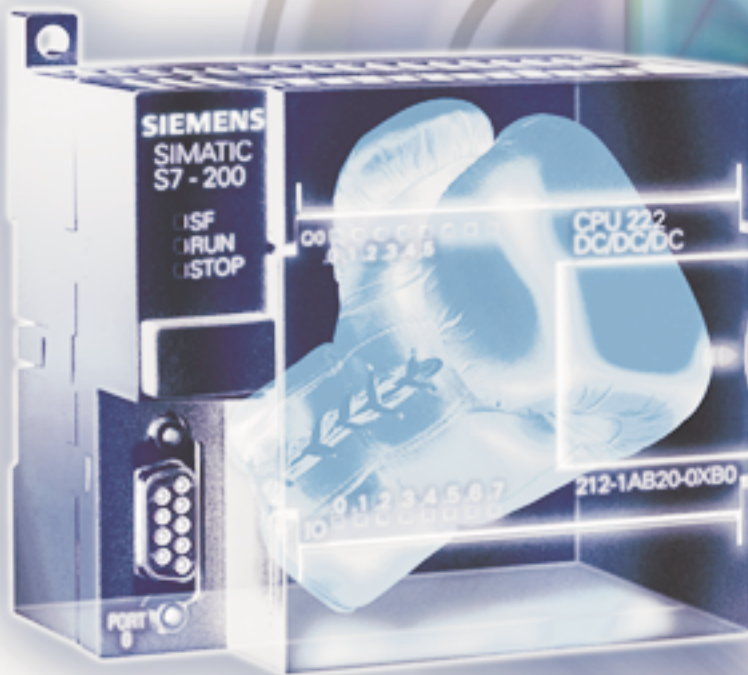


SIEMENS

Microsistema SIMATIC S7-200

S7-200 in 2 ore

Edizione 01/2000



microsystems

Avvertenze tecniche di sicurezza

Il presente opuscolo è stato redatto di proposito in forma sintetica per consentire un rapido approccio con il mondo dell'S7-200 e non sostituisce il manuale dell'S7-200.

Si prega pertanto di attenersi alle indicazioni contenute nel manuale dell'S7-200, in particolare per quanto riguarda le parti relative alla sicurezza.

Marchi

SIMATIC® e SIMATIC NET® sono marchi registrati della Siemens AG.

Gli altri nomi ricorrenti nella presente documentazione possono essere marchi il cui utilizzo improprio da parte di terzi può ledere i diritti del proprietario.

Copyright © Siemens AG 2000. All rights reserved

La duplicazione e la cessione della presente documentazione nonché l'uso improprio non sono consentite salvo autorizzazione scritta. Ogni trasgressione sarà punita. Tutti i diritti sono riservati, in particolare quelli relativi ai brevetti e ai modelli di utilità.

Siemens S.p.A.
Divisione Automation and Drives
Settore Sistemi di automazione per l'industria
Casella postale 4848, D-90327 Nuernberg

Esclusione della responsabilità

Abbiamo verificato che il contenuto della presente documentazione corrisponda all'hardware e software descritti. Non possiamo tuttavia garantire una corrispondenza totale. Il contenuto della documentazione viene tuttavia verificato regolarmente e le correzioni eventualmente necessarie vengono inserite nell'edizione successiva. Saremo lieti di ricevere proposte di miglioramento.

© Siemens AG 2000

Ci riserviamo di apportare modifiche tecniche

Sommario

Ripetizione	Due parole sulla ripetizione	5
	Ecco i bit	6
	Flusso di corrente nello schema a contatti	7
	Il ciclo del PLC	9
Autoritenuta	Introduzione	13
	Contatto normalmente chiuso	14
	Descrizione della soluzione e test	16
	L'autoritenuta in un'altra ottica...	17
Circuito a impulso	Introduzione	21
	La soluzione nel suo insieme	22
	Rilevamento del fronte	23
	Merker	25
	Descrizione della soluzione e test	27
Ritardo alla disinserzione	Introduzione	29
	Salva con nome...	31
	Inserimento segmento	32
	Descrizione della soluzione	33
	Introduzione dei commenti	36
Sequenza di passi	Introduzione	39
	Fondamenti	41
	Uso delle sequenze di passi	45
	Modifica	50
	Descrizione della soluzione, esempio	51
	Test	55
Appendice	Appendice	A1
	Indice	B1



Questa struttura dei capitoli è riportata anche in fondo a tutte le pagine. Il capitolo in cui ci si trova è sempre evidenziato.

Premessa

Egregio utente dell'S7-200,

l'efficienza applicativa dei moderni micro-controllori dipende soprattutto dalla rapidità e dal grado di sicurezza con cui l'utente è in grado di impiegare un PLC. Gli opuscoli "Un'ora per conoscerlo" e "Due ore per conoscerlo" sono stati realizzati proprio per consentire ai neo utenti di utilizzare un S7-200 in poco tempo e con grande facilità.

Questo nuovo opuscolo riprende il precedente "Un'ora per conoscerlo" per spiegare il funzionamento dell'S7-200. Inoltre esso illustra, sulla base di alcuni esempi pratici, come funziona il PLC e come si può impiegare quest'ultimo per risolvere efficacemente semplici compiti di automazione.

Dopo aver letto questo opuscolo sarete in grado di risolvere da soli delle tipiche situazioni di automazione.

Buon divertimento.

Dal dischetto allegato potrete caricare gli esempi citati nell'opuscolo.



Il team S7-200 vi augura
buon lavoro!

Ripetizione

Autorientata

Circuito a
impulso

Ritardo alla
disinserzione

Sequenza di
passi

Appendice

1

2

Ripetizione



Autoritenuta



Circuito a impulso



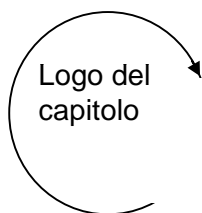
Ritardo alla disinserzione



Sequenza di passi



Appendice



Simboli utilizzati nell'opuscolo

In questo opuscolo vengono utilizzati vari simboli e altri elementi di evidenziazione il cui significato è spiegato qui di seguito.

Date subito un'occhiata all'intestazione della pagina! Ogni pagina presenta un'intestazione strutturata allo stesso modo. Il titolo nero in caratteri grandi indica la sezione attuale del capitolo. Il punto "Nuovo" in alto a destra elenca il contenuto delle pagine precedenti, in nero il contenuto della pagina attuale e quindi il contenuto della pagina o delle pagine successive.

Il testo su sfondo grigio invita ad eseguire un'azione, per es. immettere dati.



Questo simbolo indica che bisogna premere una volta il pulsante sinistro del mouse (clic) per eseguire un'azione (per es. selezionare un campo).



2 x

Questo simbolo indica che bisogna premere due volte in rapida successione il pulsante sinistro del mouse (doppio clic) per eseguire un'azione.



Premere il tasto di immissione della tastiera (detto anche INVIO o RETURN).



Indica che si possono selezionare con il mouse o con la tastiera (tasti funzione, tasti freccia) le voci elencate nella finestra di dialogo.

F2

Indica che si deve premere il tasto funzione "F2" (i tasti funzione vanno da F1 a F2). Nonostante la comodità del mouse, in certi casi si lavora più rapidamente utilizzando la tastiera.



Abbinato a un numero di pagina, questo simbolo segnala dove trovare maggiori informazioni su un determinato argomento.



Questo simbolo indica che dovete effettuare delle introduzioni nei campi di testo dell'interfaccia grafica. Oppure ricorda di annotare dei dati relativi al vostro progetto.



Menu

Selezionare una voce di menu sullo schermo con il pulsante sinistro del mouse (voce, sottovoce).

4

Ripetizione



Autoritenuta



Circuito a impulso



Ritardo alla disinserzione



Sequenza di passi



Appendice

- Due parole sulla ripetizione
- Ecco i bit
- Flusso di corrente nello schema a contatti
- Il ciclo del PLC



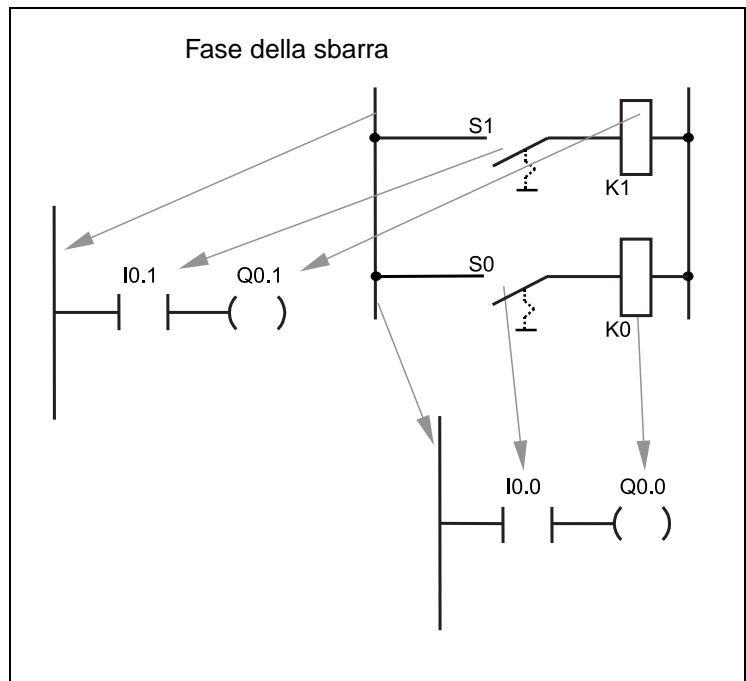
Due parole sulla ripetizione ...

Nell'opuscolo "Un'ora per conoscerlo" avete appreso che lo schema elettrico per i controllori basati su contattori presenta analogie con lo schema a contatti per programmare i PLC. Si tratta semplicemente di una rappresentazione che utilizza un'altra simbologia.

Inoltre avete imparato a programmare semplici combinazioni logiche e avete fatto conoscenza con i temporizzatori.



Cfr. "Un'ora per conoscerlo", pag. 24



- Due parole sulla ripetizione
- Ecco i bit
- Flusso di corrente nello schema a contatti
- Il ciclo del PLC

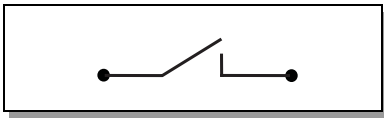


Ecco i bit

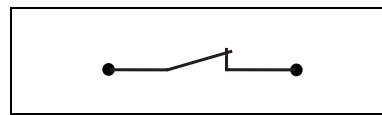
L'unità di elaborazione più piccola è il bit.
Il bit può assumere due stati:

- 1) "1" significa "bit impostato" oppure stato "vero",
- 2) "0" significa "bit non impostato" oppure stato "falso",

In un tipo di rappresentazione che ben conoscete i due stati binari "1" e "0" si possono rappresentare come circuiti elettrici, ovvero tramite interruttori.

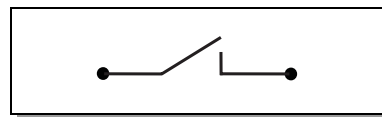


In questo caso un interruttore chiuso significa: passaggio di corrente e quindi stato del bit = "1"



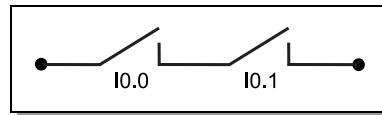
"1" = "vero"
passaggio di corrente

Mentre un interruttore aperto significa: nessun passaggio di corrente e quindi stato del bit = "0".



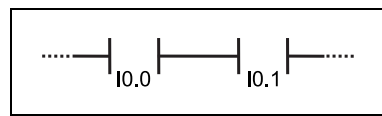
"0" = "falso"
nessun passaggio di corrente

Adesso basta poco per rappresentare sotto forma di circuiti anche le combinazioni logiche, come il collegamento in serie di due contatti. La combinazione logica in AND degli ingressi I0.0 e I0.1 viene rappresentata come illustrato qui accanto.



Combinazione logica in AND

Ciò corrisponde nella rappresentazione KOP a:



Per finire, un'ultima considerazione.

Per la logica positiva vale:

- 24 V = High = "1" e
- 0 V = Low = "0".

Logica positiva

Per la logica negativa vale:

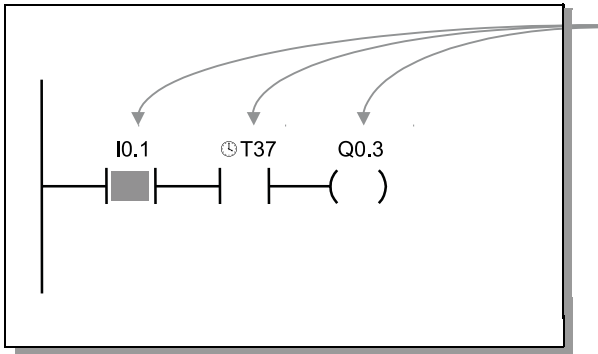
- 0 V = Low = "1"
- 24 V = High = "0".

Logica negativa

- Due parole sulla ripetizione
- Ecco i bit
- Flusso di corrente nello schema a contatti
- Il ciclo del PLC

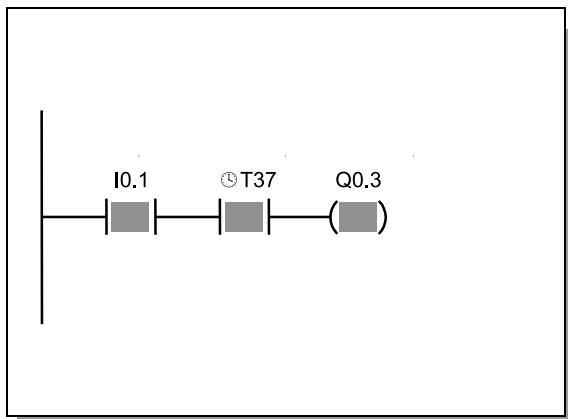


Flusso di corrente nello schema a contatti (1)



In questo esempio l'uscita Q0.3 viene inserita o messa a "1" quando il contatto in I0.1 viene chiuso ossia messo a "1" (DC 24 V all'ingresso I0.1) E, contemporaneamente, il bit di temporizzazione T37 viene attivato, ossia diventa "1".

L'ingresso I0.1 vale ora "1", per cui il contatto I0.1 è chiuso. Nella figura, T37 non è attivo, ossia è "0". Per questo Q0.3 resta disinserita, ovvero "0".



Quando anche il temporizzatore T37 è "1" (T37 è trascorso), il risultato della combinazione logica in AND diventa "1" e quindi anche l'uscita Q0.3 assume il valore "1".

Anche il bit di uscita diventa allora "vero", e ha il valore "1" (rappresentato con sfondo grigio).

Ciò corrisponde all'indicazione di stato KOP che avete già utilizzato in "Un'ora per conoscerlo" per testare il vostro programma.

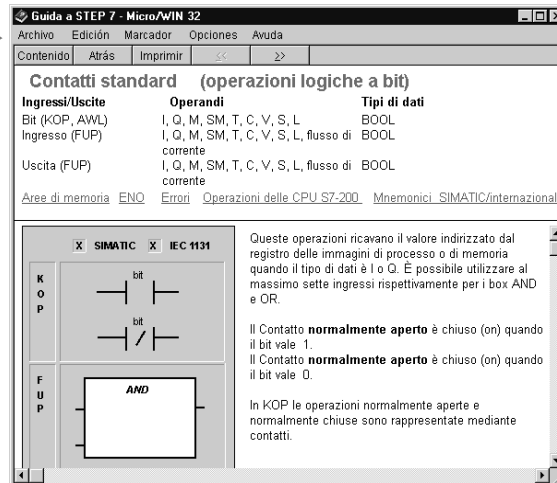
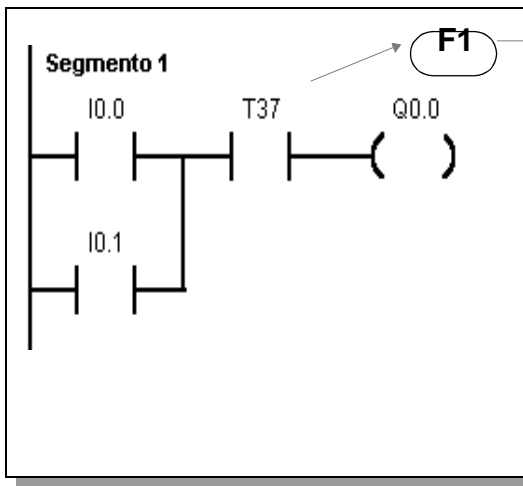
Ripetizione

Questo lo sapete già...

- Due parole sulla ripetizione
- Ecco i bit
- Flusso di corrente nello schema a contatti
- Il ciclo del PLC



Flusso di corrente nello schema a contatti (2) (uso della Guida)



Visualizzare la Guida

1. Selezionare l'elemento

2. F1

Per visualizzare nuovamente la Guida online relativa a un simbolo di contatto o ad altre funzioni:

selezionate il contatto

- nel schema a contatti (KOP) o
- nel schema logico (FUP) risp.
- di STEP 7-Micro/WIN facendovi clic sopra

con il mouse e premete quindi il tasto F1.

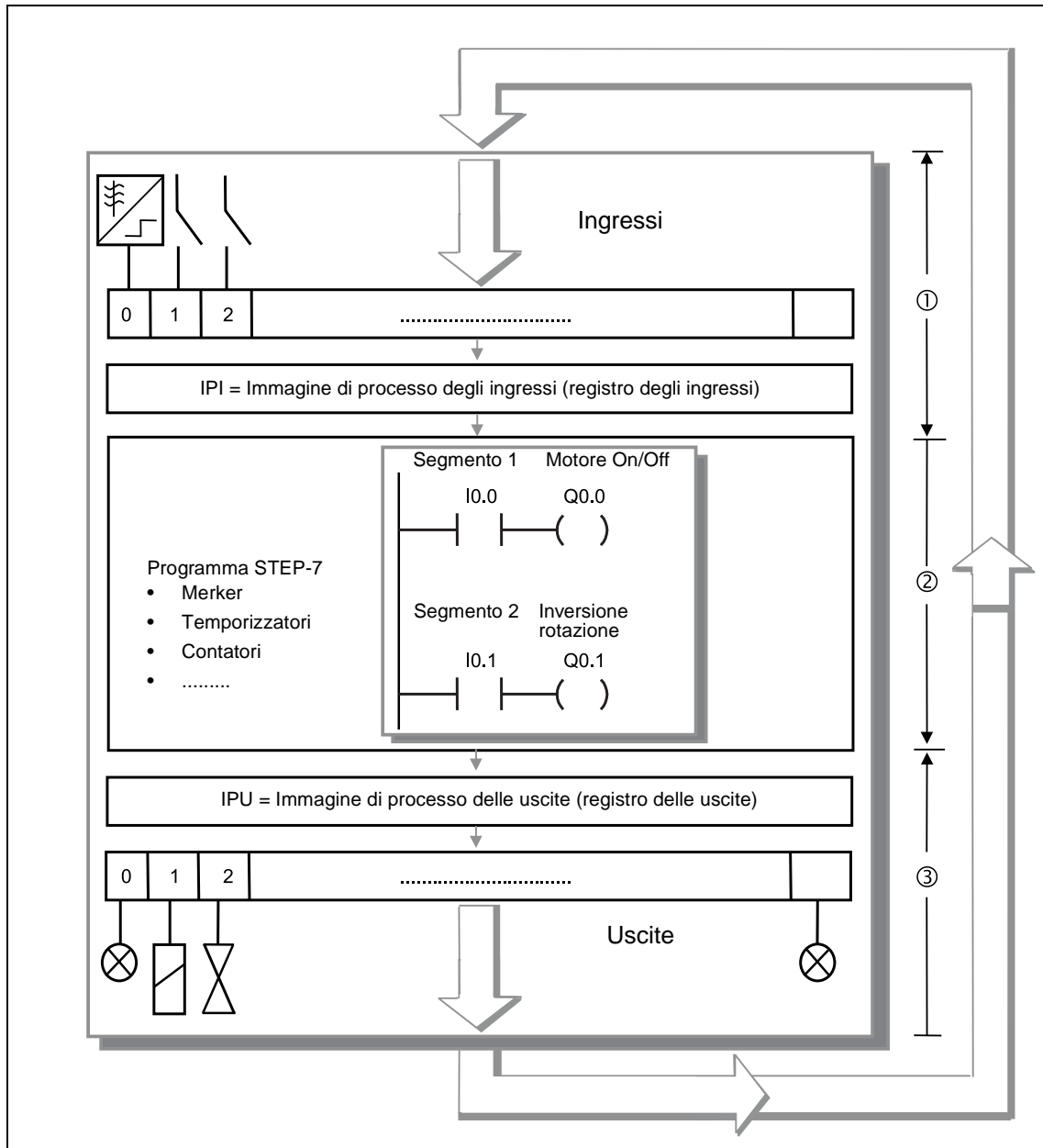
F1 Guida on-line



- Due parole sulla ripetizione
- Ecco i bit
- Flusso di corrente nello schema a contatti
- Il ciclo del PLC



Il ciclo del PLC (1)



Tutti i controllori programmabili SIMATIC funzionano con il principio della ciclicità. Nel funzionamento ciclico vengono dapprima letti gli stati di commutazione sugli ingressi e memorizzati nell'immagine di processo degli ingressi (IPI). Queste informazioni vengono quindi trasmesse al programma del PLC che le elabora.

