

# **IL PLC**

**Controllore a logica programmabile**

**Prof. Antonio Messina**

# IL 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



## PERCHÉ IL PLC ?

### Perché offre . . .

- AFFIDABILITÀ
- FLESSIBILITÀ
- SEMPLICITÀ D' USO
- FACILE MANUTENIBILITÀ
- ECONOMICITÀ
- ESPANDIBILITÀ
- NOTEVOLI POTENZIALITÀ
- DIAGNOSTICA SOFISTICATA

## DOVE USARE 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

## TIPICHE APPLICAZIONI 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

# STRUTTURA DI UN PLC

ALIMENTATORE

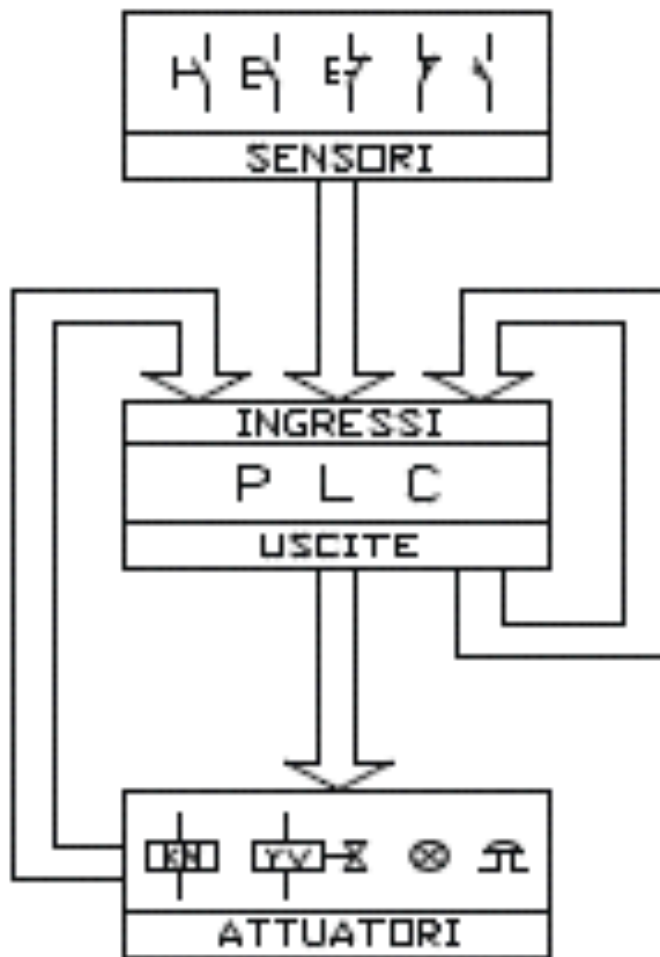
UNITÀ CENTRALE

MEMORIA DATI E MEMORIA PROGRAMMI

UNITÀ DI INPUT/OUTPUT

PERIFERICHE

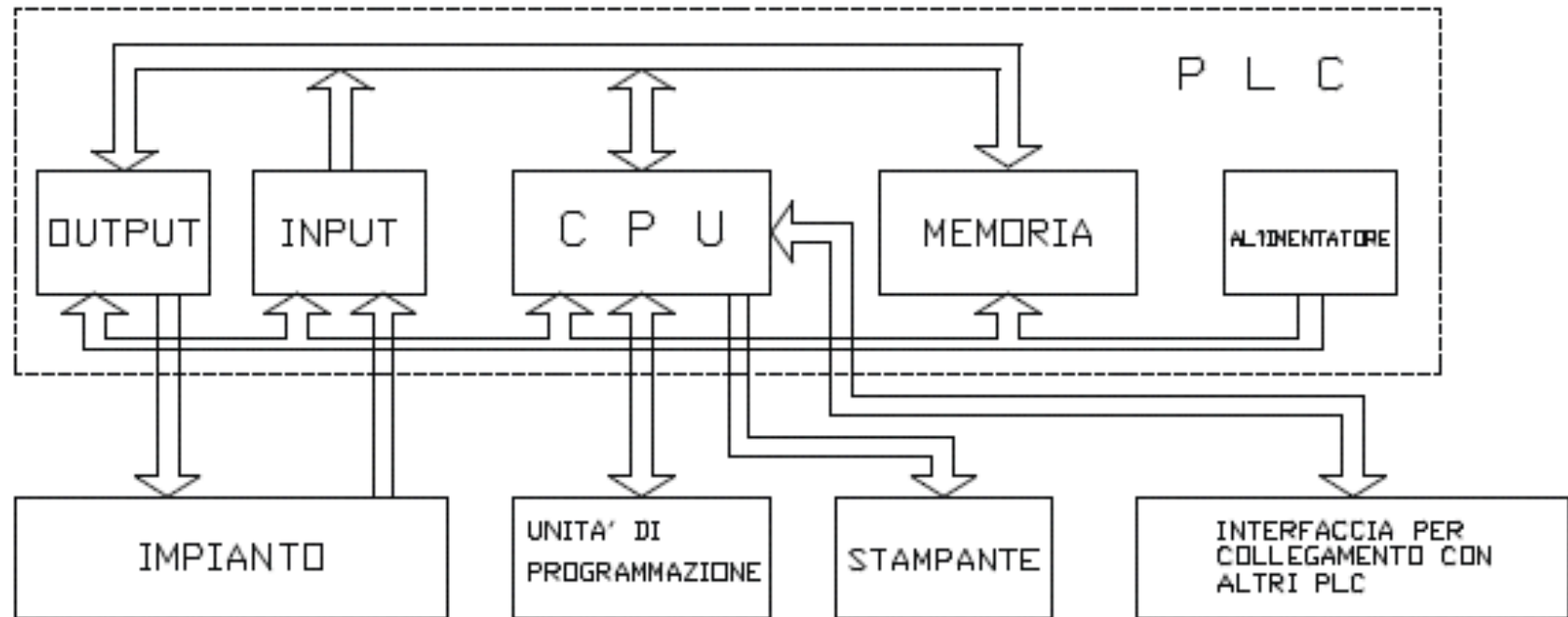
# CONTROLLO DI UN PROCESSO



Per controllare una macchina o un processo il PLC deve acquisirne lo stato istante per istante e in base alle istruzioni scritte nella memoria determinare se devono essere apportate delle modifiche allo stato attuale delle uscite

# ARCHITETTURA DI UN SISTEMA A PLC

Per controllare una macchina o un processo oltre ai moduli I/O è dotato di CPU, Memorie, alimentatore, Sistema bus suddiviso in più gruppi di segnali per la trasmissione e lo scambio di segnali tra microprocessore e schede di ingresso e uscite.





# ALIMENTATORE

- Provvede a fornire i **corretti livelli di tensione** per il funzionamento dei vari dispositivi elettronici
- Esistono diversi modelli, in funzione della tensione di rete:
  - **110 Vac**
  - **230 Vac**
  - **24 Vdc**

# 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***

# MEMORIA

Il **PLC** ha bisogno di **memoria** sia per il proprio sistema operativo sia per la memorizzazione del programma utente e sia per l'elaborazione dei dati intermedi durante l'esecuzione del programma.

Di solito il costruttore utilizza, per la memorizzazione del sistema operativo, Una memoria di tipo **ROM** (Read Only Memory), che ha appunto le caratteristiche di essere non volatile e di non poter essere modificata visto che è una Memoria di sola lettura.

Sia per il programma utente, che può essere modificato dall'utente in qualsiasi momento, sia per la memorizzazione dei risultati intermedi il costruttore fornisce una memoria **RAM** (random Acces Memory) che può essere letta e riscritta , ma che ha bisogno di una batteria tampone per mantenere i dati quando si toglie l'alimentazione.

## **ASPETTO FUNZIONALE DELLE MEMORIE**

In base al loro impiego le memorie in un PLC si possono distinguere in:

**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) scritte dall'utente  
che verrà eseguita dalla CPU

Viene realizzata con memorie di tipo RAM

# MEMORIA DI LAVORO

E' quella memoria che prevede due distinte sezioni:

## I FLAG e i REGISTRI

- I FLAG o MERKER

l'accesso a questa parte di memoria può avvenire per :

**BIT, BYTE (8bit), WORD(16bit) o DOUBLE WORD(32bit)**

Può essere utilizzata dall'utente per memorizzare risultati intermedi durante l'elaborazione del programma che possono essere utilizzati successivamente in altre parti del programma. I singoli BIT possono essere **SETTATI (1)** o **RESETTATI (0)**

# I REGISTRI

E' una RAM utilizzata per alcune funzioni durante l'esecuzione del programma.

