

# SIEMENS

## Microsistema SIMATIC S7-200

El S7-200 en dos horas

Edición 01/2000



microsystems

**Consignas de seguridad** "El S7-200 en dos horas" ha sido creado para introducirse rápidamente en el mundo del micro-PLC S7-200. Su contenido es breve, lo que se ha hecho conscientemente. No puede considerarse como sustituto del Manual del S7-200.

Por ello rogamos respetar imprescindiblemente todas las advertencias e indicaciones contenidas en el Manual del S7-200, particularmente las consignas de seguridad.

---

## **Marcas**

SIMATIC® y SIMATIC NET® son marcas registradas de Siemens AG.

Las designaciones en esta publicación pueden ser marcas cuyo empleo por parte de terceros, para sus fines, puede infringir los derechos de sus titulares.

Copyright © Siemens AG 2000 All rights reserved

La divulgación y reproducción de este documento, así como el uso y la comunicación de su contenido no están autorizados, a no ser que se obtenga el consentimiento expreso para ello. Los infractores quedan obligados a la indemnización de los daños.

Se reservan todos los derechos, en particular para el caso de concesión de patentes o de modelos de utilidad.

Siemens AG  
Automatización y Accionamientos  
Subdivisión Sistemas de automatización industrial,  
Postfach 4848, D-90327 Nuernberg

## **Exención de responsabilidad**

Hemos verificado que el contenido de esta publicación concuerde con lo descrito para el hardware y el software. Sin embargo, es posible que haya algunas desviaciones que no se impiden tomar la garantía completa por esta concordancia. El contenido de esta publicación está sometido a revisiones regularmente; en caso necesario se incluyen las correcciones en la siguiente edición. Agradecemos sugerencias de mejora.

© Siemens AG 2000

Se reserva el derecho para la realización de cambios técnicos

# Indice

Repaso	Repasemos algunas cosas	5
	Aquí están los bits	6
	Circulación de corriente en KOP	7
	El ciclo del PLC	9
Autorretención	Introducción	13
	Contacto normalmente cerrado	14
	Descripción de la solución y test	16
	Variante de autorretención...	17
Telerruptor	Introducción	21
	Solución resumida	22
	Detección de flancos	23
	Marcas	25
	Descripción de la solución y test	27
Retardo a la desconexión	Introducción	29
	Guardar como...	31
	Insertar segmento	32
	Descripción de la solución	33
	Introducir comentarios	36
Secuencia	Introducción	39
	Fundamentos	41
	Uso de secuencias	45
	Modificación	50
	Descripción de la solución, ejemplo	51
	Test	55
Anexo	Consejos	A1
	Indice alfabético	B1



Esta relación del contenido de "El S7-200 en dos horas" la encontrará en el pie de todas las páginas.  
Se resalta el capítulo abierto en cada caso.

# Prefacio

Estimado usuario del S7-200:

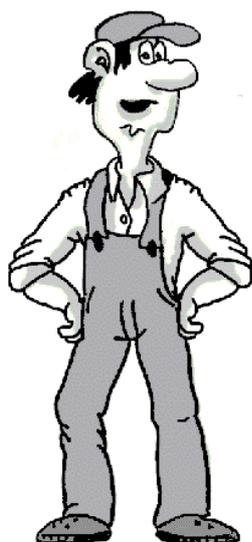
La eficiencia en la aplicación de un micro-PLC moderno depende esencialmente de la velocidad y seguridad con que un usuario puede aplicar dicho equipo de control. Para que incluso un principiante pueda aprender a aplicar rápida y simplemente un S7-200 hemos creado las guías denominadas "El S7-200 en una hora" y "El S7-200 en dos horas".

"El S7-200 en dos horas" que tiene en sus manos le permitirá, basándose en el contenido de "El S7-200 en una hora", familiarizarse en poco tiempo con el funcionamiento del PLC S7-200. En base a algunos ejemplos, esta guía le mostrará la forma de funcionamiento del PLC así como la manera más eficiente de utilizarlo para aplicaciones sencillas.

Una vez estudiado "El S7-200 en dos horas" le será fácil resolver por su cuenta tareas de control típicas.

Esperamos que se lo pase bien durante la lectura.

En el anexo encontrará un disquete del que podrá cargar los ejemplos tratados.



¡El equipo S7-200 le desea mucho éxito!

Repaso

Autorretención

Telerruptor

Retardo a la desconexión

Secuencia

Anexo

1

2

Repaso



Autorretención



Telerruptor



Retardo a la  
desconexión



Secuencia



Anexo

## Símbolos utilizados

En "El S7-200 en dos horas" se utilizan con frecuencia símbolos y formatos de resalte de texto cuyo significado se explica brevemente en esta página.

Sin ir más lejos, empiece por el encabezamiento de la página. En principio, cada página tiene un encabezamiento idéntico. El título en letras grandes grises identifica el apartado actual del capítulo. La zona "Novedades" en la parte derecha muestra el contenido de las páginas anteriores, en negro, el contenido de la página actual y, finalmente, el contenido de la(s) página(s) siguiente(s).

*Los textos resaltados en gris le invitan a actuar, p. ej. introducir algo.*



Este símbolo indica que hay que apretar (clic) una vez la tecla izquierda del ratón para confirmar una acción (p. ej. marcar campo).



2 x

Este símbolo muestra que es necesario pulsar rápidamente (doble clic) dos veces la tecla izquierda del ratón para ejecutar una acción.



En este caso es necesario pulsar la tecla ENTER (RETURN) de su teclado.



Esto significa que puede seleccionar puntos de una lista en pantalla tanto con el ratón como con el teclado (teclas de función, teclas de cursor).

F2

Esto significa que debe pulsar la tecla de función "F2" (existen las teclas de función F1 ... F12). Se dará cuenta que a pesar de que el ratón ofrece gran comodidad, el teclado resulta más rápido en determinadas situaciones.



Esto combinado con una remisión a una determinada página le indica que allí encontrará mayor información sobre un determinado tema.



En estos puntos es invitado a introducir textos en campos del interface del usuario. O se le recuerda de que conviene registrar comentarios para sus propios proyectos.



Menú

Un comando de menú del interface de usuario se activa paso a paso (general, particular) con la tecla izquierda del ratón.

4

Repaso



Autorretención



Telerruptor



Retardo a la  
desconexión



Secuencia



Anexo

- Repasemos algunas cosas
- Aquí están los bits
- Circulación de corriente en KOP
- El ciclo del PLC

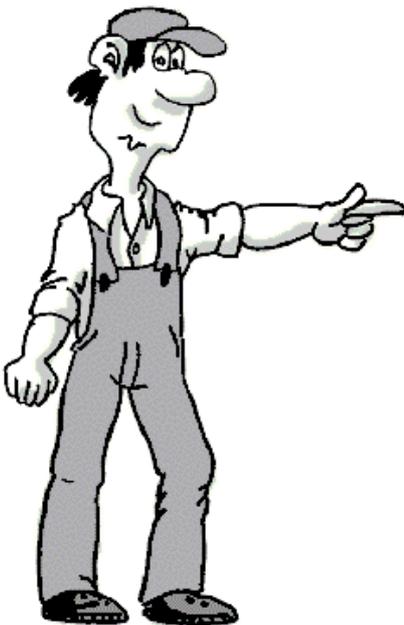


## Repasemos algunas cosas ...

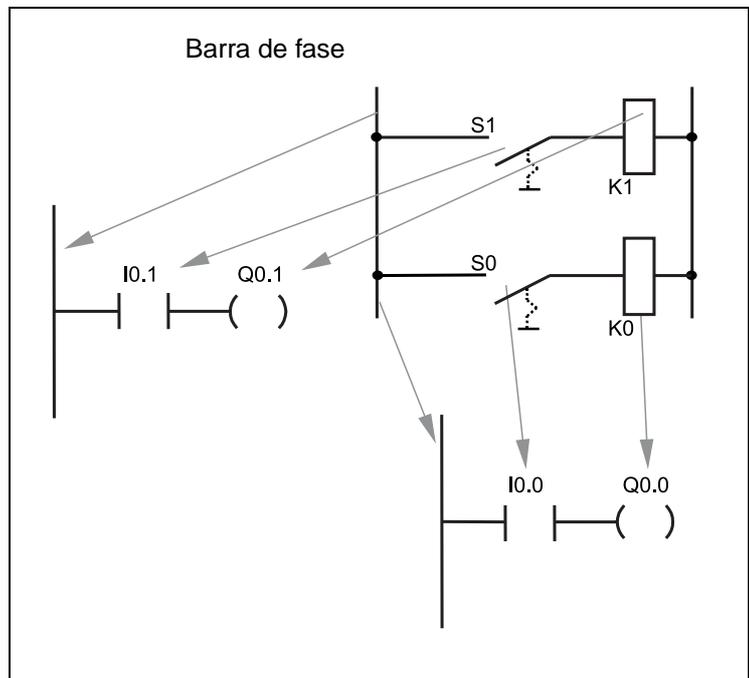
En "El S7-200 en una hora" pudo ver que el esquema eléctrico de los mandos con contactores es parecido al esquema de contactos (KOP) destinado a programar autómatas (PLC).

En principio se trata de la misma representación pero con diferentes símbolos.

También aprendió a programar pequeñas combinaciones, es decir operaciones lógicas. Incluso aprendió en poco tiempo lo que eran temporizadores.



Comparar con "EL S7-200 en una hora", página 24:



- Repasemos algunas cosas
- Aquí están los bits
- Circulación de corriente en KOP
- El ciclo del PLC

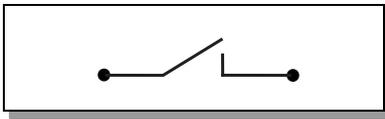


## Aquí están los bits

El bit es la menor unidad que puede procesarse.  
Un bit puede adoptar dos estados:

- 1) "1" es decir, "bit activado" o estado "verdadero",
- 2) "0" significa "bit desactivado" o estado "falso".

Los dos estados binarios "1" y "0" pueden representarse también, de una forma más familiar, como circuitos eléctricos, es decir mediante interruptores.

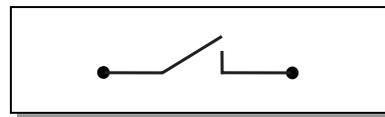


Un interruptor cerrado significa:  
circula corriente, es decir estado bit = "1"



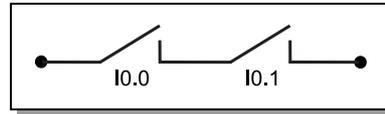
"1" = "verdadero" = circula corriente

y un interruptor abierto:  
no circula corriente, es decir estado bit = "0".



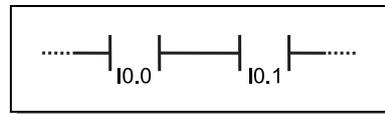
"0" = "falso" = no circula corriente

Basta un pequeño paso para poder representar también combinaciones lógicas, p. ej. la conexión en serie de dos contactos. A la derecha se muestra la combinación Y de las entradas I0.0 e I0.1.



Combinación Y

En representación KOP:



Finalicemos con un pequeño convenio.

En lógica positiva se tiene:

- 24 V = nivel High/alto = "1" y
- 0 V = nivel Low/bajo = "0".

Lógica positiva

En lógica negativa se tiene:

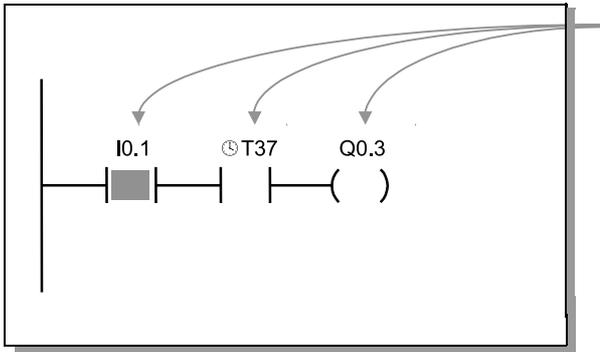
- 0 V = nivel Low/bajo = "1"
- 24 V = nivel High/alto = "0".

Lógica negativa

- Repasemos algunas cosas
- Aquí están los bits
- Circulación de corriente en KOP
- El ciclo del PLC

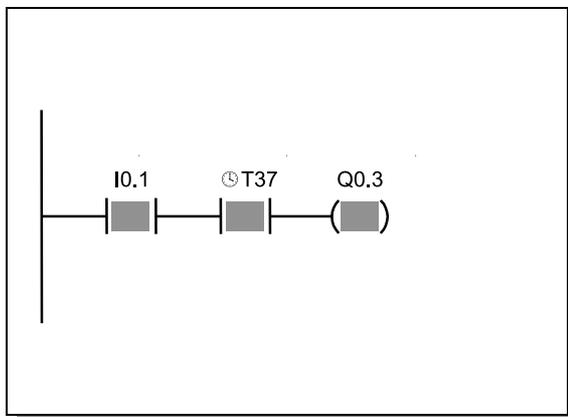


## Circulación de corriente en KOP (1)



En este ejemplo, la salida Q0.3 se conecta, o se hace "1", cuando se cierra el contacto en I0.1, es decir se hace "1" (24 V DC en la entrada I0.1) Y simultáneamente está activado el bit de tiempo T37, es decir está a "1".

La entrada I0.1 está ahora a "1", es decir está cerrado el contacto I0.1. En la figura no está activado T37, es decir está a "0". Por ello S0.3 está también desactivada, es decir a "0".



Cuando el temporizador T37 se hace también "1" (T37 ha transcurrido), se hace "1" el resultado de la combinación Y, y con ello también la salida Q0.3.

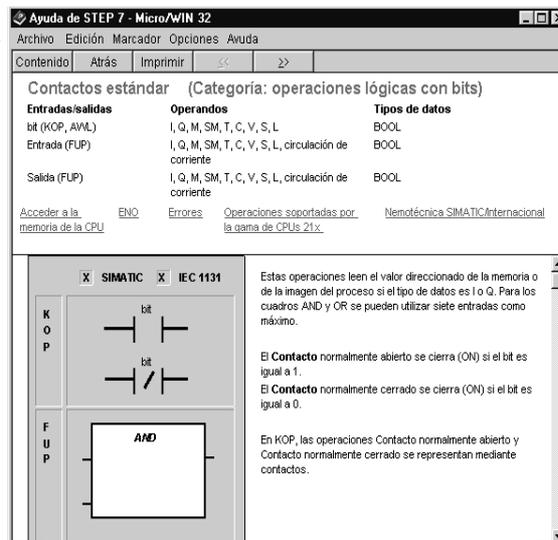
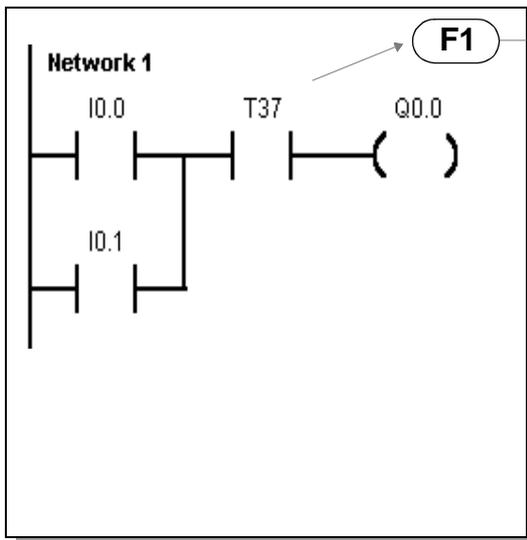
Entonces el bit de salida es también "verdadero", tiene el valor "1" (se muestra sombreado).

Esto corresponde a la indicación de estado KOP que ya utilizó en "El S7-200 en una hora" para probar sus programas.

- Repasemos algunas cosas
- Aquí están los bits
- Circulación de corriente en KOP
- El ciclo del PLC



## Circulación de corriente en KOP (2) (usar la función de ayuda)



Visualizar  
Ayuda

1. Marcar  
elemento

2. F1

Si desea ver nuevamente la ayuda online relativa a un símbolo de contacto u otras funciones:

Marcar el contacto

- en el esquema de contactos (KOP) o
- en el diagrama de funciones (FUP) o bien
- marcar el contacto en su KOP de STEP 7-Micro/WIN

haciendo clic con el ratón y pulsar seguidamente la tecla F1.

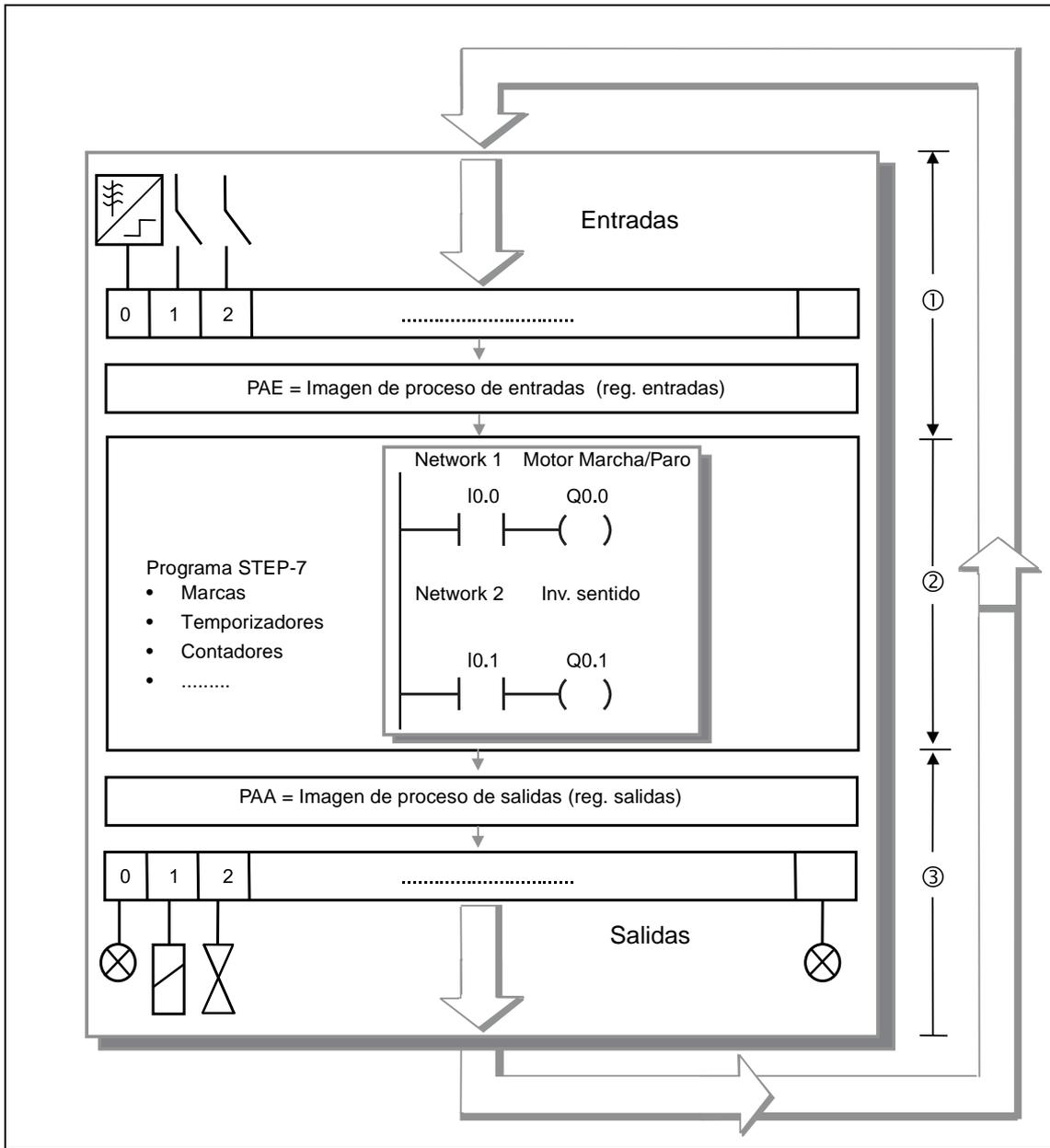
F1 Ayuda online



- Repasemos algunas cosas
- Aquí están los bits
- Circulación de corriente en KOP
- El ciclo del PLC



## El ciclo del PLC (1)



Por principio, todos los PLC SIMATIC trabajan de forma cíclica. Durante el funcionamiento cíclico, primero se leen los estados en las entradas, memorizándose en la imagen de proceso de las entradas (PAE). Con estas informaciones trabaja luego el programa de control cuando se ejecuta.

