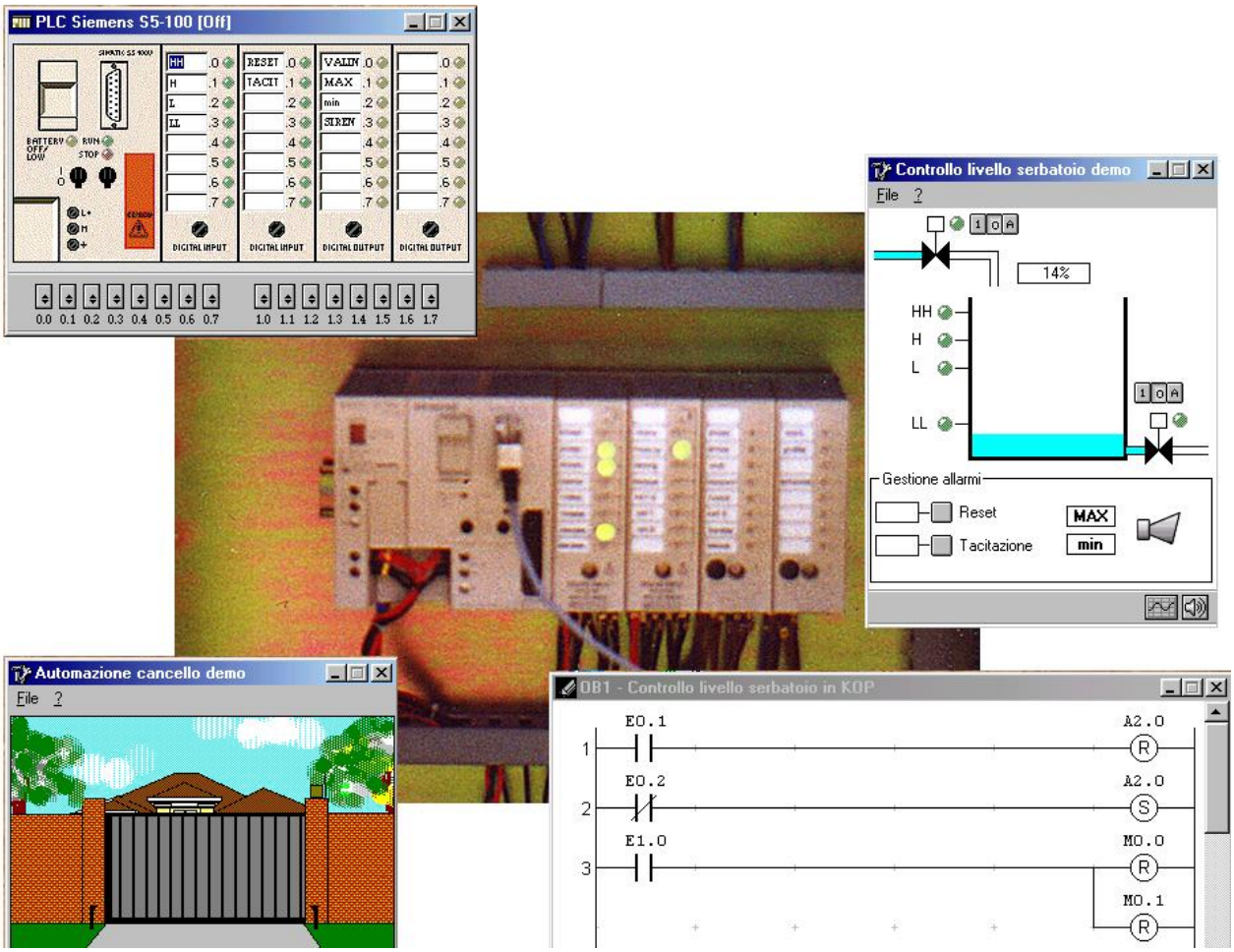
Politecnico di Bari

corso di: CONTROLLO DEI PROCESSI
del Prof. Ing. A. Magnanimo

Applicazione del PLC:

La confezionatrice teli



a cura di
Nardelli Graziano
Di Noi Andrea
Recchia Anna Rita

Il controllore logico programmabile

IL controllore logico programmabile: PLC, è il più economico microcomputer concepito per funzionare come sistema di comando per automatismi industriali.
Attraverso il modulo di input il PLC riceve dai sensori, segnali elettrici, li elabora eseguendo il programma memorizzato e attraverso un opportuno modulo di output emette segnali elettrici verso attuatori.
Per eseguire quanto più possibile in tempo reale l'evoluzione del processo controllato, l'esecuzione del processo è ciclica: un registro puntatore indicizza sequenzialmente le istruzione del programma fino all'ultima, per poi iniziare di nuovo dalla prima.
Nei casi più semplici, un PLC può limitarsi ad eseguire funzioni logiche, di temporizzazione e di conteggio ma può eseguire anche operazioni aritmetiche, confronti, regolazione, inoltre presenta al suo interno moduli PID che possono essere programmati nell'ipotesi di dover regolare un sistema in anello chiuso.

Il PLC in commercio si differenziano essenzialmente per il tipo di gestione delle informazioni di input e di output.
Nella gestione asincrona, un ingresso viene letto sul modulo di input all'atto della sua interrogazione dal programma e l'uscita a cui fa capo viene aggiornata in corrispondenza alla relativa istruzione di output. Questo metodo avvicina il PLC al funzionamento in tempo reale tipico della logica cablata.
(questo tipo di PLC è quello che viene utilizzato nella maggior parte dei casi)

Nella gestione sincrona il PLC lavora nel seguente modo:

- All'inizio di un ciclo di scansione la CPU legge tutti gli ingressi e li memorizza in una zona della memoria centrale; Tale immagine degli ingressi rimane inalterata per tutto il tempo di esecuzione del programma.
- Il microprocessore esegue le istruzioni uno dopo l'altra e se deve essere attivata un'uscita, il suo stato viene memorizzato in un'altra apposita zona della memoria centrale, che al termine conterrà la così detta immagine delle uscite .
- Al termine dell'ultima istruzione del programma, il microprocessore rende attiva l'immagine di tutte le uscite (sincronicamente) inviandone lo stato secondo la propria sequenza, al modulo di uscita.
- A questo punto vengono nuovamente letti gli ingressi , ed inizia un nuovo ciclo di scansione del programma.

Con la struttura sincrona il processo risulta più lento rispetto a quella asincrona ma nello stesso tempo nel momento in cui si presentono uscite contraddittorie, resta operativo lo stato delle uscite determinato dall'ultima istruzione di output relativa a quell'uscita (è quindi difficile che si presentono oscillazioni).

Logica cablata e logica programmata

La differenza fondamentale tra le due, è che quella programmata ha il vantaggio della flessibilità operativa: la modifica delle funzioni comando-controllo si attua semplicemente cambiando le istruzione del programma, a differenza per la logica cablata sarebbe necessario effettuare il ricablaggio.

Inoltre usando il PLC è possibile effettuare una simulazione simulando ingressi fittizi e utilizzando come uscite dei diodi led (tali diodi led si presentono gia sul PLC in corrispondenza di ogni uscita)

IL vantaggio della logica cablata è quello di essere più veloce in quanto esegue le operazioni in parallelo cioè se si verifica un cambiamento dello stato degli ingressi tale da attivare un uscita, questa viene immediatamente attivata.

Per il PLC invece è necessario un tempo detto tempo di ciclo cioè il tempo necessario per effettuare un esecuzione completa del programma. Il tempo di ciclo dipende generalmente dal tipo e dalla complessità delle istruzioni e dalla velocità del microprocessore (clock della macchina). Generalmente per i PLC più usati il tempo di ciclo si aggira sull'ordine delle decine di millisecondi.
In altre parole, il tempo di ciclo e talmente breve da dare un impressione di un funzionamento del tutto analogo a quello di una logica cablata.

Memoria di un PLC

Le memorie di un PLC possono essere divise in due aree:

- Memoria di sistema (ROM): destinata ad ospitare il sistema operativo e i dati di cui questo si serve
- Memoria applicativa (RAM): tale memoria a sua volta si divide in due aree
    - dati di tipo flag (o singolo bit):il risultato delle operazioni eseguite dal programma possono essere memorizzate in questi flag che hanno lo scopo di segnalare alcune situazioni di funzionamento del processo.
    - dati di tipo registro (a più bit): in questi tipi di registri possono essere memorizzati dati numerici, risultati di istruzioni aritmetiche, valori di temporizzazioni o di conteggio ecc…

Mentre un normale calcolatore registra i dati allocandoli dinamicamente (nelle zone che sono libere di volta in volta), il PLC segue una locazione rigida della memoria. Così ad esempio un certo terminale di ingresso corrisponderà sempre ad una precisa locazione di memoria.
Le zone per i dati di tipo flag è la più estesa dato che il PLC elabora prevalentemente informazioni di tipo ON-OFF. La zona per i dati di tipo registro è più estesa nei PLC il cui set prevede istruzioni aritmetiche e di controllo _ regolazione di processo.
I dati di tipo registro possono essere a 16 bit o 32 bit a seconda del tipo di PLC.

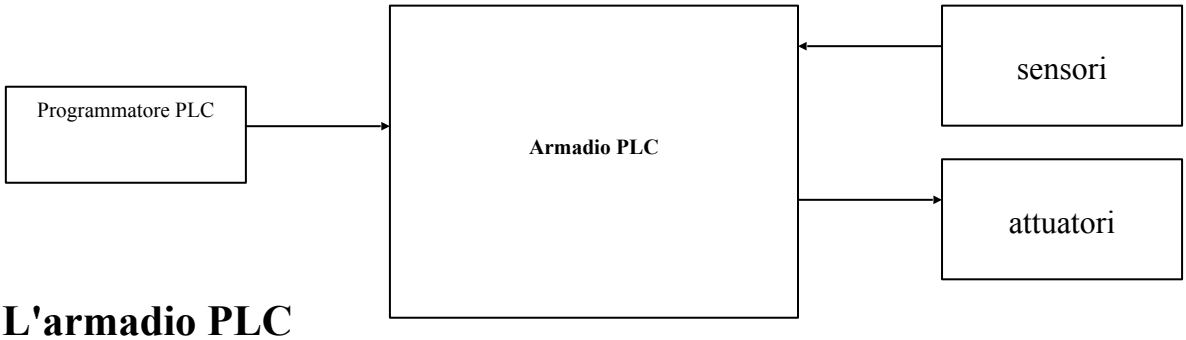
Fondamentali Parametri di un PLC

I fondamentali parametri di un PLC hardware, possono essere così riassunti:

- numero di ingressi collegabili
- numero di uscite pilotabili
- numero di temporizzatori ed intervalli di tempo che possono essere utilizzati in fase do programmazione
- numero di contatori e di unita che possono essere conteggiati
- tempo di risposta degli ingressi
- protezione in caso di mancanza di alimentazione
- caratteristiche della memoria
- caratteristiche del microprocessore: tecnologia utilizzata, lunghezza delle parole trattate
- tempo di ciclo