Immagine di processo degli ingressi: IPI

- Due parole sulla ripetizione
- Ecco i bit
- Flusso di corrente nello schema a contatti
- Il ciclo del PLC



Il ciclo del PLC (2)

Secondo la logica di commutazione del programma le uscite vengono descritte nell'immagine di processo delle uscite (IPU). Gli stati in questa immagine di processo vengono infine trasmessi alle uscite fisiche. Il ciclo ricomincia quindi da capo.

Immagine di processo delle uscite: IPU

La durata tipica di un ciclo è compresa tra 3 e 10 ms e dipende dal numero e dal tipo di istruzioni impiegate.

Il ciclo è costituito da due componenti principali:

- 1) tempo del sistema operativo, generalmente 1 ms; corrisponde alle fasi ① e ③ a pagina 9.
- 2) tempo di elaborazione dei comandi; corrisponde alla fase ② a pagina 9.

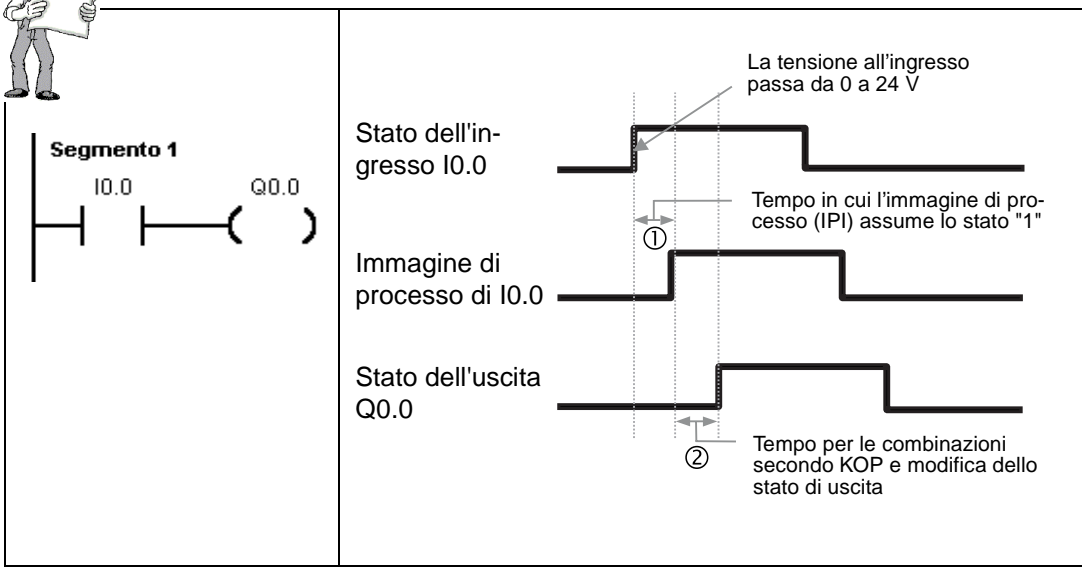
Inoltre i cicli vengono elaborati solo se il PLC "lavora", ossia se si trova nello stato operativo "RUN".



- Due parole sulla ripetizione
- Ecco i bit
- Flusso di corrente nello schema a contatti
- Il ciclo del PLC




Il ciclo del PLC (3)

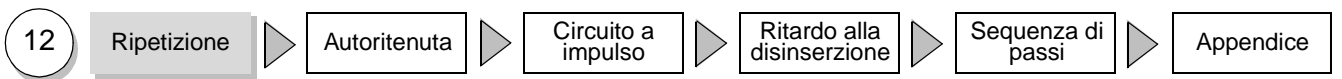


Le modifiche del segnale agli ingressi che si verificano durante un ciclo vengono inserite nel ciclo seguente nel registro degli ingressi. Qui gli stati del segnale vengono "congelati" per questo ciclo. Questa è l'immagine di processo degli ingressi IPI (cfr. ①).

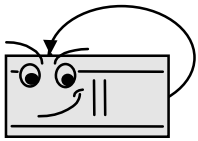
Nel ciclo successivo gli stati inseriti nel registro vengono combinati secondo lo schema a contatti (cfr. ②) e le uscite vengono aggiornate in base ai risultati della combinazione logica.

Modifica delle uscite solo al termine del ciclo successivo

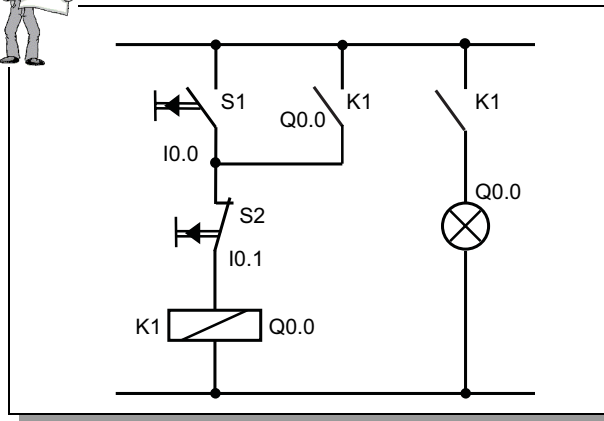
Ripetizione  **Nota**



- Introduzione
- Contatto normalmente chiuso
- Descrizione della soluzione e test
- L'autoritenuta in un'altra ottica



Introduzione



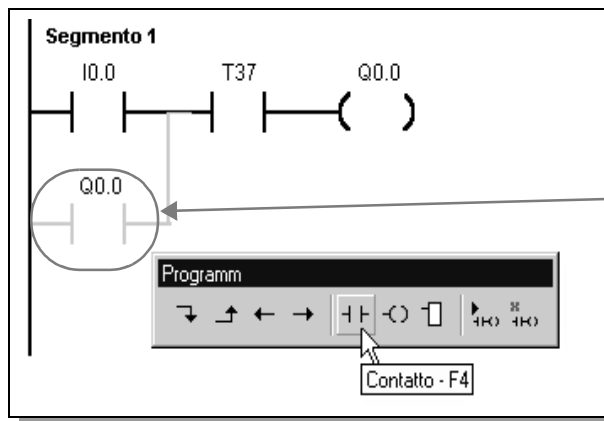
In questa sezione imparerete a programmare un tipo di autoritenuta tradizionale che vi è certamente nota. Autoritenuta tradizionale

L'esempio:

L'uscita Q0.0 deve essere attivata non appena viene azionato S1 sull'uscita I0.0. Tramite l'autoritenuta l'uscita Q0.0 deve restare attiva finché non viene azionato S2 sull'ingresso I0.1 interrompendo così l'autoritenuta.



Aprirete in STEP 7-Micro/WIN il primo progetto di esercitazione "a:\d01.mwp" dal dischetto. Nel programma mancano ancora alcuni elementi. Inserite voi come esercizio gli elementi KOP mancanti.



Affinché l'autoritenuta funzioni, l'uscita (qui I0.0) - non appena attivata - deve fare in modo di mantenere il proprio stato "vero" e quindi restare attiva.

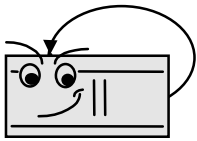
Ciò è reso possibile dal fatto che l'uscita (qui I0.0) viene collegata in parallelo all'ingresso da attivare, esattamente come per un normale circuito basato su contattori (I0.0 è paragonabile al nostro contattore K1).

L'uscita Q0.0 come ingresso provvede all'autoritenuta

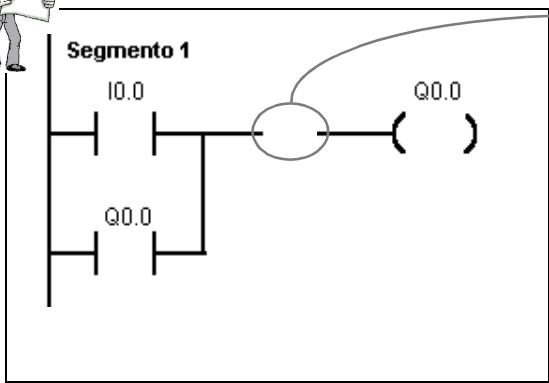
Inserite nel punto contrassegnato un contatto I0.0 (indicato dalla linea grigio) in parallelo a I0.0 procedendo come segue

- 1) Fate clic con il pulsante sinistro del mouse sul campo dello schema a contatti e premete l'icona del contatto normalmente aperto (F4) di STEP 7-Micro/WIN. Come indicato sull'icona, anziché il mouse si può anche usare il tasto funzione F4.
- 2) Per inserire la linea verticale, selezionate con il mouse il campo dello schema a contatti di I0.0 e premete l'icona (F7).

- Introduzione
- **Contatto normalmente chiuso**
- Descrizione della soluzione e test
- L'autoritenuta in un'altra ottica



Contatto normalmente chiuso (1)



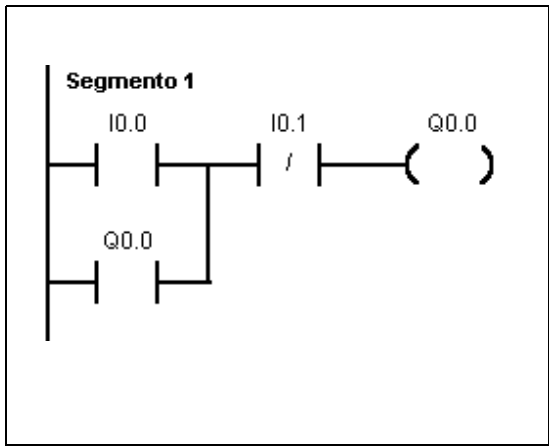
Affinché l'autoritenuta possa essere conclusa, l'ingresso IO.1 deve agire come un'interruzione del flusso di corrente quando viene azionato. Se la corrente viene interrotta (ossia in presenza dello stato "0") quando si attiva un interruttore, si parla di contatto normalmente chiuso.

Si deve pertanto inserire un elemento che opera nello schema a contatti come contatto normalmente chiuso quando vi sono DC 24V ("vero") all'ingresso IO.1.

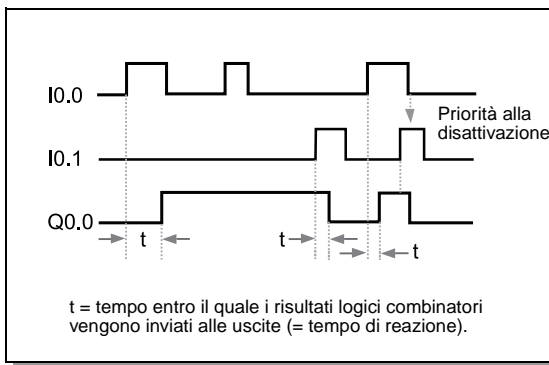
Contatto normalmente chiuso:



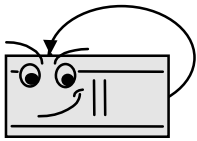
Inserite un contatto normalmente chiuso per l'interruttore S1 su IO.1. Questa operazione è descritta alla pagina seguente!



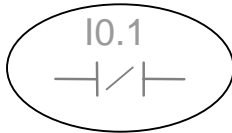
Così si presenta l'autoritenuta conclusa!
Qui sotto è riprodotto il funzionamento nel diagramma di tempo.



- Introduzione
- **Contatto normalmente chiuso**
- Descrizione della soluzione e test
- L'autoritenuta in un'altra ottica

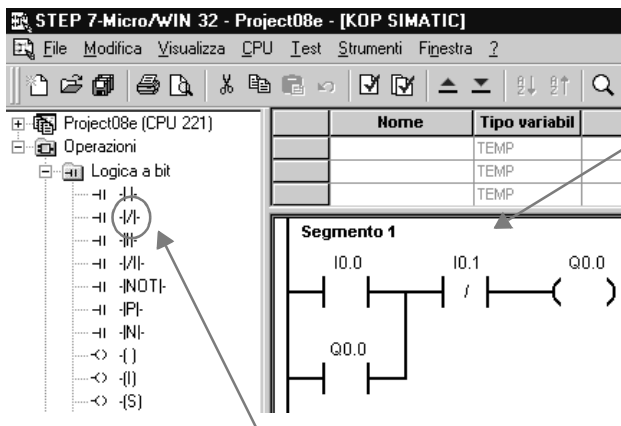


Contatto normalmente chiuso (2)



Un contatto normalmente chiuso interrompe il "flusso di corrente" nello schema a contatti se l'ingresso o l'uscita ad esso abbinata assume lo stato "vero".

Inserite come segue il contatto normalmente chiuso:

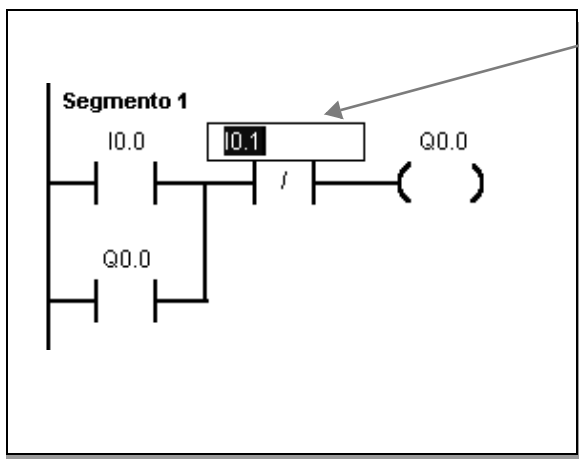


1. Selezionate con un clic del mouse il punto che deve essere sostituito da un contatto normalmente chiuso

Selezionare

2. Selezionate con un clic del mouse il contatto normalmente chiuso da una delle due barre dei pulsanti per lo schema a contatti disponibili in STEP 7-Micro/WIN.

Il contatto normalmente chiuso viene quindi inserito nel campo selezionato.



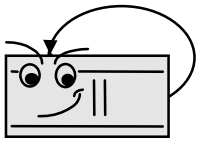
3. Infine dovete ancora assegnare al contatto normalmente chiuso l'elemento desiderato (qui I0.1). Ciò avviene mediante introduzione nel campo di testo già attivato e selezionato.

Assegnare

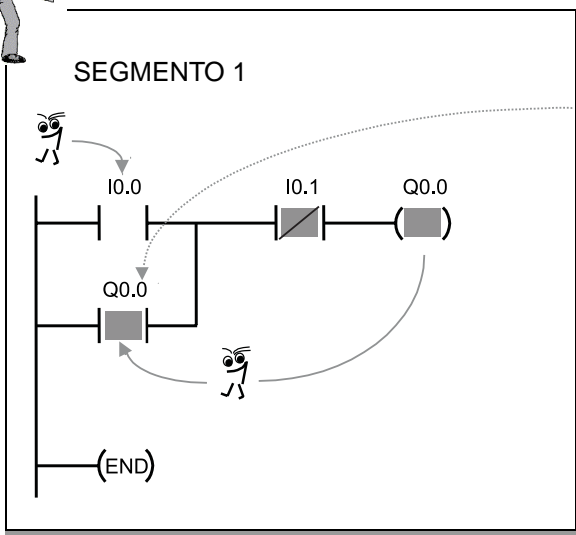
4. Concludete sempre le immissioni nel campo di testo premendo il tasto Invio.

Invio

- Introduzione
- Contatto normalmente chiuso
- Descrizione della soluzione e test
- L'autoritenuta in un'altra ottica



Descrizione della soluzione e test



Come nei circuiti di contattori, anche qui un contatto dell'uscita (Q0.0) è collegato in parallelo all'elemento da azionare (I0.0).

L'uscita I0.0 in parallelo all'ingresso si mantiene da sola

Se in un ciclo è stata attivata l'uscita Q0.0 tramite l'interruttore S1 in I0.0, già nel ciclo successivo (dopo pochi millisecondi) il contatto Q0.0 appare collegato in parallelo a I0.0. Ciò provoca l'autoritenuta. Il contatto normalmente chiuso I0.1 può interrompere questa condizione di autoritenuta se viene azionato l'interruttore S2 ad esso collegato in I0.1.



Salvate il programma completato sul disco fisso. In questo modo lo si potrà ricaricare per intero e rielaborare in qualsiasi momento (questo programma servirà più avanti per l'esempio riguardante il ritardo alla disinserzione).

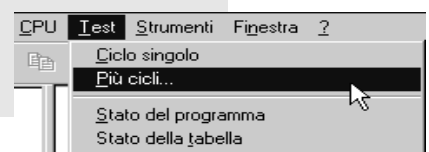


Trasferite infine il programma al PLC per testare la funzione.

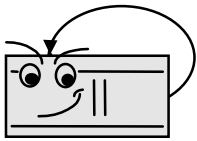


Per eseguire il test occorre commutare il PLC nello stato "RUN".

Testate il programma tramite i due interruttori del simulatore collegati agli ingressi I0.0 e I0.1. Osservate gli indicatori luminosi dell'S7-200 o controllate lo stato KOP!
 Inserite per prima cosa I0.0.
 I0.1 deve essere disattivato. Il LED di I0.0 deve accendersi.
 Q0.0 si accende.
 Non appena si attiva I0.1, l'uscita Q0.0 assume il valore "0".



- Introduzione
- Contatto normalmente chiuso
- Descrizione della soluzione e test
- L'autoritenuta in un'altra ottica

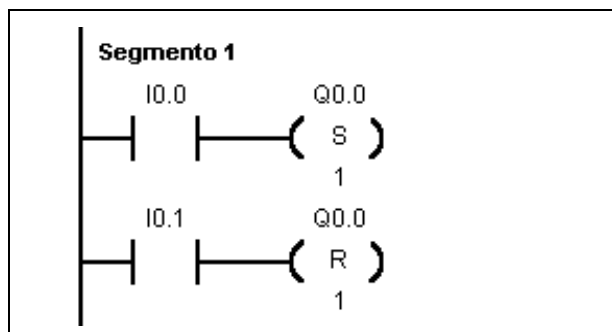


L'autoritenuta in un'altra ottica ... (1)



Nella tecnica dei PLC l'autoritenuta viene realizzata spesso anche in un'altra variante:

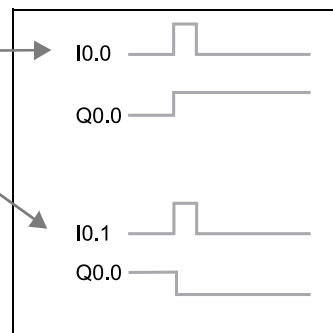
Anziché ricorrere alla retroazione dell'uscita - come nell'esempio precedente - si impiegano semplicemente le funzioni "Imposta" e "Resetta". Osservate questo schema a contatti:



Un impulso di commutazione in I0.0 determina, con l'operazione "Imposta" - **(S)**, l'inserimento con tenuta di Q0.0. -(S)
Imposta

Un impulso di commutazione in I0.1 determina invece, con l'operazione "Resetta" - **(R)**, la disinserimento di Q0.0. -(R)
Resetta

Le "bobine" - **(S)** che impostano Q0.0 su "1"
- **(R)** che resettano Q0.0 a "0"
vengono spesso impiegate nella tecnica dei PLC per inserire o disinserire permanentemente uscite e merker con un'attivazione breve (impulso) tramite un contatto a monte.