Imagen de proceso de las entradas: PAE

- Repasemos algunas cosas
- Aquí están los bits
- Circulación de corriente en KOP
- El ciclo del PLC



## El ciclo del PLC (2)

De acuerdo a la lógica definida en el programa se modifica el estado de las salidas depositadas en la imagen de proceso de las salidas (PAA). En la última etapa del ciclo, los estados memorizados en la PAA se transfieren a las salidas físicas. Seguidamente comienza de nuevo el ciclo.

Imagen de proceso de las salidas: PAA

Un ciclo dura normalmente entre 3 y 10 ms. La duración depende del número y tipo de instrucciones (operaciones) utilizadas.

El ciclo consta de dos partes principales:

- 1) Tiempo del sistema operativo, normalmente 1 ms; corresponde con las fases ① y ③, página 9.
- 2) Tiempo para ejecutar las instrucciones; corresponde con la fase ②, página 9.

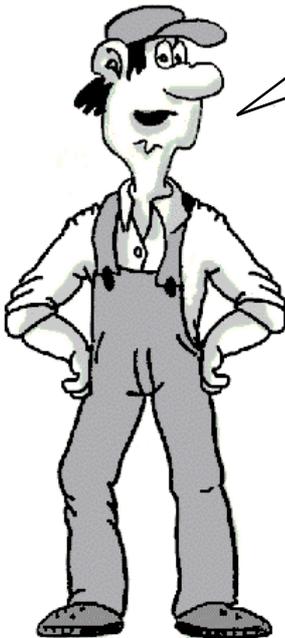
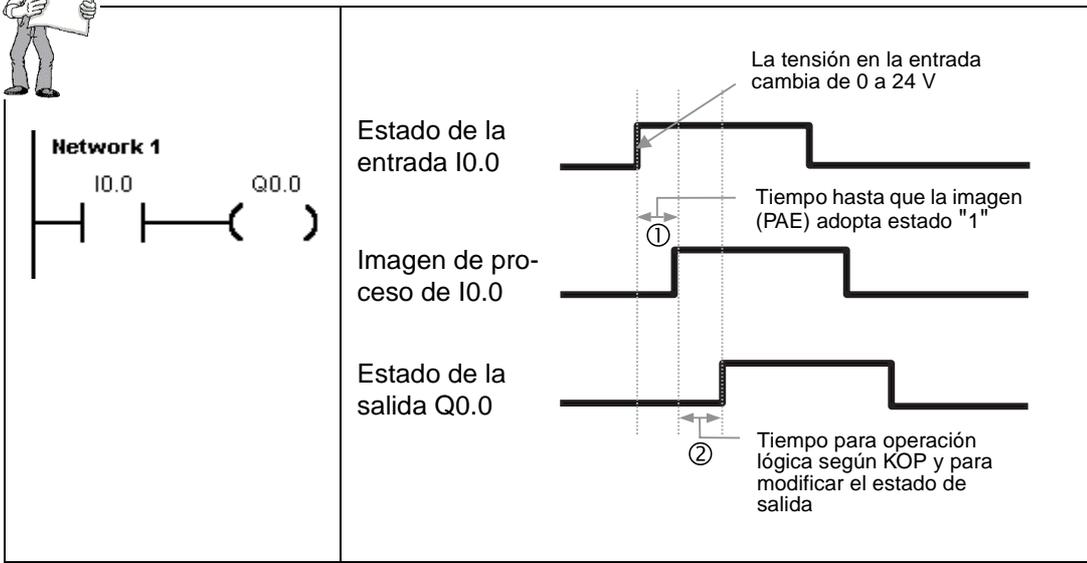
Por otro lado, el ciclo sólo se ejecuta cuando "trabaja" el PLC, es decir cuando se encuentra en el estado "RUN".



- Repasemos algunas cosas
- Aquí están los bits
- Circulación de corriente en KOP
- El ciclo del PLC



## El ciclo del PLC (3)



Los cambios que se produzcan en las entradas durante un ciclo sólo se memorizan en el registro de entrada durante el ciclo siguiente. Es decir se "congelan" mientras dura el ciclo en cuestión. Dicho registro se denomina también "Imagen de proceso de las entradas PAE" (ver ①).

Durante el siguiente ciclo los estados adoptados se combinan lógicamente de acuerdo al esquema de contactos (ver ②), actualizándose las salidas de acuerdo a los resultados lógicos.

Cambio de las salidas sólo al finalizar el ciclo siguiente

Repaso

Notas

12

Repaso

Autorretención

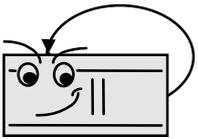
Telerruptor

Retardo a la  
desconexión

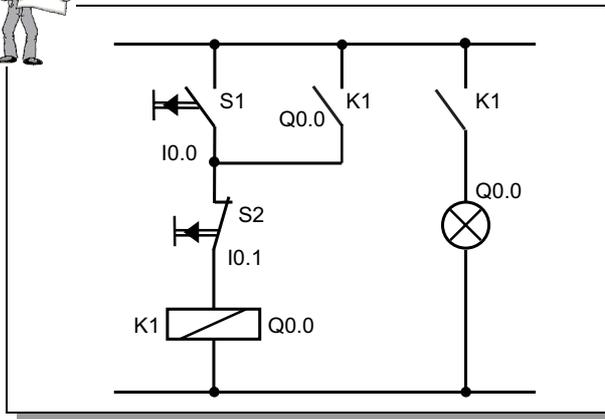
Secuencia

Anexo

- Introducción
- Contacto normalmente cerrado
- Descripción de la solución y test
- Variante de autorretención



## Introducción



Aquí le mostramos la forma de programar la conocida función de autorretención de los mandos por contactores.

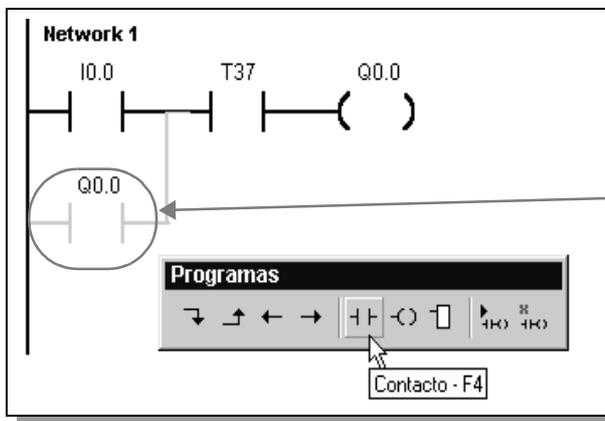
Autorretención estándar

Trataremos el ejemplo siguiente:

Deberá conectarse la salida Q0.0 tan pronto como se pulse S1 conectado a la entrada I0.0. Por autorretención, Q0.0 deberá seguir activada hasta que se pulse S2 conectado a la entrada I0.1, interrumpiéndose así la autorretención.



En STEP 7-Micro/WIN, abra el primer proyecto de ejercicio del disquete "a:\d01.prj". En el programa faltan un par de elementos. Como ejercicio, inserte ahora los elementos KOP que faltan.



Para que funcione la autorretención, la salida (en este caso Q0.0) deberá encargarse ella misma de que, tan pronto como se active, mantenga su estado "verdadero".

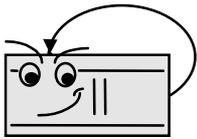
Salida Q0.0 como entrada procura la autorretención

Esto se realiza conectando la salida (aquí Q0.0) como contacto en paralelo con la entrada activadora, es decir igual a como se resuelve en un circuito con contactor convencional (Q0.0 es comparable a nuestro contactor K1).

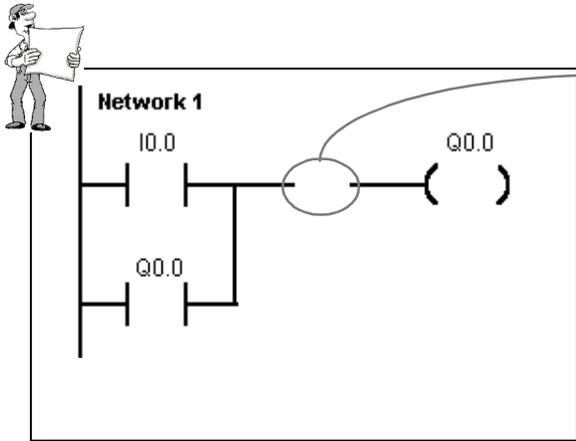
Inserte primero en el punto marcado, en paralelo a I0.0, un contacto Q0.0 (se muestra en gris). Forma de insertar el contacto:

- 1) Con la tecla izquierda del ratón haga clic en el campo del esquema de contactos y pulse el botón de contacto normalmente abierto (F4) de STEP 7-Micro/WIN. Como se muestra en el botón, en lugar del ratón puede utilizar también la tecla de función F4.
- 2) Para introducir la línea vertical, marque con el ratón en el campo del esquema de contacto de I0.0 y pulse el botón (F7).

- Introducción
- **Contacto normalmente cerrado**
- Descripción de la solución y test
- Variante de autorretención



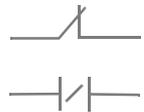
## Contacto normalmente cerrado (1)



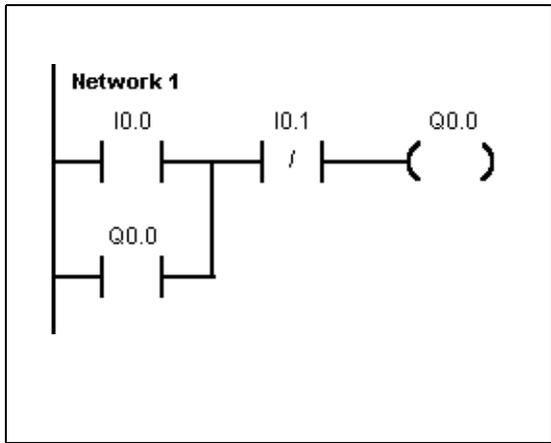
Para poder anular la autorretención, la entrada I0.1 deberá actuar, si se activa, interrumpiendo el circuito. Cuando un circuito se interrumpe (es decir resulta estado "0") al activar un interruptor se habla de un contacto normalmente cerrado o de apertura.

Por ello es necesario insertar un elemento que actúe en el esquema de contactos como un contacto NC cuando haya aplicados 24 V ("verdadero") en la entrada I0.1.

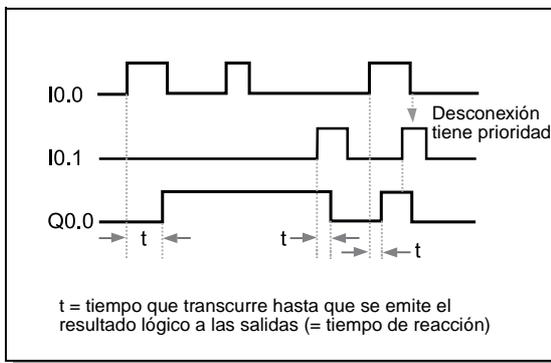
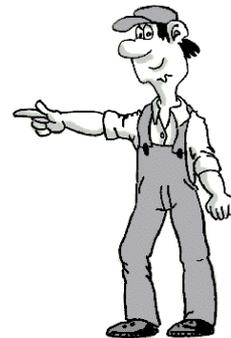
Contacto NC:



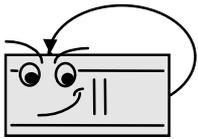
Inserte un contacto normalmente cerrado para el pulsador S1 en I0.1. Esto se describe en la página siguiente.



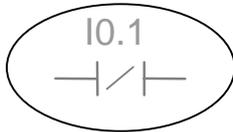
Este es el aspecto que tiene la autorretención ya terminada Su funcionamiento puede verse en el cronograma



- Introducción
- **Contacto normalmente cerrado**
- Descripción de la solución y test
- Variante de autorretención

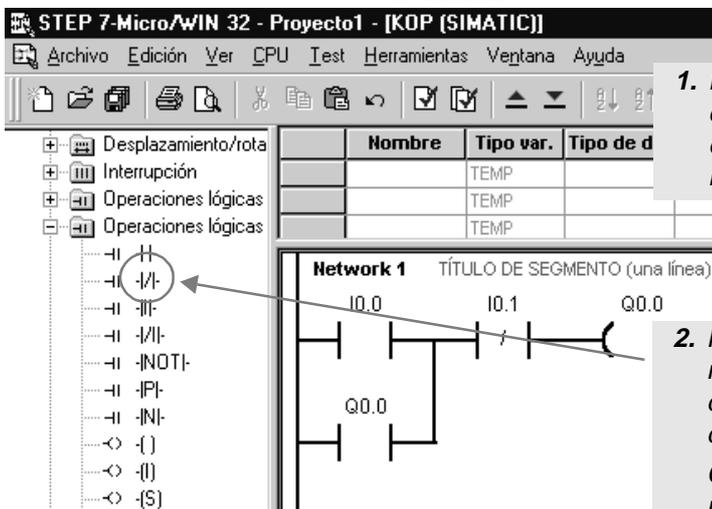


## Contacto normalmente cerrado (2)



Un contacto normalmente cerrado interrumpe la "circulación de corriente" en esquema de contactos cuando es "verdadera" la entrada o salida a él asignada.

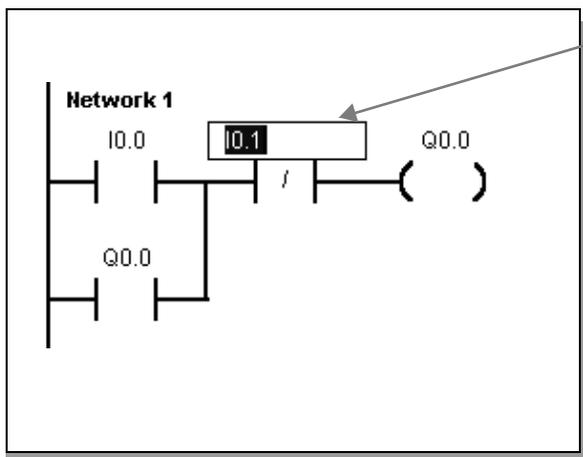
Forma de introducir un contacto normalmente cerrado:



1. Haga clic en el punto en el que desea sustituir el elemento existente por un contacto normalmente cerrado.



2. Haga clic en el botón de contacto normalmente cerrado en una de las dos barras de herramientas disponibles en STEP 7-Micro/WIN. Con ello se coloca en el campo marcado el contacto normalmente cerrado.



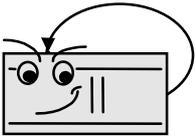
3. Finalmente es necesario asignar al contacto normalmente cerrado el elemento deseado (aquí IO.1). Esto se hace escribiendo en el campo de texto ya activado y marcado.



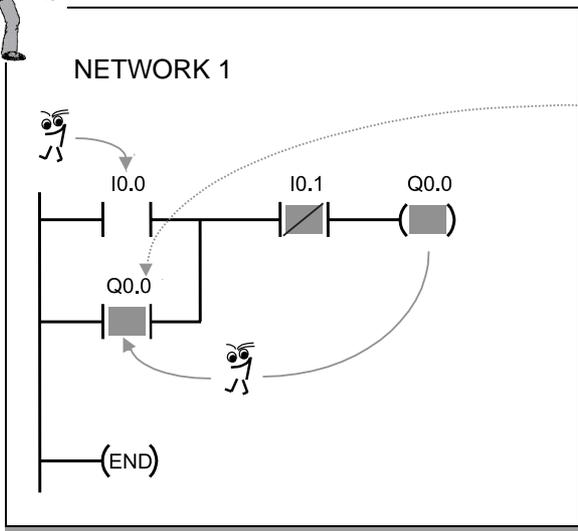
4. Una introducción en un campo de texto debe finalizarse siempre pulsado la tecla Enter



- Introducción
- Contacto normalmente cerrado
- Descripción de la solución y test
- Variante de autorretención



## Descripción de la solución y test



Al igual que en el circuito con contactor, también aquí se ha colocado en paralelo al elemento activador (Q0.0) un contacto de salida (I0.0). La salida Q0.0 paralela a la entrada se autorretiene

Si en un ciclo, al accionar el pulsador S1 conectado a I0.0 se ha activado la salida Q0.0, ya durante el ciclo siguiente (tras pocos milisegundos) aparecerá cerrado el contacto Q0.0 conectado en paralelo a I0.0. Esto hace que actúe la autorretención. El contacto normalmente cerrado I0.1 permite, si se actúa sobre el pulsador S2 asociado y conectado a I0.1, anular la autorretención.



Guardar el programa completo en el disco duro. Esto permite volverlo a cargar completamente para editarlo (lo necesitaremos de nuevo para el ejemplo "Retardo a la desconexión").

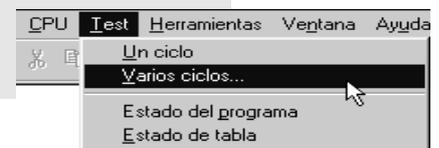


Seguidamente, transfíeralo al PLC para probar la función.

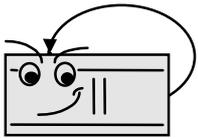


Para efectuar el test, pasar el PLC al estado "RUN".

Pruebe su programa accionando los dos interruptores del simulador conectados a I0.0 e I0.1. Observar las lámparas del S7-200 y el estado KOP.  
Comience conectando I0.0.  
I0.1 deberá estar desconectada. El LED en I0.0 deberá lucir.  
Seguidamente lucirá Q0.0.  
Tan pronto como se conecte I0.1, Q0.0 = "0".



- Introducción
- Contacto normalmente cerrado
- Descripción de la solución y test
- Variante de autorretención

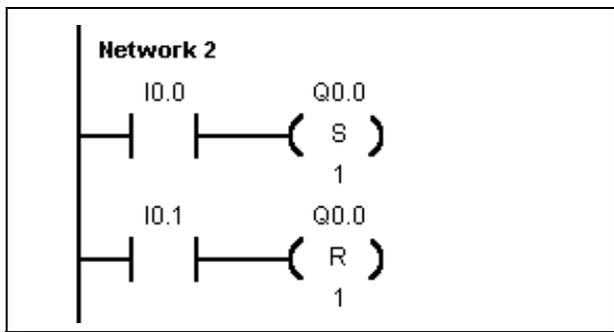


## Variante de autorretención ... (1)



En PLCs, una autorretención se programa también con frecuencia con otra variante:

En lugar de realimentar la salida - como en el ejemplo anterior - se recurre simplemente a las funciones "Poner a 1" y "Poner a 0".  
Veamos en primer lugar el esquema de contactos.



Un impulso en I0.0 permite, gracias a la operación "Poner a 1" - (S), que se conecte permanentemente Q0.0.

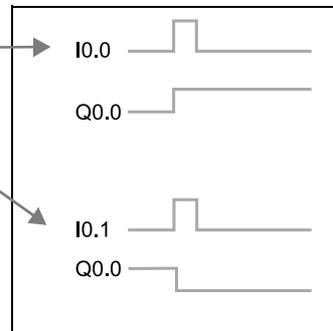
-(S)  
Poner a 1

En cambio, un impulso en I0.1 hace - gracias a la operación "Poner a 0" - (R), que Q0.0 vuelva a desconectarse.

-(R)  
Poner a 0

Las "bobinas" - (S) Poner Q0.0 a "1"  
- (R) Poner Q0.0 a "0"

se utilizan con frecuencia en PLCs para mantener permanentemente activadas o desactivadas entradas, salidas o marcas cuando se active brevemente (por impulso) un contacto antepuesto.



-(S) ⇔ 1

-(R) ⇔ 0

Una salida o marca "puesta a 1" permanece en ese estado hasta que sea borrada por la instrucción - (R) ("falsa").

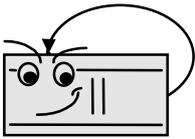
Poner a 1 permanente con (S)  
Poner a 0 con (R)

Si en la bobina de poner a 1 y en su bobina asociada de poner a 0 de una salida se aplica señal "1", tiene prioridad la operación que está después en el programa.

