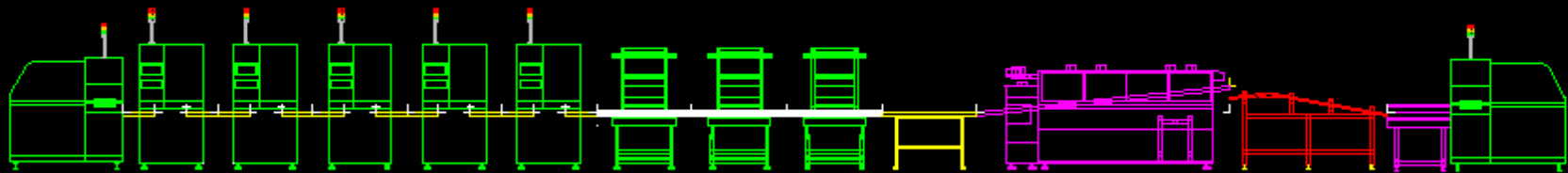


Università degli Studi di Siena – Sede di Arezzo  
Corso di Laurea in Ingegneria dell'Automazione

# Automazione Industriale

A.A. 2004 - 2005

Ing. Daniele Rapisarda



## **4° PARTE**

---

# Elementi introduttivi ai PLC

# PLC

**P - Programmable**

**L - Logic**

**C - Controller**

- semplice ed economico microcomputer dedicato all'ambiente industriale, per il quale è dotato delle opportune interfacce.

# PLC

Secondo il **NEMA** (National Electrical Manufacturers Association), un controllore a logica programmabile è una apparecchiatura di controllo avente le seguenti caratteristiche:

- apparecchiatura elettronica a microprocessore operante in modo digitale
- usa moduli di I/O per recepire segnali da dispositivi sensoriali e per comandare dispositivi di attuazione
- ha una memoria di lavoro (di tipo RAM) per immagazzinare internamente tali "dati" e le istruzioni per specifiche funzioni, come funzioni logiche, sequenziali, di temporizzazione, di conteggio o aritmetiche
- esegue le funzioni assegnate ad un automatismo, espresse attraverso un apposito programma di istruzioni.

# PLC

- Programmable Logic Controller (Controllore a Logica Programmabile)
- Apparecchiatura elettronica programmabile per il controllo di macchine / processi industriali
- Nasce come elemento sostitutivo della logica cablata e dei quadri di controllo a relè

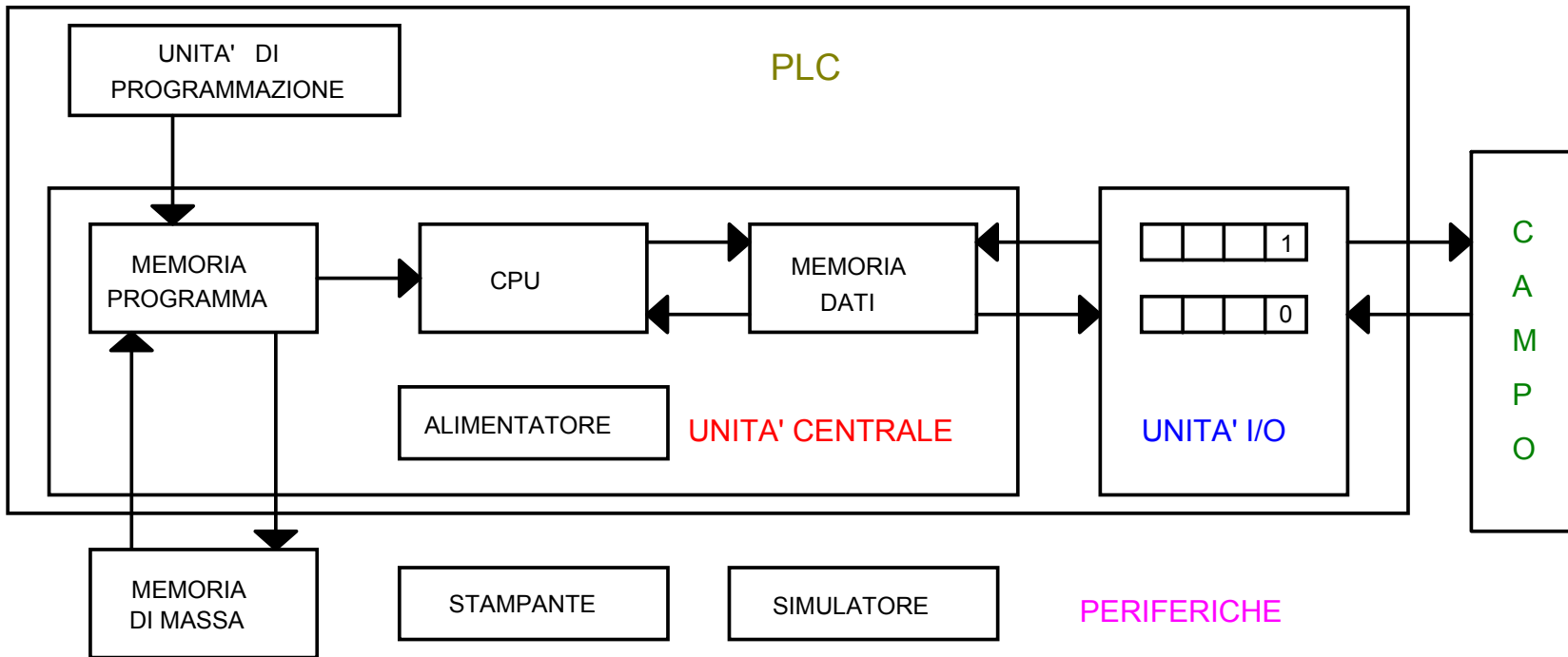


- Si qualifica in breve tempo come elemento insostituibile nell'automazione di fabbrica, ovunque sia necessario un controllo elettrico di una macchina

