativo. Consulte la siguiente regla de cálculo.

## Cronograma

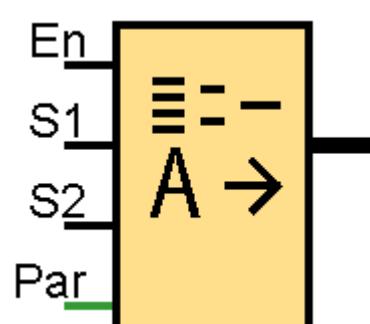
**Cronograma A: función con valor diferencial Delta negativo**



**Cronograma B: función con valor diferencial Delta positivo**



## MUX analógico



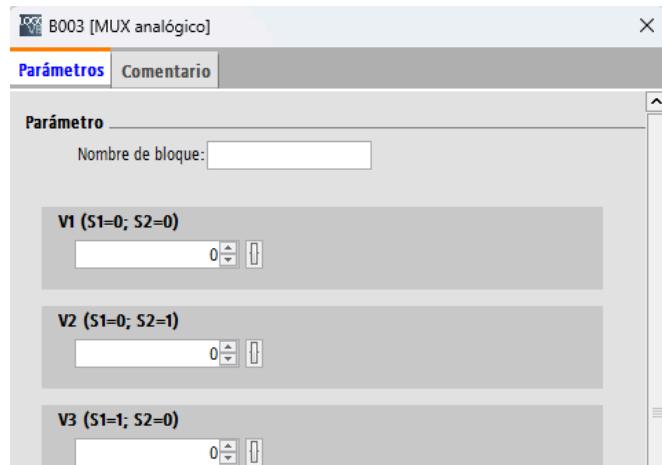
El multiplexor analógico selecciona y envía uno de varios valores analógicos predeterminados según el estado de sus

entradas de control. Cuando el bloque está habilitado, evalúa las condiciones de selección y entrega como salida uno de hasta cuatro valores configurados. Esto permite cambiar dinámicamente la señal analógica en función de diferentes condiciones lógicas dentro del programa.

## **Pinout y descripción de conexión**

<b>Entrada (En)</b>	<i>El estado de señal 1 en la entrada En (Enable) devuelve en la salida AQ un valor analógico parametrizado, en función de S1 y S2.</i>  <i>El estado de señal 0 en la entrada EN pone a 0 la salida AQ.</i>
<b>Entradas (S1 y S2)</b>	<i>S1 y S2 (selectores) para seleccionar el valor analógico que debe devolverse.</i>  <i>S1 = 0 y S2 = 0: se devuelve el valor V1</i> <i>S1 = 0 y S2 = 1: se devuelve el valor V2</i> <i>S1 = 1 y S2 = 0: se devuelve el valor V3</i> <i>S1 = 1 y S2 = 1: se devuelve el valor V4</i>
<b>Parámetro (Par)</b>	<i>Configuración interna de los parámetros del bloque</i>
<b>Salida (AQ)</b>	<i>Salida analógica</i> <i>Rango de valores para AQ: de -32768 a +32767</i>

## **Párametros**



### **V1 a V4:**

Valores analógicos que se devolverán.  
Rango de valores: de -32768 a +32767

p:

número de decimales

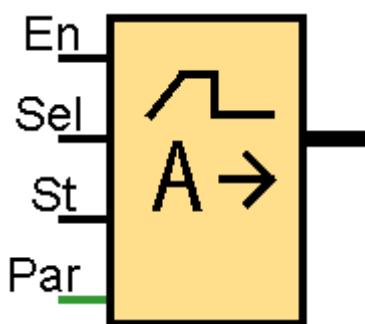
Ajustes posibles: 0, 1, 2, 3

### ***Descripción de la función***

Si se activa la entrada En, la función devuelve en la salida AQ uno de los cuatro posibles valores analógicos V1 a V4 en función de las entradas S1 y S2.

Si la entrada En no se activa, la función devuelve el valor analógico 0 en la salida AQ.

### ***Rampa analógica***



El bloque Rampa Analógica genera una señal analógica que aumenta o disminuye de forma lineal con el tiempo, partiendo de un valor inicial y avanzando hacia un valor final con una pendiente definida. La velocidad de cambio (tiempo de subida o bajada) puede configurarse, permitiendo crear transiciones suaves en procesos donde no es deseable un cambio brusco de la señal.

Este bloque se utiliza comúnmente para:

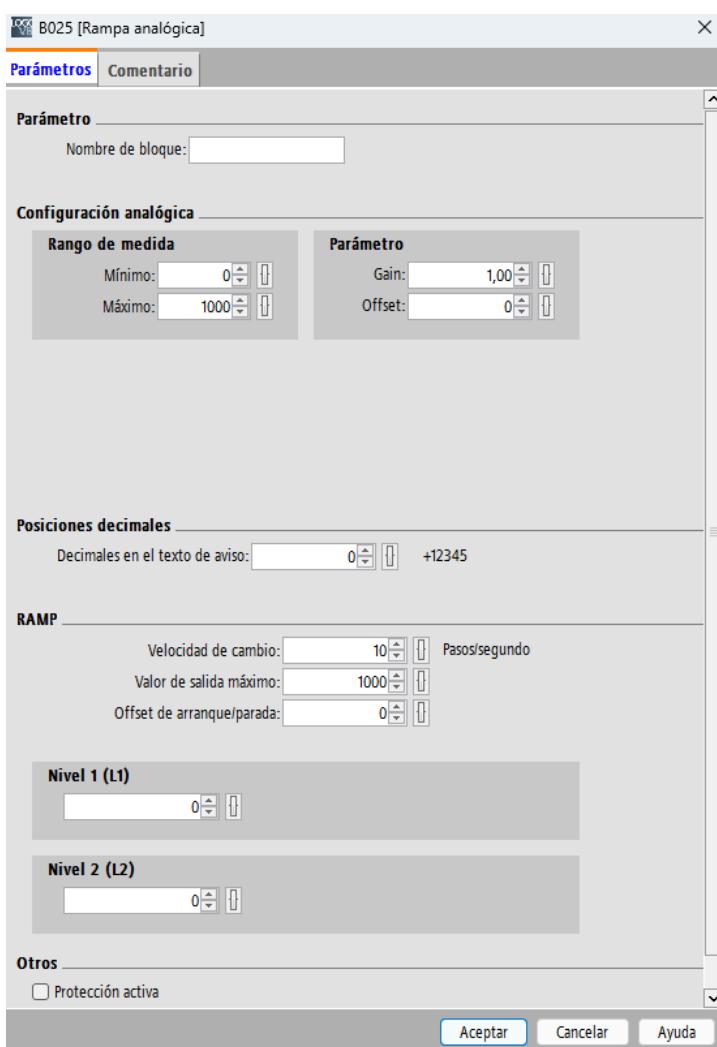
- Realizar arranques y paros graduales en sistemas de control analógico.
- Evitar picos repentinos de corriente, presión o velocidad.
- Simular señales de referencia crecientes o decrecientes.
- Generar perfiles de movimiento o control progresivo en actuadores.

Una vez alcanzado el valor objetivo, la salida permanece estable hasta recibir una nueva orden o reinicio del bloque.

### ***Pinout y descripción de conexión***

<b><i>Entrada (En)</i></b>	<p>Un cambio de estado de 0 a 1 en la entrada En (Enable) activa el nivel de arranque/parada (offset "B" + StSp) en la salida durante 100 ms e inicia la operación de rampa hasta el nivel deseado.</p> <p>Un cambio de estado de 1 a 0 pone inmediatamente el nivel actual al offset "B" y, por tanto, la salida AQ a 0.</p>
<b><i>Entrada Sel</i></b>	<p>SeI = 0: se selecciona el nivel 1 (Level 1).</p> <p>SeI = 1: se selecciona el nivel 2 (Level 2).</p> <p>Un cambio de estado de Sel ocasiona el desplazamiento desde el nivel actual hasta el nivel deseado a la velocidad indicada.</p>
<b><i>Entrada St</i></b>	<p>Un cambio de estado de 0 a 1 en la entrada St (parada desacelerada) hace que el nivel actual se reduzca a una velocidad constante hasta alcanzar el nivel de arranque/parada (offset "B" + StSp). El nivel de arranque/parada se mantiene durante 100 ms y el nivel actual se pone luego al offset "B". Por tanto, la salida AQ se pone a 0.</p>
<b><i>Parámet ro</i></b>	Configuración interna de los parámetros del bloque
<b><i>Salida AQ</i></b>	<p>La salida AQ se escala con la fórmula siguiente: (nivel actual - offset "B") / ganancia "A"</p> <p>Nota: si AQ se visualiza en el modo de parametrización o de textos de aviso, se mostrará como valor no escalado (en unidades de ingeniería: nivel actual).</p> <p>Rango de valores para AQ: 0 a 32767</p>

## Párametros



**L1 y L2:** niveles que deben alcanzarse. Rango de valores por nivel: de -10000 a 20000

**MaxL:** Valor máximo que no debe excederse.

Rango de valores: de -10000 a 20000

**StSp:** offset de arranque/parada: valor que se suma al offset "B" para crear el nivel de arranque/parada. Si el offset de arranque/parada = 0, el nivel de arranque/parada es igual al offset "B".

Rango de valores: de 0 a 20000

**Velocidad:** Velocidad a la que se alcanzan los niveles 1 o 2, o bien el offset. Se indican pasos por segundo.

Rango de valores: 1 a 10000

### Ganancia

Rango de valores: 0 a 10,00

### Offset

Rango de valores: de -10000 a +10000

**p:** número de decimales

Ajustes posibles: 0, 1, 2, 3

## Descripción de la función

Si se activa la entrada En, la función ajusta el valor StSp + offset "B" durante 100 ms.

Luego, dependiendo de la conexión de Sel, la función efectúa un desplazamiento desde el nivel StSp + offset "B" hasta el nivel 1 o 2 a la velocidad ajustada en Velocidad.

Si se activa la entrada St, la función efectúa un desplazamiento al nivel StSp + B con la aceleración ajustada en Velocidad. Seguidamente la función retiene el nivel a StSp + offset "B" durante 100 ms. Después de 100 ms, el nivel se ajusta al offset "B". Salida AQ. El valor escalado (salida AQ) es 0.

Si se activa la entrada St, la función solo puede reiniciarse una vez desactivadas las entradas St y En.

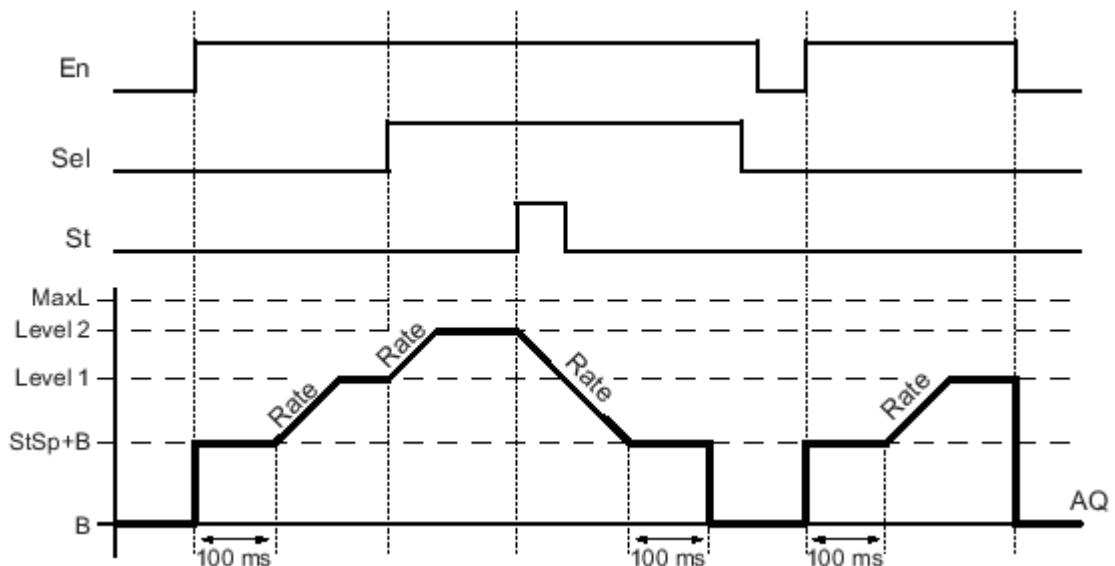
Si se modifica la entrada Sel, la función efectúa un desplazamiento (dependiendo de la conexión de Sel) desde el nivel de destino actual hasta el nuevo nivel de destino a la velocidad indicada.