Di solito si hanno REGISTRI::

1.Speciali di sistema ; mantiene traccia dei lavori di processo della CPU e non è direttamente accessibile dai moduli input e output

2.Di input ; accessibile dai moduli input. Ad ogni modulo input corrisponde un registro. Ciascun bit di stato del registro controlla lo stato di un ingresso. Quando l'ingresso è ON viene messo a UNO il bit della casella corrispondente. Se OFF a ZERO

3.Di output ; ha le stesse caratteristiche di quello input. Quando il bit è a UNO il corrispondente output viene messo ON. Se ZERO a OFF



# STRUTTURA DI UN PLC

## 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);

# STRUTTURA DI UN PLC

## MODULI DI I/O

### SCHEDE DI INGRESSO DIGITALI

Permettono il collegamento del PLC al mondo esterno.  
Convertono lo stato dell'ingresso associato in uno stato logico (1 -0) interpretabile dalla CPU

### SCHEDE DI USCITA DIGITALI

Convertono gli stati logici presenti nella memoria dati di Output in segnali elettrici che commutano fisicamente il punto di uscita

#### **Sono disponibili:**

- Moduli di INGRESSO DIGITALE e ANALOGICI
- Moduli di USCITA DIGITALE e ANALOGICI
  
- Moduli di comunicazione che gestiscono i *protocolli di comunicazione* per le diverse tipologie di reti informatiche per lo scambio di dati, messaggi, informazioni tra un insieme di PLC
- Moduli speciali (AD-DA, Contatori veloci, PID, ...)

# STRUTTURA DI UN PLC

## PERIFERICHE

Permettono il "colloquio" tra l'operatore (programmatore) ed il PLC

Console di programmazione

Console di programmazione grafica

Interfaccia per personal computer

Interfaccia stampante

Programmatore di EPROM

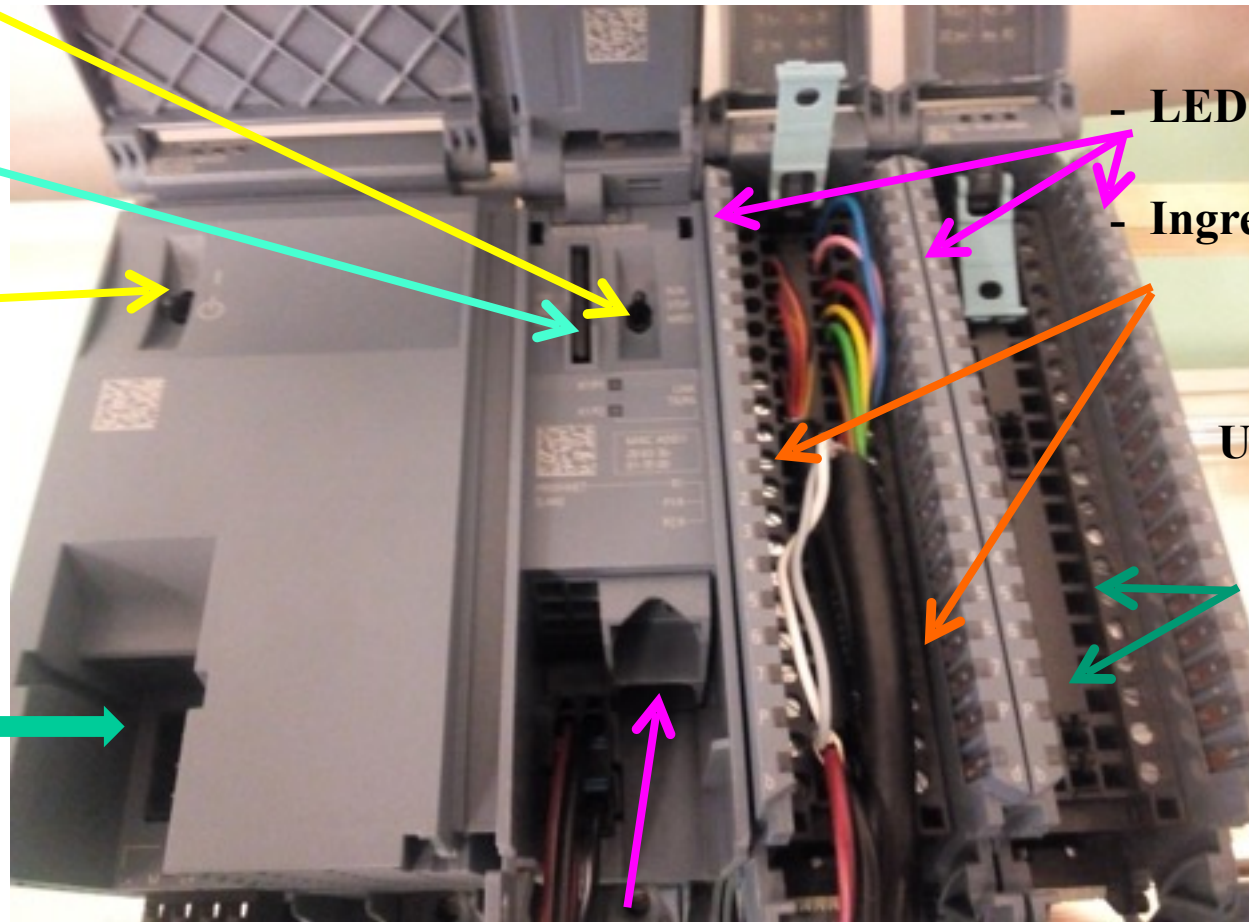
## ARMADIO (CESTELLO O RACK)

-Interruttore  
RUN-STOP  
Memory Card

-Interruttore  
ON / OFF



Alimentazione  
230V e 24V



- LED di stato

- Ingressi

Uscite

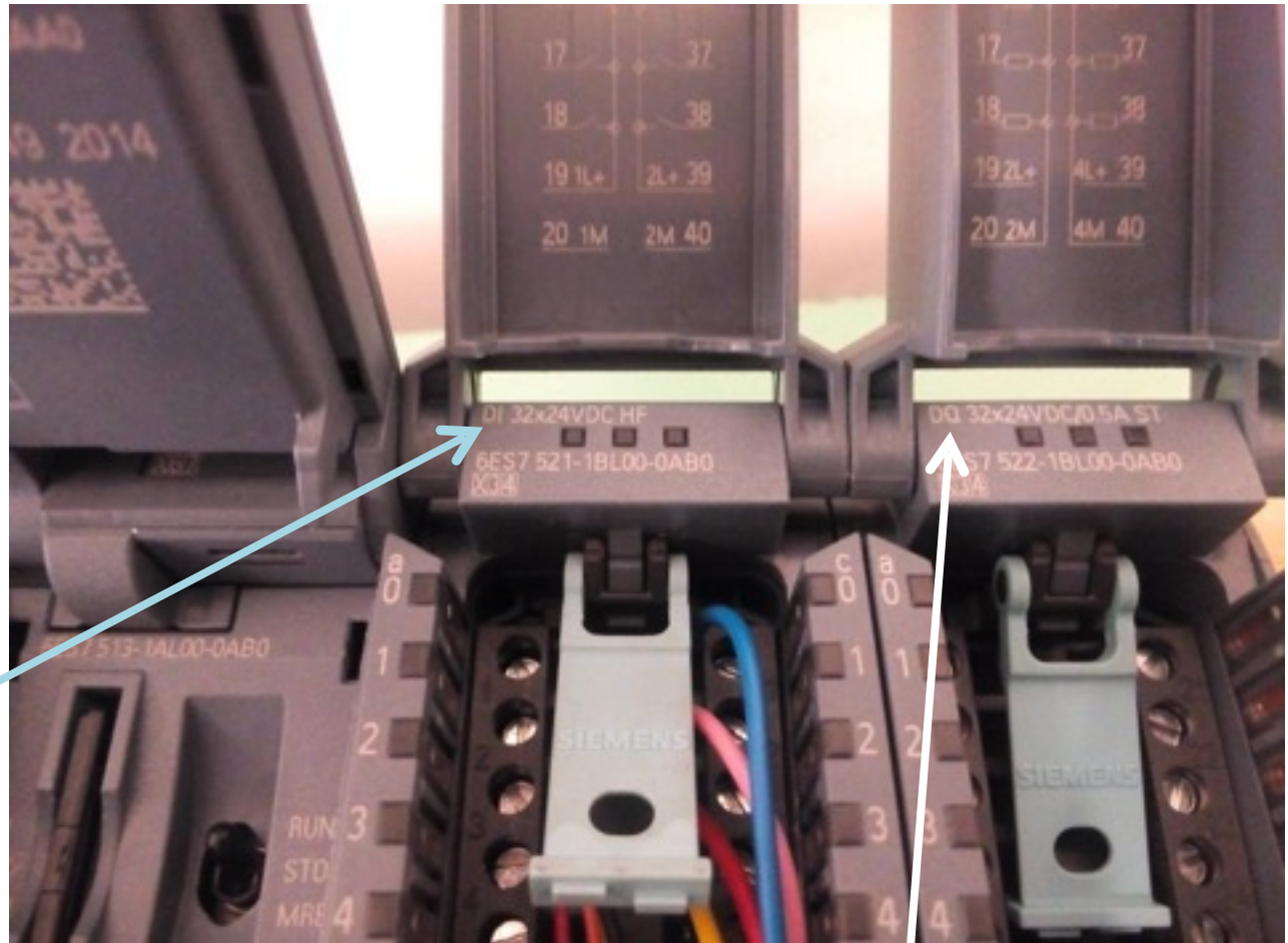
- Porta di comunicazione  
TCP/IP

# COME SI PRESENTA UN PLC

## MODULI I/O

- schede che permettono l'interfacciamento del modulo processore
- con il mondo esterno

