



I CONTROLLORI A LOGICA PROGRAMMABILE

Prof. Giuseppe Porcaro
Prof.ssa Maria Tattoli

Sommario

I SISTEMI AUTOMATICI	3
DISPOSITIVI DI I/O: SENSORI ED ATTUATORI	3
I CONTROLLORI A LOGICA PROGRAMMABILE	5
INPUT E OUTPUT DIGITALI.....	6
INPUT.....	6
OUTPUT.....	8
SCHEMI DI COLLEGAMENTO DEGLI I/O	10
CARATTERISTICHE TECNICHE DI UN PLC.....	11
Taglia di un PLC.....	11
Ciclo del PLC e tempo del ciclo.....	11
Set di istruzioni	12
IL DIAGRAMMA LADDER	12
ISTRUZIONI SET E RESET	15
PUNTI INTERNI (o RELE' INTERNI o MERKERS).....	17
OPERAZIONI DI TEMPORIZZAZIONE E CONTEGGIO.....	18
IL TIMER.....	18
IL CONTATORE	20
ONE SHOT	22
OPERAZIONI DI CONFRONTO	23
SISTEMI SEQUENZIALI	24
INPUT E OUTPUT ANALOGICI	27
OPERAZIONI MATEMATICHE.....	31
OPERAZIONI DI TRASFERIMENTO	32

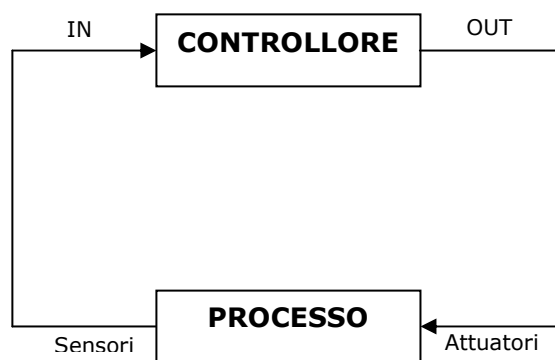
I SISTEMI AUTOMATICI

Un **sistema** può essere definito come **un'unità fisica e funzionale costituita da più parti interagenti tra loro, formando un tutt'uno in cui ogni parte dà un contributo per una finalità comune.**

Un **sistema automatico** è un sistema non manuale, cioè un **sistema in grado di agire e prendere decisioni senza l'intervento dell'uomo.**

L'automatizzazione di un sistema prevede:

- un **processo**,
- un **controllore** del processo.



Per **processo** si intende l'insieme di parametri (variabili) che caratterizzano il sistema e che nel tempo evolvono. I valori di questi parametri sono comunicati al **controllore** attraverso "organi di senso", detti **sensori** o **trasduttori**, cioè dispositivi in grado di trasformare grandezze fisiche di varia natura in segnali elettrici. Gli **attuatori** sono invece gli strumenti attraverso cui il controllore determina variazioni nei parametri.

Il controllore può essere di due tipi:

CLC – controllore a logica cablata (cabled logic controller)

PLC – controllore a logica programmabile (programmable logic controller)

Per **logica** si intende la modalità di intervento sui parametri.

Nei CLC la logica è nell'hardware, cablata nella circuiteria, e non è modificabile. Ne sono esempi le centraline dei cancelli elettrici, che si montano, collegando ad essi i componenti del cancello.

Nei PLC, invece, la logica è determinata da una serie di istruzioni, cioè da un programma. Essi consentono sia facilità di realizzazione e test delle soluzioni proposte, sia un'agevole manutenibilità di quanto sviluppato.

DISPOSITIVI DI I/O: SENSORI ED ATTUATORI

I dispositivi di I/O utilizzati nei sistemi automatici possono essere di due tipi:

- **digitale**
- **analogico**

È digitale tutto ciò che può assumere solo due stati (ON/OFF, vero/falso, 1/0). È analogico invece tutto ciò che può assumere valori in un intervallo (**range**).

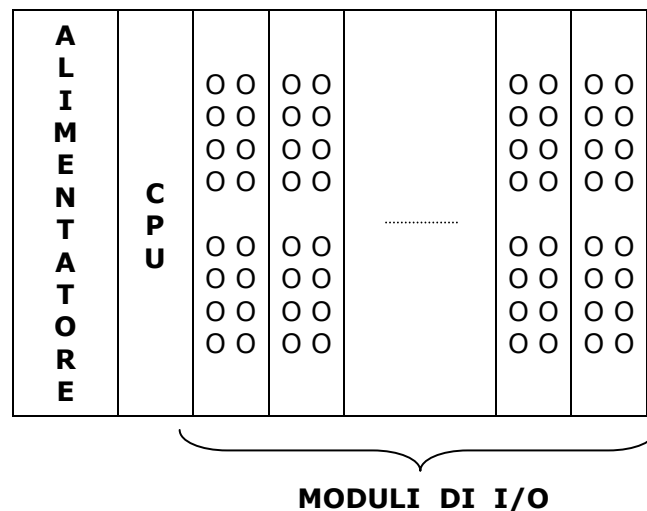
Esempi di dispositivi di I/O sono:

- il **pulsante**, i cui due stati sono "premuto" e "non premuto"; è un dispositivo di input digitale.
- l'**interruttore**, i cui due stati sono "ON" e "OFF"; è un dispositivo di input digitale.
- la **fotocellula**, che è un dispositivo elettronico che restituisce un contatto aperto o chiuso a seconda dell'intensità luminosa alla quale è esposta, segnalando così la presenza o assenza di qualcosa che si frapponga tra essa e una sorgente di luce; gli impieghi sono molteplici e vanno dai dispositivi d'allarme agli automatismi per cancelli o porte. È un dispositivo di input digitale.
- il **fincorsa**, che ricalca il funzionamento del pulsante, con la differenza che l'azione meccanica non viene impressa dall'operatore, ma da un organo meccanico in movimento, come un cancello, nel momento in cui raggiunge la posizione di massima apertura o di massima chiusura, o un ascensore, nel momento in cui raggiunge uno dei piani; è un dispositivo di input digitale.
- il **termostato**, che segnala se la temperatura di un ambiente è al di sotto o al di sopra di un valore di riferimento; altrettanto fa il **pressostato** per la pressione. Sono entrambi dispositivi di input digitale.
- la **lampada** o la **sirena** o tutti i dispositivi che si possono accendere o spegnere; sono dispositivi di output digitale.
- un **motore** che si muove a velocità costante (come quelli di apertura e chiusura dei cancelli, di salita e discesa degli ascensori, dei nastri trasportatori delle casse dei supermercati), il cui stato è "in movimento" o "fermo"; è un dispositivo di output digitale. In genere nell'automazione industriale si usano motori ON/OFF. Un motore che invece può muoversi a velocità variabile è un dispositivo di output analogico, perché non ha solo due stati, essendo il suo stato definito dai diversi valori di velocità.
- il **termometro** (trasduttore di temperatura), che rileva il valore della temperatura di un ambiente; è un dispositivo di input analogico.
- il **pressometro** (trasduttore di pressione), che rileva il valore della pressione di un fluido; è un dispositivo di input analogico.
- il **luxmetro** che rileva il valore di luminosità; è un dispositivo di input analogico.

I CONTROLLORI A LOGICA PROGRAMMABILE

Il **PLC** è un dispositivo utilizzato per il controllo di processi industriali; si tratta di un piccolo computer i cui componenti sono stati progettati per operare in ambiente industriale, per cui le sue caratteristiche risultano affidabili anche quando si opera in ambiente ostile; inoltre occupa poco spazio, è di facile manutenzione, è riprogrammabile e di impiego flessibile.

La struttura base di un PLC è la seguente:



Nella scheda denominata **CPU** e' presente sia il microprocessore che la memoria del PLC.

La **memoria** si divide in

- **MEMORIA PROGRAMMA:** costituita da componenti RAM ed EPROM, è usata per memorizzare il programma di lavoro del PLC. Nella RAM viene memorizzato il programma in fase di sviluppo e test, mentre nella EPROM viene scritto il programma definitivo.
- **MEMORIA AUSILIARIA:** costituita da componenti RAM, è usata per memorizzare lo stato dei segnali, sia di ingresso che di uscita, e le variabili usate dal programma.

Compito del PLC è acquisire segnali in INPUT e generare, in risposta, segnali in OUTPUT (ciclicamente). Le **unità di I/O** (generalmente organizzate in "schede") hanno pertanto un'importanza rilevante nella configurazione di un PLC e ne identificano le caratteristiche funzionali.

Si fa distinzione tra:

- **unità di I/O digitali**, per il controllo di dispositivi di tipo binario (ON/OFF)
- **unità di I/O analogiche**, per il controllo di grandezze continue.

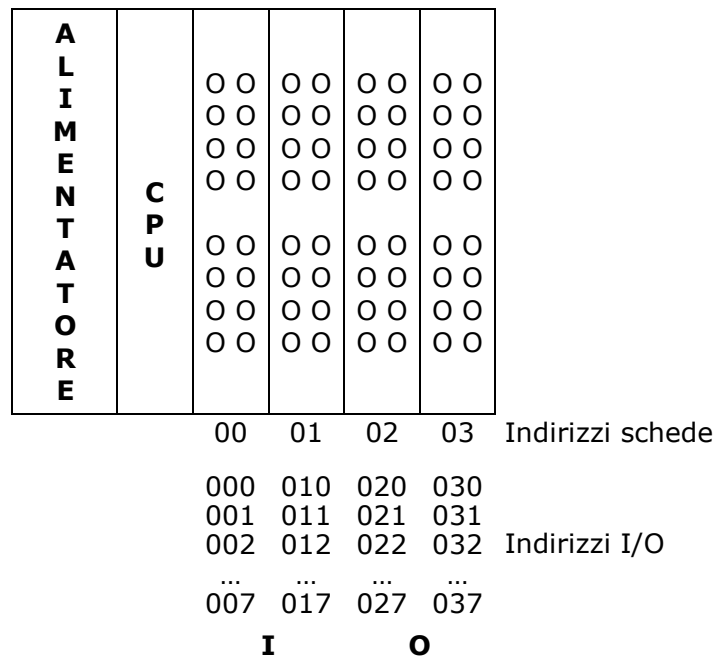
Ad ogni unità di I/O è associato un indirizzo che lo identifica univocamente. Ci sono due tipi di indirizzamento:

- **Indirizzamento di tipo geografico:** l'indirizzo di una scheda di I/O dipende dalla posizione in cui è collocata nel rack; è il tipo di indirizzamento più usato.
- **Indirizzamento programmabile o per codice:** si può stabilire, attraverso particolari microswitch, l'indirizzo di un I/O indipendentemente dalla posizione nel rack.

Esempio:

Consideriamo un PLC dotato di due schede di Input e due schede di Output, ciascuna contenente otto unità, rispettivamente di ingresso e uscita.

L'indirizzo (geografico) di ogni unità è costituito dal numero della scheda nel rack (00, 01, 02, 03) seguito dal numero d'ordine, nella scheda, dell'ingresso o dell'uscita.



Nel **PLC SIEMENS** con **CPU 214**, a cui si farà riferimento in tutta la trattazione successiva, sono presenti (incorporate nel rack):

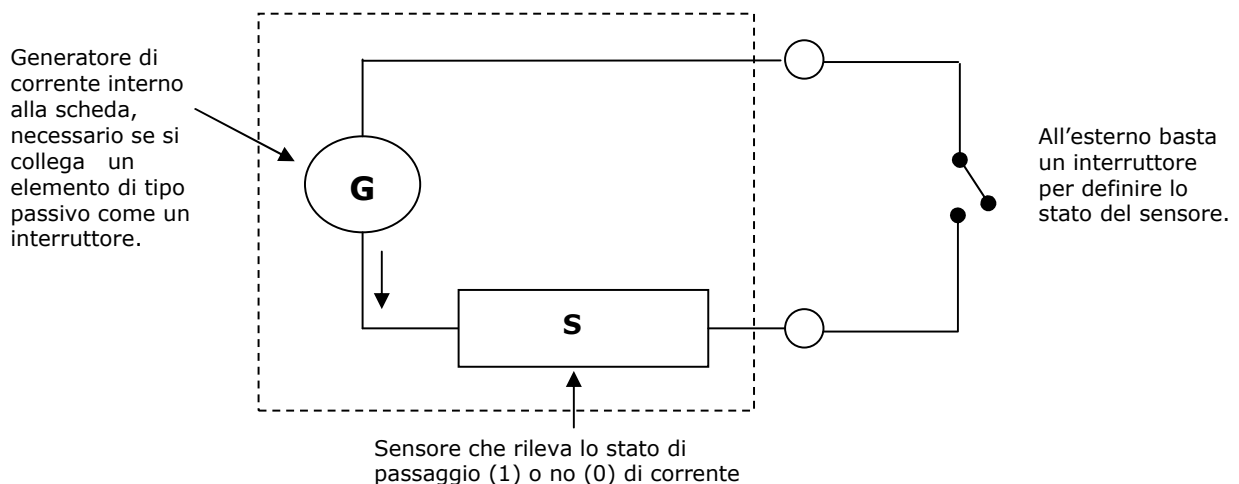
- una scheda di input digitali composta da 14 ingressi, i cui indirizzi sono I0.0, I0.1, I0.2, I0.3, I0.4, I0.5, I0.6, I0.7, I1.0, I1.1, I1.2, I1.3, I1.4, I1.5;
- una scheda di output digitali composta da 10 uscite, i cui indirizzi sono Q0.0, Q0.1, Q0.2, Q0.3, Q0.4, Q0.5, Q0.6, Q0.7, Q1.0, Q1.1.