Si se desactiva la entrada En, la función ajusta inmediatamente el nivel actual al offset "B".

El nivel actual se actualiza cada 100ms. Tenga en cuenta la relación entre la salida AQ y el nivel actual:

Salida AQ = (nivel actual – offset "B" ) / ganancia "A"

### Cronograma



### Regulador PI



El bloque Regulador PI (Proporcional–Integral) es un controlador analógico diseñado para minimizar el error entre una señal de referencia y la señal de proceso medida. Su funcionamiento combina dos acciones:

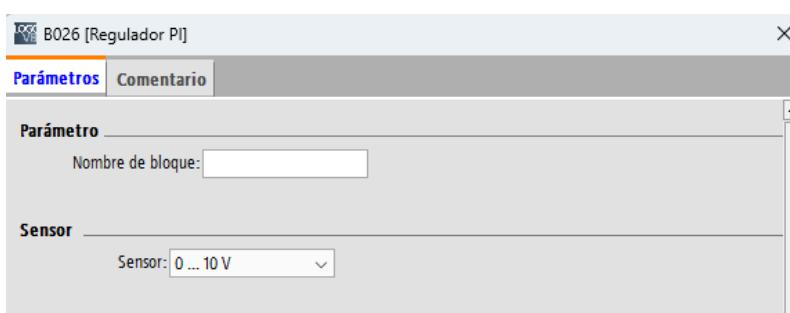
- Acción proporcional (P): responde de forma inmediata al error actual, generando una salida proporcional a la diferencia entre el valor deseado y el valor real.
- Acción integral (I): acumula el error a lo largo del tiempo, corrigiendo desviaciones persistentes y eliminando el error en estado estacionario.

El Regulador PI ajusta automáticamente la salida para lograr que el sistema alcance el punto de consigna con estabilidad y precisión, evitando oscilaciones excesivas y reduciendo el tiempo de establecimiento. Es ampliamente utilizado en procesos industriales de temperatura, presión, velocidad, nivel o caudal, donde se requiere control continuo y suave de la variable controlada.

### ***Pinout y descripción de conexión***

<b>Entrada (A/M)</b>	Permite ajustar el modo del regulador: 1: modo automático 0: modo manual
<b>Entrada (R)</b>	La entrada R sirve para desactivar la salida AQ. Mientras esta entrada está activada, la entrada A/M está desactivada. La salida AQ se pone a 0.
<b>Entrada (PV)</b>	Valor analógico: valor real, influye en la salida
<b>Parámetro (Par)</b>	Configuración interna de los parámetros del bloque
<b>Salida (AQ)</b>	Salida analógica (magnitud manipulada) Rango de valores para AQ: de 0 a 1.000

### ***Párametros***



**Sensor:** Tipo de sensor utilizado

**Mín.:** valor mínimo de PV

Rango de valores: de -10.000 a +20.000

**Máx.:** valor máximo de PV

Rango de valores: de -10.000 a +20.000

### **Ganancia**

Rango de valores: de -10,00 a +10,00

Offset

Rango de valores: de -10.000 a +10.000

**SP:** Asignación del valor de ajuste

Rango de valores: de -10.000 a +20.000

**Mq:** valor de AQ en modo manual.

Rango de valores: de 0 a 1.000

Juegos de parámetros: valores predeterminados según la aplicación para KC, TI y Dir (v. abajo)

**KC:** ganancia

Rango de valores: de 00,00 a 99,99

**TI:** tiempo integral

Rango de valores: de 00:01 min a 99:59 min

**Dir:** sentido de actuación del regulador

Ajustes posibles: + o -

**p:** número de decimales

Ajustes posibles: 0, 1, 2, 3

### **Descripción de la función**

Si la entrada A/M es 0, la función especial emite en la salida AQ el valor que se ha ajustado en el parámetro Mq.

Si la entrada A/M es 1, se inicia el modo automático. Puesto que el valor Mq adoptado es una suma integral, la función del regulador comienza a realizar los cálculos de acuerdo con las fórmulas indicadas en los principios básicos de control y regulación. Utilice el valor actualizado PV en las fórmulas:

Valor actualizado PV = (PV \* ganancia) + offset

Si el valor actualizado PV = SP, la función especial no modifica el valor en AQ.

Dir = hacia arriba /+ (cifras 1 y 3 del cronograma)

- Si el valor real PV > SP, la función especial reduce el valor de AQ.
- Si el valor real PV < SP, la función especial incrementa el valor de AQ.

Dir = hacia abajo /- (cifra 2 del cronograma)

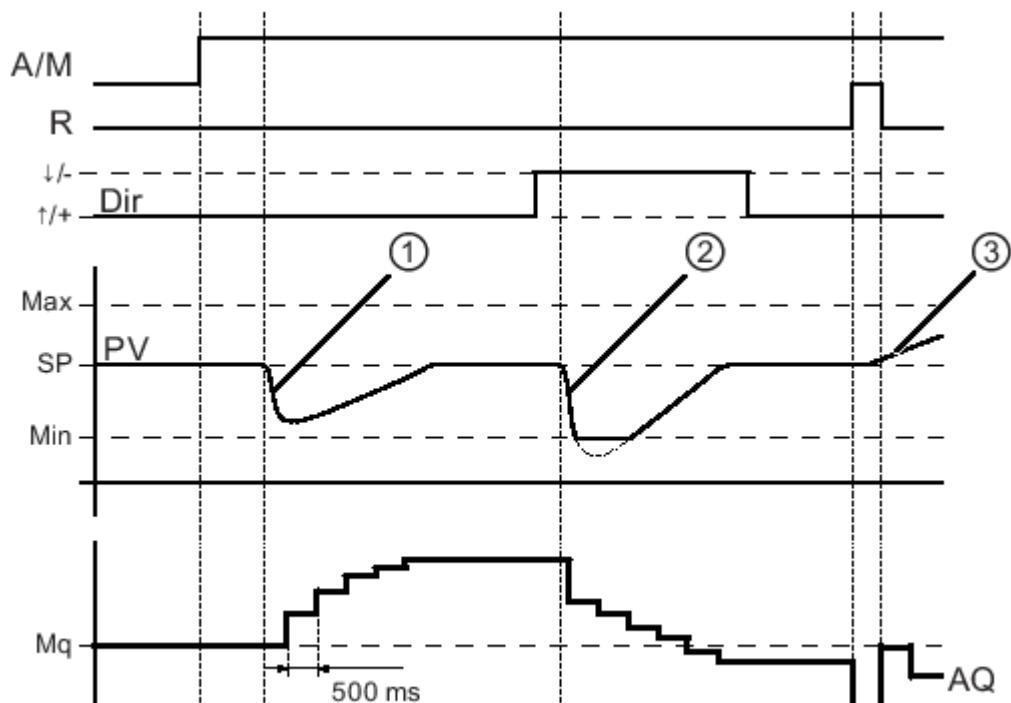
- Si el valor real PV > SP, la función especial incrementa el valor de AQ.
- Si el valor real PV < SP, la función especial reduce el valor de AQ.

Si se produce un fallo, AQ se incrementa/reduce hasta que el valor real PV vuelva a equivaler a SP. La rapidez con que cambia AQ depende de los parámetros KC y TI.

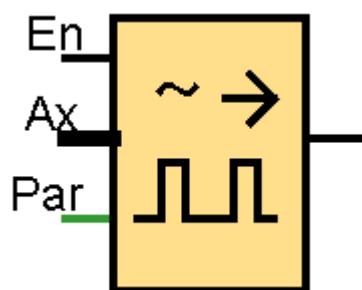
Si el valor de la entrada PV es superior al parámetro Max, el valor real PV adopta el valor de Max. Si el valor de PV es inferior al parámetro Min, el valor real PV adopta el valor Min.

Si la entrada R es 1, se desactiva la salida AQ. Mientras R está activada, la entrada A/M está desactivada.

### Cronograma



## **Modulación de ancho de impulsos (PWM)**



El bloque Modulación de Ancho de Impulso (PWM) convierte una señal analógica o un valor de control en una señal digital pulsada, cuyo ciclo de trabajo (duty cycle) se ajusta proporcionalmente al valor de entrada. Aunque la señal generada es digital, su efecto promedio equivale a un nivel analógico, permitiendo controlar potencia, velocidad o intensidad en actuadores eléctricos.

El bloque PWM mantiene una frecuencia fija y varía únicamente el tiempo en que la salida permanece en nivel alto dentro de cada ciclo. De esta manera, un ciclo de trabajo bajo entrega poca energía, mientras que un ciclo de trabajo alto suministra mayor energía al dispositivo controlado.

Este bloque se utiliza comúnmente en el control de motores, regulación de luminarias, control de calentadores y aplicaciones donde se requiere ajustar la potencia sin necesidad de conversión analógica completa.

### **Pinout y descripción de conexión**

<b>Entrada (En)</b>	Un flanco ascendente (cambio de 0 a 1) en la entrada En habilita el bloque de función PWM.
<b>Entrada (Ax)</b>	Señal analógica que debe modularse a una señal de salida de impulso digital.

<b>Parámetro (Par)</b>	Configuración interna de los parámetros del bloque
<b>Salida (Q)</b>	Q se activa o desactiva durante la proporción de cada período de tiempo según la proporción del valor normalizado Ax respecto al rango de valores analógicos.

## **Descripción de la función**

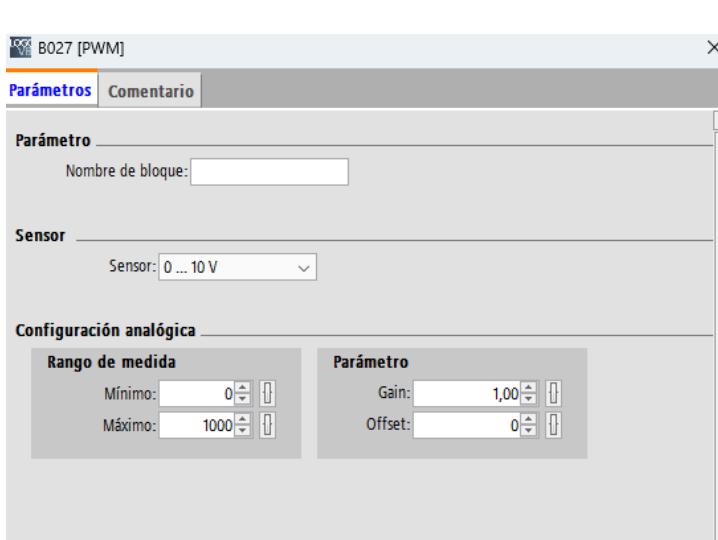
La función lee el valor de la señal aplicada en la entrada analógica Ax.

Este valor se multiplica por el valor del parámetro A (ganancia). El parámetro B (offset) se suma al producto del siguiente modo:

$$(Ax * \text{ganancia}) + \text{offset} = \text{valor real Ax}$$

El bloque de función calcula la proporción del valor Ax respecto al rango. El bloque activa la salida digital Q durante la misma proporción del parámetro PT (tiempo periódico) y desactiva Q durante el tiempo restante

## **Párametros**



Ganancia

Rango de valores: de -10,00 a 10,00

Offset

Rango de valores: de -10.000 a 10.000

PT: Tiempo periódico en el que se modula la salida digital

p: Número de decimales

Ajustes posibles: 0, 1, 2, 3

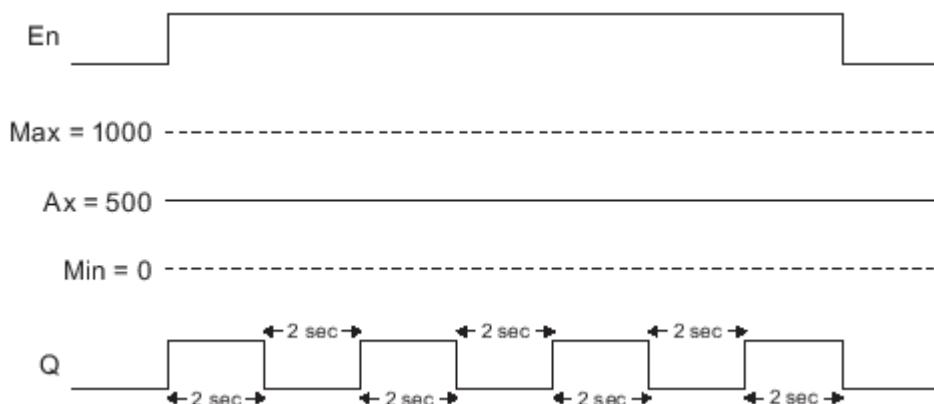
## Cronogramas

Ejemplo 1

Valor de la entrada analógica: 500 (rango de 0 a 1000)

Tiempo periódico T: 4 segundos

La salida digital de la función PWM está activada 2 segundos, desactivada 2 segundos, activada 2 segundos, desactivada 2 segundos, etc. mientras el parámetro "En" = high.