## STRUTTURA DI UN PLC

- ALIMENTATORE
  - UNITÀ CENTRALE
  - MEMORIA DATI E MEMORIA PROGRAMMI
  - UNITÀ DI INPUT/OUTPUT
  - PERIFERICHE
- 
- Il gruppo di alimentazione fornisce la tensione e la corrente di lavoro ai dispositivi elettronici. Sono disponibili alimentatori per il collegamento diretto alla rete elettrica (110/220 VAC) oppure alimentatori in corrente continua (24 VDC) particolarmente indicati quando è necessario realizzare un quadro elettrico collegabile a batterie tampone per assicurare il funzionamento anche in condizioni di mancanza della tensione di rete.

# STRUTTURA DI UN PLC



## CPU

- È quel dispositivo che determina l'esecuzione del programma, dei calcoli e di tutte le elaborazioni logiche
- Interagisce con la memoria, i moduli di I/O e le periferiche
- La sua potenza si esprime attraverso il *set delle istruzioni* e la *velocità di elaborazione*
- La CPU è il cuore del PLC, perché contiene i circuiti che gestiscono tutte le attività del controllore e, in particolare, esegue la scansione del programma, la lettura degli ingressi ed il pilotaggio delle uscite. La CPU è in genere dotata di LED di segnalazione che indicano lo stato di funzionamento: presenza dell'alimentazione, modalità operativa, condizioni di allarme e di errore.



## Parametri che caratterizzano una CPU

I parametri che caratterizzano la CPU sono:

- Massimo numero di I/O indirizzabili
- Velocità di scansione per 1000 istruzioni (1Kbyte)
- Numero e tipologia delle istruzioni eseguibili
- Massima dimensione del programma (espressa in Kbyte)
- Tipologia di moduli I/O e speciali collegabili
- Possibilità di interfacciamento con computer e/o altri PLC

# MEMORIA

Esistono diversi dispositivi di memoria:

- **MEMORIA DI SISTEMA**
- **MEMORIA DI PROGRAMMA**
- **MEMORIA DATI**

## MEMORIA DI SISTEMA

- Contiene il sistema operativo (firmware) del PLC, costituito da:
  - routine di autotest iniziale
  - dati del setup
  - librerie

## MEMORIA DI PROGRAMMA

- Contiene la sequenza di istruzioni (programma utente) che verrà eseguita dalla CPU

Esistono diverse possibilità:

- **RAM:** (per sviluppo e collaudo) - Consente rapide operazioni di lettura/scrittura. Deve essere sempre alimentata. Una batteria "tampono" mantiene i dati nel caso di caduta della tensione di alimentazione
- **EPROM:** (per programma definitivo) - Non richiede alimentazione, ma per poter essere "scritta" richiede un particolare dispositivo (programmatore di EPROM). La cancellazione avviene tramite raggi ultravioletti
- **EEPROM:** (sia per fase di sviluppo che per versione definitiva)  
Riassume i vantaggi di Ram ed EPROM

## MEMORIA DATI

- Contiene le informazioni relative alle varie aree dati interne e di I/O
- Poiché, in funzione delle elaborazioni del programma, è necessario effettuare sulle aree dati veloci operazioni di lettura e di scrittura, è possibile utilizzare soltanto memorie di tipo RAM

## BATTERIA

- Il mantenimento della memoria dati anche a fronte di cadute di alimentazione, viene assicurato da una batteria tampone
- Questa batteria alimenta anche l'eventuale RAM utilizzata per la memoria programmi
- La batteria ha una durata nominale di circa 5 anni (in relazione all' uso e all' ambiente)
- La fase di scaricamento della batteria viene segnalata in modo automatico dal PLC
- Uno scaricamento completo determina la perdita di dati e programma (se questo è in RAM)

## MODULI DI / DO

- Permettono il collegamento del PLC al mondo esterno
- Sono disponibili:
  - Moduli di INGRESSO DIGITALE (AC,DC,AC/DC)
  - Moduli di USCITA DIGITALE (Relè, Transistor, Triac)
  - Moduli di comunicazione
  - Moduli speciali (AD-DA, Contatori veloci, Controllo assi, PID, ...)