- **(S)** ⇔ 1

- **(R)** ⇔ 0

Un'uscita o un merker "impostati" restano tali finché non vengono nuovamente resettati ("falso") da un'istruzione - **(R)**.

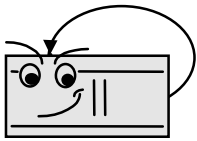
Valore impostato con (S) e resettato con (R)

Se sulla bobina di uscita che imposta e su quella corrispondente che resetta viene a trovarsi il segnale "1", la priorità spetta all'ultima operazione nella sequenza del programma.

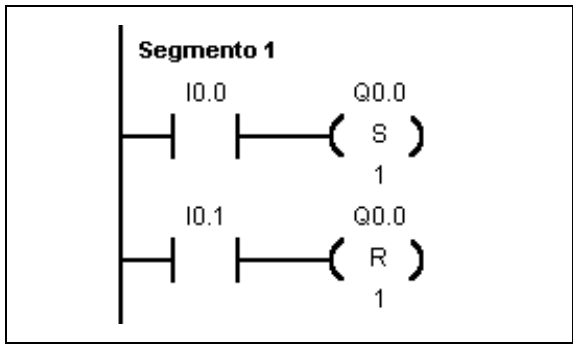
Priorità all'ultima operazione del ciclo



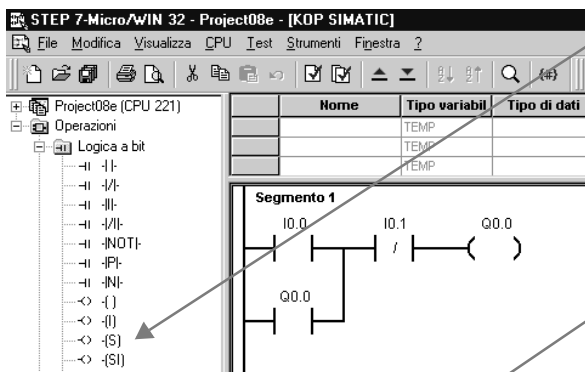
- Introduzione
- Contatto normalmente chiuso
- Descrizione della soluzione e test
- L'autoritenuta in un'altra ottica



L'autoritenuta in un'altra ottica ... (2)



Avete già imparato a inserire gli ingressi I0.0 e I0.1. Procedete come segue per inserire le bobine di impostazione e di resettaggio:



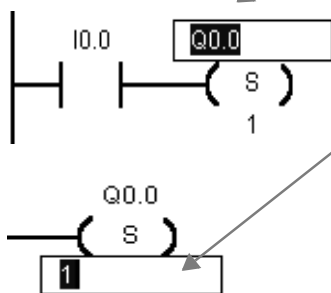
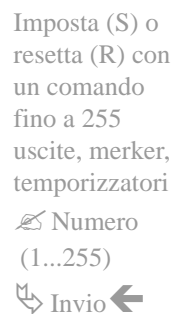
1. Dopo aver selezionato il campo KOP desiderato, selezionate con un clic del mouse "-(S)-" dalla lista dei gruppi di operazioni.



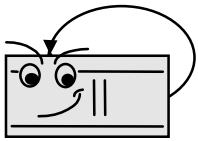
2. Digitate nel campo di testo già attivato l'indirizzo di uscita da influenzare, in questo caso Q0.0.



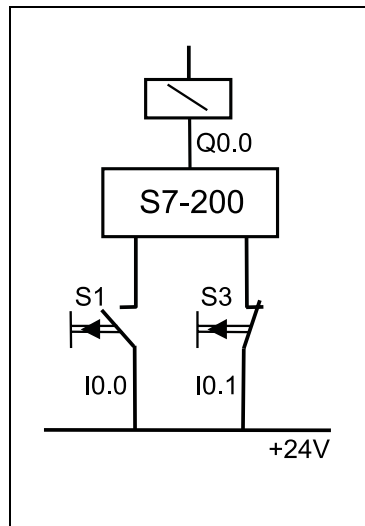
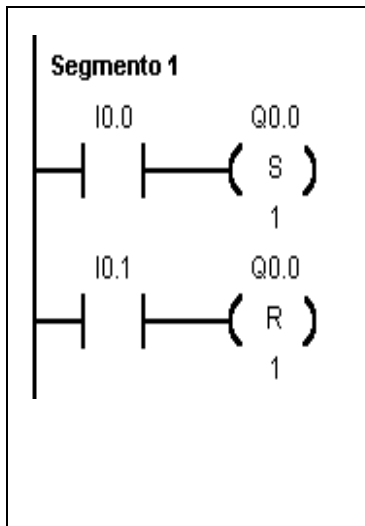
3. Definite infine quale numero di uscite (max. 255), temporizzatori o merker devono essere influenzati nello stesso modo a partire dall'indirizzo selezionato (in questo caso un solo bit).



- Introduzione
- Contatto normalmente chiuso
- Descrizione della soluzione e test
- L'autoritenuta in un'altra ottica



Problematiche di sicurezza Disinserzione per rottura del cavo su S3



S3 è un interruttore con contatto normalmente chiuso che in caso di azionamento fornisce il segnale "0".

In KOP questo segnale viene invertito dal contatto normalmente chiuso I0.1.



Ciò significa che quando si aziona l'interruttore S3, l'uscita Q0.0 viene resettata.



Avvertenze di sicurezza

- La soluzione suddetta utilizza per il resettaggio un interruttore S3 con contatto normalmente chiuso. Azionando I0.0 l'uscita Q0.0 viene impostata in modo permanente. Se su I0.1 vi sono +24 V, il "contatto normalmente chiuso" assume in KOP lo stato "0". L'uscita Q0.0 non viene resettata. Il "flusso di corrente" in KOP viene interrotto e la bobina di resettaggio viene disattivata.

Se non vi è segnale (0V) su I0.1 (S3 è aperto), il contatto normalmente chiuso di I0.1 in KOP è = "1" e l'uscita viene resettata.

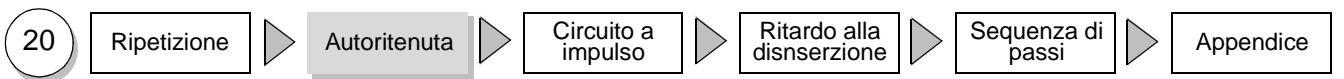
Se si impiega un interruttore con contatto normalmente chiuso in I0.1, l'uscita ad autoritenuta Q0.0 viene resettata (nuovamente disattivata) quando:

- viene azionato l'interruttore S3 (I0.1 = "0") oppure
 - si interrompe una linea di connessione tra I0.1 e l'interruttore con contatto normalmente chiuso. In questo modo, anche in caso di rottura del cavo, è garantito che un componente dell'impianto azionato in autoritenuta, come per es. un motore, venga disinserito.
- L'operazione "Resetta Q0.0" è stata inserita in sequenza dopo l'operazione "Imposta Q0.0", poichè in questo modo - nel caso di azionamento contemporaneo dei due interruttori - la conclusione dell'autoritenuta avrà la priorità.

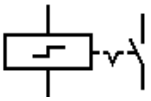


Aprire in STEP 7-Micro/WIN l'esempio finito "a:ld02.mwp" dal dischetto e provate le funzioni

Autoritenuta  Nota



- Introduzione
- La soluzione nel suo insieme
- Rilevamento del fronte
- Merker
- Descrizione della soluzione e test

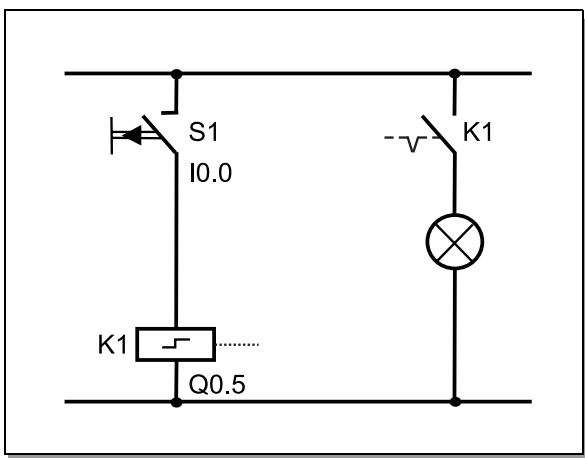


Introduzione



In questa lezione imparerete a realizzare un cosiddetto circuito a impulso di corrente e farete conoscenza con il rilevamento del fronte di impulso dei segnali e con i merker.

Funzionamento

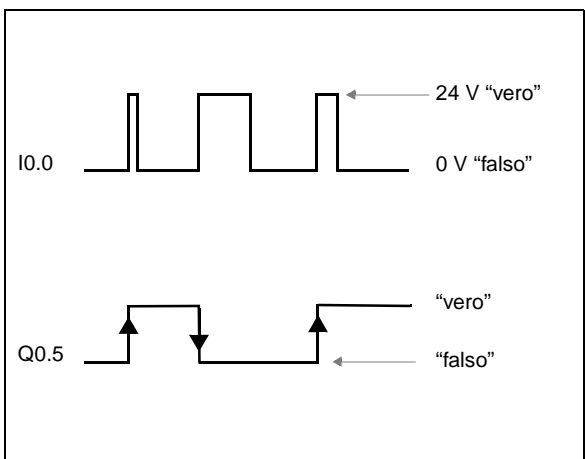


Una lampada all'uscita Q0.5 deve accendersi non appena viene brevemente azionato S1 all'ingresso I0.0.

Se S1 (I0.0) viene nuovamente azionato, Q0.5 deve cadere e la lampada deve spegnersi. Ad ogni azionamento dell'interruttore S1 l'uscita Q0.5 deve invertire il proprio stato.

Si tratta di un cosiddetto "circuito a impulso di corrente".

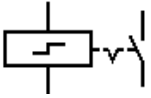
Diagramma di tempo



L'uscita Q0.5 deve sempre invertire **una volta** il proprio stato attuale quando l'interruttore in I0.0 commuta da "aperto" a "chiuso".

Se l'interruttore resta chiuso o aperto non si verifica alcun cambiamento

- Introduzione
- **La soluzione nel suo insieme**
- Rilevamento del fronte
- Merker
- Descrizione della soluzione e test



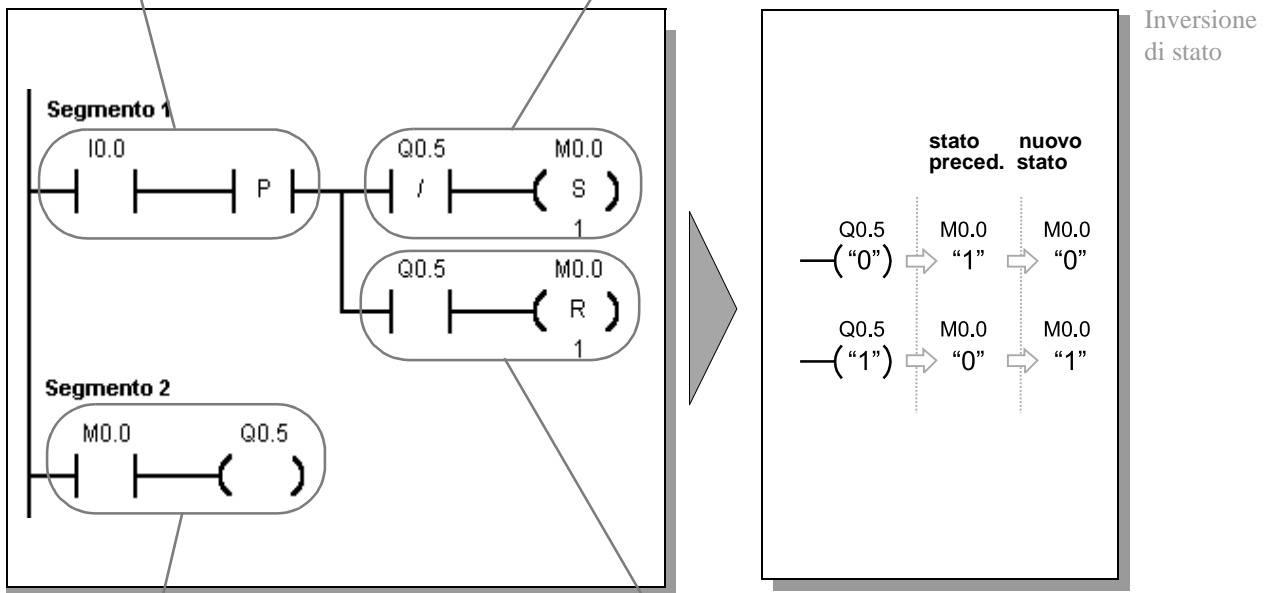
La soluzione nel suo insieme



Per avere una visione globale illustriamo la soluzione finita prima della sequenza di operazioni su cui si essa basa.

Rileva se su I0.0 si è verificato un cambiamento di stato da "0" a "1" (= fronte positivo).

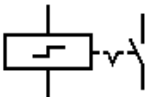
Se l'uscita Q0.5 è "0", viene impostato il merker M0.0, il quale "annota" che nel segmento 2 Q0.5 deve diventare "1".



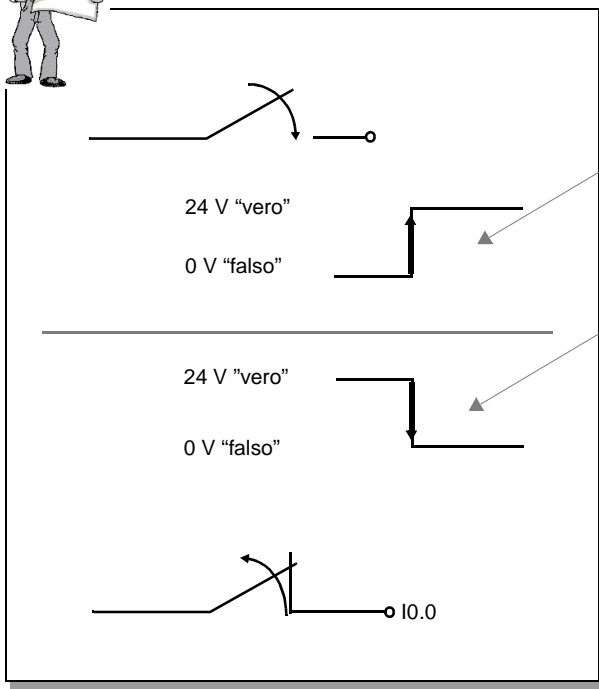
Assegna lo stato di M0.0 all'uscita Q0.5.

Se l'uscita Q0.5 è "1", il merker M0.0 viene resettato e "annota" che Q0.5 nel segmento 2 deve diventare "0".

- Introduzione
- La soluzione nel suo insieme
- Rilevamento del fronte
- Merker
- Descrizione della soluzione e test



Rilevamento del fronte (1)



L'istante in cui un contatto (ingresso, uscita ...) passa da "aperto" a "chiuso" o da "falso" a "vero" è detto fronte di salita o fronte positivo.



Riconoscimento del fronte di salita

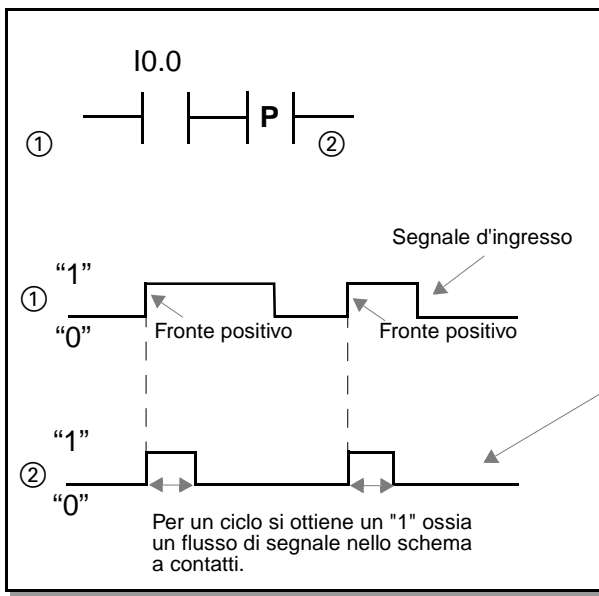
Analogamente il passaggio da "chiuso" ad "aperto" o da "vero" a "falso" è detto fronte di discesa o fronte negativo.



Riconoscimento del fronte di discesa

Per rilevare questi stati l'S7-200 dispone di due funzioni —|N|— e —|P|— che riconoscono il fronte di salita o di discesa.

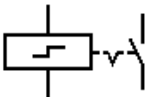
In questo esempio la funzione —|P|— viene utilizzata come segue:



Così si presenta il segnale generato dalla funzione —|P|—



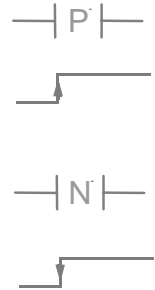
- Introduzione
- La soluzione nel suo insieme
- Rilevamento del fronte
- Merker
- Descrizione della soluzione e test



Rilevamento del fronte (2)

Il contatto P per il riconoscimento dei fronti di salita viene chiuso per la durata di un ciclo quando il contatto a monte passa da "falso" a "vero".

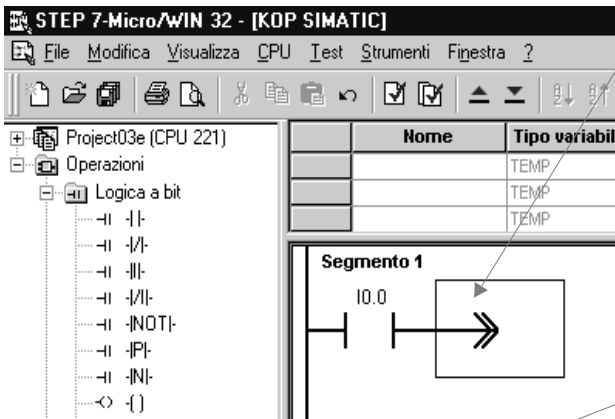
Analogamente, il contatto N per il riconoscimento dei fronti di discesa viene chiuso per la durata di un ciclo quando il contatto a monte passa da "vero" a "falso".



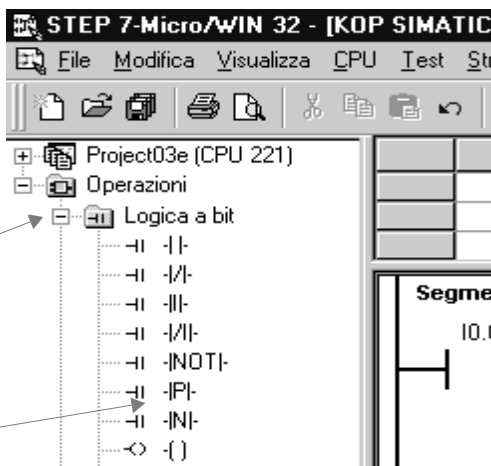
In questo esempio di "circuito a impulso di corrente" si utilizza pertanto P per inviare un segnale alle combinazioni successive solo nel momento in cui viene azionato il pulsante in I0.0.

Ecco come introdurlo ...

Aprire in STEP 7-Micro/WIN il progetto "a:\d03.mwp" dal dischetto. Anche questo progetto è incompleto e dovrà essere completato passo dopo passo.