32 Ingressi  
+ alimentazione



32 Uscite + alimentazione

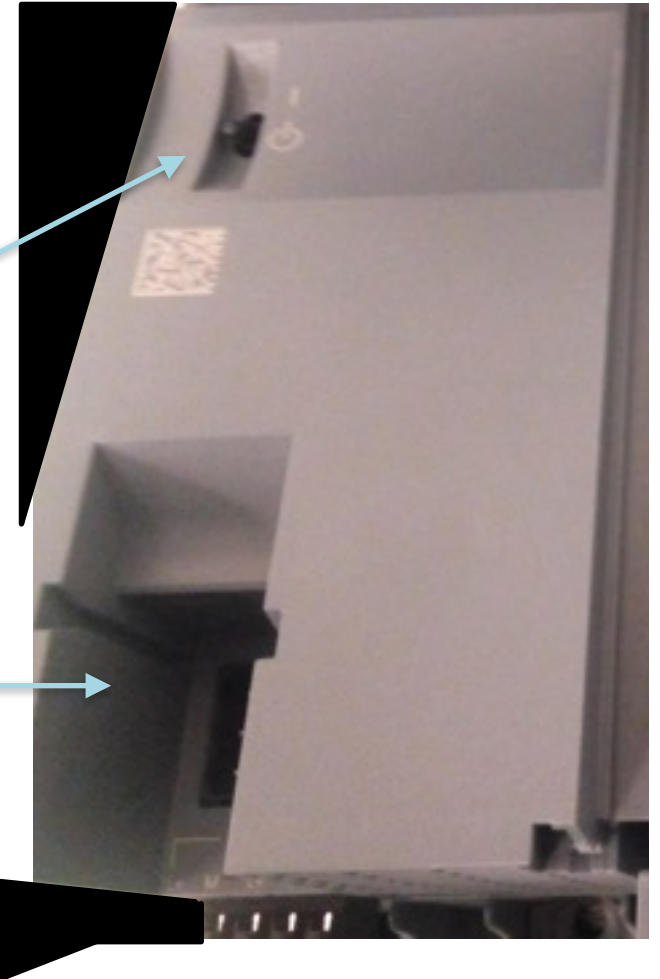
# COME SI PRESENTA UN PLC

## ALIMENTATORE

alimentazione  
per tutte le  
schede  
presenti nel  
cestello



ON OFF



## MODULI DI INGRESSO DIGITALE

- Vengono collegati a questo modulo i sensori come finecorsa, pulsanti, trasduttori ecc.
- Convertono una tensione *presente o assente* in uno stato logico (**1 – 0**) interpretabile dalla CPU

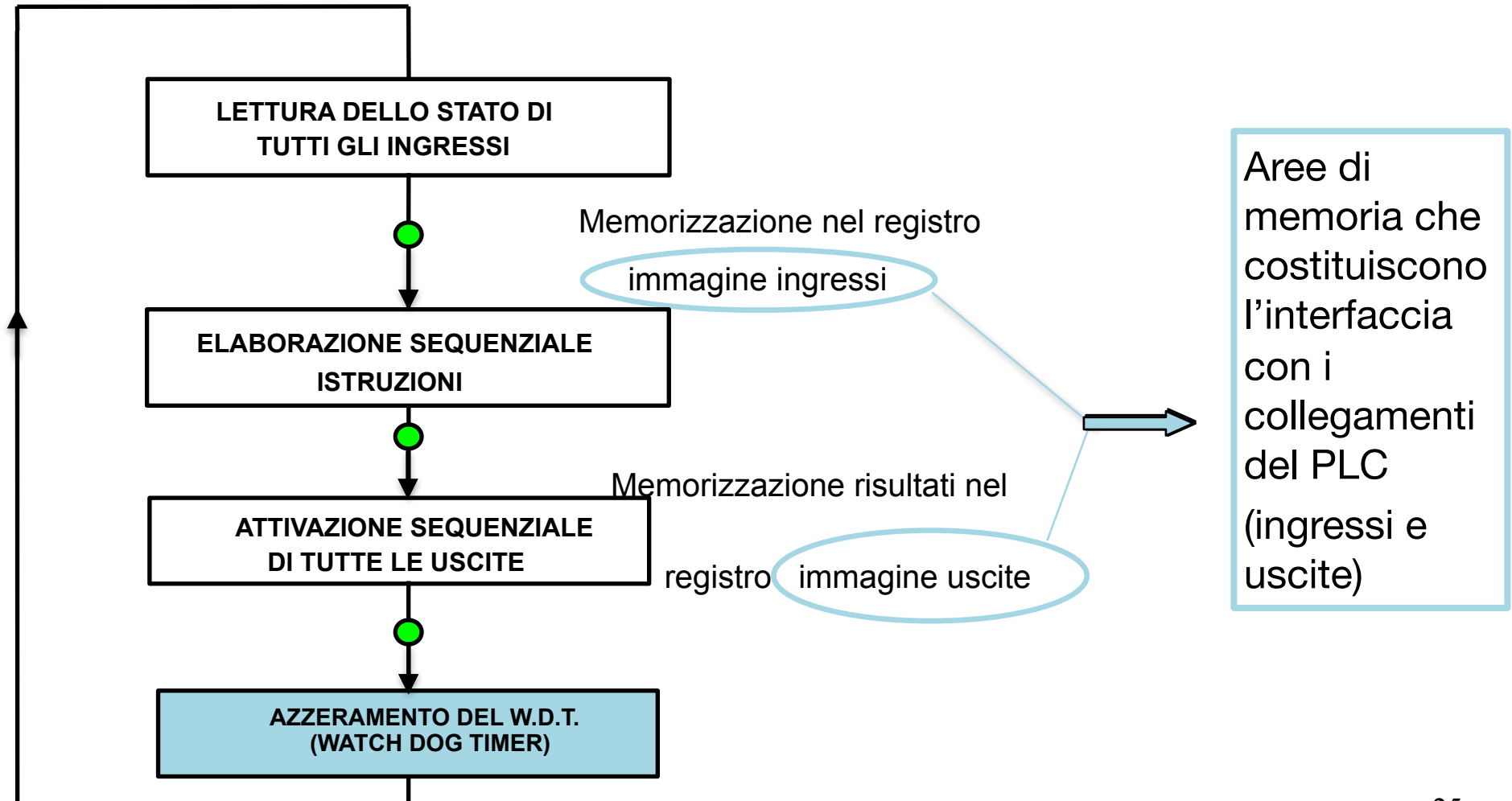
# MODULI DI USCITA DIGITALE

- Attivano gli attuatori del sistema (contattori, relè, lampade, ecc.) in base agli stati logici **1-0** (tensione *presente o assente*) situati nella memoria dati di output



# LA SCANSIONE DEL PLC

## Scansione sincrona di ingresso e di uscita



# INDIRIZZAMENTO DEL PLC S7 - 1500

L'indirizzo **I 0.1** indica:

- I** blocco di ingresso
- 0** indirizzo a byte (numero di un gruppo di 8 bit) 1° blocco ingressi
- 1** numero del bit in un byte

L'indirizzo **Q 0.2** indica:

- Q** blocco di uscita
- 0** indirizzo a byte (numero di un gruppo di 8 bit) 1° blocco uscite
- 2** numero del bit in un byte

# INDIRIZZAMENTO DEL PLC S7 - 1500

Aree di indirizzamento del PLC

**I0.1**



7 6 5 4 3 2 1 0

Indirizzo a BYTE →

Blocchi  
Ingressi

**I0.**

I1.

I2.

I3.

I4.

I5.

I6.