Gli indirizzi sono espressi in ottale.

INPUT E OUTPUT DIGITALI

INPUT

Lo schema di **un'unità di input digitale** è il seguente

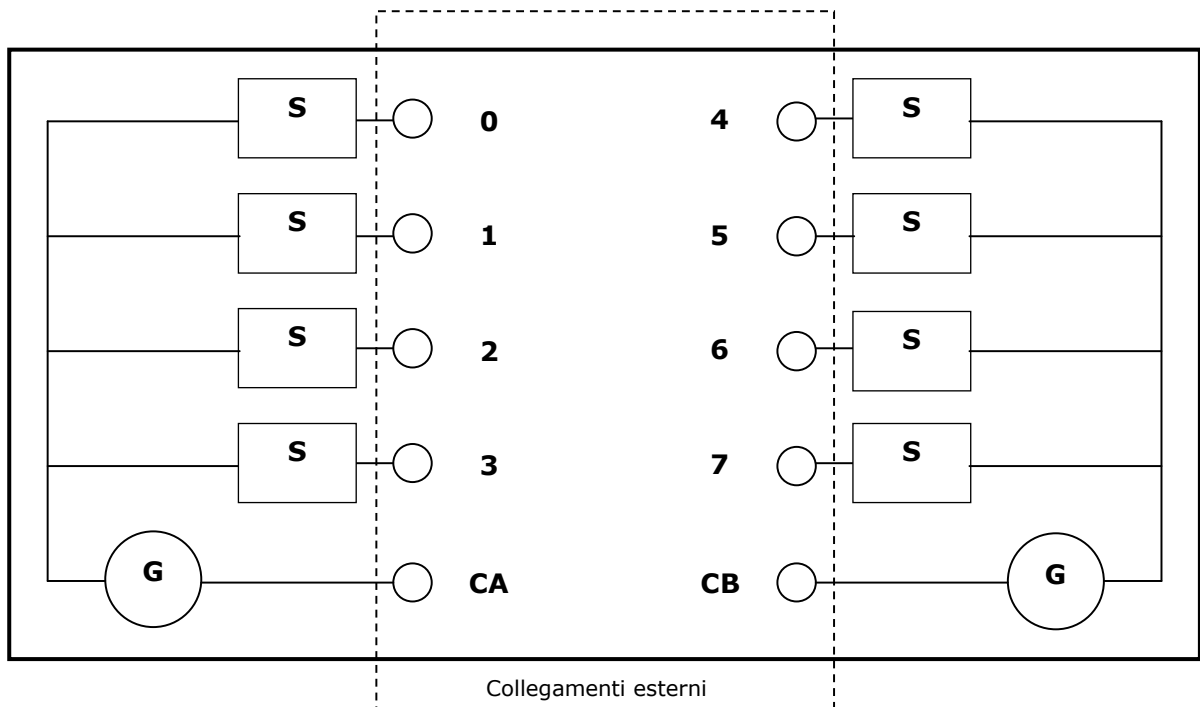


Lo schema visto è quello di una **scheda per input liberi da tensione**.

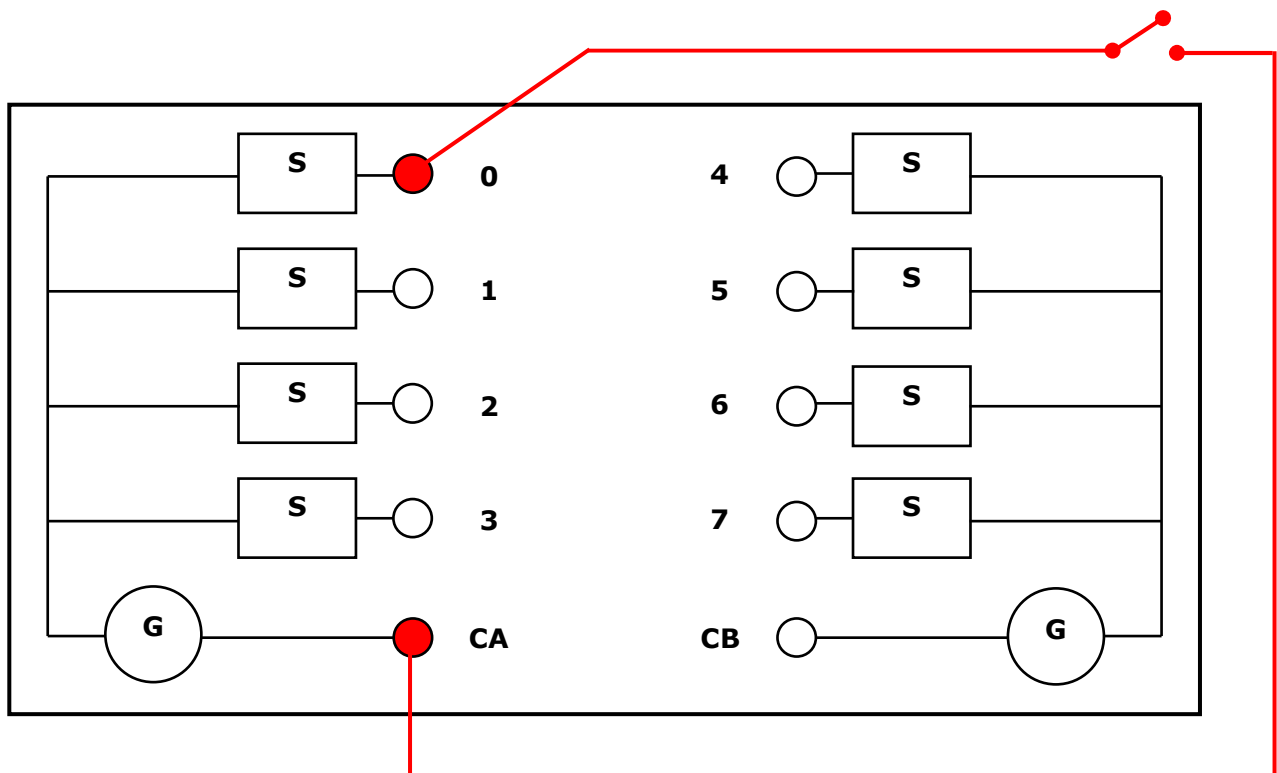
Nel caso in cui la scheda non presenti un generatore interno, il dispositivo ON-OFF collegato dovrà disporre di un generatore esterno.

Una **scheda di input digitale** è generalmente organizzata in gruppi di unità; ciascun gruppo ha lo stesso comune.

Viene di seguito illustrata una scheda con 2 gruppi da 4 unità



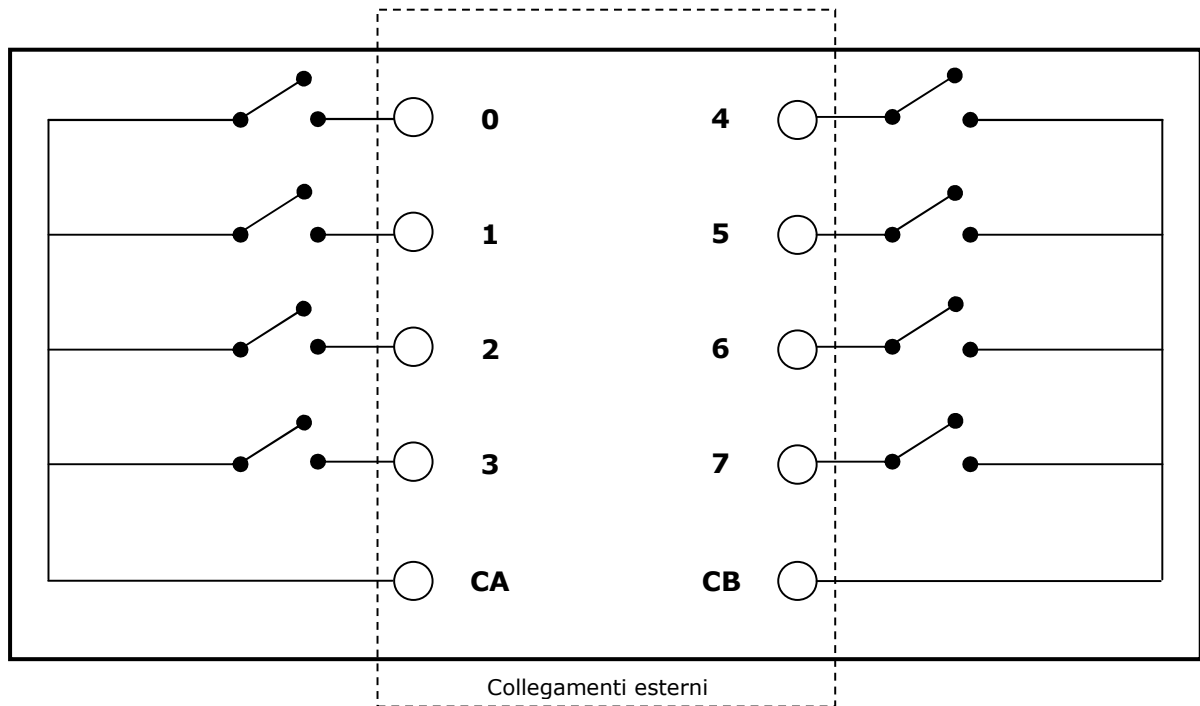
Lo schema seguente mostra il modo in cui collegare un interruttore per attivare un ingresso di una scheda di input.



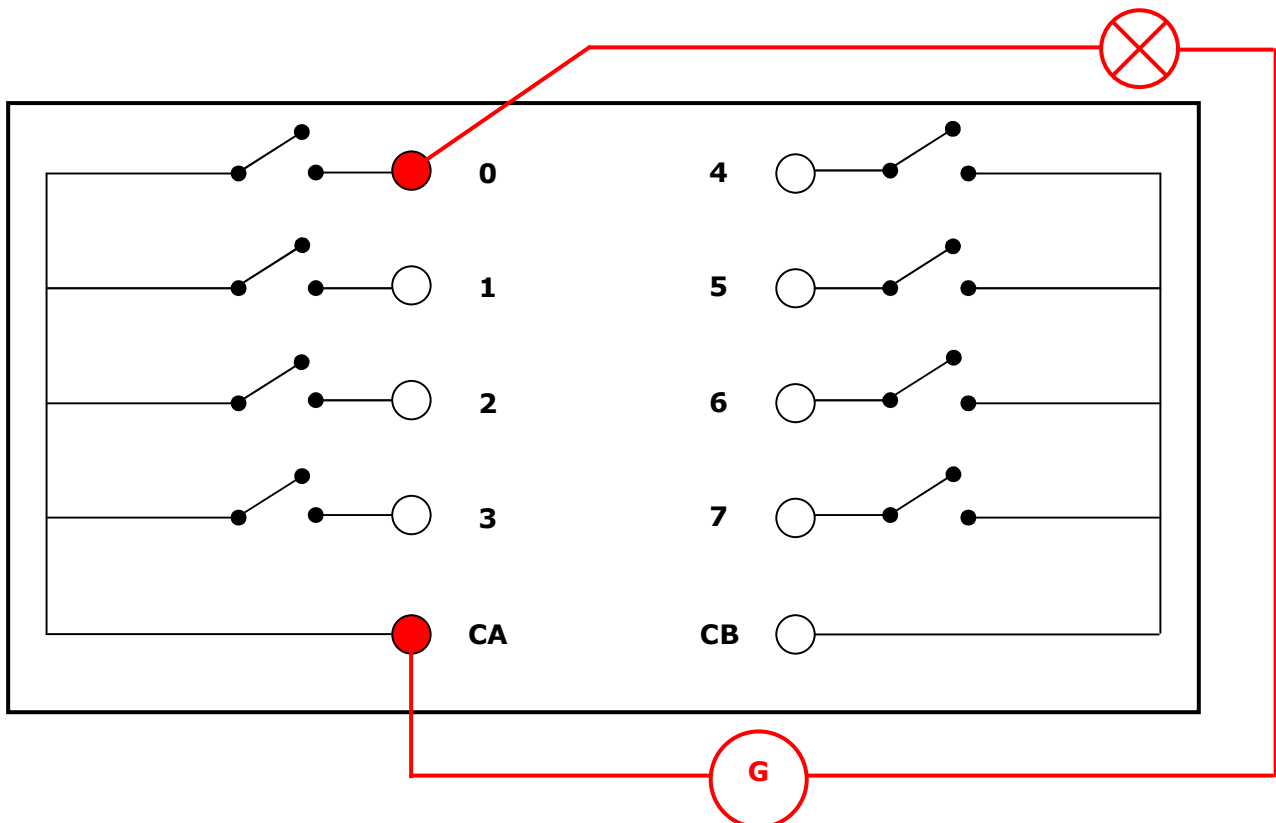
OUTPUT

Un'unità di output digitale può essere invece vista come un interruttore di cui il PLC determina lo stato (aperto o chiuso).

Anche una scheda di output digitale, come già visto per la scheda di input, generalmente presenta le unità divise in gruppi che condividono lo stesso comune, come illustrato nello schema seguente:



Lo schema che segue illustra il modo in cui è possibile collegare una lampada a un'unità di una scheda di output.