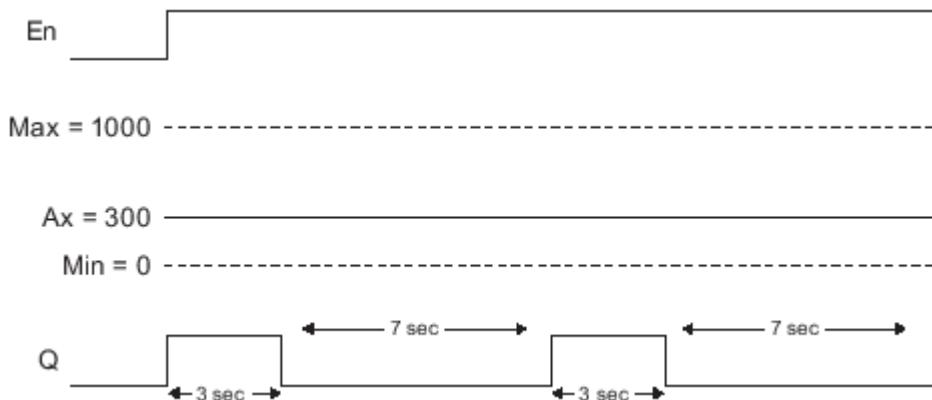


## Ejemplo 2

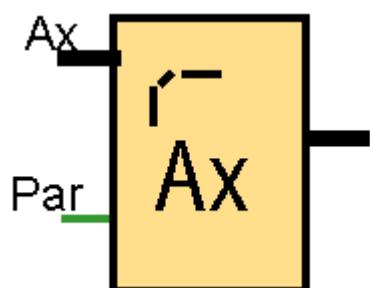
Valor de la entrada analógica: 300 (rango de 0 a 1000)

Tiempo periódico T: 10 segundos

La salida digital de la función PWM está activada 3 segundos, desactivada 7 segundos, activada 3 segundos, desactivada 7 segundos, etc. mientras el parámetro "En" = high.



## Filtro analógico



El bloque Filtro Analógico se emplea para procesar y acondicionar una señal analógica de entrada mediante la atenuación de componentes no deseados, tales como ruido, fluctuaciones o picos transitorios. Su función es suavizar la señal, conservando las variaciones lentas y relevantes del proceso mientras reduce las perturbaciones de alta frecuencia.

Dependiendo de su configuración, el filtro puede comportarse como paso bajo, paso alto, paso banda o eliminador de banda. Esto permite adaptar la señal a las necesidades específicas del sistema de control, mejorar la precisión de mediciones, estabilizar señales antes de convertirlas a digital, o evitar activaciones erróneas de dispositivos sensibles al ruido.

En resumen, el bloque “Filtro Analógico” contribuye a obtener una señal más limpia y estable para un procesamiento confiable dentro del sistema.

## Pinout y descripción de conexión

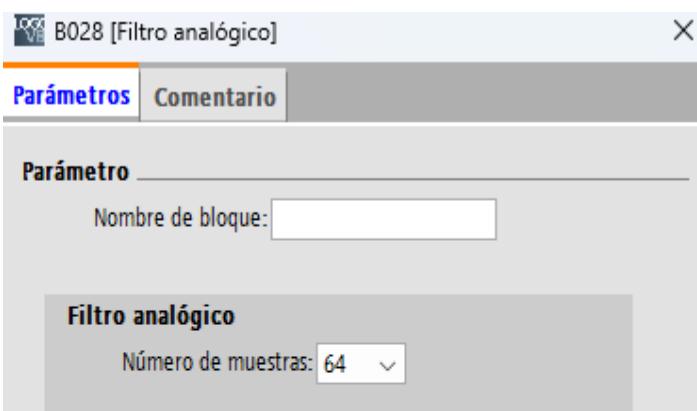
<b>Entrada (Ax)</b>	La entrada Ax es una de las siguientes señales analógicas:
	<ul style="list-style-type: none"> <li>• AI1 a AI8 (*)</li> <li>• AM1 a AM16 (para 0BA7) o AM1 a AM64 (para 0BA8)</li> <li>• NAI1 a NAI32</li> <li>• AQ1 y AQ2 (para 0BA7) o AQ1 a AQ8 (para 0BA8)</li> <li>• NAQ1 a NAQ16</li> <li>• El número de bloque de una función con salida analógica</li> </ul>
<b>Parámetro (Par)</b>	Configuración interna de los parámetros del bloque
<b>Salida (AQ)</b>	AQ emite un valor medio de la entrada analógica Ax a lo largo del número actual de muestras y se activa o desactiva en función de la entrada analógica y del número de muestras.

**\* AI1 a AI8: 0 a 10 V es proporcional a 0 a 1000 (valor interno).**

### Descripción de la función

La función emite el valor medio después de muestrear la señal de entrada analógica conforme al número de muestras ajustado. Este SFB reduce el error de la señal de entrada analógica.

### Párametros



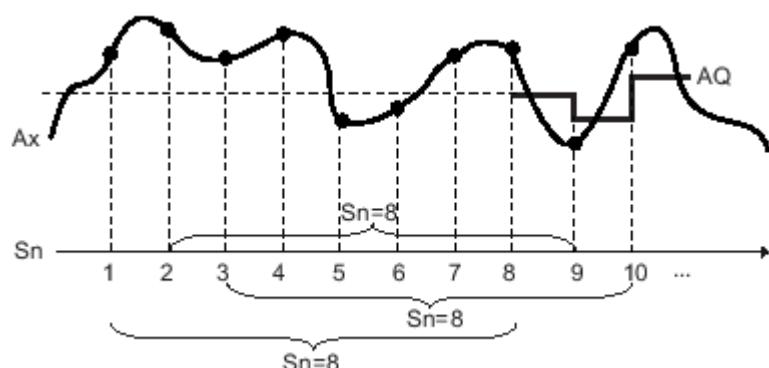
**Sn (Número de muestras):** determina cuántos valores analógicos se muestrean dentro de los ciclos del programa que están determinados por el número de muestras ajustado. LOGO! muestrea un valor analógico en

cada ciclo del programa. El número de ciclos del programa coincide con el número de muestras ajustado.

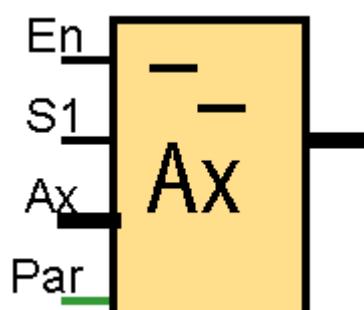
Ajustes posibles:

8, 16, 32, 64, 128, 256

### Cronograma



### Máx/Mín



El bloque Máx/Mín permite registrar y conservar el valor máximo o mínimo alcanzado por una señal de entrada a lo largo del tiempo. Según su configuración, el bloque compara continuamente el valor actual con el valor almacenado y actualiza el registro únicamente cuando se supera el límite correspondiente.

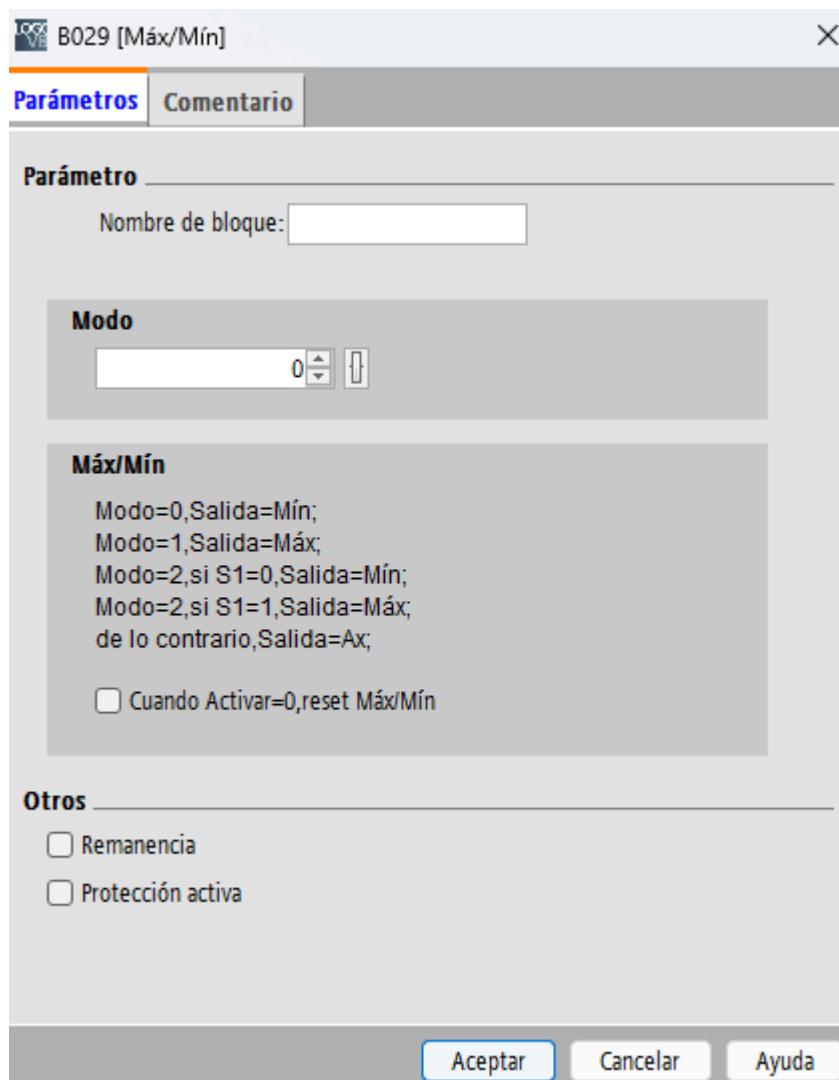
Este bloque es útil para monitorear picos de señal, detectar condiciones extremas, analizar tendencias o activar alarmas cuando se exceden valores críticos. También puede emplearse para diagnóstico y supervisión de procesos, permitiendo identificar variaciones inusuales en la variable medida.

Algunas implementaciones incluyen opciones de reinicio para borrar el valor guardado y comenzar un nuevo registro.

### Pinout y descripción de conexión

<b>Entrada En</b>	La función de la entrada En (Enable) depende del ajuste del parámetro "Modo" y de la selección de la casilla de verificación "cuando En = 0, resetear Máx/Mín".
<b>Entrada S1</b>	<p>Esta entrada se activa cuando se ajusta Modo =2:</p> <p>Un flanco ascendente (0 a 1) en la entrada S1 ajusta la salida AQ al valor máximo.</p> <p>Un cambio descendente (1 a 0) en la entrada S1 ajusta la salida AQ al valor mínimo.</p>
<b>Entrada (Ax)</b>	<p>La entrada Ax es una de las siguientes señales analógicas:</p> <ul style="list-style-type: none"> <li>• AI1 a AI8 (*)</li> <li>• AM1 a AM16 (para 0BA7) o AM1 a AM64 (para 0BA8)</li> <li>• NAI1 a NAI32</li> <li>• AQ1 a AQ2 (para 0BA7) o AQ1 a AQ8 (para 0BA8)</li> <li>• NAQ1 a NAQ16</li> <li>• Número de bloque de una función con salida analógica</li> </ul>
<b>Parámetro (Par)</b>	Configuración interna de los parámetros del bloque
<b>Salida (AQ)</b>	AQ emite un valor mínimo, máximo o real dependiendo de las entradas o bien se pone a 0 si ha sido configurada para tal fin cuando la función está desactivada
<p><b>* AI1 a AI8: 0 a 10 V es proporcional a 0 a 1000 (valor interno).</b></p>	

## Párametros



Si se selecciona la casilla de verificación "Cuando Activar = 0, reset Máx/Mín":

- Activar = 0: la función pone el valor AQ a 0.