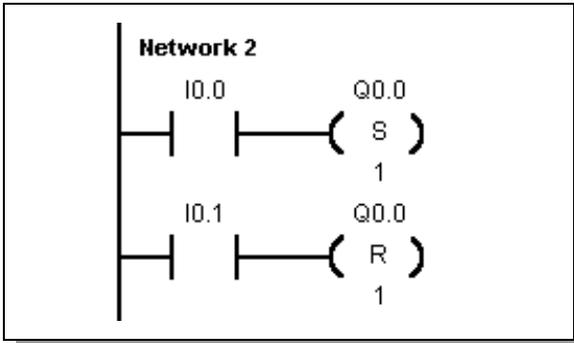
Tiene prioridad la última operación en el ciclo



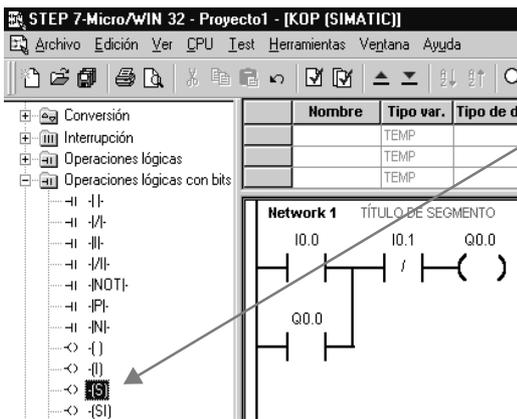
- Introducción
- Contacto normalmente cerrado
- Descripción de la solución y test
- Variante de autorretención



## Variante de autorretención ... (2)



La forma de introducir I0.0 e I0.1 ya la ha aprendido. Seguidamente mostraremos la forma de introducir la bobina de poner a 1 ó 0:



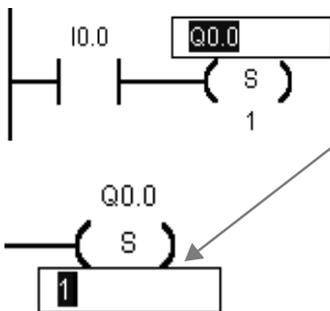
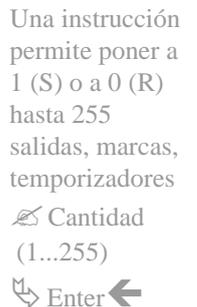
1. Tras marcar el campo KOP deseado, haga clic en una de las "bobinas" de la lista desplegable de familias de operaciones.



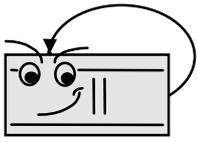
2. Introduzca en el campo de texto ya activado la dirección de salida a modificar, aquí Q0.0.



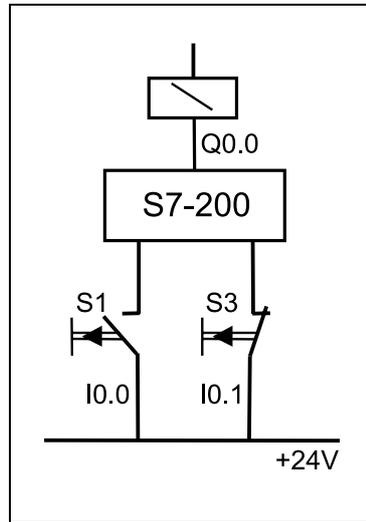
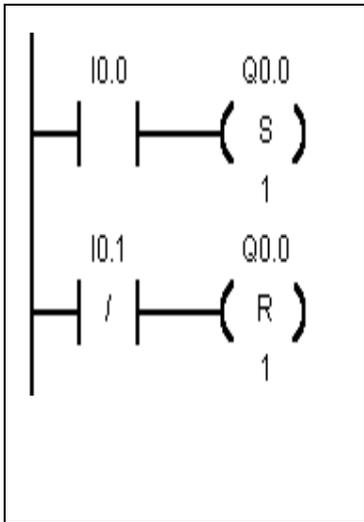
3. Seguidamente defina la cantidad (máx. 255) de salidas, temporizadores o marcas que deben modificarse simultáneamente a partir de la dirección seleccionada (en este caso sólo un bit).



- Introducción
- Contacto normalmente cerrado
- Descripción de la solución y test
- Variante de autorretención



## Aspectos de seguridad: Desconectar en caso de rotura del cable de conexión a S3 (3)



S3 es un pulsador con contacto normalmente cerrado que suministra señal "0" al accionarlo.

En KOP, esta señal se invierte mediante un contacto normalmente cerrado

—|/|— I0.1. Es decir, Q0.0 se pone a 0 cuando se aprieta el pulsador S3.



### Indicaciones relativas a la seguridad

- En la solución anterior, para poner a 0 se utiliza un pulsador S3 con contacto NC. Al activar I0.0 se pone permanentemente a 1 la salida Q0.0. Si en I0.1 están aplicados +24 V, el contacto "NC" en KOP entrega el estado "0". No se pone a 0 la salida Q0.0. Con ello queda interrumpida la "circulación de corriente" en KOP, desactivándose la bobina de puesta a 0. Si no hay señal (0 V) aplicada en I0.1 (S3 está abierto), el contacto NC de I0.1 en KOP = "1" y se pone a 0 la salida.

Si se utiliza un pulsador con contacto NC en I0.1, la salida Q0.0 autorretenida se pone a 0 (vuelve a desconectar):

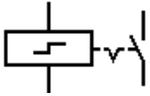
- cuando se activa el pulsador S3 (I0.1 = "0") o
- se interrumpe el cable entre I0.1 y el pulsador con contacto NC. Es decir, si se rompe el cable también queda garantizada la desconexión del elemento que se manda con autorretención, p. ej. un motor.
- La operación "Poner a 0 Q0.0" se introdujo después de la operación "Poner a 1 Q0.0", por lo que, si se activan simultáneamente ambos pulsadores, tiene prioridad la anulación de la autorretención.



En STEP 7-Micro/WIN, abra en el disquete el ejemplo "a:\d02.prj" y pruebe las funciones.



- **Introducción**
- Solución resumida
- Detección de flancos
- Marcas
- Descripción de la solución y test

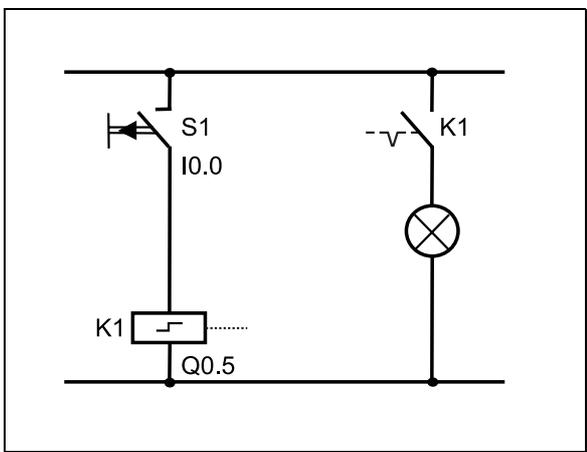


## Introducción



Aquí vamos a materializar un tipo de circuito denominado "telerruptor". En este contexto se familiarizará con la detección de flancos y las marcas.

### Modo de funcionamiento



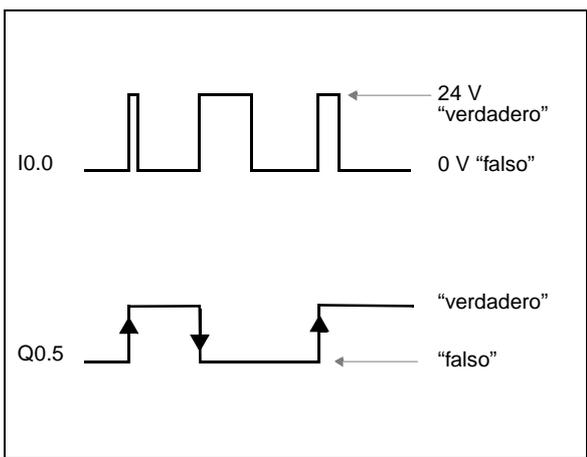
Se desea encender una lámpara conectada a la salida Q0.5 tan pronto como se apriete brevemente el pulsador S1 conectado a la entrada I0.0.

Si se activa nuevamente S1 (I0.0), Q0.5 deberá desactivarse y apagarse así la lámpara.

Cada vez que se active el pulsador S1 deberá invertirse el estado de Q0.5.

En este caso se trata de un tipo de circuito denominado "telerruptor" o biestable por impulso de corriente.

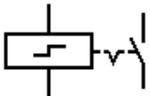
### Cronograma



La salida Q0.5 deberá invertir **una vez** su estado cuando el pulsador conectado a I0.0 pase de "abierto" a "cerrado".

Si el pulsador permanece cerrado o abierto no deberá producirse cambio alguno.

- Introducción
- **Solución resumida**
- Detección de flancos
- Marcas
- Descripción de la solución y test



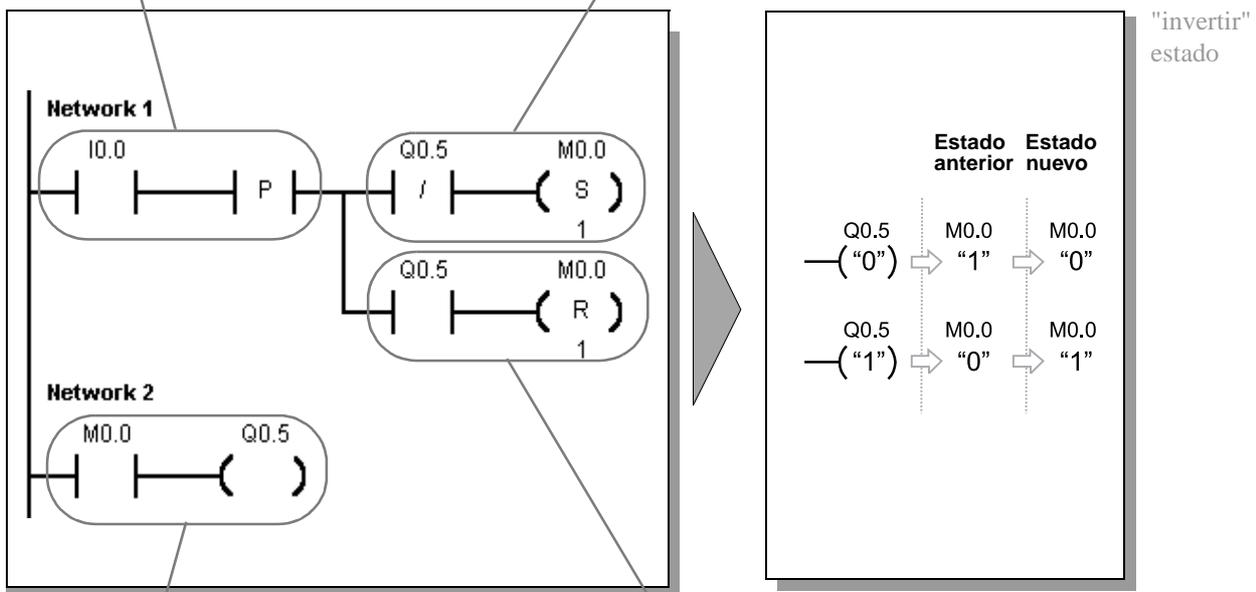
## Solución resumida



Antes de resolver paso a paso la tarea planteada le mostraremos primero la solución completa a modo de resumen.

Detectar si en I0.0 ha habido un cambio de estado de "0" a "1" (= flanco positivo).

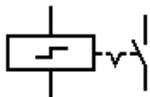
Cuando la salida Q0.5 está a "0" se pone a "1" la marca M0.0, esto "marca" que Q0.5 debe ser "1" en el segmento 2.



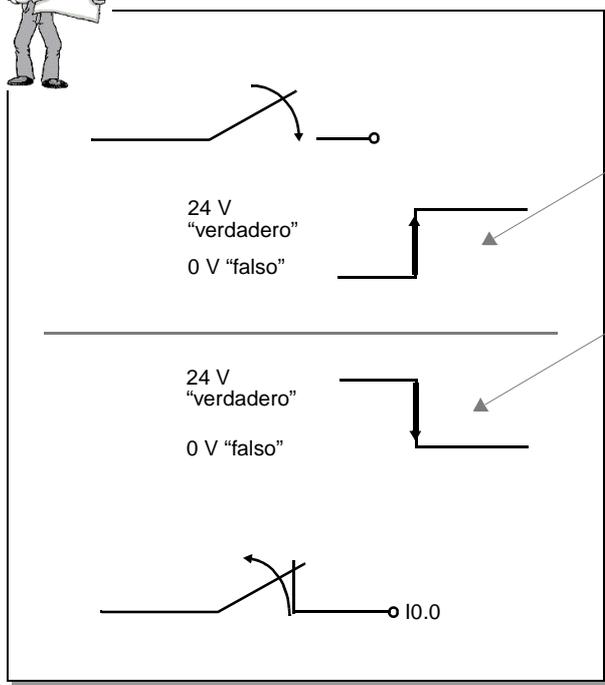
Asignar el estado de marca M0.0 a la salida Q0.5.

Cuando la salida Q0.5 está a "1" se pone a 0 la marca M0.0, con ello se "marca" que Q0.5 debe ser "0" en el segmento 2.

- Introducción
- Solución resumida
- **Detección de flancos**
- Marcas
- Descripción de la solución y test



## Detección de flancos (1)



La transición de un contacto (entrada, salida ...) de "abierto" a "cerrado" o de "falso" a "verdadero" se designa como flanco creciente o positivo.



Detectar flanco positivo

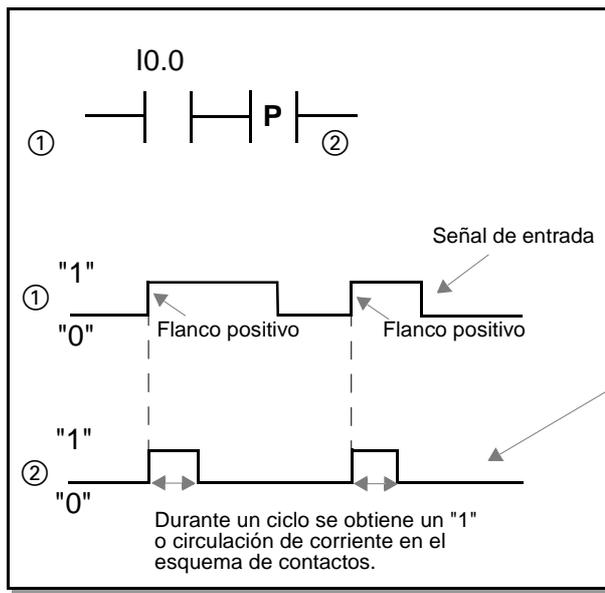
Lógicamente, la transición de "cerrado" a "abierto" o de "verdadero" a "falso" se designa como flanco decreciente o negativo.



Detectar flanco negativo

Para poder detectar estos estados, el S7-200 ofrece dos funciones de detección del flanco,  $\text{—|P|—}$  para flanco creciente y  $\text{—|N|—}$  para flanco negativo.

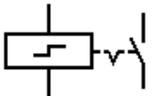
En este ejemplo utilizaremos la función  $\text{—|P|—}$  de la forma siguiente:



Esta es la señal que genera la función  $\text{—|P|—}$ .



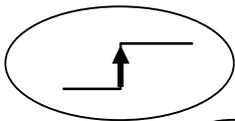
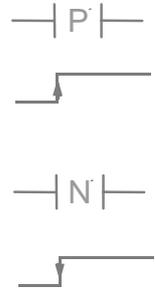
- Introducción
- Solución resumida
- **Detección de flancos**
- Marcas
- Descripción de la solución y test



## Detección de flancos (2)

El contacto  $\text{--}P\text{--}$  para detectar flancos positivos se cierra durante un ciclo cuando el contacto asociado pasa de "falso" a "verdadero".

Correspondientemente, el contacto  $\text{--}N\text{--}$  para detectar flancos negativos se mantiene cerrado durante un ciclo en las transiciones de "verdadero" a "falso".

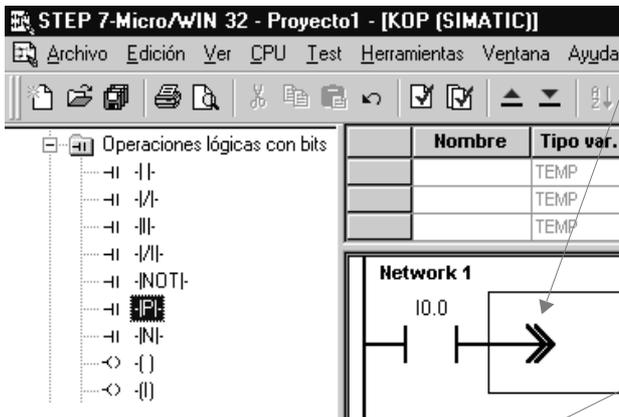


En nuestro ejercicio "Telerruptor" se utiliza  $\text{--}P\text{--}$  para transferir a las combinaciones siguientes una señal sólo en el momento de apretar el pulsador conectado en I0.0.

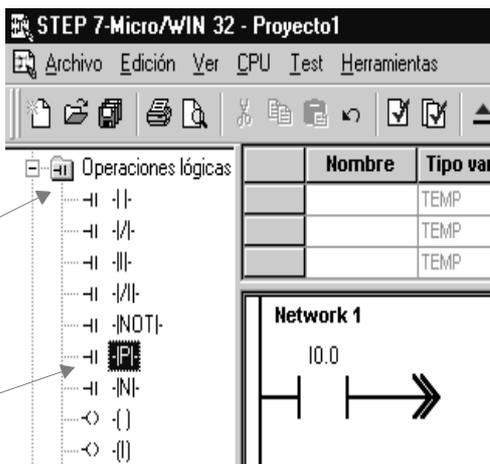
Así se introduce esta función:



Abra en STEP 7-Micro/WIN el proyecto de prácticas "a:\d03.prj" en el disquete. Este proyecto también está incompleto y será terminado paso a paso.



1. Con el ratón, haga clic en el punto cuyo elemento desea sustituirse por una detección de flanco.



2. Haciendo clic en el lista de familias de operaciones, seleccione "Operaciones lógicas".

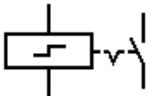
3. En la lista de operaciones que se abre a continuación seleccione "Flanco positivo" o "Flanco negativo".

⌘ Marcar

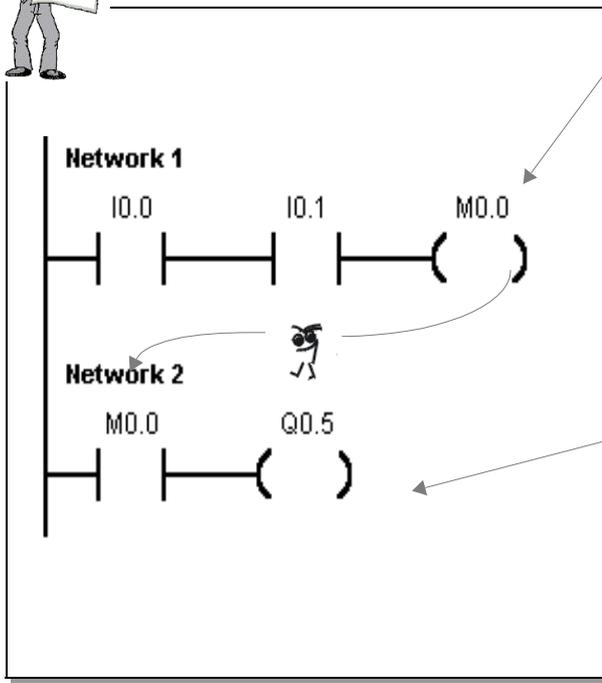
⌘ Marcar

⏏ Flanco

- Introducción
- Solución resumida
- Detección de flancos
- **Marcas**
- Descripción de la solución y test



## Marcas (1)



Para materializar el telerruptor se precisan marcas.  
La forma de trabajar con ellas se explica brevemente en un ejemplo.