Struttura di un PLC

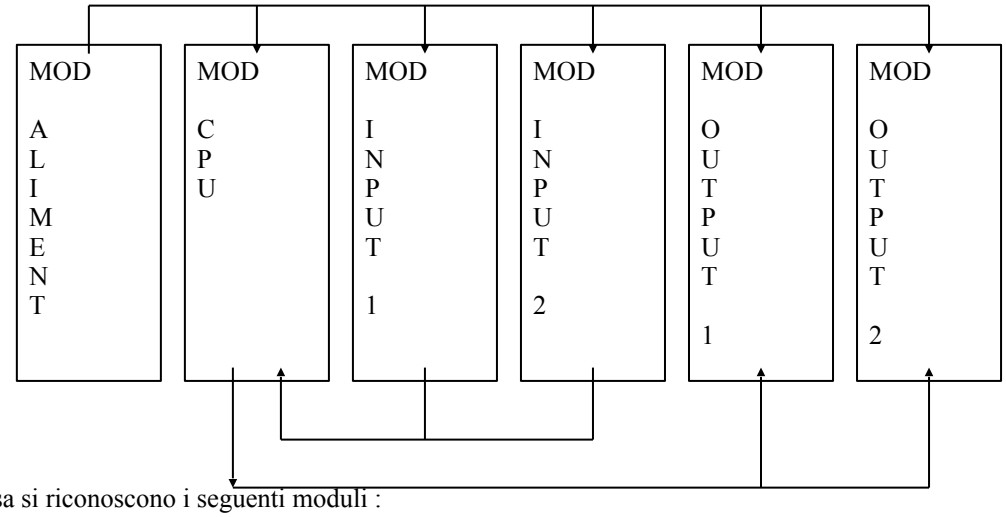
La struttura di un PLC è schematizzata nel seguente disegno:



L'armadio PLC

L'armadio del PLC contiene e racchiude tutti gli altri moduli assicurandone la connessione meccanica e il collegamento elettrico. Ha in genere la forma di un parallelepipedo aperto su di un lato per permettere l'inserimento dei moduli. Il collegamento elettrico tra i moduli è realizzato da una piastra a circuito stampato sul quale sono montati dei connettori. L'armadio è di metallo e deve essere connesso elettricamente a terra, sia per ragioni di sicurezza, sia per meglio schermare i moduli alloggiati.

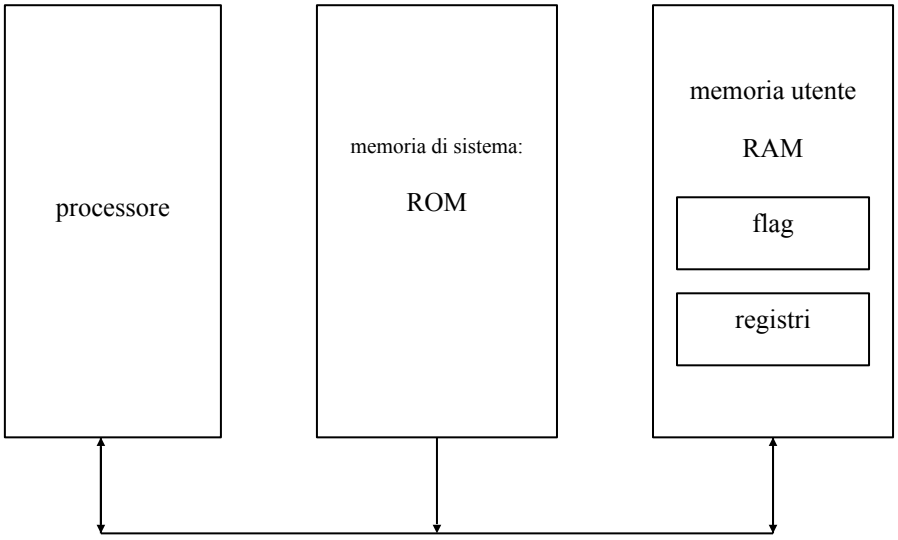
Riportiamo adesso una tipica connessioni dei moduli che si trovano all'interno dell'armadio:



In essa si riconoscono i seguenti moduli :

- il modulo di alimentatore: esso fornisce l'alimentazione elettrica a tutti gli altri moduli ed alla strumentazione del campo. Le tensioni tipiche sono 24V CC, 120V CA, 220V CA
- I moduli di ingresso e di uscita svolgono il ruolo di interfaccia fra i dispositivi di campo ed il modulo processore. Essi possono essere:
    - digitali
    - analogici (in tal caso per inviare i dati tra i moduli di ingresso e di uscita e tra la CPU occorre intramettere rispettivamente dei dei convertitori analogici digitali e dei convertitori digitali analogici)

- Il modulo CPU è il cuore del PLC. In tale CPU è presente la memoria le cui caratteristiche sono state descritte precedentemente ed è inoltre presente il processore che esegue tutti i calcoli descritti del programma scritto dall'utente. La schematizzazione interna è la seguente:



CLASSIFICAZIONE DEI PLC

I PLC in commercio si possono classificare in quattro categorie: micro PLC, piccoli PLC, medi PLC e grandi PLC.
I micro PLC trattano fino a 64 punti di ingresso/uscita sono generalmente di tipo digitale con memorie di 1 o 2K. Non hanno di solito una struttura modulare ad armadio.
I piccoli PLC trattano da 64 a 512 punti I/O, sono in predominanza digitali con una memoria che arriva fino a 4K. Hanno una struttura modulare ad armadio.
I medi PLC possono arrivare a trattare da 256 a 2048 punti di I/O e avere una memoria che arriva a qualche decina di K. Hanno struttura modulare ad armadio ed elevate capacità di comunicazione in rete informatica.
I grandi PLC possono trattare qualche migliaio di punti di I/O, ed hanno una memoria di centinaia di K. Solitamente sono utilizzati come supervisori di celle automatizzate e come interfacciamento di PLC di minori prestazione e calcolatori di gestione.

Le principali aziende produttrici di controllori a logica programmabile presenti sul mercato italiano sono:

- Siemens,
- Allen-Bradley,
- Omron,
- Telemécanique,
- Toshiba,
- Klockner-Moeller,
- Mitsubishi,
- Bosch,
- Hitachi,

Linguaggi di programmazione del PLC

I linguaggi di programmazione per PLC previsti dallo standard IEC 1131 del Comitato Elettrotecnico nazionale sono cinque:

- diagramma funzionale sequenziale (o SFC, acronimo di Sequential Functional Chart): permette di formulare le applicazioni di controllo di processi fisici utilizzanti i concetti di fase e di transizione. Brevemente le fasi rappresentano le azioni da compiere, eventualmente in parallelo, mentre le transizioni rappresentano le condizioni da soddisfare affinché si possa passare da una fase all'altra.
- linguaggio a contatti (Ladder Diagram): utilizza gli elementi tipici degli schemi di controllo a relè elettromeccanici: contatto aperto, contatto chiuso, bobina e le funzioni di temporizzazione e di conteggio. E' stato concepito per trattare principalmente la logica binaria.
- diagramma a blocchi funzionali (o FBD, Functional Block Diagram):è anch'esso un linguaggio concepito per trattare principalmente la logica binaria.

- lista di istruzioni : è un linguaggio di programmazione testuale simile ai linguaggi assemblativi usati per programmare gli attuali microprocessori. E' presente soprattutto nei PLC di origine tedesca (Siemens) con la sigla AWL ed è utilizzato soprattutto da programmatori che provengono dal mondo informatico.
- testo strutturato: è un linguaggio di programmazione testuale ad un alto livello, simile al Pascal con adattamento all'uso nei sistemi di controllo.

Per il nostro progetto è stato utilizzato il linguaggio a contatti.
adesso vediamo più dettagliatamente tale tipo di linguaggio.

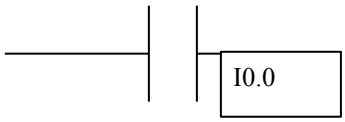
Linguaggio di programmazione a contatti

Il linguaggio a contatti più noto come il nome in inglese Ladder diagram è il più diffuso linguaggio di programmazione per il PLC.
La forma che assume un programma scritto con il linguaggio a contatti deriva dalla logica a relè dalla quale ha conservato la simbologia. Questo significa che, se si esprime un problema di controllo secondo il linguaggio a contatti quello che si ottiene non è altro che lo schema elettrico che si dovrebbe realizzare con dei relè per ottenere il controllo desiderato.
Il linguaggio a contatti non è introdotto liberatamente dai costruttori di PLC ma è stato imposto dalle condizioni di mercato.
I primi sistemi di controllo industriale sono stati realizzati utilizzando dispositivi elettromeccanici come i relè, i temporizzatori, i contatori, i sequenziatori a cilindro. Quando cominciarono ad essere disponibili i primi controllori a programma memorizzato, si rese necessario sviluppare un linguaggio di programmazione che fosse vicino alla tecnologia a relè e che potesse essere facilmente essere compreso da chi non avesse competenze di elettronica e di informatica.

Un programma scritto con il ladder è costituito da 2 linee verticali, inserite una a sinistra e una a destra del foglio di lavoro, che rappresentano l'alimentazione . Tra queste 2 linee verticali ci sono poi varie linee orizzontali (rung) . Tali linee orizzontali contengono i relè (segnali di ingresso) che devono abilitare opportunamente una determinata uscita. In genere i relè vengono posti dalla parte di sinistra mentre le uscite vengono poste sulla destra.

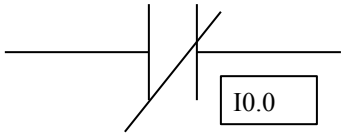
Adesso riportiamo come vengono rappresentati i relè e le uscite:

ingressi:



Questo contatto presente nel linguaggio del ladder è una copia del relè che nella situazione attuale si trova in un opportuno ingresso di uno dei moduli del PLC. (Nel disegno riportato sopra l'ingresso corrispondente è quello n°0 del modulo 0). Come ingresso si può presentare anche un flag memorizzato della memoria utente del PLC.

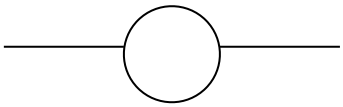
In questa situazione pertanto, se all'ingresso opportuno il cui contatto fa riferimento, arriva un segnale alto, (livello logico 1), allora il contatto si chiude, caso contrario rimane aperto.

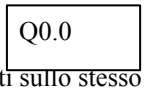


Questo contatto presente nel linguaggio del ladder è una copia dell'opposto del relè che nella situazione attuale si trova in un opportuno ingresso di uno dei moduli del PLC. (Nel disegno riportato sopra l'ingresso corrispondente è quello n°0 del modulo 0). Come ingresso si può presentare anche un flag memorizzato della memoria utente del PLC.