- Activar = 1: la función emite un valor en AQ, dependiendo de los ajustes de Mode y S1.

Si no se selecciona la casilla de verificación "Cuando Activar = 0, reset Máx/Mín":

- Activar = 0: la función mantiene el valor de AQ en el valor actual.

- Activar = 1: la función emite un valor en AQ, dependiendo de los ajustes de Mode y S1.

Modo = 0: la función pone AQ al valor mínimo

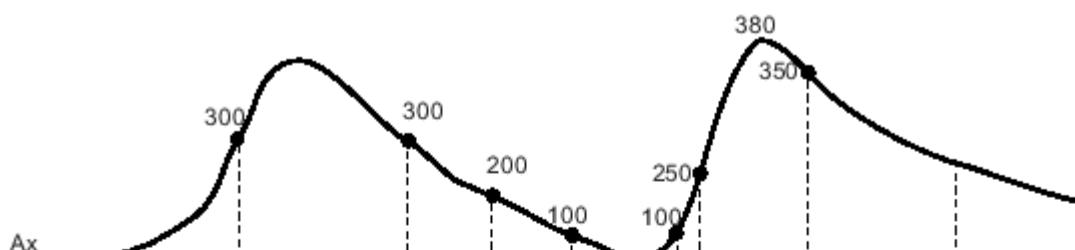
Modo = 1: la función pone AQ al valor máximo

Modo = 2 y S1 = 0: la función pone AQ al valor mínimo

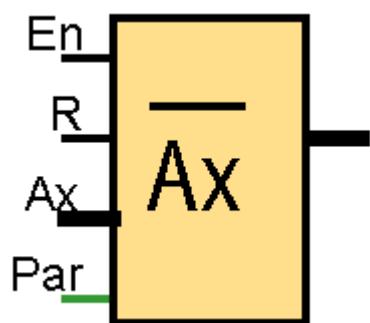
Modo = 2 y S1 = 1: la función pone AQ al valor máximo

Modo = 3 o el valor de un bloque está referenciado: la función emite el valor real de la entrada analógica.

## Cronograma



## **Valor medio**



El bloque Valor Medio calcula el promedio de una señal analógica de entrada a lo largo de un período de muestreo configurado por el usuario. Durante este intervalo, el bloque registra múltiples valores de la señal y, al finalizar el período, entrega en la salida analógica (AQ) el valor medio obtenido.

Esta función es útil para suavizar fluctuaciones breves, reducir efectos de ruido y obtener un valor representativo del comportamiento real de la variable, especialmente en mediciones que presentan variaciones rápidas o aleatorias. El resultado permite un control más estable y una evaluación más precisa del proceso.

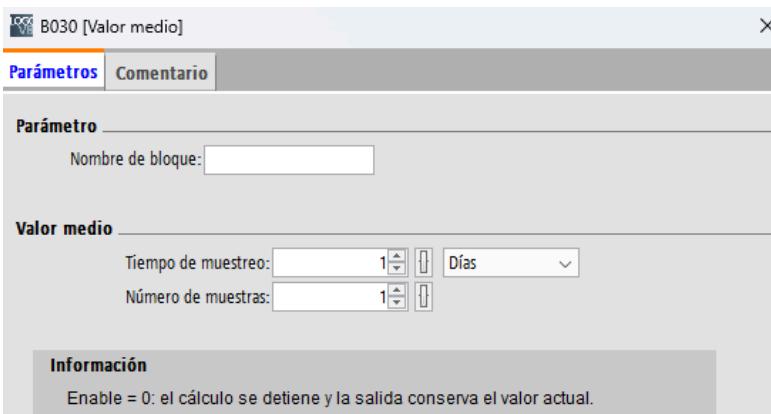
## **Pinout y descripción de conexión**

<b>Entrada (En)</b>	Un flanco ascendente (cambio de 0 a 1) en la entrada En (Enable) ajusta la salida AQ al valor medio de la entrada Ax después del tiempo
---------------------	---

	configurado. Un flanco descendente (cambio de 1 a 0) mantiene la salida en su último valor calculado.
<b>Entrada (R)</b>	Un flanco ascendente (cambio de 0 a 1) en la entrada R (Reset) pone la salida AQ a 0.
<b>Entrada (Ax)</b>	<p>La entrada Ax es una de las siguientes señales analógicas:</p> <ul style="list-style-type: none"> <li>• AI1 a AI8 (*)</li> <li>• AM1 a AM16 (para 0BA7) o AM1 a AM64 (para 0BA8)</li> <li>• NAI1 a NAI32</li> <li>• AQ1 y AQ2 (para 0BA7) o AQ1 a AQ8 (para 0BA8)</li> <li>• NAQ1 a NAQ16</li> <li>• El número de bloque de una función con salida analógica</li> </ul>
<b>Parámetro (Par)</b>	Configuración interna de los parámetros del bloque
<b>Salida (AQ)</b>	AQ emite el valor medio a lo largo del tiempo de muestreo especificado.

\* **AI1 a AI8: 0 a 10 V es proporcional a 0 a 1000 (valor interno).**

## Párametro



**St (Tiempo de muestreo):** se puede ajustar en segundos, días, horas y minutos.

Rango de valores:

Si St = segundos: 1 a 59

Si St = días: 1 a 365

Si St = horas: 1 a 23

Si St = minutos: 1 a 59

Sn (Número de muestras):

Rango de valores:

Si St = segundos: 1 a St\*100

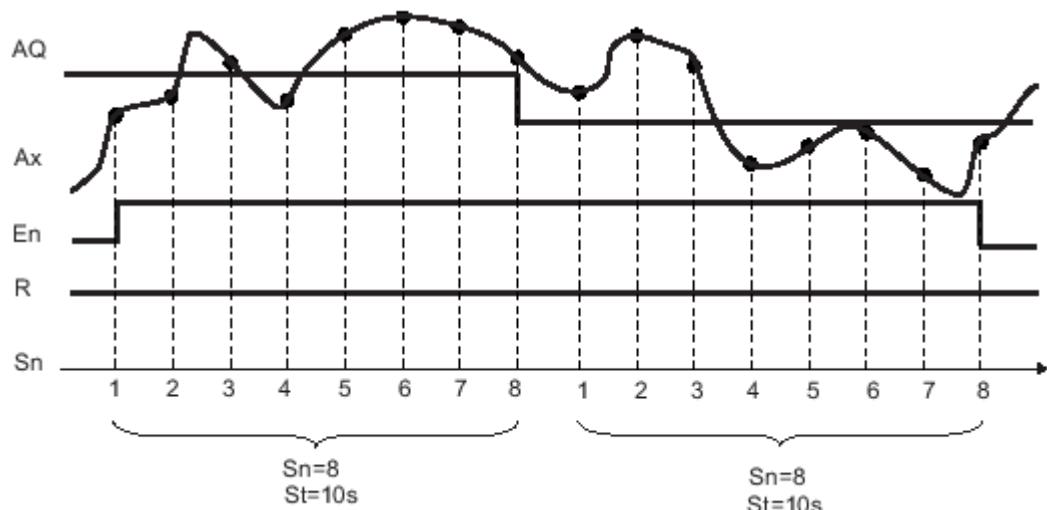
Si St = días: 1 a 32767

Si St = horas: 1 a 32767

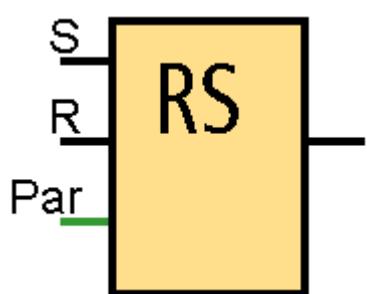
Si St = minutos y St  $\leq$  5 minutos: 1 a St\*6000

Si St = minutos y St  $\geq$  6 minutos: 1 a 32767

## Cronograma



## Relé autoenclavador



El bloque Relé Autoenclavador es un elemento de memoria binario que mantiene su estado de salida incluso después de que

la condición de activación haya desaparecido. La salida depende tanto del estado actual de las entradas como del estado previo que el relé tenía almacenado.

Cuando la entrada de activación se energiza, el relé comuta su salida y queda “enclavado” en ese estado. Dicho estado permanece hasta que una entrada de reset o desactivación lo cambie nuevamente. Gracias a esta capacidad de retención, el relé autoenclavador se emplea como memoria simple para funciones de encendido/apagado, secuencias de control y procesos que requieren mantener un estado lógico sin necesidad de señal continua.

La tabla lógica del relé define el comportamiento específico ante las entradas de activación, reset y el valor previo de la salida.

### ***Pinout y descripción de conexión***

<b><i>Entrada (S)</i></b>	La salida Q se activa con una señal en la entrada S (Set).
<b><i>Entrada (R)</i></b>	La salida Q se desactiva con una señal en la entrada R (Reset). La salida Q se desactiva si están activadas tanto S como R (la desactivación tiene prioridad sobre la activación).
<b><i>Parámetros (Par)</i></b>	Remanencia activada (ON) = el estado se guarda de forma remanente.
<b><i>Salida (Q)</i></b>	Q se activa con una señal en la entrada S y permanece activada hasta que es desactivada con una señal en la entrada R.

### ***Descripción de la función***

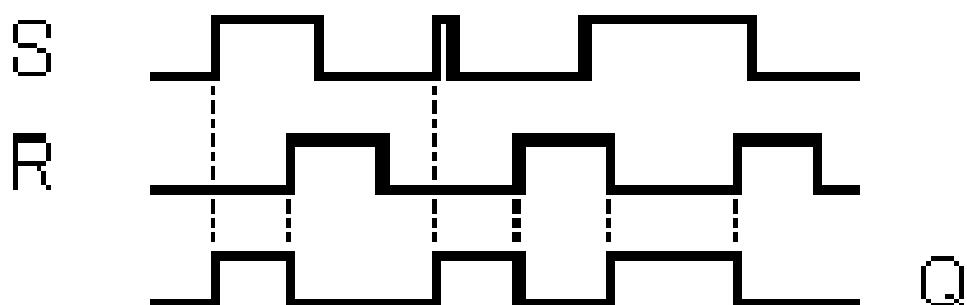
Un relé autoenclavador es un elemento de memoria binario simple. El valor de la salida depende del estado de las entradas y del estado anterior de la salida.

Tabla lógica del relé autoenclavador:

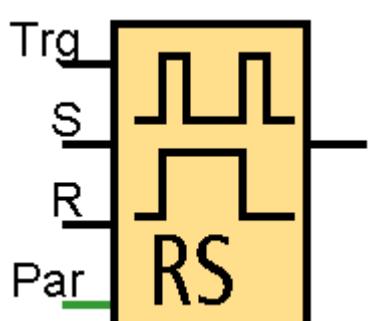
S	R	Q	Observación
0	0	x	El estado no cambia
0	1	0	Desactivación
1	0	1	Activación
1	1	0	Desactivación

Si está habilitada la remanencia, se aplica a la salida el mismo estado de señal que tenía antes del corte de alimentación.

### Cronograma



### Relé de impulsos



El bloque Relé de Impulsos es un elemento de conmutación binario que cambia el estado de su salida cada vez que recibe un pulso en la entrada. Un breve impulso es suficiente para alternar

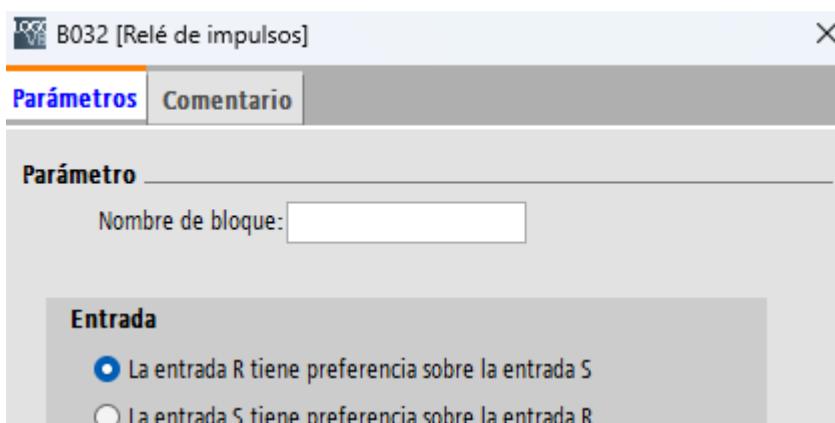
la salida entre encendido y apagado, sin necesidad de mantener la señal activa.