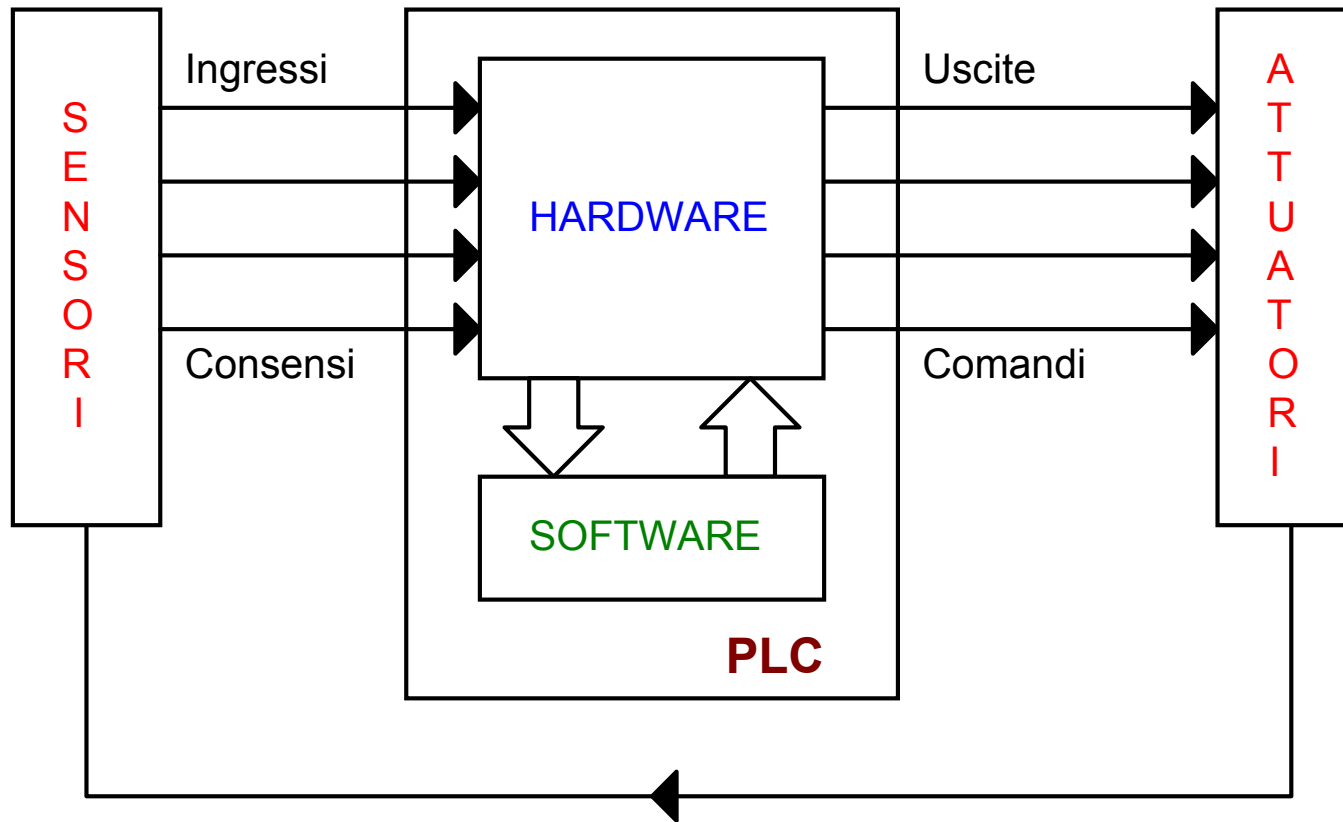
## PERIFERICHE

- Permettono il "colloquio" tra l'operatore (programmatore) ed il PLC
- Consolle di programmazione
- Consolle di programmazione grafica
- Interfaccia per personal computer
- Interfaccia stampante
- Programmatore di EPROM

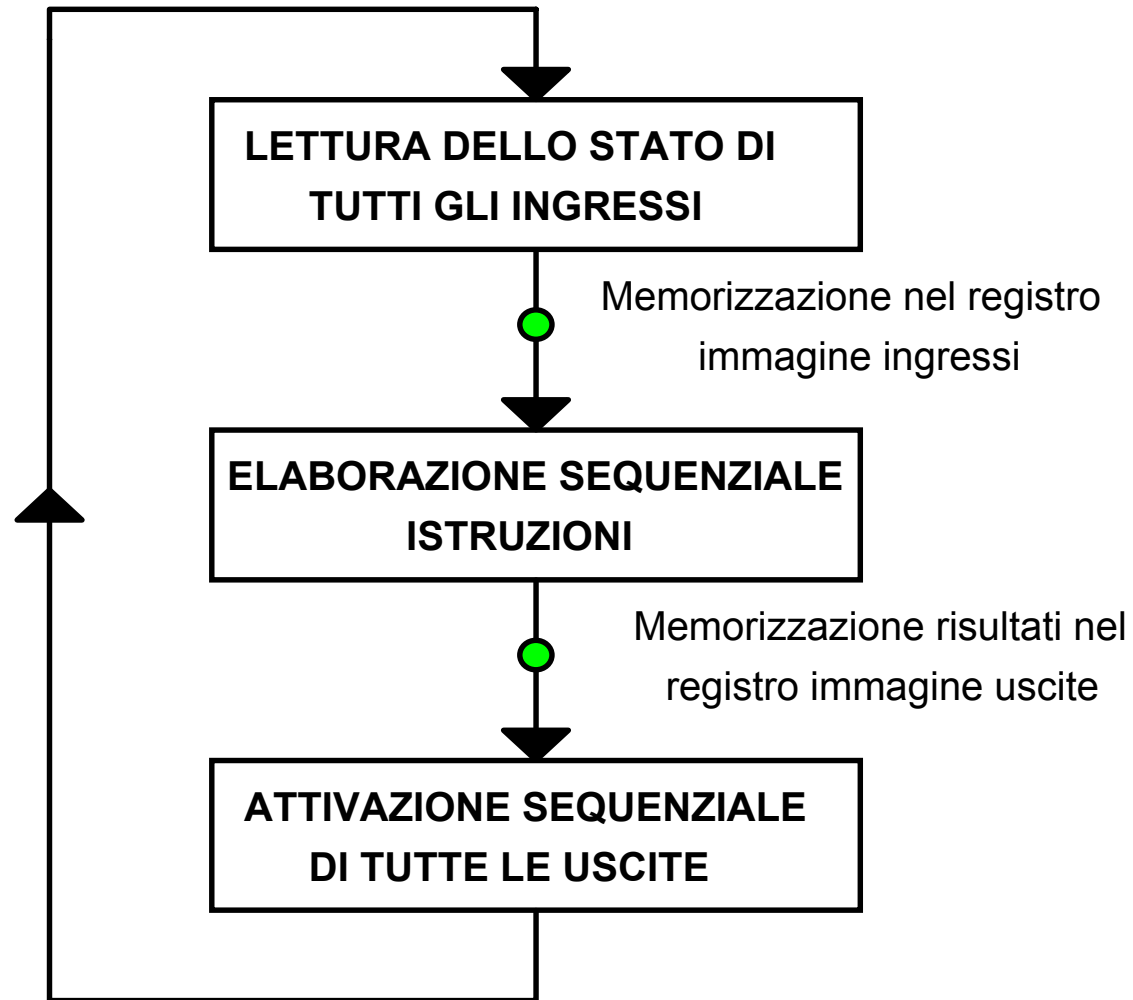


# PRINCIPIO DI FUNZIONAMENTO

- Un PLC è un sistema di comando a cui si collegano quindi sensori ed attuatori per l'elaborazione delle relative informazioni