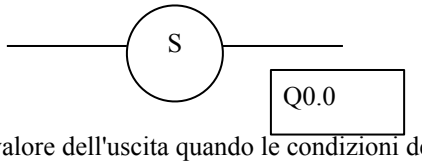
In questa situazione pertanto, se all'ingresso opportuno il cui contatto fa riferimento, arriva un segnale alto, (livello logico 1), allora il contatto si apre, caso contrario rimane chiuso.

uscite

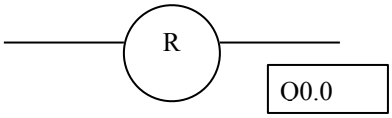




se le istruzioni di ingresso presenti sullo stesso rung è vera allora l'uscita relativa viene posta ad 1.
L'uscita in tal caso può essere uno presente in un modulo di uscita oppure un flag di sistema.

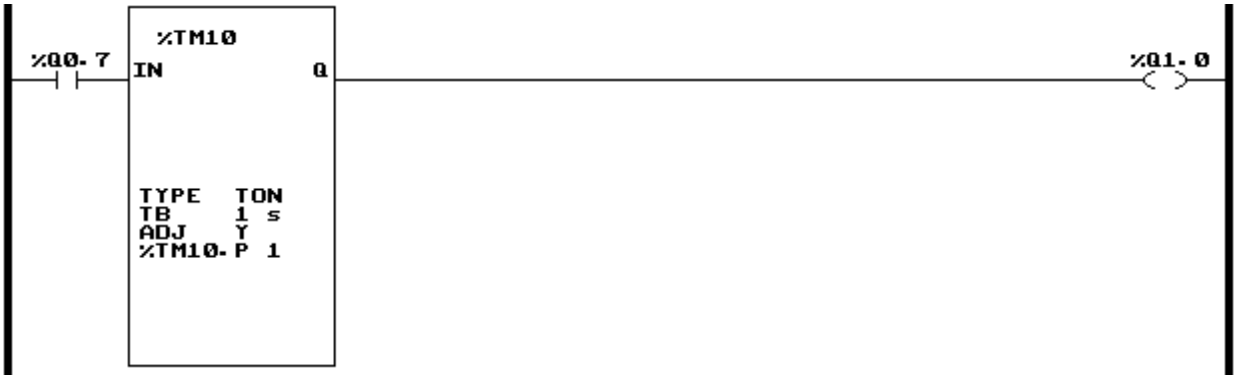


Setta il valore dell'uscita quando le condizioni dell'ingresso sono verificate.
Settare significa portare ad uno il valore dell'uscita quando le condizioni sono verificate e continuare a rimanere a uno anche quando successivamente le condizioni di ingresso vengono modificate. Per riportare a 0 l'uscita occorre un operazione di reset.



Serve a resettare l'uscita nell'ipotesi che tutte le condizioni di ingresso sono verificate. Il comportamento del reset è il contrario di quello del set.

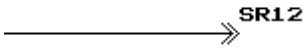
Tra i blocchi di temporizzazione e di conteggio descriviamo solo il seguente blocco di temporizzazione: (questo blocco è l'unico utilizzato nel nostro progetto)



Questo blocco fa si che l'uscita diventa attiva dopo un determinato tempo dall'attivazione dell'ingresso del temporizzatore.

Per progetti di grandi dimensioni (come quello trattato) si preferisce suddividere il programma in tante subroutine, ognuno delle quali esegue un compito specifico. In questo modo è più semplice capire il programma stesso e nello stesso modo diventa più semplice individuare eventuali errori presenti nel programma.

Per eseguire il passaggio alla subroutine si utilizza nel ladder come indicatore di uscita il seguente simbolo:



In tal caso passa alla subroutine se tutte le condizioni di ingresso vengono rispettate.
Per ritornare alla rung successivo a quello che ha determinato il salto alla subroutine bisogna introdurre alla fine della subroutine la seguente uscita.

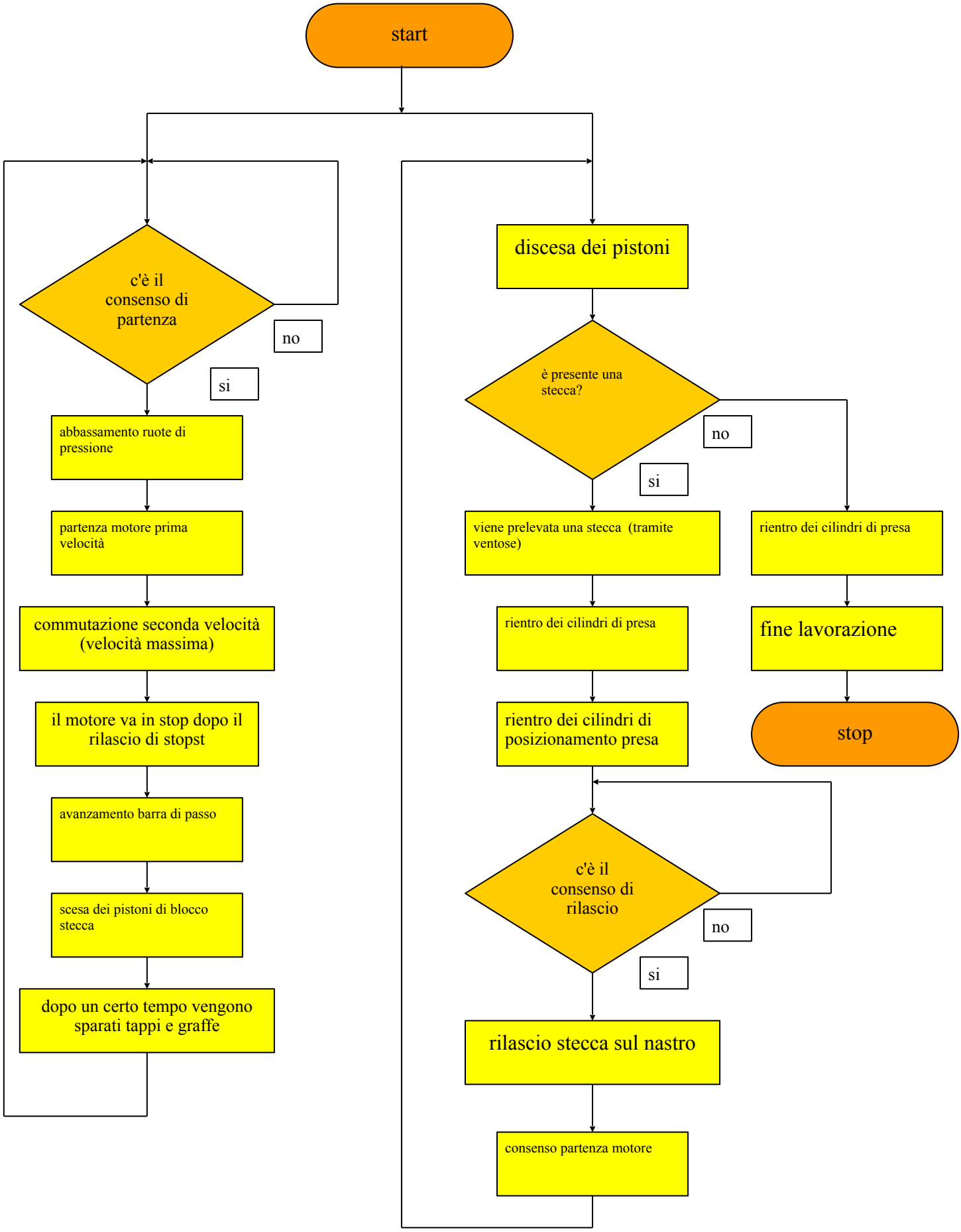
Parte hardware della confezionatrice teli

La confezionatrice teli è una macchina che funziona in parte elettronicamente e in parte pneumaticamente. Questa macchina viene pilotata da un PLC.
La macchina è corredata da dei pistoni ad aria, dei finecorsa a contatto, un motore AC per il movimento del nastro trasportatore, 2 graffatrici ad aria ed un vacuostato. La macchina e munita anche di un pannello di controllo nel quale sono alloggiati il PLC e la scheda motore per la regolazione della velocità, una ventola di raffreddamento, un interruttore differenziale (utile per bloccare la tensione di rete) ed una scheda matrice diodi.
Sulla parte superiore è montato un selettore a 12 posizioni con il quale è possibile lavorare sia in ciclo continuo che in manuale; questa operazione viene fatta tramite due pulsanti: uno avvia tutte le funzioni manuali (MANUAL) e l'altro il ciclo continuo (START).
Inoltre sono presenti 2 potenziometri utili per variare la velocità minima e massima del motore. (questo potenziometro agisce direttamente sulla scheda motore).
E' presenta anche un pulsante a fungo utilizzato come pulsante di emergenza.
La scheda matrice diodi serve per far sì che il selettore presenta una codifica binaria per le 8 posizioni (su 12) utilizzate.
I finecorsa fanno parte dei sensori di posizione. Questi sono molto importanti per il funzionamento della macchina dando la possibilità di fornire informazioni di tipo ON-OFF garantendo il controllo dei vari passi che compie la macchina.

Descrizione sintetica del funzionamento della macchina

Premuto il pulsante di ciclo continuo (START) posizionando il selettore su 0, scendono i 2 pistoni di presa per prelevare una stecca.
La stecca viene prelevata dalle ventose messe in depressione da un venturi. A questo punto il PLC, ricevuto il segnale dal vacuostato che la stecca è stata presa, fa salire i pistoni portandosi su la stecca.
Una volta finita l'ascesa vengono comandati i pistoni di posizione presa. A questo punto la stecca viene lasciata sul nastro trasportatore, tenuta fissa tramite delle ruote di pressione. Dopo di che parte il motore, facendo avanzare la stecca alla prima velocità (Vmin). Quando la stecca si è infilata nella stecca precedentemente processata, il motore aumenta la velocità (Vmax) fino quando questa non è completamente infilata nell'altra, quindi il motore si ferma, le ruote di pressione si alzano e una barra (detta di passo) fa scorrere un avanti il telo. Dopo di ciò la stecca viene bloccata e la barra di passo torna indietro. Successivamente vengono inseriti i tappi tramite due pistoni (detti martello tappo) e vengono sparate le graffe per mezzo di graffratrice ad aria.
Quest'esecuzione si ripete fino a quando finiscono le stecche nel caricatore infatti quando i cilindri di presa scendono e non trovano la stecca, non rilevano, tramite i venturi, differenza di pressione, perciò si portando in posizione di standby (in alto) e mandano in stop il PLC (questo non significa che il programma non cicla più). Si è così giunti a fine lavorazione, praticamente il telo è stato completato.