Este comportamiento lo convierte en una forma de “interruptor electrónico” controlado por pulsos, útil para activar o desactivar cargas, memorizar estados lógicos y realizar funciones de control con un solo botón o señal momentánea. La salida permanece en su estado actual hasta que se recibe un nuevo impulso.

### ***Pinout y descripción de conexión***

<b><i>Entrada (Trg)</i></b>	Por medio de la entrada Trg (Trigger) se activa y desactiva la salida Q.
<b><i>Entrada (S)</i></b>	Un impulso en la entrada S (Set) pone la salida a 1.
<b><i>Entrada (R)</i></b>	Un impulso en la entrada R (Reset) pone la salida a 0.
<b><i>Parámetro (Par)</i></b>	Selección: RS (la entrada R tiene prioridad) o SR (la entrada S tiene prioridad)  Remanencia activada (ON) = el estado se guarda de forma remanente.
<b><i>Salida (Q)</i></b>	Q se activa con una señal en Trg y se desactiva con el siguiente impulso de Trg si S y R = 0.

### ***Párametros***



#### ***Selección:***

RS (la entrada R tiene prioridad) o

SR (la entrada S tiene prioridad)

Remanencia activada (ON) = el estado se guarda de forma remanente.

### ***Descripción de la función***

Cada vez que el estado de la entrada Trg cambia de 0 a 1 y si las entradas S y R = 0, cambia también el estado de la salida Q, es decir, la salida se activa o desactiva.

La entrada Trg no afecta a la función especial si S = 1 o R = 1.

Por medio de la entrada S se activa el relé de impulsos, es decir, la salida se pone a 1.

Por medio de la entrada R se restablece el estado inicial del relé de impulsos, es decir, la salida se pone a 0.

Dependiendo de la configuración, la entrada R tiene prioridad sobre la entrada S (la señal en la entrada S no tiene efecto mientras R = 1), o bien la entrada S tiene prioridad sobre la entrada R (la señal en la entrada R no tiene efecto mientras S = 1).

#### **0BA0-0BA3:**

La función válida es:

**0BA0**  
.  
**0BA8**

El estado de la salida Q se commuta cada vez que el estado de la entrada Trg cambia de 0 a 1, es decir, la salida se activa o desactiva.

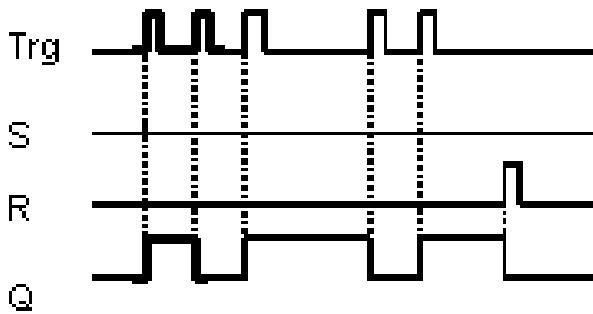
Por medio de la entrada R se restablece el estado inicial del relé de impulsos, es decir, la salida se pone a 0.

Tras conectar la alimentación o después de un reset, el relé de impulsos se reinicializa y la salida Q se pone a 0.

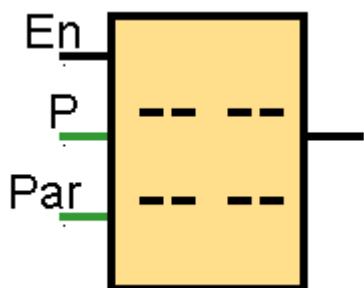
Cuidado

Si Trg = 0 y Par = RS, la función especial "Relé de impulsos" equivale a la función especial "Relé autoenclavador".

## Cronograma



## Texto de aviso



El bloque Texto de Aviso permite mostrar mensajes informativos, estados del sistema y valores relevantes de otros bloques directamente en el display integrado del LOGO! o en el módulo externo LOGO! TDE.

Cuando el controlador se encuentra en modo RUN, este bloque presenta textos predefinidos o dinámicos según las condiciones del programa, facilitando la supervisión local del proceso y proporcionando al usuario una referencia visual inmediata sobre el funcionamiento, alarmas o parámetros activos dentro de la automatización.

## Pinout y descripción de conexión

<b>Entrada (En)</b>	Una transición de 0 a 1 en la entrada En (Enable) inicia la visualización del texto de aviso.
<b>Entrada (P)</b>	P es la prioridad del texto de aviso. 0 es la prioridad más baja y 127 la más alta. Ack: acuse del texto del aviso

<p><b>Parámetro (Par)</b></p>	<p>Texto: entrada del texto de aviso</p> <p>Par: parámetro o valor real de otra función preprogramada que puede visualizarse de forma numérica o en un diagrama de barras (consulte "Parámetros o valores reales visualizables")</p> <p>Hora: muestra la hora continuamente actualizada</p> <p>Fecha: muestra la fecha continuamente actualizada</p> <p>Hora En: visualización de la hora de cambio de estado de señal de 0 a 1 en la entrada En</p> <p>Fecha En: visualización de la fecha de cambio de estado de señal de 0 a 1 en la entrada En</p> <p>Nombres de estado de E/S: visualización del nombre de un estado de entrada o salida digital, p. ej. "On" u "Off". Los dispositivos LOGO! 0BA8 pueden visualizar los nombres de estado de los siguientes elementos:</p> <ul style="list-style-type: none"> <li>• Entradas digitales</li> <li>• Salidas digitales</li> <li>• Marcas</li> <li>• Teclas de cursor</li> <li>• Teclas de función del LOGO! TDE</li> <li>• Bits de registro de desplazamiento</li> <li>• Salidas del bloque de función</li> </ul> <p>Entrada analógica: indicación del valor de la entrada analógica que debe visualizarse en el texto de aviso y actualizarse conforme al tiempo analógico.</p> <p>Escala de tiempo: indicación del valor real de un bloque de función referenciado como un valor de tiempo escalado de acuerdo con la base</p>
-------------------------------	---

	<p>de tiempo configurada para el bloque de función de textos de aviso. A continuación se indican los formatos de tiempo disponibles:</p> <ul style="list-style-type: none"> <li>• horas : minutos : segundos . milisegundos</li> <li>• horas : minutos : segundos</li> <li>• horas : minutos</li> <li>• horas</li> </ul> <p>(por ejemplo, "01: 20 : 15 .15")</p> <p>Símbolo: indicación de los caracteres seleccionados en los juegos de caracteres soportados</p>
<b><i>Salida (Q)</i></b>	Q permanece activada mientras está pendiente el texto de aviso.

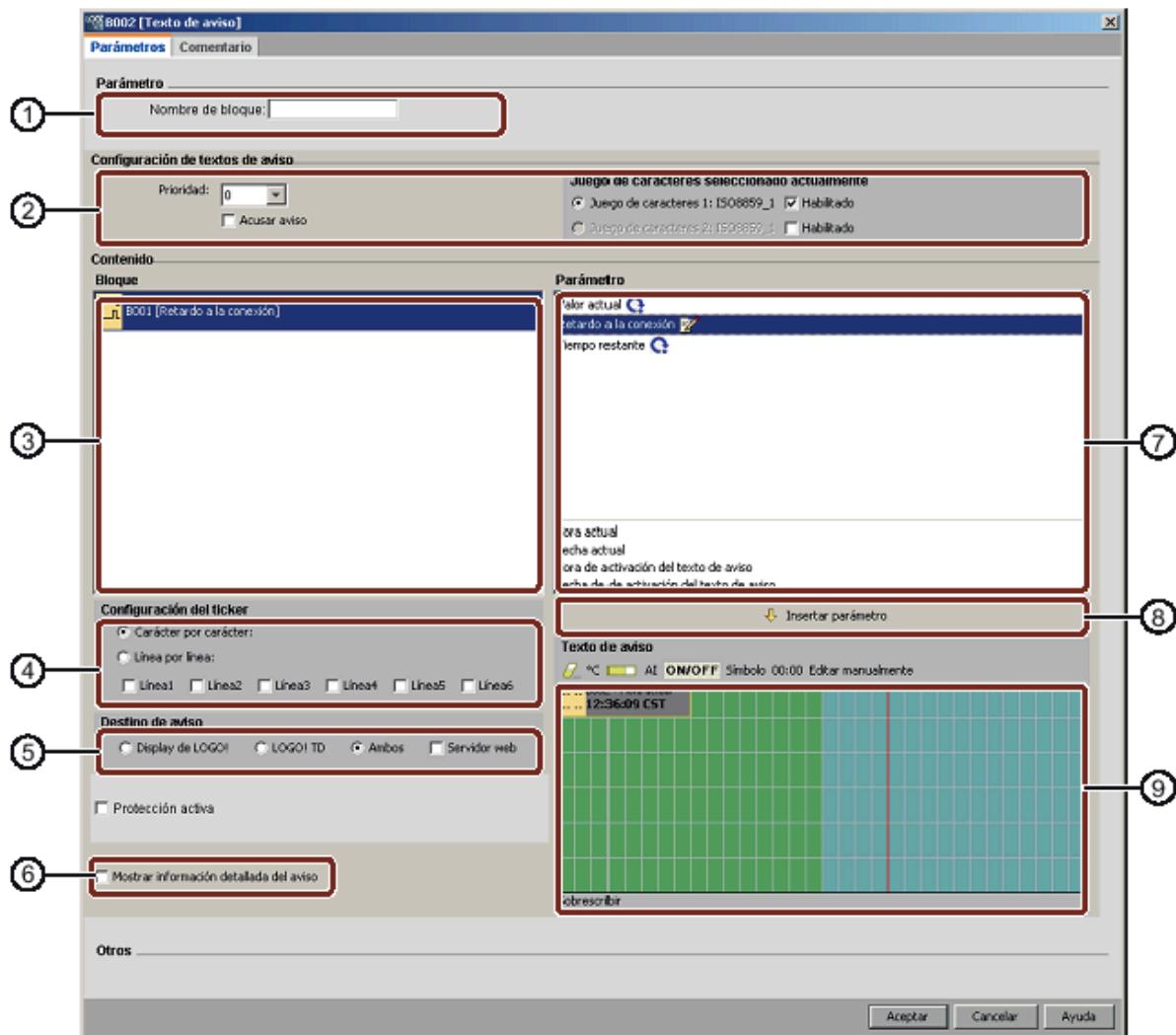
## ***Configuración***

Además de las entradas del bloque de función y los parámetros del texto de aviso, los siguientes ajustes ofrecen un control adicional de la visualización de textos de aviso:

- Selección del juego de caracteres: Puede elegir entre crear un texto utilizando el juego de caracteres primario o el secundario. Estos dos juegos de caracteres se seleccionan bien sea en el menú "Config. avisos" del módulo base LOGO! bien utilizando el comando de menú Archivo → Configuración del texto de aviso en LOGO!Soft Comfort.
- Destino del aviso: Aquí puede seleccionar si el texto de aviso debe visualizarse en el display integrado de LOGO!, en el LOGO! TDE o bien en ambos. Si selecciona como destino el servidor web, los textos de aviso de LOGO! pueden visualizarse en el servidor web.
- Configuración del ticker: El texto de aviso puede visualizarse en un ticker (texto en movimiento), o bien de forma fija. Hay dos tipos de tickers para avisos, a saber:
  - Carácter por carácter
  - Línea por línea

- Cada línea de un texto de aviso también puede configurarse como ticker. El ajuste "carácter por carácter" o "línea por línea" se aplica a todas las líneas que se configuren como ticker. Active la casilla de verificación junto a un número de línea para configurar esa línea como ticker. El número de líneas depende de la selección realizada en el cuadro de diálogo Archivo → Configuración del texto de aviso.

## *Particularidades acerca de la configuración*



① Área "Nombre de bloque"

Aquí puede introducir un nombre para el bloque.

② Área "Configuración"

Aquí puede configurar los ajustes siguientes:

- Prioridad del texto de aviso
- Casilla de verificación "Acusar aviso": si está activada, un aviso se debe acusar para poder cerrarlo.
- Juego de caracteres para el texto de aviso

③ Área "Bloque"

Aquí puede seleccionar bloques de entre todos los bloques del programa. Luego puede seleccionar parámetros de estos bloques para visualizarlos en el texto de aviso.

