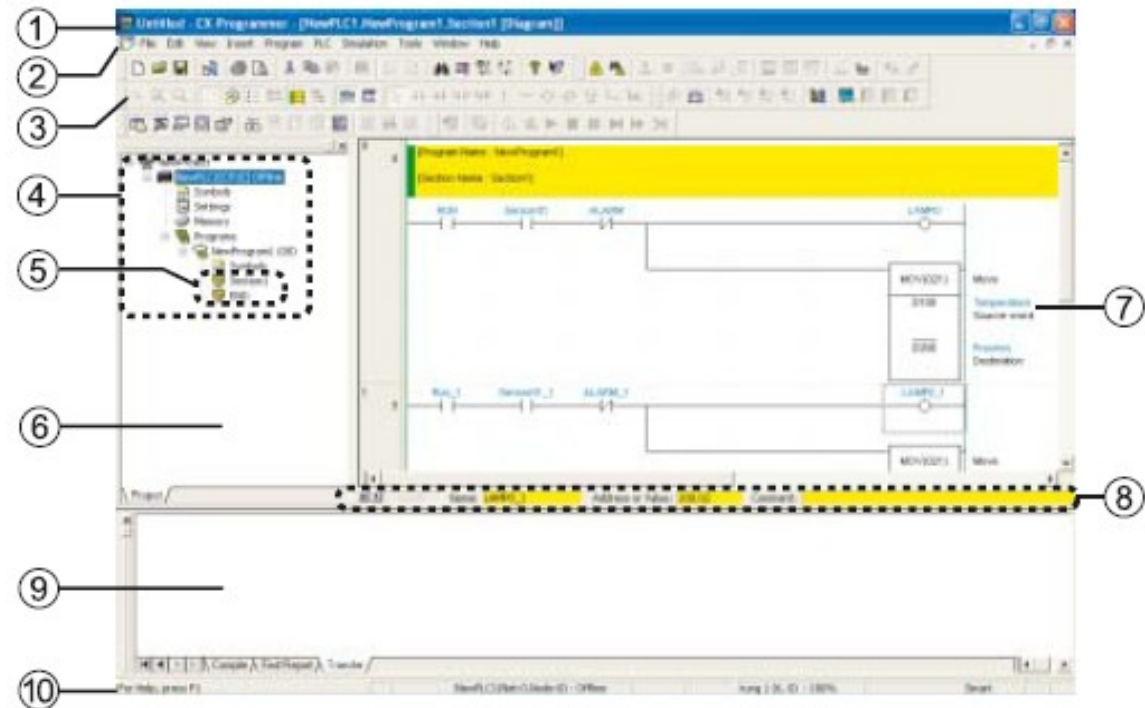


# Introduzione alla programmazione dei PLC



ITI "Alessandro Volta" - Sassuolo  
Anno Scolastico 2009/2010

# Programmazione dei PLC – 1

## Concetti introduttivi

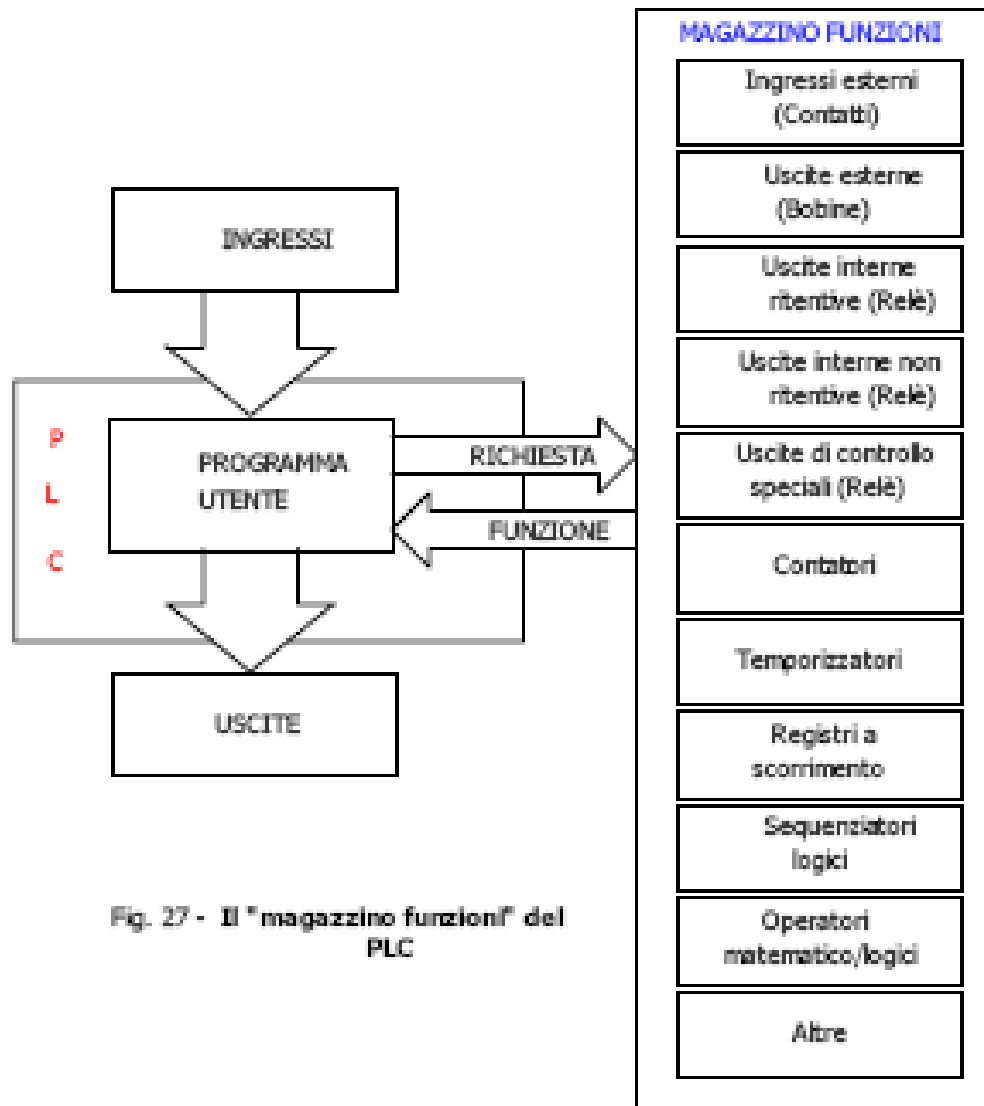


Fig. 27 - Il "magazzino funzioni" del PLC

- La scrittura di un programma per PLC si basa sull'utilizzo di un "magazzino" di funzioni del quale lo stesso dispone al suo interno (vedi figura);
- Il programma "preleva" in sequenza dal suddetto magazzino le funzioni che di volta in volta servono per il controllo dei vari attuatori;
- Il tipo ed il numero delle funzioni disponibili dipende dal modello del PLC utilizzato;
- In generale gli elementi funzionali dei PLC si possono suddividere in:
  - Dispositivi logici combinatori;
  - Dispositivi logici sequenziali;
  - Operatori e/o funzioni matematiche di vario tipo.

# Programmazione dei PLC – 2

## Elementi funzionali logici combinatori

### Ingressi esterni

Possono essere costituiti da *pulsanti, finecorsa, contatti di relè, sensori ottici, di prossimità ecc. fisicamente connessi al PLC* e sono visti dallo stesso come contatti normalmente aperti (contatti NA); nel contatto NA non si ha circolazione di corrente quando esso si trova in posizione di riposo, mentre la si ha quando il contatto viene azionato.

### Uscite esterne

Vengono viste dal PLC come bobine di relè; per questo motivo vengono anche chiamate direttamente relè di uscita o più semplicemente relè.

Quando il programma decide di “azionare” un'uscita esterna generalmente viene eccitato un relè interno al PLC che chiude un contatto, fornendo così la tensione di +24V che comanda il dispositivo fisicamente connesso all'uscita stessa; tale dispositivo può essere costituito da un *teleruttore, un'elettrovalvola, una lampada di segnalazione ecc.*

### Uscite di controllo interne (relè interni)

Sono dispositivi che non sono presenti fisicamente all'interno del PLC ma vengono “simulati” da quest'ultimo per offrire al programmatore la possibilità di memorizzare lo stato di un dispositivo esterno (I/O) oppure di effettuare una memorizzazione intermedia da utilizzare successivamente nel programma. I relè interni possono essere:

- **Ritentivi**: mantengono lo stato anche in caso di interruzione dell'alimentazione del PLC;
- **Non ritentivi**: in caso di interruzione dell'alimentazione lo stato viene perso ed il relè assume sempre lo stato OFF.