Diagramma di flusso del ciclo continuo



Descrizione completa dei passi del ciclo continuo

il ciclo continuo inizia tramite la subroutine SR15

in tal caso ci deve essere il consenso del flag RUN che rappresenta il flag di ciclo continuo. Tale bit viene messo ad 1 dalla sobroutine che si interessa della verifica delle condizioni che permettono di fare avviare il ciclo continuo.

discesa cilindri di presa ed azionamento venturi:
se è 1 il flag di RUN e se:
il pistone di posizionamento presa è in avanti ,
il pistone di presa è sopra,
e se in tale condizione non è presente la stecca attaccate alle ventose, (stecsi1=0)
allora fa scendere il pistone di presa e nello stesso tempo aziona il venturi utile per aspirare la stecca una volta che le ventose la toccano.

salita cilindri di presa in presenza di stecca:
Se il pistone di posizionamento presa è in avanti (finecorsa pospr+=1)
se il pistone di presa è giù, (finecorsa presa+=1)
e se è presente la stecca (in tal caso l'ingresso proveniente dal vacuostato è 1)
allora fa salire il pistone di presa.
Nello stesso tempo setta il flag stecsi1 che dice che nella posizione in cui il pistone di presa è in alto e quello di posizionamento presa è in avanti, è presente una stecca.
Tale posizione possiamo definirla posizione di inizializzazione.

salita cilindri di presa in assenza di stecca
se nella parte precedente di programma non è stata trovata la stecca (l'ingresso del vacuostato non è 1) allora l'uscita che da aria al pistone di presa è sempre attivo. In tal caso pertanto se non è presente la stecca, tale uscita può segnalare che non è stata trovata la stecca.
In tal caso i pistoni di presa devono comunque ritornare su (nella posizione che si può definire iniziale) . Tale operazione viene fatta dopo un tempo di temporizzazione.
Dopo tale tempo il pistone di presa ritorna sopra,
viene spento il venturi,
viene resettata la condizione che indica la presenza della stecca nella posizione di inizializzazione,
viene resettato il flag di RUN
viene resettato inoltre il consenso di far partire l'eccitazione che poi farà partire il motore.
Questa parte di programma ha senso quando sul piano di carico non sono presenti le stecche. Questa condizione si ha quando il telo è stato completato pertanto è chiaro che si deve disattivare il flag RUN e nello stesso tempo spegnere l'eccitazione del motore. L'eccitazione del motore è un uscita che permette di disattivare la scheda del motore presente nel pannello di controllo.

blocco ciclo se la stecca è persa
Se i pistoni di presa e di posizionamento presa stanno nelle condizioni di inizializzazione e se è attivo il flag di stecsi1 però nello stesso tempo non è attivo l'ingresso del vacuostato allora sicuramente nel tragitto di presa della stecca, si è persa la stecca. In tal caso quindi bisogna prendere dei provvedimenti.
Si resetta il flag RUN
si resetta il flag stecsi1 (infatti ci troviamo nella posizione di inizializzazione senza la stecca)
si attiva inoltre un flag di blocco lavorazione. Questo flag segnala che si è interrotta la lavorazione a causa della perdita di una stecca.

Rientro dei cilindri di posizionamento presa
Se per caso i cilindri stanno nella posizione di inizializzazione ed è presente la stecca attaccate alle ventose (in tal caso sia il vacuostato che il flag stecsi1 devono essere 1) e se è attivo almeno uno di questi due flag:
caricast: consenso che viene dato da parte di condizioni che assicurano che il rientro dei cilindri non crea problemi (per esempio la presenza di un'altra stecca sul nastro trasportatore)
primast: rappresenta il flag che indica che la stecca presa è la prima del processo.
in tal caso il pistone di posizionamento presa ritorna indietro e fa cadere la stecca sul nastro trasportatore. nello stesso tempo disattiva il flag di caricast in quanto non deve essere possibile che una altra stecca che può essere prelevata dai pistoni venga messa sul nastro trasportatore.

Avviamento motore nastro
Quando la stecca è presente sul nastro trasportatore allora si può dare il consenso per far partire il motore a patto che la barra di passo non occupa il nastro trasportatore (parleremo successivamente della barra di passo). Il consenso può essere dato da pospr-.

Abbassamento ruote di pressione e ritorno dei pistoni della condizione di inizializzazione

una volta che si è avviato il motore dopo un tempo T1 (pochi secondi), si abbassano le ruote di pressione. ritornano indietro i pistoni di posizionamento presa ritornando alla condizione di inizializzazione. In tal caso comunque si devono resettare i flag di primast (in quanto non di deve prendere poi la prima stecca del processo) e il flag di stecsi1 in quanto ci troviamo nella posizione di inizializzazione senza la presenza della stecca collegata alle ventose.
L'avviamento del motore non setta le ruote di pressione, pertanto le ruote di pressione si alzano subito nel momento in cui si ferma il motore

Comando seconda velocità

Nel momento in cui la stecca attiva entrambi i finecorsa di rallst e di stopst allora il motore che muove il nastro trasportatore aumenta la velocità

Blocco motore

Quando la stecca riapre il finecorsa di stopst allora si ferma. In tal caso per ciò che è stato detto due punti precedenti, le ruote di pressione si alzano. In tal caso diciamo che la stecca si trova alla fine del nastro trasportatore. (per indicare se è presente una stecca alla fine del nastro trasportatore, si utilizza un flag: flagsi2)

- Sul nastro trasportatore sono presenti 2 finecorsa: rallst e stopst per fare in modo che quando il motore aumenta la velocità, la stecca sia infilata di una quantità pari alla distanza tra i due finecorsa nella stecca processata precedentemente che la barra di passo ha spostato in avanti.

Avanzamento barra di passo e sblocco stecca

Nel momento in cui le ruote di pressione si sono alzate e il finecorsa stopst è stato aperto e se la barra di passo è indietro, allora dopo un piccolo tempo di temporizzazione si aziona la barra di passo in modo da portare la stecca in avanti. Nello stesso momento si sblocca la stecca che nel ciclo precedente era stato spostato in avanti in modo tale che la barra di passo nel momento in cui spinge il telo in costruzione, non trova ostacoli che fa scorrere il telo verso avanti
In tal caso si resetta anche il flag stecsi2 in quanto si toglie la stecca dalla fine del nastro trasportatore.
A questo punto la disattivazione del finecorsa della barra di passo negativo: barpas- fa si che si attiva il flag caricast (visto precedentemente). L'attivazione di questo flag invita a dire che il nastro trasportatore è libero e quindi è logico che i pistoni di presa stecca e di presa hanno il consenso di prendere un'altra stecca dal piano di carico.

Rientro barra di passo

Se per caso la barra di passo è in avanti (in tal caso barpas+=1) allora dopo un piccolo tempo di ritardo si fa ritornare indietro la barra di passo.

Blocco della stecca

Una volta che la barra di passo sporta la stecca in avanti (cioè nel momento in cui barpas+=1) viene bloccata la stecca con i pistoni di blocst.

Comando martello tappo

Una volta che la stecca è stata bloccata allora si devono inserire i tappi dopo un piccolo tempo di ritardo determinato da un temporizzatore

Comando graffatrice

Una volta che la stecca è stata bloccata allora si devono inserire le graffe dopo un piccolo tempo di ritardo determinato da un temporizzatore.
Prima di inserire le graffe devono però essere inseriti i tappi pertanto il tempo di temporizzazione per l'inserimento delle graffe deve essere più lungo di quello utilizzato per inserire i tappi.

- Se non finiscono le stecche oppure non si perdono le stecche durante il processo di presa, allora il processo va avanti senza problemi. Inoltre il modo di procedere permette di prelevare una stecca dal piano di carico prima ancora che la stecca precedente arriva alla fine del ciclo di lavorazione. Questo aumenta di molto il tempo impiegato per completare un telo.

Fine lavorazione telo

Se per caso le stecche sono finite oppure si perdono durante il processo di presa, allora viene resettato il flag di RUN (come è stato visto in precedenza).
A questo punto viene completata il processo della stecca che era stata precedentemente prelevata e nel momento in cui viene portata in avanti dalla barra di stato, bloccata dal pistone di blocst e inseriti i tappi e graffe allora si spegne il motore e si setta il flag di finetelo che indica che il processo è stato fermato.
Precisiamo che alla fine di questo passo, la stecca rimane ancora bloccata.

Se il flag di RUN non è stato resettato da una perdita di stecca (cioè non viene settato bloclav) allora evidentemente il processo di RUN è stato disattivato dal fatto che le stecche nel piano di carico sono finite pertanto il telo è stato completato.
Quindi adesso possiamo dire che se bloclav=0 allora possiamo sbloccare la stecca.
Se invece questo non si verifica (cioè bloclav=1), allora la stecca rimane ancora bloccata.
La stecca rimane bloccata in quanto il telo ancora non è stato completato quindi si può riprendere il processo da dove era stato interrotto, una volta che si è provveduti a risistemare la stecca persa nel piano di carico e una volta che si respinge il pulsante di start presente sul pannello di controllo.

Lista attribuzione ingressi

I0.0	**POSPR-**	**finecorsa negativo per i cilindri di posizionamento presa** (vengono azionati se il pistone di posizionamento presa è in indietro)
I0.1	**POSPR+**	**finecorsa positivo per i cilindri di posizionamento presa** (vengono azionati se il pistone di posizionamento presa è in avanti)
I0.2	**PRESA-**	**finecorsa negativo per i cilindri di presa** (vengono azionati se il pistone di presa è sopra)
I0.3	**VACUO**	**vacuostato** (serve per rilavare se la stecca è appiccicata alle ventose)
I0.4	**RALLST**	finecorsa che serve per fare aumentare la velocità al motore in una determinata posizione del nastro trasportatore (questo finecorsa si trova su una ruota di pressione)
I0.5	**STOST**	finecorsa che serve per fare fermare la stecca in una determinata posizione del nastro trasportatore (tale finecorsa si trova su una ruota di pressione)
I0.6	**BARPAS-**	**finecorsa negativo della barra di passo** (viene azionato se la barra di passo si trova dietro)
I0.7	**BARPAS+**	**finecorsa positivo della barra di passo** (viene azionato se la barra di passo si trova in avanti)
I1.1		non utilizzato
I1.2	**EMERG**	**pulsante di emergenza** (presente sul pannello di controllo)
I1.3	**MANUAL**	**pulsante comando manuale** (presente sul pannello di controllo)
I1.4	**SEL 0**	**selettore bit 0** (bit meno significativo)
I1.5	**SEL 1**	**selettore bit 1**
I1.6	**SEL 2**	**selettore bit 2** (bit più significativo)
I1.7		non utilizzato