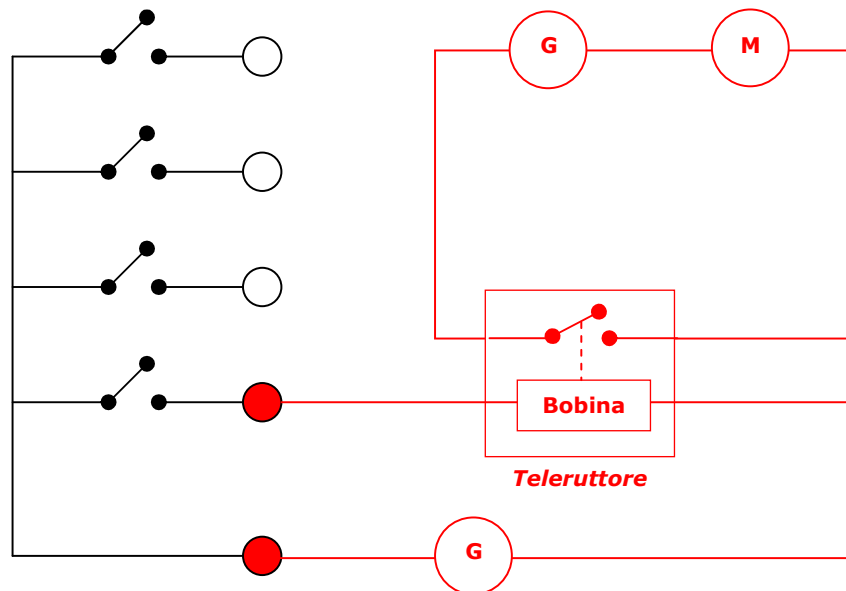


Come si può notare, la lampada è alimentata esternamente, in quanto le unità di uscita svolgono solo le funzioni di interruttori. Lo schema riportato è dunque quello di una **scheda di output liberi da tensione**, per cui è necessario che gli attuatori collegati abbiano un'alimentazione esterna (gli output si limitano a chiudere o aprire il circuito). Il caso alternativo è quello in cui gli output sono alimentati all'interno del PLC (gli attuatori collegabili devono essere ovviamente adeguati al valore di tensione fornito).

L'uscita di un PLC può anche essere utilizzato per "attivare" circuiti più complessi, che richiedono valori di tensione, e quindi di potenza, più elevati. È sufficiente collegare all'uscita di un PLC un **relè**, che è un dispositivo in cui è presente una bobina che controlla una o più coppie di contatti; nel momento in cui la bobina viene eccitata (ad esempio, facendo circolare in essa corrente), chiude i contatti; in tal modo è possibile, attraverso l'output digitale di un PLC, che opera a bassa tensione, chiudere o aprire un circuito che opera a tensione più elevata.

Esempio:

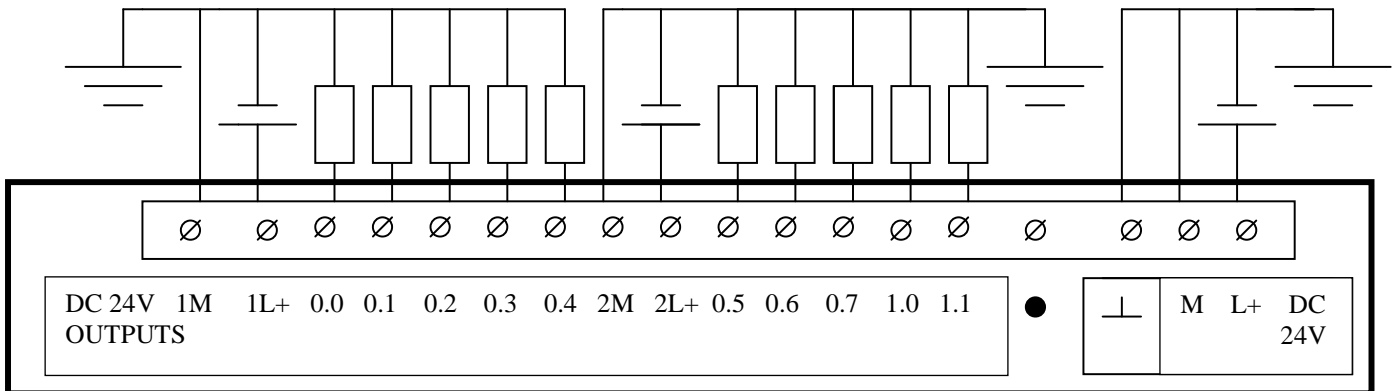
Lo schema seguente illustra il collegamento ad un motore mediante un teleruttore, che è un particolare tipo di relè.



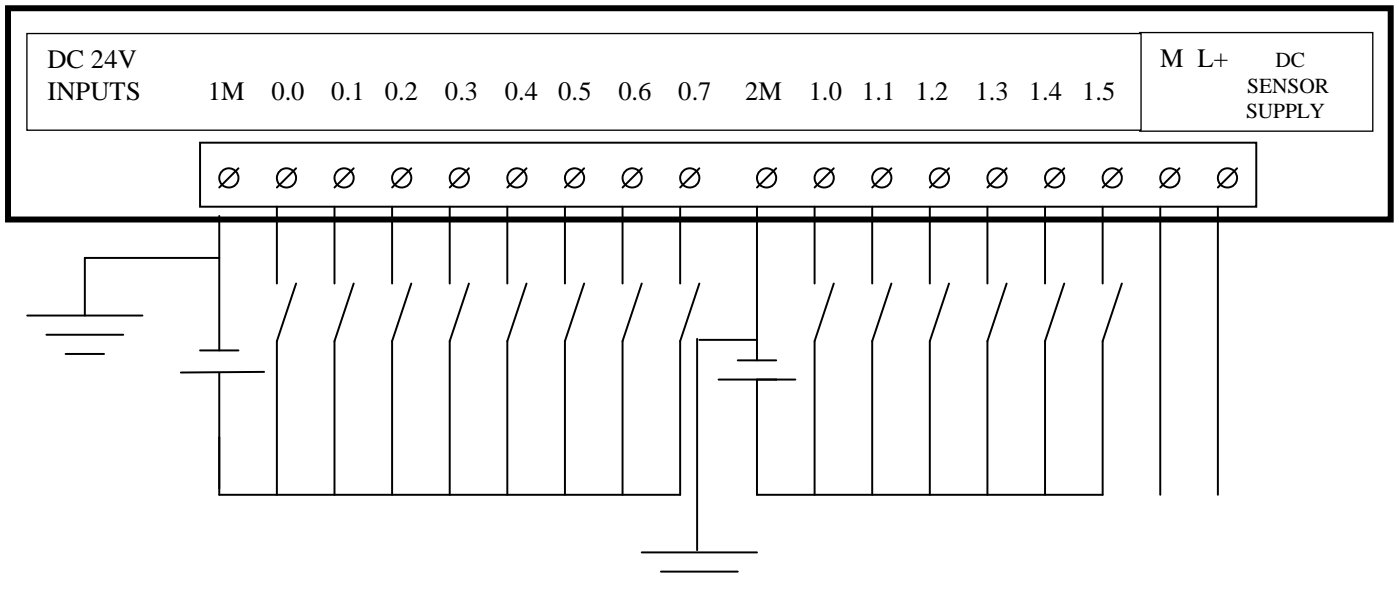
SCHEMI DI COLLEGAMENTO DEGLI I/O

Per il PLC Siemens con CPU 214 gli schemi di collegamento degli I/O ai dispositivi esterni sono i seguenti:

OUTPUT (scheda con 10 output liberi da tensione)



INPUT a 24 V (scheda per 14 input con alimentazione esterna)



CARATTERISTICHE TECNICHE DI UN PLC

Un PLC è classificato in genere sulla base delle seguenti caratteristiche:

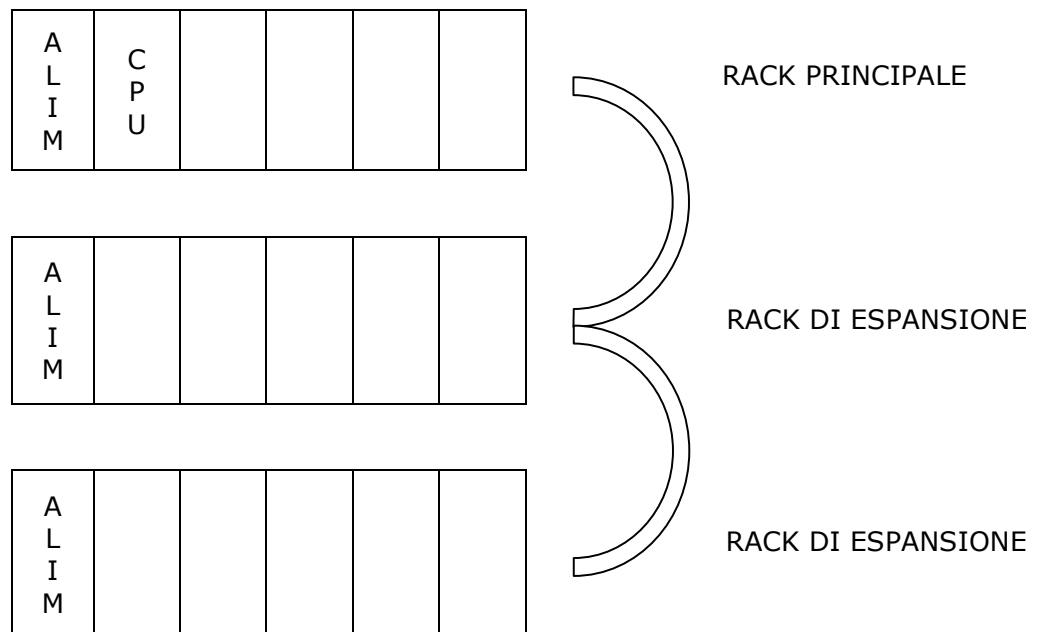
- la taglia
 - il tempo del ciclo
 - il set di istruzioni
 - la capacità di memoria
- } *sono le due caratteristiche più importanti*

Taglia di un PLC

Si parla di **taglia** di un PLC come del numero massimo di Input e di Output (e quindi di schede di I/O) di cui può disporre.

PLC di **taglia piccola** arrivano fino a 128 I/O, PLC di **taglia media** raggiungono i 512 I/O, PLC di **taglia grande** superano i 512 I/O.

E' possibile espandere la struttura di un PLC aggiungendo rack di espansione e quindi aumentando la disponibilità di slot in cui inserire schede di I/O, come illustrato nello schema seguente:



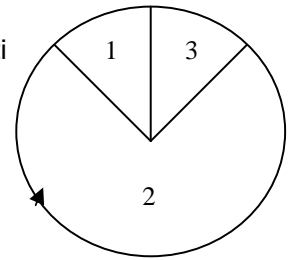
Ciclo del PLC e tempo del ciclo

Come si è detto, il PLC è un dispositivo programmabile.

Il programma viene eseguito dal PLC ciclicamente e in modo ininterrotto. Il tempo necessario al PLC per eseguire l'intero programma prende il nome di **tempo del ciclo**; esso dipende sia dalla lunghezza del programma che dalle caratteristiche del PLC ed è dell'ordine di qualche millisecondo (ms).

Nel corso di ogni ciclo vengono svolte le seguenti fasi:

- **fase 1:** gli input vengono prelevati dalle schede di input e portati nella memoria interna;
- **fase 2:** il programma viene elaborato e vengono generati gli output nella memoria interna;
- **fase 3:** gli output vengono portati all'esterno sulle schede di output.



Si può definire anche il concetto di **frequenza** del PLC.

Un PLC che impiega, ad esempio, 20 ms per compiere un ciclo ha frequenza pari a 50 Hz, cioè gli input vengono letti 50 volte al secondo e gli output vengono aggiornati 50 volte al secondo.

Set di istruzioni

Anche i programmi per PLC si compongono di istruzioni. Il set di istruzioni è l'insieme delle istruzioni che un PLC è in grado di eseguire.

Le istruzioni si classificano in :

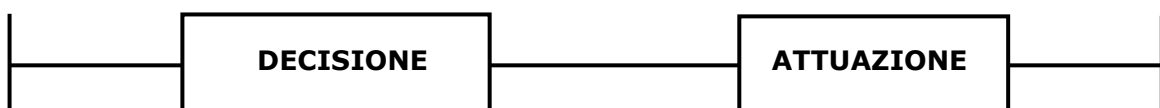
- **istruzioni numeriche (o di calcolo):** sono le istruzioni che compiono operazioni di calcolo numerico (addizioni, sottrazioni, moltiplicazioni, divisioni).
- **istruzioni logiche:** sono le istruzioni realizzate attraverso gli operatori dell'algebra di Boole.

IL DIAGRAMMA LADDER

Un **programma logico** (programma con istruzioni logiche) può essere rappresentato mediante un **DIAGRAMMA LADDER**, che si presenta composto da una sequenza di **rami** o **segmenti**, ciascuno dei quali si compone di

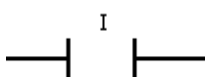
- **una parte decisionale** (a sinistra)
- **una parte attuativa** (a destra).

La logica del LADDER è una logica a relè: se dalla parte decisionale arriva corrente, l'output viene attivato.