### Relè interni “speciali”

Sono dedicati a particolari funzioni di controllo come, ad esempio, la segnalazione di malfunzionamenti della CPU, il controllo dello stato della batteria tampone, la generazione di impulsi periodici ed altri ancora.

# Programmazione dei PLC – 3

## Elementi funzionali logici sequenziali

Vengono anche chiamati funzioni di box o di blocco funzionale e simulano all'interno del PLC i rispettivi dispositivi fisici: le loro modalità di funzionamento e programmazione dipendono sostanzialmente dal modello di PLC impiegato; i tipi principali sono i seguenti:

### Contatori (Counter)

Sono blocchi funzionali dedicati al conteggio di eventi.

In generale il blocco contatore richiede il collegamento ad un ingresso esterno che genera gli impulsi da contare (p. es. un fincorsa o una fotocellula) e l'impostazione di un valore di conteggio che rappresenta il numero di eventi da contare (in genere “a decremento”); l'uscita del contatore viene poi connessa ad un relè esterno che aziona un dispositivo quando il conteggio è arrivato al valore prefissato; spesso è presente anche un ingresso di reset che permette l'azzeramento “asincrono” del contatore.

### Temporizzatori (Timer)

Sono blocchi funzionali dedicati alla temporizzazione; la loro programmazione è abbastanza simile, in generale, a quella dei contatori: il blocco funzionale prevede l'impostazione di un valore numerico che esprime, ad esempio in secondi, il tempo da far trascorrere; un ingresso esterno fa partire il contatore che, sotto l'azione di un segnale di clock interno al PLC decrementa il valore di conteggio impostato fino ad arrivare a zero, azionando così il dispositivo di uscita previsto dal programma.

### Registri a scorrimento

Sono aree di memoria presenti all'interno del PLC la cui gestione simula il funzionamento di un registro a scorrimento (shift register):

# Programmazione dei PLC – 4

## Funzioni matematiche

Permettono al programmatore di eseguire diversi tipi di operazioni logiche e/o matematiche e/o il calcolo relativo alle più comuni funzioni generalmente impiegate nello sviluppo del software.

A tali scopi possono essere quindi disponibili, ad esempio:

- *Operatori di tipo aritmetico (+, -, \*, /);*
- *Operatori per effettuare confronti fra valori (<, >, =, <=, >=);*
- *Operatori logici (AND, OR, NOT, XOR);*
- *Operatori per effettuare conversioni numeriche di codice (binario-BCD, BCD-binario ecc.)*
- *Funzioni trigonometriche;*
- *Funzioni esponenziali e logaritmiche;*
- *Funzioni di elevamento a potenza ed estrazione di radice.*

Le tipologie di operatori e/o funzioni disponibili dipendono in larga misura dal modello di PLC e dal linguaggio di programmazione impiegato per la stesura del software.

# Funzionamento dei PLC

## Ciclo di scansione

Dal momento in cui il PLC viene messo in stato di RUN il suo funzionamento è di tipo ciclico sequenziale: ciò significa in pratica che esso esegue in modo ripetitivo le funzioni programmate fino a quando viene messo in stato di STOP, si toglie l'alimentazione o si verifica una situazione di anomalia e/o malfunzionamento che provoca l'arresto del ciclo.

Il ciclo di funzionamento viene chiamato ciclo di scansione, mentre il tempo impiegato a compierlo prende il nome di tempo di scansione.