1. Selezionate con un clic del mouse il punto che deve essere sostituito da un riconoscimento del fronte.



Selezionare

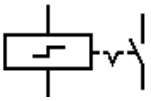
2. Selezionate con un clic del mouse "Logica a bit" dalla gruppi di operazioni.

Selezionare

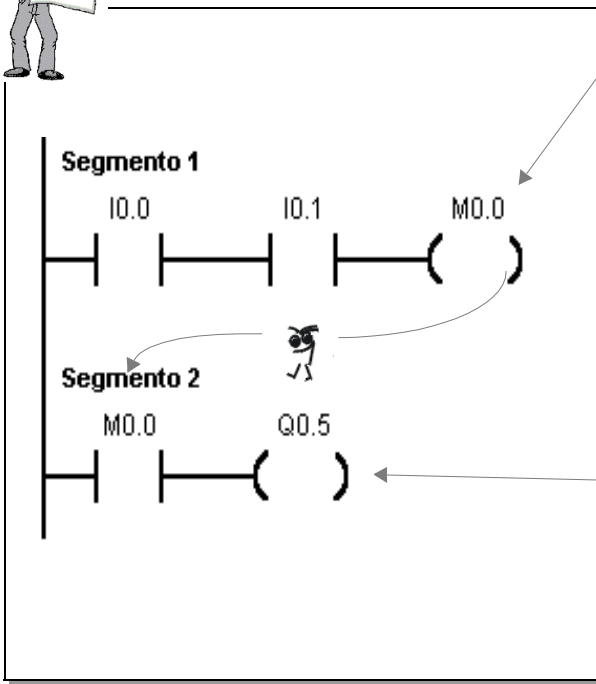
3. Dalla finestra successiva delle operazioni selezionate "Transizione positiva" P o "Transizione negativa" N .

Fronte

- Introduzione
- La soluzione nel suo insieme
- Rilevamento del fronte
- Merker
- Descrizione della soluzione e test



Merker (1)



Per realizzare il circuito a impulso di corrente si devono impiegare i merker. Questo esempio ne mostra il funzionamento.

Il merker di bit "M0.0" viene utilizzato come locazione di memoria interna del PLC per il risultato intermedio della combinazione logica "IO.0 AND IO.1" anziché come uscita.

In questo segmento il merker di bit viene usato come "contatto normalmente aperto di ingresso" e comanda pertanto l'uscita Q0.3. Il merker può essere utilizzato anche in altri punti del programma.

Utilizzabile illimitatamente come contatto normalmente chiuso o aperto

I merker consentono di memorizzare risultati parziali, analogamente alla memoria di una calcolatrice.

Nella tecnica dei PLC i merker vengono utilizzati come uscite e, nel funzionamento, sono simili ai contattori ausiliari. Un merker può essere impiegato in qualsiasi punto come contatto normalmente chiuso o aperto.

Impiego come uscita

Effetto di un contactore ausiliario

In caso di caduta della tensione di esercizio il contenuto del merker va perduto. Il compito della "rimanenza" è di ovviare a questo inconveniente.

Il contenuto dei merker è subito disponibile (nello stesso ciclo) per le combinazioni logiche successive.

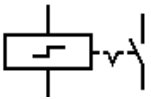
Il contenuto viene subito aggiornato

I merker si usano quando il risultato (intermedio) di un segmento deve essere rielaborato in altri segmenti (come le somme parziali nell'addizione manuale di numeri) oppure per memorizzare temporaneamente gli stati successivi analizzati.

Scrivibile più volte con -(S) o (R)
Con -()- si assegna **una sola volta**



- Introduzione
- La soluzione nel suo insieme
- Rilevamento del fronte
- **Merker**
- Descrizione della soluzione e test



Merker (2)

Ora conoscete la funzione dei merker e potete comprendere la soluzione del circuito a impulso di corrente.

Ad ogni pressione del pulsante in I0.0 la funzione $\neg P$ permette il flusso di segnale per un ciclo (riconoscimento del fronte) nel segmento 1.



Per ogni fronte $\neg P$ l'uscita Q0.5 deve invertire il suo stato.

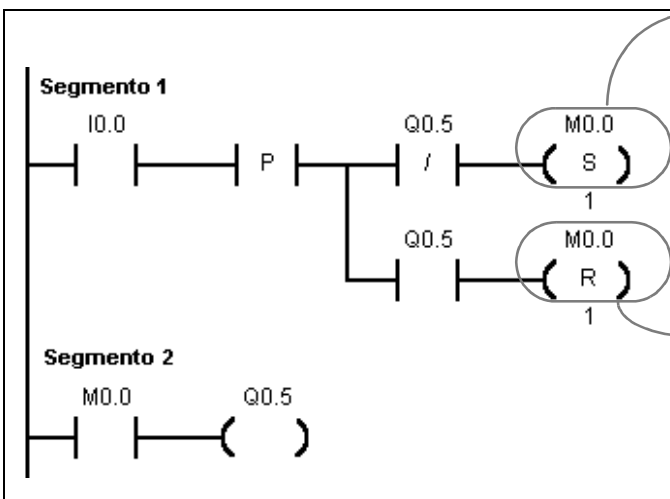
Lo stato invertito (stato successivo) non viene scritto direttamente nell'uscita Q0.5, perché l'uscita appena impostata nella diramazione "superiore" è stata subito resettata nella diramazione "inferiore". Lo stato successivo verrà quindi scritto nel merker M0.0 (= salvare prima di sovrascrivere).

Nel segmento 2 lo stato "impostato" del merker viene assegnato all'uscita.

-(S)
Imposta

-(R)
Resetta

"Salvare" lo stato successivo nel merker prima di sovrascrivere



In questo punto si deve inserire una bobina per impostare il merker M0.0. Il numero sotto la bobina indica quanti elementi devono essere impostati a partire dall'indirizzo iniziale specificato. In questo caso: impostazione di un bit a partire dal merker M0.0.

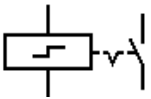
M0.0 viene impostato se Q0.5 non era attiva ("falso")

Dato che la diramazione inferiore realizza la funzione inversa a quella superiore, il bit del merker M0.0 deve essere resettato, quando in seguito alla pressione del pulsante questa diramazione diventa "conduttrice di corrente".

M0.0 viene resettato se Q0.5 era attiva ("vero")

Completate infine in STEP 7-Micro/WIN l'esempio nel vostro attuale progetto di esercitazione come indicato sopra.

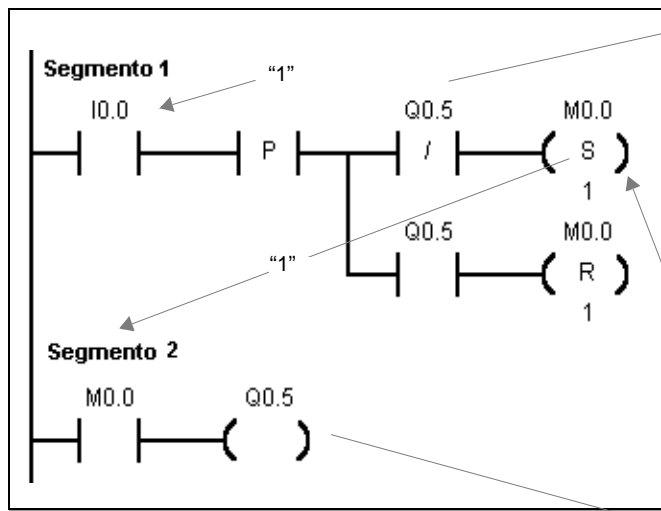
- Introduzione
- La soluzione nel suo insieme
- Rilevamento del fronte
- Merker
- Descrizione della soluzione e test



Descrizione della soluzione e test

Illustriamo in sintesi ancora una volta la funzione del programma ormai completato sull'esempio della diramazione superiore del segmento 1 (che termina con (S), ossia con il processo di inserzione).

Il "flusso di corrente" nello schema a contatti è rappresentato nel ciclo del fronte positivo in I0.0.



Se I0.0 viene momentaneamente azionato (riconoscimento del fronte -IPI-)

e
l'uscita Q0.5 nel ciclo attuale vale "0" (la diramazione superiore risulta vera quando interrogata con il contatto normalmente chiuso)

allora...
Annotare lo stato successivo di Q0.5 impostando il merker M0.0:
-(S) Impostazione di un bit a partire da M0.0

M0.0 ha già assunto qui lo stato successivo di Q0.5.

Q0.5 riceve il nuovo stato al termine del ciclo e pertanto appare solo nel ciclo successivo come "vero" o "1" nella rappresentazione KOP.



Salvate il programma finito sul disco fisso.



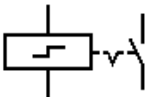
Trasferite il programma al PLC.



Per il test il PLC deve essere nel modo operativo "RUN".

Testate il programma: azionate l'interruttore in I0.0 e osservate l'uscita Q0.5.

- Introduzione
- La soluzione nel suo insieme
- Rilevamento del fronte
- Merker
- Descrizione della soluzione e test



E ora mettetevi alla prova

... dato che a questo punto avete appreso molte nozioni!

Leggete queste domande e provate a rispondere.

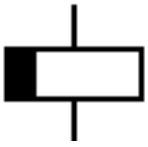
- ✓ Cos'è il ciclo di un PLC?
... quali sono i tre componenti principali del ciclo? v. pagina 9
- ✓ Come si realizza l'autoritenuta nella tecnica dei PLC? v. pagina 13
- ✓ Contatto normalmente chiuso: come si presenta nello schema a contatti (KOP), che effetto ha, quale misura di sicurezza consente di realizzare? v. pagina 14
- ✓ Cos'è un fronte, come viene rilevato e a quale scopo? v. pagina 23
- ✓ Cosa sono i merker e a cosa servono? v. pagina 25
- ✓ Come vengono inserite le bobine "Imposta" e "Resetta" e che effetto hanno? v. pagina 26



Avete sicuramente risposto con facilità a queste domande, anche se forse avete dovuto consultare nuovamente le pagine corrispondenti.

Ma almeno adesso è tutto compreso!

- Introduzione
- Salva con nome ...
- Inserimento segmento
- Descrizione della soluzione
- Introduzione dei commenti

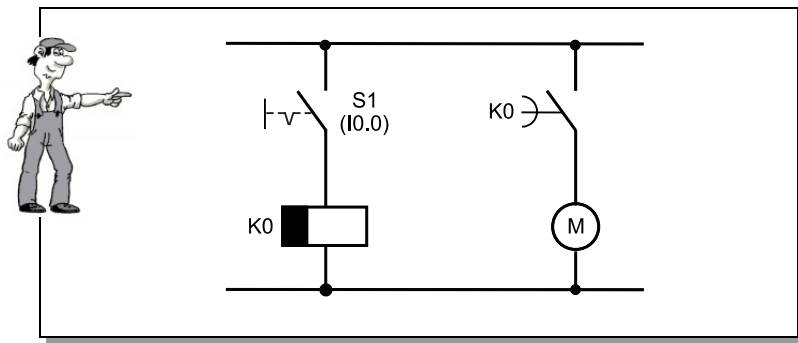


Introduzione

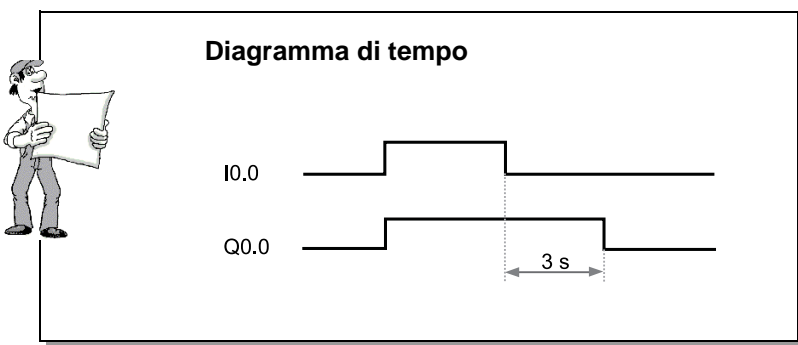


Nell'opuscolo "Un'ora per conoscerlo" avete già conosciuto il ritardo all'inserzione. Ora realizzeremo insieme un ritardo alla disinserzione

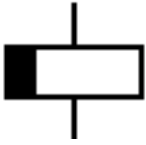
Azionando S1 (I0.0) si attiva un ventilatore all'uscita Q0.0. Quando S1 (I0.0) viene disattivato, il ventilatore deve continuare a funzionare per 3 secondi prima di fermarsi.



Se si disattiva S, il ventilatore deve continuare a funzionare per 3 secondi



- Introduzione
- Salva con nome ...
- Inserimento segmento
- Descrizione della soluzione
- Introduzione dei commenti



Introduzione



Procedura

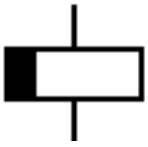
- 1) Carichiamo innanzitutto sul disco fisso il circuito di autotenuta completato del primo esempio.
- 2) Salviamo quindi l'esempio sul disco fisso con un nuovo nome.
- 3) Creiamo spazio tramite il comando "Inserisci segmento".
- 4) Completiamo infine insieme il ritardo alla disinserzione e aggiungiamo i commenti.
- 5) Testiamo il funzionamento del programma.

Nelle pagine successive eseguiremo insieme tutte le procedure necessarie per realizzare in maniera sicura il ritardo all'inserzione.



Buon lavoro!

- Introduzione
- **Salva con nome ...**
- Inserimento segmento
- Descrizione della soluzione
- Introduzione dei commenti



Salva con nome ...



Per il nostro progetto utilizzeremo il circuito di autoritenuta che abbiamo messo a punto nel primo capitolo.

Duplicate il progetto completo caricandolo e salvandolo subito con un nuovo nome.



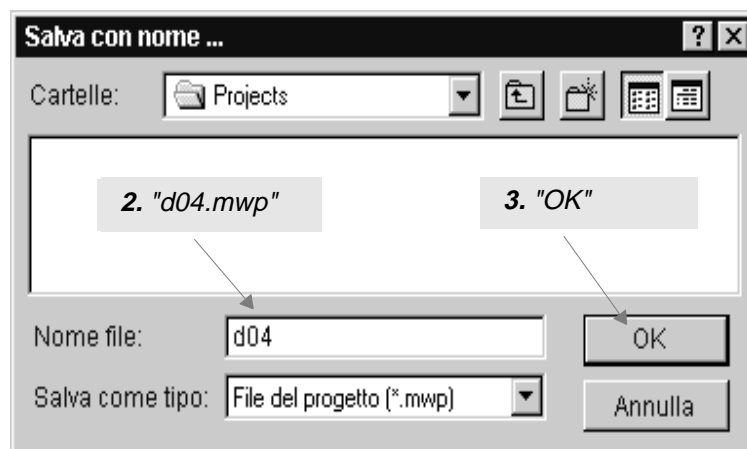
Caricate in STEP 7-Micro/WIN il progetto "d01.mwp" (autoritenuta) dal disco fisso, dove lo avevate salvato nel primo capitolo.

Adesso dovete salvare il progetto con un nuovo nome. Salvate il progetto, come descritto qui di seguito, con il nome "d04.mwp".



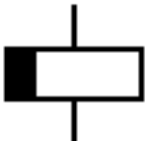
1. Richiamate la funzione di menu "File > Salva con nome ..."

➔ Menu: File, Salva con nome...



Progetto1.mwp
OK

- Introduzione
- Salva con nome ...
- **Inserimento segmento**
- Descrizione della soluzione
- Introduzione dei commenti



Inserimento segmento

Per realizzare il ritardo di disinserzione si deve inserire al posto del segmento 2 un segmento aggiuntivo, procedendo come segue:

1. Attivate con un clic del mouse il campo con il titolo Segmento 2.

2. Inserite un nuovo segmento al posto del segmento 2.

Inserisci segmento

Selezionare

Pulsante Segmento nella barra degli strumenti (F10)

Adesso avete creato lo spazio per il nuovo segmento 2, che utilizzerete per realizzare il ritardo alla disinserzione. I contenuti del segmento 2 originario sono "slittati" in avanti di un segmento.

Avvertanza:

Per creare spazio per inserire gli elementi KOP esiste anche questa possibilità:



3. Selezionate dal menu Modifica la sottovoce "Inserisci".

Menu: Modifica, Inserisce contenuti/segmento.

STEP 7-Micro/WIN 32 - Progetto1 - [KOP SIMATIC]

File Modifica Visualizza CPU Test Strumenti Finestra ?

Modifica menu:

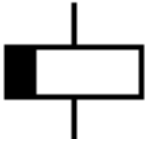
- Annulla Ctrl+Z
- Taglia Ctrl+X
- Copia Ctrl+C
- Incolla Ctrl+V
- Seleziona tutto Ctrl+A
- Inserisci**
- Cancella
- Trova... Ctrl+F
- Sostituisci... Ctrl+H
- Vai a... Ctrl+G

Sub-menu:

- Riga Ctrl+I
- Riga giù Shift+Ctrl+J
- Colonna
- Verticale
- Segmento**
- Interrupt
- Sottoprogramma

4. Selezionate "Segmento".

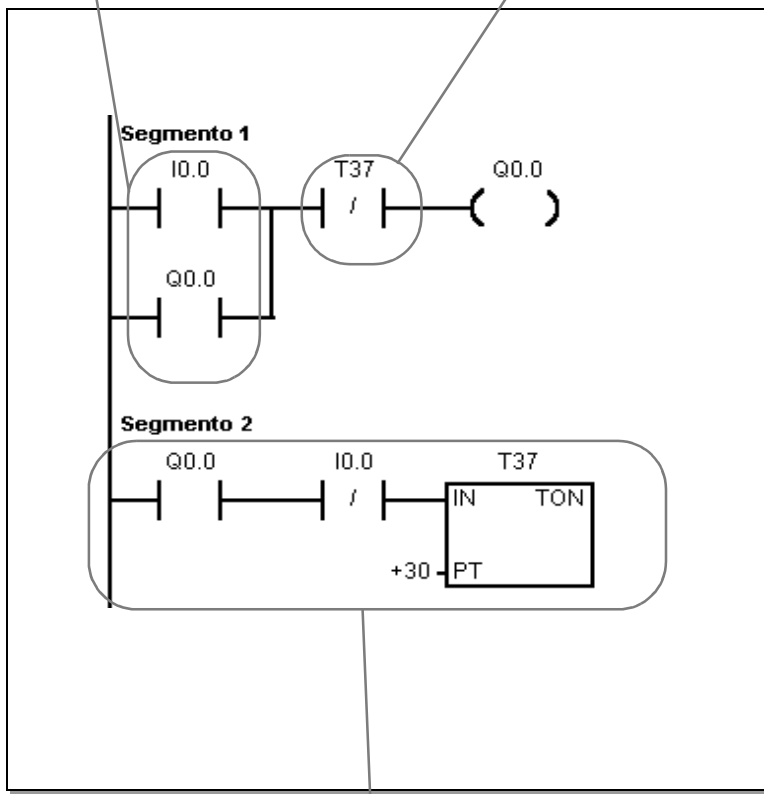
- Introduzione
- Salva con nome ...
- Inserimento segmento
- **Descrizione della soluzione**
- Introduzione dei commenti



La soluzione nel suo insieme

I0.0 attiva Q0.0.
Q0.0 si arresta da solo, dato che contemporaneamente è collegato in parallelo a I0.0.

Se T37 è scaduto, l'autoritenuta viene interrotta da questo contatto.
Il motore si arresta.
Se T37 non è scaduto, l'autoritenuta si "arresta".

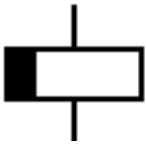


Così si presenta il programma finito.