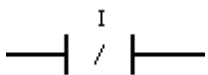


Il **blocco di decisione** è costituito da una combinazione di elementi logici, realizzata attraverso gli operatori booleani (AND, OR, NOT) ed espressa come schema a contatti; ogni contatto rappresenta lo stato logico di un elemento binario del PLC (ad esempio di un ingresso, di un'uscita, di un bit di memoria). In corrispondenza del contatto si indica l'indirizzo dell'elemento che esso rappresenta o il suo nome simbolico.

Si distinguono due tipi di contatti:

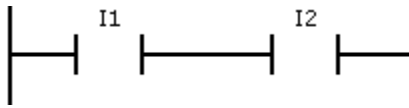


Contatto normalmente aperto: è chiuso (ON) se il valore del bit associato è 1, è aperto (OFF) se il valore del bit associato è 0.

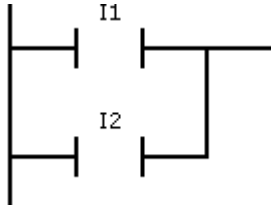


Contatto normalmente chiuso: è chiuso (ON) se il valore del bit associato è 0, è aperto (OFF) se il valore del bit associato è 1.

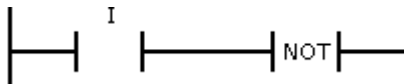
Le operazioni logiche tra contatti vengono così indicate:



I1 AND I2 – vale 1 solo nel caso in cui sia I1 che I2 valgano 1, 0 in caso contrario.



I1 OR I2 – vale 1 nel caso in cui almeno uno tra I1 e I2 valga 1.

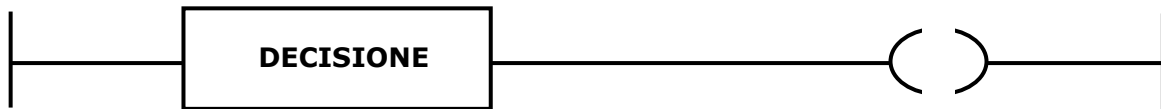


NOT I - inverte lo stato dei segnali.

Se il segnale che raggiunge il contatto NOT è 1, in uscita si avrà 0. Se il segnale entrante è 0, in uscita si avrà 1.

Un contatto normalmente chiuso si comporta come un contatto normalmente aperto seguito dall'operatore NOT.

Il **blocco di attuazione** presenta uno o più elementi il cui valore viene definito dal risultato della parte decisionale. L'attuazione più semplice è del tipo = (l'elemento oggetto dell'attuazione assume lo stesso valore logico della parte decisionale):



Esempio: comandare il movimento di un nastro trasportatore che si deve muovere quando è inserita una chiave di attivazione e si deve fermare quando un oggetto da esso trasportato transita davanti ad una fotocellula.

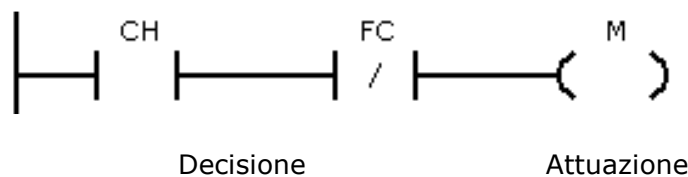
Input: CH – chiave di attivazione del movimento del nastro
FC – fotocellula (normalmente aperta)

Output: M – motore nastro

Il motore si deve muovere quando

- la chiave è inserita (CH=1)
- la fotocellula non rileva la presenza di un pacco (FC=0).

Il diagramma LADDER che descrive questo modo di operare è il seguente:



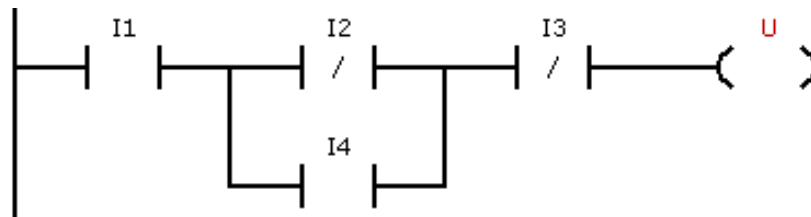
In algebra di Boole:

$$CH \text{ AND } (\text{NOT } FC) = M$$

Si osservi che il movimento del motore è descritto anche dalla seguente tabella di verità:

CH	FC	M
0	0	0
0	1	0
1	0	1
1	1	0

Esempio: si consideri il seguente ramo di diagramma ladder

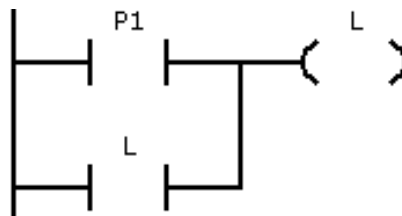


La funzione corrispondente può essere così formalizzata:

$$I1 \text{ AND } ((\text{NOT } I2) \text{ OR } I4) \text{ AND } (\text{NOT } I3) = U$$

Normalmente è possibile utilizzare gli output anche nella parte decisionale del LADDER .

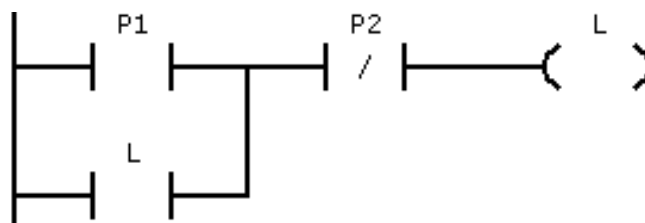
Consideriamo, ad esempio, lo **SCHEMA DI AUTORITENUTA**



P1 è un pulsante, L1 una lampada. Quando il pulsante viene premuto, la lampada si accende e rimarrà accesa anche quando il pulsante verrà rilasciato.

Questo schema viene detto di **AUTORITENUTA PURA**, in quanto **non c'è condizione di sgancio**, cioè nulla potrà far spegnere la lampada, una volta che è stata accesa.

Lo schema seguente non è invece di autoritenuta pura: è infatti presente una condizione di sgancio, esercitata premendo il pulsante P2.



ISTRUZIONI SET E RESET

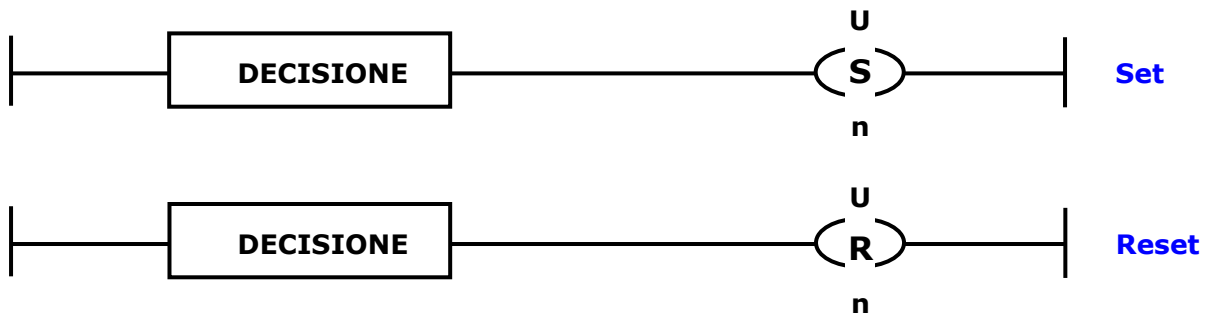
Negli esempi visti precedentemente, l'attuazione era del tipo = (uguale), cioè l'uscita assume il valore booleano che risulta dalla parte decisionale.

Diverso è il comportamento delle istruzioni SET e RESET.

L'istruzione SET pone a 1 l'uscita se la parte decisionale è verificata, altrimenti l'uscita resta inalterata. L'istruzione RESET, invece, pone a 0 l'uscita se la parte decisionale è verificata, altrimenti l'uscita resta inalterata.

L'azione delle istruzioni SET e RESET può anche interessare più uscite contemporaneamente, purché connesse ad unità di uscita con indirizzi successivi.

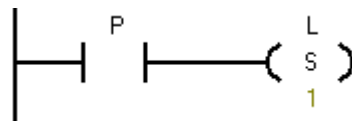
Le loro rappresentazioni nel diagramma ladder sono le seguenti:



Il numero n individua il numero di indirizzi a cui è rivolta l'operazione di set o di reset.

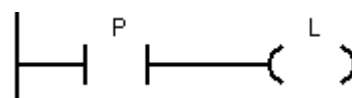
Esempio: Siano P un pulsante e L una lampada.

Il segmento

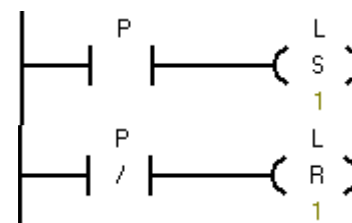


fa in modo che, premendo il pulsante P, la lampada L si accenda. Anche rilasciando P la lampada resta accesa, perché se la parte decisionale non è verificata la parte attuativa non cambia stato.

Diverso è invece il comportamento del ramo



in cui l'output è 1 se la parte decisionale è verificata, 0 in caso contrario. Per ottenere con SET e RESET un comportamento di questo tipo, occorre invece realizzare due rami:

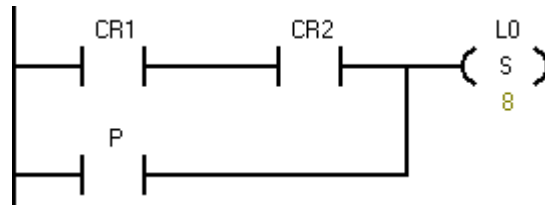


Premendo il pulsante P, la lampada si accende, rilasciando il pulsante la lampada si spegne.

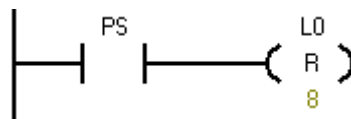
Esempio: Si consideri un sistema d'illuminazione composto da due sensori crepuscolari, CR1 e CR2, un pulsante P e 8 lampade, L0, L1, ..., L7. Le lampade si accendono quando entrambi i sensori segnalano un livello di luminosità inferiore ad un valore di riferimento. Il pulsante

ha il solo compito di accendere le luci nel caso in cui vi sia un'avaria ad almeno uno dei sensori.

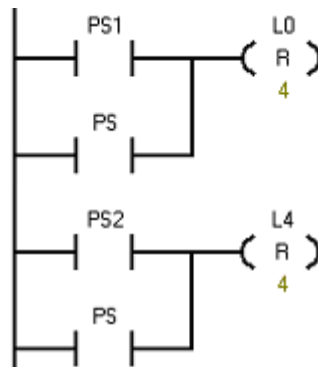
Se gli indirizzi delle lampade sono consecutivi (es. Q0.0, Q0.1, Q0.2, ecc.), è sufficiente un solo ramo per controllare il sistema:



Se sono attivi i sensori crepuscolari o se è premuto il pulsante, si accendono le 8 lampade. Per il loro spegnimento, si deve attuare il reset mediante il verificarsi di un'ulteriore condizione, ad esempio la pressione di un secondo pulsante:

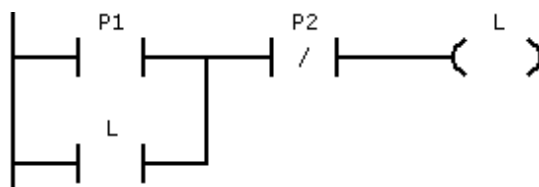


oppure si può agire su gruppi di 4 lampade, ad esempio introducendo il pulsante PS1 per spegnere solo le prime 4 lampade e il pulsante Ps2 per le altre 4, fermo restando che PS le spegne tutte.

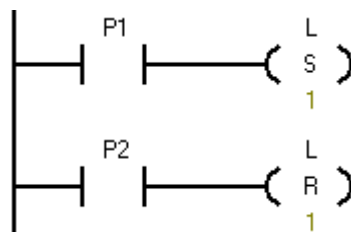


Osservazione: L'uso di SET e RESET consente di realizzare in modo diverso l'autoritenuta.

Esempio: P1 e P2 pulsanti, L lampada. Il ramo con autoritenuta



è equivalente a:



Anche in questo modo, infatti, P1 accende L e P2 la spegne.