④ Área "Ticker"

Aquí se definen los parámetros del ticker para el texto de aviso:

- Formato de ticker "Carácter por carácter"
- Formato de ticker "Línea por línea"
- Casilla de verificación para activar el ticker de cada línea

⑤ Área "Destino del aviso"

Aquí puede seleccionar si el destino del aviso es el display integrado de LOGO!, el LOGO! TDE o bien ambos. Si selecciona como destino el servidor web, los textos de aviso de LOGO! pueden visualizarse en el servidor web.

⑥ Mostrar información detallada del aviso

Aquí se elige si los textos de aviso deben visualizarse con detalles. En el ajuste predeterminado, el texto del aviso muestra únicamente el estado de la prioridad, el aviso de acuse y los juegos de caracteres. Si se selecciona esta función se verán el ajuste del ticker, el ajuste de los textos de aviso y los textos de aviso en el editor de esquemas.

⑦ Área "Parámetros de bloque"

Aquí puede seleccionar los parámetros a visualizar en el texto de aviso, conforme al bloque seleccionado en el área "Bloque".

⑧ Botón "Insertar parámetro"

Este botón sirve para insertar un parámetro de bloque seleccionado en el texto de aviso.

## ⑨ Área "Aviso"

Aquí se configura el texto de aviso. La información introducida en esta área se corresponde con la que se visualizará en el display integrado de LOGO! o en el LOGO! TDE.

Por encima de esta área se encuentran más botones, a saber:



Botón "Borrar": permite borrar entradas en el área de avisos



Botón "Caracteres especiales": permite insertar caracteres especiales en el área de avisos



Botón "Diagrama de barras": permite insertar un diagrama de barras horizontal o vertical en el área de avisos



Botón "AI": permite insertar un valor de entrada analógico en el área de avisos



Botón "ON/OFF": permite especificar un valor digital que deba representarse en una de las dos cadenas correspondientes a los estados 0 y 1, p. ej. "OFF" y "ON".



Botón "Símbolo": permite abrir el teclado virtual. Se pueden insertar caracteres del juego de caracteres actual.



Botón "Escala de tiempo": permite referenciar y visualizar un valor real con una base de tiempo cuyo formato puede especificarse.

### **Editar manualmente**

Botón "Editar manualmente": permite utilizar el editor estático con objeto de agregar, mover o borrar elementos de textos de aviso sin modificar la posición de los demás elementos.

Nota: Con LOGO!Soft Comfort V8.3 y superiores, es posible copiar y pegar parámetros de textos de aviso en el mismo esquema.

## **Cómo configurar un texto de aviso**

Los dispositivos LOGO! 0BA8 y versiones posteriores soportan la visualización de textos de aviso de seis líneas. El área de avisos muestra una cuadrícula compuesta por seis líneas y posiciones de caracteres.

Si se selecciona como destino del aviso la pantalla integrada de LOGO!, el área del texto de aviso tiene un ancho de 32 caracteres para los idiomas de Europa occidental y 16 caracteres para los idiomas asiáticos. Si se selecciona como destino del aviso el LOGO! TDE, el área del texto de aviso tiene un ancho de 40 caracteres para los idiomas de Europa occidental y 20 caracteres para los idiomas asiáticos. En ambos casos, el ancho de caracteres de cada línea es el doble que el del display integrado de LOGO! o del LOGO! TDE. Si se seleccionan ambos como destino, el área del texto de aviso sigue teniendo un ancho de 32 caracteres para los idiomas de Europa occidental y 16 caracteres para los idiomas asiáticos. En este caso, hay una línea vertical roja en el área del texto de aviso que indica el límite del display del LOGO! TDE. Si también se selecciona el servidor web como destino, el texto de aviso se muestra por el servidor web igual que en el display integrado de LOGO!.

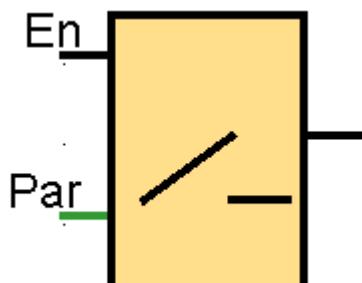
Las líneas de avisos cuya longitud excede el ancho del display pueden configurarse como "ticker" (texto en movimiento). En el área de avisos, LOGO!Soft Comfort muestra en color verde el área visible correspondiente al display integrado de LOGO! o al LOGO! TD. El área que solo puede visualizarse en el aviso en movimiento (ticker) se representa en color azul.

Para configurar el contenido de un texto de aviso, proceda del siguiente modo:

1. En el área "Bloque", seleccione el bloque cuyos parámetros deben visualizarse.
2. Arrastre los parámetros deseados desde el área "Parámetro" hasta el área "Texto de aviso". También existe la posibilidad de insertar un valor de parámetro mediante el botón "Insertar parámetro".
3. En el área "Texto de aviso" puede agregar datos de parámetros y valores de hora y fecha del área "Parámetros de bloque", así como introducir texto. Para introducir texto, seleccione el juego de caracteres para el aviso e introduzca luego el texto Los botones ubicados por encima del área "Texto de aviso" también pueden utilizarse para introducir

caracteres especiales, diagramas de barras, valores de entrada analógicos y nombres de estado de E/S digitales.

## ***Interruptor Software***



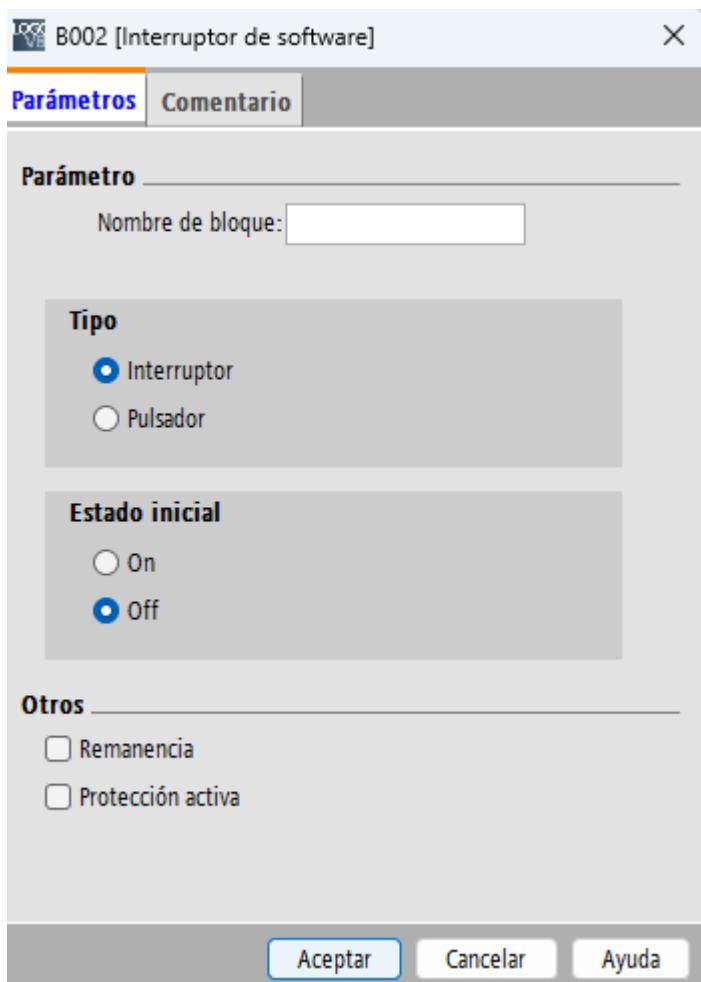
El Interruptor Software es una función especial que reproduce el comportamiento de un pulsador o interruptor mecánico dentro del programa. Permite activar o desactivar señales lógicas de forma manual mediante la interfaz del LOGO! o el LOGO! TDE, sin necesidad de incorporar un dispositivo físico.

Este bloque es útil para pruebas, control manual, habilitación de funciones o simulación de entradas, proporcionando una forma práctica y flexible de operar el sistema directamente desde el entorno del controlador.

### ***Pinout y descripción de conexión***

<b>Entrada (En)</b>	Un cambio de estado 0 a 1 en la entrada En (Enable) activa la salida Q si además se ha confirmado 'Estado = On' en el modo de configuración.
<b>Parámetro (Par)</b>	Switch: permite seleccionar si la función debe utilizarse como pulsador momentáneo durante un ciclo (On) o como interruptor (Off).  Estado: estado On u Off que se aplica en el primer ciclo al iniciar el programa.
<b>Salida (Q)</b>	La salida Q sigue puesta a 1 mientras En = 1 y los parámetros Tipo = Switch (interruptor) y Estado = On.  La salida Q se activa durante un ciclo si EN = 1. Switch = On (pulsador momentáneo) y Estado = On.

## Parametros



Switch: permite seleccionar si la función debe utilizarse como pulsador momentáneo durante un ciclo (On) o como interruptor (Off).

Estado: estado On u Off que se aplica en el primer ciclo al iniciar el programa.

## Descripción de la función

La salida se activa con una señal en la entrada En si el parámetro 'Estado' se ha ajustado a 'On' y confirmado con OK. Esto se realiza independientemente de si la función está programada como interruptor o pulsador.

La salida se pone a '0' en los tres casos siguientes:

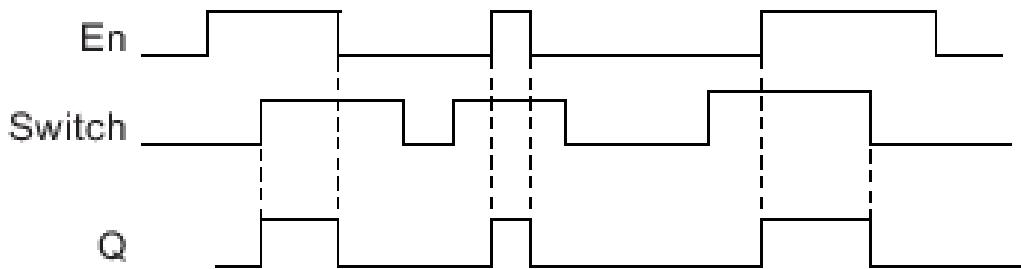
- Si el estado de la entrada En cambia de 1 a 0.
- Si la función se ha configurado como pulsador y ha transcurrido un ciclo tras su activación.
- Si el parámetro 'Estado' ajusta el estado 'Off' en el modo de configuración y esto se ha confirmado con OK.

## **Particularidades acerca de la configuración**

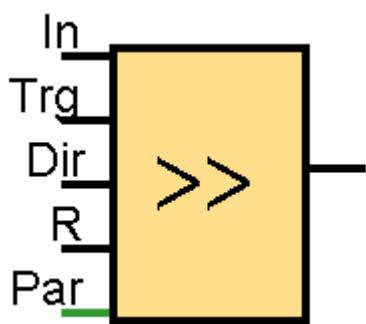
El interruptor software puede utilizarse como pulsador momentáneo o interruptor. Por medio del parámetro 'Estado' es posible definir si el interruptor o el pulsador está activado (accionado) o desactivado.

Si el interruptor software está configurado como pulsador, la salida se activa durante un ciclo siempre que con el pulsador conectado se produzca un cambio de 0 a 1 en la entrada En, o que con En=1 el estado del pulsador cambie de Off a On.

## **Cronograma**



## **Registro de desplazamiento**



El bloque Registro de Desplazamiento permite leer el valor lógico de una entrada y desplazar secuencialmente los bits almacenados dentro del registro interno del LOGO!. La salida del bloque corresponde al bit específico del registro que el usuario haya configurado.

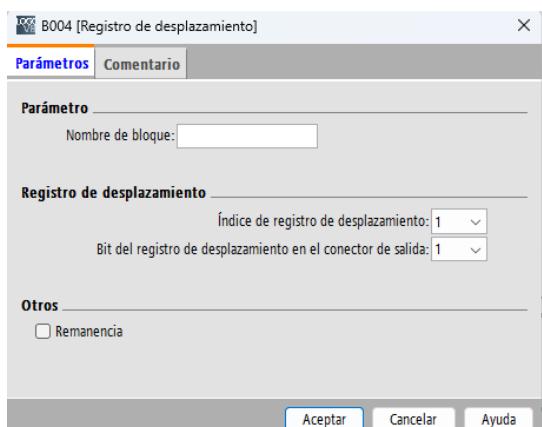
El sentido del desplazamiento puede controlarse mediante una entrada dedicada, permitiendo mover los bits hacia la izquierda o hacia la derecha según la lógica del proceso.