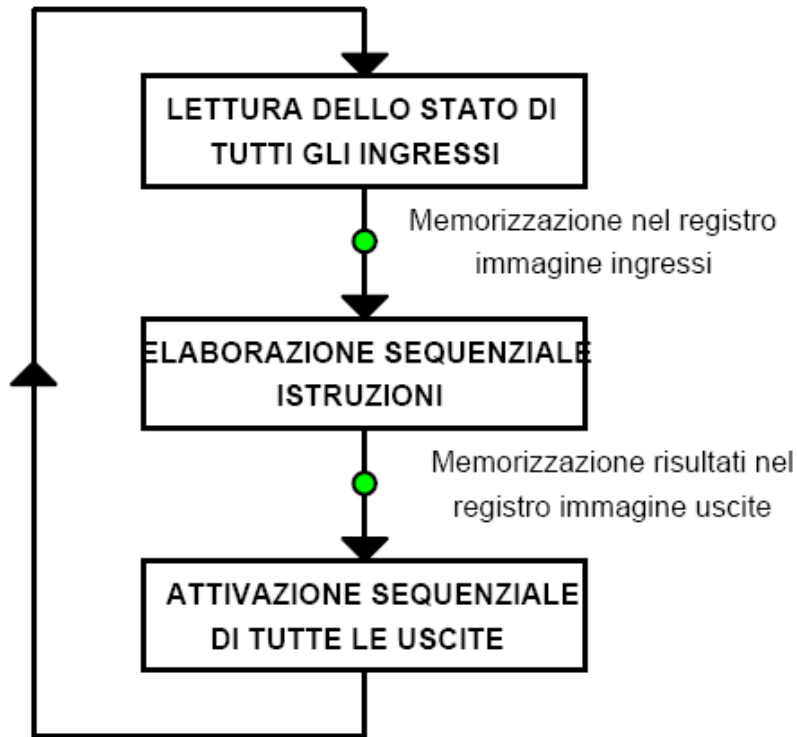
Ciascun PLC viene costruito per funzionare secondo una delle seguenti modalità di esecuzione del ciclo di scansione:

- Sincrona di ingresso e di uscita
- Sincrona di ingresso e asincrona di uscita
- Asincrona di ingresso e di uscita

# Ciclo di scansione

## Modalità sincrona di ingresso e di uscita

Le fasi attraverso le quali avviene questa modalità di funzionamento sono illustrate nella figura a fianco:



- Acquisizione dello stato dei moduli di ingresso, secondo una gerarchia che dipende dall'indirizzamento dei punti di input e memorizzazione dello stato logico di questi ultimi nel registro immagine degli ingressi;
- Esecuzione delle istruzioni del programma e memorizzazione dello stato delle uscite che a mano a mano si determinano nel registro immagine delle uscite, senza tuttavia attivare queste ultime;
- Al termine dell'esecuzione del programma avviene l'aggiornamento delle uscite, secondo una gerarchia che dipende dall'indirizzamento dei punti di output.

Terminato un ciclo di scansione si riprende da capo, continuando fino a che il PLC permane in stato di RUN.

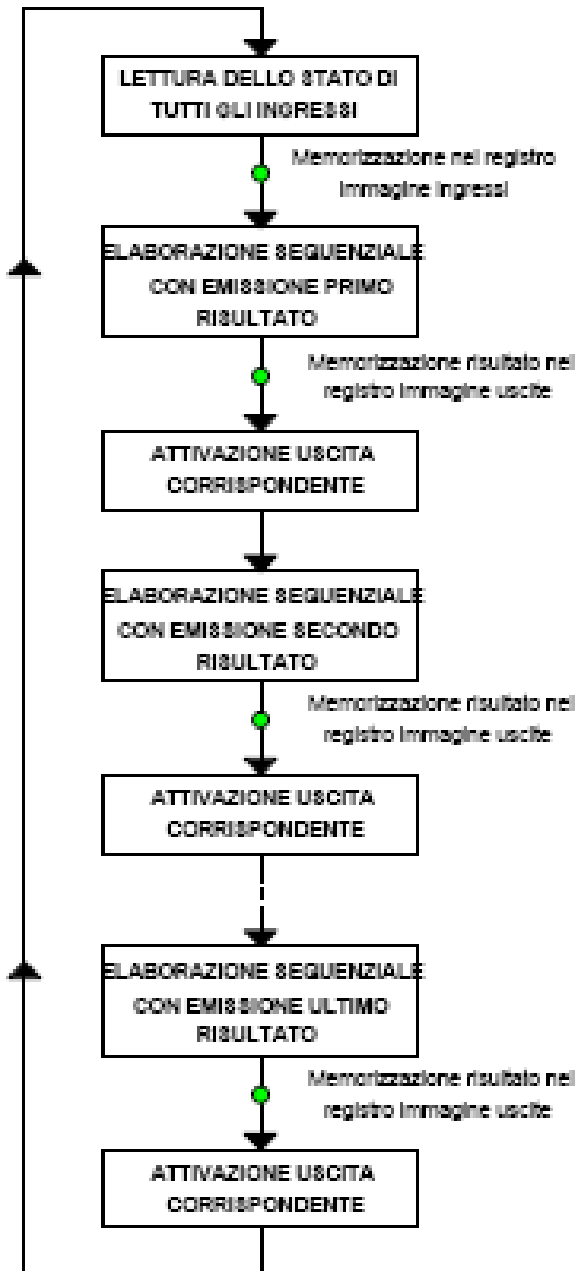
Se durante l'esecuzione delle istruzioni (fase che occupa la maggior parte del tempo di scansione) avviene una variazione dello stato degli ingressi essa non può essere acquisita dal PLC se non al ciclo successivo; il tempo di risposta del sistema risulta quindi sempre superiore al tempo di scansione

Questo inconveniente non rappresenta tuttavia un problema particolarmente grave in molte applicazioni, dato che il tempo di scansione è solitamente molto piccolo (da 5 a 45ms circa a seconda del numero di istruzioni presenti nel programma).

## Ciclo di scansione

### Modalità sincrona di ingresso e asincrona di uscita

Le fasi attraverso le quali avviene questa modalità di funzionamento sono illustrate nella figura a fianco:

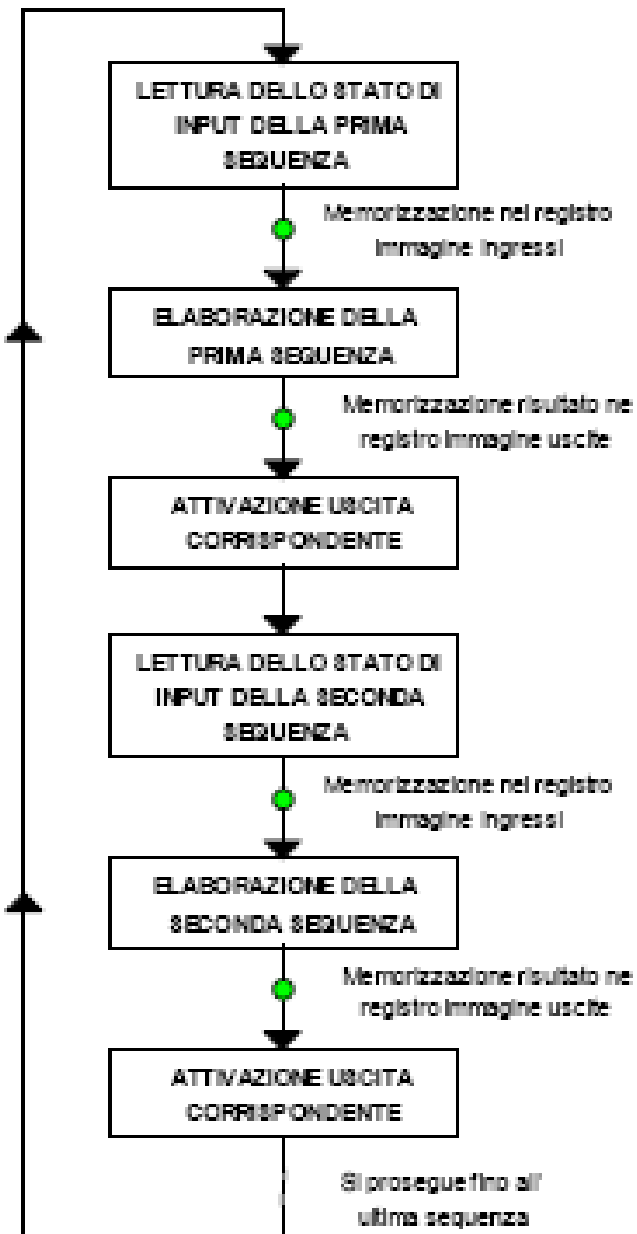


- Acquisizione dello stato dei moduli di ingresso, secondo una gerarchia che dipende dall'indirizzamento dei punti di input e memorizzazione dello stato logico di questi ultimi nel registro immagine degli ingressi, ottenendo in questo modo una lettura sincrona dello stato degli ingressi;
- Esecuzione delle istruzioni del programma e attivazione delle diverse uscite a mano a mano che il programma produce i risultati necessari allo scopo: si ottiene così un aggiornamento asincrono dello stato delle uscite.

Con questo tipo di scansione si ha un aggiornamento più rapido delle uscite rispetto al caso precedente, mentre per quanto riguarda lo stato degli ingressi il cambiamento può essere rilevato solamente al termine di ogni ciclo.

# Ciclo di scansione

## Modalità asincrona di ingresso e di uscita



Il modo di funzionamento è descritto nella figura a fianco; le operazioni relative agli ingressi e alle uscite sono del tutto asincrone:

- Gli ingressi vengono aggiornati ogni volta che il programma incontra un'istruzione che obbliga il PLC ad acquisirne lo stato;
- Se un bit del registro di stato delle uscite viene modificato durante l'elaborazione, lo stato logico risultante viene immediatamente inviato al corrispondente modulo di uscita.

In questo modo il funzionamento del PLC è del tutto simile a quello di un computer, con un tempo di risposta molto inferiore rispetto ai due casi precedenti.

# Programmazione dei PLC

## Linguaggi di programmazione

- La programmazione di un PLC consiste, come nel caso di un qualunque dispositivo programmabile, nel trasferire nella sua memoria una serie di istruzioni che hanno lo scopo di fare svolgere allo stesso i compiti di controllo ai quali è preposto;
- Per la programmazione dei PLC possono essere utilizzati diversi linguaggi concepiti, nella maggior parte dei casi, per applicazioni di tipo industriale;
- I linguaggi di programmazione dei PLC possono essere sostanzialmente suddivisi in due gruppi:
  - Linguaggi grafici: le istruzioni vengono date tramite simboli grafici che richiamano in qualche modo gli elementi propri di un impianto, come, ad esempio contatti e bobine di relè oppure blocchi funzionali come contatori, temporizzatori, funzioni logiche ecc.  
Tra i più utilizzati troviamo il linguaggio LADDER (Ladder Diagram – LD o linguaggio a contatti), il linguaggio a blocchi funzionali (Functional Block Diagram – FBD) e il Grafcet (Sequential Function Chart – SFC);
  - Linguaggi letterali: vengono detti anche linguaggi a lista di istruzioni ed utilizzano una serie di parole chiave che identificano le varie funzioni disponibili; questi linguaggi sono molto simili a quelli “ad alto livello” utilizzati ad esempio sui PC.

# Linguaggio LADDER

## Introduzione

- Il linguaggio LADDER è un linguaggio di tipo grafico ed è stato storicamente il primo utilizzato per la programmazione dei PLC, in quanto questi ultimi andavano inizialmente a sostituire i quadri elettrici a logica cablata che utilizzavano relè e/o altri componenti di tipo elettromeccanico;
- Il diagramma ladder, chiamato anche schema a contatti o diagramma a relè rimane attualmente il linguaggio di programmazione più diffuso, in quanto consente in modo relativamente semplice il passaggio da una rappresentazione grafica di tipo elettrico (come può essere ad esempio lo schema funzionale dei circuiti di comando e/o segnalazione) al diagramma ladder che rappresenta il programma da inserire nel PLC;
- Le attuali versioni del linguaggio ladder sono dotate di istruzioni molto evolute come temporizzatori, contatori, operazioni matematiche, di manipolazione e trasferimento dati ed altre ancora.

# Esempio di programma in linguaggio LADDER

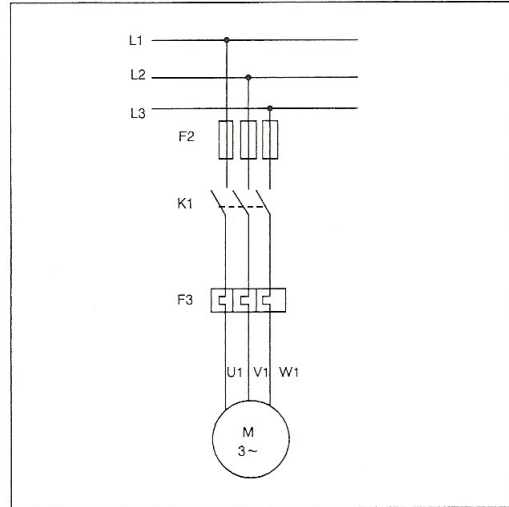
## Comando “marcia – arresto” di un motore trifase - 1