## Scansione sincrona di ingresso e di uscita



## Scansione sincrona di ingresso e di uscita



### MODALITÀ CICLICA

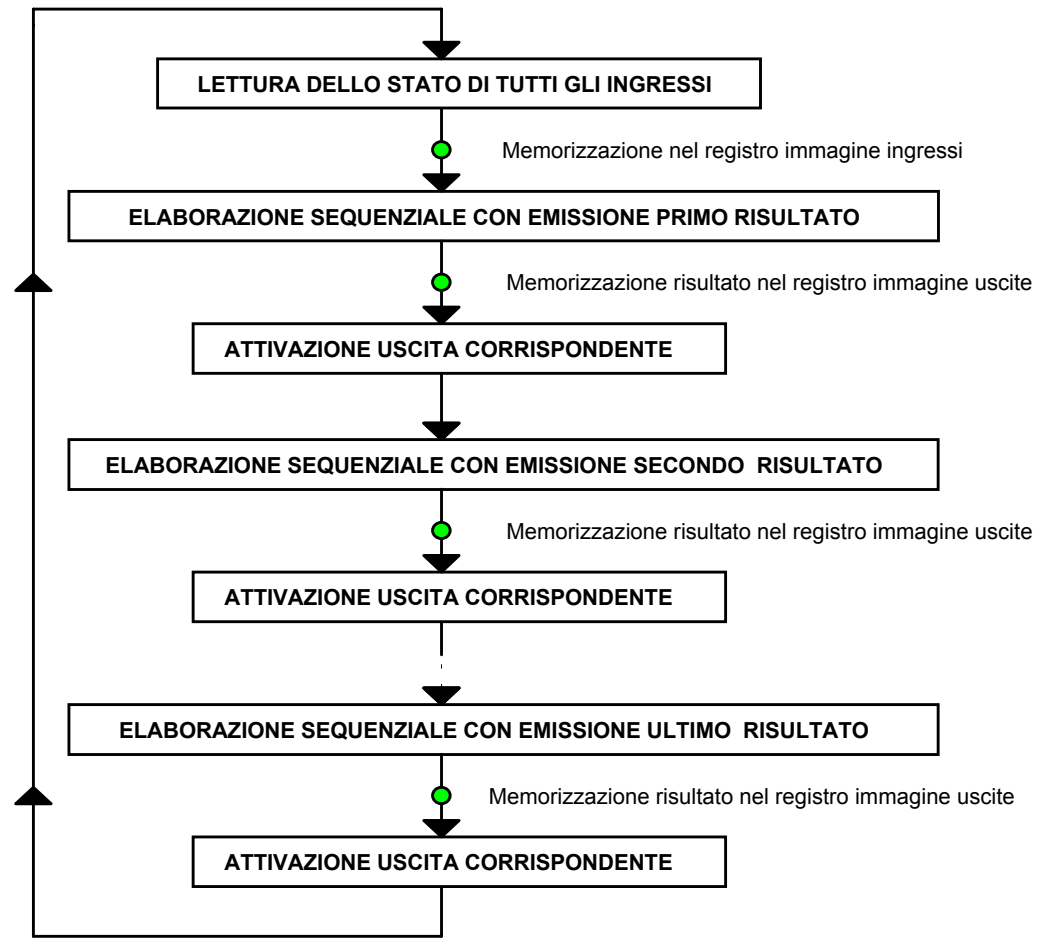
#### IN SEQUENZA

- LETTURA INGRESSI
- ESECUZIONE PROGRAMMA
- TEMPO DI ATTESA
- AGGIORNAMENTO USCITE
- GESTIONE DELLA RETE

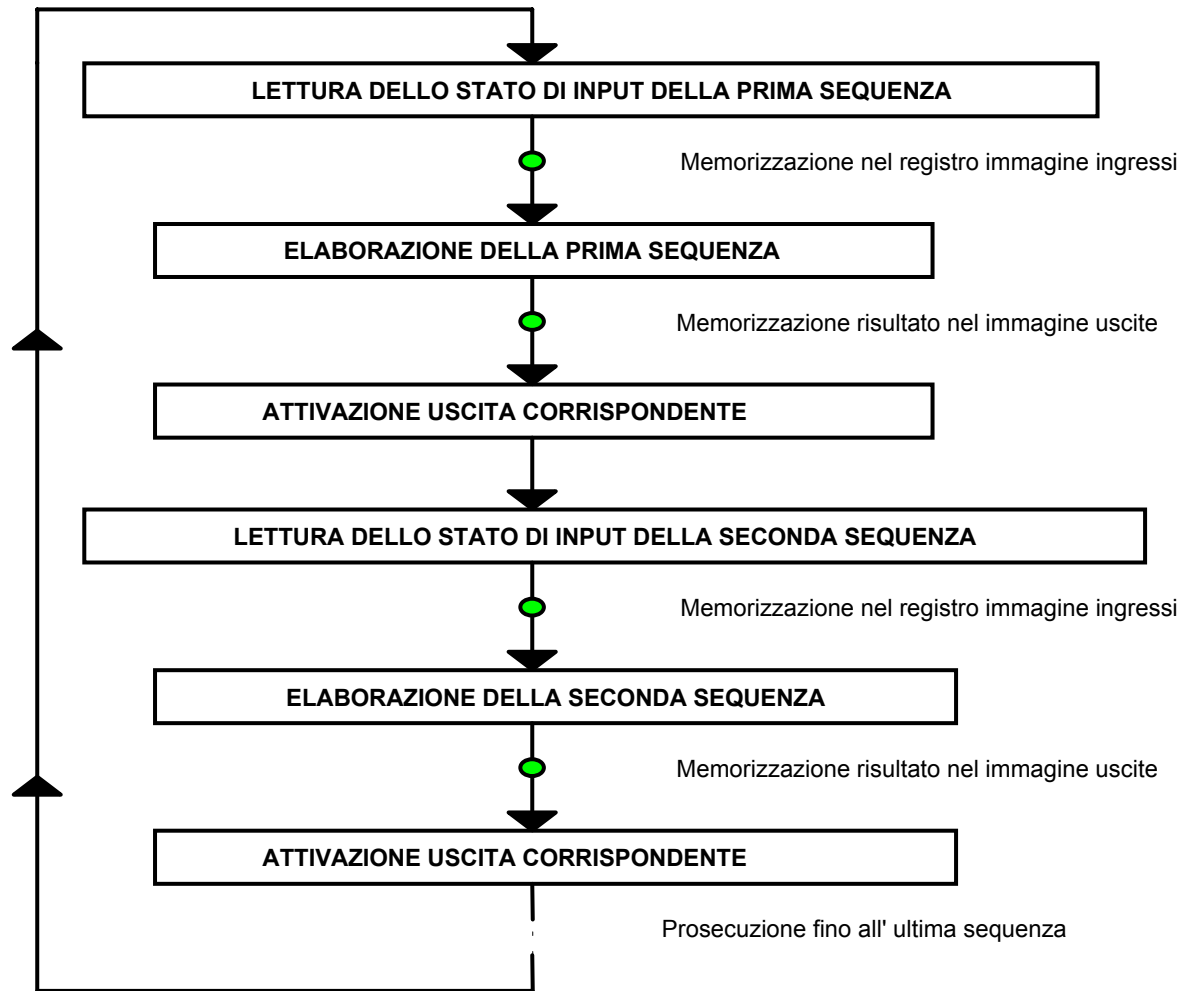
### TEMPO DI SCANSIONE:

INTERVALLO DI TEMPO  
NECESSARIO PER ESEGUIRE  
UN CICLO DEL PROGRAMMA

# Scansione sincrona di ingresso e asincrona di uscita



# Scansione asincrona di ingresso e di uscita



## DOVE SI USANO I PLC

*In tutte quelle applicazioni dove ...*

- Sono richiesti più di 10 I/O
- Si deve garantire un prodotto affidabile
- È richiesta una apparecchiatura con caratteristiche industriali
- Si devono prevedere espansioni e modifiche nella logica di controllo
- Sono richieste funzioni sofisticate come:
  - √ Connessioni a computer, terminali, stampanti, . . .
  - √ Elaborazioni matematiche
  - √ Posizionamenti
  - √ Regolazioni PID

## APPLICAZIONI TIPICHE DEI PLC

- MACCHINE UTENSILI
- MACCHINE PER LO STAMPAGGIO
- MACCHINE PER IMBALLAGGIO
- MACCHINE PER IL CONFEZIONAMENTO
- ROBOT / MONTAGGIO
- REGOLAZIONE PROCESSI CONTINUI
- MACCHINE TESSILI
- SISTEMI DI MOVIMENTAZIONE/TRASPORTO