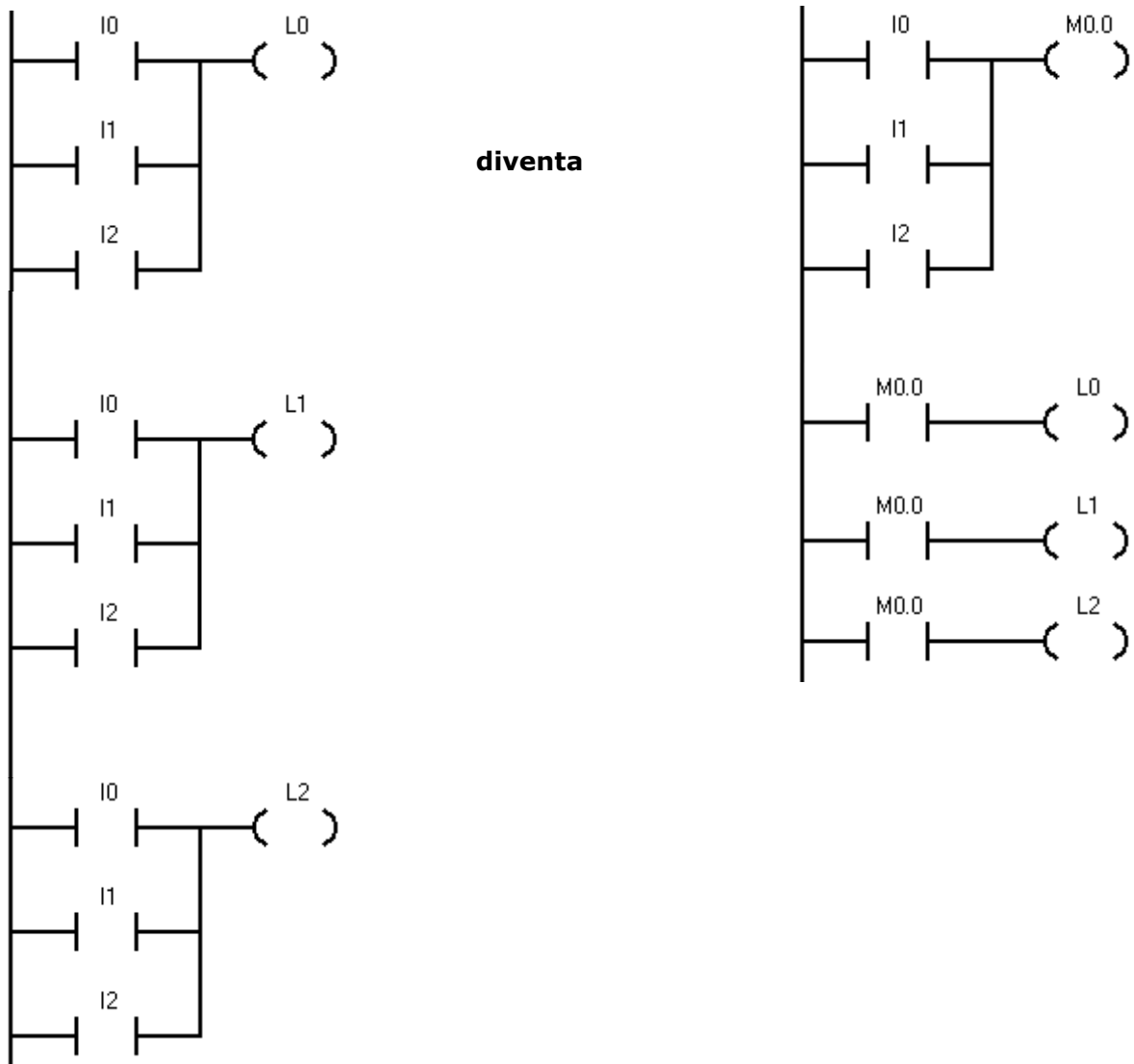
PUNTI INTERNI (o RELE' INTERNI o MERKERS)

Il punto interno è un'unità elementare di memorizzazione (un bit) in cui è possibile memorizzare uno stato logico che può poi essere utilizzato come input in altri punti del programma. Nel PLC in esame (SIEMENS con CPU 214), i punti interni hanno indirizzi compresi tra M0.0 e M31.7.

Esistono inoltre particolari punti interni detti LATCH (ritentivi), che hanno la caratteristica di essere tamponati, nel senso che, se viene tolta l'alimentazione, i dati in essi contenuti sono mantenuti grazie ad una batteria tampone. Gli indirizzi per questi tipi di elementi vanno da MB0 a MB13 (sempre nel PLC Siemens con CPU 214).

L'uso dei punti interni consente, in diverse situazioni, di semplificare notevolmente i segmenti dei programmi, evitando, ad esempio, di ripetere condizioni logiche complesse in più rami di un programma.

Esempio: usiamo il punto interno M0.0 per memorizzare uno stato logico che viene poi utilizzato per pilotare tre uscite diverse.

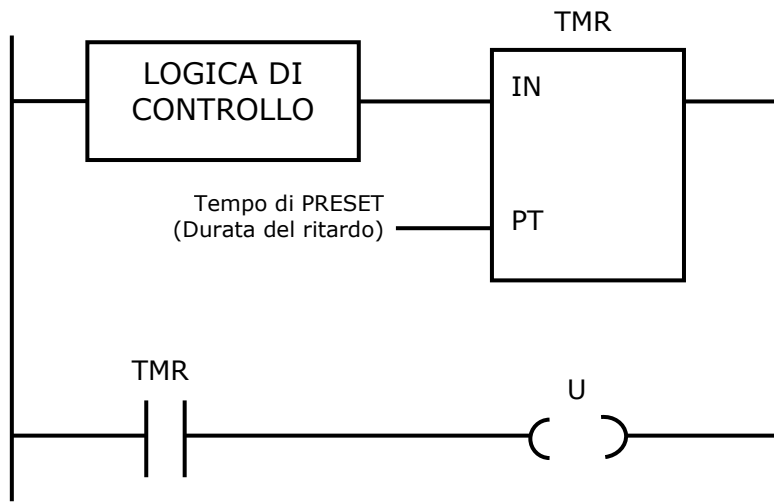


OPERAZIONI DI TEMPORIZZAZIONE E CONTEGGIO

Ogni PLC dispone di TIMER e CONTATORI, elementi che possono essere programmati per determinare ritardi nell'impostazione degli stati logici delle uscite, per sincronizzare l'esecuzione di determinate operazioni o semplicemente per eseguire azioni dopo il verificarsi di un certo numero di eventi. Le tipologie di TIMER e CONTATORI e i relativi comportamenti dipendono dal tipo di PLC.

IL TIMER

Realizza un ritardo dell'output rispetto all'input



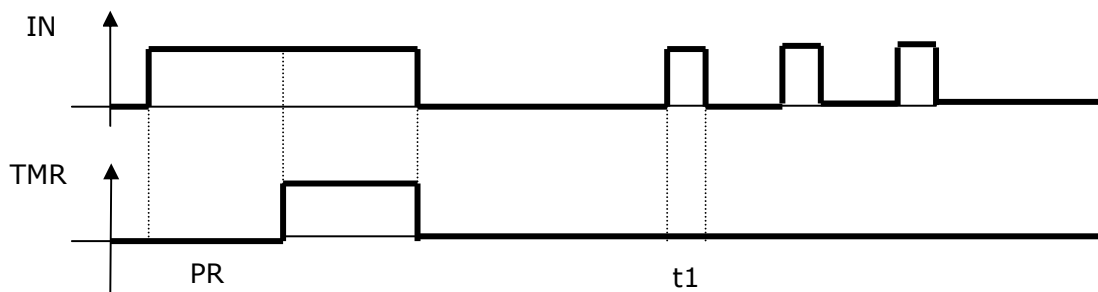
Se abilitato dalla logica di controllo ($IN=1$), il timer avvia la fase di misura del tempo, fino ad arrivare al valore impostato all'ingresso PT (tempo di preset). A questo punto lo stato logico del temporizzatore, che precedentemente era 0, diventa 1.

Si distinguono due tipi di temporizzatori:

- temporizzatori come ritardo all'inserzione
- temporizzatori come ritardo all'inserzione con memoria.

Se disattivato ($IN=0$), il temporizzatore come ritardo all'inserzione si resetta (la misura del tempo viene azzerata), mentre il temporizzatore come ritardo all'inserzione con memoria semplicemente si arresta. Per entrambi lo stato logico diventa 0.

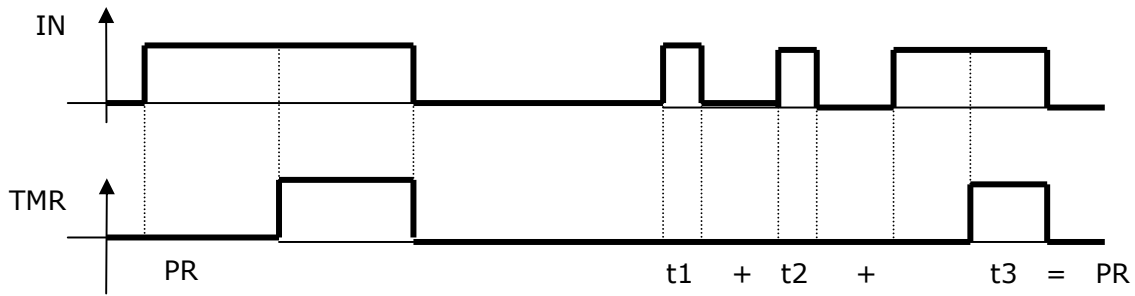
Vediamo il diagramma temporale per timer senza memoria



Quando l'ingresso è a 0, lo stato logico del timer è 0.

Quando l'ingresso è a 1, lo stato logico del timer diventa 1 solo dopo che è trascorso il tempo definito da PR. Se l'ingresso rimane alto per un tempo inferiore a PR, lo stato logico del timer rimane 0; per questo motivo gli impulsi successivi indicati nel diagramma non producono effetti.

Vediamo il diagramma temporale per timer con memoria



Quando l'ingresso è a 0, lo stato logico del timer è 0.
 Quando l'ingresso è a 1, lo stato logico del timer diventa 1 solo dopo che è trascorso il tempo definito da PR. Se l'ingresso rimane alto per un tempo inferiore a PR, lo stato logico del timer rimane 0, ma la misura del tempo rimane memorizzata ed ogni impulso successivo ne provoca l'incremento; quando si raggiunge il valore del preset, lo stato logico diventa 1; in questo caso gli impulsi successivi indicati nel diagramma producono effetti perchè si accumulano. Per resettare un timer con memoria è necessaria un'istruzione di reset.

Il **tempo di preset** è un multiplo della base di tempo del timer. La **base di tempo**, detta anche **risoluzione**, rappresenta l'unità di tempo elementare adottata nella misura del tempo. Il PLC in esame dispone di temporizzatori con tre diverse risoluzioni: 1ms, 10ms, 100ms. Ad esempio, un valore di preset pari a 50 su un temporizzatore da 100ms provoca un ritardo di 5s ($50 * 100ms = 5000ms = 5s$).

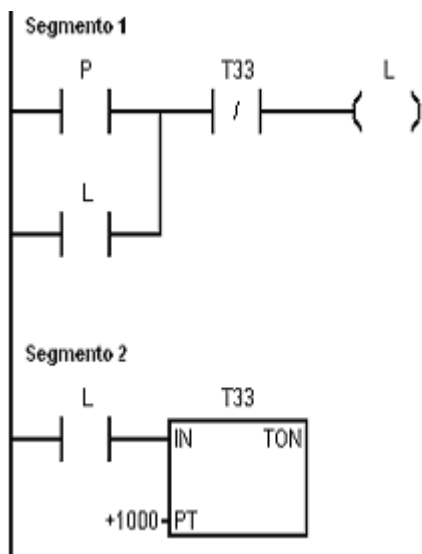
La seguente tabella riporta le caratteristiche dei timer del PLC SIEMENS con CPU 214:

TIMER	RISOLUZIONE	VALORE MASSIMO	INDIRIZZI
Senza memoria	1 ms	32,767s	T32, T96
	10 ms	327,67s	T33 - T36, T97 - T100
	100 ms	3276,7s	T37 - T63, T101 - T127
Con memoria	1 ms	32,767s	T0, T64
	10 ms	327,67s	T1 - T4, T65 - T68
	100 ms	3276,7s	T5 - T31, T69 - T95

Esempio: Illuminazione di una scalinata; premendo un pulsante, si accendono le luci, che si spengono dopo 10 secondi.

Input: Pulsante P (I0.0)

Output: Luce L (Q0.0)



Segmento 1: Quando il pulsante P è premuto e il temporizzatore T33 è allo stato logico 0 (condizione iniziale di un temporizzatore), la lampada si accende e rimane accesa perché è autoritenuta.

Segmento 2: La lampada avvia il timer T33, che, dopo 10 secondi, si porta a 1, facendo spegnere la lampada e riportandosi allo stato logico 0.

Il tempo di preset del timer è 1000, essendo:

$$10ms * 1000 = 10 s$$

dove 10 ms è la risoluzione del timer T33.

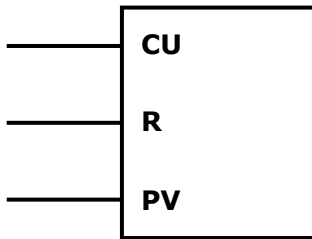
IL CONTATORE

Viene usato per il conteggio di eventi, che si presentano al suo ingresso sotto forma di impulsi. Vi sono diverse tipologie di contatori. Il PLC in esame dispone principalmente di due tipi di contatori:

- contatore in avanti
- contatore in avanti/indietro.

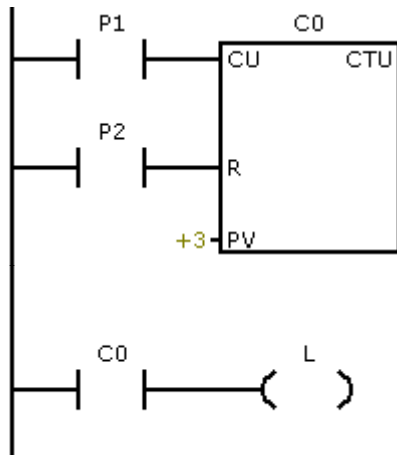
In tutto sono 128, con indirizzi da C0 a C127.