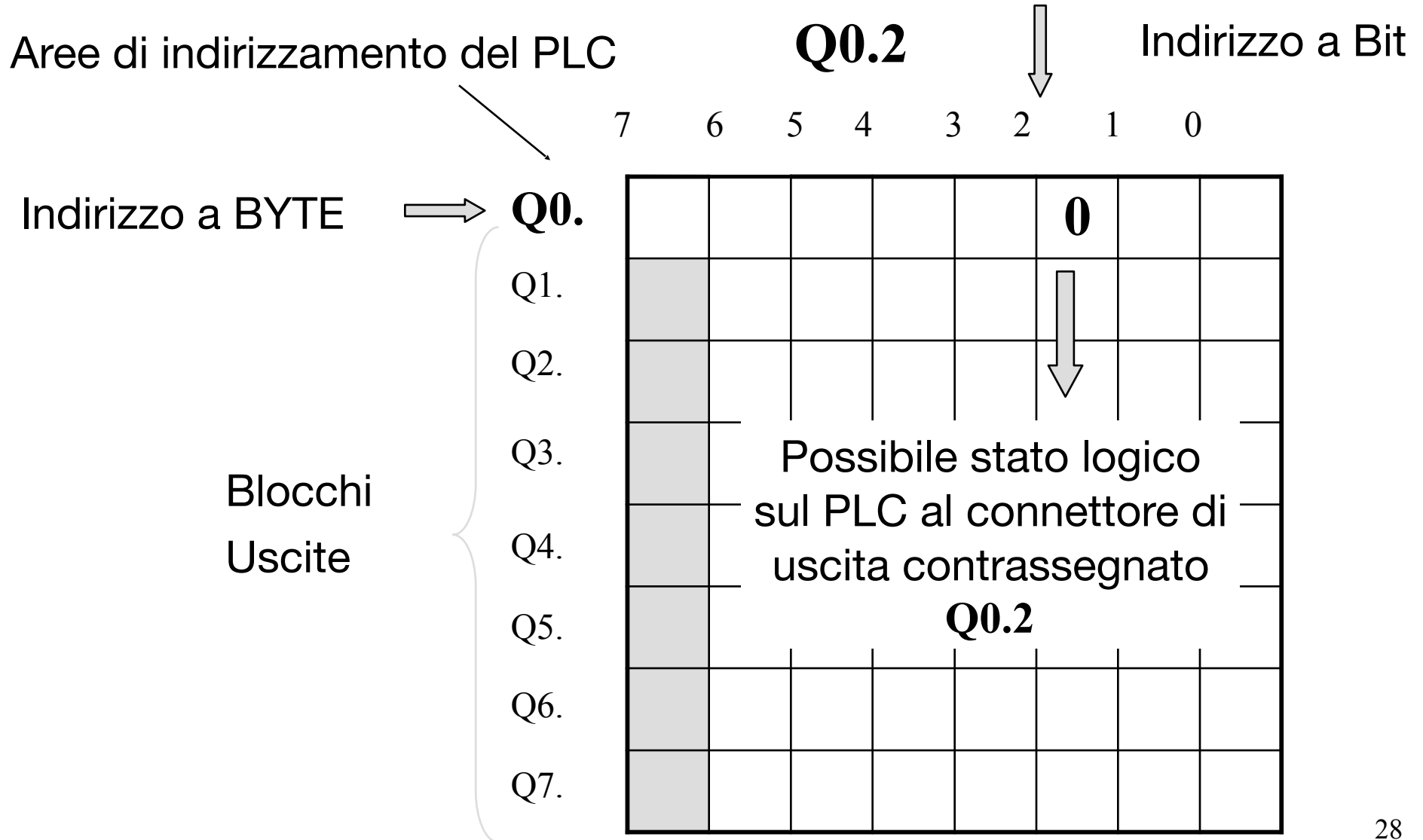
I7.

							<b>1</b>	

Possibile stato logico sul PLC al  
connettore di ingresso  
contrassegnato **I0.1**



# INDIRIZZAMENTO DEL PLC S7 - 1500



# INDIRIZZAMENTO DEL PLC S7 - 1500

Le aree di indirizzamento sono aree di memoria del PLC nei quali vengono riprodotti i singoli ingressi e le singole uscite.

Se ad esempio su un **ingresso è presente tensione**, questo segnale “1” viene riprodotto nella memoria all’indirizzo dell’ingresso.

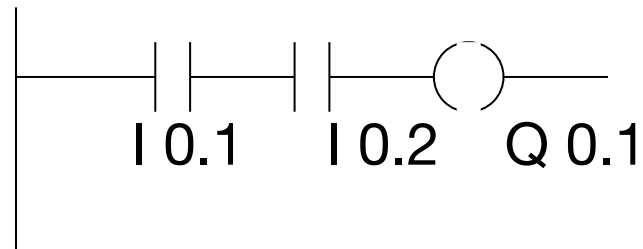
Se ad esempio su una **uscita non è presente tensione**, questo segnale “0” viene riprodotto nella memoria all’indirizzo dell’uscita.

# LINGUAGGIO DI PROGRAMMAZIONE STEP 7

Per scrivere un programma esistono alcuni modi di rappresentazione. I più utilizzati sono:

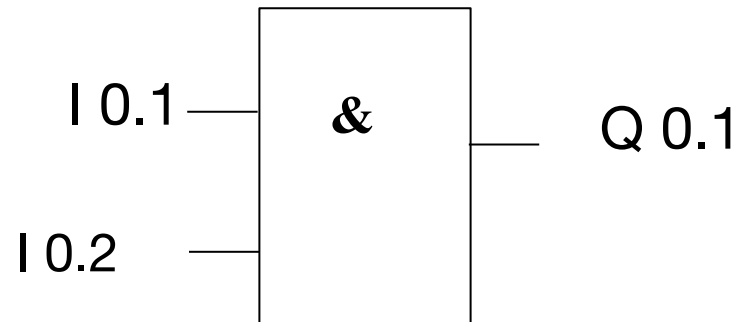
KOP

SCHEMA A CONTATTI



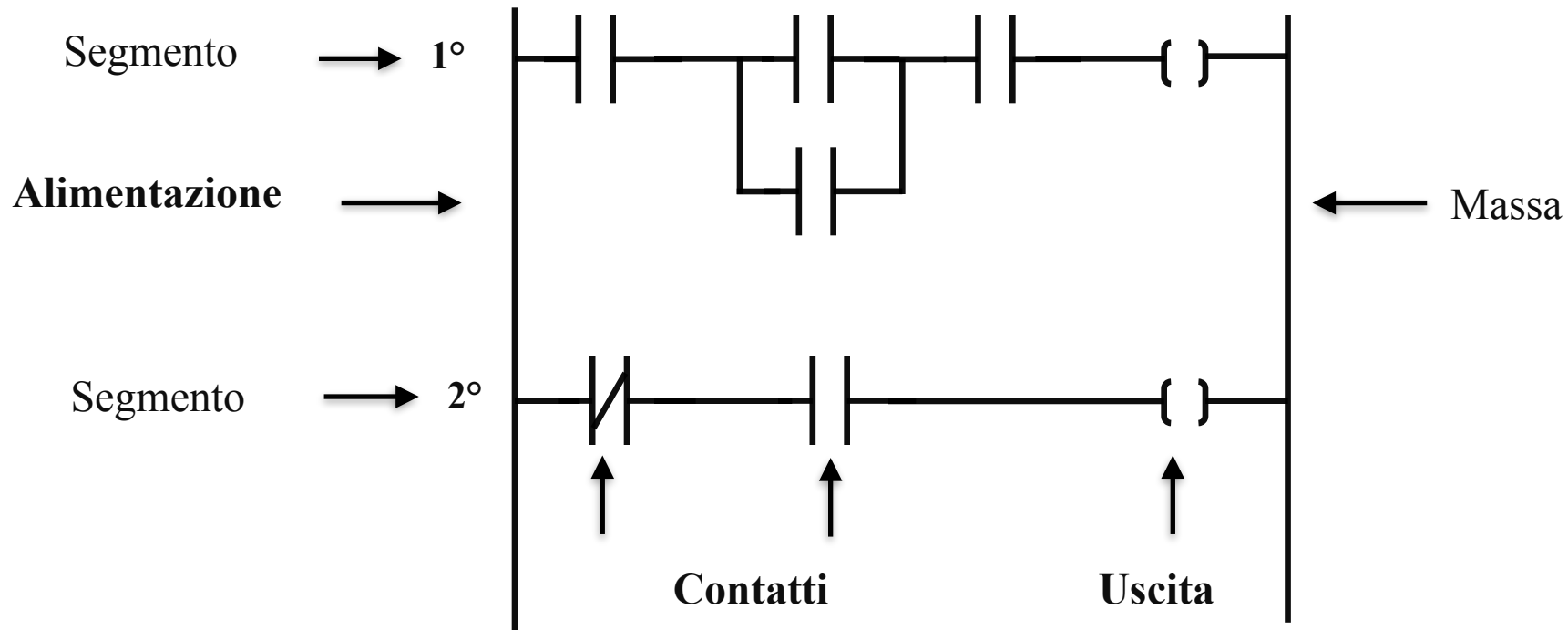
FUP

SCHEMA LOGICO

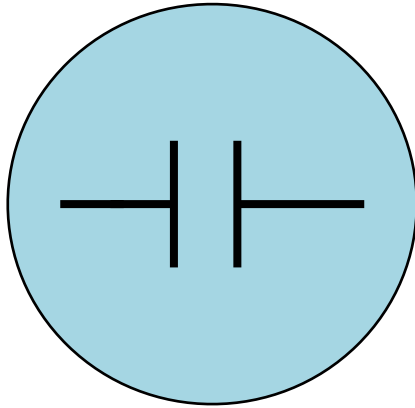


# LINGUAGGIO DI PROGRAMMAZIONE KOP

il linguaggio più diffuso è il KOP detto anche LADDER che utilizza una serie di segni grafici e da due linee verticali e da linee orizzontali sulle quali vengono disegnati gli elementi che costituiscono l'impianto da controllare.



## ANALIZZIAMO ORA IL SIGNIFICATO DEI SIMBOLI UTILIZZATI



Questo simbolo, presente nel linguaggio KOP , rappresenta un contatto generico di un qualsiasi componente, a cui è associato un livello logico che sarà letto dal PLC.

### Cosa fa il PLC ?

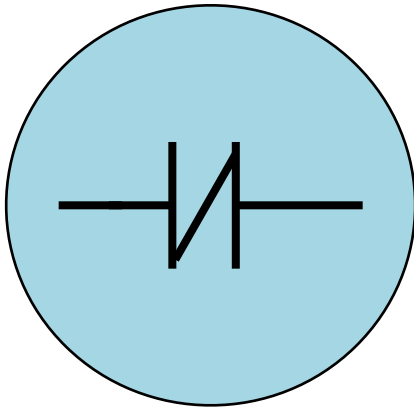
Es.

Se al simbolo è associato un pulsante di stop o uno di marcia  
Un pulsante di **STOP**, che è normalmente chiuso (**N.C.**) collegato ad un ingresso del PLC sarà letto come livello logico **1**, mentre un pulsante di **MARCIA**, normalmente aperto (**N.A.**), sarà letto come **0**.

In pratica il PLC legge semplicemente lo stato logico che c'è in quell'istante e lo acquisisce.



## ANALIZZIAMO ORA IL SIGNIFICATO DEI SIMBOLI UTILIZZATI



Questo simbolo invece rappresenta un contatto generico **NEGATO**

### Cosa fa il PLC ?

Es.

Se al simbolo è associata un'uscita o un contatto di un sensore

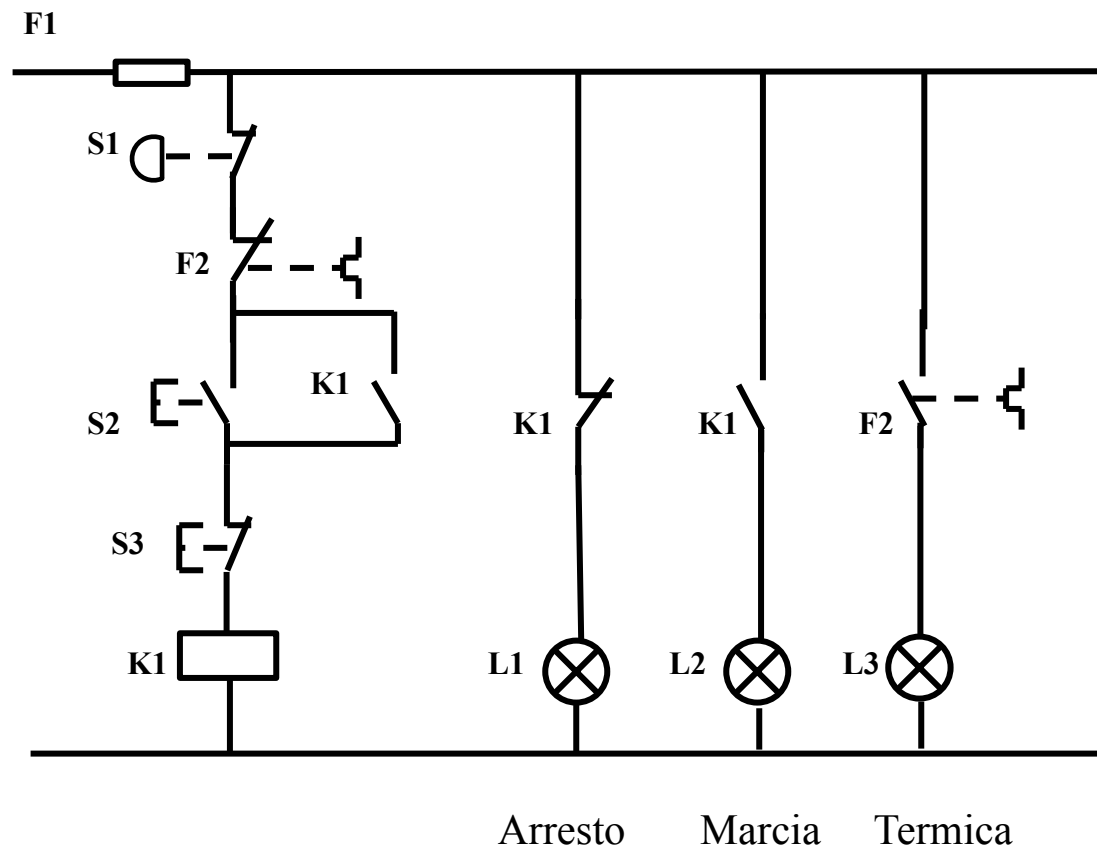
Se lo stato logico rilevato sull'uscita è **0** il PLC negherà il risultato e darà **1**

Se lo stato logico rilevato è **1** il PLC negherà il risultato e darà **0**

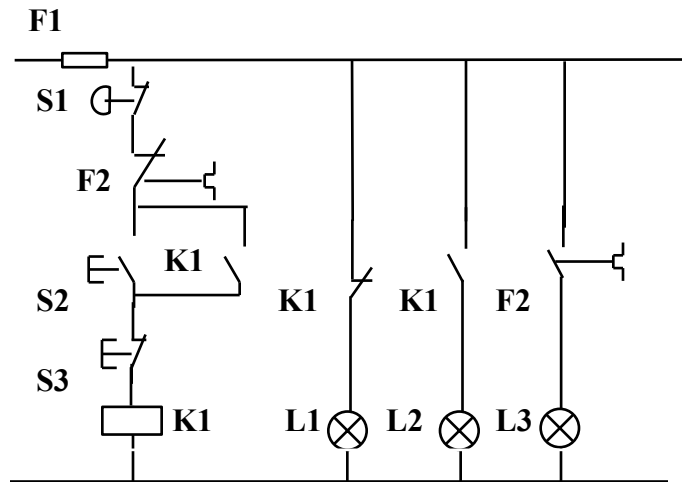
Lo stesso sarà per il contatto di un sensore (es. finecorsa, fotocellula...)

In pratica se un contatto di un sensore collegato ad un ingresso, è normalmente chiuso (N.C.) sarà letto dal PLC come **0**, mentre un contatto normalmente aperto (N.A.), sarà letto come **1**

Prendiamo in esame il semplice schema di un **tele-**  
**avviamento** di un motore asincrono trifase.



## Costruiamo la tabella delle assegnazioni relativa agli **INGRESSI**



Nome	Indirizzo	Stato logico	Funzione
S1	I0.0	n.c. =1	Pulsante di alt generale
S2	I0.1	n.a. =0	Pulsante di marcia
S3	I0.2	n.c. =1	Pulsante di stop
F2	I0.3	n.c. =1	Relè termico

La tabella relativa alle uscite potrà assumere la forma:

N.B.

Si nota che nella tabella delle uscite, K1 è presente ben quattro volte. Normalmente si segna nella tabella solo un K1, considerato che l'indirizzo (Q0.0) è unico. Quando si scriverà la lista istruzioni tutti i contatti relativi a K1 avranno, ovviamente, lo stesso indirizzo della bobina del contattore K1.

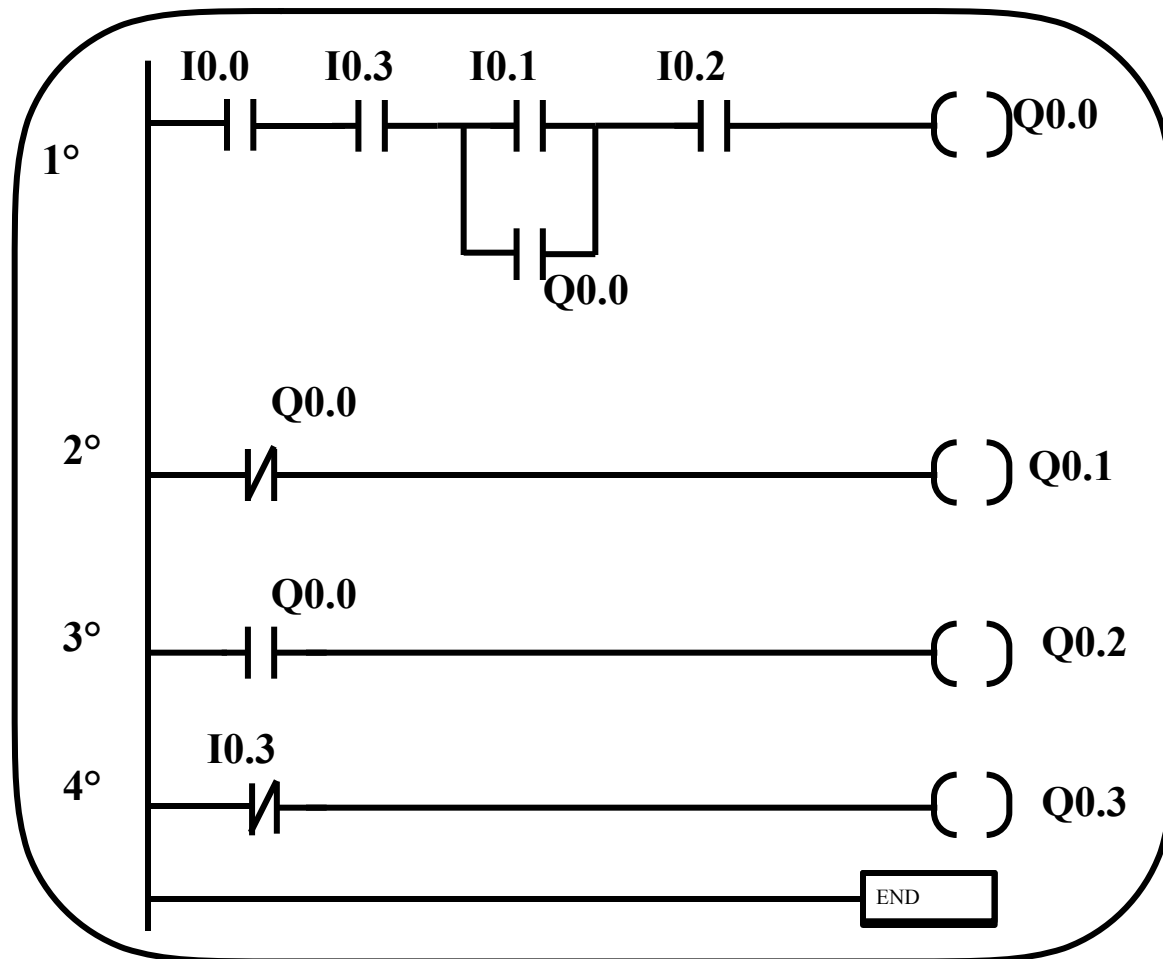
Nome	Indirizzo	Stato logico	Funzione
K1	Q 0.0	0	Bobina contattore K1 Marcia avanti
K1	Q 0.0		Contatto auto ritenuta
K1	Q0.0		Contatto per L1 Arresto
K1	Q0.0		Contatto per L2 Marcia
<b>tabella precedente</b>			

Nome	Indirizzo	Stato logico	Funzione
K1	Q0.0	0	Bobina contattore K1 Marcia avanti
<b>tabella semplificata</b>			

A queste tabelle si aggiunge quella delle segnalazioni che fanno parte anch'esse delle uscite:

<b>Nome</b>	<b>Indirizzo</b>	<b>Stato logico</b>	<b>Funzione</b>
<b>L1</b>	<b>Q0.1</b>	<b>0</b>	<b>Arresto</b>
<b>L2</b>	<b>Q0.2</b>	<b>0</b>	<b>Marcia avanti</b>
<b>L3</b>	<b>Q0.3</b>	<b>0</b>	<b>Intervento termico L3</b>

Una volta realizzate le tabelle di assegnazione possiamo costruire lo schema KOP il quale si identifica con quello di comando utilizzato nella logica cablata.



#### Come si legge lo schema KOP

Se S1 di alt (I0.0), il contatto della termica (I0.3), S2 di marcia (I0.1) e S3 di stop (I0.2) sono chiusi, allora eccita K1 (Q0.0) e autoritienilo.

Se uno solo dei contatti I0.0, I0.3, I0.1, I0.2 è aperto non eccitare K1

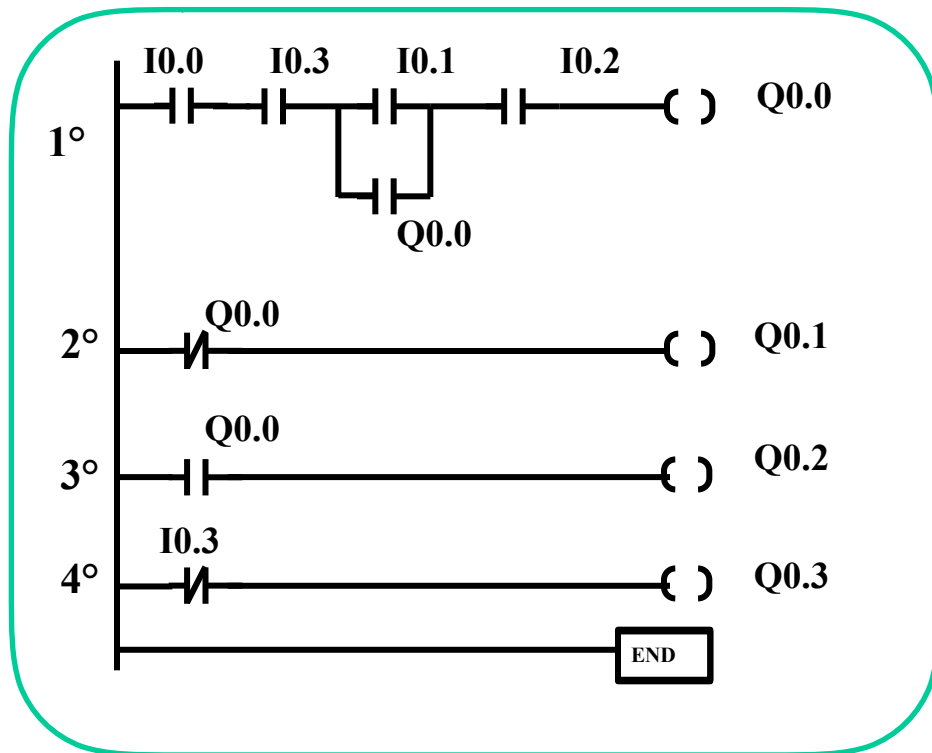
Se K1 non è eccitato allora accendi la lampada L1 di arresto

Se K1 è eccitato allora accendi la lampada L2 di marcia avanti

Se è intervenuta la termica accendi la lampada L3

Fine programma

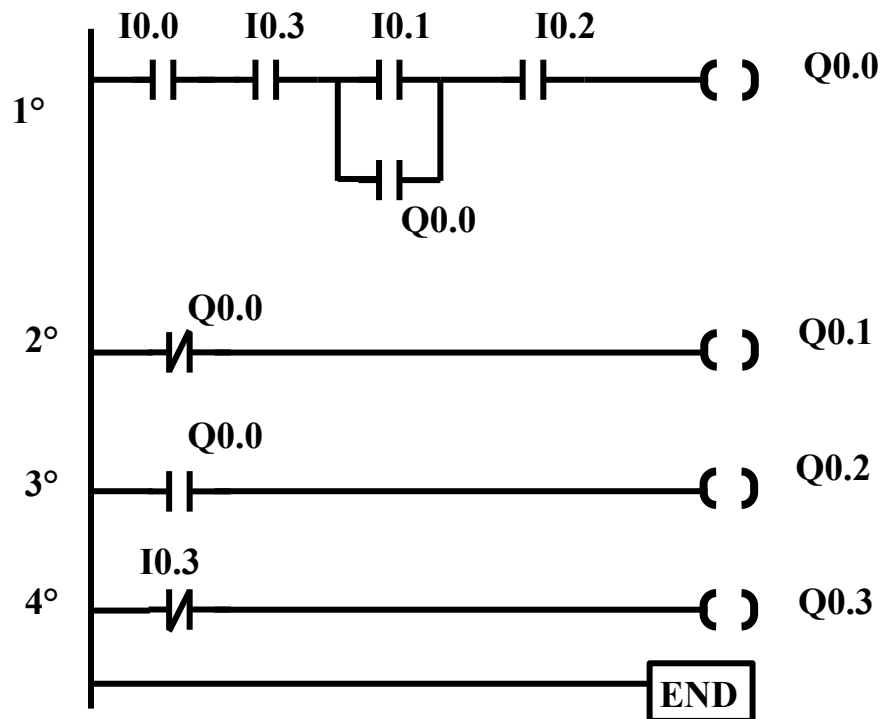
Per poter attivare un'uscita (es. **Q0.0** ) è necessario che il PLC legga, alla fine di un segmento e dopo aver effettuato tutte le operazioni logiche presenti, il livello logico **1**.



Facendo riferimento al 1° segmento del nostro schema KOP possiamo osservare che l'unico contatto che il PLC leggerà come **0** sarà quello di S2, (pulsante di marcia) ossia I0.1

# PER CAPIRE IL A KOP

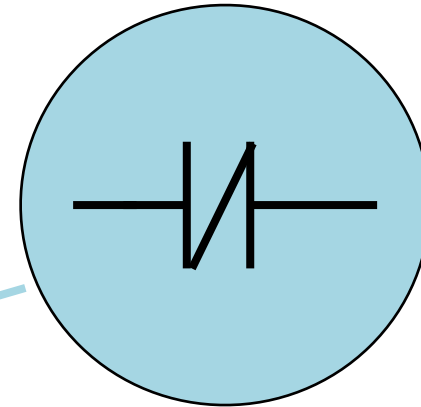
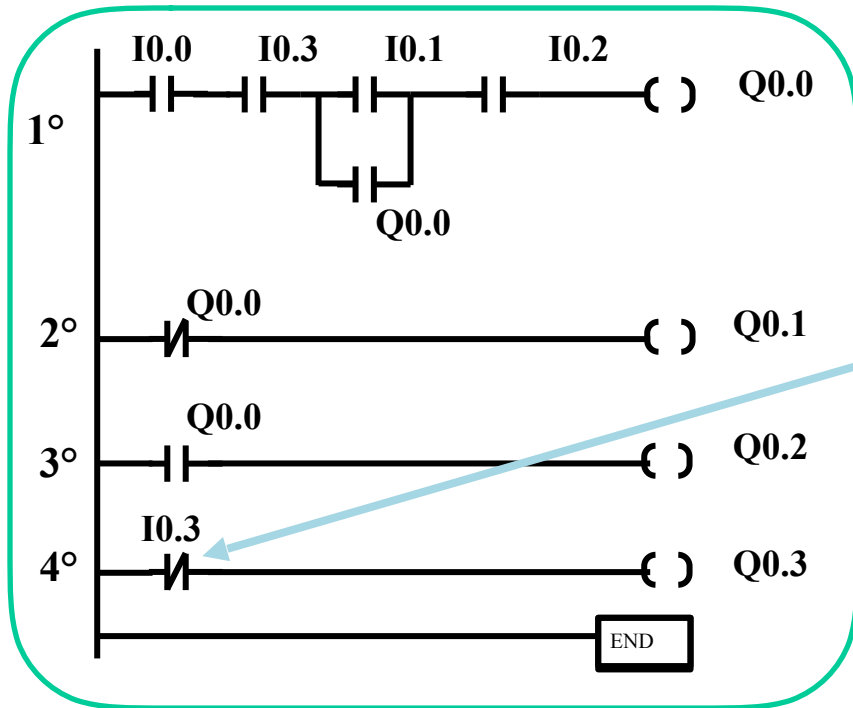
Analizziamo ora il significato dei simboli utilizzati