Q0.0	**POSPR**	**comando cilindro posizionamento presa** (questi cilindri sono quelli che vanno avanti ed indietro ed hanno il compito di portare la stecca sul nastro trasportatore)
Q0.1	**PRESA**	**comando cilindri di presa** (questi cilindri non quelli che vanno sopra e sotto ed hanno il compito di prendere la stecca dal piano di carico)
Q0.2	**MOTARM**	**comando motore** (fa partire il nastro trasportatore)
Q0.3	**VENTURI**	**comando elettrovalvola venturi** (serve per dare la possibilità alle ventose di aspirare le stecche)
Q0.4	**RUOTE**	**comando ruote di pressione** (serve per bloccare la stecca durante il tragitto del nastro trasportatore)
Q0.5	**VEL2**	**comando seconda velocità** (da la possibilità al motore di aumentare la velocità. Tale uscita si collega alla scheda del motore)
Q0.6	**BARPAS**	**comando barra di passo** (serve per portare in avanti la stecca in modo che il nastro trasportatore risulti poi libero. La stecca successiva che scorrerà sul nastro, si infilerà in quella spostata in avanti proprio dalla barra di passo)
Q0.7	**BLOCST**	**comando blocco stecca** (serve per bloccare la stecca una volta che è stata portata avanti dalla barra di passo)
Q1.0	**MART**	**comando martello tappo** (serve per inserire il tappo nella stecca una volta che è stata bloccata dal pistone di blocco stecca)
Q1.1	**GRAFFA**	**comando graffatrice** (serve per inserire la graffa nella stecca una volta che la stecca è stata bloccata e dopo che è stato inserito il tappo)
Q1.2	**MOTECC**	**comando eccitazione motore** (comando che alimenta la scheda del motore)
Q1.3	**ALLARME**	**allarme lampada** (serve per dare il segnale alla spia luminosa che si trova sul pannello di controllo che indica che qualche cosa non funziona bene)
Q1.4		non utilizzato
Q1.5		non utilizzato
Q1.6		non utilizzato
Q1.7		non utilizzato

Lista attribuzione uscite

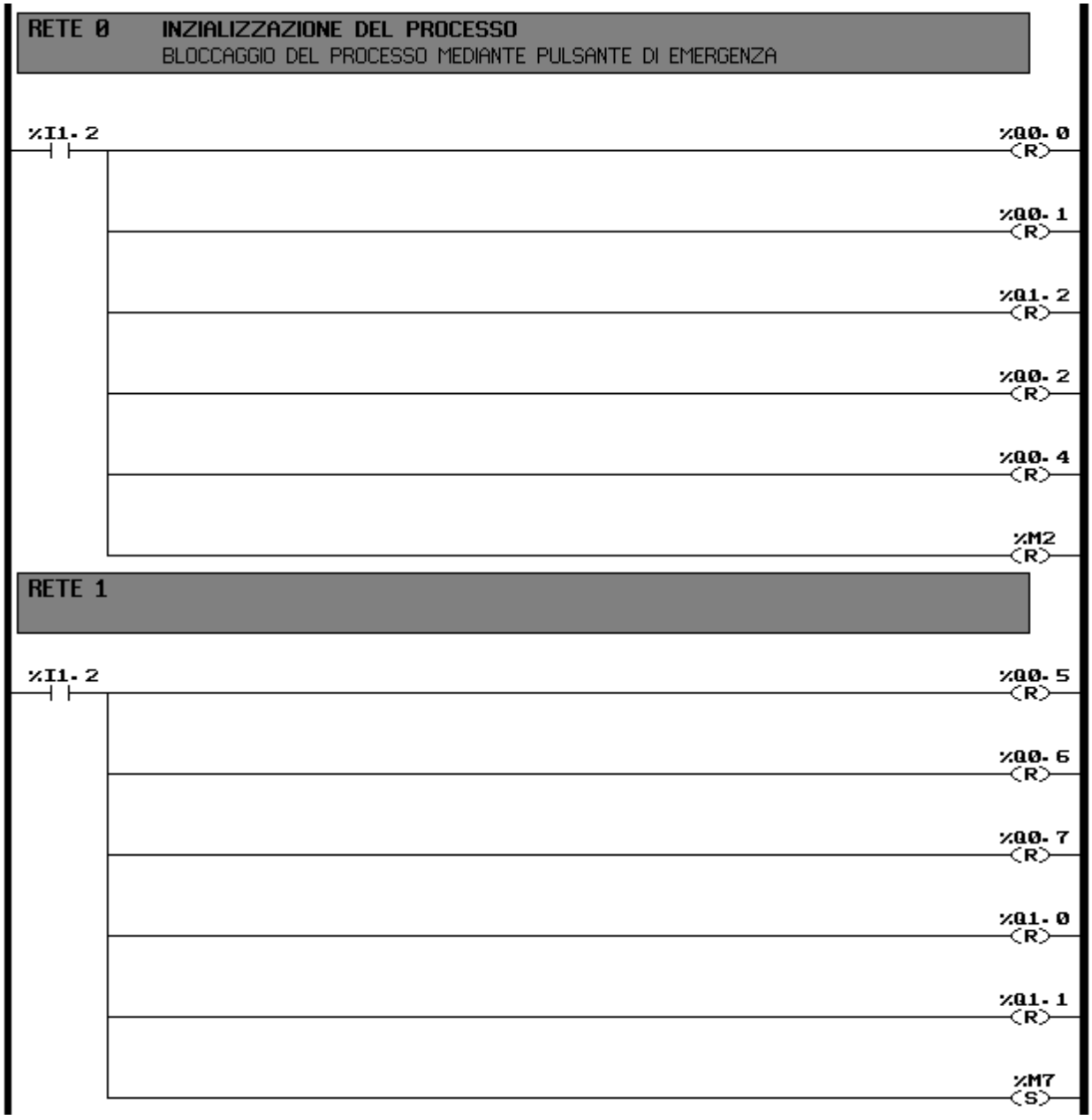
Lista attribuzione flag (merker)

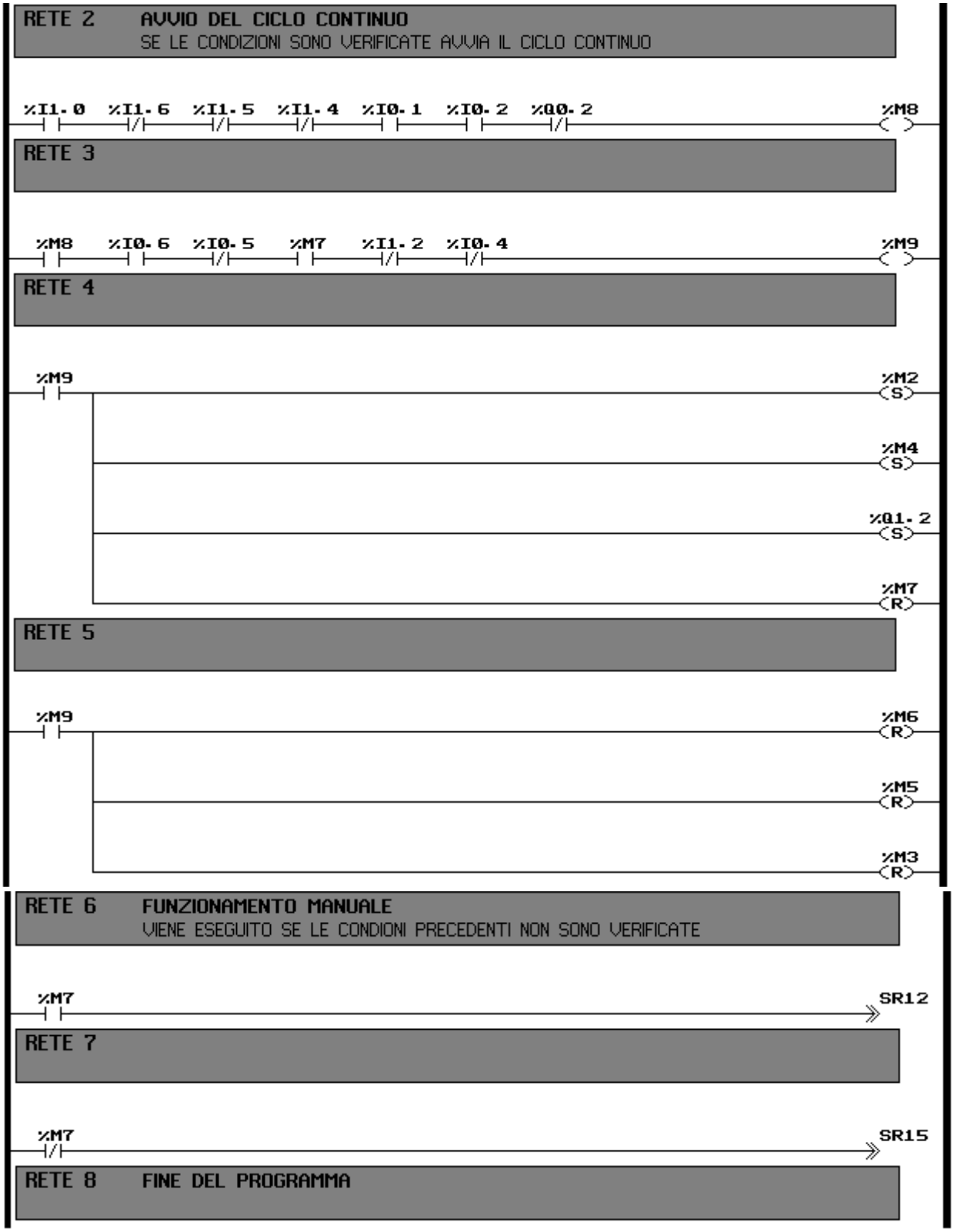
M1	**STECSI1**	**flag presenza stecca** (questo flag indica se è presente la stecca nella posizione di inizializzazione)
M2	**RUN**	**flag di ciclo continuo** (questo flag indica che il processo della macchina è in esecuzione)
M3	**BLOCLAV**	**flag di blocco lavorazione** (questo flag serve per indicare che il processo è stato interrotto a causa di una perdita accidentale della stecca da parte delle ventose nel momento in cui la si stava prelevando dal piano di carico
M4	**CARICAST**	**flag consenso di caricamento della stecca** (questo flag serve per informare che c'è il consenso o meno che la stecca può essere appoggiata sul nastro trasportatore)
M5	**PRIMAST**	**flag prima stecca** (questo flag serve per indicare che la stecca che si sta prelevando dal piano di carico è la prima)
M6	**STECSI2**	**flag presenza stecca** (questo flag indica se è presente la stecca alla parte finale del nastro trasportatore)
M7	**FINETELO**	**flag di fine telo** (questo flag indica che il telo è stato completato: il processo è concluso)