Il **Contatore in avanti** si presenta nel seguente modo:



L'ingresso CU è quello in cui devono entrare gli impulsi da contare. Il contatore conta in avanti fino al valore impostato all'ingresso PV, incrementandosi ad ogni fronte di salita che si presenta all'ingresso di conteggio CU. Al contatore è associato un bit che ne rappresenta lo stato logico; il suo valore diventa 1 quando il conteggio raggiunge il valore PV, mentre vale 0 prima del raggiungimento di tale valore. Un impulso all'ingresso di reset (R) azzerava il conteggio e riporta a 0 lo stato logico del contatore.

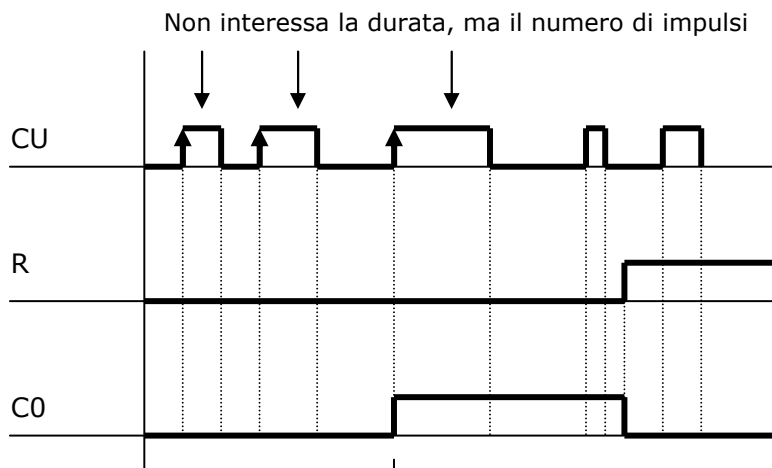
Esempio:



Il pulsante P1 deve essere premuto 3 volte per far accendere la lampada L1, che resterà accesa fino al momento in cui sarà premuto il pulsante P2.

A questo punto il conteggio sarà azzerato e lo stato logico del contatore tornerà a 0.

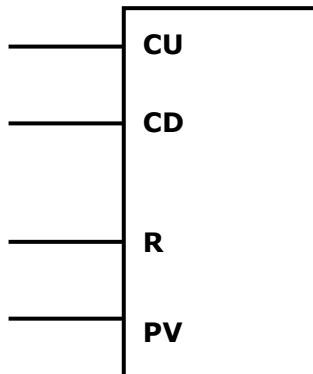
Vediamo il diagramma temporale



Non interessa la durata, ma il numero di impulsi

Si alza al raggiungimento del 3° fronte di salita che si presenta all'input di conteggio e si abbassa solo quando si alza il segnale di reset.

Il **Contatore in avanti/indietro** si presenta invece nel seguente modo:

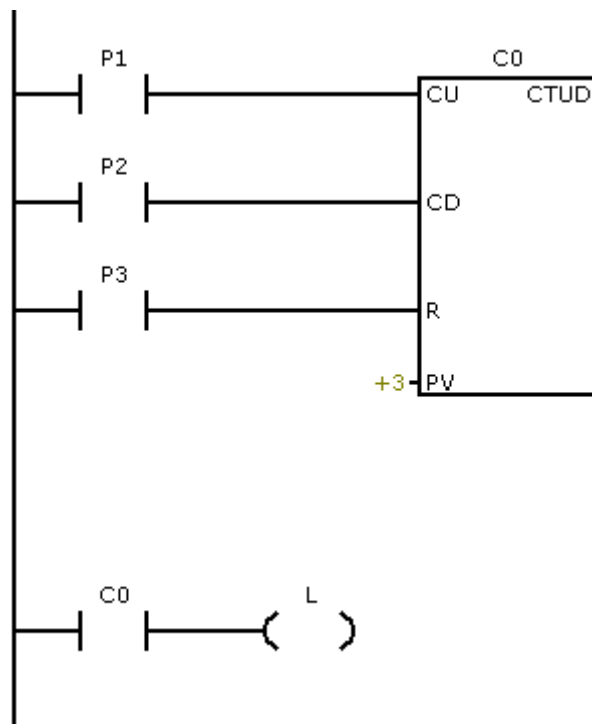


Il conteggio avviene

- in avanti (si incrementa di 1) per ogni fronte di salita che si presenta all'ingresso CU,
- all'indietro (si decrementa di 1) per ogni fronte di salita che si presenta all'ingresso CD.

Anche in questo caso lo stato logico del contatore diventa 1 quando il conteggio raggiunge il valore PV, che può essere anche un numero negativo. Un impulso all'ingresso di reset (R) azzerava il conteggio e riporta a 0 lo stato logico del contatore.

Esempio:



Il pulsante P1 fa incrementare il contatore, mentre ogni pressione del pulsante P2 lo fa decrementare.

Quando il conteggio arriva a 3, la lampada si accende; si spegnerà se, premendo il pulsante P2, il valore del conteggio scenderà al di sotto di 3 o non appena sarà premuto il pulsante P3 (il conteggio è azzerato). In tali casi, infatti, lo stato logico del contatore tornerà al valore 0.

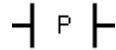
ONE SHOT

È una funzione logica del PLC usata per far accadere qualcosa per un solo ciclo di PLC.

È di due tipi:

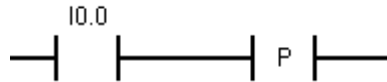
- one shot sul fronte di salita
- one shot sul fronte di discesa

Il one shot sul fronte di salita, così rappresentato:

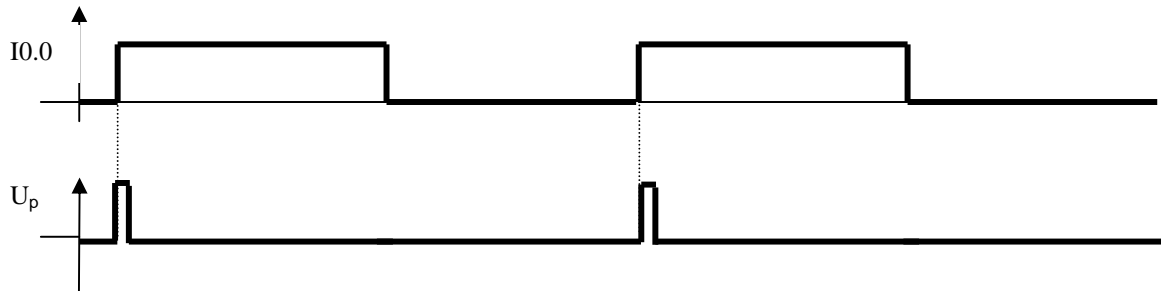


restituisce valore Vero quando la condizione che lo precede diventa Vera (fronte di salita del segnale) e ciò avviene per un solo ciclo di PLC (si genera un impulso).

Ad esempio, se la parte decisionale di un segmento è la seguente:

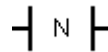


il diagramma temporale è:



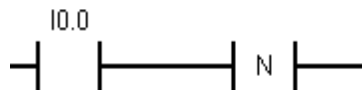
dove U_p è il valore che esce dalla funzione di one shot e quindi dalla parte decisionale.

Il one shot sul fronte di discesa è invece così rappresentato:

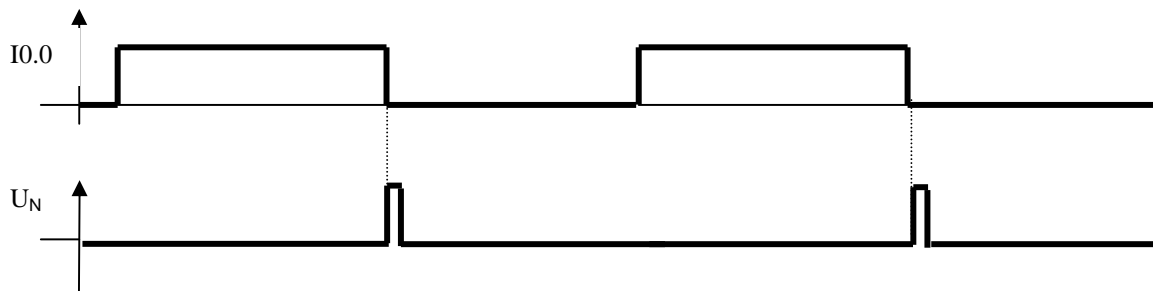


e restituisce valore Vero quando la condizione che lo precede diventa Falsa (fronte di discesa del segnale) e ciò avviene per un solo ciclo di PLC (si genera un impulso)..

Ad esempio, se la parte decisionale di un segmento è la seguente:



il diagramma temporale è:

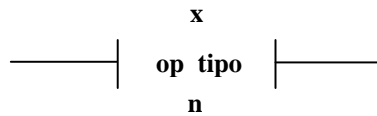


dove U_N è il valore che esce dalla funzione di one shot e quindi dalla parte decisionale.

OPERAZIONI DI CONFRONTO

Effettuano confronti tra numeri. I numeri possono essere dei seguenti tipi: Byte (intero a 8 bit), Integer (intero a 16 bit), Double (intero a 32 bit), Real (reale).

Un operatore di confronto si presenta nel seguente modo:



dove

x è la grandezza che viene confrontata

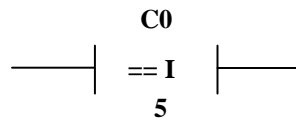
n è il numero da confrontare con x

op è l'operatore di confronto (> maggiore, >= maggiore o uguale, < minore, <= minore o uguale, == uguale, <> diverso)

tipo è la lettera che definisce il tipo (B - Byte, I - Integer, D - Double Integer, R - Real)

Il contatto è attivo quando il confronto è vero.

Ad esempio, il contatto



diventa 1 quando il conteggio del contatore C0 arriva a 5.

Non tutti gli operatori di confronto sono disponibili in tutti i tipi di PLC. Il PLC con CPU 214, ad esempio, non dispone degli operatori >, < e <>.

Esempio

Lampeggio di una lampada (tempo di accensione 1 secondo, tempo di spegnimento ½ secondo). Una chiave accende/spegne il sistema.

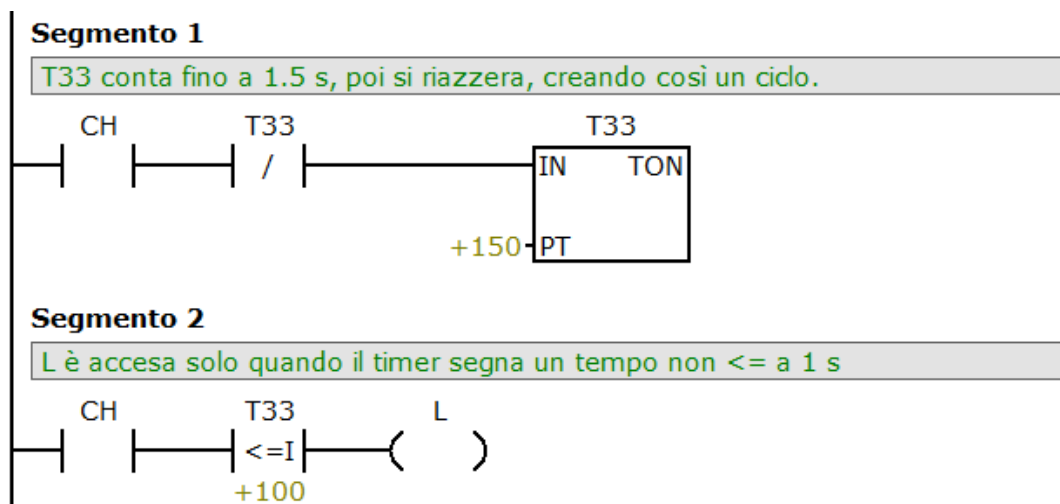
Analisi

Si può risolvere il problema usando un timer e un operatore di confronto. Il timer conta il tempo fino a 1.5 s, dopodiché si riavvizia e riparte. In ogni conteggio del tempo, fino ad 1 s la lampada è accesa, per il tempo successivo fino a 1.5 s è spenta.

Input: CH = chiave (I0.0)

Output: L = lampada (Q0.0)

Diagramma ladder

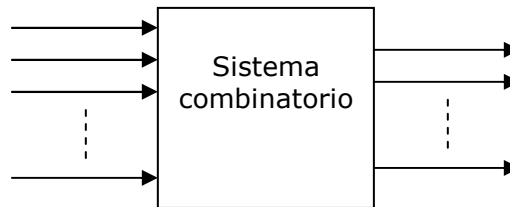


SISTEMI SEQUENZIALI

I sistemi si classificano essenzialmente in due categorie:

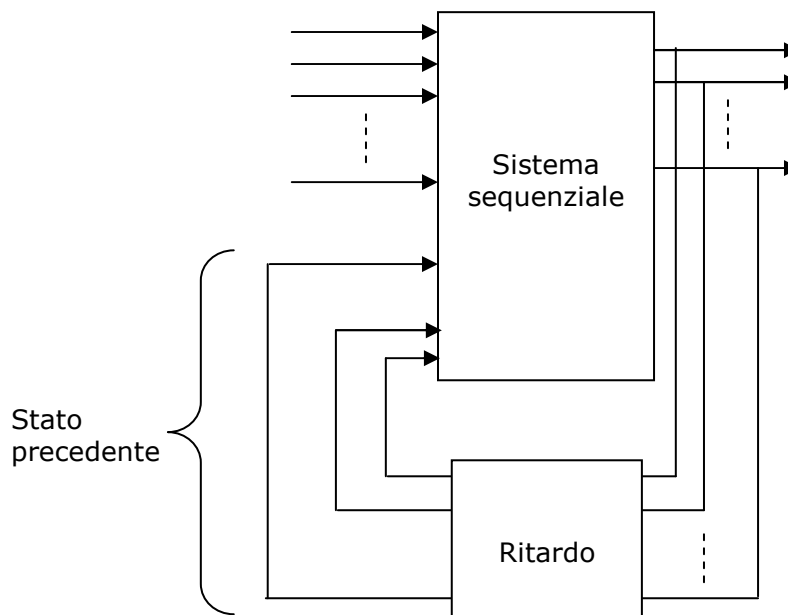
- sistemi combinatori
- sistemi sequenziali

Nei sistemi combinatori le uscite dipendono, istante per istante, esclusivamente dagli ingressi.