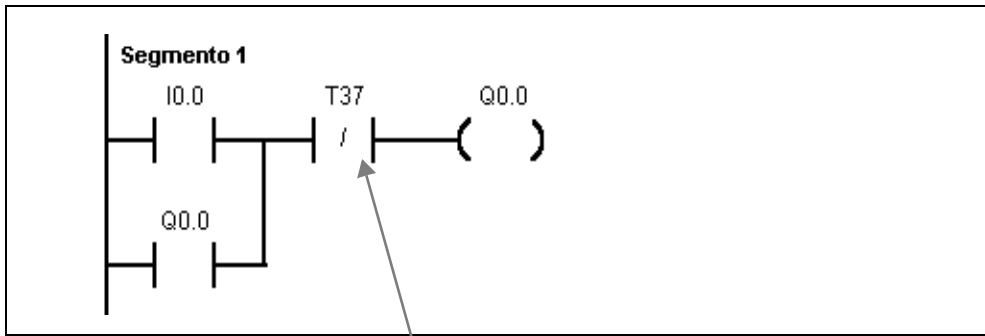
Se Q0.0 è azionato e I0.0 è di nuovo "0" (S1 non è più azionato), il temporizzatore T37 inizia a scorrere.

- Introduzione
- Salva con nome ...
- Inserimento segmento
- **Descrizione della soluzione**
- Introduzione dei commenti



Soluzione - Immissione del programma

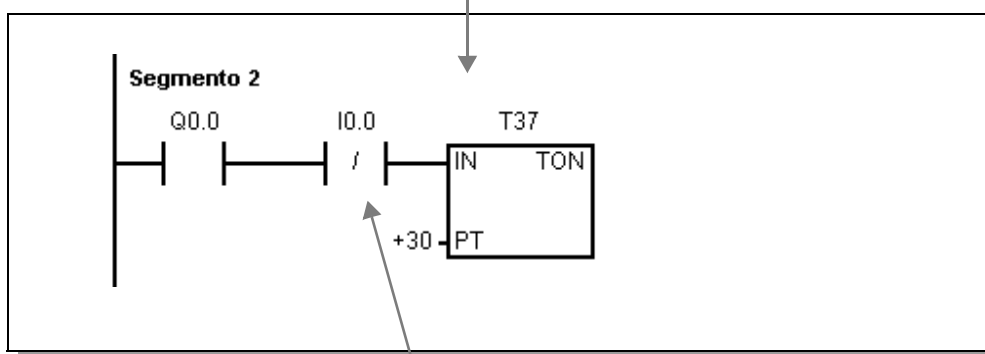
Il segmento 1 deve presentarsi così:



Sovrascrivete I0.1 del circuito di autoritenuta con T37.

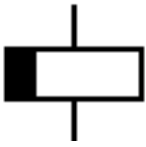
Inserite nel segmento 2 il seguente programma:

*Inserite T37 con:
F2 Temporizzatori/Contatori e
F3 Temporizzatore come ritardo all'inserzione.*



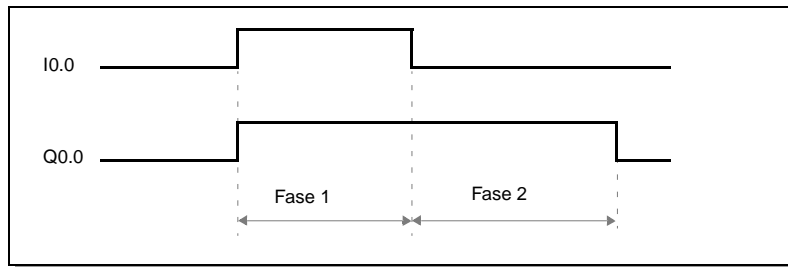
*T37 ha una base di tempo di 100 ms (cfr. anche
"Un'ora per conoscerlo", pagina 36).
Il valore di temporizzazione è quindi $30 * 100 \text{ ms} = 3 \text{ s}$.*

- Introduzione
- Salva con nome ...
- Inserimento segmento
- **Descrizione della soluzione**
- Introduzione dei commenti

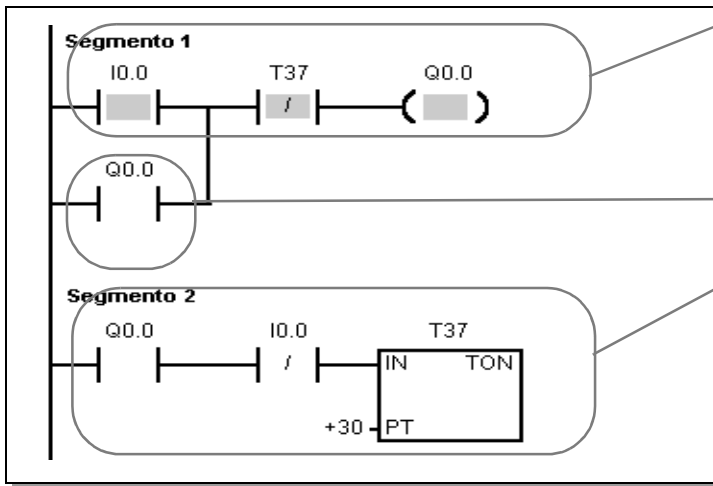


Descrizione della soluzione

Così funziona il nostro programma. Le fasi attive sono due.



Fase 1: Attivate l'autoritenuta, I0.0 vale "1" (si presume che l'uscita Q0.0 non sia attivata).



Se viene azionato I0.0

E

T37 non è scaduto

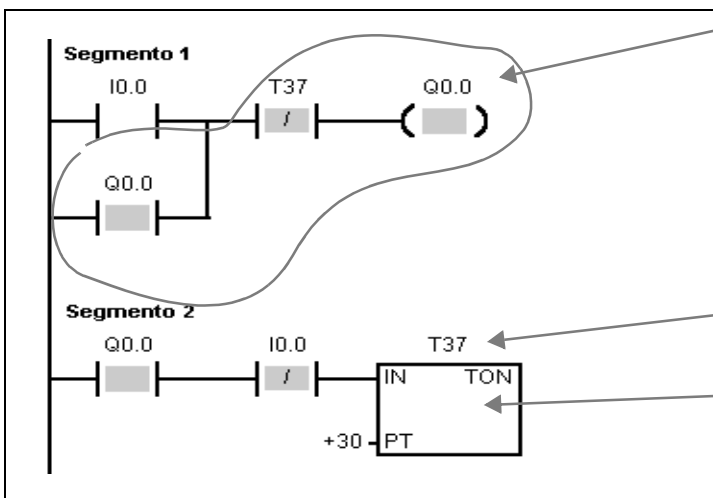
ALLORA

viene attivata Q0.0 (= "1").

Tramite questo contatto l'uscita Q0.0 si interrompe da sola.

T37 non è attivo perché I0.0 è ancora "1".

Fase 2:



I0.0 non è più azionato.

L'autoritenuta resta attiva finché non scade T37.

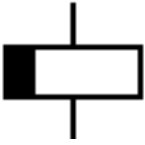
Fintanto che il tempo scorre, T37 vale "0" e il contatto normalmente chiuso lascia passare la corrente.

Nel funzionamento di test si vede scorrere qui il tempo.

Se Q0.0 è attivo E I0.0 non è più azionato, il temporizzatore T37 è in funzione.



- Introduzione
- Salva con nome ...
- Inserimento segmento
- Descrizione della soluzione
- **Introduzione dei commenti**

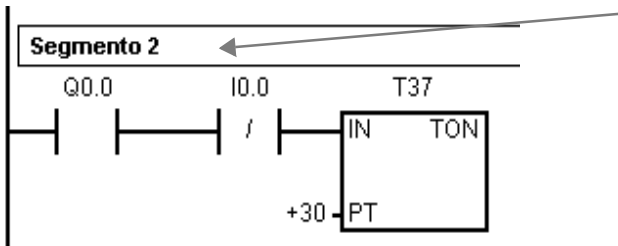


Introduzione dei commenti (1)



Salvate e provate il vostro nuovo programma! Quando azionate I0.0, l'uscita Q0.0 viene attivata. Quando I0.0 viene disinserito, Q0.0 si spegne dopo 3 secondi.

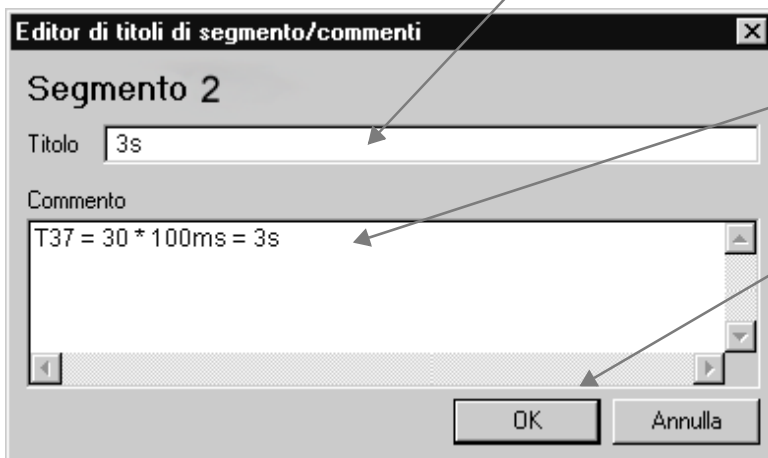
Ottimo lavoro! Probabilmente vorrete inserire nel programma delle informazioni relative al suo funzionamento, magari a vantaggio di progetti futuri (modifiche o altro). Abbiamo pensato anche a questo. E' infatti possibile inserire un titolo e dei commenti per ogni segmento. Ora vedremo come.



1. Fate doppio clic sul campo del titolo del Segmento 2.



2. Viene visualizzato l'editor dei commenti. Digitate qui il titolo del segmento ...



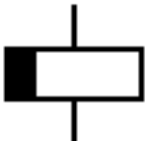
3. ... e qui il commento al segmento.



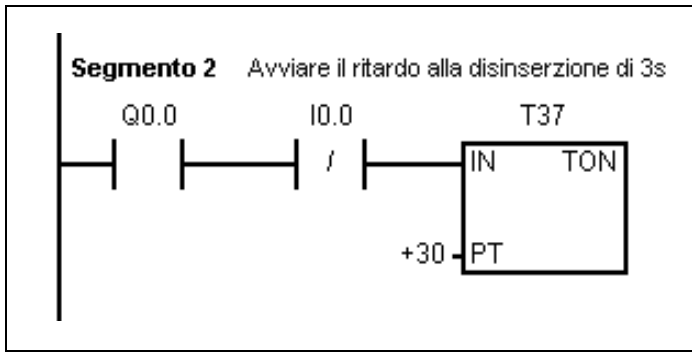
4. Concludete con OK l'introduzione.



- Introduzione
- Salva con nome ...
- Inserimento segmento
- Descrizione della soluzione
- Introduzione dei commenti

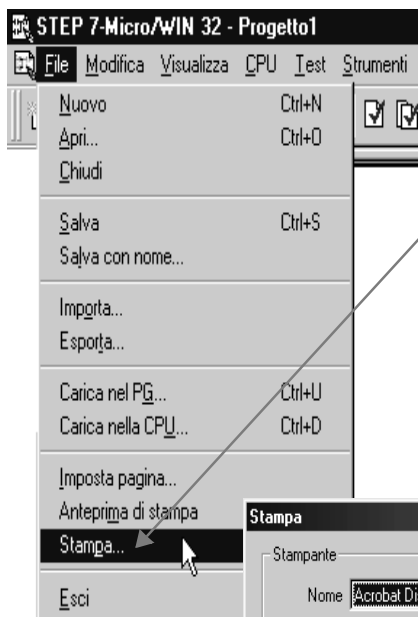


Introduzione dei commenti (2)



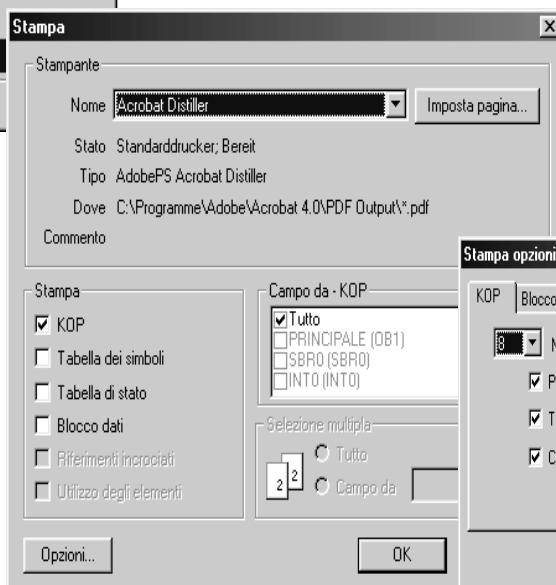
Dopo aver inserito il commento, sullo schermo si vede solo il titolo del segmento.

I commenti si possono poi visualizzare di nuovo attivando l'editor dei commenti.



Se volete che i commenti vengano stampati, attivate la funzione di menu "File >Stampa ...".

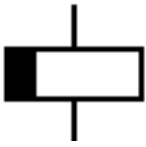
➔ Menu:
File, Stampa



Stampa
commenti del
segmento

OK

- Introduzione
- Salva con nome ...
- Inserimento segmento
- Descrizione della soluzione
- Introduzione dei commenti



E ora mettetevi alla prova

Leggete queste domande e provate a rispondere:

- ✓ Come si realizza un ritardo alla disinserzione? Disegnate lo schema a contatti per due soluzioni possibili. Uno schema con la normale bobina —()— e uno con **(S)** e **(R)**.
- ✓ Come si salva un progetto?
- ✓ Come si definisce il valore di un temporizzatore?
- ✓ Quali commenti si possono aggiungere ai segmenti?

v. pagina 29

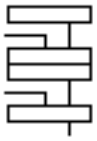
v. pagina 31

v. pagina 36 in
"Un'ora per ..."

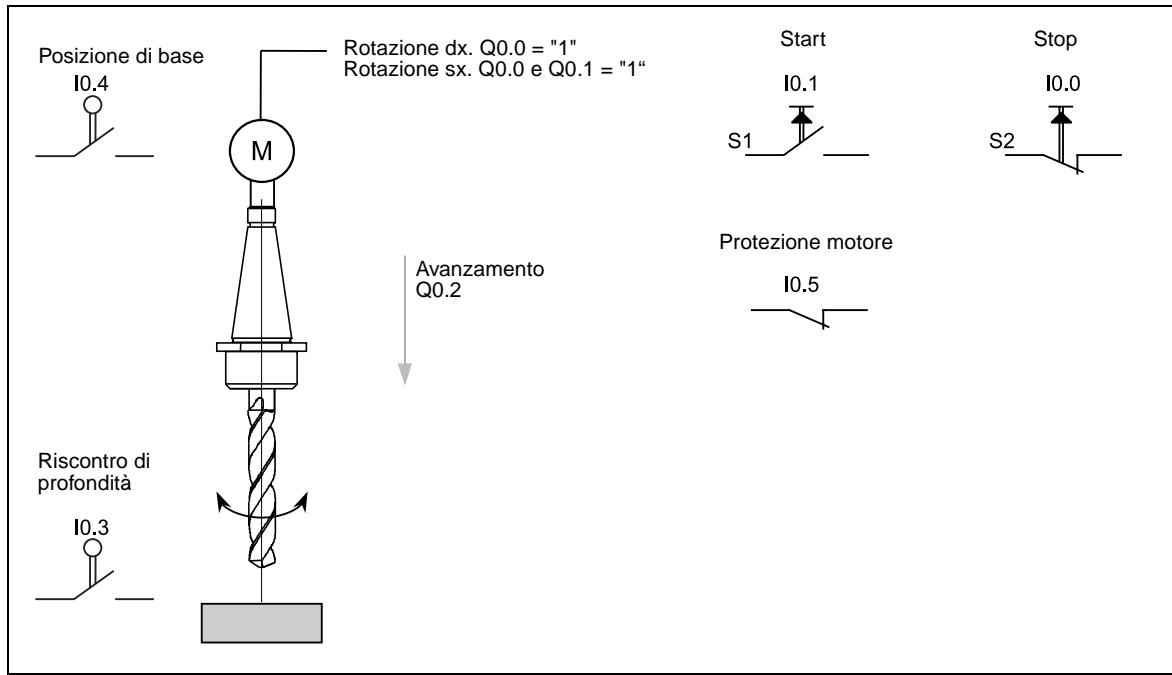
v. pagina 36



- Introduzione
- Fondamenti
- Uso delle sequenze di passi
- Modifica
- Descrizione della soluzione, esempio
- Test



Introduzione



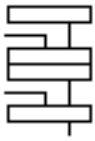
Realizziamo ora insieme una sequenza di passi.

Il motore di una trapanatrice viene avviato con rotazione destrorsa tramite S1. Dopo 3s viene inserito l'avanzamento. L'avanzamento viene disinserito quando viene raggiunto il riscontro di profondità in I0.3. Una molla riporta la macchina nella posizione di base. Il motore ruota in senso sinistrorso (Q0.0 e Q0.1 valgono "1"). Quando viene raggiunta la posizione di base I0.4 = "1", il motore gira ancora per 1s prima che la macchina venga completamente disinserita. Con Stop è sempre possibile disinserire la macchina (attivazione con I0.0 = "0").

Sequenza di passi

Comando a sequenza di passi

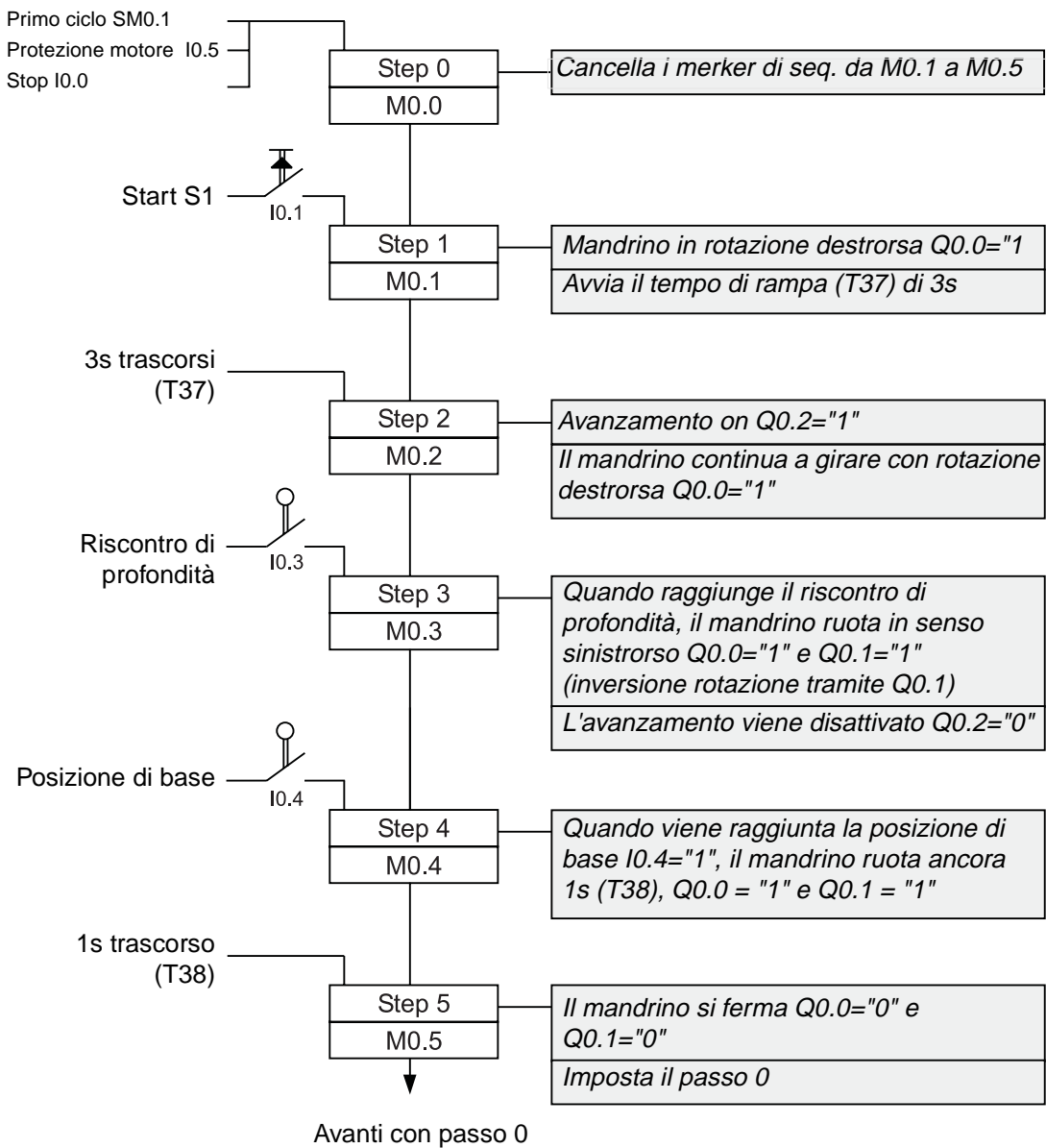
- Introduzione
- Fondamenti
- Uso delle sequenze di passi
- Modifica
- Descrizione della soluzione, esempio
- Test



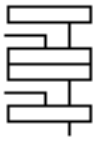
La soluzione



Così si presenta la soluzione per la sequenza di passi dell'esempio di trapanatrice



- Introduzione
- **Fondamenti**
- Uso delle sequenze di passi
- Modifica
- Descrizione della soluzione, esempio
- Test



Fondamenti (1)

Risolviamo ora il comando della trapanatrice mediante una sequenza di passi.