En las series de dispositivos 0BA4 a 0BA6, esta función está limitada a un único registro de desplazamiento por programa, debido a restricciones de hardware y memoria. El bloque resulta especialmente útil para generar secuencias, patrones de activación, efectos de desplazamiento y funciones temporizadas sin necesidad de múltiples elementos adicionales.

## ***Pinout y descripción de conexión***

<b><i>Entrada (In)</i></b>	Entrada cuyo valor se lee al inicio de la función.
<b><i>Entrada (Trg)</i></b>	Un flanco ascendente (cambio de 0 a 1) en la entrada Trg (Trigger) inicia la función especial. Un cambio de estado de 1 a 0 no es relevante.
<b><i>Entrada (Dir)</i></b>	Por medio de la entrada Dir se define el sentido de desplazamiento de los bits de registro de desplazamiento S1...S8:  Dir = 0: Desplazamiento hacia arriba (S1 >> S8) Dir = 1: Desplazamiento hacia abajo (S8 >> S1)
<b><i>Parámetros (Par)</i></b>	Configuración interna de los parámetros del bloque.
<b><i>Salida (Q)</i></b>	El valor de la salida equivale al bit de registro de desplazamiento configurado.

## ***Parametros***



Bit de registro de desplazamiento que determina el valor de la salida Q.

Ajustes posibles: S1 a S8

Remanencia activada (ON) = el estado se guarda de forma remanente.

## **Descripción de la función**

Con el flanco ascendente (cambio de 0 a 1) en la entrada Trg (Trigger), la función lee el valor de la entrada In.

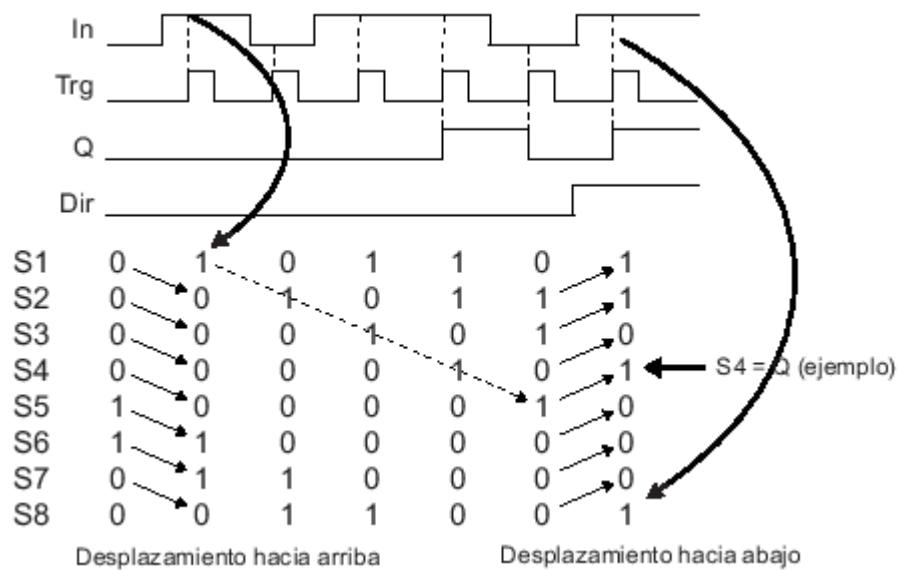
Dependiendo del sentido de desplazamiento, este valor se aplica en el bit de registro de desplazamiento S1 ó S8:

- Desplazamiento hacia arriba: S1 adopta el valor de la entrada In; el valor anterior de S1 se desplaza a S2; el valor anterior de S2 se desplaza a S3; etc.
- Desplazamiento hacia abajo: S8 adopta el valor de la entrada In; el valor anterior de S8 se desplaza a S7; el valor anterior de S7 se desplaza a S6; etc.

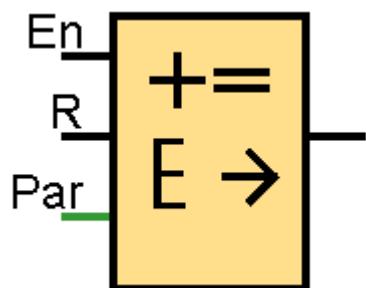
En la salida Q se devuelve el valor del bit de registro de desplazamiento configurado.

Si la remanencia no está activada, tras producirse un corte de alimentación, la función de desplazamiento comienza de nuevo en S1 ó S8.

## **Cronograma**



## **Detección de error de la instrucción aritmética**



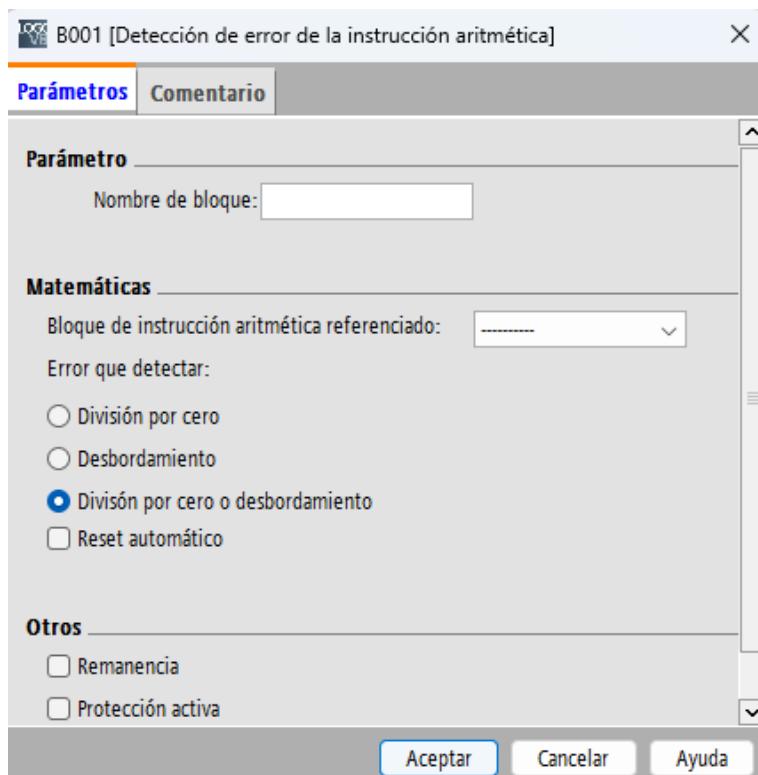
El bloque Detección de Error de la Instrucción Aritmética supervisa el correcto funcionamiento de un bloque de instrucción aritmética previamente referenciado. Su función consiste en activar una salida cuando se detecta un error en la operación aritmética ejecutada, ya sea por valores inválidos, desbordamientos u otras condiciones no permitidas por el cálculo.

Este bloque permite implementar mecanismos de seguridad, diagnóstico y manejo de fallos dentro del programa, proporcionando al usuario una señal inmediata que facilita la toma de acciones correctivas o la generación de alarmas en el sistema.

### **Pinout y descripción de conexión**

Entrada (En)	Habilitación del bloque de función de detección de error de la instrucción aritmética.
Entrada ®	Desactivación de la salida.
Parámetro (Par)	Configuración interna de los parámetros del bloque
Salida (Q)	Q se activa si el error detectado ha ocurrido en la última ejecución del bloque de función de instrucción aritmética referenciado.

## Párametros



FB referenciado: número de bloque de una instrucción aritmética

Error a detectar: división por cero, desbordamiento o bien división por cero O desbordamiento.

Reset automático: La salida se desactiva una vez corregido el error.

## Descripción de la función

El bloque de detección de error de la instrucción aritmética activa la salida si ocurre un error en el bloque de función de instrucción aritmética referenciado. La función se puede programar de manera que la salida se active cuando ocurra un error de división por cero, un error de desbordamiento, o cualquiera de estos errores.

Si activa la casilla de verificación "Reset automático", la salida se desactiva antes de la siguiente ejecución del bloque de función. Si no activa la casilla, la salida conserva su estado hasta que el bloque de detección de error de la instrucción aritmética sea desactivado con el parámetro R.

En cualquier ciclo, si el bloque de función de instrucción aritmética referenciado se ejecuta antes del bloque de detección de error de la instrucción aritmética, el error se detecta en ese mismo ciclo. Si el bloque de función de instrucción aritmética referenciado se ejecuta después del bloque de detección de error de la instrucción aritmética, el error se detecta en el siguiente ciclo.

## Tabla lógica para la detección de error de la instrucción aritmética

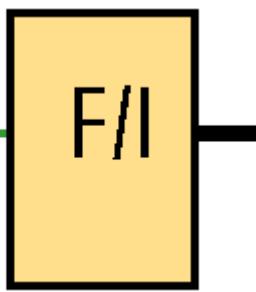
En la tabla siguiente, "Error a detectar" representa el parámetro de la instrucción de detección de error de la instrucción aritmética que indica qué error se debe detectar. Cero representa el bit de división por cero activado por la instrucción aritmética al final de su ejecución: 1 si ha ocurrido el error, 0 en caso contrario. OF es el bit de desbordamiento activado por la instrucción

aritmética: 1 si ha ocurrido el error, 0 en caso contrario. División por cero O desbordamiento representa la combinación lógica OR del bit de división por cero y del bit de desbordamiento de la instrucción aritmética referenciada. Q representa la salida de la función de detección de error de la instrucción aritmética. Una "x" indica que el bit puede ser 0 o 1 sin que ello tenga influencia en la salida.

Error a detectar	Cero	OF	Salida (Q)
División por cero	1	x	1
División por cero	0	x	0
Desbordamiento	x	1	1
Desbordamiento	x	0	0
División por cero O desbordamiento	1	0	1
División por cero O desbordamiento	0	1	1
División por cero O desbordamiento	1	1	1
División por cero O desbordamiento	0	0	0

## **Convertidor flotante/entero**

Par



El bloque Convertidor Flotante/Entero permite transformar un número en coma flotante, previamente almacenado en una marca de memoria (VM), en un valor entero que pueda ser procesado por el LOGO!. Debido a que el controlador solo trabaja internamente con enteros, los valores de coma flotante recibidos desde otros sistemas mediante protocolos como S7 o Modbus no pueden utilizarse directamente.

Este bloque realiza la conversión dividiendo el valor flotante entre una resolución configurada por el usuario, de manera que el resultado sea un número entero apto para su uso en cálculos, comparaciones o salidas analógicas (AQ o eAQ).

Para garantizar una conversión precisa, es necesario ajustar correctamente la resolución en la ficha de parámetros del bloque, tomando en cuenta el rango y la escala del valor flotante de entrada.