Posizioni del selettore

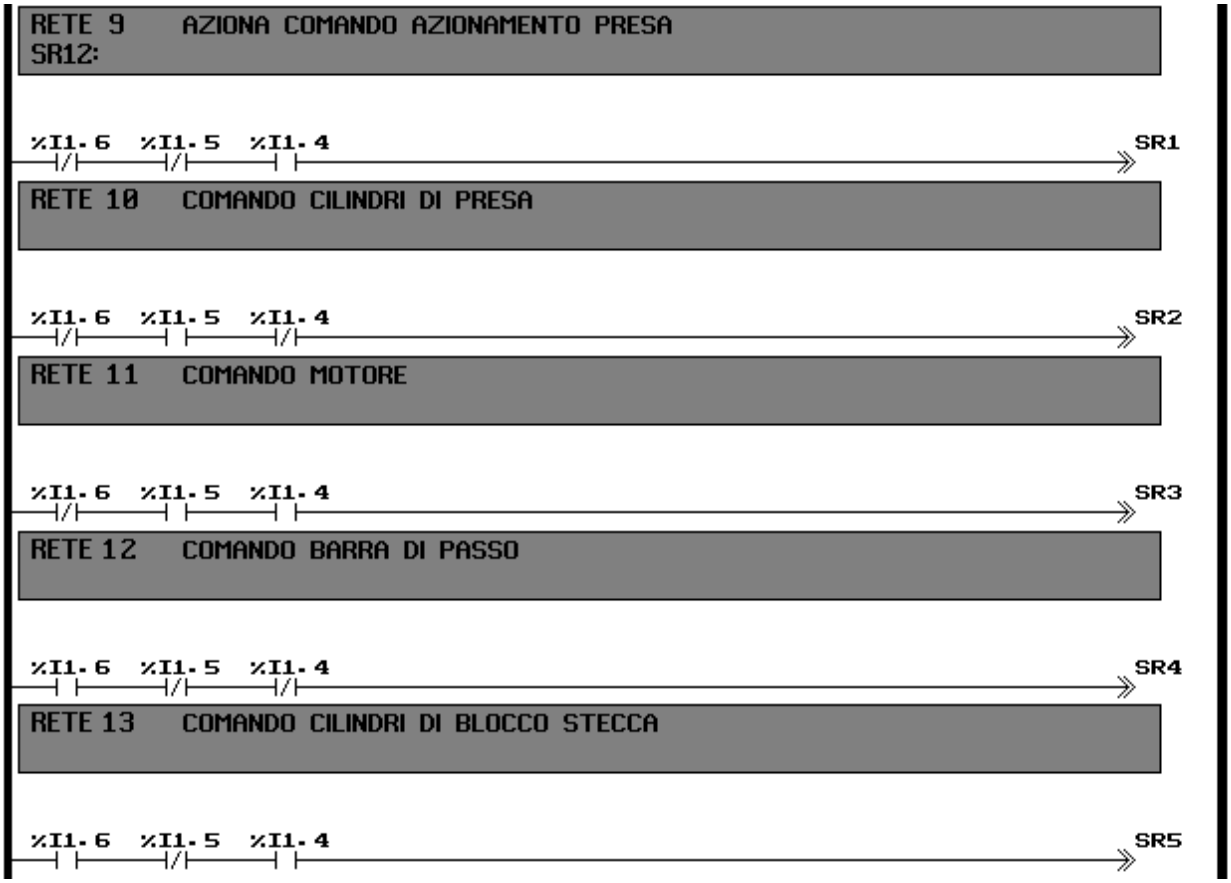
0 comando ciclo continuo
1 aziona comando posizionamento stecca
2 comando cilindri di presa
3 comando motore
4 comando barra di passo
5 comando cilindri di blocco stecca
6 comando cilindri martello tappo
7 comando graffatrici

Programma Ladder




RETE 2 AVVIO DEL CICLO CONTINUO
SE LE CONDIZIONI SONO VERIFICATE AVVIA IL CICLO CONTINUO
%I1.0 %I1.6 %I1.5 %I1.4 %I0.1 %I0.2 %Q0.2 %M8
RETE 3
%M8 %I0.6 %I0.5 %M7 %I1.2 %I0.4 %M9
RETE 4
%M9 %M2 %M4 %Q1.2 %M7
RETE 5
%M9 %M6 %M5 %M3
RETE 6 FUNZIONAMENTO MANUALE
VIENE ESEGUITO SE LE CONDIONI PRECEDENTI NON SONO VERIFICATE
%M7 SR12
RETE 7
%M7 SR15
RETE 8 FINE DEL PROGRAMMA

Link alla subroutine che eseguono i comandi manuali



Comandi manuali

RETE 17 COMANDO MANUALE SELETTORE 1
SR1: PORTA AVANTI E INDIETRO I PISTONI DI POSIZIONAMENTO PRESA

%I1.3 %I0.0 %Q0.0 (R)

%I0.1 %I0.2 %Q0.0 (S)

RETE 18 RILASCIO STECCA SE I PISTONI DI POSIZIONAMENTO SONO IN INDIETRO

%I0.0 %Q0.3 (R)

RETE 19

RET

RETE 20 SPOSTA SOPRA E SOTTO I PISTONI DI PRESA
SR2: PORTA I PISTONI GIÙ

%I1.3 %I0.1 %Q0.1 (S)

%Q0.3 (S)

RETE 21 SE È PRESENTE LA STECCA ALLORA PORTA I PISTONI SÙ

%I0.3 %Q0.1 (R)

RETE 22 SE NON È PRESENTE LA STECCA ALLORA DOPO 5S I PISTONI VANNO SU

%Q0.1 %TM1 IN Q Q0.1 (R)

TYPE TON
TB 1 min
ADJ Y
%TM1.P 5

RETE 23

RET

RETE 24 COMANDO MANUALE SELETTORE 3
SR3: ABBASSA LE RUOTE DI PRESSIONE E AVVIA IL MOTORE

%I1.3 %I0.4 %I0.5 %Q1.2 (S)
%Q0.4 (S)

RETE 25 PARTENZA MOTORE DOPO UN RITARDO DI 2 SEC.

%Q1.2 %TM6 IN Q %Q0.2
TYPE TON
TB 1 s
ADJ Y
%TM6.P 2

RETE 26 AUMENTO DELLA VELOCITA'
QUESTO AVVIENE SE ENTRAMBI I FINECORSA VENGONO ATTIVATI

%I0.5 %I0.4 %Q0.5

RETE 27 STOP MOTORE
QUANDO IL PRIMO FINECORSA VIENE DISATTIVATO IL MOTORE SI FERMA

%I0.5 %I0.4 %Q1.2 (R)
%Q0.4 (R)

RETE 28

RET

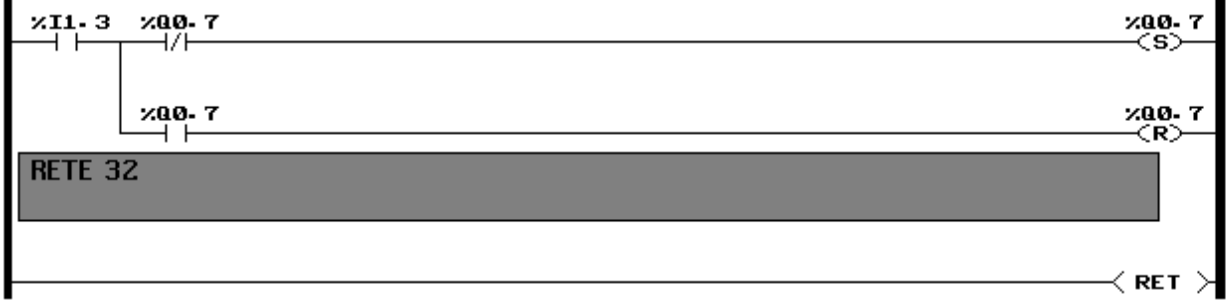
RETE 29 COMANDO MANUALE SELETTORE 4
SR4: ATTIVAZIONE E DISATTIVAZIONE BARRA DI PASSO

%I1.3 %Q0.4 %Q0.7 %I0.6 %Q0.6 (S)
%I0.7 %Q0.6 (R)

RETE 30

RET

RETE 31 COMANDO MANUALE SELETTORE 5
SR5: BLOCCAGGIO E SBLOCCAGGIO STECCA

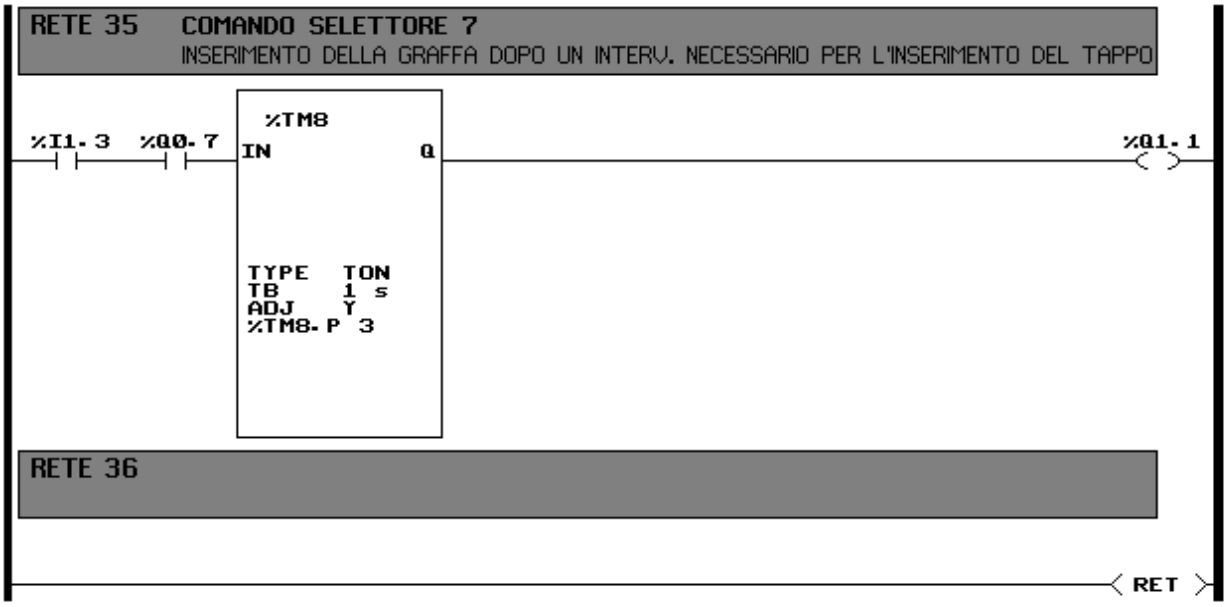


RETE 32

RETE 33 COMANDO MANUALE SELETTORE 6
SR6: ATTIVAZIONE E DISATTIVAZIONE MARTELLO TAPPO

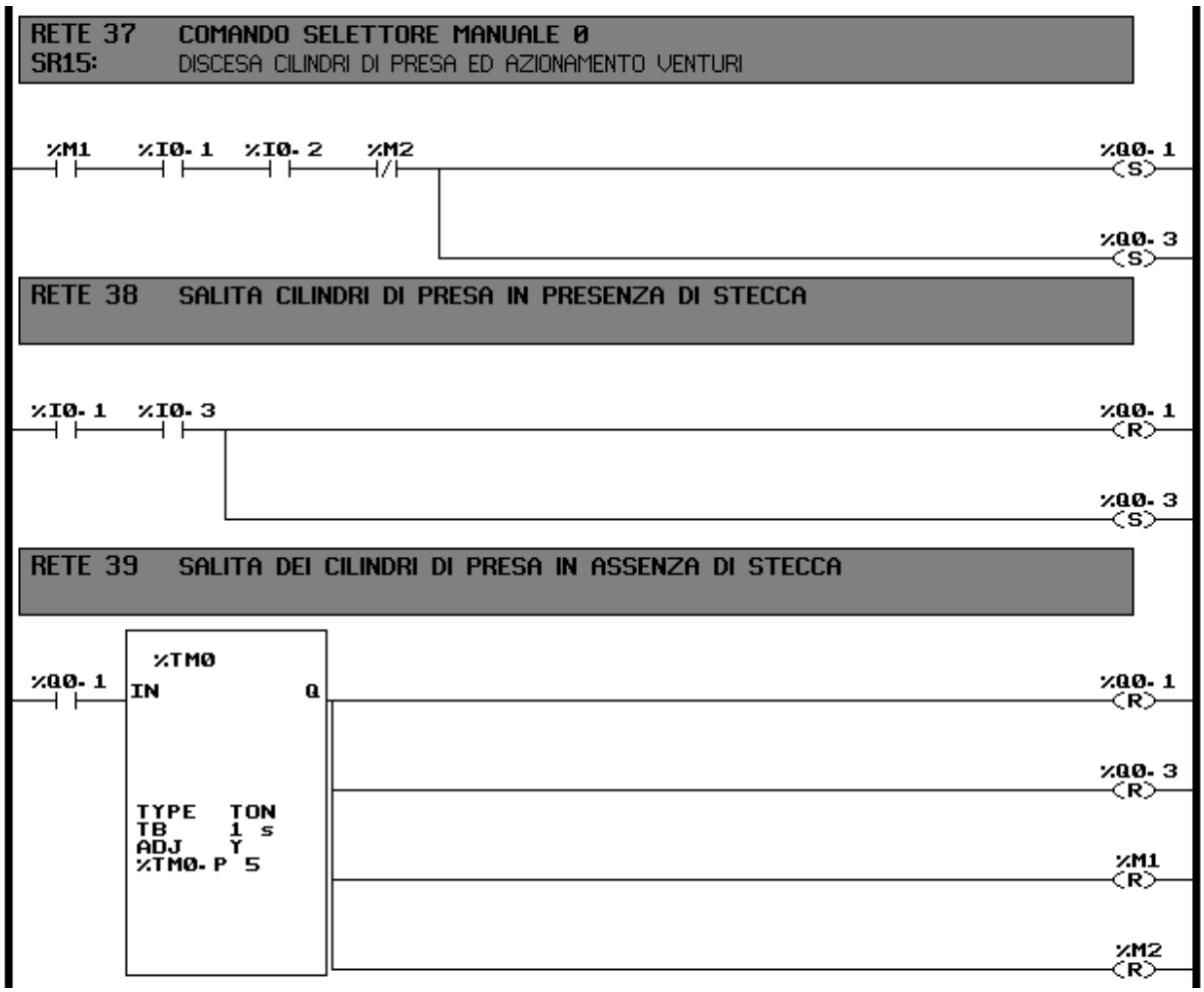
RETE 34 FINE DEL PROGRAMMA

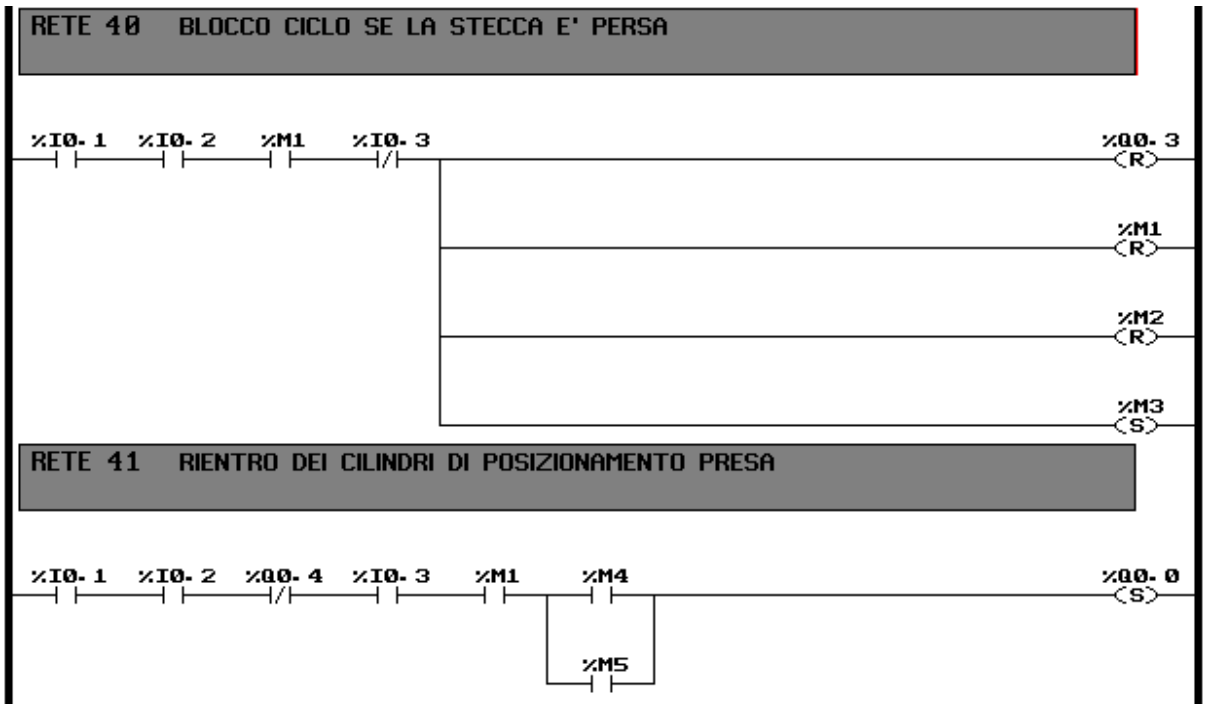
RETE 35 COMANDO SELETTORE 7
INSERIMENTO DELLA GRAFFA DOPO UN INTERV. NECESSARIO PER L'INSERIMENTO DEL TAPPO



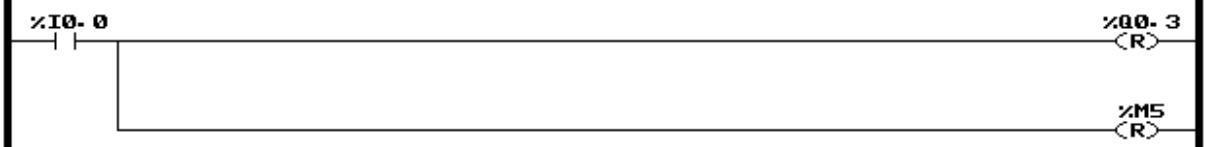
RETE 36

Ciclo continuo

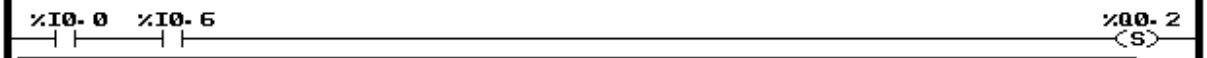




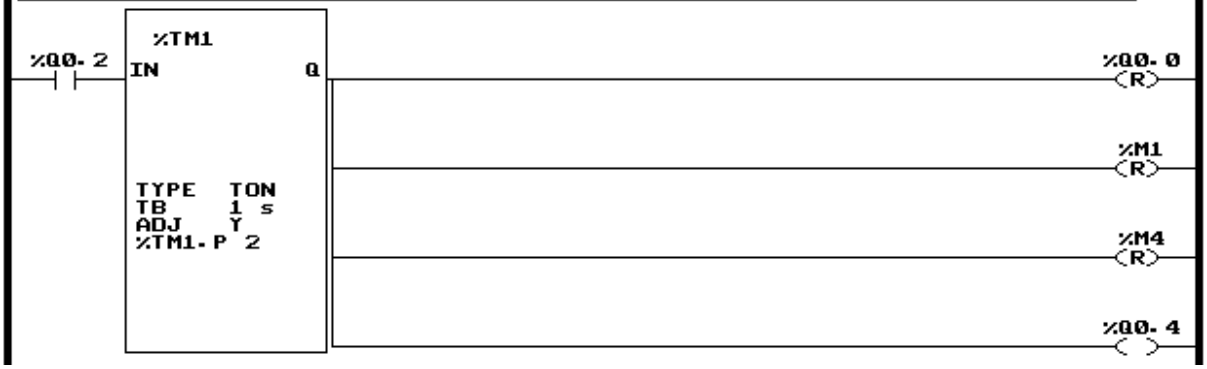
RETE 42 RILASCIO STECCA SUL NASTRO



RETE 43 AVVIAMENTO MOTORE NASTRO

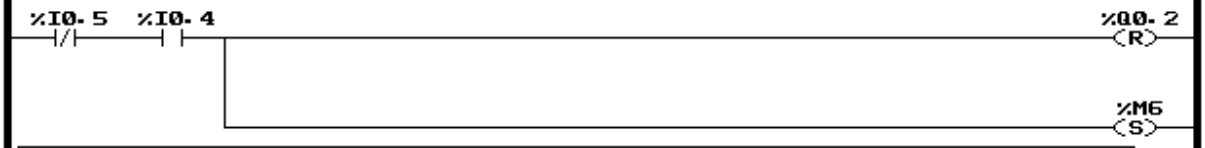


RETE 44 ABBASSAMENTO RUOTE DI PRESSIONE
RITORNO DEI PISTONI NELLA CONDIZIONE DI INIZIALIZZAZIONE

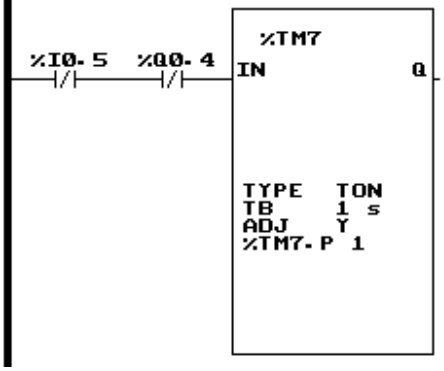


RETE 45 COMANDO SECONDA VELOCITA'

RETE 46 BLOCCO MOTORE



RETE 47 AVANZAMENTO BARRA DI PASSO E SBLOCCO STECCA SUCCESSIVA



RETE 48

%TM7.Q %I0.6 %M6 %Q0.7 (R)

%Q0.6 (S)

RETE 49 RIENTRO BARRA DI PASSO

%I0.7 %TM9 IN Q %Q0.6 (R)

%M6 (R)

TYPE TON
TB 1 s
ADJ Y
%TM9.P 2

RETE 50 BLOCCO DELLA STECCA

%I0.7 %Q0.7 (S)