La marca de bit "M0.0" se utiliza dentro del PLC para memorizar el resultado intermedio de la combinación "IO.0 Y IO.1" en lugar de la salida.

En este segmento, la marca de bit se utiliza como "contacto NA de entrada", controlando así la salida Q0.3. La marca puede utilizarse en todos los puntos que se desee dentro del programa.

Aplicable las veces que se desee como contacto NA o NC

Las marcas se utilizan, como la memoria de una calculadora de bolsillo, para guardar resultados intermedios.

En PLCs, las marcas se utilizan como salidas; su efecto es similar a los relés o contactores auxiliares utilizados en la técnica convencional. Una marca puede utilizarse todas las veces que se desee como contacto NA o NC.

Aplicación como salida

Efecto de contactor auxiliar

Si se corta la alimentación se pierde el estado de la marca.  
Para evitar esto existe la función de "remanencia".

El contenido de marcas está inmediatamente disponible (en el mismo ciclo) para las combinaciones siguientes.

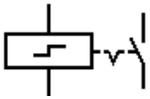
El contenido se actualiza inmediatamente

Las marcas se utilizan cuando el resultado intermedio de un segmento debe procesarse en otros segmentos (como sumas parciales al sumar a mano números) o para guardar estados sucesivos evaluados.

Modificables con -(S) o (R)  
Asignar sólo **una vez** con -( )-



- Introducción
- Solución resumida
- Detección de flancos
- **Marcas**
- Descripción de la solución y test



## Marcas (2)

Ahora ya conoce como funcionan las marcas, con lo que podrá entender la solución del circuito del telerruptor.

La función  $\neg P$  permite, cada vez que se aprieta el pulsador conectado a I0.0, la circulación de corriente durante un ciclo (detección de flanco) en el segmento

Por cada flanco,  $\neg P$  Q0.5 deberá invertir su estado.

El estado invertido (sucesivo) no escribe directamente en la salida Q0.5 ya que la salida recién puesta a 1 en la rama "superior" sería puesta a 0 inmediatamente en la rama "inferior". Por ello, el estado sucesivo lo escribimos en la marca M0.0 (= salvar de borrado).

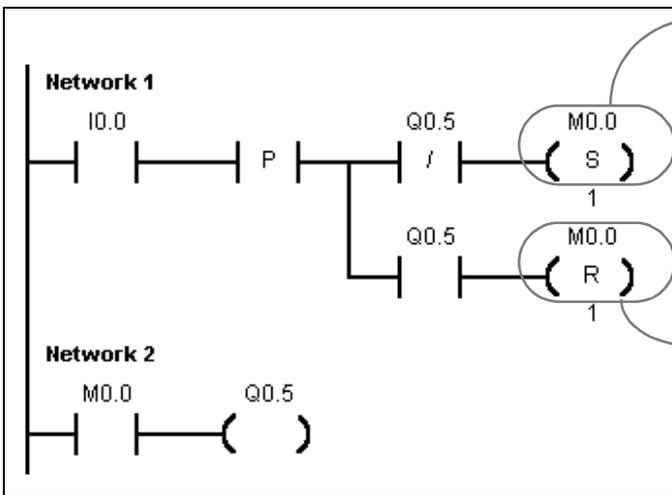
En el segmento 2 se asigna a la salida el estado "1" de la marca M0.0.



-(S)  
Poner a 1

-(R)  
Poner a 0

Salvar estado sucesivo en marcas para evitar borrado



En este punto es necesario posicionar una bobina para poner a 1 la marca M0.0. El número bajo la bobina indica cuantos elementos a partir de la dirección inicial indicada deben ponerse a 1. Aquí: poner un bit a 1 a partir de la marca M0.0.

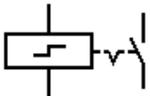
Como la rama inferior materializa la función inversa de la rama superior, es necesario poner a 0 el bit de marca M0.0, es decir "circule corriente" por esta rama.

M0.0 se pone a 1 si Q0.5 no estaba activada ("falso")

M0.0 se pone a 0 si Q0.5 estaba a 1 ("verdadero")

Complete el proyecto actual en STEP 7-Micro/WIN de la forma indicada anteriormente.

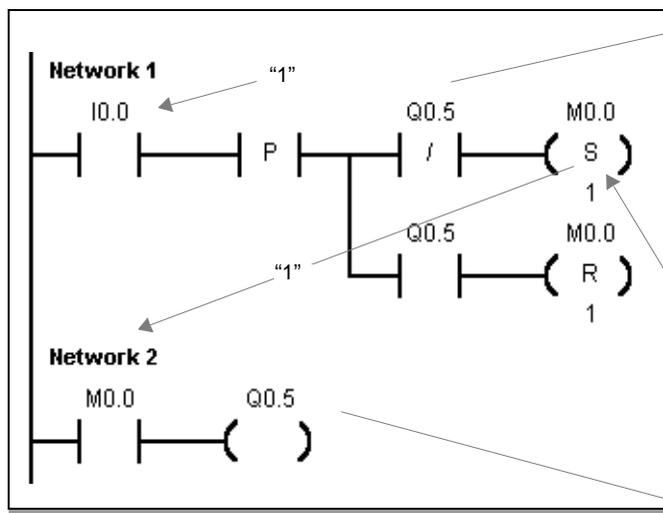
- Introducción
- Solución resumida
- Detección de flancos
- Marcas
- Descripción de la solución y test



## Descripción de la solución y test

Tomando como ejemplo la rama superior del segmento 1 (termina con (S), operación de conexión) explicaremos de nuevo la función en este programa ahora terminado:

La "circulación de corriente" en el esquema de contactos se muestra en el ciclo del flanco positivo en I0.0.



Al activarse I0.0 (detección de flanco -IPI-)

y si Q0.5 es "0" en el ciclo actual (la rama superior es verdadera, tras consulta con contacto normalmente cerrado)

**entonces...**

retener el estado sucesivo de Q0.5 poniendo a 1 la marca M0.0: **-(S)** Poner a 1 un bit a partir de M0.0.

M0.0 tiene ya aquí el estado sucesivo de Q0.5.

A Q0.5 sólo se asigna el nuevo estado al finalizar el ciclo, por lo que sólo aparece como "verdadero" o "1" en el siguiente ciclo de la representación KOP.



Guardar el programa terminado en el disco duro.

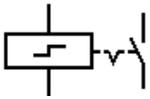


Transferir el programa al PLC.



Para probar, pasar el PLC al estado "RUN".

Pruebe su programa: active el pulsador en I0.0 y observe la salida Q0.5.



## Demuestre lo que sabe

... ya que entretanto ha aprendido bastante cosas!

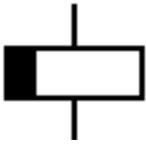
- ✓ Por ello, lea y conteste a las cuestiones siguientes.
- ✓ ¿Qué es el ciclo de un PLC?  
... qué tres partes importantes incluye un "ciclo"? ver página 9
- ✓ ¿Cómo se materializa con un PLC una autorretención? ver página 13
- ✓ Contacto normalmente cerrado: ¿Qué aspecto tiene un esquema de contactos (KOP), qué medida de seguridad permite realizar? ver página 14
- ✓ ¿Qué es un flanco, con qué se detecta y para qué se usa? ver página 23
- ✓ ¿Qué son marcas y para qué sirven? ver página 25
- ✓ ¿Cómo se introducen las bobinas "Poner a 1" y "Poner a 0" y qué efecto tienen? ver página 26



Seguro que ha podido contestar fácilmente a estas preguntas. Aunque para ello haya tenido que consultar las páginas indicadas.

¡Suponemos que lo ha entendido todo!

- Introducción
- Guardar como ...
- Insertar segmento
- Descripción de la solución
- Introducir comentarios

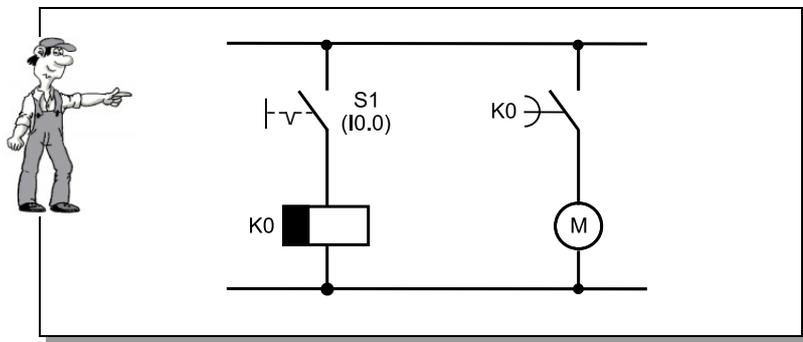


## Introducción

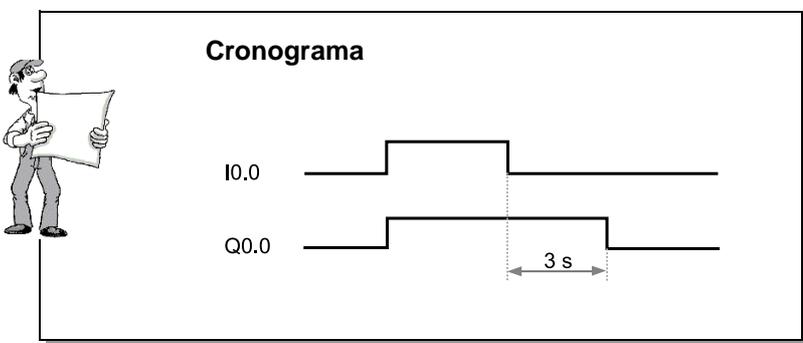


El retardo a la conexión ya lo tratamos en "El S7-200 en 1 hora". Ahora materializaremos un retardo a la desconexión.

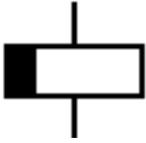
Al pulsar S1 (I0.0) se activará un ventilador cuyo motor está conectado a la salida Q0.0. Cuando se suelte S1 (I0.0), el ventilador deberá funcionar otros 3 segundos y luego pararse.



Cuando se suelte S1 el ventilador deberá funcionar aún 3 segundos



- **Introducción**
- Guardar como ...
- Insertar segmento
- Descripción de la solución
- Introducir comentarios



## Introducción



### Forma de proceder

- 1) Primeramente cargue en el disco duro la Autorretención, nuestro primer ejemplo.
- 2) Luego guarde este ejemplo bajo otro nombre en el disco duro.
- 3) Seguidamente crearemos sitio con "Insertar segmento".
- 4) Finalmente completaremos el retardo a la desconexión y lo dotaremos de comentarios.
- 5) Para acabar probaremos conjuntamente el programa.

En las próximas páginas seguiremos conjuntamente todos los pasos para materializar de forma segura el retardo a la desconexión.

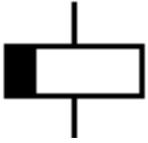


Le deseamos mucho éxito.

## Retardo a la desconexión

### Retardo a la desconexión

- Introducción
- Guardar como ...
- Insertar segmento
- Descripción de la solución
- Introducir comentarios



## Guardar como ...



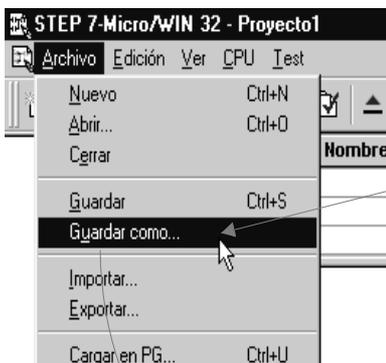
Para nuestro proyecto utilizaremos como base la Autorretención estudiada en el primer capítulo.

La forma más fácil de duplicar todo el proyecto es cargarlo y guardarlo seguidamente con otro nuevo nombre.



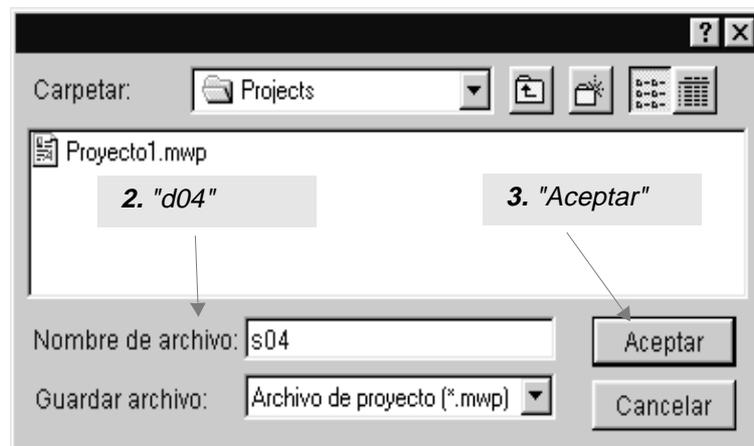
En STEP 7-Micro/WIN, cargar el proyecto "d01.prj" (Autorretención) del disco duro. Allí es donde se guardó en el primer capítulo.

Ahora queremos guardar el proyecto con otro nombre. Para ello proceder como se describe seguidamente, usar el nombre "d04.prj".



1. Seleccione el comando de menú "Archivo > Guardar como".

→ Menú: Archivo, Guardar como...



d04.mwp  
Aceptar

Repaso

Autorretención

Telerruptor

Retardo a la desconexión

Secuencia

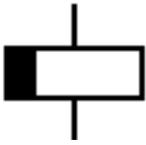
Anexo

31

## Retardo a la desconexión

### Retardo a la desconexión

- Introducción
- Guardar como ...
- Insertar segmento
- Descripción de la solución
- Introducir comentarios



## Insertar segmento

En la posición del segmento 2 se desea insertar un segmento más para poder materializar el retardo a la desconexión. Para ello se precisan los pasos siguientes:

1. Active, haciendo clic con el ratón, el campo de título de segmento 2.

2. Inserte un nuevo segmento en lugar del segmento 2.

Marcar

Botón de segmento en la barra de herramientas (F10)

Con ello se ha creado espacio para el nuevo segmento 2 que utilizará para materializar el retardo a la desconexión. El contenido del segmento 2 anterior se ha desplazado en un segmento.

#### Nota

Hay otra forma de crear espacio para introducir elementos KOP:

3. En el menú Edición, seleccionar el punto "Insertar".

Menú: Edición, Insertar...

STEP 7-Micro/WIN 32 - s04 - [KOP (SIMATIC)]

Archivo Edición Ver CPU Test Herramientas Ventana Ayuda

Deshacer Ctrl+Z

Cortar Ctrl+X

Copiar Ctrl+C

Pegar Ctrl+V

Seleccionar todo Ctrl+A

Insertar

Borrar

Buscar... Ctrl+F

Reemplazar... Ctrl+H

Ir a... Ctrl+G

Nombre	Tipo var.
	TEMP
	TEMP
	TEMP

Fila Ctrl+I

Fila abajo Shift+Ctrl+I

Columna

Vertical

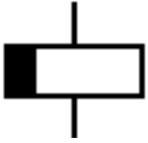
Segmento

Interrupción

Subrutina

4. Seleccionar "Segmento".

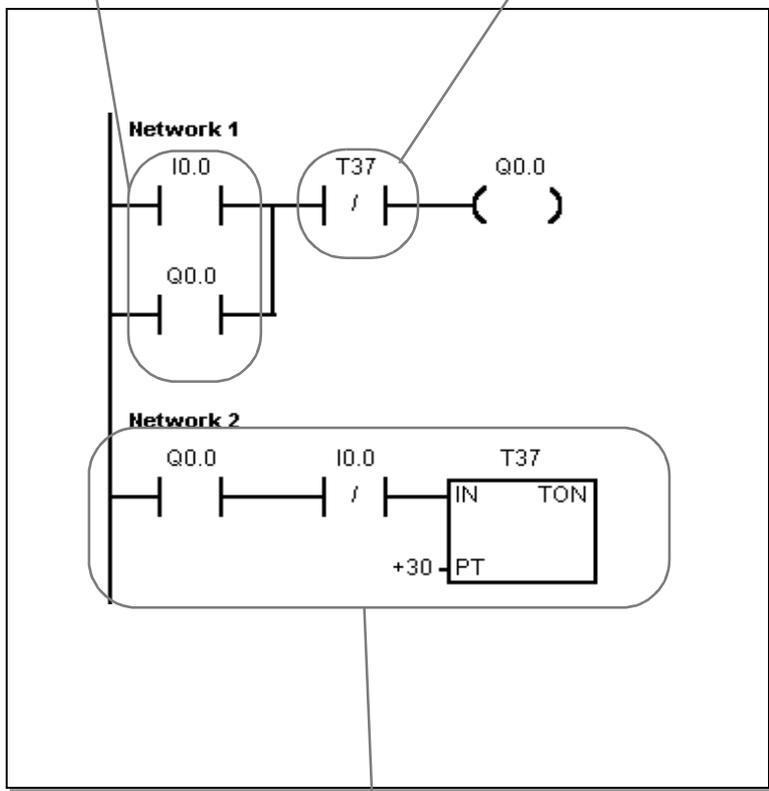
- Introducción
- Guardar como ...
- Insertar segmento
- Descripción de la solución
- Introducir comentarios



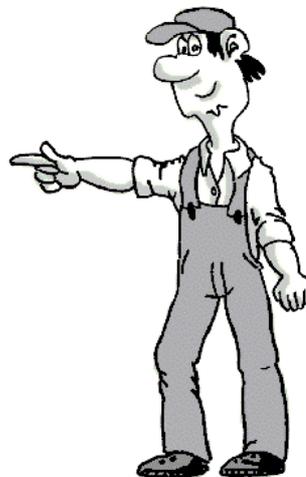
## Solución resumida

I0.0 activa a Q0.0  
Q0.0 se autorretiene ya que existe una conexión en paralelo a I0.0.

Cuando transcurre T37 se interrumpe la autorretención a través de este contacto. Con ello se para el motor. Si no ha transcurrido el temporizador T37 se mantiene la autorretención.

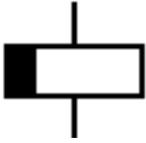


El programa terminado tiene el aspecto siguiente.



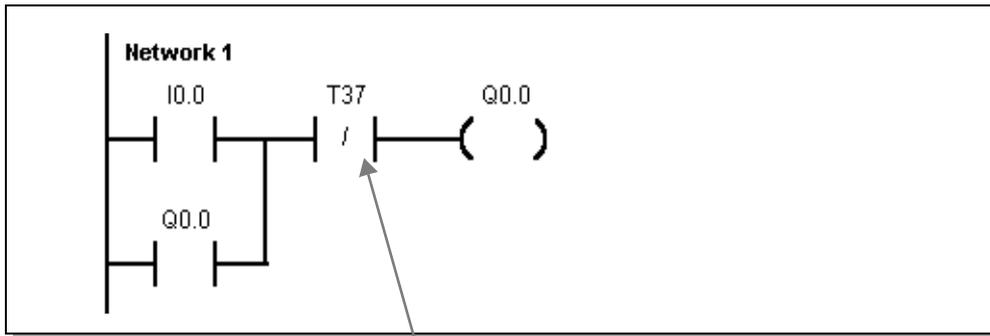
Si está accionada Q0.0 e I0.0 está de nuevo a "0" (S1 no está ya pulsado), entonces comienza a correr el temporizador T37.

- Introducción
- Guardar como ...
- Insertar segmento
- Descripción de la solución
- Introducir comentarios



## Solución - Introducir el programa

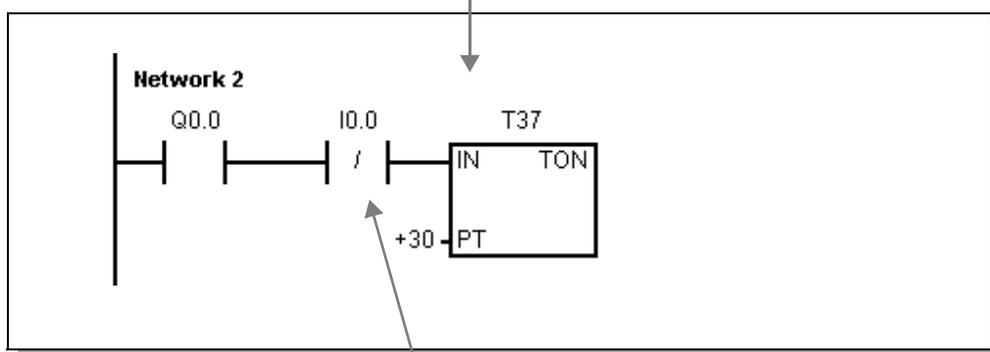
El segmento 1 debe tener el aspecto siguiente:



*En el ejemplo Autorretención, sobrescribir I0.1 con T37.*

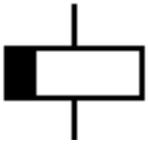
En el segmento 2, insertar el programa siguiente:

*T37 se introduce con:  
F2 Temporizadores/contadores y  
F3 Temporizador como retardo a la conexión.*



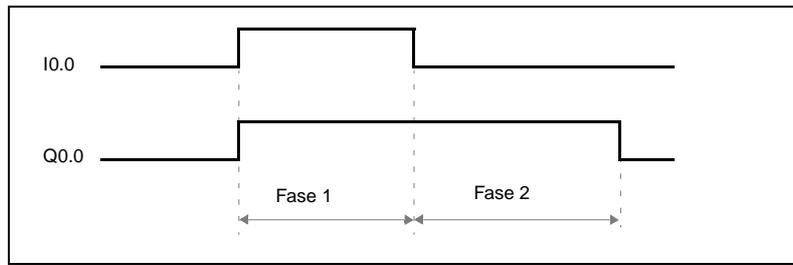
*T37 tiene una base de tiempo de 100 ms (vea también "El S7-200 en una hora", página 36).  
La temporización vale por ello  $30 * 100 \text{ ms} = 3 \text{ s}$ .*

- Introducción
- Guardar como ...
- Insertar segmento
- Descripción de la solución
- Introducir comentarios

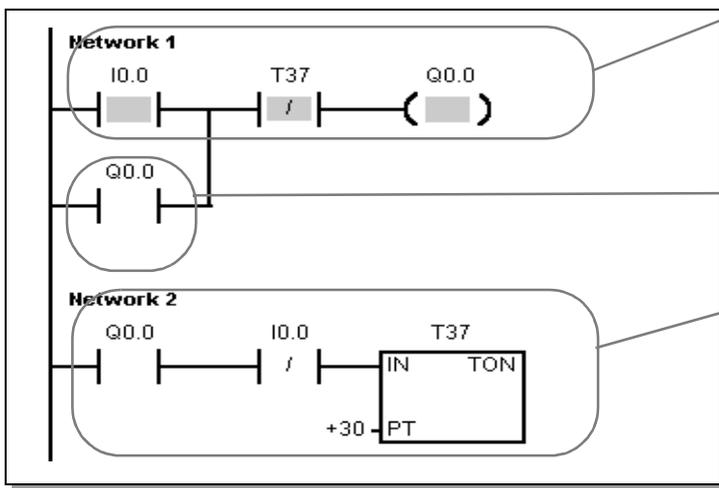


## Descripción de la solución

Así funciona nuestro programa. Tiene dos fases activas.

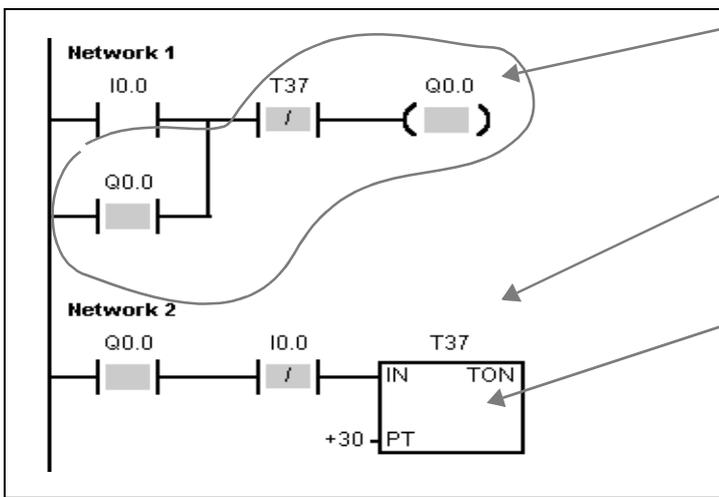


**Fase 1:** Activación de la autorretención, I0.0 está a "1" (suponemos que Q0.0 no está activada).



Si se acciona I0.0  
Y  
T37 aún no ha transcurrido  
ENTONCES  
se activa Q0.0 (= "1").  
A través de este contacto se produce la autorretención de Q0.0.  
T37 no corre aún ya que I0.0 todavía no está a "1".

**Fase 2:**



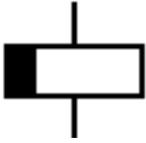
I0.0 no está ya activada.  
La autorretención se mantiene hasta que transcurra T37.  
Mientras corra la temporización, T37 está a "0" y circula corriente por el contacto NC.  
Si Q0.0 está activada Y I0.0 ya no está accionada, entonces corre la temporización T37.



## Retardo a la desconexión

### Retardo a la desconexión

- Introducción
- Guardar como ...
- Insertar segmento
- Descripción de la solución
- Introducir comentarios



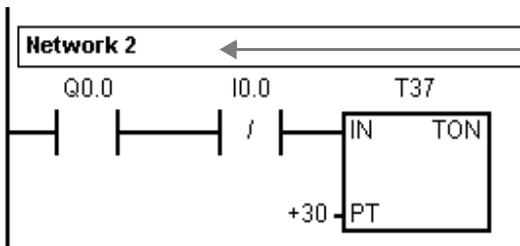
## Introducir comentarios (1)

¡Guarde y pruebe su nuevo programa! Si acciona I0.0 se activará Q0.0.  
Si desconecta I0.0, Q0.0 se apagará tras 3 segundos.



¡Bien hecho! Quizás desee añadir ya, para trabajos posteriores (cambios o similares), notas al programa para que sirvan de orientación.

También hemos pensado en ello. Para ello se ofrece la posibilidad de añadir título y comentario a cada segmento. Le muestro la forma de hacerlo.



1. Hacer clic en el campo de título del segmento 2.

2x

2. Se visualiza el editor de comentarios. Introduzca ahora el título o nombre de su segmento ...

Título

Título de segmento / Editor de comentarios

Network 2

Título 3s

Comentario  
T37 = 30 \* 100ms = 3s

Aceptar Cancelar

3. ... y aquí el comentario al segmento.

Comentario

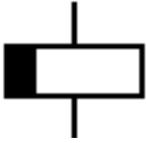
4. Pulsar Aceptar para terminar la introducción.

Aceptar

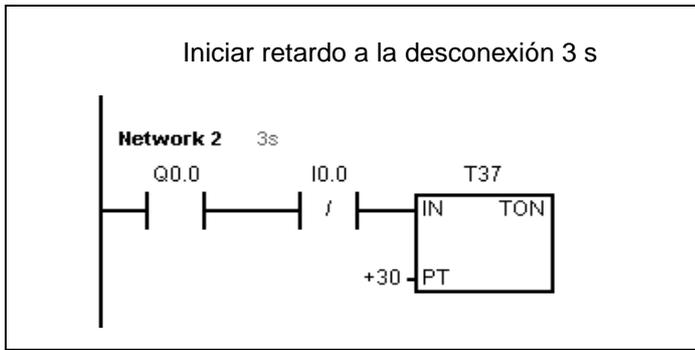
## Retardo a la desconexión

## Retardo a la desconexión

- Introducción
- Guardar como ...
- Insertar segmento
- Descripción de la solución
- Introducir comentarios

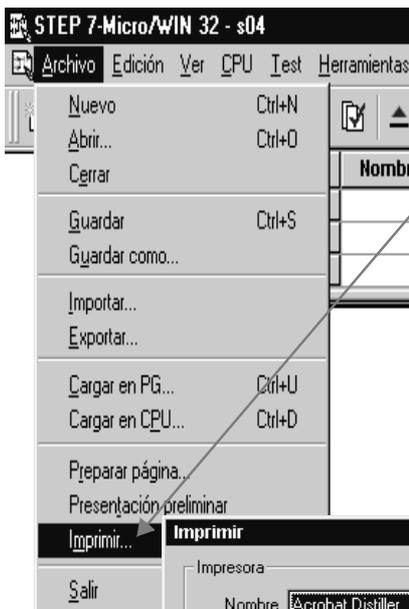


## Introducir comentarios (2)



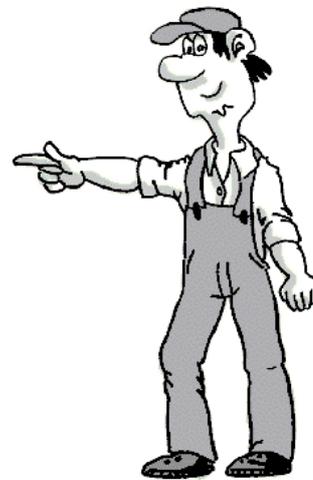
Tras añadir el comentario sólo se visualiza en pantalla el título del segmento.

Los comentarios pueden visualizarse posteriormente activando de nuevo el editor de comentarios.

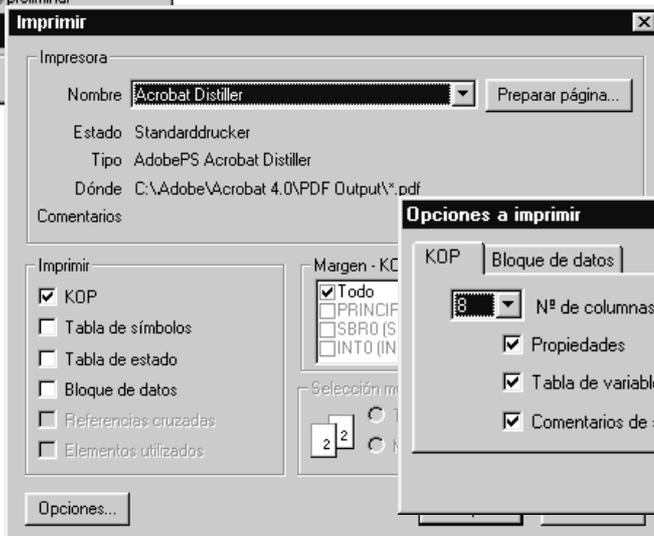


Si desea que los comentarios se impriman también con el programa, utilizar el comando de menú "Archivo/Imprimir".

→ Menú:  
Archivo,  
Imprimir



Imprimir  
comentarios de  
segmento



Acceptar

Repaso

Autorretención

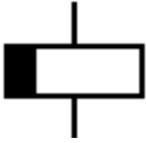
Telerruptor

Retardo a la  
desconexión

Secuencia

Anexo

37



## Demuestre lo que sabe

Lea las siguientes preguntas y contéstelas:

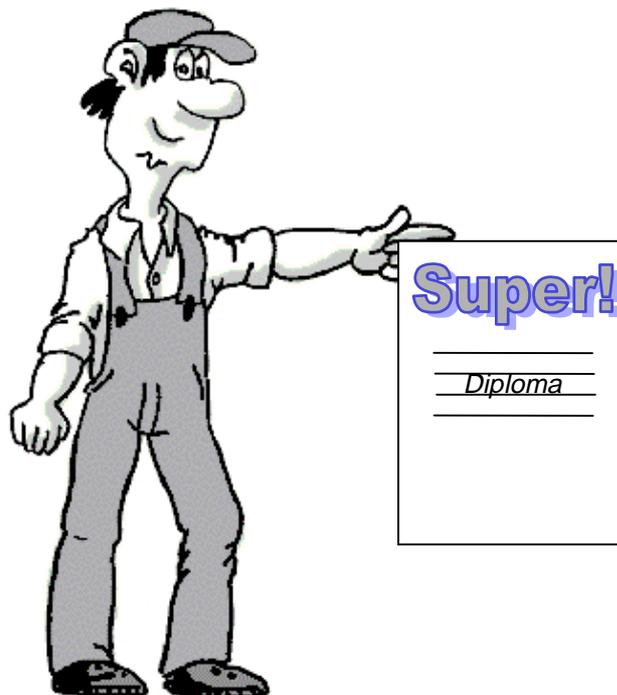
- ✓ ¿Cómo puede realizarse un retardo a la desconexión? Dibuje el esquema de contactos para dos soluciones posibles. Primero con una bobina normal —( )— y luego otra vez con **(S)** y **(R)**.
- ✓ ¿Cómo se guarda un proyecto?
- ✓ ¿Cómo se define el valor de una temporización?
- ✓ ¿Qué comentarios pueden hacerse a los segmentos?

Ver página 29

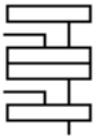
Ver página 31

Ver página 36 en "El S7-200 en una hora"

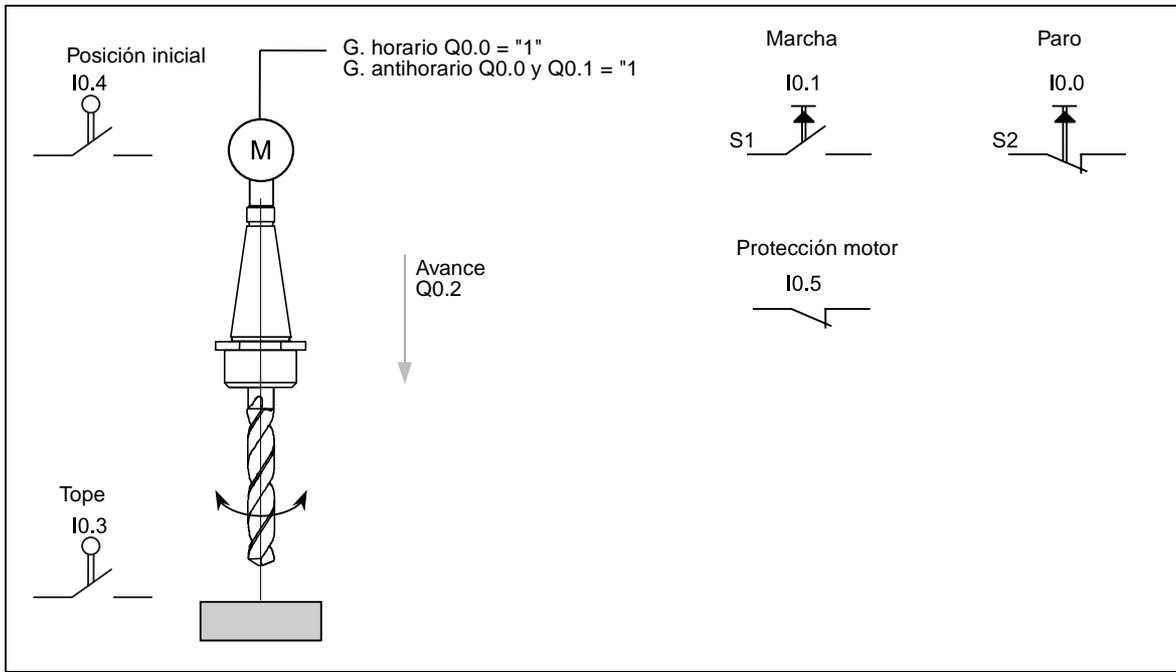
Ver página 36



- Introducción
- Fundamentos
- Uso de secuencias
- Modificación
- Descripción de la solución, ejemplo
- Test



## Introducción



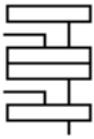
Ahora realizaremos conjuntamente una secuencia.

Con S1 se pone en marcha con giro horario el motor de una taladradora automática. Tras 3 s se conecta el avance.

Cuando se alcanza el tope en I0.3, se desconecta el avance. Un resorte lleva la máquina a la posición inicial. Para ello el accionamiento gira en sentido antihorario (Q0.0 y Q0.1 están a "1").

Una vez alcanzada la pos. inicial I0.4 = "1", el accionamiento sigue funcionando otro segundo hasta que se desconecta la máquina. Con paro es siempre posible desconectar la máquina (se activa con I0.0 = "0").

- Introducción
- Fundamentos
- Uso de secuencias
- Modificación
- Descripción de la solución, ejemplo
- Test

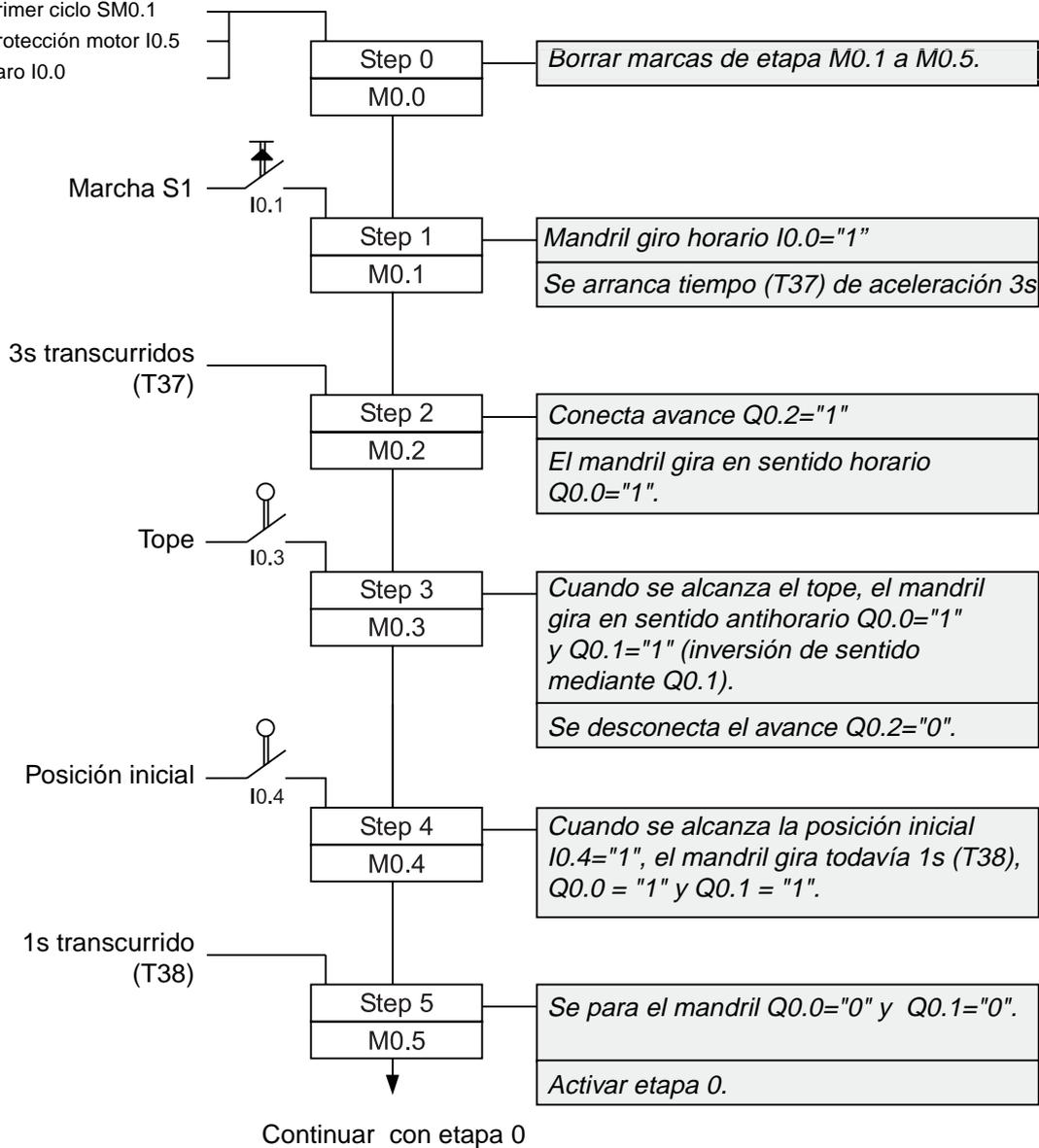


## Planteamiento

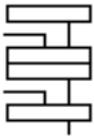


Este es el aspecto que tiene la solución para la secuencia del ejemplo de la taladradora automática.

Primer ciclo SM0.1  
Protección motor I0.5  
Paro I0.0



- Introducción
- Fundamentos
- Uso de secuencias
- Modificación
- Descripción de la solución, ejemplo
- Test



## Fundamentos (1)

Vamos a resolver el mando de la taladradora con una secuencia o cadena secuencial.

### ¿Qué es un mando o control secuencial?



- Se trata de un método de mando para el cual la tarea a resolver se divide en pequeñas etapas o pasos sucesivos (p. ej. motor con, avance con, avance des ...).
- Las tareas parciales (funciones) se denominan etapas.
- Normalmente es necesario ejecutar una etapa antes de poder comenzar la siguiente.
- Se activa una nueva etapa cuando se cumple la condición de transición.
- Una etapa está activa cuando su marca asociada, p. ej. M0.1 = "1".

Etapas

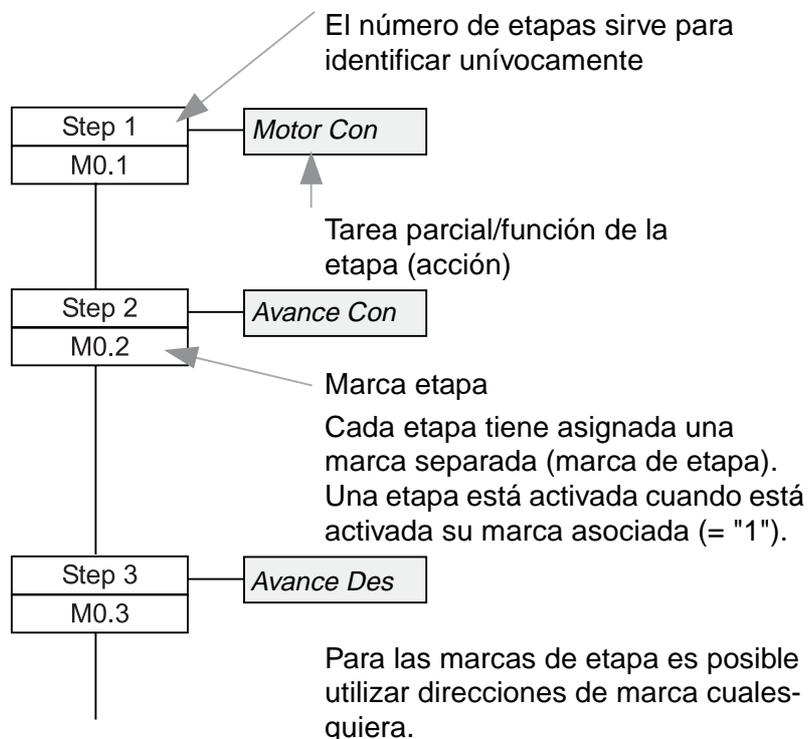
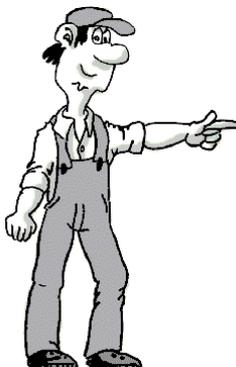
Condición de transición

Etapa activa

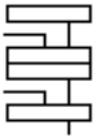


marca etapa  
MX.Y = "1"

Para cada estado importante se define una etapa o paso.



- Introducción
- Fundamentos
- Uso de secuencias
- Modificación
- Descripción de la solución, ejemplo
- Test



## Fundamentos (2)



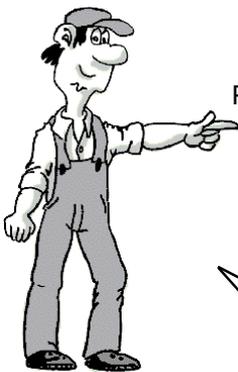
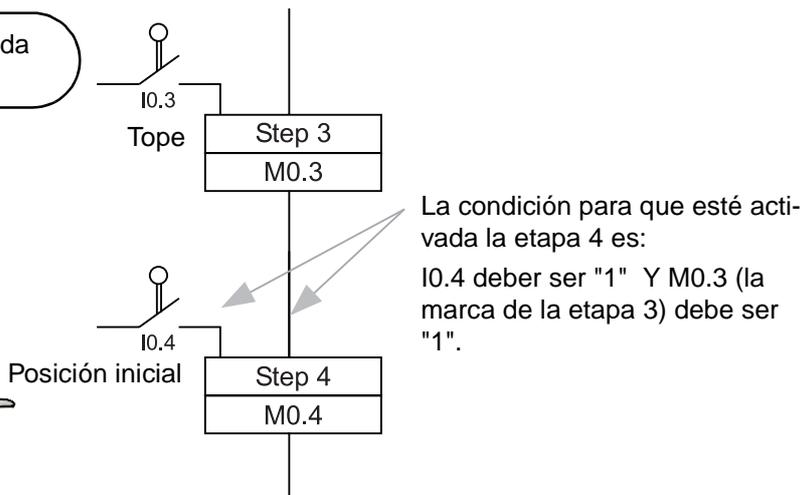
### ¿Qué es una condición de transición?

- Cada etapa se arranca (activa) al cumplirse una cierta condición. La condición resulta generalmente de diferentes estados de la máquina. Por ejemplo del estado de fines de carrera, pulsadores de mando, temperaturas alcanzadas u temporizaciones.
- Casi siempre, una etapa anterior activa forma también parte de la condición.
- Cuando se pone a 1 una nueva marca de etapa se pone a 0 la marca de la etapa precedente.

Condición de transición activa la marca de etapa

Marca de etapa activa "1"

Siempre está activada sólo una etapa.

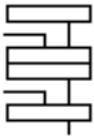


Si se cumple esta condición, p. ej. temporización transcurrida, fin de carrera accionado, se activa una nueva etapa. Normalmente se desactiva entonces otra etapa activa.

Al avanzar en la secuencia no nos preocupamos todavía de la activación de las salidas. Esto se realizará en otra sección del programa. Es decir, un mando materializado con secuencias consta de dos secciones:

- 1) El avance propiamente dicho entre las diferentes etapas siempre que se cumplan las condiciones necesarias (condición de transición).
- 2) La activación de las salidas (mandar válvulas y accionamiento).

- Introducción
- Fundamentos
- Uso de secuencias
- Modificación
- Descripción de la solución, ejemplo
- Test



## Fundamentos (3)

### Las dos secciones de un programa de mando secuencial:

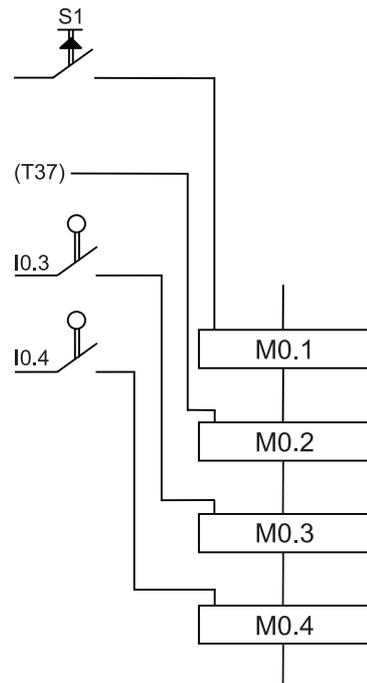
1) Las condiciones de activación de las diferentes etapas (tareas parciales) se combinan lógicamente con las diferentes marcas de etapa.

Cuando se activan sucesivamente las marcas M0.1... entonces se ejecuta la secuencia completa.

**Con ello se define la secuencia total de la tarea.**

Marcha S1 I0.1, retardo 3 s, tope I0.3, posición inicial I0.4, etapa anterior respectiva.

Marca de etapa M0.1, M0.2, M0.3, M0.4



1. Sec. programa

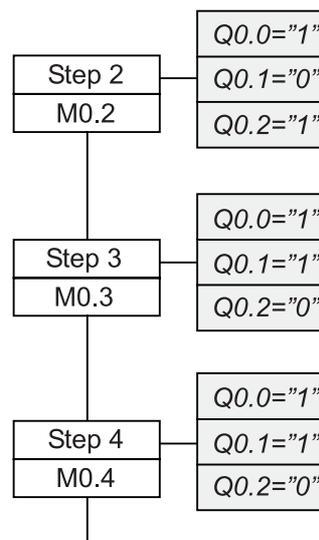
Marcha

Secuencia

2) Las marcas activas se asignan respectivamente a las salidas del PLC las cuales, por su parte, mandan p. ej. contactores o válvulas

**Esto constituye el interface a la instalación/máquina.**

Q0.1, Q0.2, Q0.0

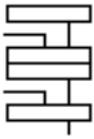


2. Sec.programa

Acciones

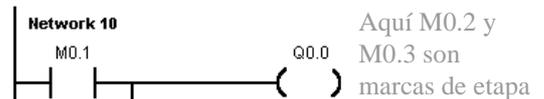
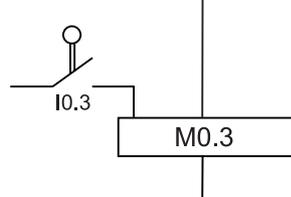
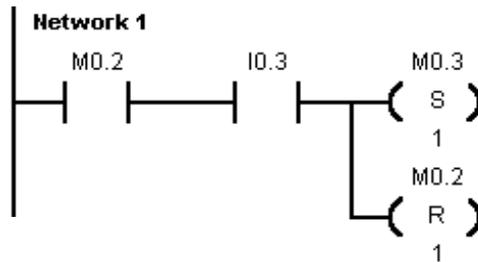
P. ej. motores  
válvulas

- Introducción
- Fundamentos
- Uso de secuencias
- Modificación
- Descripción de la solución, ejemplo
- Test

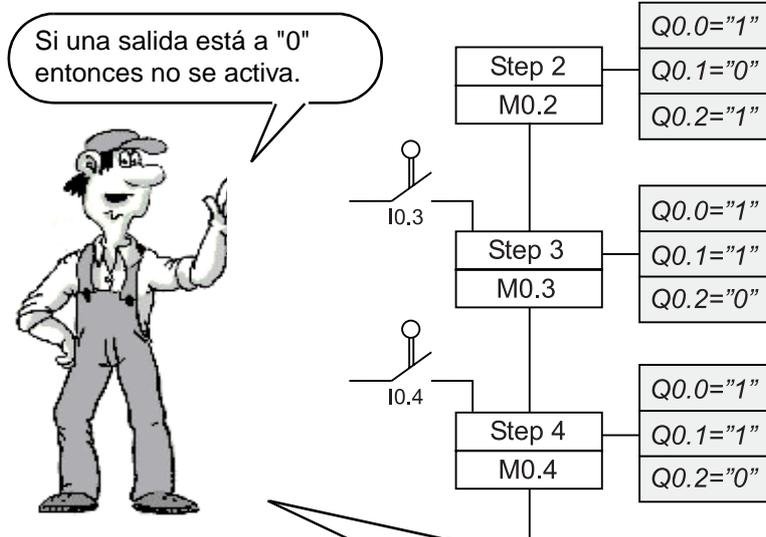


## Fundamentos (4)

### 1) Mando/avance de la secuencia



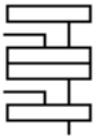
### 2) Mando de las salidas a través de marcas de etapa



Las salidas están mandadas únicamente por las marcas de etapa. ¡Asignando a las salidas bobinas — ( ) — normales queda garantizado el que la salida sólo se active en la etapa respectiva!

Si una salida debe ser "1" (p. ej. Q0.0), en varias etapas, entonces las marcas de etapa respectiva se combinan con una función "O" y se asignan a la salida.

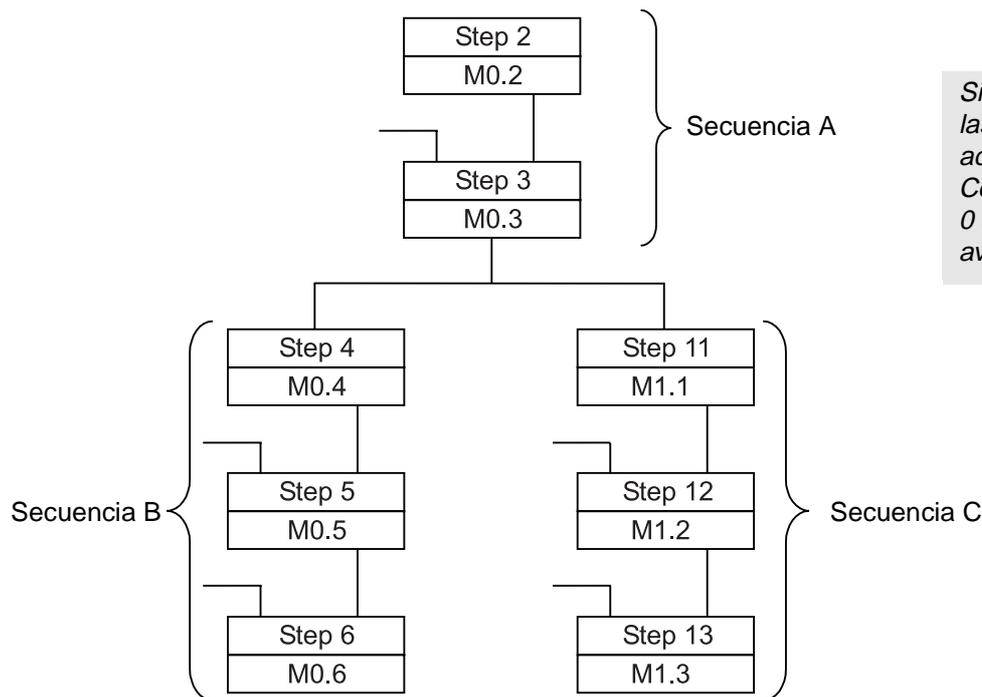
- Introducción
- Fundamentos
- **Uso de secuencias**
- Modificación
- Descripción de la solución, ejemplo
- Test



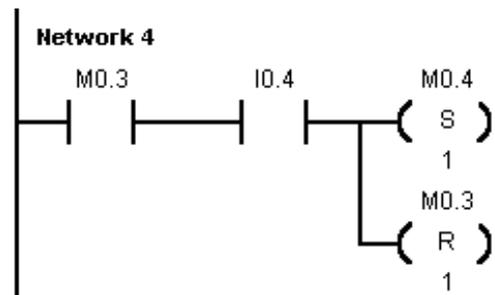
## Trabajar con secuencias (1)



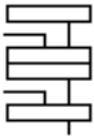
- Cada etapa tiene asignada una marca separada (marca de etapa). Esta está a "1" cuando está activa la etapa.
- Para mayor claridad, en la secuencia o cadena secuencial que nos ocupa nunca habrá más de una etapa simultáneamente activa. Es decir, sólo debe estar a "1" una marca de etapa.
- Si la tarea es más compleja es más favorable utilizar otra secuencia más.
- En el caso de que deban controlarse independientemente y de forma simultánea 2 o más operaciones, entonces se trabaja con secuencias separadas. Esto se muestra en el gráfico siguiente.



*Si M0.3="1" entonces arrancan las dos secuencias B y C. M0.3 activa las marcas M0.4 y M1.1. Como consecuencia se pone a 0 M0.3 y las secuencias B y C avanzan con independencia.*



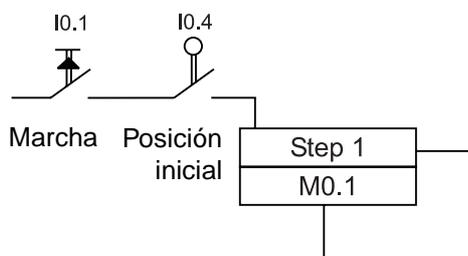
- Introducción
- Fundamentos
- **Uso de secuencias**
- Modificación
- Descripción de la solución, ejemplo
- Test



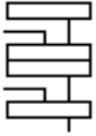
## Trabajar con secuencias (2)

En la práctica, una condición de transición puede estar compuesta de varios contactos.

En el caso de nuestro ejemplo, éste podría p. ej. ampliarse con la condición de que sólo sea posible poner en marcha cuando la taladradora esté en la posición inicial. En este punto, la secuencia tiene entonces el aspecto siguiente:



- Introducción
- Fundamentos
- **Uso de secuencias**
- Modificación
- Descripción de la solución, ejemplo
- Test

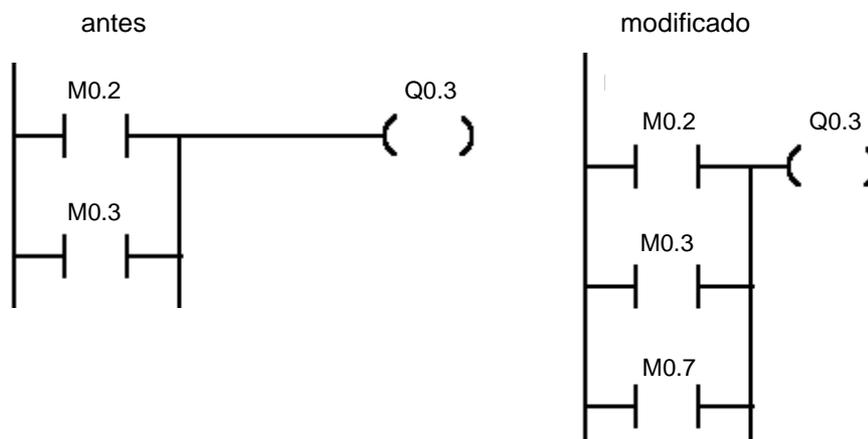


## Trabajar con secuencias (3)



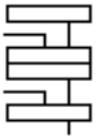
### Ventajas

- **La selección de mando en la secuencia y la activación de las salidas están separadas**
  - Así, si una salida no sólo debe estar activa en las etapas 1, 2 y 3 sino también en la 7, esto sólo exige modificar el programa en un punto.



- Los cambios en la sección de mando de la secuencia no aceptan a la activación de las salidas.
- **El programa puede probarse fácilmente**
  - Cada etapa puede seguirse en la unidad de programación.
  - Si no avanza la secuencia puede detectarse fácilmente qué condición falta.
- **Se reducen los tiempos de parada de la máquina**
  - Si no funciona la máquina, en base a la posición mecánica de la misma y de la marca de etapa activa es fácil detectar qué condición de transición falta.
- **Menores errores de programación, puesta en marcha más rápida**
  - El uso de una secuencia obliga a una estructura de programa que minimiza los errores de programación.

- Introducción
- Fundamentos
- **Uso de secuencias**
- Modificación
- Descripción de la solución, ejemplo
- Test

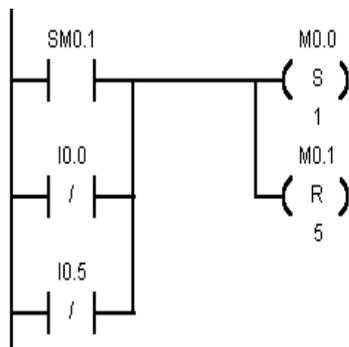


## Importante para trabajar con seguridad (1)



En la primera etapa (posición inicial) no conviene que se activen p. ej. accionamientos o válvulas. En nuestro ej. se trata de la etapa 0 o de la marca de etapa M0.0.

*Si se pulsa "PARO" o se dispara la protección del motor, basta con activar la primera marca de etapa (en nuestro ej., M0.0) para que se paren todos los accionamientos. Simultáneamente es necesario poner a cero las restantes marcas de etapa.*



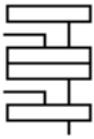
*Se activa M0.0, se ponen a cero M0.1 a M0.5*

- en el primer ciclo tras el restablecimiento de la alimentación mediante SM0.1 o
- si I0.0="0" o
- si I0.5="0".

En el primer ciclo tras la reconexión, SM0.1 está a "1" durante un ciclo

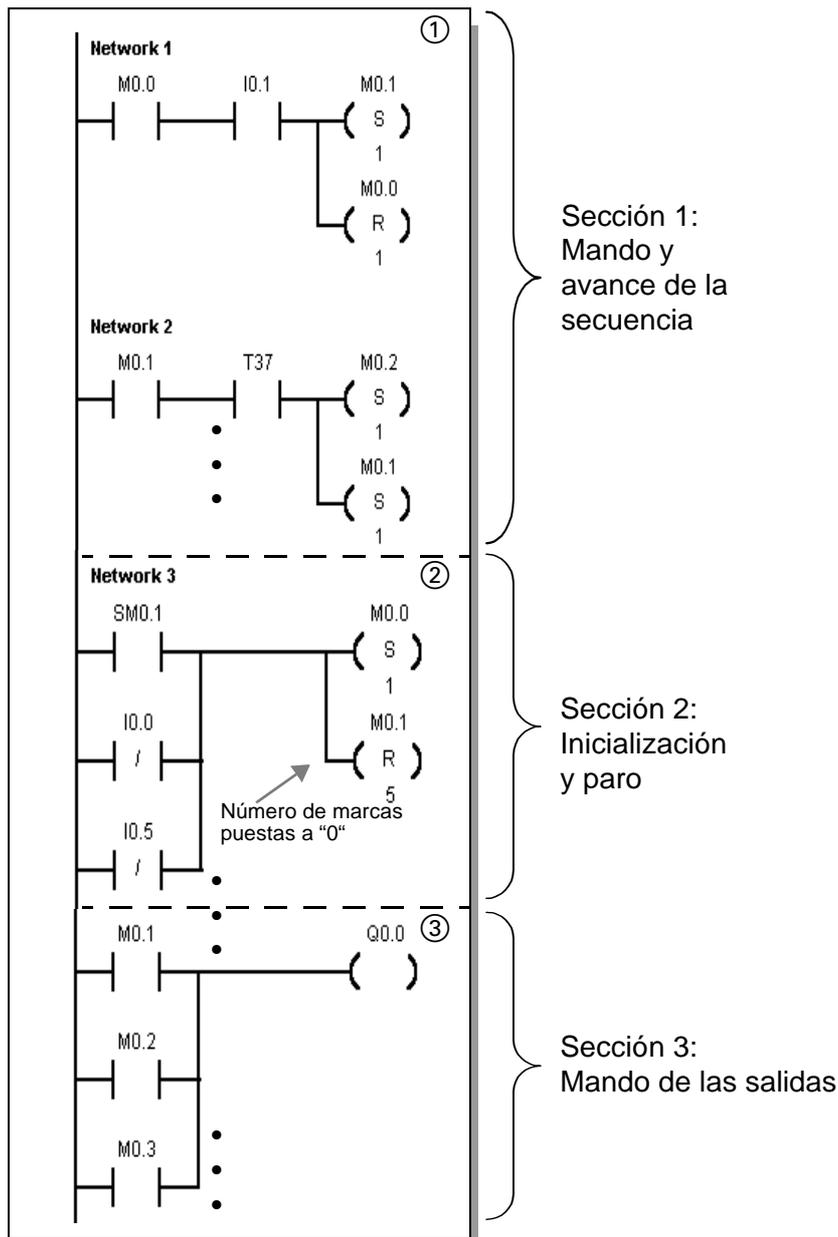
La parte del programa mostrada en el ejemplo deberá figurar al final de las condiciones de transición "normales" de la secuencia. Esto garantiza una desconexión eventualmente necesaria antes de que se activen las salidas.

- Introducción
- Fundamentos
- **Uso de secuencias**
- Modificación
- Descripción de la solución, ejemplo
- Test



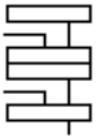
## Importante para trabajar con seguridad (2)

Un programa de secuencia debe tener el aspecto siguiente:



Antes de la asignación la primera salida ③ deberá estar dispuesta la sección para activar la posición inicial ②. De esta forma tiene la máxima prioridad la sección que trata de la activación de la posición inicial.

- Introducción
- Fundamentos
- Uso de secuencias
- **Modificación**
- Descripción de la solución, ejemplo
- Test



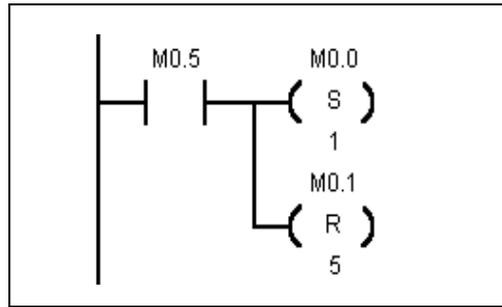
## Modificación



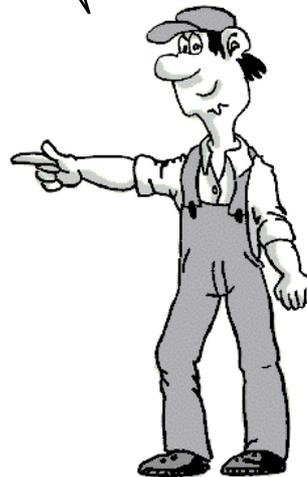
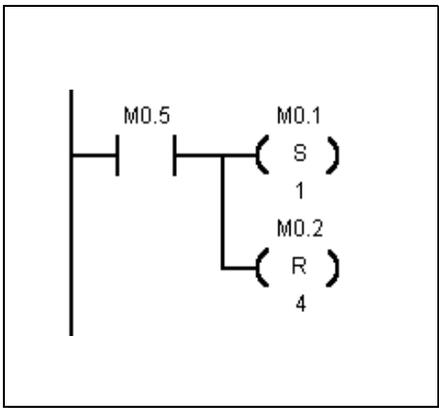
En el segmento 6 se define en qué etapa el programa salta a la etapa 5. En el ejemplo se trata de la etapa 0.

Esto es controlado por:

*Puesta a 1 de M0.0 y puesta a 0 de M0.1 a M0.5.*

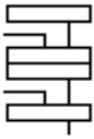


Si el programa debe saltar automáticamente tras la etapa 5 a la etapa 1, entonces el segmento 6 debe tener el aspecto siguiente.



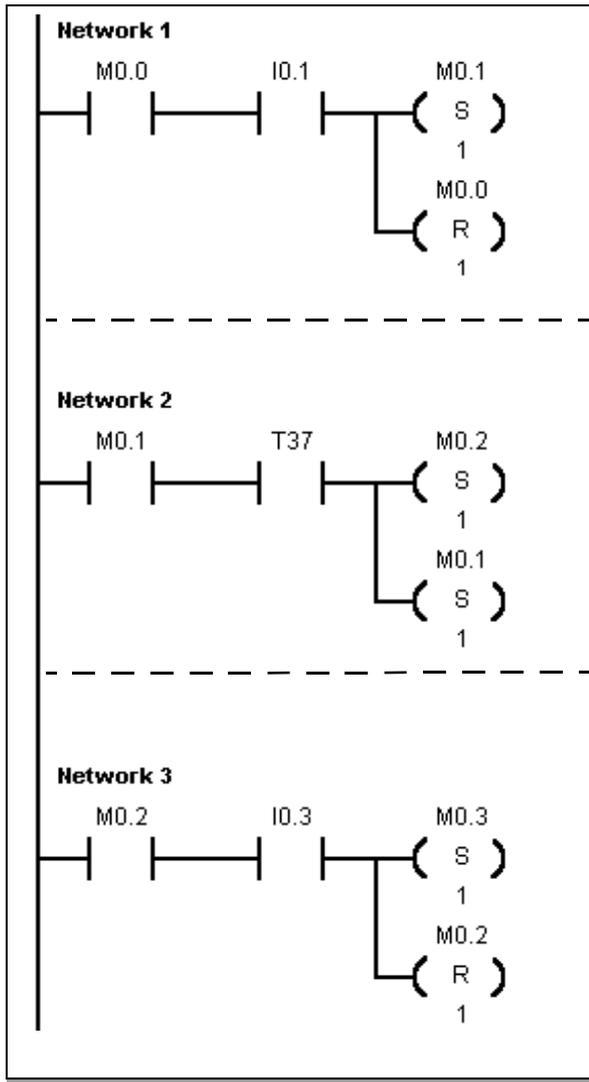
Con esta modificación, la "taladradora automática" funciona automáticamente hasta que sea detenida por I0.0 ó I0.5.

- Introducción
- Fundamentos
- Uso de secuencias
- Modificación
- Descripción de la solución, ejemplo
- Test



## Descripción de la solución, ejemplo (1)

### Sección 1 del programa - Avance de la secuencia



#### Activación de la etapa 1

Se pone a 1 la marca de etapa M0.1 cuando la secuencia está en la posición inicial (M0.0 = "1") Y I0.1 se activa. Simultáneamente se pone a 0 M0.0, la marca de etapa de la posición inicial.

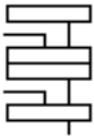
#### Activación de la etapa 2

La marca de etapa M0.2 se pone a 1 cuando la secuencia está en la etapa 1 (M0.1 = "1") Y transcurre la temporización T37. Simultáneamente se pone a 0 la marca de etapa M0.1.

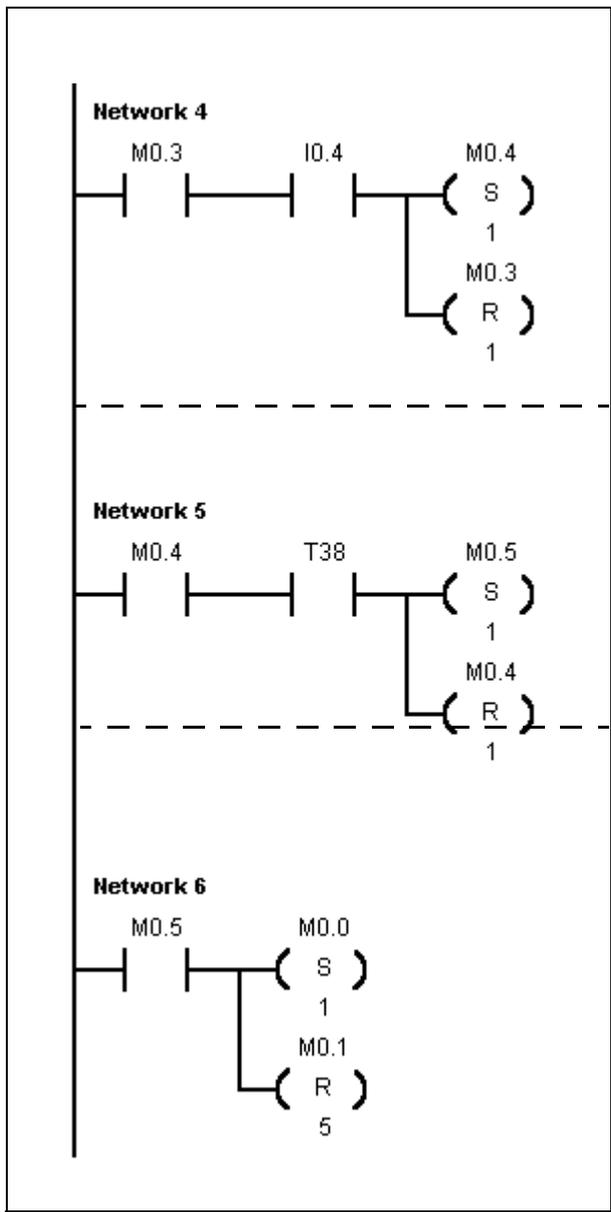
#### Activación de la etapa 3

La marca de etapa M0.3 se activa cuando la secuencia está en la etapa 2 (M0.2 = "1") Y la entrada I0.3 Tope está a "1". Simultáneamente se pone a 0 M0.2.

- Introducción
- Fundamentos
- Uso de secuencias
- Modificación
- Descripción de la solución, ejemplo
- Test



## Descripción de la solución, ejemplo (2)



### Activación de la etapa 4

Se pone a 1 la marca de etapa M0.4 cuando la secuencia está en la etapa 3 (M0.3 = "1") Y está a "1" la entrada I0.4 (posición inicial). Simultáneamente se pone a 0 M0.3.

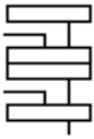
### Activación de la etapa 5

Se pone a 1 la marca de etapa M0.5 cuando la secuencia está en la etapa 4 (M0.4 = "1") Y ha transcurrido la temporización T38. Simultáneamente se pone a 0 la marca de etapa M0.4.

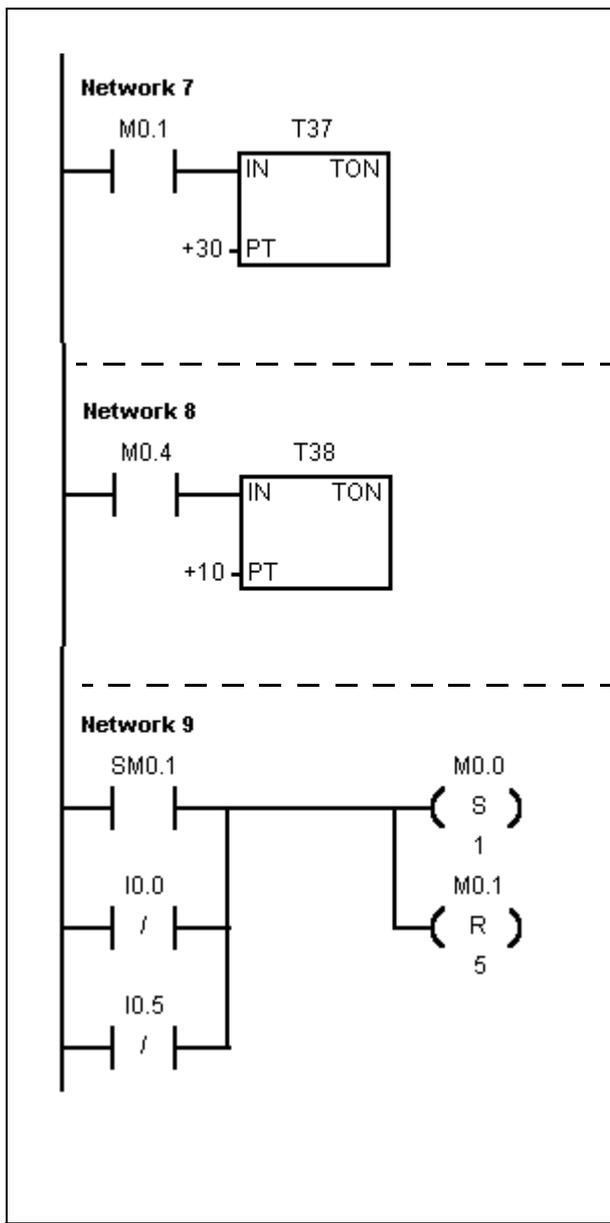
### Activación de la etapa 0

Si está activa la marca de etapa M0.5 (temporización T38 acabada) entonces la secuencia activa la etapa 0 (etapa de inicialización). Esta etapa en el segmento 6 se ha incorporado a propósito porque en este punto es posible prever la consulta de otras condiciones p. ej. la retirada de la pieza, antes de volver a activar la etapa 0. En tal caso, esta condición deberá conectarse en serie con el contacto M0.5.

- Introducción
- Fundamentos
- Uso de secuencias
- Modificación
- Descripción de la solución, ejemplo
- Test



## Descripción de la solución, ejemplo (3)



### Activación de la temporización T37

Cuando se active la etapa 1 (M0.1 = "1"), se arranca la temporización T37.

### Activación de la temporización T38

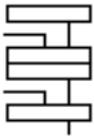
Cuando se active la etapa 4 (M0.4 = "1"), se arranca la temporización T38.

### Inicialización de la secuencia

La marca de etapa M0.0 se pone a "1"

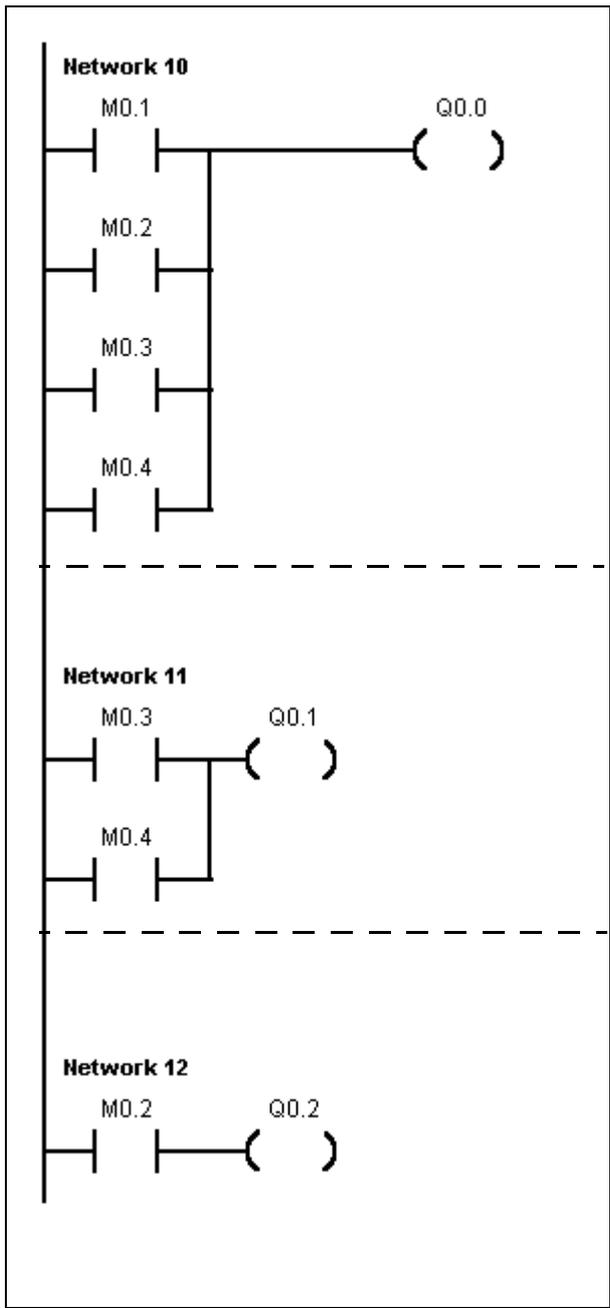
- 1) Durante el primer ciclo (SM0.1 vale aquí durante un ciclo "1")  
O
- 2) Si se pulsa PARO (I0.0 = "0")  
O
- 3) Si se ha disparado la protección del motor (I0.5 = "0"). Simultáneamente se ponen a 0 todas las marcas de etapa M0.1 a M0.5.

- Introducción
- Fundamentos
- Uso de secuencias
- Modificación
- Descripción de la solución, ejemplo
- Test



## Descripción de la solución, ejemplo (4)

### Sección 2 del programa - Mando de las salidas



#### Activar salida Q0.0

##### (giro horario)

La salida Q0.0 está a "1" en las etapas 1, 2, 3, 4, es decir cuando están a "1" M0.1 ó M0.2 ó M0.3 ó M0.4.

#### Activar salida Q0.1

##### (inversión sentido giro)

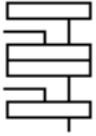
La salida Q0.1 está "1" en las etapas 3 y 4, es decir cuando M0.3 ó M0.4 = "1".

#### Activar salida Q0.2

##### (avance Con)

Si la marca M0.2 = "1", entonces la salida Q0.2 = "1".

- Introducción
- Fundamentos
- Uso de secuencias
- Modificación
- Descripción de la solución, ejemplo
- **Test**



## Test

Este programa puede introducirlo Ud. mismo o cargarlo del disquete; está en el archivo "**d05.prj**". Recuerde que el contacto del pulsador Paro I0.0 y de la Protección de motor I0.5 son "contactos normalmente cerrados". Esto es así por motivos de seguridad, ya que permite detectar la rotura del cable entre dichos contactos y el PLC, parando entonces la máquina.

Para test, I0.5 e I0.0 deben estar a "1", es decir deben lucir los LEDs de entrada.

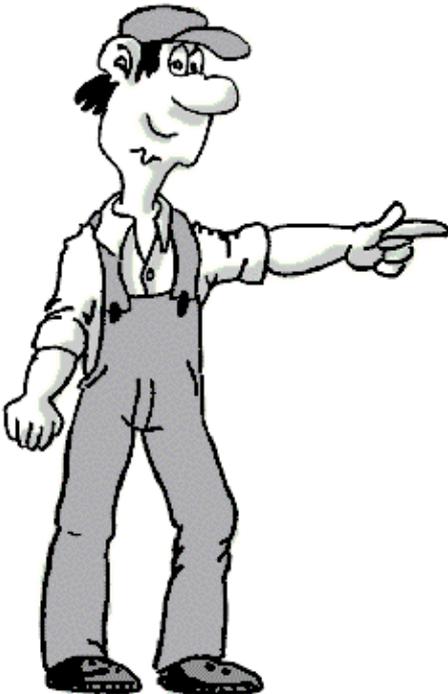
Una breve pulsación de I0.1 arranca el accionamiento. Tras 3 s se pone en marcha el avance Q0.2. Tras pulsar I0.3 el accionamiento invierte su sentido de giro y se detiene el avance Q0.2.

Cuando se alcanza la posición inicial (breve activación de I0.4), el accionamiento se detiene tras 1 segundo.

I0.0 e I0.5 permiten parar el accionamiento en cualquier fase.

Siga la evolución del programa en el modo Test. En cada caso podrá ver exactamente qué entrada es necesaria para que avance la secuencia.

¡Pruébelo Ud. mismo!

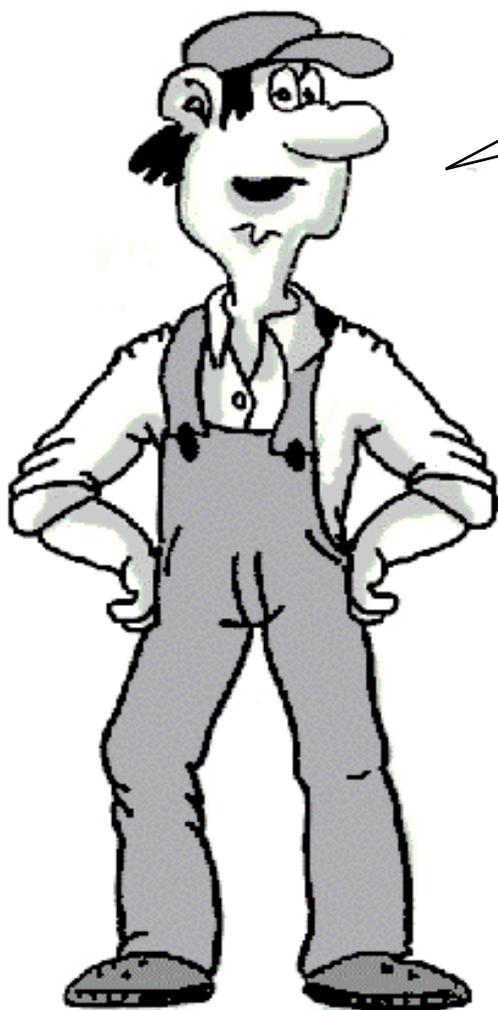


Secuencia



Notas





Trabajo realizado.

Ahora podrá resolver Vd. mismo tareas con el S7-200. Si quiere traducir circuitos de contactores más complicados, en el Anexo figuran algunos ejemplos prácticos.

Repaso



Autorretención



Telerruptor



Retardo a la desconexión



Secuencia

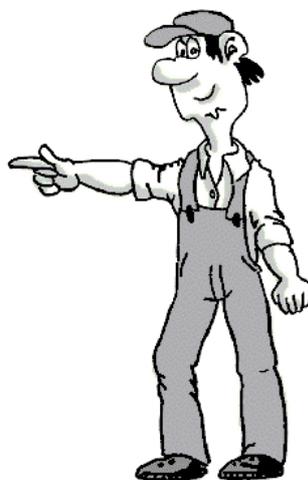


Anexo

## ¿Le ha sabido a poco?

Otros ejercicios figuran en el directorio "Samples" en el directorio STEP 7-Micro/WIN o en los "Tips & Tricks" (Ejemplos de aplicación) relativos al S7-200. El CD-ROM "Tips & Tricks" puede obtenerse a través de un interlocutor SIMATIC. Informaciones más detalladas pueden consultarse en los manuales del S7-200. Para ampliar conocimientos, asista a un curso sobre el S7-200 en el Centro de formación de Siemens o en su interlocutor SIMATIC.

Cuestiones pendientes o problemas técnicos: Los interlocutores SIMATIC le asisten con gusto.



Contacte con el interlocutor SIMATIC del que ha recibido el Kit de inicialización. Le ayudará con mucho gusto.

Si es posible contactar con su interlocutor, llame al Teléfono de atención al cliente SIMATIC: +49 911/895-7000.

Repaso

Autorretención

Telerruptor

Retardo a la desconexión

Secuencia

Anexo

59

Repaso



Autorretención



Telerruptor



Retardo a la  
desconexión



Secuencia



Anexo

Seguidamente hemos agrupado algunos ejemplos que le facilitarán traducir a un esquema de contactos circuitos con contactos y diodos complicados.



Repaso

Autorretención

Telerruptor

Retardo a la desconexión

Secuencia

Anexo

61

Repaso



Autorretención



Telerruptor



Retardo a la desconexión



Secuencia



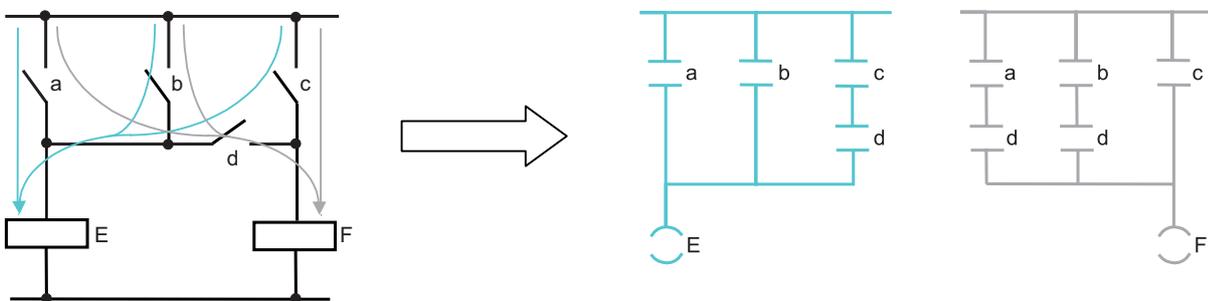
Anexo



## Circuito de puente

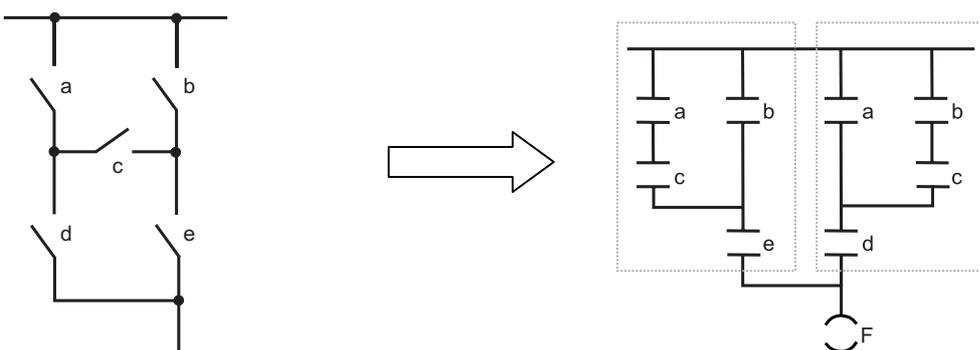
Si ha pasado de mandos por contactores a PLCs entonces conocerá circuitos con contactos que no es posible convertir directamente a un esquema de contactos KOP. Entre ellos cabe mencionar el circuito en puente. Seguidamente se muestra brevemente la forma de resolver un circuito en puente simple y otro complicado

### 1) Circuito en puente simple



El circuito en puente simple (izquierda) se materializa con dos segmentos. Para ello basta dividirlo en las diferentes ramas posibles. Para mejor comparación, el esquema de contactos se ha dibujado también vertical.

### 2) Circuito en puente complicado



Las dos ramas posibles se han transformado nuevamente y re combinado. Por un lado, a, c paralela a b, por otro lado b, c paralela a a. Para mejor comparación, el esquema de contactos se ha dibujado también vertical.

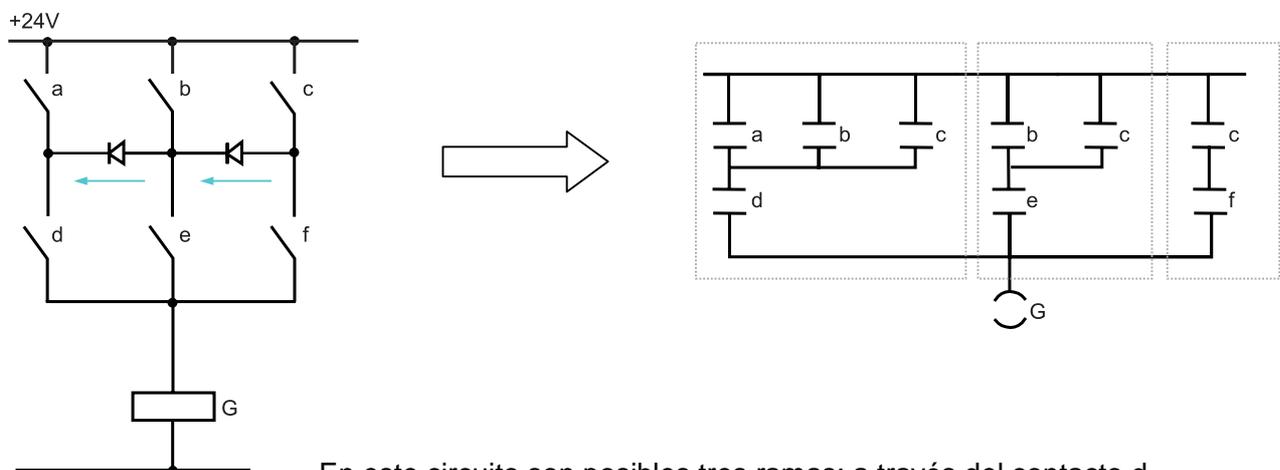
En nuevos proyectos, evite dibujar el esquema eléctrico de circuitos en puente, piense de partida de un "esquema de contactos KOP".



## Circuito con diodos

Si en los esquemas eléctricos "antiguos" se utilizaron diodos, su conversión a un esquema de contactos KOP no es posible sin más.

Como, por principio, los diodos representan conductores de unión, pero que sólo conducen en un sentido, esto se resuelve de forma similar al método seguido en el circuito en puente. Para poderlo comparar mejor con el esquema eléctrico, el esquema de contactos KOP se ha dibujado vertical.



En este circuito son posibles tres ramas: a través del contacto d, el contacto e así como el contacto f.

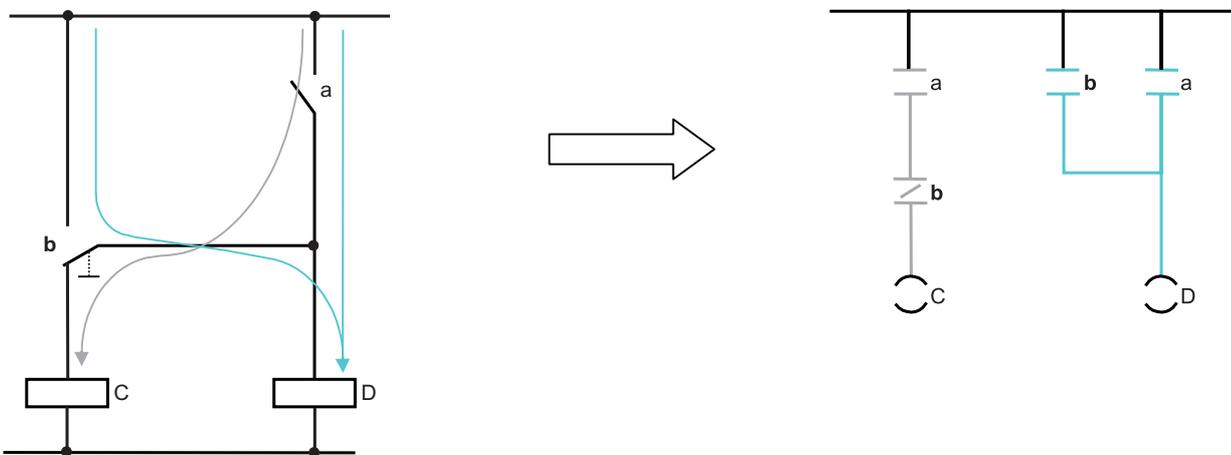
La corriente sólo puede circular por los diodos de b en sentido a d o de c en sentido a e.

Debido a las tres ramas resultan los tres segmentos parciales marcados en el esquema de contactos. Como los contactos d, e y f están en la misma barra que la salida G, estos tres subsegmentos se han unido también formando un segmento único.



## Conmutador

Los conmutadores tampoco tienen por qué causarle quebraderos de cabeza a la hora de transformar un esquema eléctrico en un esquema de contactos KOP. Seguidamente se describe brevemente esta transformación.



Para ver mejor la vía de circulación, ésta se ha resaltado gráficamente.

Para ello, el conmutador b se divide en un contacto normalmente cerrado que se considera en serie con a y que participa en la salida C y en un contacto normalmente abierto que tiene validez en paralelo a a y que conmuta D.

Es decir, por principio un conmutador puede convertirse en un esquema de contactos mediante un contacto normalmente cerrado y un contacto normalmente abierto que tienen la misma dirección de entrada.

Anexo

**Consejos**  
Notas.



# Notas

A4

Repaso



Autorretención



Telerruptor



Retardo a la  
desconexión



Secuencia



Anexo



## Índice A...H

Este índice incluye los conceptos más importantes para programar el S7-200. Encontrará breves explicaciones a las abreviaturas incluidas en esta obra así como remisiones a "El S7-200 en una hora".

Se utiliza el símbolo siguiente:

1h- Remisión a página de "El S7.200 en una hora"

- A**
- Autorretención: 13 y siguientes
  - AWL: Lista de instrucciones (inglés: STL)
  - Ayuda online: 8
- B**
- Bit de tiempo: 7
  - Binario: Representación de números en bits (dos valores posibles, 0 ó 1)
  - Bit: Cifra binaria: 6
  - Bloque de datos: Memoria de variables del S7-200; en él pueden almacenarse valores actuales para su uso en el programa de control
  - Bloque de organización: Incluye el programa de usuario del PLC que se ejecuta cíclicamente
  - Bobina: Representación en KOP de un elemento de salida (comparable a un contactor): 17
  - Byte: Valor de 8 bits de ancho: 1h- 48
- C**
- Ciclo de un PLC (3...10 ms): 9 y siguientes
  - Circuito con diodos: A2
  - Circuito en puente: A1
  - Circulación de corriente en KOP: 7
  - Condición de transición: 40
  - Contacto NA: 8
  - Contacto NC: 14, 15
  - Cortar elementos: 1h- 32
  - CPU: Central Processing Unit, unidad central, p. ej. del S7-200
- D**
- DB1: Bloque de datos del S7-200
  - DIV: Calcular división
- E**
- END: Instrucción de fin de programa: 31
  - Esquema de contactos: 1h- 25
  - Estado: Permite observar un proceso en el programa o en una tabla al efecto. Util para fines de test y diagnóstico.
  - Estado KOP: 7, 1h- 26
- F**
- Falso, verdadero: 6
  - Flancos: 21, 22
- G**
- Guardar el programa: 1h- 41
- H**
- HMI: Human-Machine-Interface (Manejo y visualización)



## Índice I...R

### I

I: Entrada, p. ej. I0.0  
 IB: Byte de entrada (8 bits), p. ej. IB0  
 Imagen del proceso: El programa del PLC trabaja con la imagen de la periferia. Al comenzar el ciclo se carga la imagen de las entradas; al final se emite a las salidas físicas la imagen de las salidas: 9 y siguientes  
 Insertar elementos: 1h-📖 30  
 IW: Palabra de entrada (16 bits), p. ej. IWO

### M

Mando combinacional: Mando que actúa en base a estados de entradas/salidas  
 Mando secuencial: Mando dividido en etapas sucesivas con condiciones de transición entre las mismas. En las etapas se producen las acciones programadas.  
 Mando semiautomático: Mando que ejecuta automáticamente ciertas acciones pero precisa en otros puntos la intervención del usuario.  
 Maqueta: 1h-📖 7  
 Marca: 25 y siguientes  
 Marca de etapa: 41  
 MB: Byte de marcas (8 bits)  
 MD: Palabra doble de marcas (32 bits)  
 Memoria V: Bloque de datos del S7-200  
 M+V: Manejo y visualización, p. ej. con visualizadores de textos, paneles de operador y paneles táctiles; hoy HMI  
 MW: Palabra de marcas (16 bits)

### O

OB1: Bloque de organización del S7-200

### P

PAA: Imagen de proceso de las salidas: 10  
 PAE: Imagen de proceso de las entradas: 9  
 Palabra: un valor representado por 2 bytes (16 bits)  
 PLC: Programable Logic Controller: 1h-📖 5  
 Poner a 0, poner a 1: 16 y siguientes

### Q

Q: Salida, p. ej. Q0.0  
 QB: Byte de salida (8 bits), p. ej. QB0  
 QW: Palabra de salida (16 bits), p. ej. QW0

### R

Remanencia: 23  
 Retardo a la conexión: 1h-📖 35  
 Retardo a la desconexión: 29 y siguientes  
 RET: Return, finalizar subprogramas  
 RUN: Posición del selector de modo del S7-200 para el (re) arranque manual del PLC



## Índice S...Z

- S**
- SBR: SubRoutine, subprograma
  - Secuencia: Cadena generalmente cerrada de etapas que se ejecuta secuencialmente: 39 y siguientes
  - Segmento, insertar: 32
  - Seguridad, aspecto de: 19
  - Selector de modo: Selector en el S7-200 con 3 posiciones: STOP, TERM, RUN
  - Símbolos utilizados: 3
  - SMB: Byte de marcas especiales (8 bits), p. ej. SMB28
  - SMB28: Potenciómetro del S7-200
  - SMD: Palabra doble de marcas especiales (32 bits)
  - SMW: Palabra de marcas especiales (16 bits)
  - Solución de autorretención: 15 y siguientes
  - Solución de retardo a la desconexión: 29 y siguientes
  - Solución de secuencia: 39 y siguientes
  - Solución de telerruptor: 21 y siguientes
  - STOP: Posición del selector de modo del S7-200 para detener manualmente el PLC
- T**
- T37 (Timer): 29 y siguientes
  - Telerruptor: 21 y siguientes
  - Temporizador: Tipo de datos para todas las funciones de tiempo (aquí también llamado "Timer")
  - TERM: Posición del selector de modo del S7-200. Permite intervenir en el PLC desde STEP 7-Micro/WIN
  - Timer: Temporizador en inglés.
  - TON: Temporizador del S7-200, también llamado Timer: 1h-📖 36 y siguientes
  - TONR: Retardo a la conexión con memoria
- V**
- V: Bit de variable, p. ej. V0.0
  - VB: Bytes de variables, p. ej. VB0
  - VD: Byte doble de variables, p. ej. VD45X
  - Verdadero, falso: 6
  - VW: Palabra de variables, p. ej. VW45
- X**
- XOR: O-Exclusiva, operación (combinación lógica); sólo conmuta cuando hay estados diferentes (antivalencia) en las entradas.
- Y**
- Y, operación lógica: 6
- Z**
- Z0: Contador simple (CTU)

Siemens AG  
A&D AS MVM  
Gleiwitzer Str. 555

Fax: +49-911 895-2786

90475 Nürnberg  
Alemania

## Resonancia de la publicación "El S7-200 en dos horas"

Estimado usuario del micro-PLC S7-200:

Hemos creado "El S7-200 en dos horas" para - basándose en "El S7-200 en una hora" - facilitarle la familiarización con el micro-PLC S7-200.

Estamos seguros que esta guía le permitirá resolver con facilidad tareas de control típicas. Si a pesar de todo tiene alguna sugerencia personal, su opinión nos es muy importante.

Utilice este formulario para informarnos. Indiquenos también su nombre y dirección para que podamos dirigirnos directamente a Vd. para consultas.

Muchas gracias

A&D AS MVM

---

### Sugerencias y propuestas de mejora

De

Nombre \_\_\_\_\_ Función \_\_\_\_\_

Empresa \_\_\_\_\_ Teléfono \_\_\_\_\_

Dirección \_\_\_\_\_ Lugar \_\_\_\_\_

Mis sugerencias:

---

---

---

---

---

---

---

---

Anexo



**Consejos**  
Notas.