- CONTROLLO ACCESSI

## **PROGRAMMAZIONE DEI PLC**

I circuiti a logica interna dei PLC sostituiscono i relè, i temporizzatori, i contatori e gli altri dispositivi discreti normalmente utilizzati per costruire sistemi di controllo elettromeccanici.

L'effettivo funzionamento della macchina avviene come se tali componenti fossero presenti, ma con caratteristiche di flessibilità ed affidabilità notevolmente superiori. Nella programmazione con i diagrammi a relè restano tuttavia in uso i simboli dei quadri elettromeccanici.



## PROGRAMMAZIONE DEI PLC

### *METODI DI PROGRAMMAZIONE:*

- DIAGRAMMI A RELE' (reti ladder)
- LINGUAGGI LOGICI
- LINGUAGGI INFORMATICI
- GRAFCET (programmazione grafica)

## DIAGRAMMI A RELE' (KOP)

- Primo linguaggio sviluppato per i PLC (*diagrammi a relè o reti ladder o schemi a contatti o KOP*).
- I PLC si sostituiscono ai quadri di comando a relè.
- Macchine utilizzabili esattamente come se contenessero al loro interno dei componenti quali relè, temporizzatori e contatori.
- Elettromeccanica "virtuale".
- Linguaggio che riporta tutti i simboli a cui i tecnici sono abituati: contatti normalmente aperti o chiusi, bobine, connessioni in serie (funzioni AND) o in parallelo (funzioni OR), ecc.