RETE 51 COMANDO MARTELLO TAPPI

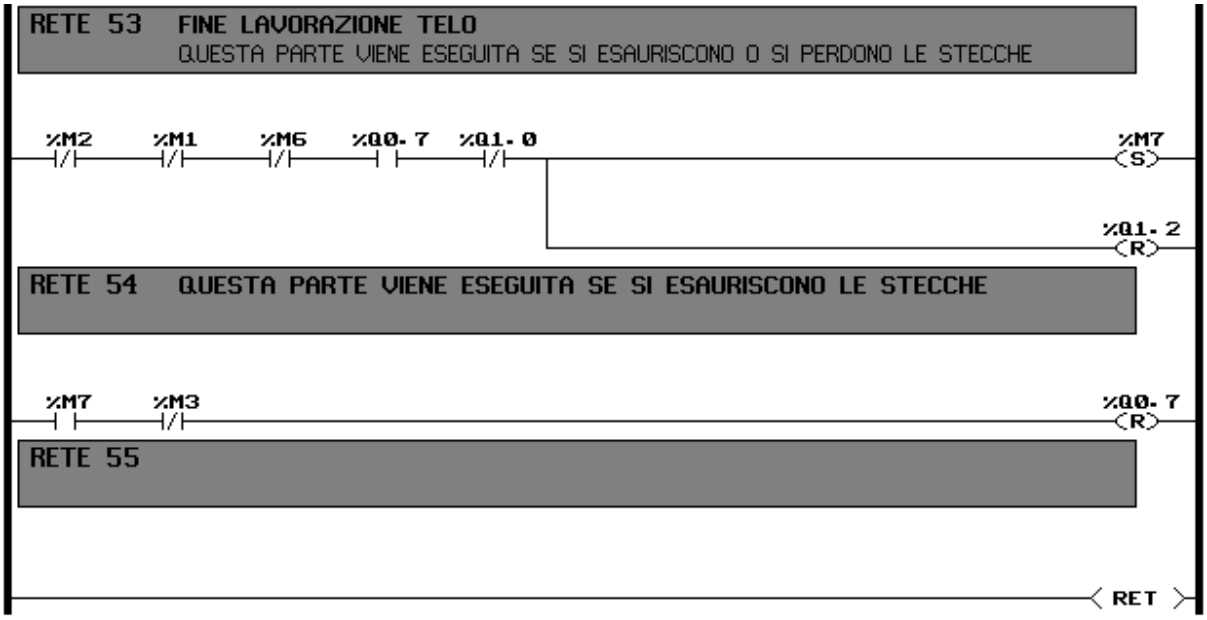
%Q0.7 %TM10 IN Q %Q1.0 ( )

TYPE TON
TB 1 s
ADJ Y
%TM10.P 1

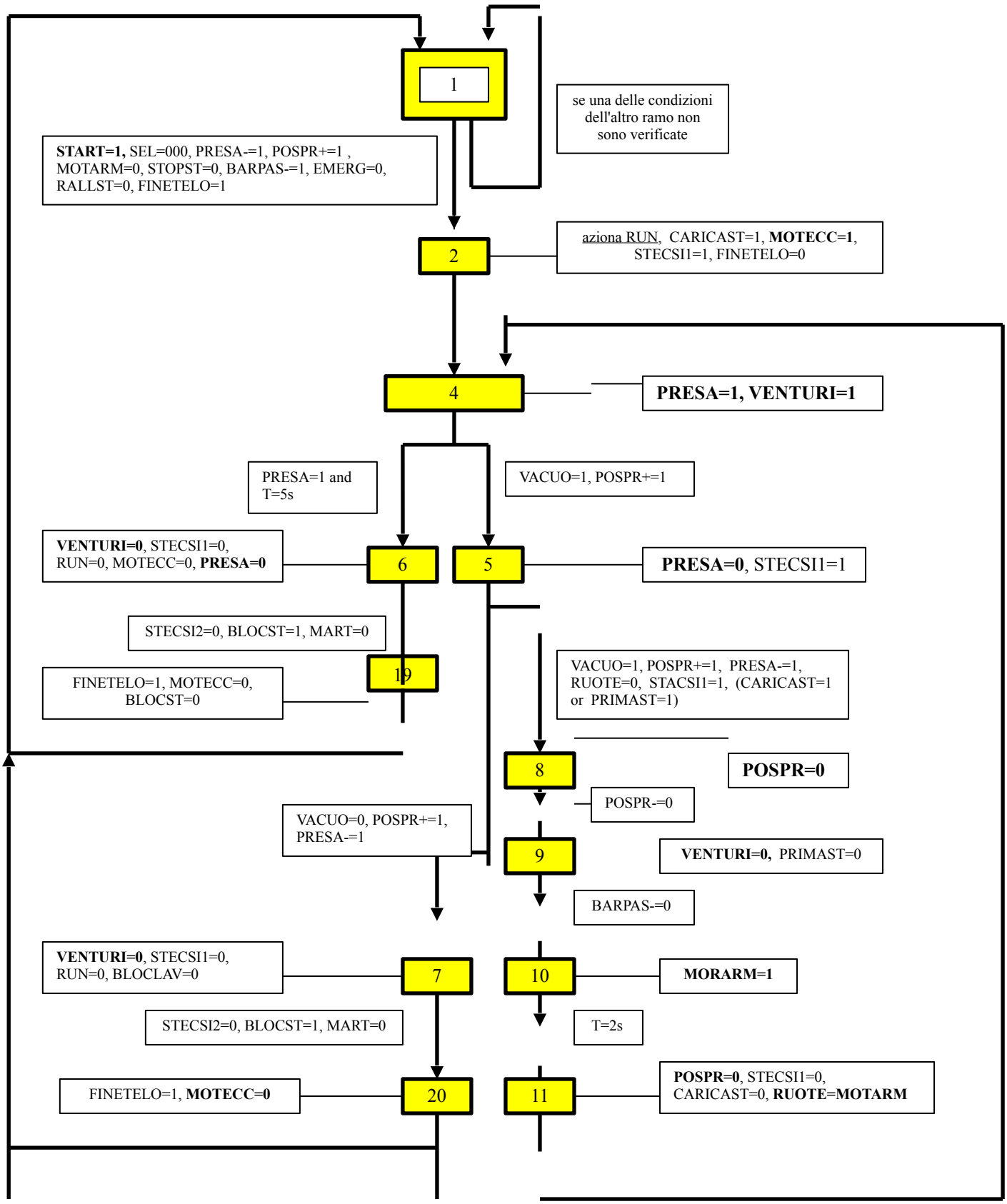
RETE 52 COMANDO GRAFFATRICE

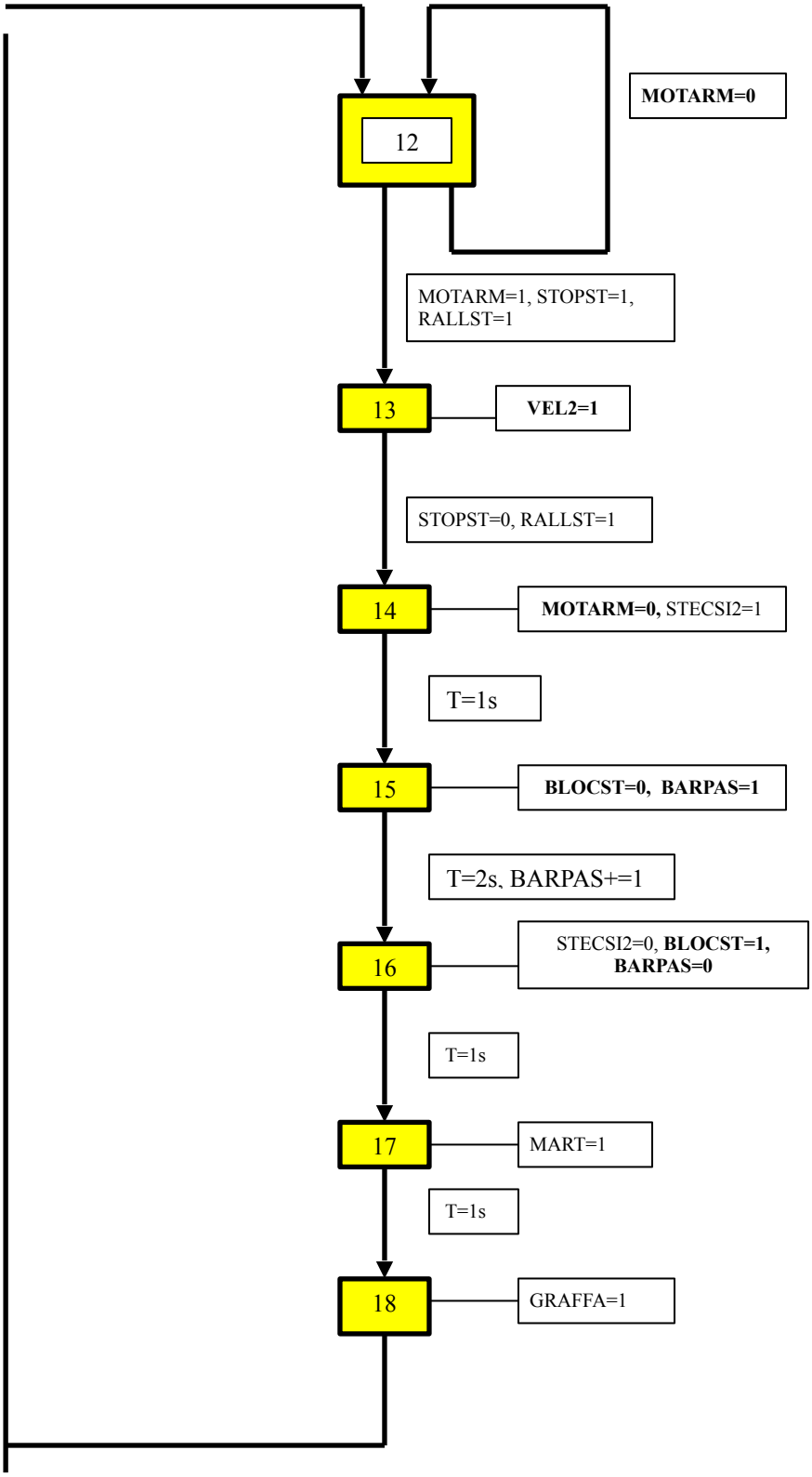
%Q0.7 %TM11 IN Q %Q1.1 ( )

TYPE TON
TB 1 s
ADJ Y
%TM11.P 2


RETE 53 FINE LAVORAZIONE TELO
QUESTA PARTE VIENE ESEGUITA SE SI ESAURISCONO O SI PERDONO LE STECCHE
%M2
%M1
%M6
%Q0.7
%Q1.0
%M7
S
%Q1.2
R
RETE 54 QUESTA PARTE VIENE ESEGUITA SE SI ESAURISCONO LE STECCHE
%M7
%M3
%Q0.7
R
RETE 55
RET

SFC del ciclo continuo




12
MOTARM=0
MOTARM=1, STOPST=1, RALLST=1
13
VEL2=1
STOPST=0, RALLST=1
14
MOTARM=0, STECSI2=1
T=1s
15
BLOCST=0, BARPAS=1
T=2s, BARPAS+=1
16
STECSI2=0, BLOCST=1, BARPAS=0
T=1s
17
MART=1
T=1s
18
GRAFFA=1