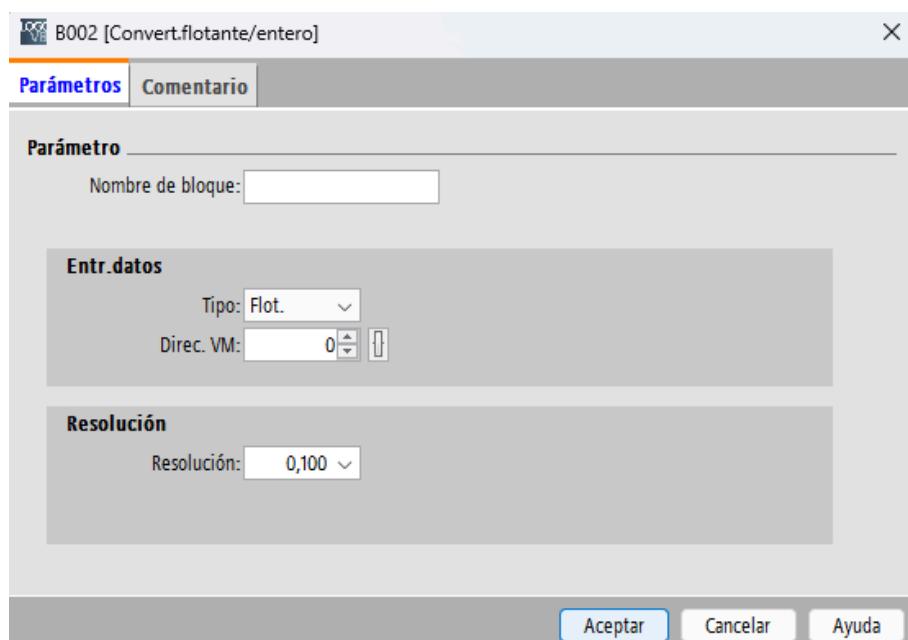
### **Pinout y descripción de conexión**

<b>Salida analógica (AQ)</b>	AQ es el valor de salida analógica. Tiene las características siguientes. <ul style="list-style-type: none"><li>• Se utiliza como valor de entrada del otro bloque de función.</li><li>• Valor de 16 bits con signo.</li><li>• Rango de valores: -32768 a 32767.</li></ul>
<b>Salida analógica ampliada (eAQ)</b>	Salida analógica ampliada para programación mediante parámetro de referencia. <ul style="list-style-type: none"><li>• Se utiliza como parámetro de referencia del otro bloque de función.</li><li>• Valor de 32 bits con signo.</li><li>• Rango de valores: de -999.999.999 a 999.999.999.</li></ul>

## **Parámetro (Par)**

- Tipo: especifica el tipo de los datos de entrada.
    - Flotante: es un número en coma flotante de 32 bits y precisión individual;
    - Doble: es un número en coma flotante de 64 bits y precisión doble.
  - VM: dirección de memoria variable, dirección inicial del flotante o doble almacenado en VM.
- Rango de valores:
- Para Flotante: 0-847
  - Para Doble: 0-843
- Resolución: es un divisor para datos de entrada.
  - Rango de valores: 0,001 a 1000

## **Parámetros**



Entrada de datos: la entrada para este SFB es desde VM.

- Tipo: pueden elegirse los datos de entrada como flotantes o dobles.
- Dirección VM: especifica la dirección VM inicial en la que se guarda el flotante de entrada.
  - Tamaño: el flotante de entrada ocupa 4 bytes y el doble, 8 bytes.
  - Rango: Float (0-847), Double (0-843).

Resolución: es posible ajustar el multiplicador para el flotante de entrada.

- Rango: 0.001-1000
- Precisión: 1/1000

### ***Descripción de la función***

Para finalizar la tarea suelen necesitarse tanto el convertidor flotante/entero como el convertidor entero/flotante. Un uso típico de este bloque de función es el siguiente:

1. Transferir los números en coma flotante de otro sistema a través de la red (con el protocolo S7/Modbus) y guardarlos en VM.
2. Convertir los números en coma flotante de VM a enteros utilizando el convertidor flotante/entero.
3. Procesar el entero con LOGO! BM.
4. Convertir el resultado a números en coma flotante utilizando el convertidor entero/flotante y guardarlos en VM.
5. Transferir los números en coma flotante a un sistema de terceros (con el protocolo S7/Modbus).

## **Regla de cálculo**

Definir Q = entrada de datos/resolución

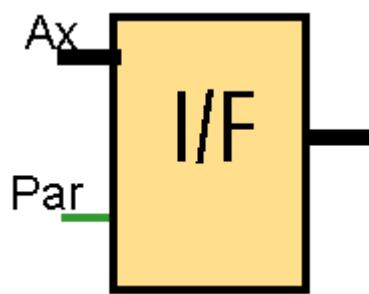
Salida analógica (AQ)

- si  $-32768 \leq Q \leq 32767$ , la salida analógica = Q.
- si  $Q \geq 32767$ , la salida analógica = 32767.
- si  $Q \leq -32768$ , la salida analógica = -32768.

Salida analógica ampliada (eAQ)

- si  $-999,999,999 \leq Q \leq 999,999,999$ , la salida analógica ampliada = Q.
- si  $Q \geq 999,999,999$ , la salida analógica ampliada = 999,999,999.
- si  $Q \leq -999,999,999$ , la salida analógica ampliada = -999,999,999.

## **Convertidor entero/floatante**



El bloque Convertidor Entero/Flotante permite transformar un valor entero utilizado internamente por el LOGO! en un número en coma flotante, el cual es almacenado en una marca de memoria (VM) para su transmisión a otros sistemas. Dado que el LOGO! solo procesa enteros, este bloque resulta esencial cuando se requiere enviar valores en formato flotante hacia dispositivos o sistemas externos mediante protocolos como S7 o Modbus.

La conversión se realiza multiplicando el valor entero por una resolución configurada por el usuario, generando así un número en coma flotante que representa adecuadamente la magnitud

original. Para obtener resultados precisos, es necesario definir una resolución adecuada en la ficha de parámetros del bloque, considerando el rango y la escala del valor entero de entrada.

### ***Pinout y descripción de conexión***

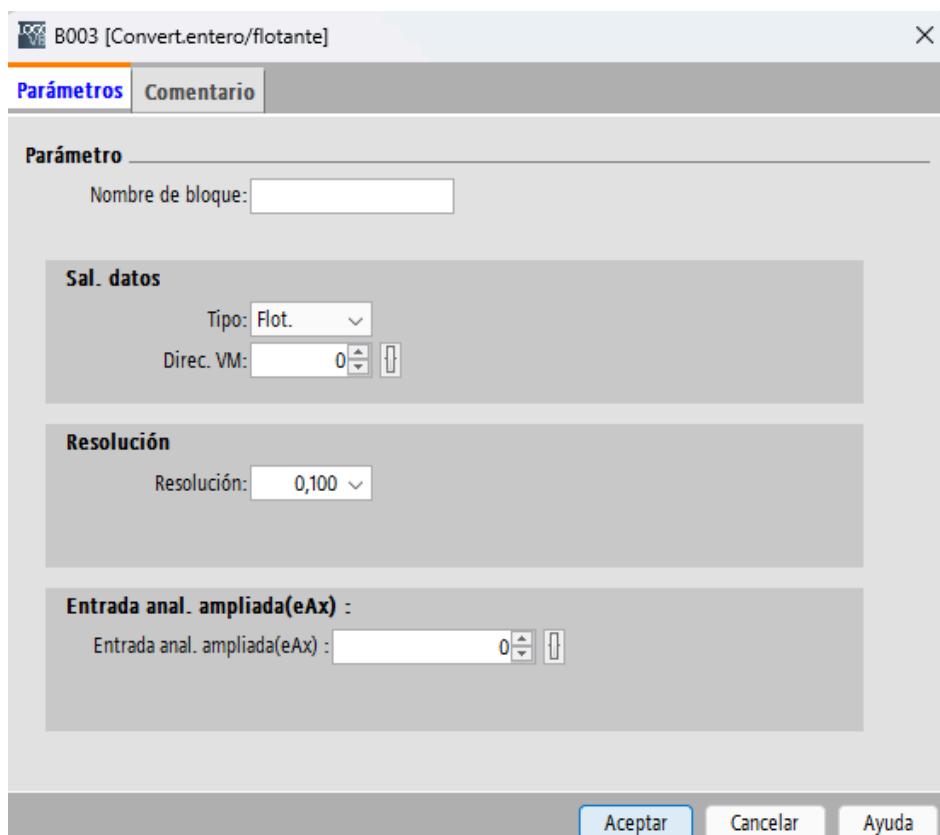
<b><i>Entrada analógica (Ax)</i></b>	<p>La entrada Ax es una de las siguientes señales analógicas:</p> <ul style="list-style-type: none"><li>• AI1 a AI8 (*)</li><li>• AM1 a AM16 (para 0BA7) o AM1 a AM64 (para 0BA8)</li><li>• NAI1 a NAI32</li><li>• AQ1 a AQ2 (para 0BA7) o AQ1 a AQ8 (para 0BA8)</li><li>• NAQ1 a NAQ16</li><li>• Número de bloque de una función con salida analógica</li></ul>
<b><i>Entrada analógica ampliada (eAx)</i></b>	<p>Si la entrada analógica (Ax) no está disponible, es posible asignar un valor a la entrada analógica ampliada (eAx) introduciendo un valor en el campo eAx o referenciando otro parámetro del FB.</p> <p>Rango de valores: de -999.999.999 a 999.999.999.</p>

<b>Parámetro (Par)</b>	<ul style="list-style-type: none"> <li>• Tipo: Especifica el tipo de los datos de salida.             <ul style="list-style-type: none"> <li>◦ Flotante: es un número en coma flotante de 32 bits y precisión individual;</li> <li>◦ Doble: es un número en coma flotante de 64 bits y precisión doble.</li> </ul> </li> <li>• VM: Dirección de memoria variable, dirección inicial del flotante o doble almacenado en VM:             <ul style="list-style-type: none"> <li>◦ Para Flotante: 0-847</li> <li>◦ Para Doble: 0-843</li> </ul> </li> <li>• Resolución: es un multiplicador de los datos de entrada. Rango de valores: 0,001 a 1000</li> </ul>
<b>Salida (AQ)</b>	<p>AQ es el valor de salida analógica. Tiene las características siguientes.</p> <ul style="list-style-type: none"> <li>• Se utiliza como valor de entrada del otro bloque de función.</li> <li>• Valor de 16 bits con signo.</li> <li>• Rango de valores: -32768 a 32767.</li> </ul>
<b>Salida analógica ampliada (eAQ)</b>	<p>Salida analógica ampliada para programación mediante referencia a parámetro.</p> <ul style="list-style-type: none"> <li>• Se utiliza como parámetro de referencia del otro bloque de función.</li> <li>• Valor de 32 bits con signo.</li> </ul>

- Rango de valores: de -999.999.999 a 999.999.999.

\* AI1 a AI8: 0 a 10 V es proporcional a 0 a 1000 (valor interno).

## Parámetros



Salida de datos: la salida del entero/floatante se guarda en VM. Es posible elegir el tipo de salida como flotante o doble y la dirección VM inicial en la que se guarda el floatante convertido.

- Tipo: pueden elegirse los datos de salida como flotantes o dobles.
- Dirección VM: especifica la dirección VM inicial en la que se guarda el floatante de salida.
  - Tamaño: el floatante de entrada ocupa 4 bytes y el doble, 8 bytes.
  - Rango: Float (0-847), Double (0-843).

Resolución: es posible ajustar el dividendo para el entero de entrada.

- Rango: 0.001-1000
- Precisión: 1/1000

Entrada analógica ampliada: aquí también puede ajustarse el valor de entrada. Si no se conecta el pin de entrada In, el SFB utiliza como entrada el valor ajustado aquí. También puede utilizarse como referencia para otro valor de parámetro, como un valor de contador.

- Rango: de -999.999.999 a 999.999.999

### ***Descripción de la función***

Para finalizar la tarea suelen necesitarse tanto el convertidor flotante/entero como el convertidor entero/flotante. Un uso típico de este bloque de función es el siguiente:

1. Transferir los números en coma flotante de otro sistema a través de la red (con el protocolo S7/Modbus) y guardarlos en VM.
2. Convertir los números en coma flotante de VM a enteros utilizando el convertidor flotante/entero.
3. Procesar el entero con LOGO! BM.
4. Convertir el resultado a números en coma flotante utilizando el convertidor entero/flotante y guardarlos en VM.
5. Transferir los números en coma flotante a un sistema de terceros (con el protocolo S7/Modbus).

### ***Regla de cálculo***

Valor flotante en dirección VM

Valor flotante en dirección VM = entrada analógica x resolución

Salida analógica (AQ)

- El conector de la entrada analógica está enchufado: Salida analógica = entrada analógica
- El conector de la entrada analógica no está enchufado:

- si  $-32768 \leq$  entrada analógica ampliada  $\leq 32767$ , la salida analógica = entrada analógica ampliada.
- si la entrada analógica  $\geq 32767$ , la salida analógica = 32767.
- si la entrada analógica  $\leq -32768$ , la salida analógica = -32768.

Salida analógica ampliada (eAQ):

- Si el conector de la entrada analógica está enchufado: Entrada analógica ampliada = entrada analógica.
- El conector de la entrada analógica ampliada no está enchufado:
  - si  $-999.999.999 \leq$  entrada analógica ampliada  $\leq 999.999.999$ , la salida analógica = entrada analógica ampliada.
  - si la entrada analógica ampliada  $\geq 999.999.999$ , la salida analógica ampliada = 999.999.999.
  - si la entrada analógica ampliada  $\leq -999.999.999$ , la salida analógica ampliada = -999.999.999.