## LINGUAGGI LOGICI

- Linguaggio per "elettronici" o per "logici".
- Simbologia utilizzata nella logica booleana.
- Si stabiliscono i legami richiesti fra i segnali di ingresso e le variabili di uscita (tabelle della verità).
- Sintesi del sistema (mappe di Karnaugh, diagrammi di Quine-McCluskey, ecc...).
- Risultato finale: espressioni logiche basate su operatori logici (AND, OR, NOT, ecc...).
- Gli operatori logici sono specificati nel PLC mediante codici corrispondenti

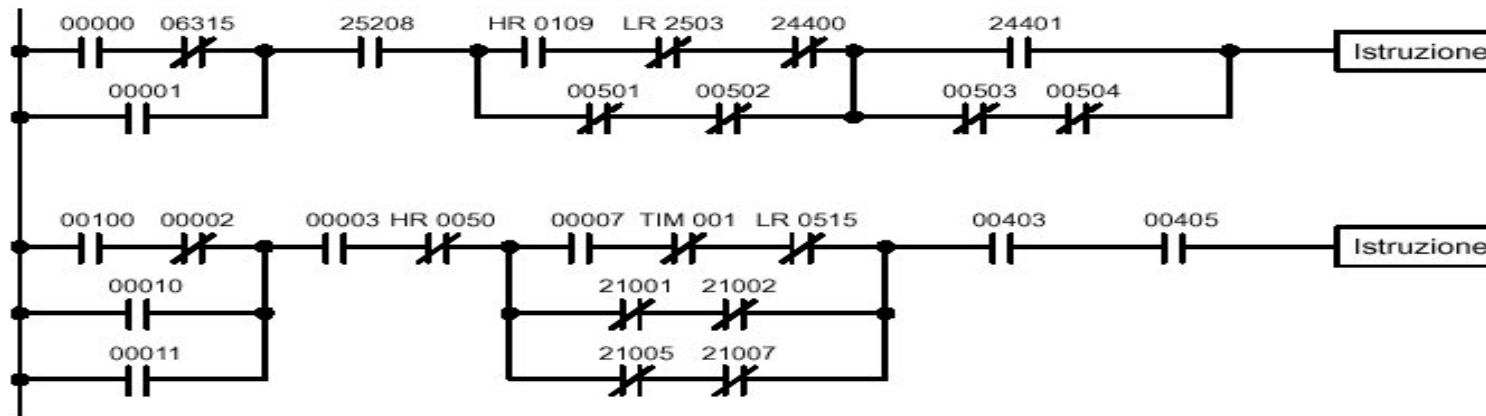
## LINGUAGGI INFORMATICI

- Versioni semplificate del BASIC o linguaggi sviluppati ad hoc, chiamati "colloquiali" perché basati su comandi espressi con parole di facile comprensione (SE...ALLORA....., ALTRIMENTI....).
- Aggiungono alle funzioni di controllo utili funzioni di calcolo.
- Facilità di costruire programmi modulari, quindi più flessibili e riutilizzabili.

# DIAGRAMMI A RELE' (KOP)

Un *diagramma a relè* consiste di una barra verticale posta sul lato sinistro (chiamata "bus") e di un insieme di diramazioni orizzontali (chiamate "righe circuitali") che partono dalla linea verticale.

Sulle righe circuitali vengono posti i vari contatti che si collegano alla parte destra. La combinazione logica di questi contatti determina quando e come le istruzioni poste a destra devono essere eseguite.



*Esempio di riga circuitale*

*Le righe circuitali possono avere a loro volta diverse diramazioni che poi si collegano nuovamente.*

*Ogni coppia di tratti verticali è chiamata **contatto**.*

*I contatti non dotati di un tratto diagonale, vengono detti contatti **aperti** e corrispondono alle istruzioni **LOAD, AND o OR**.*

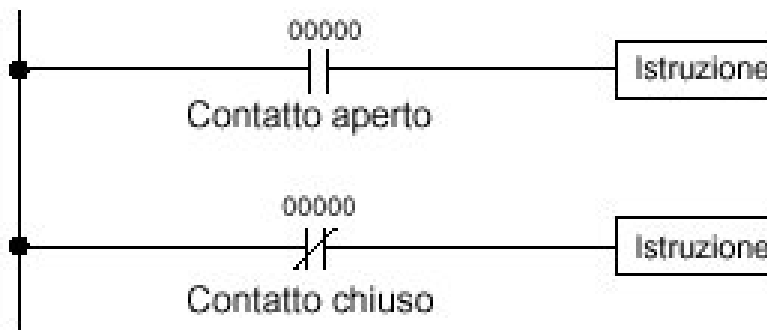
*I contatti che prevedono dei tratti diagonali vengono chiamati contatti **chiusi** e corrispondono alle istruzioni **LOAD NOT, AND NOT o OR NOT**.*

*Il numero in corrispondenza di ciascun contatto indica **l'operando** (in questo caso un bit) dell'istruzione.*

*Lo stato del bit associato a ciascun contatto determina la **condizione di esecuzione** per l'istruzione seguente.*

## TERMINI DI BASE

- Ogni contatto in un diagramma a relè può essere ON o OFF in base dello stato del bit associato.
- Nel caso di **condizione normale**:  
stato ON se bit ON – stato OFF se bit OFF.
- Nel caso di **condizione negata**:  
stato ON se bit OFF – stato OFF se bit ON.
- In generale viene usata una condizione normale quando si vuole che accada qualcosa quando il bit è ON, mentre si usa una condizione negata quando si vuole che accada qualcosa quando il bit è OFF.



L'istruzione viene eseguita quando il bit IR 00000 è ON.

L'istruzione viene eseguita quando il bit IR 00000 è OFF.

## CONDIZIONI PER L'ESECUZIONE

- In un programma in diagramma a relè, la combinazione logica di condizioni ON e OFF prima di una istruzione determina la condizione necessaria affinché l'istruzione venga eseguita.
- Questa condizione, che sia ON o OFF, viene chiamata **condizione per l'esecuzione** dell'istruzione.
- Tutte le istruzioni, a parte LOAD e LOAD NOT, richiedono una condizione per l'esecuzione.