Per poter attivare l'uscita **Q0.0**, ossia poter eccitare il contattore K1, è necessario che l'ingresso **I0.1** si porti quindi a livello logico **1**. Ciò avviene semplicemente quando **premiamo** il pulsante S2 che chiudendosi passerà dal livello logico **0** a quello **1** e il PLC potrà attivare l'uscita Q0.0 dopo aver svolto le operazioni logiche AND e OR presenti.

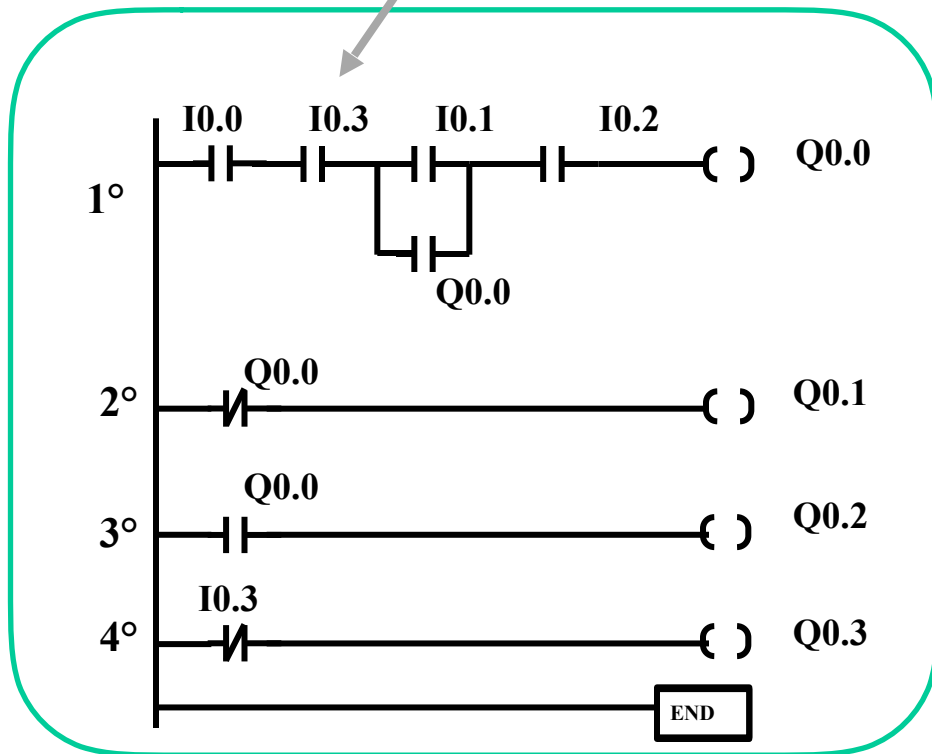


# PER CAPIRE IL LINGUAGGIO KOP



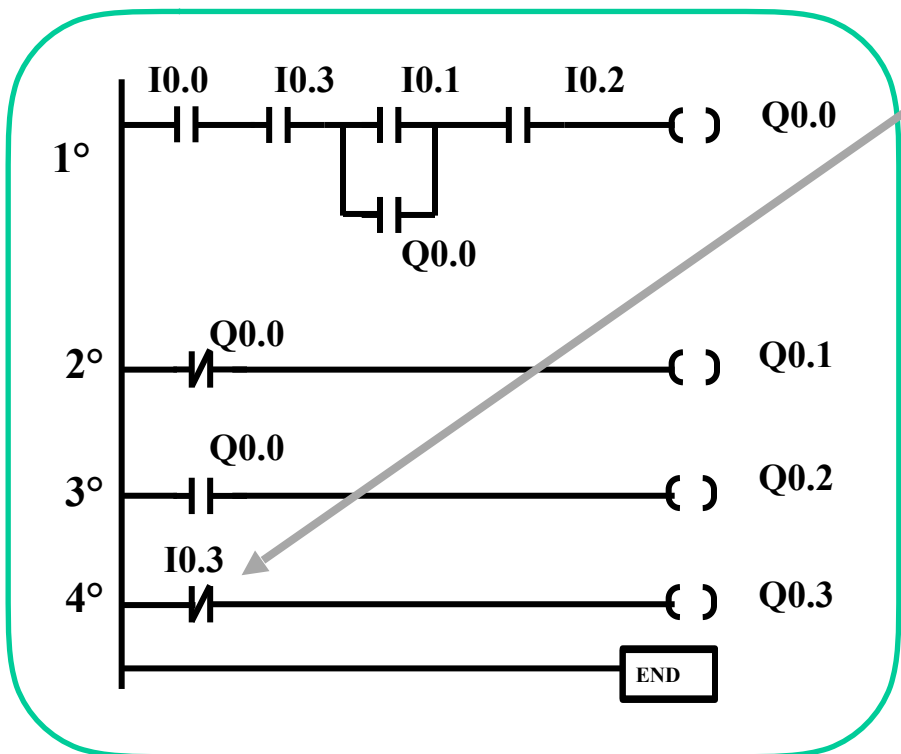
Questo simbolo rappresenta, come già detto, un **contatto negato**.

Facendo riferimento al nostro schema in KOP, possiamo notare, per esempio, che tale contatto ( F2 =**I0.3**) è quello che comanderà l'accensione della lampada L3 relativa all'intervento del relè termico.



Si noterà, inoltre, che l'altro contatto della termica, quello **N.C.** che è sempre contrassegnato con **I0.3** è disegnato, nel 1° segmento, senza la barra che indica la negazione.

Il contatto I0.3 **senza** la **negazione** ( $\neg$ ) sarà letto dal PLC con livello logico 1 essendo riferito al contatto della termica N.C. considerato nella tabella delle assegnazioni degli ingressi.

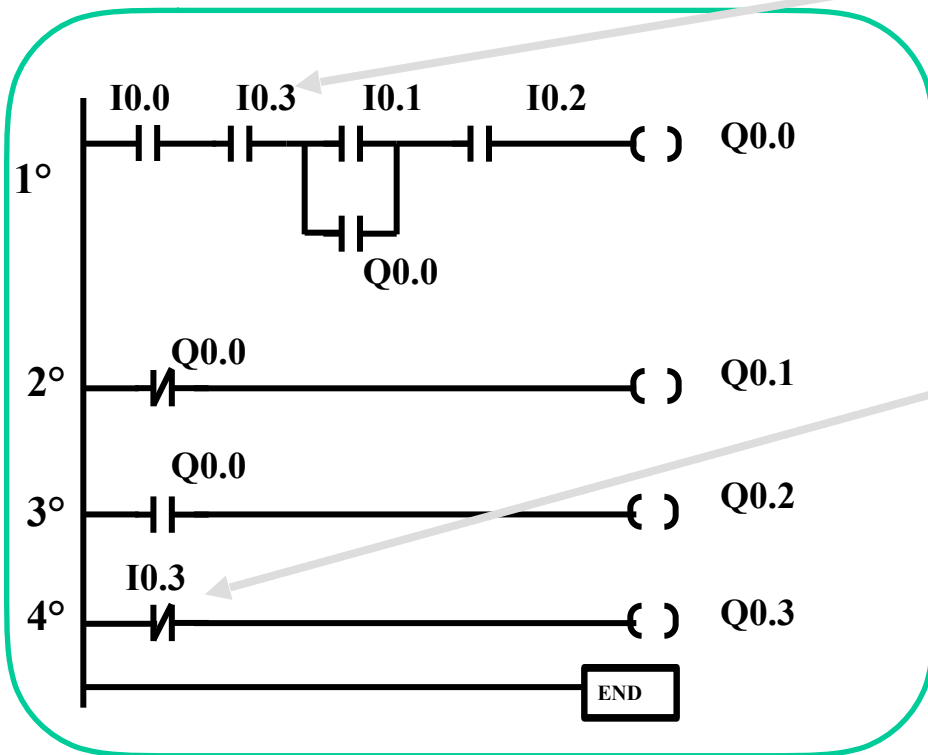


Lo stesso contatto **I0.3** con la negazione ( $\overline{I}$ ) sarà letto dal PLC in questo modo:

Se lo stato logico rilevato sul contatto **I0.3** (posto sul morsetto di ingresso) è **0** il PLC negherà il risultato e darà **1**

Se lo stato logico rilevato è **1** il PLC negherà il risultato e darà **0**

Dal nostro schema di comando si può notare che la lampada **L3** si accende solo quando il relè termico interviene aprendo il suo contatto **N.C.** e chiudendo il contatto **N.A.** collegato alla lampada L3.

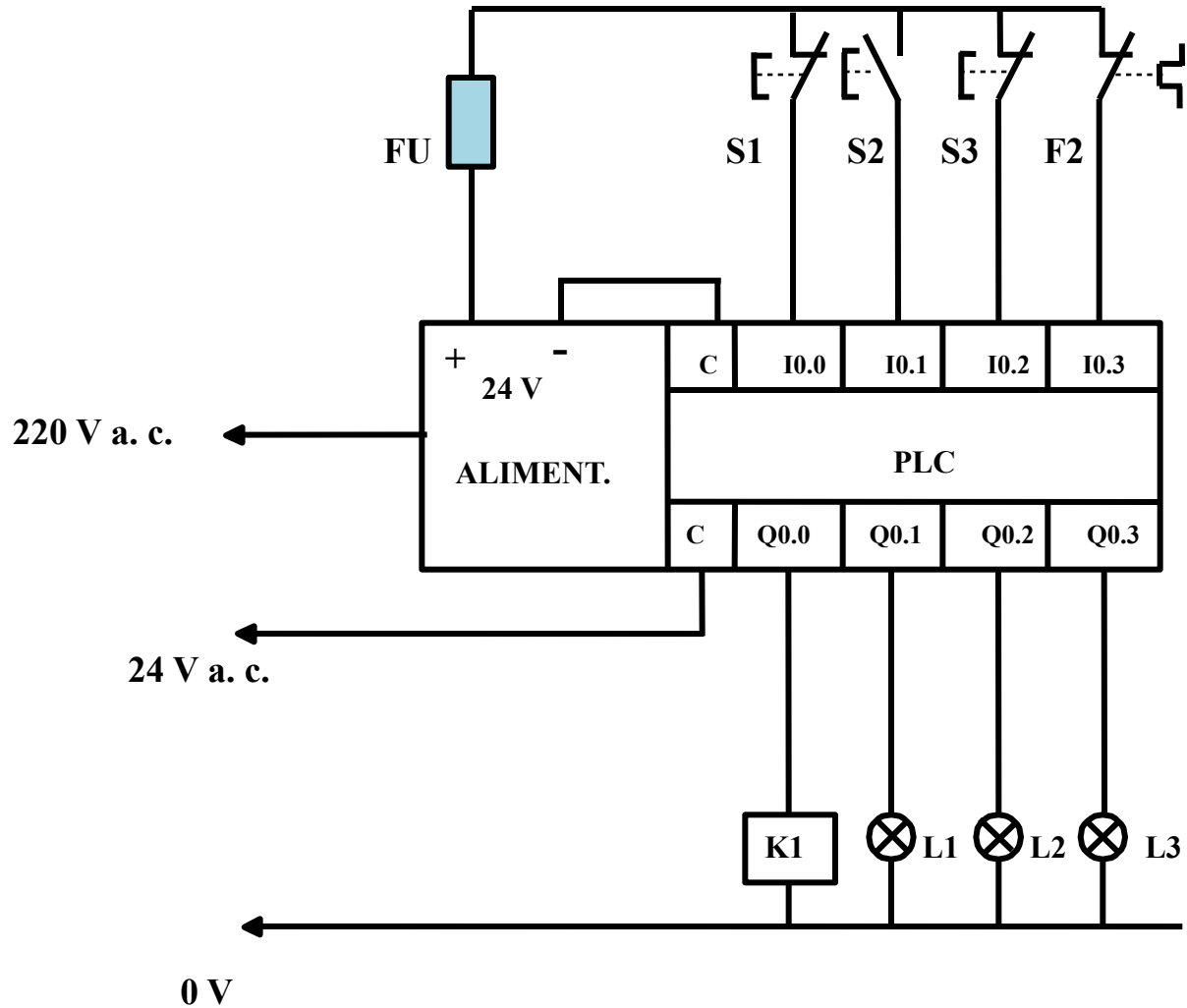


Nello schema KOP il contatto della termica (I0.3)  $\overline{I}$  del 1° segmento sarà, come detto, letto come livello logico 1.

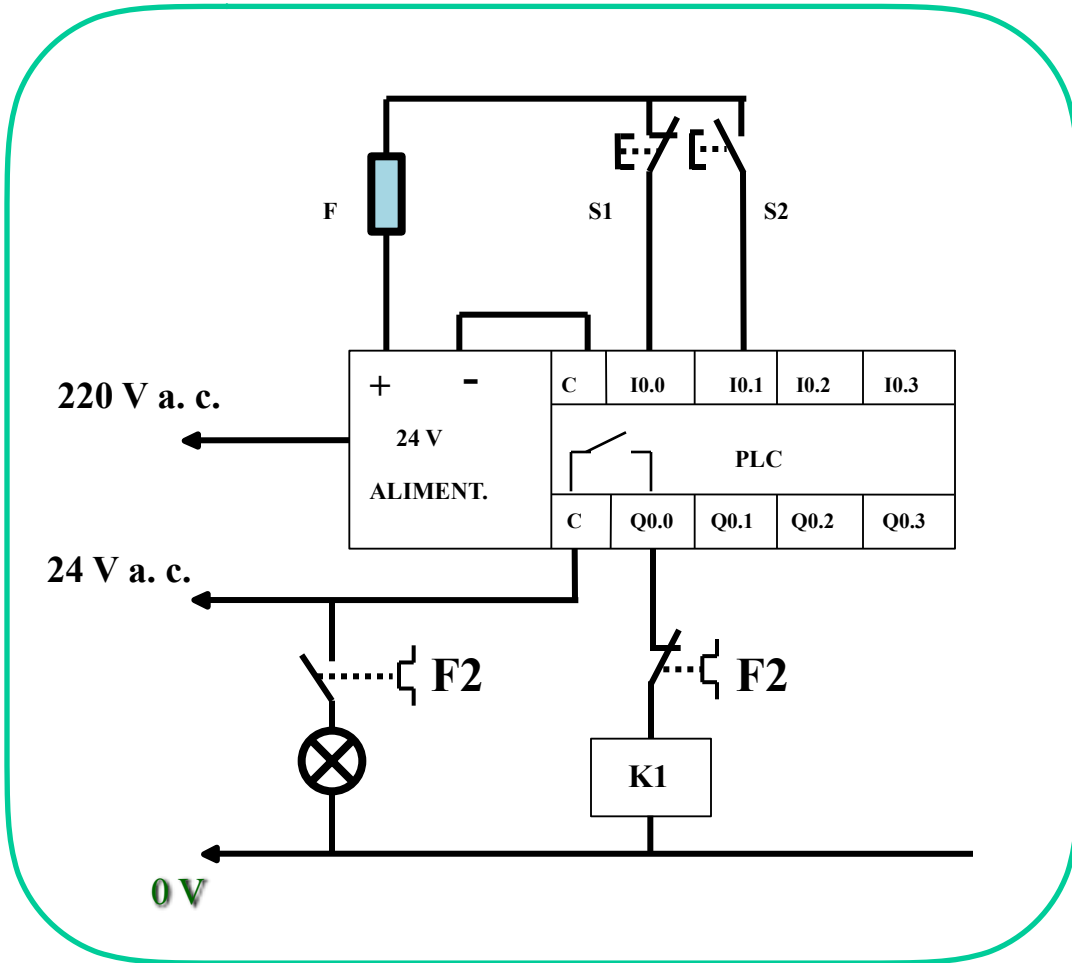
Lo stesso contatto (I0.3)  $\overline{I}$  posto nel 4° segmento sarà letto come livello logico 0 e la lampada L3 rimane spenta.

Quando la termica interverrà il contatto N.C. si aprirà e il PLC leggerà, su I0.3, il livello logico 0 e disecciterà K1 (Q0.0). Contemporaneamente un'altra istruzione data al PLC (ossia quella di negare lo stesso livello logico letto su I0.3, rappresentato col simbolo di contatto negato in KOP) darà livello logico 1 sull'uscita Q0.3 e ciò consentirà l'accensione della lampada L3.

# SCHEMA DI MONTAGGIO DEL CIRCUITO IN ESAME

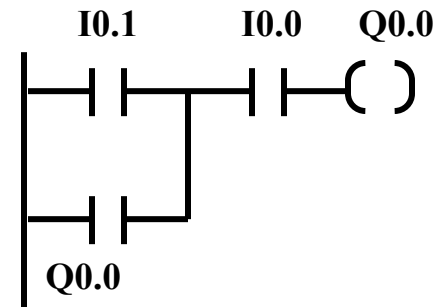


# ESEMPIO DI COLLEGAMENTO I/O – SCHEMA KOP E LISTA ISTRUZIONI AWL



S1	Pulsante di stop	I0.0
S2	Pulsante di marcia	I0.1
K1	Contattore	Q0.0

## Schema KOP



## AWL

```
LD I0.1
O I0.0
A I0.0
= Q0.0
```