Cos'è un comando a sequenza di passi?



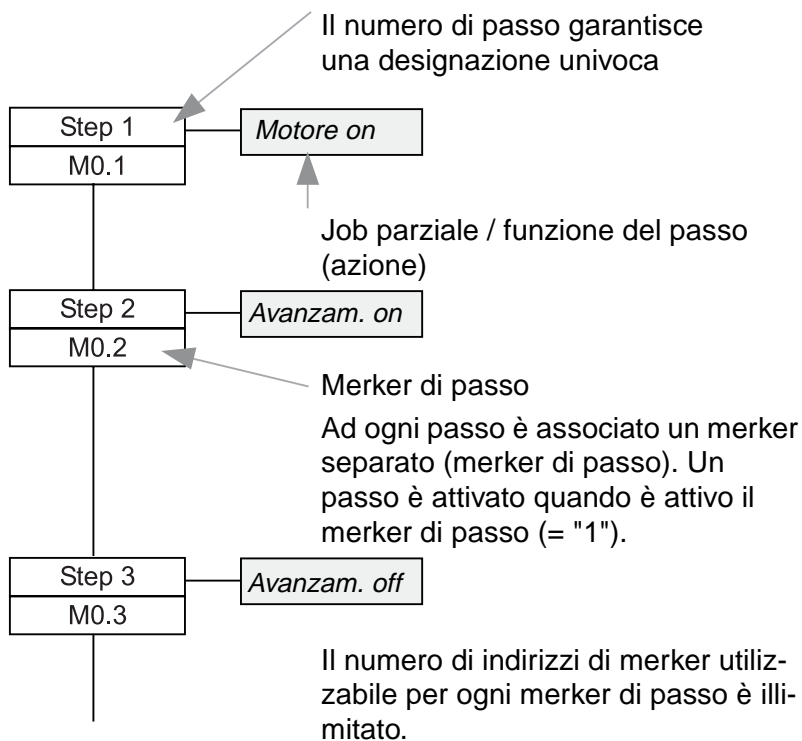
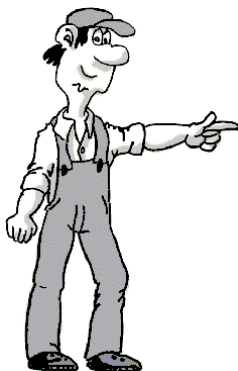
- Una tecnica di controllo in cui un job viene suddiviso in una sequenza di job parziali (per es. Motore on, Avanzamento on, Avanzamento off, ...).
- I job parziali (funzioni) sono detti passi.
- Normalmente deve essere elaborato un passo prima che possa iniziare quello successivo.
- Un nuovo passo diventa attivo quando è attiva la relativa condizione di proseguimento.
- Un passo è attivo quando il corrispondente merker di passo, per es. M0.1, è uguale a "1".

Passi

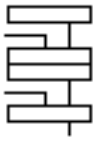
Condizione di proseguimento

Passo attivo
→
merker di passo
MX.Y = "1"

Per ogni stato importante viene definito un passo



- Introduzione
- **Fondamenti**
- Uso delle sequenze di passi
- Modifica
- Descrizione della soluzione, esempio
- Test



Fondamenti (2)



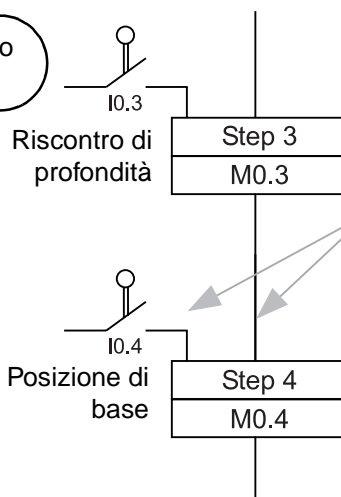
Cos'è una condizione di proseguimento?

- Ogni passo viene avviato (attivato) da una condizione. La condizione viene generalmente derivata dagli stati della macchina. Questi possono essere finecorsa attivati, tasti di comando, temperature raggiunte o tempi.
- Quasi sempre una parte della condizione è data dal fatto che vi sia un passo attivo precedente.
- Se viene impostato un nuovo merker di passo, il merker di passo del passo precedente viene resettato.

La condizione di proseguimento attiva il merker di passo

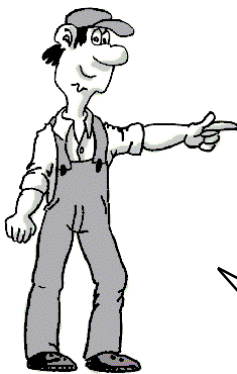
Merker di passo attivo "1"

Attivare sempre un solo passo.



Condizione affinché il passo 4 diventi attivo:

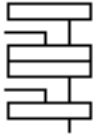
10.4 deve valere "1" E M0.3 (il merker di passo del passo 3) deve essere "1".



Se questa condizione è soddisfatta, per es. è trascorso il tempo o è stato azionato il finecorsa, viene attivato un nuovo passo. Solitamente viene allora resettato un altro passo attivo.

Nel proseguimento della sequenza di passi non ci preoccupiamo ancora dell'attivazione delle uscite. Questa funzione viene svolta da un'altra parte di programma. Ciò significa che un comando basato su sequenze di passi è costituito da due parti di programma:

- 1) Il vero e proprio proseguimento dei singoli passi, in presenza delle condizioni necessarie (condizione di proseguimento).
- 2) L'attivazione delle uscite (pilotaggio valvole e motori).



Sequenza di passi

Comando a sequenza di passi

- Introduzione
- **Fondamenti**
- Uso delle sequenze di passi
- Modifica
- Descrizione della soluzione, esempio
- Test

Fondamenti (3)

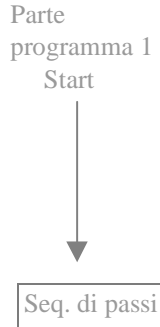
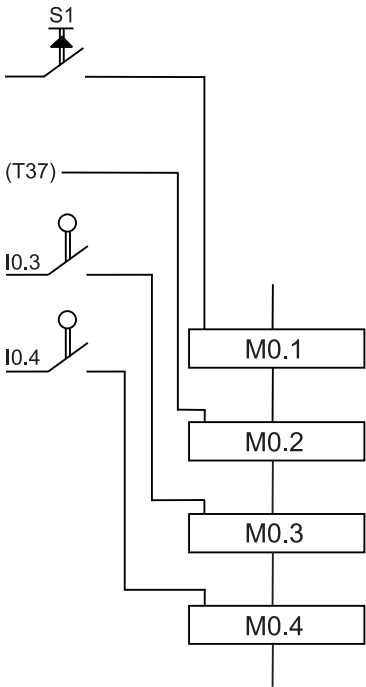
Le due parti di programma di un comando a sequenza di passi:

1) Le condizioni per l'attivazione dei singoli passi (job parziali) vengono combinate logicamente con i singoli merker di passo.

Se i merker M0.1... diventano attivi uno dopo l'altro, viene elaborata l'intera sequenza di passi.

Start S1 I0.1, ritardo di 3s, riscontro di profondità I0.3, posizione di base I0.4, passo rispettivamente precedente.

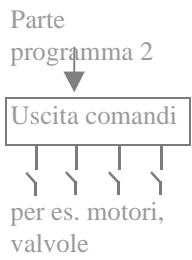
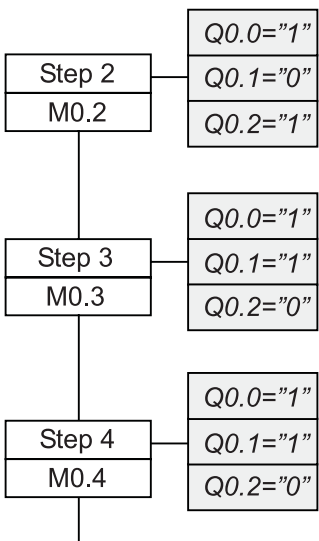
Merker di passo M0.1, M0.2, M0.3, M0.4



In questo modo è definita l'intera sequenza del job.

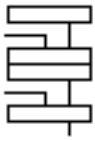
2) I merker attivi vengono assegnati alle uscite del PLC, che a loro volta comanda per es. dei relè o delle valvole.

Q0.1, Q0.2, Q0.0



Questa è l'interfaccia con l'impianto/macchina.

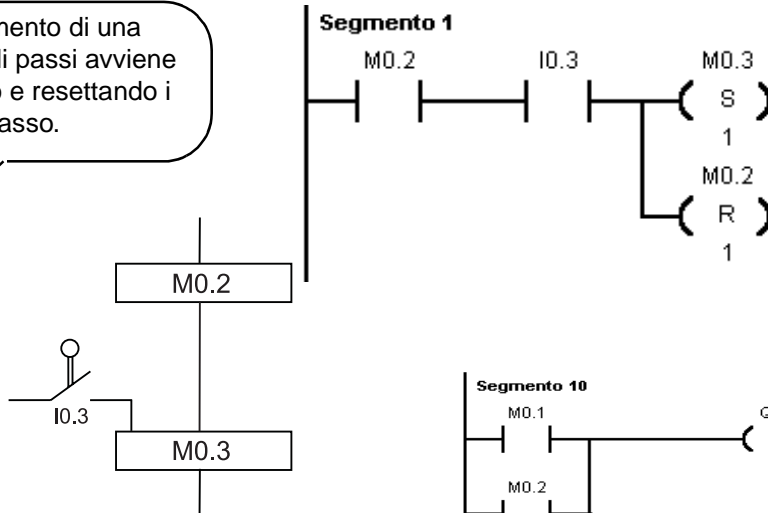
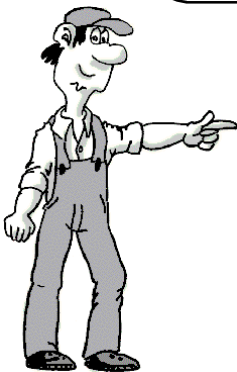
- Introduzione
- Fondamenti
- Uso delle sequenze di passi
- Modifica
- Descrizione della soluzione, esempio
- Test



Fondamenti (4)

1) Comando/proseguimento della sequenza di passi

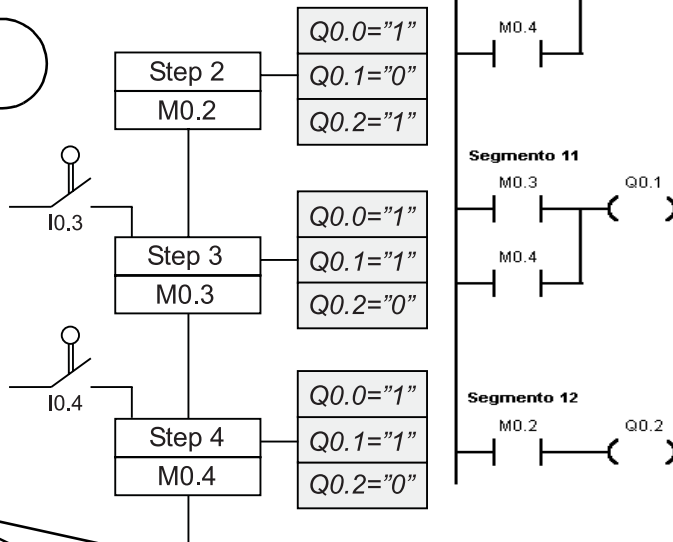
Il proseguimento di una sequenza di passi avviene impostando e resettando i merker di passo.



M0.2 e M0.3 sono merker di passo

2) Pilotaggio delle uscite tramite merker di passo

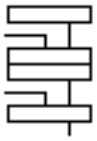
Se un'uscita è un passo "0", non viene pilotata.



Le uscite vengono pilotate solo dai merker di passo. L'assegnazione delle uscite con una bobina normale —()— garantisce che l'uscita stessa venga attivata solo nel rispettivo passo!

Se un'uscita deve valere "1" in diversi passi (per es. Q0.0), i rispettivi merker di passo vengono combinati logicamente in "OR" e assegnati all'uscita.

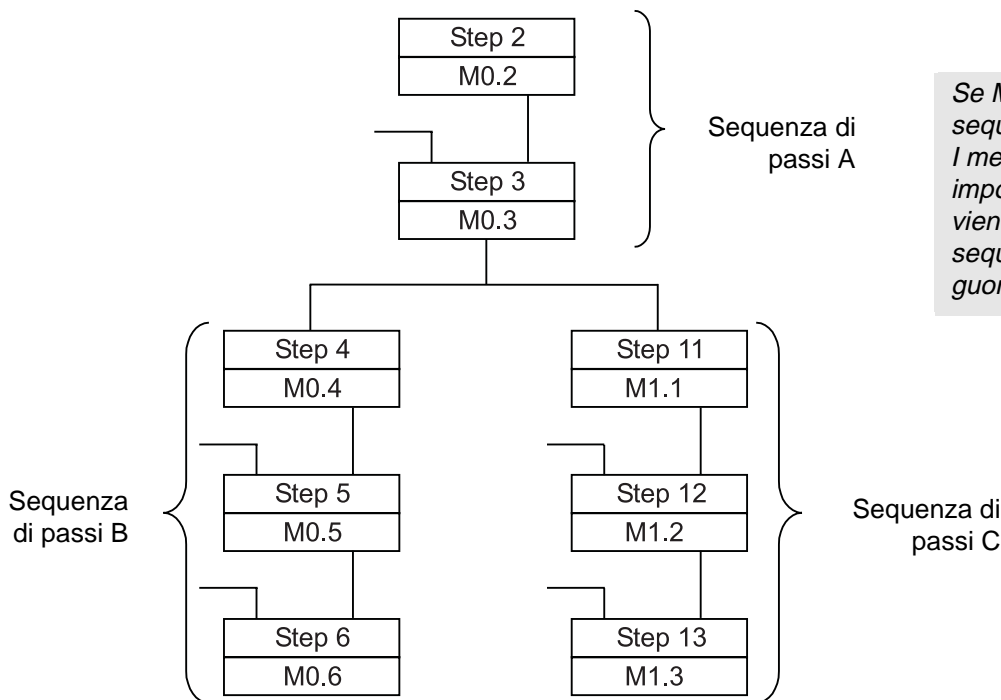
- Introduzione
- Fondamenti
- **Uso delle sequenze di passi**
- Modifica
- Descrizione della soluzione, esempio
- Test



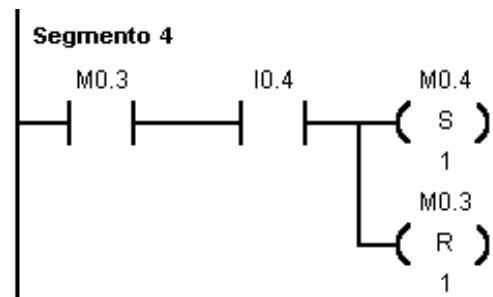
Uso delle sequenze di passi (1)

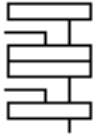


- Ad ogni passo è correlato un merker separato (merker di passo). Questo merker vale "1" quando il passo è attivo.
- Per motivi di chiarezza è bene che in una sequenza di passi vi sia sempre solo un passo attivo alla volta, ossia che un solo merker di passo alla volta sia "1".
- Se il compito da risolvere è più complesso, conviene utilizzare un'ulteriore sequenza di passi.
- Se si devono comandare due o più processi simultanei ma indipendenti, si utilizzano delle sequenze di passi separate. Questo viene illustrato nel grafico seguente.



Se M0.3="1", si avviano le due sequenze di passi B e C. I merker M0.4 e M1.1 vengono impostati da M0.3, dopodiché viene resettato M0.3 e le sequenze di passi B e C proseguono in modo indipendente.





Sequenza di
passi

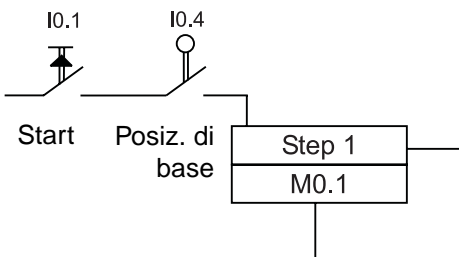
Comando a sequenza di passi

- Introduzione
- Fondamenti
- **Uso delle sequenze di passi**
- Modifica
- Descrizione della soluzione, esempio
- Test

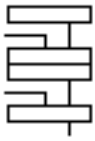
Uso delle sequenze di passi (2)

La condizione di proseguimento è composta nella pratica anche da più contatti.

Il nostro esempio si potrebbe ampliare in modo che lo start si verifichi solo quando la trapanatrice si trova nella posizione di base. La sequenza di passi si presenta allora così in questo punto:



- Introduzione
- Fondamenti
- **Uso delle sequenze di passi**
- Modifica
- Descrizione della soluzione, esempio
- Test

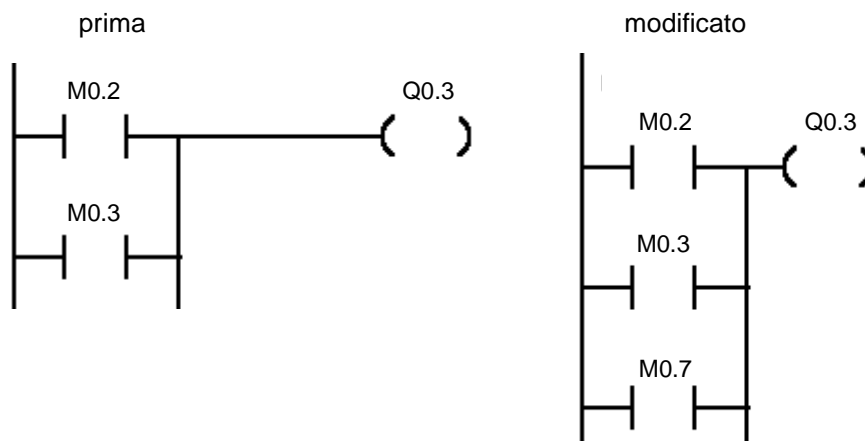


Uso delle sequenze di passi (3)



Vantaggi

- **La sezione di comando della sequenza di passi e il pilotaggio delle uscite sono separati**
 - Se un'uscita deve essere attiva anche nel passo 7 e non solo nei passi 2 e 3, è sufficiente modificare il programma in un solo punto.