## BIT DI OPERANDO

- Gli operandi possono essere, per ciascuna istruzione, qualsiasi bit delle aree IR, SR, HR, AR, LR o TC.
- Ciò significa che un contatto, in un diagramma a relè, può essere dato dallo stato degli I/O, dei flag, dei bit interni, dei temporizzatori/contatori, ecc.
- Le istruzioni LOAD e OUT possono anche utilizzare i relè temporanei TR, ma questi vengono usati solo in casi speciali.

## BLOCCHI LOGICI

- Il modo in cui i contatti influenzano le istruzioni è determinato dalle relazioni logiche tra i vari contatti presenti all'interno delle singole righe circuitali.
- Ciascun gruppo di contatti che definisce un certo risultato logico, viene detto **blocco logico**.
- Sebbene i diagrammi a relè possano venire scritti senza analizzare i singoli blocchi logici, è necessario comprenderne bene il significato per realizzare una efficace programmazione, quando si programma in lista istruzioni.

## **BLOCCO DI ISTRUZIONI**

- Un blocco di istruzioni è l'insieme di tutte quelle istruzioni che sono interconnesse all'interno di un diagramma a relè.
- L'inizio e la fine di un blocco di istruzioni sono identificabili con due punti consecutivi del diagramma a relè in cui può essere tracciata una linea orizzontale senza che questa intersechi alcuna linea verticale.
- Tutte le istruzioni all'interno di tali punti costituiscono il blocco di istruzioni.

## CREAZIONE DI UN PROGRAMMA

Si devono seguire **otto fasi** fondamentali:

- Determinare che cosa il sistema di controllo deve fare e secondo quale sequenza.
- Selezionare i dispositivi di ingresso e di uscita del PLC.
- Disegnare uno schema di controllo, utilizzando i simboli dei diagrammi a relè. Come in un normale schema funzionale, è necessario riportare, nella corretta sequenza, le funzioni richieste e le loro correlazioni.
- Codificare i simboli dello schema in un formato adatto per l'inserimento nella CPU tramite la consolle di programmazione, in base al tipo di PLC scelto ed alle funzioni offerte dalla consolle stessa.
- Trasferire lo schema codificato (programma) nella CPU.
- Attivare il programma, ossia eseguirlo.
- Verificare se il programma contiene degli errori (debug)
- Correggere gli errori e memorizzare il programma finito.

## PRIMA FASE

Determinare che cosa il sistema di controllo deve fare e secondo quale sequenza (**SPECIFICHE DEL SISTEMA**)

- Valutare il numero ed il tipo di I/O richiesti dall'applicazione.
- Identificazione di tutti i dispositivi che devono inviare un segnale di ingresso al PLC (sensori, pulsanti, periferiche, ecc...) o che da questo devono ricevere un segnale (attuatori, servocomandi, sistemi di visualizzazione, ecc...) decidendo, in base alla natura di ciascuno di essi, quanti punti elementari (bit) sono necessari.
- Il numero dei punti di ingresso, sommato al numero dei punti di uscita, fornisce il numero complessivo dei punti I/O, che non deve mai superare la capacità massima del controllore (riportata fra i dati caratteristici).
- Preparare una tabella che mostri la corrispondenza tra i bit di I/O ed i dispositivi di I/O.

## SECONDA FASE

Selezionare i dispositivi di ingresso e di uscita del PLC.

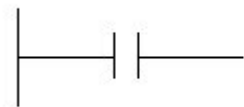
- I **canali I/O** identificano i singoli punti.
- Ciascun canale comprende normalmente 16 punti ed ogni punto è identificato mediante un numero di quattro cifre.
- Le due cifre a sinistra identificano il canale, mentre le due cifre a destra identificano il punto all'interno del canale.
- 0000 identifica quindi il primo punto del primo canale, mentre 0104 identifica il quinto punto (04) del secondo canale (01).
- Oltre ai punti di I/O, occorre assegnare i **relè ausiliari interni** del PLC. Si tratta di relè che non controllano direttamente dei dispositivi esterni, ma sono usati come aree di memorizzazione o elaborazione dei dati, per controllare altri relè, temporizzatori, e contatori.

## SECONDA FASE

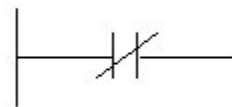
- Funzionalmente, I relè ausiliari interni equivalgono quindi ai relè interni dei quadri elettromeccanici. Si parla quindi, in genere, di **uscite interne**.
- Anche ai **temporizzatori** ed ai **contatori** devono essere assegnati dei numeri di identificazione, evitando di dare lo stesso numero ad un contatore e ad un temporizzatore. Ad esempio, non vi possono essere contemporaneamente un temporizzatore 01 ed un contatore 01.
- Dopo aver reso identificabili mediante indirizzi tutti gli elementi di ingresso e uscita e gli indirizzi interni che faranno parte dello schema di controllo, si può passare alla terza fase.

## TERZA FASE

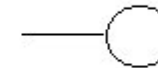
- Un programma scritto in un diagramma a relè è formato da una serie di righe circuitali, delimitate sul lato sinistro da una barra verticale.



Contatto  
normalmente aperto



Contatto  
normalmente chiuso



Uscita

Simboli fondamentali dei diagrammi a relè

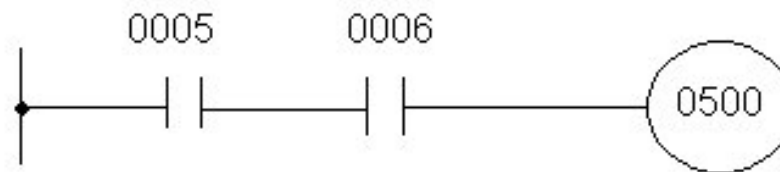
E' necessario ricordare che:

- ogni riga deve terminare con la bobina di un relè, di un temporizzatore/contatore o con un blocco funzionale (istruzione speciale);
- a differenza degli schemi elettrici, nei diagrammi a relè non è sempre necessario disegnare la barra destra;
- il numero dei contatti in serie o in parallelo utilizzabili su una riga circuitale è spesso limitato.