### Schemi tratti da documentazione SIEMENS

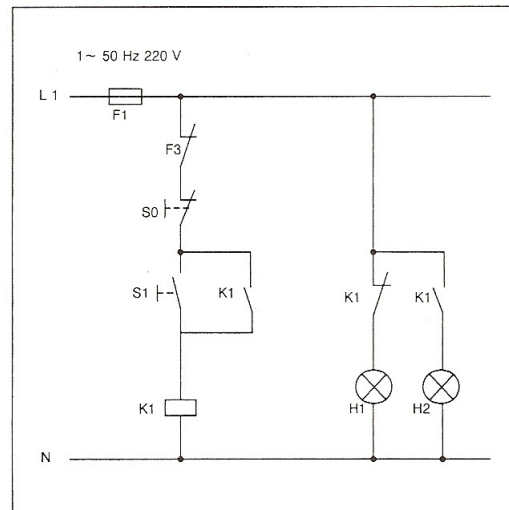
Comando “marcia/arresto” di un motore trifase

Esempio 1

Circuito di potenza



Circuito di comando



La figura a fianco mostra i circuiti di potenza e di comando per il problema in esame:

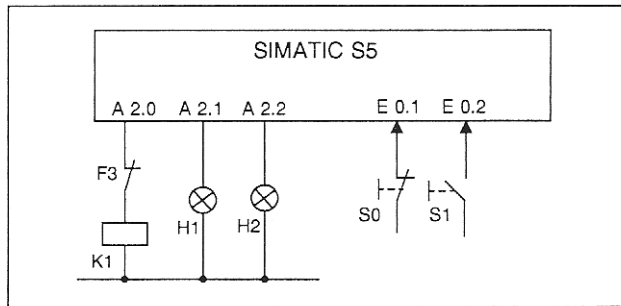
- Un motore elettrico trifase viene avviato tramite il teleruttore K1 premendo il pulsante S1 (Marcia) e fermato premendo il pulsante S0 (Arresto);
- Il teleruttore K1 viene comandato dai pulsanti S0 e S1 e provvede ad inserire o disinserire il motore dalla linea trifase di alimentazione;
- Le lampade di segnalazione H1 e H2 mostrano, rispettivamente, lo stato di marcia o arresto del motore.

Si tratta del classico “comando di un relè con autoritenuta”, tipico di questo tipo di azionamenti.

# Comando “marcia – arresto” di un motore trifase – 2

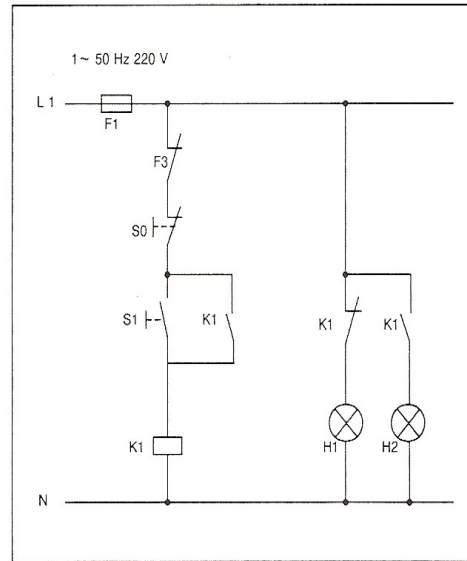
Schemi tratti da documentazione SIEMENS

1

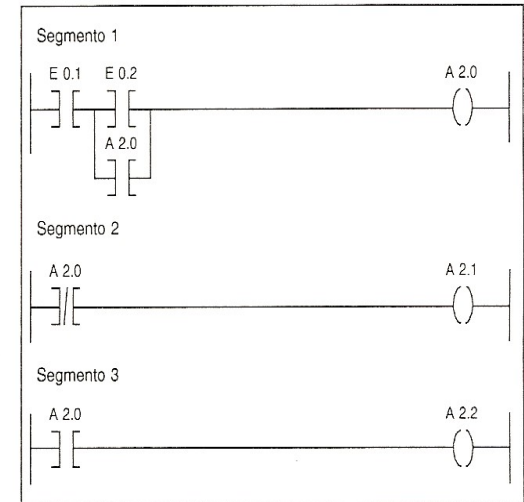


2

| Simbolo | Operando | Descrizione                      |
|---------|----------|----------------------------------|
| S0      | E 0.1    | OFF (contatto norm. chiuso N.C.) |
| S1      | E 0.2    | ON (contatto N.A.)               |
| K1      | A 2.0    | Contattore del motore            |
| H1      | A 2.1    | Segnalazione motore fermo        |
| H2      | A 2.2    | Segnalazione motore in marcia    |



3



4

Nelle figure sono rappresentati:

- (1) Lo schema di collegamento dei vari dispositivi al PLC utilizzato (SIEMENS SIMATIC S5);
- (2) La lista di attribuzione (detta anche “di assegnazione I/O”) degli ingressi e delle uscite del PLC ai vari dispositivi: (ingressi: pulsanti S0 e S1; uscite: bobina di comando del teleruttore K1, lampade di segnalazione H1 e H2);
- (3) Lo schema funzionale del circuito di comando;
- (4) La realizzazione dello stesso tramite linguaggio ladder: nel segmento 1 si ha la programmazione del comando di K1 con autoritenuta e nei segmenti 2 e 3 si programmano, rispettivamente, l'accensione delle lampade H1 (motore fermo) e H2 (motore in funzione).

# Fasi principali per la progettazione di un programma per PLC

- **Definizione del problema**

Questa prima fase prevede lo studio delle caratteristiche del sistema da controllare, che può essere costituito da una o più macchine o da un intero processo industriale; questo studio è fondamentale per la scelta del PLC più adatto all'applicazione in esame.

Sempre in questa prima fase è possibile predisporre un diagramma di flusso generale del programma: quest'ultimo, se ben realizzato, potrà costituire un ottimo ausilio per la successiva stesura del codice da implementare sul PLC.

- **Assegnazione ingressi/uscite (assegnazione I/O)**

Ogni PLC ha un proprio codice di identificazione dei vari elementi funzionali; in questa seconda fase si abbinano tali codici ai dispositivi di I/O esterni da controllare (pulsanti, fincorsa, relè, lampade di segnalazione ecc). L'operazione di assegnazione degli I/O deve essere fatta prima della scrittura del programma, in quanto gran parte delle istruzioni hanno un preciso riferimento ai codici di identificazione degli elementi funzionali; per facilitare questa operazione si possono utilizzare moduli prestampati realizzati appositamente per il modello di PLC prescelto.

- **Scrittura del programma**

In questa terza fase si imposta la schematizzazione del ciclo operativo del sistema, formulata in base a linguaggio di programmazione prescelto: facendo riferimento al linguaggio ladder risultano molto utili in questa fase gli schemi elettrici funzionali e/o gli schemi logici di funzionamento; anche in questo caso è possibile utilizzare moduli prestampati (fogli di programmazione) oppure, come ormai quasi sempre accade, utilizzare un apposito software di programmazione su PC (p. es. "CX-ONE Supervisor" in dotazione ai PLC Omron) che in genere prevede anche la possibilità di simulare il comportamento del programma e trasferirlo al PLC tramite porta seriale e/o USB.

- **Trasferimento del programma, debug e documentazione dello stesso**

In questa quarta fase il programma viene caricato sul PLC e successivamente testato (ricorrendo spesso alla possibilità di forzare artificialmente lo stato degli I/O), apportando le necessarie correzioni per arrivare alla versione definitiva, rilasciata all'utente assieme ad un'adeguata documentazione d'uso e manutenzione.

# Bibliografia - Sitografia

- Massimo Barezzi  
*PLC – Controllori logici programmabili*  
*Programmazione, applicazioni ed esercitazioni*  
Editrice San Marco
- Giorgio Portaluri – Enea Bove  
*Tecnologie e disegno per la progettazione elettronica – Vol. 3*  
Editrice Tramontana
- De Santis, Cacciaglia, Saggese  
*Corso di sistemi – Vol. 2*  
Editrice Calderini
- <http://www.plcforum.it/>  
Portale dedicato al mondo dei PLC; della sezione *didattica* sono stati utilizzati i seguenti materiali:
  - Claudio Ferrari: *Dispense sui PLC* (formato PDF);
  - Claudio Ferrari: *Corso introduttivo ai PLC* (slides POWER POINT);