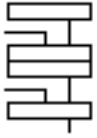


- Le modifiche nella sezione di comando della sequenza di passi non influiscono sul pilotaggio delle uscite.
- **Il programma si può testare facilmente**
 - Ogni passo si può seguire sul dispositivo di programmazione.
 - Se il ciclo non prosegue si può facilmente riconoscere la condizione mancante.
- **Minori tempi di inattività della macchina**
 - Se una macchina non funziona, dalla posizione della macchina e dal marker di passo attivo si può facilmente individuare la condizione di proseguimento mancante.
- **Meno errori di programmazione, messa in servizio più rapida**
 - L'uso della sequenza di passi implica una strutturazione del programma che riduce al minimo gli errori di programmazione.

Sequenza di passi

Comando a sequenza di passi

- Introduzione
- Fondamenti
- **Uso delle sequenze di passi**
- Modifica
- Descrizione della soluzione, esempio
- Test

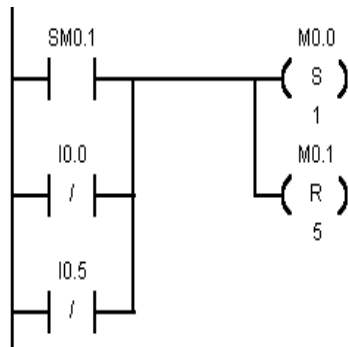


Informazioni importanti per la sicurezza (1)



Nel primo merker di passo (posizione di base) non devono essere attivati motori o valvole. Nel nostro esempio questo è il passo 0 o il merker di passo M0.0.

Se viene azionato "STOP" o se interviene una protezione motore, solo il primo merker di passo (nell'esempio M0.0) deve essere impostato e tutti i motori devono arrestarsi. Contemporaneamente tutti gli altri merker di passo devono essere resettati.



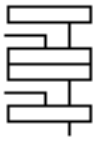
M0.0 viene impostato, da M0.1 a M0.5 vengono resettati

- nel primo ciclo dopo il ritorno della tensione tramite SM0.1 oppure
- se I0.0="0" o
- se I0.5="0".

SM0.1 fornisce lo stato "1" nel primo ciclo dopo la reinserzione per un ciclo

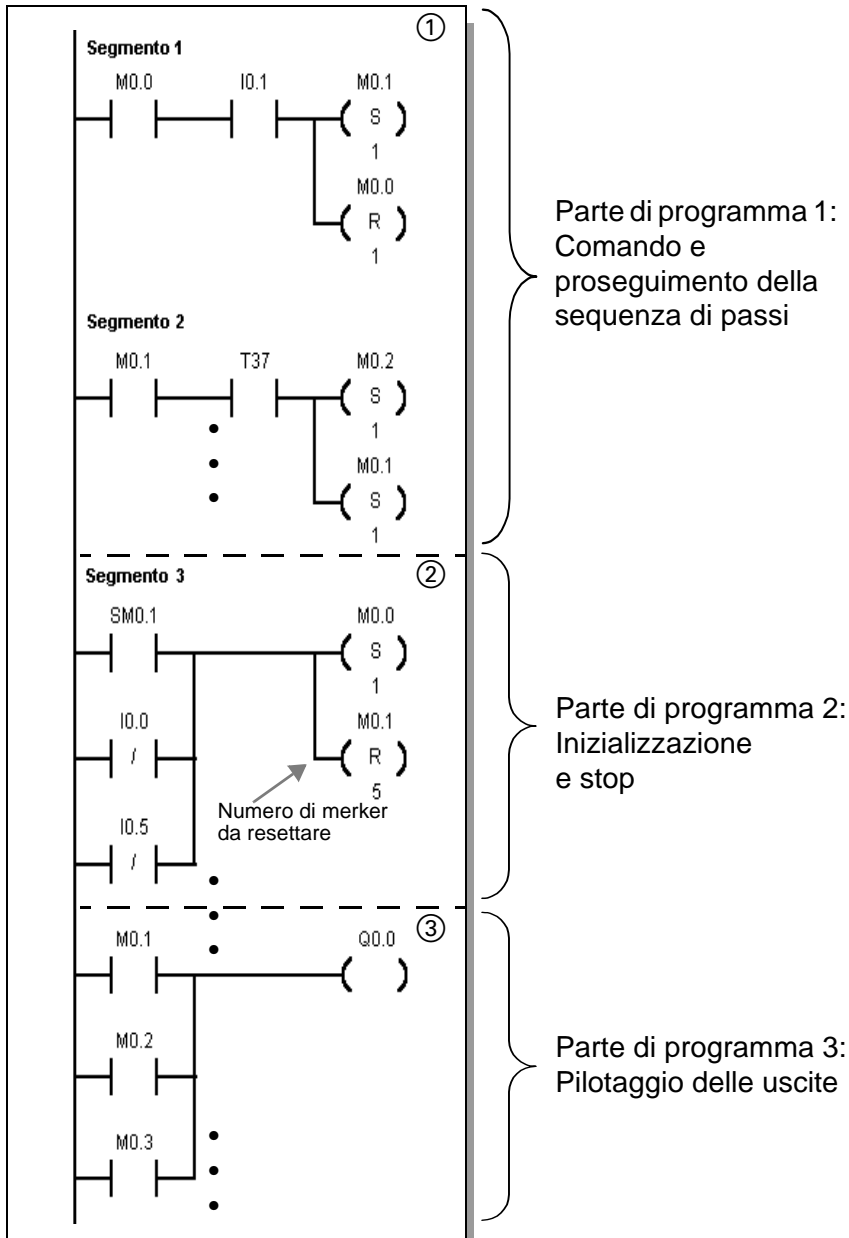
La parte di programma mostrata nell'esempio deve trovarsi alla fine delle "normali" condizioni di proseguimento della sequenza di passi. Questo garantisce che prima dell'attivazione delle uscite si verifichi una disinserzione eventualmente richiesta.

- Introduzione
- Fondamenti
- **Uso delle sequenze di passi**
- Modifica
- Descrizione della soluzione, esempio
- Test



Informazioni importanti per la sicurezza (2)

La struttura di un programma a sequenza di passi dovrebbe presentarsi così:

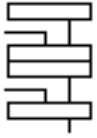


L'assegnazione della prima uscita ③ deve essere preceduta dalla parte di programma per l'attivazione della posizione di base ②. Ciò garantisce che l'attivazione della posizione di base abbia la priorità più elevata.

Sequenza di passi

Comando a sequenza di passi

- Introduzione
- Fondamenti
- Uso delle sequenze di passi
- **Modifica**
- Descrizione della soluzione, esempio
- Test



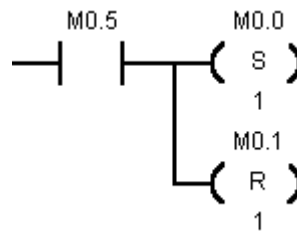
Modifica



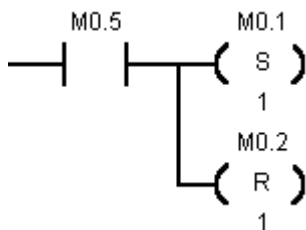
Il segmento 6 determina a quale passo salta il programma dopo il passo 5. In questo esempio salta al passo 0.

Questo processo viene comandato da:

Impostazione di M0.0 e resettaggio di M0.1 ... M0.5.

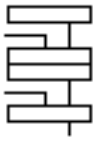


Se dopo il passo 5 il programma deve saltare al passo 1, il segmento 6 deve presentarsi così.



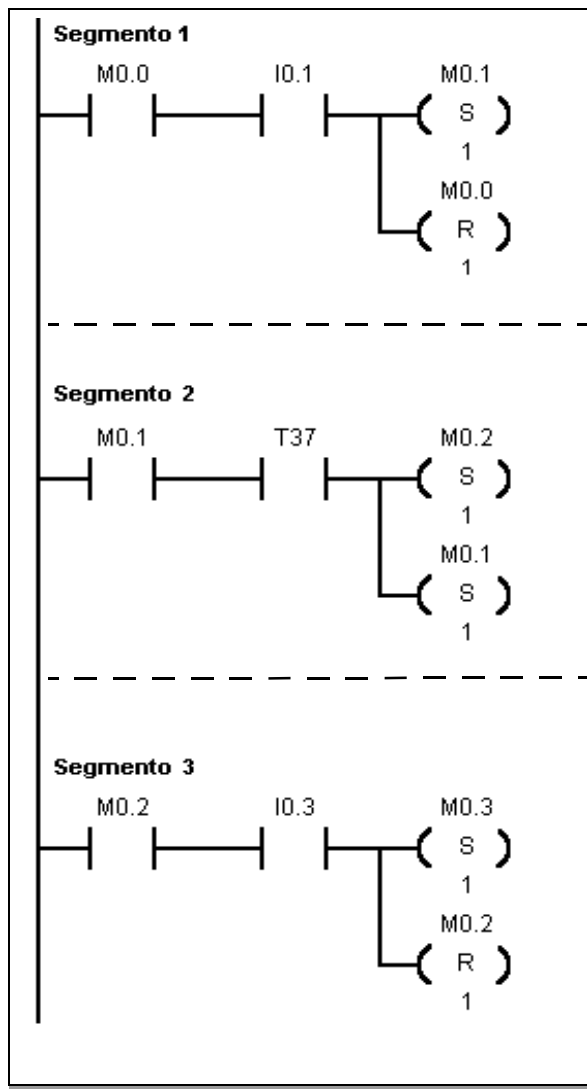
Con questa modifica la "trapanatrice automatizzata" funziona automaticamente finché non viene arrestata da I0.0 o I0.5.

- Introduzione
- Fondamenti
- Uso delle sequenze di passi
- Modifica
- Descrizione della soluzione, esempio
- Test



Descrizione della soluzione, esempio (1)

Parte di programma 1 - proseguimento della sequenza di passi



Attivazione del passo 1

Il merker di passo M0.1 viene impostato quando la sequenza di passi si trova nella posizione di base (M0.0 = "1") E viene azionato IO.1. Contemporaneamente viene resettato M0.0, il merker di passo della posizione di base.

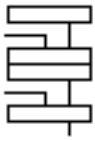
Attivazione del passo 2

Il merker di passo M0.2 viene impostato quando la sequenza di passi si trova nel passo 1 (M0.1 = "1") E il temporizzatore T37 è scaduto. Contemporaneamente viene resettato il merker di passo M0.1.

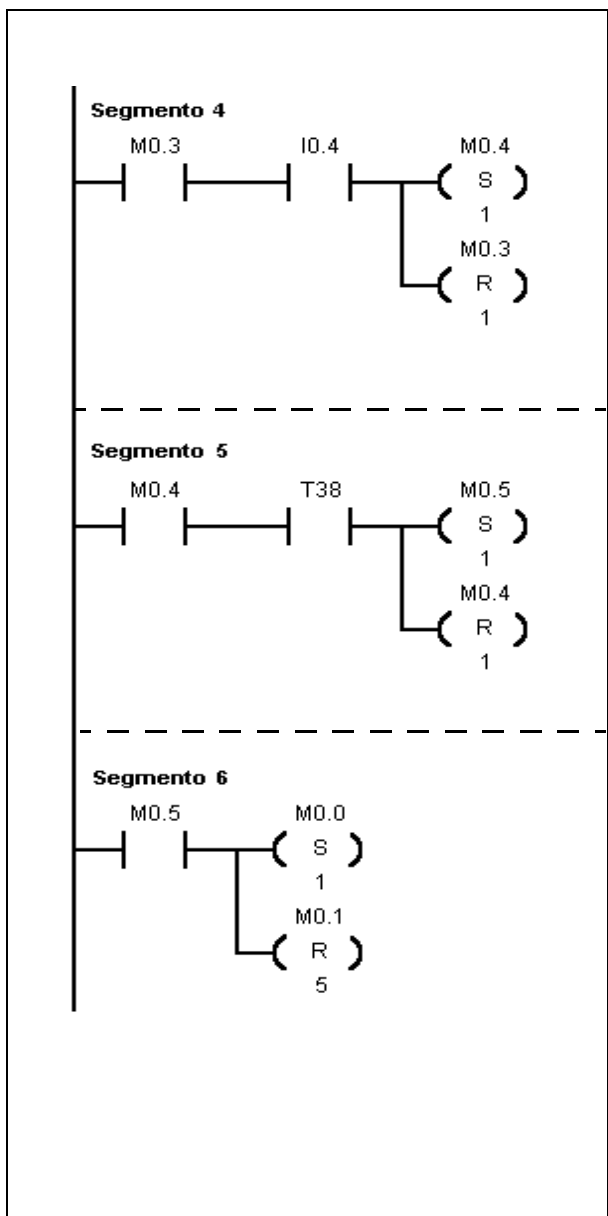
Attivazione del passo 3

Il merker di passo M0.3 viene impostato quando la sequenza di passi si trova nel passo 2 (M0.2 = "1") E l'ingresso IO.3 del riscontro di profondità vale "1". Contemporaneamente viene resettato M0.2.

- Introduzione
- Fondamenti
- Uso delle sequenze di passi
- Modifica
- Descrizione della soluzione, esempio
- Test



Descrizione della soluzione, esempio (2)



Attivazione del passo 4

Il merker di passo M0.4 viene impostato quando la sequenza di passi si trova nel passo 3 (M0.3 = "1") E l'ingresso I0.4 (posizione di base) vale "1". Contemporaneamente viene resettato M0.3.

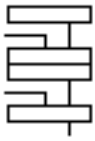
Attivazione del passo 5

Il merker di passo M0.5 viene impostato quando la sequenza di passi si trova nel passo 4 (M0.4 = "1") E il temporizzatore T38 è scaduto. Contemporaneamente viene resettato M0.4.

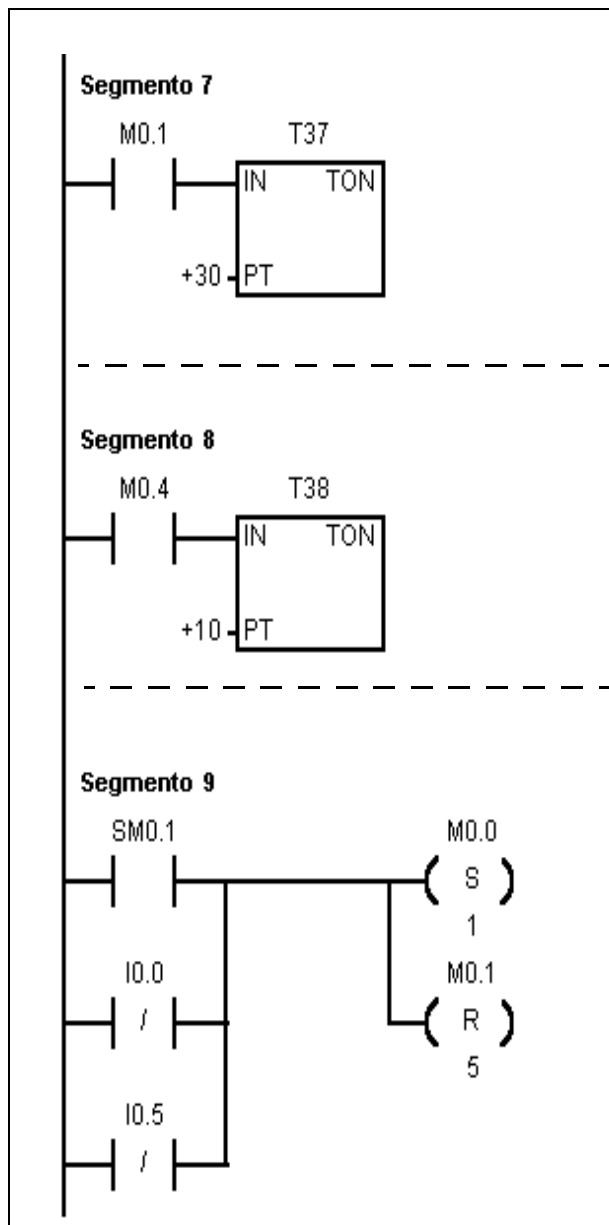
Attivazione del passo 0

Quando è attivo il merker di passo M0.5 (il tempo di inseguimento T38 è trascorso), la sequenza di passi attiva il passo 0 (passo di inizializzazione). Questo passo è stato deliberatamente inserito nel segmento 6 perché in questo punto possono essere richieste anche altre condizioni (come l'estrazione del pezzo lavorato), prima che diventi nuovamente attivo il passo 0. Questa condizione dovrebbe essere collegata in parallelo al contatto M0.5.

- Introduzione
- Fondamenti
- Uso delle sequenze di passi
- Modifica
- Descrizione della soluzione, esempio
- Test



Descrizione della soluzione, esempio (3)



Attivazione del temporizzatore T37

Il temporizzatore T37 viene avviato quando è attivo il passo 1 (M0.1 = "1").

Attivazione del temporizzatore T38

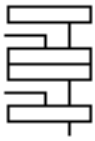
Il temporizzatore T38 viene avviato quando è attivo il passo 4 (M0.4 = "1").

Inizializzazione della sequenza di passi

Il merker di passo M0.0 viene impostato

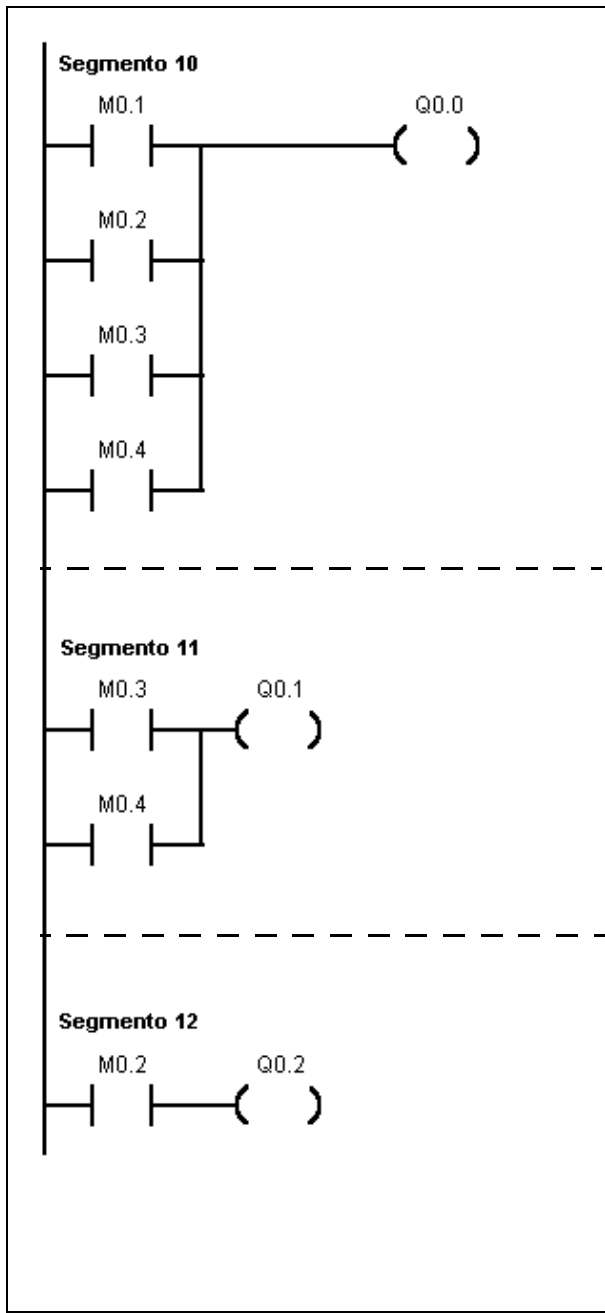
- 1) Nel primo ciclo (SM0.1 vale qui "1" per un ciclo)
OPPURE
- 2) Quando viene azionato Stop (I0.0 = "0")
OPPURE
- 3) Quando interviene la protezione motore (I0.5 = "0").
Contemporaneamente vengono resettati i merker di passo da M0.1 a M0.5.