## QUARTA FASE

Trasformare funzioni di controllo espresse sotto forma di diagramma a relè nel linguaggio utilizzabile dal PLC



Esempio di riga circuitale

- Gli indirizzi sono delle posizioni di memoria nelle quali si possono registrare istruzioni o dati.
- Le istruzioni servono per spiegare al PLC che cosa deve fare, utilizzando i dati da cui sono seguite

## AREE DI MEMORIA

I dati e i flag che possono essere usati con ciascuna istruzione, sono indicati da sigle. Per determinarli in modo univoco ci si deve riferire al manuale specifico del PLC interessato

### Area dati Descrizione

- IR** Bit che possono essere assegnati ai terminali di I/O esterni, relè interni o relè speciali
- SR** Bit che svolgono funzioni specifiche come bit di controllo e flag.
- HR** Bit che memorizzano i dati e mantengono lo stato ON/OFF quando viene disattivata l'alimentazione.
- AR** Bit che svolgono funzioni specifiche come bit di controllo e flag.
- LR** Utilizzata per le connessioni di dati 1 a 1 con altri PC.
- TC** Vengono utilizzati gli stessi numeri per temporizzatori e contatori.

## Area IR - Bit di I/O

I bit dell'area IR da IR 00000 a IR 01915 vengono assegnati ai terminali del modulo CPU e dei moduli I/O. Questi bit riflettono lo stato ON/OFF dei segnali di ingresso e uscita. I bit di ingresso iniziano da IR 00000, mentre quelli di uscita da IR 01000.

La seguente tabella indica i bit IR assegnati ai terminali di I/O dei moduli CPU del PLC Omron CPM1

Modulo CPU CPM1	I/O	Terminali modulo CPU	Terminali unità I/O
CPM1-10CDR-□	Ingressi	6 punti: 00000 a 00005	12 punti: 00100 a 00111
	Uscite	4 punti: 01000 a 01003	8 punti: 01100 a 01107
CPM1-20CDR-□	Ingressi	12 punti: 00000 a 00011	12 punti: 00100 a 00111
	Uscite	8 punti: 01000 a 01007	8 punti: 01100 a 01107
CPM1-30CDR-□	Ingressi	18 punti: 00000 a 00011, 00100 a 00105	12 punti: 00200 a 00211
	Uscite	12 punti: 01000 a 01007, 01100 a 01103	8 punti: 01200 a 01207
CPM1-30CDR-□ -V1	Ingressi	18 punti: 00000 a 00011, 00100 a 00105	36 punti: 00200 a 00211 00300 a 00311 00400 a 00411
	Uscite	12 punti: 01000 a 01007, 01100 a 01103	24 punti: 01200 a 01207 01300 a 01407 01400 a 01407

### Area SR – Flag

Questi bit vengono utilizzati soprattutto come flag relativi al funzionamento del PLC oppure come contenitori dei valori impostati e correnti per le diverse funzioni. I canali da SR 244 a SR 247 possono essere utilizzati come bit di lavoro quando gli **interrupt** di ingresso non vengono impiegati in modalità contatore.

### Area TR - Flag

Questi bit vengono utilizzati per memorizzare temporaneamente le condizioni ON/OFF nei rami del programma (**flag**). Sono impiegati solo per il codice mnemonico. Programmando direttamente con il software di programmazione, i bit TR vengono gestiti in modo automatico.

Gli stessi bit TR non possono essere utilizzati più di una volta nell'ambito dello stesso blocco di istruzioni, ma possono essere impiegati più volte in blocchi di istruzioni diversi. Lo stato ON/OFF dei bit TR non può essere controllato da una unità periferica.

### **Area HR – Mantenimento stato**

Questi bit conservano lo stato ON/OFF anche quando viene disattivata l'alimentazione oppure quando si avvia o si arresta il funzionamento del PLC. Questi bit lavorano allo stesso modo come bit di lavoro.

### **Area AR – Bit di controllo**

Si tratta di bit che vengono utilizzati soprattutto come **flag** relativi al funzionamento del PLC. Questi bit conservano lo stato ON/OFF anche quando viene disattivata l'alimentazione oppure quando si avvia o si arresta il funzionamento del PLC.

### **Area LR – Bit di connessione**

Quando esiste una connessione tra un due PLC Omron CPM, questi bit vengono utilizzati per la condivisione dei dati. Quando non vengono utilizzati nelle connessioni PC link 1:1, i bit LR possono essere usati come bit di lavoro.

### **Area TC – Temporizzatori / Contatori**

Si tratta di un'area utilizzata per gestire i temporizzatori e i contatori creati con le istruzioni TIM, TIMH(15), CNT e CNTR(12).

Gli stessi numeri vengono utilizzati sia per i temporizzatori che per i contatori ed ogni numero può essere impiegato solo una volta nel programma utente.

Non è possibile utilizzare lo stesso numero TC per più di una volta anche per istruzioni diverse.

I numeri TC vengono utilizzati per la creazione di temporizzatori e contatori e per l'accesso a flag di completamento e valori correnti (PV). Se un numero TC viene designato per i dati di canali, accede al valore corrente (PV); se viene utilizzato per i dati dei bit, accede al flag di completamento per il temporizzatore/contatore.