Nei sistemi sequenziali le uscite dipendono, istante per istante, dagli ingressi e dallo stato in cui si trova il sistema, determinato, in genere, dalle uscite all'istante precedente.

L'insieme delle uscite, riportate indietro con ritardo, determina lo stato.



Lo studio di un sistema sequenziale richiede che si determinino:

- gli stati in cui esso può trovarsi,
- le modalità di transizione di stato, vale a dire in corrispondenza di quali eventi il sistema cambia stato.

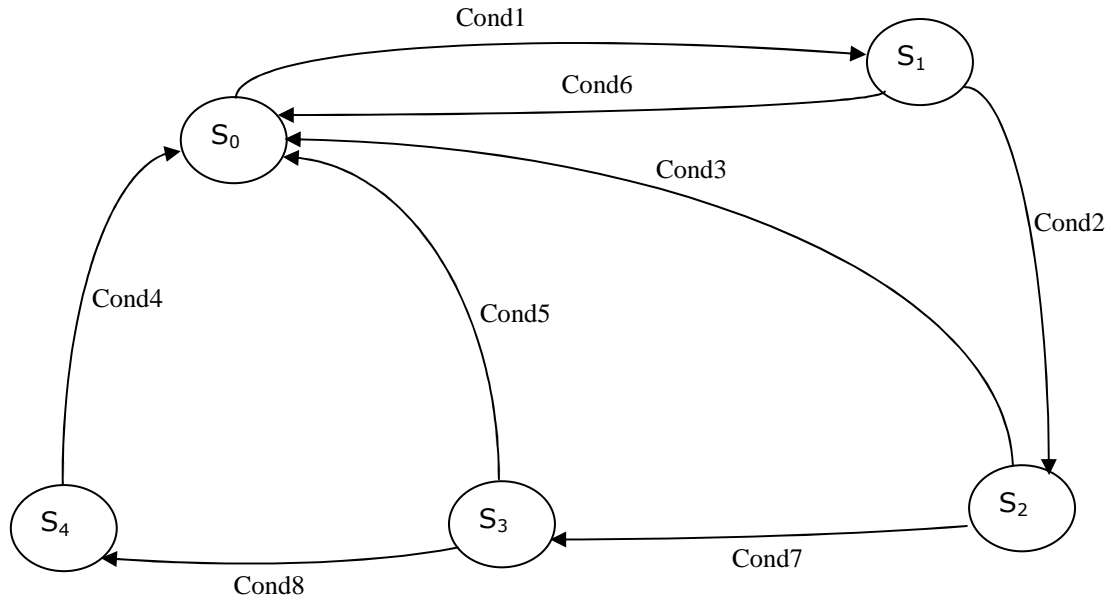
Un esempio di macchina sequenziale è un semaforo (passa attraverso stati successivi, determinati dalle lampade accese), ma anche un robot (per afferrare un oggetto, ad esempio, la sequenza di movimenti dipende dalla posizione da cui parte il braccio)

La descrizione degli stati può essere effettuata attraverso una tabella del tipo seguente:

Stati	Uscite			
	U_1	U_2	...	U_n
S_0				
S_1				
...				
S_m				

Ogni stato è definito da una n-pla di valori delle uscite.

La transizione da uno stato all'altro è invece descritta dal **Diagramma degli stati e delle transizioni**, in cui ogni stato è rappresentato da un cerchio e le transizioni sono rappresentate da archi orientati che uniscono lo stato di partenza e lo stato di arrivo; in corrispondenza dell'arco è riportata la condizione che determina la transizione, ad esempio un valore di ingresso o un intervallo di tempo che deve trascorrere perché avvenga la transizione.



Esempio

Realizzare un programma per controllare il sistema di illuminazione di un laboratorio, attivato da un pulsante.

Il sistema si compone di due file di neon. Quando tutte le luci sono spente, se si preme il pulsante, si accende la prima fila di neon, se lo si preme ancora, si accende la seconda fila e si spegne la prima, se lo si preme ancora, si accendono entrambe le file, se lo si preme ancora, tutte le luci si spengono e il ciclo si ripete.

Analisi

Il sistema è sequenziale. Si possono infatti rilevare facilmente, dalla traccia, gli stati successivi attraverso cui evolve il funzionamento del sistema.

Input

P = Pulsante (I0.0)

Output

Luci gruppo A = LGA (Q0.0)

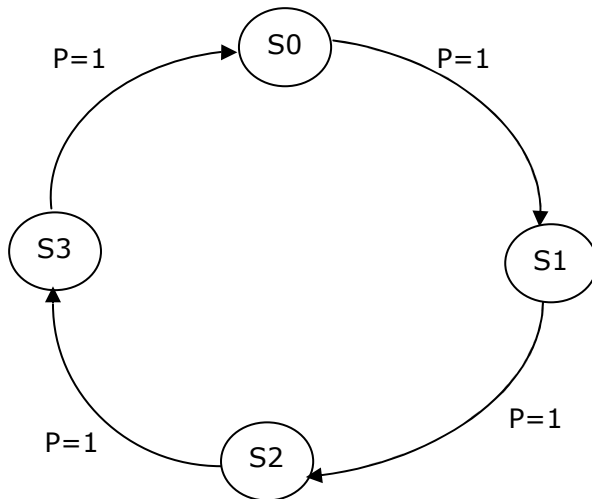
Luce gruppo B = LGB (Q0.1)

Stati

Gli stati sono 4:

STATO	G1	G2	DESCRIZIONE
S0	0	0	Luci spente
S1	1	0	Accese solo le luci del gruppo A
S2	0	1	Accese solo le luci del gruppo B
S3	1	1	Accese tutte le luci

Diagramma degli stati e delle transizioni



S0 è lo stato iniziale, tutte le luci sono spente, entrambi gli output sono a 0.

Quando si preme il pulsante, il sistema si porta nello stato S1 e l'uscita LGA assume il valore 1 (accese solo le luci del primo gruppo).

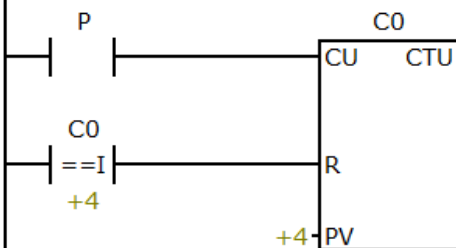
Premendo di nuovo il pulsante, il sistema si porta nello stato S2 e l'uscita LGB assume il valore 1, mentre LGA si riporta a 0 (accese solo le luci del secondo gruppo).

Premendo di nuovo il pulsante, il sistema si porta nello stato S3 e l'uscita LGB si riporta a 0 (spente tutte le luci).

Diagramma ladder

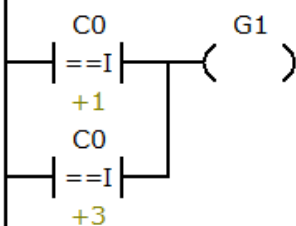
Segmento 1

Ogni volta che si preme il pulsante, cambia lo stato, dato dal conteggio del contatore C0, che quindi si incrementa di 1. Lo stato successivo a 3 è 0, per cui, quando C0 assume lo stato logico 1 (il conteggio è 4), si resetta e si torna allo stato 0.



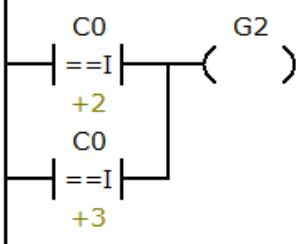
Segmento 2

Il primo gruppo di luci si deve accendere se lo stato è 1 o 3.



Segmento 3

Il secondo gruppo di luci si deve accendere se lo stato è 2 o 3.



INPUT E OUTPUT ANALOGICI

Sono segnali elettrici convertiti dal PLC in numeri e viceversa. Per il loro utilizzo è necessario che il PLC sia dotato di una scheda analogica. Nel nostro caso la scheda, che può essere aggiunta e rimossa, dispone di:

- 3 input analogici a 12 bit,
- 1 output analogico a 12 bit.

Ad un **input analogico** è generalmente collegato un trasduttore, che fornisce una misura di una grandezza fisica (analogica) e la trasforma in tensione variabile.



A differenza degli input digitali, che assumono solo due valori (on/off, 0/1), ad un input analogico arriva un valore numerico che esprime una misura, quindi non limitato a due valori. Grandezze fisiche possono essere: peso, pressione, luminosità, umidità, velocità, flusso, temperatura, profondità, inclinazione, accelerazione, forza, potenza, energia; qualunque "cosa" possa essere misurata.

Per ogni grandezza fisica si ha un diverso tipo di trasduttore. Il **tipo** di un trasduttore individua dunque il tipo di grandezza fisica che il trasduttore è in grado di misurare.

I trasduttori non misurano tutta l'estensione di valori che una grandezza può assumere, ma limitano la misura ad un **range** di possibili valori. Ad esempio, un trasduttore che rileva la temperatura all'interno di un ambiente domestico deve fornire valori nettamente diversi da quelli rilevati all'interno di un altoforno. O anche, nel caso di un trasduttore di peso, i valori da misurare in ambito chimico-farmaceutico è di gran lunga di ordine inferiore rispetto a quelli rilevati nella pesa di automezzi.

Quando un trasduttore rileva la misura di una grandezza fisica, fornisce un segnale elettrico che segue un preciso **standard di uscita elettrica**.

Gli standard più diffusi sono:

- $0 \div 10V$
- $-10 \div 10 V$
- $0 \div 5 V$
- $0 \div 20mA$
- $4 \div 20 mA$

I primi tre sono intervalli di valori di tensione, gli altri due intervalli di valori di corrente.

Il segnale elettrico fornito dal trasduttore è ovviamente funzione del valore misurato, in genere proporzionale ad esso.

I principali parametri che individuano un trasduttore sono pertanto: **tipo, range di misura, standard di uscita elettrica**. Ad essi si deve far riferimento in fase di acquisto del trasduttore.

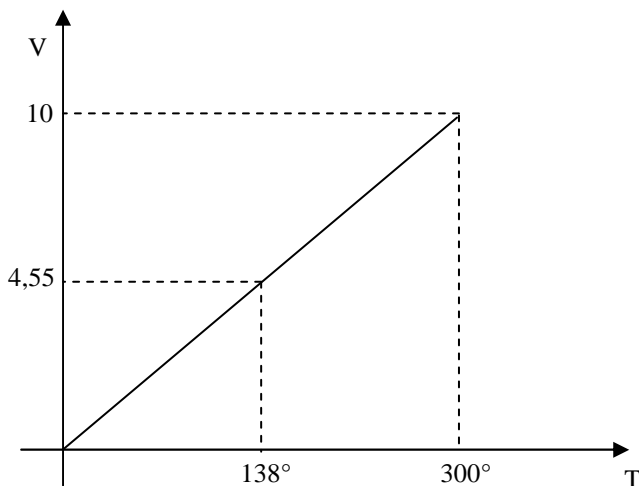
Esempio 1: temperatura in un forno domestico.

Caratteristiche del trasduttore:

Tipo: temperatura

Range: $0^{\circ} \div 300^{\circ}$

Standard di uscita elettrica: $0V \div 10V$



Un semplice calcolo consente di determinare il valore del segnale elettrico in uscita per i diversi valori di temperatura.

Ad ogni grado, corrisponde un incremento di valore di tensione pari a

$$10V/300=0,033V.$$

Quindi basta moltiplicare questo valore per la temperatura, per sapere qual è il corrispondente valore di tensione; ad esempio, il valore di tensione corrispondente a 138° è:

$$0,033V*138=4,55V$$

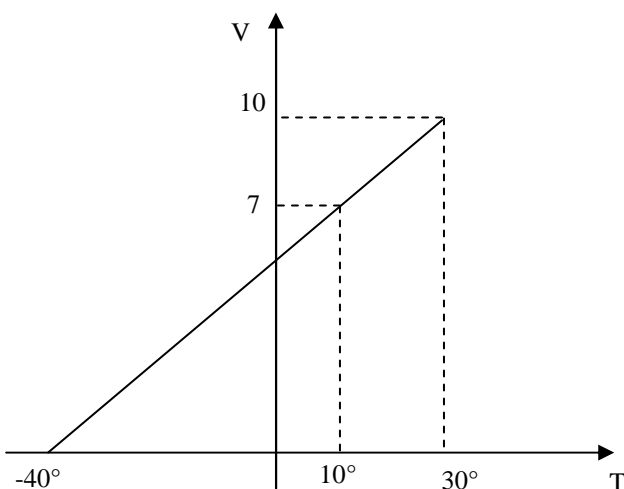
Esempio 2: temperatura in una cella frigorifera.

Caratteristiche del trasduttore:

Tipo: temperatura

Range: $-40^{\circ} \div 30^{\circ}$

Standard di uscita elettrica: $0V \div 10V$



Anche in questo caso il calcolo è semplice.

Ad ogni grado, corrisponde un incremento di valore di tensione pari a

$$10V/(30-(-40))=10V/70=0,14V.$$

Questa volta però si deve moltiplicare questo valore per la temperatura incrementata di 40, per sapere qual è il corrispondente valore di tensione; ad esempio, a 10° corrisponde un valore di tensione pari a:

$$0,14V*50=7V$$

Il segnale elettrico fornito da un trasduttore e che arriva al PLC attraverso un ingresso analogico è rappresentato all'interno del PLC stesso come numero binario e memorizzato in una variabile numerica di tipo Word (16 bit), per cui i valori rappresentabili vanno da -32768 a +32767. I nomi delle variabili di input sono:

AIW0, AIW2, AIW4, ...

La sigla AIW sta per Analogic Input Word. La numerazione di 2 in 2 è dovuta alla gestione in byte della memoria di PLC, per cui ogni variabile occupa 2 locazioni:

Byte 0
Byte 1
Byte 2
Byte 3
...

Mentre gli input analogici sono collegati a trasduttori, gli **output analogici** sono collegati ad azionamenti (sistemi di controllo che, in base al valore di tensione prodotto in uscita, controllano ulteriori circuiti). I nomi delle variabili di output sono:

AQW0, AQW2, AQW4, ...

Basta assegnare un valore ad una di queste variabili che automaticamente il PLC fornirà un'uscita (in corrente o in tensione) proporzionale al valore inserito nella variabile.

Oltre alle variabili per l'input e l'output, possono essere utilizzate altre **variabili interne**, che possono essere di vario tipo. Il nome della variabile include la specifica del tipo e dell'indirizzo.

Esempio:

VB1000 accede al byte di indirizzo 1000

VW1000 accede alla parola (16 bit) che occupa gli indirizzi 1000 e 1001

VD1000 accede alla doppia parola (32 bit) che occupa gli indirizzi 1000, 1001, 1002 e 1003.

Per le considerazioni fatte sulla struttura della memoria, la specifica dell'indirizzo tiene conto della dimensione del dato, per cui

- le variabili di tipo Byte sono VB0, VB1, VB2, ...
- le variabili di tipo Word sono VW0, VW2, VW4, ...
- le variabili di tipo Double sono VD0, VD4, VD8, ...

L'occupazione della memoria, che giustifica la numerazione, è la seguente:

Byte 0	VB0	VW0	VD0
Byte 1	VB1		
Byte 2	VB2	VW2	VD0
Byte 3	VB3		
Byte 4	VB4	VW4	VD4
Byte 5	VB5		
Byte 6	VB6	VW6	VD4
Byte 7	VB7		
...
...	
...
...	

Ai fini della programmazione del PLC, anziché operare sui valori di tensione o corrente, si fa riferimento ai valori numerici con cui essi vengono rappresentati.

Esempio

Si consideri il sistema di controllo di una temperatura in una serra. Un sensore di temperatura rileva valori compresi tra -20° e $+60^{\circ}$. Si vuole che il sistema funzioni nel seguente modo:

- se la temperatura rilevata è al di sotto di 5° , si accende un bruciatore;
- se la temperatura rilevata è tra 5° e 30° , non succede nulla;
- se la temperatura rilevata è al di sopra di 30° , si attiva un sistema di ventilazione.

Una chiave attiva/disattiva il sistema.

Realizzare il programma per controllare il sistema attraverso un PLC.

Supponiamo di utilizzare un trasduttore con standard di uscita elettrica $0 \div 5$ V.

Le caratteristiche del trasduttore sono dunque

Tipo: temperatura

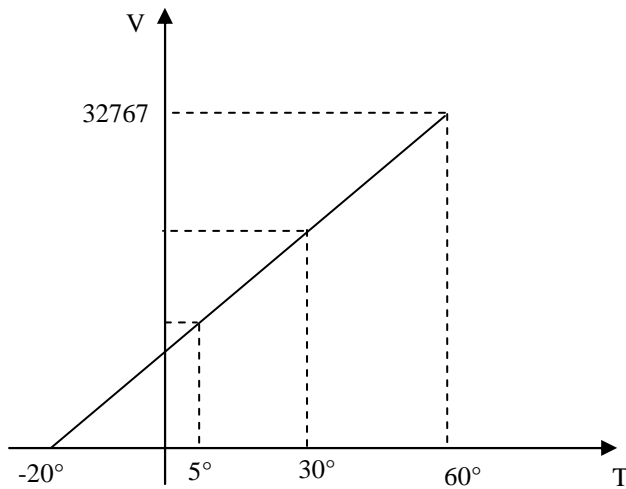
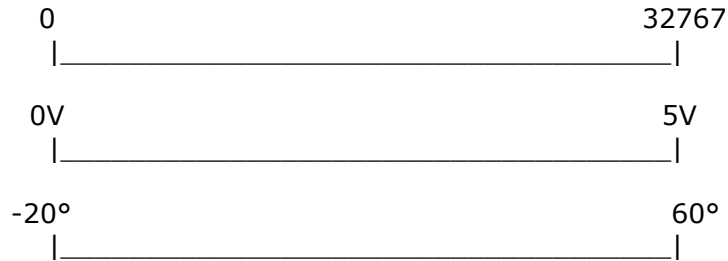
Range: $-20^{\circ} \div 60^{\circ}$

Standard di uscita elettrica: $0V \div 5V$

La risoluzione del problema richiede che innanzitutto si determinino i valori di tensione a cui corrispondono i valori di temperatura 5° e 30° . Il PLC rappresenta i valori di tensione con

numeri da -32768 a 32767. Essendo in questo caso i valori di tensione positivi (da 0 a 5V), i numeri che si otterranno vanno da 0 a 32767.

Dobbiamo quindi trovare i numeri interi corrispondenti ai due valori di tensione, a loro volta corrispondenti ai due valori di temperatura, sapendo che:



Il procedimento è sempre lo stesso.

Ad ogni grado, corrisponde un incremento di valore numerico di tensione pari a:

$$32767 / (60 - (-20)) = 32767 / 80 = 409,6.$$

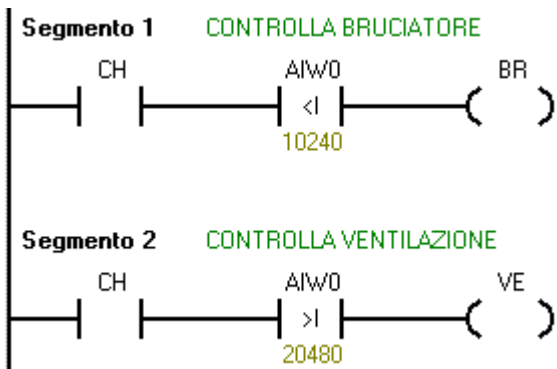
Si deve poi moltiplicare questo valore per ciascun valore di temperatura incrementato di 20, per sapere quali sono i corrispondenti valori numerici (interi) di tensione. Quindi:

a 5° corrisponde $409,6 * 25 = 10240$

a 5° corrisponde $409,6 * 50 = 20480$

A questo punto è possibile realizzare il diagramma ladder, dove:

- AIW0 è l'input analogico a cui arriva il segnale prodotto da un sensore di temperatura.
- Q0.0 è l'output digitale che accende il bruciatore.
- Q0.1 è l'output digitale che attiva il sistema di ventilazione.



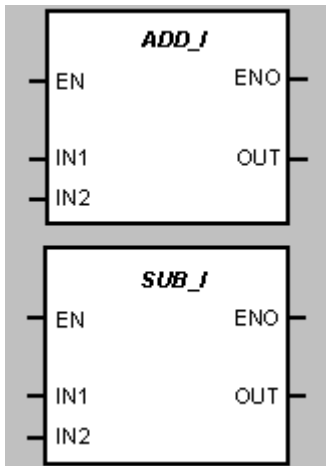
Se la temperatura è minore di 5° (il numero intero che lo rappresenta è minore di 10240), il bruciatore viene acceso.

Se la temperatura è maggiore di 30° (il numero intero che lo rappresenta è maggiore o uguale a 20480), il sistema di ventilazione viene attivato.

OPERAZIONI MATEMATICHE

Sono realizzate attraverso i seguenti operatori: ADD (addizione), SUB (sottrazione), MUL (moltiplicazione) e DIV (divisione).

Addizione e sottrazione

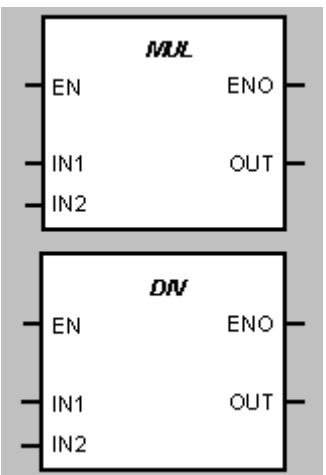


Le funzioni di addizione e sottrazione sommano o sottraggono IN1 e IN2 e collocano il risultato in OUT. EN è l'ingresso che abilita l'operatore. L'uscita ENO assume valore 0 in caso di errore (ad esempio, overflow).

Sono disponibili tre operatori di addizione:
ADD_I, ADD_DI e ADD_R
e tre di sottrazione:
SUB_I, SUB_DI e SUB_R.

ADD_I e SUB_I operano su due numeri interi a 16 bit e producono in uscita un numero intero a 16 bit.
ADD_DI e SUB_DI operano su due numeri interi a 32 bit e producono in uscita un numero intero a 32 bit.
ADD_R e SUB_R operano su numeri reali a 32 bit e producono in uscita un numero reale a 32 bit.

Moltiplicazione e divisione



La funzione di Moltiplicazione moltiplica IN1 e IN2 e colloca il risultato nella variabile specificata da OUT.

La funzione di Divisione divide IN1 per IN2 e colloca il risultato nella variabile specificata da OUT.

EN è l'ingresso che abilita l'operatore. L'uscita ENO assume valore 0 in caso di errore (ad esempio, overflow o divisione per zero).

Sono disponibili quattro operatori di moltiplicazione:
MUL, MUL_I, MUL_DI, MUL_R
e quattro di divisione:
DIV, DIV_I, DIV_DI e DIV_R.

MUL moltiplica due numeri interi a 16 bit e dà un risultato a 32 bit.
DIV divide due numeri interi a 16 bit e dà un risultato a 32 bit, costituito da un resto a 16 bit (i 16 bit più significativi) e un quoziente a 16 bit (i 16 bit meno significativi).

MUL_I moltiplica due numeri interi a 16 bit e produce un risultato a 16 bit.

DIV_I divide due numeri interi a 16 bit e produce un quoziente a 16 bit. L'eventuale resto viene eliminato.

MUL_DI moltiplica due numeri interi a 32 bit e produce un risultato a 32 bit.

DIV_DI divide due numeri interi a 32 bit e produce un quoziente a 32 bit. L'eventuale resto viene eliminato.

MUL_R moltiplica due numeri reali a 32 bit e produce come risultato un numero reale a 32 bit.

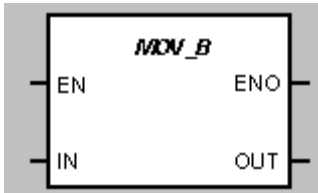
DIV_R divide tra loro due numeri reali a 32 bit e produce come risultato un quoziente in numero reale a 32 bit.

Solo MUL e DIV sono disponibili nella CPU 214

OPERAZIONI DI TRASFERIMENTO

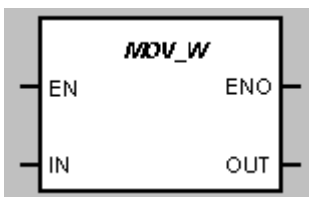
Sono realizzate attraverso i seguenti operatori: MOV_B, MOV_W, MOV_DW, MOV_R.

MOV_B



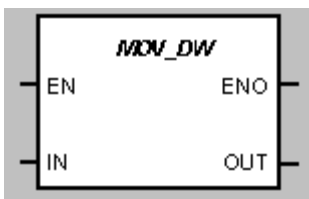
Trasferisce il byte di ingresso (IN) nel byte di uscita (OUT).
Il byte di ingresso non viene modificato dall'operazione.

MOV_W



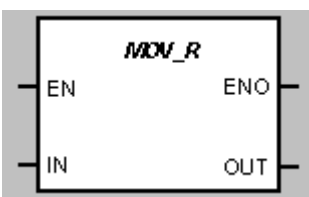
Trasferisce la parola di ingresso (IN) nella parola di uscita (OUT).
La parola di ingresso non viene modificata dall'operazione.

MOV_DW



Trasferisce la doppia parola di ingresso (IN) nella doppia parola di uscita (OUT).
La doppia parola di ingresso non viene modificata dall'operazione.

MOV_R



Trasferisce il numero reale (32 bit) di ingresso (IN) nella doppia parola di uscita a 32 bit (OUT).
Il numero reale in ingresso non viene modificato dall'operazione.