- Introduzione
- Fondamenti
- Uso delle sequenze di passi
- Modifica
- Descrizione della soluzione, esempio
- Test



Descrizione della soluzione, esempio (4)

Parte di programma 2 - Pilotaggio delle uscite



Attivazione dell'uscita Q0.0

(rotazione destrorsa)

L'uscita Q0.0 vale "1" nei passi 1,2,3,4, ossia quando M0.1 o M0.2 o M0.3 o M0.4 valgono "1".

Attivazione dell'uscita Q0.1

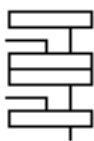
(inversione di direzione)

L'uscita Q0.1 è "1" nei passi 3 e 4, ossia quando M0.3 o M0.4 valgono "1".

Attivazione dell'uscita Q0.2

(Avanzamento on)

Se il marker M0.2 = "1" l'uscita Q0.2 = "1".



Test

Sequenza di
passi

Comando a sequenza di passi

- Introduzione
- Fondamenti
- Uso delle sequenze di passi
- Modifica
- Descrizione della soluzione, esempio
- **Test**

Potete introdurre voi stessi il programma o caricare da dischetto il file "**d05.mwp**".
Accertatevi che l'interruttore di stop sia I0.0 e la protezione motore sia I0.5 "Contatti normalmente chiusi". Questa condizione è imposta da motivi di sicurezza.

La rottura del cavo tra l'interruttore e il PLC arresta la macchina!

I0.5 e I0.0 devono valere "1" per il test, ossia i LED di ingresso devono essere accesi.

Azionando brevemente I0.1 si avvia il motore. Dopo 3s si attiva l'avanzamento Q0.2.
Dopo aver azionato I0.3 il motore inverte il senso di rotazione e l'avanzamento Q0.2 si arresta.

Una volta raggiunta la posizione di base (breve azionamento di I0.4), il motore si arresta dopo 1s.

I0.0 e I0.5 fermano il motore in ogni fase.

Osservate il programma nel funzionamento di test. Vedrete sempre con precisione quale ingresso è necessario affinché la sequenza di passi possa proseguire.

Provatelo voi stessi !



Ripetizione

Autoritenuta

Circuito a
impulso

Ritardo alla
disinserzione

Sequenza di
passi

Appendice

55



Sequenza di passi **Nota**



Fatto.

Adesso potete risolvere da soli i compiti che vi aspettano con il vostro S7-200. Se dovete convertire dei circuiti di contattori più complessi, troverete degli utili consigli nell'appendice.

Ripetizione

Autoritenuta

Circuito a impulso

Ritardo alla disinserzione

Sequenza di passi

Appendice

57

Ripetizione



Autoritenuta



Circuito a impulso



Ritardo alla disinserzione



Sequenza di passi



Appendice

Volete saperne di più?

Per chi vuole saperne di più, nella directory "**Samples**" in STEP 7-Micro/WIN o nei "Tips & Tricks" dell'S7-200 sono contenuti ulteriori esempi. "Tips & Tricks" può essere richiesto ai partner SIMATIC della filiale di zona.

È inoltre disponibile la documentazione relativa all'S7-200, nonché una serie di corsi organizzati dalla Scuola Automazione della Siemens.

Rivolgetevi al partner
SIMATIC.
Sarà lieto di rispondere
alle vostre domande.



Per informazioni rivolgersi al partner SIMATIC presso il quale si è acquistato il pacchetto per il neo-utente.

È inoltre disponibile una Hotline presso la Siemens di Norimberga, tel.: 0049/911/895-7000.

Ripetizione



Autoritenuta



Circuito a impulso



Ritardo alla disinserzione



Sequenza di passi



Appendice

Qui di seguito troverete alcuni esempi che vi aiuteranno a realizzare facilmente anche delle "combinazioni di interruttori" complesse nello schema a contatti.



Ripetizione



Autoritenuta



Circuito a impulso



Ritardo alla disinserzione



Sequenza di passi



Appendice

61

Ripetizione



Autoritenuta



Circuito a impulso



Ritardo alla disinserzione



Sequenza di passi



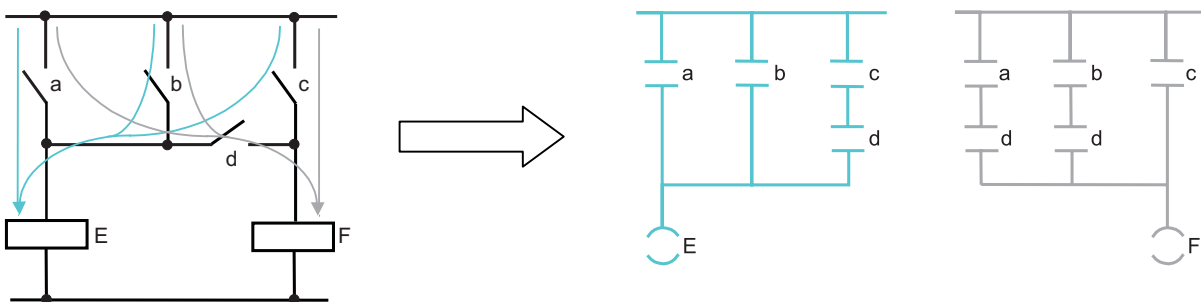
Appendice



Collegamento a ponte

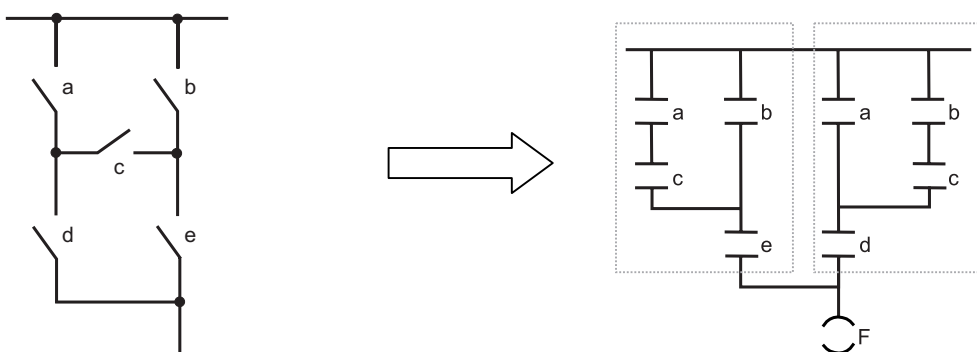
Chi passa dalla tecnica dei contattori a quella dei PLC incontra probabilmente delle combinazioni di interruttori che non si possono direttamente convertire nello schema a contatti. E' il caso, per esempio, del collegamento a ponte. Qui di seguito viene proposta una soluzione per il collegamento a ponte semplice e per quello complesso.

1) Collegamento a ponte semplice



Il collegamento a ponte semplice (a sinistra) viene sostituito da due segmenti. I singoli possibili percorsi della corrente vengono semplicemente suddivisi. Per facilitare il confronto, anche lo schema a contatti è stato riprodotto in verticale.

2) Collegamento a ponte complesso



I due percorsi possibili della corrente sono stati qui nuovamente convertiti e ricombinati. Da un lato a,c con b in parallelo, dall'altro b,c con a in parallelo. Per facilitare il confronto, lo schema a contatti è stato riprodotto in verticale.

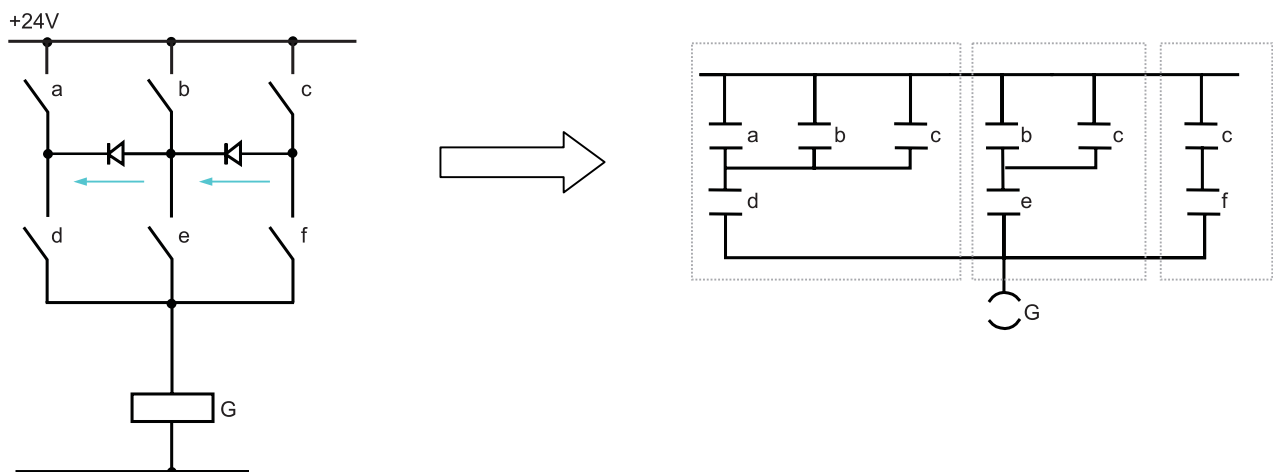
Cercate di evitare nei nuovi progetti il collegamento a ponte nel circuito! Pensate direttamente "in termini di schema a contatti".



Circuito a diodi

Se nei "vecchi" schemi elettrici sono stati utilizzati dei diodi, non è possibile convertirli direttamente nello schema a contatti.

Siccome i diodi rappresentano in principio linee di collegamento, che lasciano tuttavia passare la corrente in un solo senso, la soluzione qui proposta è simile a quella del collegamento a ponte. Lo schema a contatti è riprodotto in verticale per agevolare il confronto con lo schema elettrico.



In questo circuito i percorsi della corrente possibili sono tre: per l'interruttore d, per l'interruttore e ed infine per l'interruttore f.

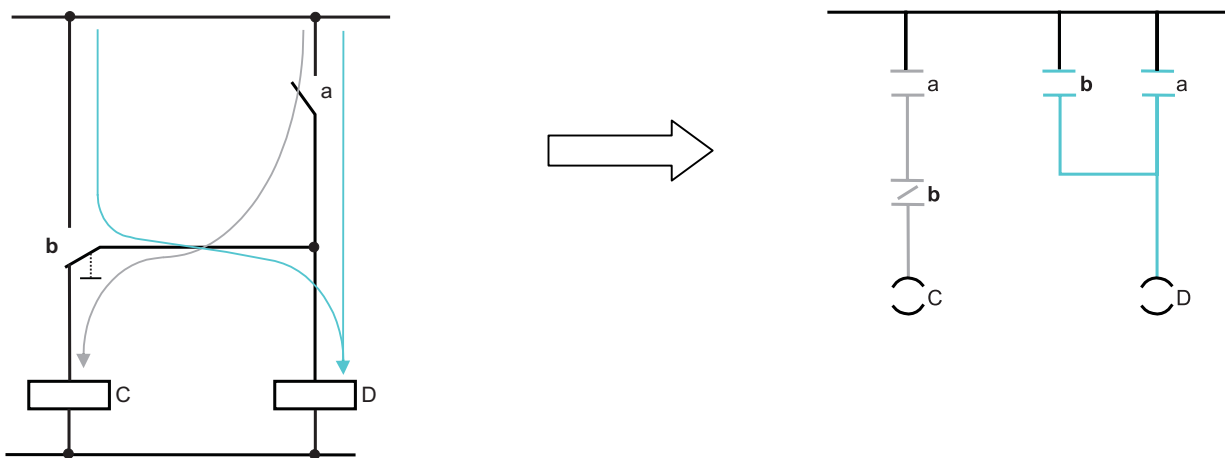
La corrente può passare nei diodi solo nella direzione da b verso d oppure da c verso e.

Questi tre percorsi determinano anche i tre segmenti parziali che nella soluzione dello schema a contatti sono circondati da una linea tratteggiata. Siccome gli interruttori d, e ed f si trovano sulla stessa barra dell'uscita G, anche questi tre segmenti parziali sono collegati in un unico segmento.



Commutatori

Neanche i commutatori dovrebbero rappresentare un problema quando si converte uno schema elettrico in uno schema a contatti. Qui di seguito viene brevemente descritta questa conversione.



Per maggiore chiarezza il percorso della corrente è stato evidenziato graficamente.

Il commutatore b viene quindi suddiviso in un contatto normalmente chiuso che funziona in serie con a ed ha effetto all'uscita C, e in un contatto normalmente aperto che funziona in parallelo con a e commuta D.

In questo modo si può convertire nello schema a contatti un commutatore tramite un contatto normalmente chiuso e un contatto normalmente aperto con lo stesso indirizzo di ingresso.

Appendice

Suggerimento
Appunti



Appunti

A4

Ripetizione



Autoritenuta



Circuito a impulso



Ritardo di disinserzione



Sequenza di passi



Appendice



Indice A...D

In questo indice sono riportati i termini chiave principali per la programmazione dell'S7-200, con una breve spiegazione delle sigle utilizzate nell'opuscolo e riferimenti incrociati all'opuscolo "Un'ora per conoscerlo".

Nell'indice viene utilizzato il simbolo seguente:

1h- Rimandi alle pagine dell'opuscolo "Un'ora per conoscerlo"

A

Aspetti di sicurezza: 19
 Autoritenuta: 13 ss.
 AW: parola d'uscita (16 bit), per es. AW0
 AWL: lista istruzioni (in inglese STL)

B

Binario: tipo di rappresentazione numerica in bit (due valori possibili, 0 e 1)
 Bit: cifra binaria: 6
 Bit di temporizzazione: 7
 Blocco dati: memoria delle variabili dell'S7-200, incui salvare i valori che devono essere utilizzati nel programma di comando
 Blocco organizzativo: contiene il programma utente del PLC che viene elaborato su base ciclica.
 Bobina: rappresentazione per un elemento di uscita nello schema a contatti (simile a un relè): 17
 B&B: servizio e supervisione, per es. con Text Display, Operator Panel e Touch Panel; oggi HMI
 Byte: valore di ampiezza 8 bit: 1h- 48

C

Cancellazione di elementi: 1h- 32
 Ciclo di un PLC (3...10ms): 9 ss.
 Circuito a diodi: A2
 Circuito a impulso di corrente: 21
 ss.Collegamento a ponte: A1
 Comando sequenziale: controllore che esegue o trasmette dei passi in base agli eventi. Questi passi attivano a loro volta delle azioni perdefinite.
 Combinazione logica in AND: 6
 Commenti, inserimento: 36 ss.
 Condizione di smistamento: 40
 Contatto normalmente aperto: 8
 Contatto normalmente chiuso: 14, 15
 Controllore a logica combinatoria: controllore che commuta delle azioni in base allo stato degli ingressi e delle uscite.
 Controllore semiautomatico: controllore che esegue autonomamente determinate sequenze, ma che in altre occasioni dipende dai dati introdotti dell'utente.
 CPU: Central Processing Unit, unità centrale di elaborazione, p. es. S7-200

D

DB1: blocco dati dell'S7-200
 DIV: calcolo della divisione



Indice E...S

E

END: istruzione di fine programma 31

F

Falso, vero: 6
Flusso di corrente nello schema a contatti: 7
Fondamenti della sequenza di passi: 39-42
Fronti: 21,22

G

Guida online: 8

H

HMI: Human-Machine-Interface (servizio e supervisione)

I

I: ingresso, per es. I0.0
IB: byte d'ingresso (8 bit), per es. IB0
Immagine di processo: il programma di un PLC lavora sull'immagine della periferia. A inizio ciclo viene letta l'immagine degli ingressi, a fine ciclo viene inviata alla periferia l'immagine delle uscite: 9 ss.
Imposta, Resetta: 17 ss.
Inserimento di elementi: 1h-📖 30
IPI: immagine di processo degli ingressi: 9
IPU: immagine di processo delle uscite: 10
IW: parola d'ingresso (16 bit), per es. IW0

M

MB: byte di merker (8 bit)
MD: parola doppia di merker (32 bit)
Memoria V: blocco dati dell'S7-200
Memorizzazione del programma: 1h-📖 41
Merker: 25 ss.
Merker di passo: 41
Modello: 1h-📖 7
MW: parola di merker (16 bit)

O

OB1: blocco organizzativo dell'S7-200
OB: byte di uscita (8 bit), per es. AB0
Q: uscita, per es. Q0.0

P

Parola: un valore rappresentato da 2 byte (16 bit).
PLC: controll. logico programmabile: 1h-📖 5
PLC: programmable logic controller, abbreviazione inglese per controllore logico programmabile.

R

RET: Return, termina il sottoprogramma
Resetta, Imposta: 16 ss.
Rimanenza: 23
Ritardo alla disinserzione: 29 ss.
Ritardo all'inserzione: 1h-📖 35
RUN: posizione del selettore dei modi operativi dell'S7-200 per l'avvio o il riavvio manuale del controllore

S

SBR: SubRoutine, sottoprogramma
Schema a contatti: 1h-📖 25
Segmento, inserimento: 32
Selettore dei modi operativi: selettore dell'S7-200 con tre posizioni: STOP, TERM, RUN.
Sequenza di passi: catena di passi in sé conclusa che viene elaborata per gradi nel comando sequenziale: 39 ss.
Simboli dell'opuscolo: 3
SMB: byte di merker speciale (8 bit), per es. SMB28
SMB28: potenziometro dell'S7-200
SMW: parola di merker speciale (16 bit)
SMD: doppia parola di merker speciale (32 bit)
Soluzione del ritardo all'inserzione: 29 ss.



Indice S...Z

S (seguito)

Soluzione dell'autoritenuta: 15 ss.
 Soluzione della catena di passi: 39 ss.
 Soluzione del circuito a impulso: 21 ss.
 Stato: permette di osservare un processo a livello di programma o in una tabella di stato separata. Utile per test e diagnosi.
 Stato KOP: 7, 1h-📖 26
 Stato nello schema a contatti: 1h-📖 26
 STOP: posizione del selettore dei modi operativi dell'S7-200 per l'arresto manuale del controllore.

T

T37 (Timer): 29 ss.
 Temporizzatore: tipo di dati per tutte le funzioni di temporizzazione (qui chiamato anche "Timer").
 TERM: posizione del selettore dei modi operativi dell'S7-200. Permette di influire sul PLC da STEP 7-Micro/WIN
 Timer: espressione inglese per temporizzatore.
 TON: interruttore a tempo dell'S7-200, detto anche timer: 1h-📖 36 s.
 TONR: ritardo alla disinserzione con memoria

U

Uso delle sequenze di passi: 45 ss.

V

V: bit di variabile, per es. V0.0
 VB: byte di variabile, per es. VB0
 VD: parola doppia di variabile, per es. VD45
 Vero, falso: 6
 VW: parola di variabile, per es. VW45

X

XOR: OR esclusivo, combinazione logica, interviene solo quando sull'ingresso gli stati sono diversi (antivalenze).

Z

Z0: contatore semplice (CTU).

Ad
Siemens AG
A&D AS MVM
Gleiwitzer Str. 555

Fax: 0049 911 895-2786

D-90475 Nuernberg

"Due ore per conoscerlo"

Egregio utente del micro PLC S7-200,

questo breve manuale è stato ideato per consentire un rapido approccio con il micro PLC S7-200 ai neo utenti.

Siamo convinti che in questo modo i primi passi nel mondo dell'S7-200 saranno per voi "una piacevole passeggiata".

Per poter offrire ai nostri clienti un livello qualitativo sempre maggiore chiediamo ora la vostra gentile collaborazione! Esprimete un giudizio sul presente manuale e non esitate a farci avere proposte e suggerimenti. Ne terremo conto nella successiva edizione!

Grazie

A&D AS MVM

Proposte di miglioramento, consigli, suggerimenti

Mittente

Nome _____ Funzione _____

Ditta _____ Telefono _____

Via _____ Località _____

I miei suggerimenti:
