

appunti del corso di



Tecnologie dei **S**istemi di **A**utomazione

Anno accademico 2009/2010

Corso per allievi di Ingegneria Informatica (allievi A-I) ed Elettronica

III anno

ing. Vincenzo Lippiello



I PARTE

Tecnologie dei Sistemi di Automazione

Sensori

Attuatori

Dispositivi per il controllo

informazioni generali

Docente



- **Ing. Vincenzo Lippiello**
 - <http://wpage.unina.it/lippiell>
 - vincenzo.lippiello@unina.it
 - Tel. 081 76 83635
- *Ricevimento*
 - Lunedì dalle 10.30 alle 12.30
 - Per appuntamento via e-mail
- *Studio*
 - Biennio a Via Claudio, 21 – Edificio 3/A – Stanza 3.12



informazioni generali

Testi e materiale di riferimento per il corso

- **I PARTE: Sensori ed Attuatori:**
 - *“Tecnologie dei sistemi di controllo”*, G. Magnani, G. Ferretti e P. Rocco, McGraw-Hill, II ed.
- **II PARTE: Controllori e programmazione**
 - *“Tecnologie informatiche per l’automazione”*, P. Chiacchio e F. Basile, McGraw-Hill, II ed.
 - **UniSim, GrafChart** – *Simulatori algoritmi di controllo in SFC*
- Slides del corso



introduzione

Automazione industriale

- **Automazione industriale:** disciplina che studia le metodologie e le tecnologie che permettono il controllo dei flussi di energia, di materiali e di informazioni necessarie alla realizzazione di processi produttivi, senza l'intervento dell'uomo
 - Termine “automazione” coniato dalla Ford Motor nel 1946
- Personalizzazione di massa:
 - Produzione artigianale → altamente personalizzabile
 - Produzione di massa → prodotti uniformi
 - Personalizzazione di massa → varietà di caratteristiche



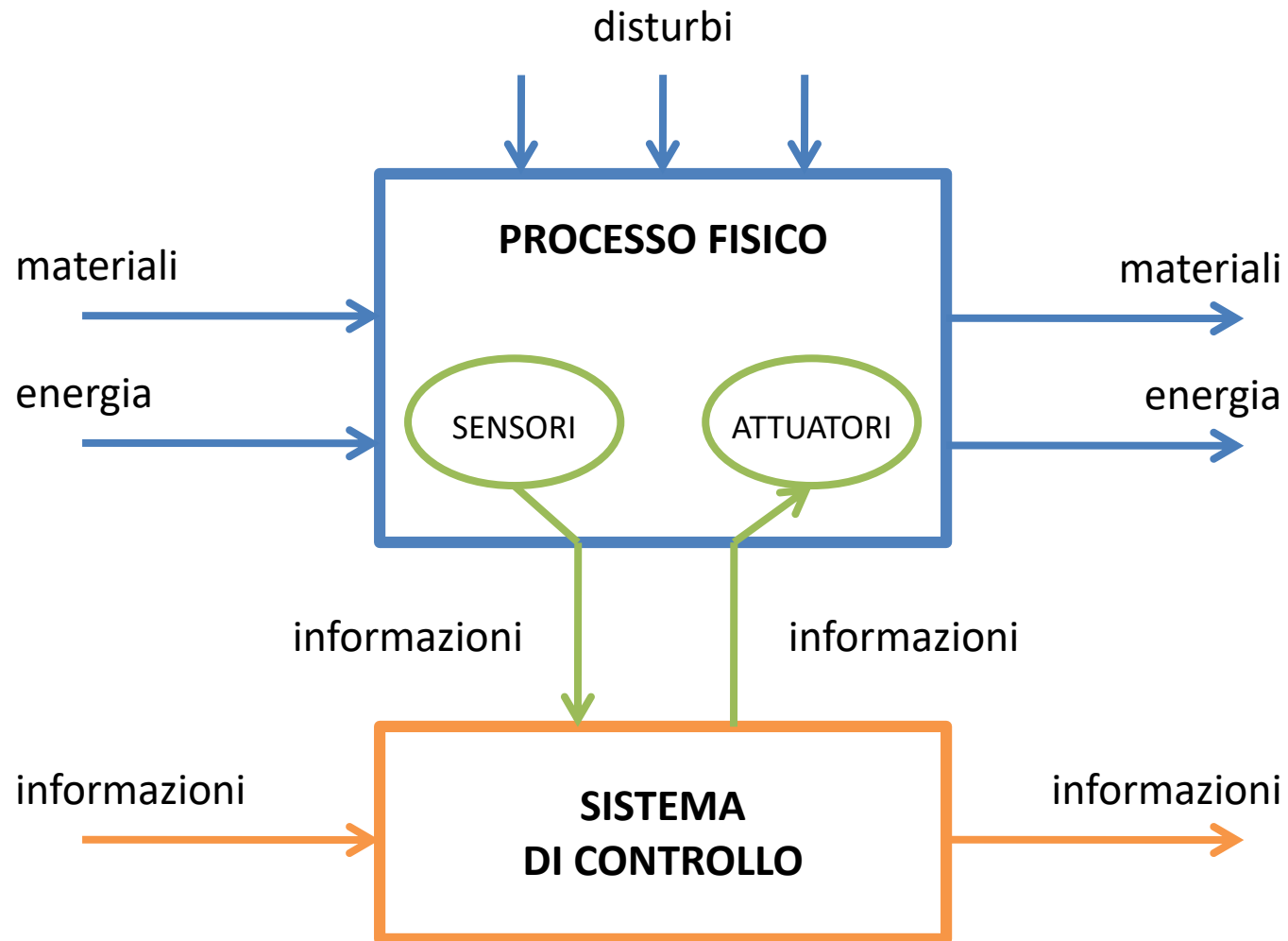
introduzione

Benefici dell'automazione

- ↑ qualità dei prodotti
- ↑ flessibilità dei produzione
- ↓ tempi di produzione
- ↓ magazzini in ingresso ed in uscita
- ↓ scarti di lavorazione
- ↓ costi produzione
- ↑ conformità a leggi e regolamenti
- ↓ impatto ambientale + ↑ risparmio energetico
- ↑ competitività

introduzione

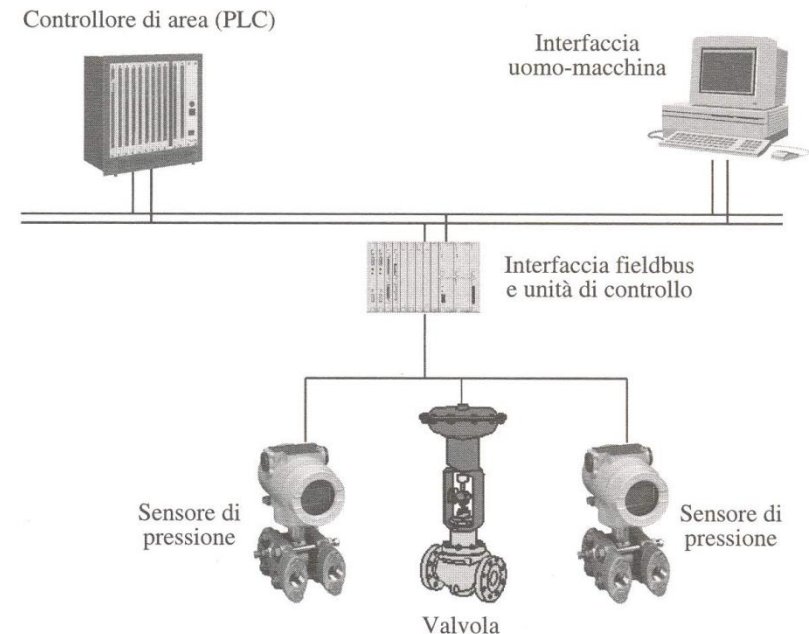
Sistema di controllo automatico



introduzione

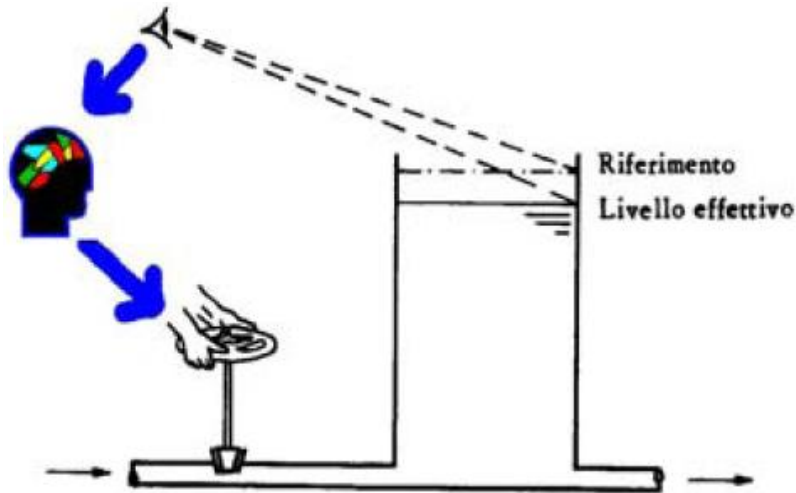
Elementi principali di un sistema di controllo

- Processo fisico
 - Sensori (elemento sensore + trasduttore)
 - Condizionamento dei segnali
 - Attuatori
- Sistema di controllo
- *Inoltre:*
 - Sistemi di comunicazione
 - Interfacce uomo-macchina



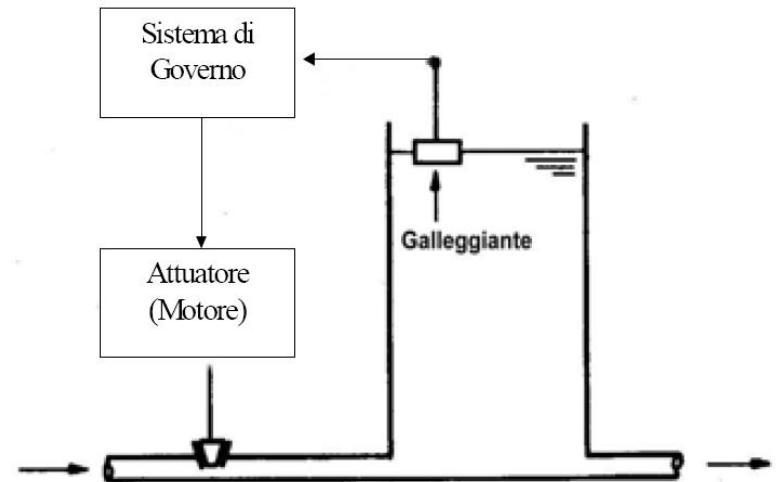
esempi di sistemi di controllo automatici

Regolazione del livello di un serbatoio



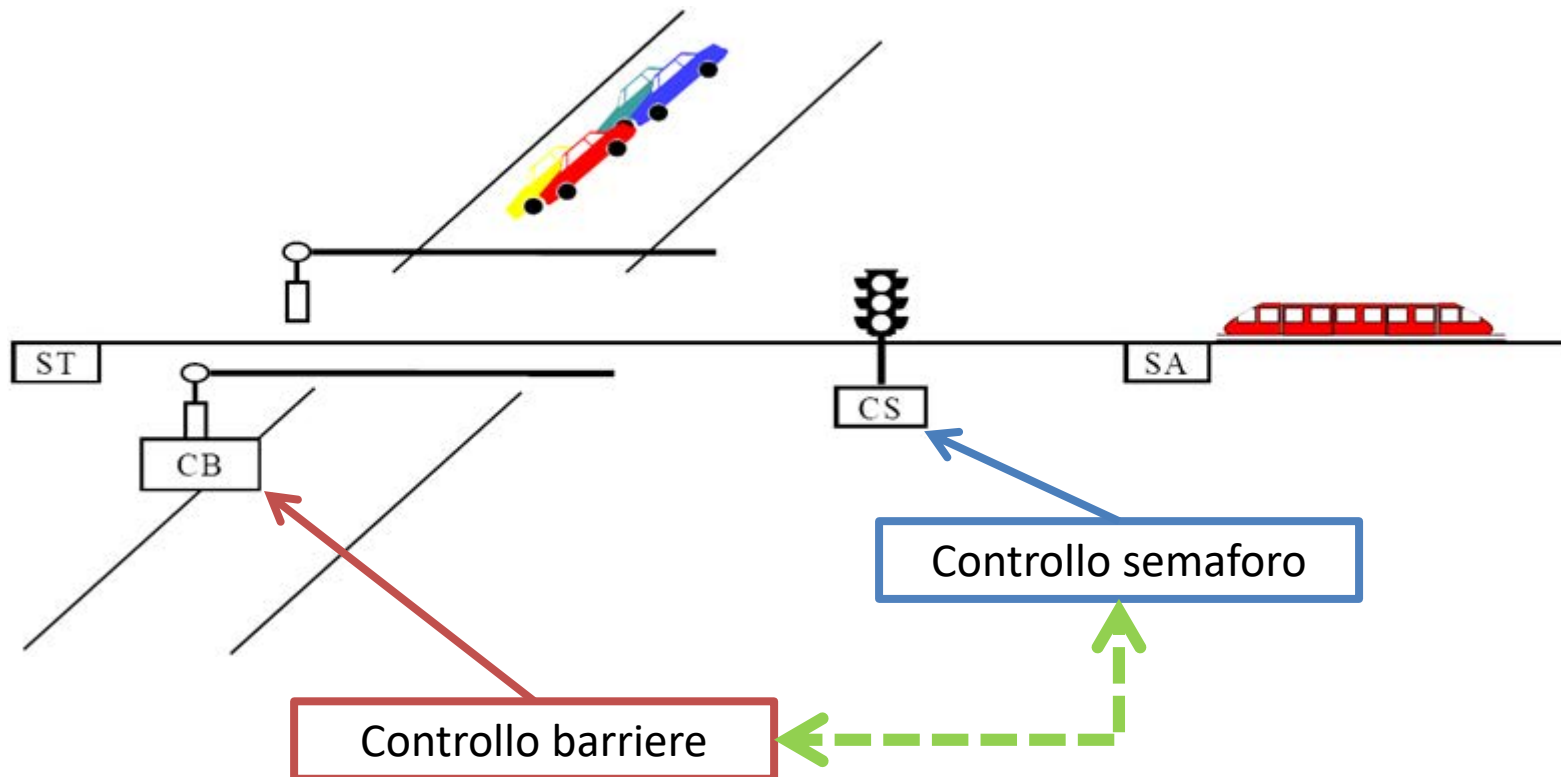
Regolazione manuale

Regolazione automatica



esempi di sistemi di controllo automatici

Controllo di un passaggio a livello ferroviario



esempi di sistemi di controllo automatici

Processo di purificazione del propilene

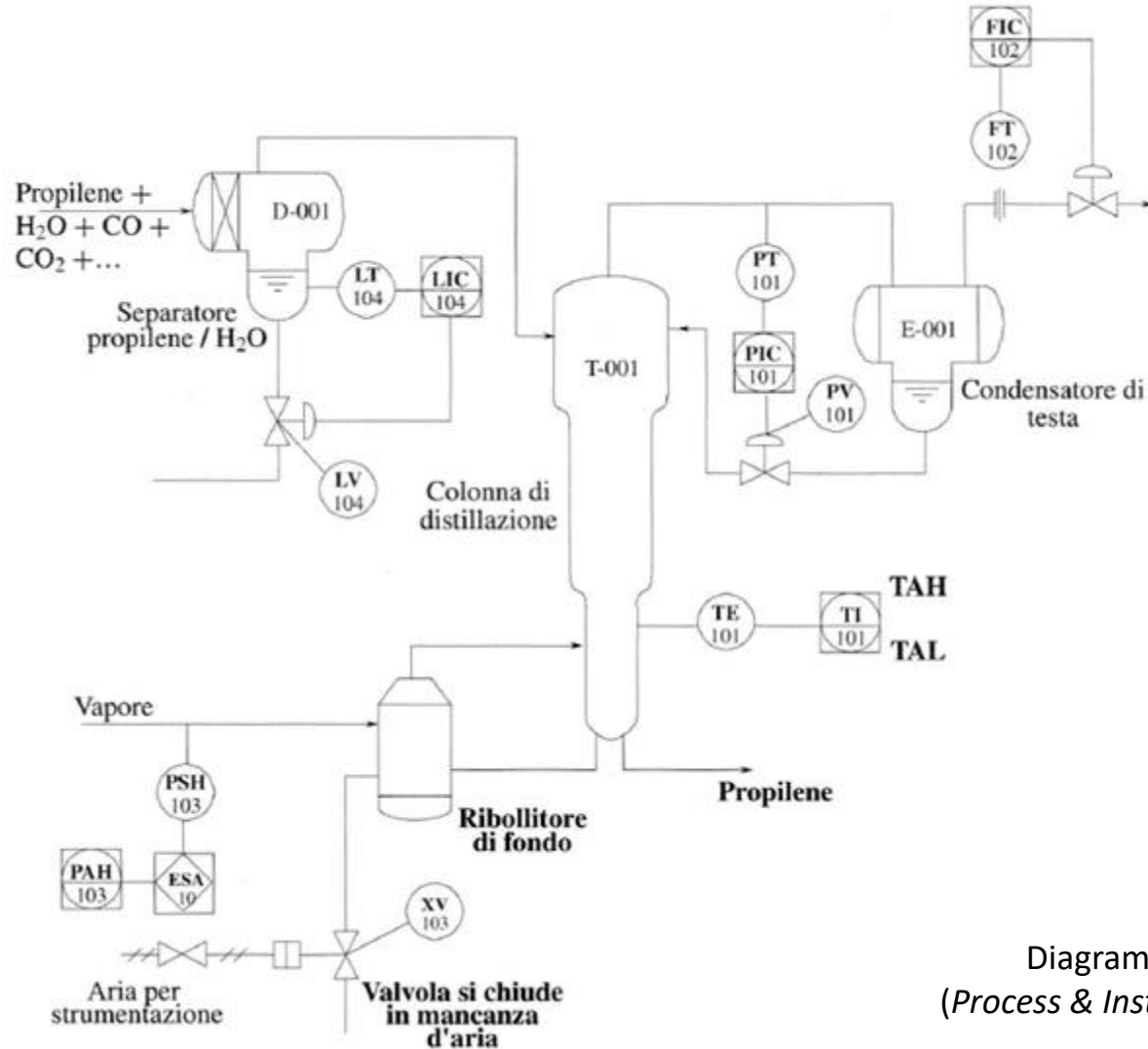


Diagramma P&I
(Process & Instrumentation)

elementi di modellistica

introduzione



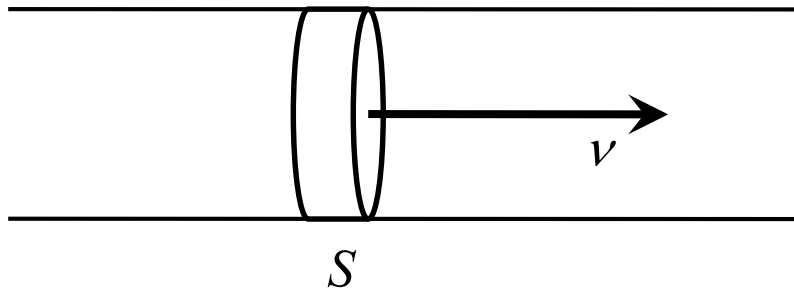
- Modelli matematici
 - Rappresentazione della realtà di interesse
 - Processo fisico da controllare
 - Caratteristiche dinamiche degli attuatori
- Elementi di modellistica:
 - Equazioni di conservazione dei processi a fluido
 - Elementi di meccanica
 - Conversione elettromeccanica

equazioni di conservazione per processi a fluido

Portata massica e volumetrica



- La **portata massica** di un fluido in una tubazione esprime la quantità di fluido, in unità di massa, che attraversa la sezione nell'unità di tempo [kg/s]



$$w_S = \rho A v$$

$$\rho = \rho(T, p)$$

- La **portata volumetrica** [m^3/s]

$$q_S = A v$$

equazioni di conservazione per processi a fluido

Moto turbolento e laminare



- La velocità di un fluido in una sezione di un condotto rettilineo e circolare è
 - se moto *turbolento* → uniforme
 - se moto *laminare* → massima al centro, decresce rapidamente verso la periferia del condotto
- Il tipo di moto dipende dalla velocità del fluido, dalla sua densità e viscosità e dal diametro della condotta
 - **Numero di Reynolds:** $Re = \frac{\rho v D}{\mu}$
 - $Re < 2000 \rightarrow$ laminare, $Re > 4000 \rightarrow$ turbolento

equazioni di conservazione per processi a fluido

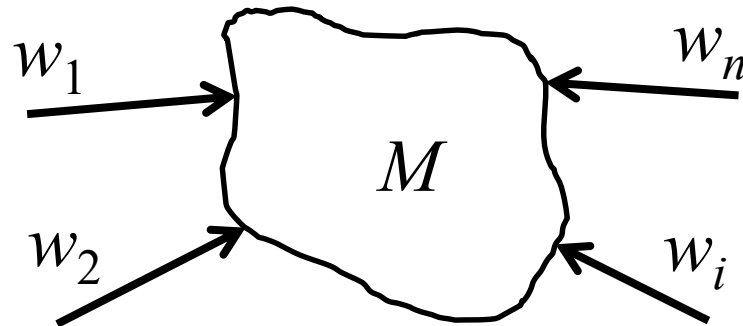
Introduzione



- Le portate di fluidi ed energia hanno spesso grande influenza sul comportamento dinamico dei processi in cui intervengono trasporto e trasformazione di materiali ed energia
- Risulta importante poter esprimere le portate di uscita in funzione delle variabili di ingresso del processo
 - *Esempio*: determinare la relazione tra la portata di ingresso e quella di uscita di un fluido in una tubazione

equazioni di conservazione per processi a fluido

Principio di conservazione della massa

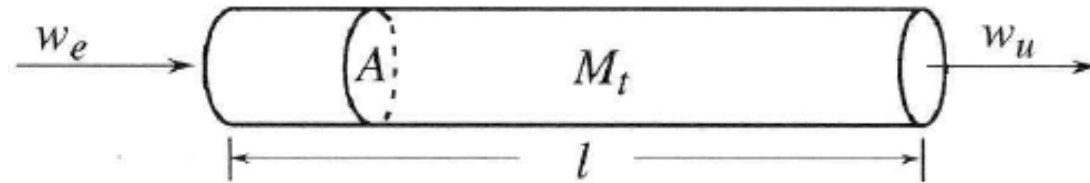


- Principio di conservazione della massa:

$$\frac{dM(t)}{dt} = \sum_{i=1}^n w_i(t)$$

equazioni di conservazione per processi a fluido

Principio di conservazione della massa – esempio: tubazione



Applicando il principio di conservazione della massa:

$$\frac{dM_t(t)}{dt} = w_e(t) - w_u(t)$$

da cui

$$M_t(t) = lA\rho(t) \Rightarrow \frac{dM_t(t)}{dt} = lA \frac{d\rho(t)}{dt}$$

Si noti che

$$\rho(t) = \text{const} \Rightarrow w_u(t) = w_e(t)$$

equazioni di conservazione per processi a fluido

Principio di conservazione della massa – esempio: serbatoio

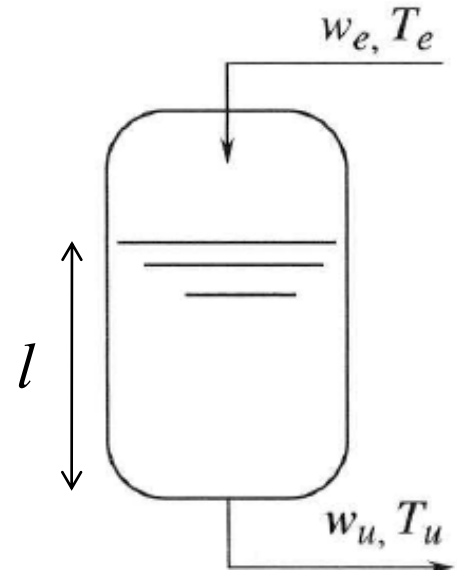


Applicando il principio di conservazione della massa:

$$\frac{dM_t(t)}{dt} = w_e(t) - w_u(t)$$

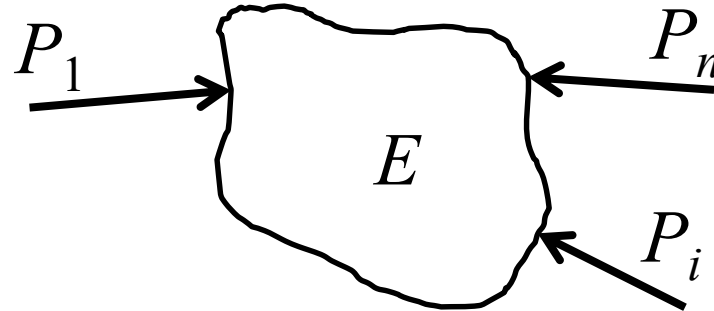
da cui

$$M_t = \rho A l \quad \Rightarrow \quad \frac{dl}{dt} = \frac{1}{\rho A} (w_e - w_u)$$



equazioni di conservazione per processi a fluido

Principio di conservazione dell'energia



- Principio di conservazione dell'energia:

$$\frac{dE(t)}{dt} = \sum_{i=1}^n P_i(t)$$

- **Energia:** termica e meccanica (cinetica e potenziale)
- **Potenza:** associate alle masse (portate) entranti e uscenti, termica entrante e uscente, lavoro meccanico compiuto da o sul volume

equazioni di conservazione per processi a fluido

Principio di conservazione dell'energia



- Principio di conservazione dell'energia esplicitato:

$$\begin{aligned}\frac{dE}{dt} &= \sum_{j=1}^{n-2} w_j \left(\left(e_j + \frac{p_j}{\rho_j} \right) + \frac{v_j^2}{2} + gz_j \right) + Q - L \\ &= \sum_{j=1}^{n-2} w_j \left(h_j + \frac{v_j^2}{2} + gz_j \right) + Q - L\end{aligned}$$

- Lavoro, nell'unità di tempo, del fluido per entrare

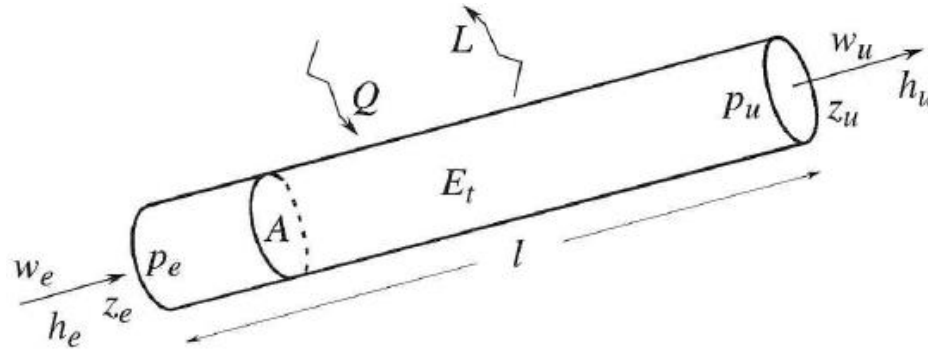
$$w_j \left(p_j / \rho_j \right) = p_j A_j v_j$$

- Entalpia del fluido

$$h_j = e_j + p_j / \rho_j$$

equazioni di conservazione per processi a fluido

Principio di conservazione dell'energia – esempio: tubazione



Principio di conservazione dell'energia:

$$\frac{dE_t}{dt} = w_e \left(h_e + \frac{v_e^2}{2} + gz_e \right) - w_u \left(h_u + \frac{v_u^2}{2} + gz_u \right) + Q - L$$

Energia totale:

$$E_t = M_t \left(\bar{e} + \frac{\bar{v}^2}{2} + g\bar{z} \right) = lA\bar{\rho} \left(\bar{e} + \frac{\bar{v}^2}{2} + g\bar{z} \right)$$

... se tubazione corta, altrimenti studio agli elementi finiti o distribuiti

equazioni di conservazione per processi a fluido

Equazioni di Bernoulli



- **L'equazione di Bernoulli** deriva dalla particolarizzazione del principio di conservazione dell'energia per una tubazione nelle seguenti ipotesi:
 - Condizioni stazionarie ($dE/dt = 0$)
 - Pareti della tubazione adiabatiche e rigide ($Q = L = 0$)
 - Energia termica trasportata dal fluido trascurabile rispetto a quella meccanica
 - Densità del fluido costante (*fluido incompressibile*)
 - Assenza di attriti (interni al fluido e tra fluido e pareti)
- ⇒ Potenza entrante = Potenza uscente (in ogni istante!)
- con riferimento ad un elemento infinitesimo: $dE_A = dE_B$

equazioni di conservazione per processi a fluido

Equazioni di Bernoulli – Carico idrostatico



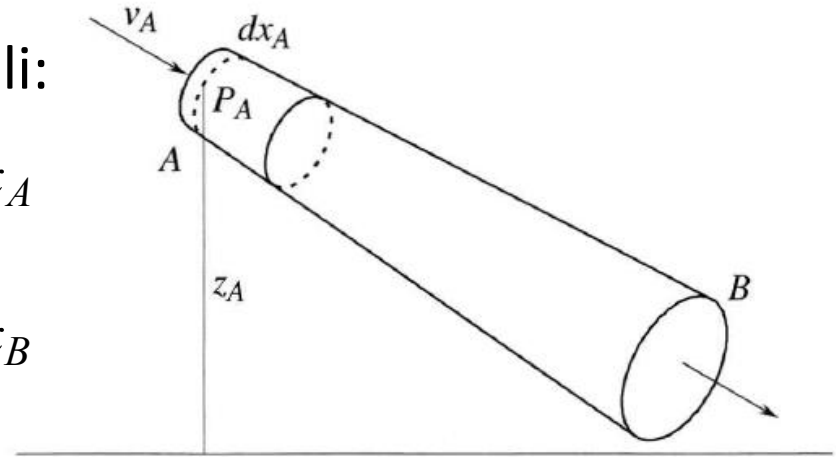
Ovvero sono istantaneamente uguali:

$$dE_A = A_A dx_A \rho \frac{v_A^2}{2} + A_A dx_A \rho z_A g + p_A A_A dx_A$$

$$dE_B = A_B dx_B \rho \frac{v_B^2}{2} + A_B dx_B \rho z_B g + p_B A_B dx_B$$

Per la conservazione della massa

$$A_A dx_A = A_B dx_B \quad \Rightarrow \quad \frac{v_A^2}{2g} + z_A + \frac{p_A}{\rho g} = \frac{v_B^2}{2g} + z_B + \frac{p_B}{\rho g}$$



In generale si definisce **carico idrostatico** del fluido la quantità *costante* lungo la condotta nelle ipotesi citate

$$\frac{v^2}{2g} + z + \frac{p}{\rho g} = \text{const}$$

equazioni di conservazione per processi a fluido

Equazioni di Bernoulli – Perdite di carico



- Nel caso di fluidi reali, a causa degli attriti il carico idrostatico si ridurrà lungo la condotta
 - Energia meccanica → Energia termica → Riscaldamento del fluido
- Le **perdite di carico** (interne al fluido e tra fluido e pareti) sono funzione del quadrato della velocità

$$\frac{v_A^2}{2g} + z_A + \frac{p_A}{\rho g} = \frac{v_B^2}{2g} + z_B + \frac{p_B}{\rho g} + \alpha \frac{v_B^2}{2g}$$

- Le perdite di carico si dividono in
 - perdite di carico **distribuite** lungo i tratti rettilinei
 - perdite di carico **concentrate** nei raccordi

equazioni di conservazione per processi a fluido

Equazioni di Bernoulli – Perdite di carico distribuite



- Le **perdite di carico distribuite** in una tubazione rettilinea isoterma di lunghezza l e diametro D

$$F = \left(\frac{4fl}{D} \right) \frac{v^2}{2g_c} = \frac{32flw^2}{\pi^2 \rho^2 g_c D^5} \quad [\text{J/kg}]$$

espressa in termini di energia specifica, dove f è il *fattore di attrito di Fanning* che dipende da Re e dalla rugosità delle pareti della condotta e g_c è un coefficiente dimensionale

– Il fattore di Fanning può essere ricavato dai *diagrammi di Moody*, oppure con l'*equazione di Colebrook*

- Dimensionamento delle tubazioni: **velocità** nominale tra 1 e 3 m/s per avere moto turbolento, limitata per ridurre le perdite

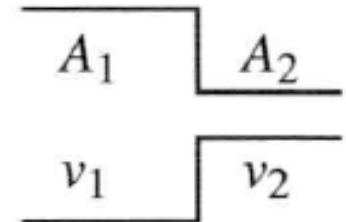
equazioni di conservazione per processi a fluido

Equazioni di Bernoulli – Perdite di carico concentrate



- Le **perdite di carico concentrate** dipendono dal tipo di raccordo in esame (→datasheet)
 - Brusche contrazioni di sezione:

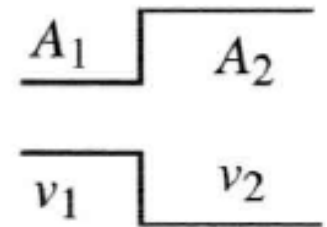
$$F = K_C \frac{v_2^2}{2g_c}$$



dove se il moto è turbolento K_C è funzione di A_2/A_1 e Re

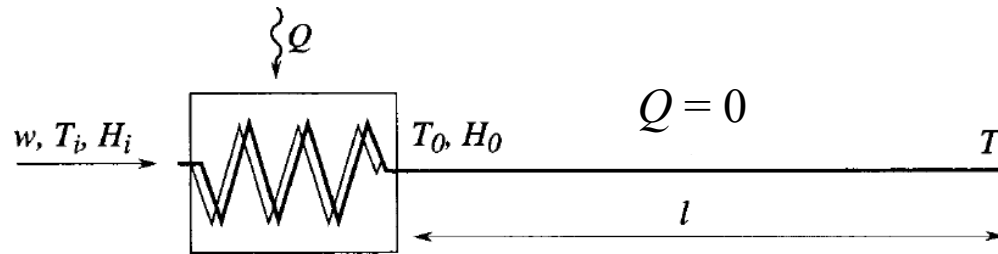
- Bruschi allargamenti di sezione (in moto turbolento):

$$F = \frac{(v_1 - v_2)^2}{2g_c} = \frac{v_1^2}{2g_c} \left(1 - \frac{A_1}{A_2} \right)^2$$



equazioni di conservazione per processi a fluido

Processi termici – esempio: scambiatore di calore a flusso termico imposto



- Scambiatore a flusso termico imposto seguito da una condotta adiabatica
 - Variazione di energia meccanica trascurabile rispetto a quella termica
- Si vuole valutare la funzione di trasferimento tra T_1 e w
- Dal principio di conservazione dell'energia per lo scambiatore:

$$\frac{dE}{dt} = P_e - P_u = w(h_i - h_0) + Q$$

- Per i liquidi monofase incomprimibili: $e \approx h \approx cT$
- In condizioni stazionarie: $\bar{T}_0 = \bar{T}_i + \bar{Q}/(c\bar{w})$ ($c = \text{calore specifico}$)
→ dimensionamento dello scambiatore (Q), noti T_i , w e assegnata T_0

equazioni di conservazione per processi a fluido



Processi termici – esempio: scambiatore di calore a flusso termico imposto

- Ipotesi di zona di scambio “corta” → temperatura fluido nella zona di scambio uniforme e pari a T_0 (ipotesi di *buona miscelazione*) → $E \cong mcT_0$

$$mc \frac{dT_0}{dt} = cw(T_i - T_0) + Q$$

- Linearizzazione del modello intorno ad una condizione di regime:

$$w = \bar{w} + \Delta w, \quad T_i = \bar{T}_i + \Delta T_i, \quad T_0 = \bar{T}_0 + \Delta T_0, \quad Q = \bar{Q} + \Delta Q$$

- Sostituendo si ottiene

$$\frac{m}{\bar{w}} \frac{d\Delta T_0}{dt} + \Delta T_0 = \Delta T_i + \frac{\bar{T}_i - \bar{T}_0}{\bar{w}} \Delta w + \frac{\Delta Q}{c\bar{w}}$$

avendo trascurato i prodotti di variazione e ricordando che

$$c\bar{w}(\bar{T}_i - \bar{T}_0) + \bar{Q} = 0$$

equazioni di conservazione per processi a fluido

Processi termici – esempio: scambiatore di calore a flusso termico imposto

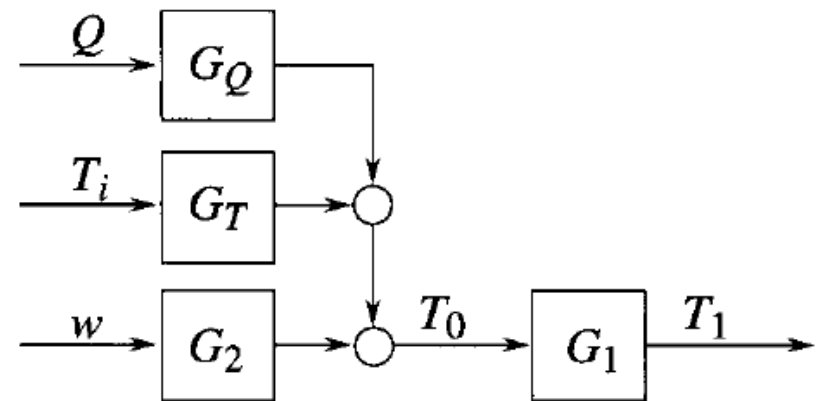


- Trasformata di Laplace:

$$G_2(s) = \frac{\Delta T_0(s)}{\Delta w(s)} = \frac{\bar{T}_i - \bar{T}_0}{\bar{w}} \frac{1}{1 + s \frac{m}{\bar{w}}}$$

$$G_T(s) = \frac{\Delta T_0(s)}{\Delta T_i(s)} = \frac{1}{1 + s \frac{m}{\bar{w}}}$$

$$G_Q(s) = \frac{\Delta T_0(s)}{\Delta Q(s)} = \frac{1}{c\bar{w}} \frac{1}{1 + s \frac{m}{\bar{w}}}$$



- In ipotesi di tubazione adiabatica, una variazione di temperatura in ingresso si manifesta in uscita dopo un tempo $\tau = l/\bar{v} = M/\bar{w}$, da cui

$$G_1(s) = \frac{\Delta T_1(s)}{\Delta T_0(s)} = e^{-s\tau}$$

e infine

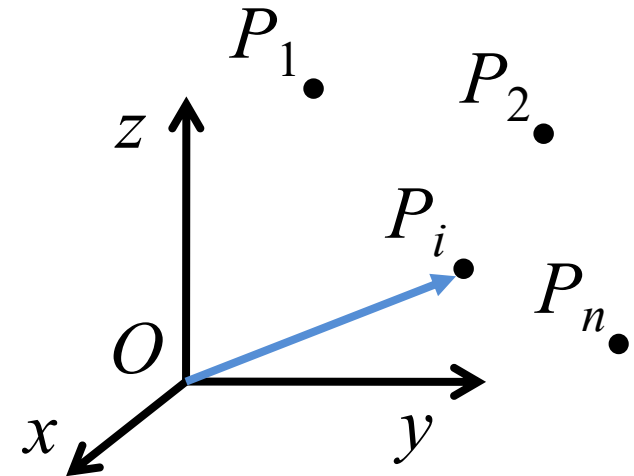
$$\frac{\Delta T_1(s)}{\Delta w(s)} = \frac{\bar{T}_i - \bar{T}_0}{\bar{w}} \frac{1}{1 + s \frac{m}{\bar{w}}} e^{-s\tau}$$

elementi di meccanica

Quantità meccaniche

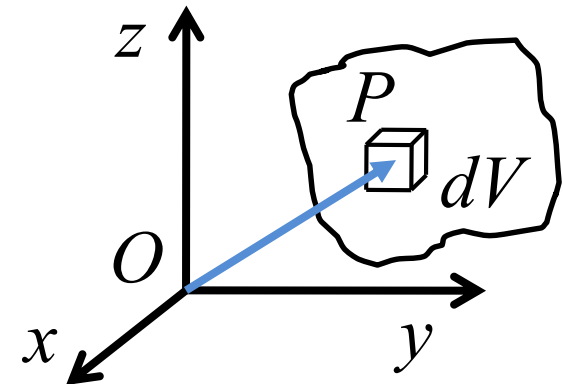
- **Centro di massa (o baricentro)**
di un *sistema di punti materiali*

$$\mathbf{p}_c = \frac{\sum_{i=1}^n \mathbf{p}_i m_i}{m} \quad \text{con} \quad m = \sum_{i=1}^n m_i$$



- **Baricentro di un corpo rigido**

$$\mathbf{p}_c = \frac{\int_V \mathbf{p} \rho dV}{m}$$



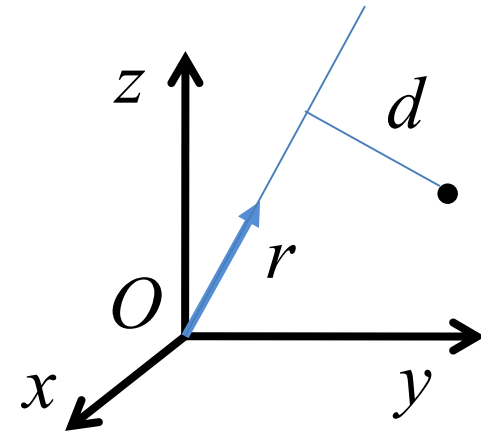
elementi di meccanica

Quantità meccaniche

- **Momento di inerzia** di un *punto materiale* rispetto ad un asse

$$I_r = md^2$$

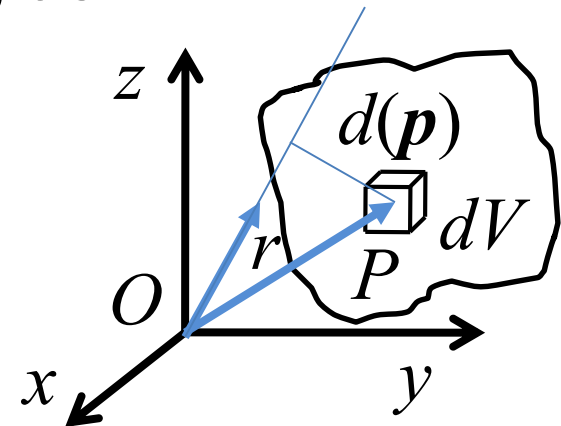
(sommatoria nel caso di sistemi di punti)



- **Momento di inerzia** di un *corpo rigido*

$$I_r = \int_V d^2(\mathbf{p}) \rho dV$$

- Momento rispetto ad un asse parallelo all'asse passante per il baricentro: $I_A = I_C + m\Delta^2$



elementi di meccanica

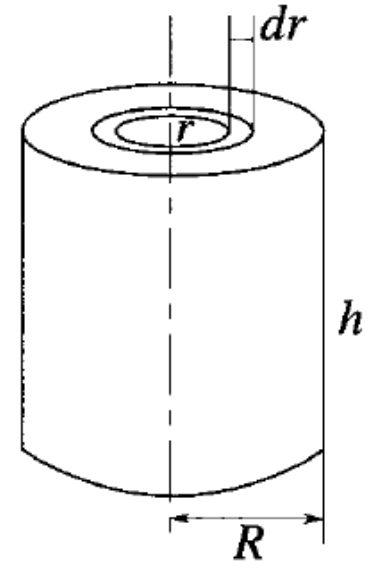
Quantità meccaniche – esempio: cilindro circolare omogeneo



- Momento di inerzia di un cilindro rispetto al suo asse baricentrale

$$I_C = \int_V \rho r^2 dV \overset{dV = d(\pi h r^2)}{\downarrow} = \int_0^R \rho r^2 2\pi r h dr = 2\pi \rho h \frac{R^4}{4} = m \frac{R^2}{2}$$

essendo $m = \pi \rho R^2 h$





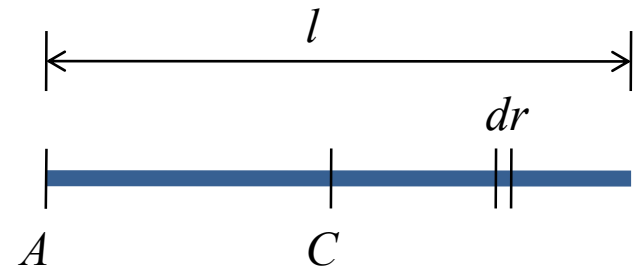
elementi di meccanica

Quantità meccaniche – esempio: asta omogeneo

- Asta è un corpo rigido in cui una direzione è preponderante rispetto all'altra
- Momento di inerzia rispetto all'asse ortogonale e passante per il centro

$$I_C = 2 \int_0^{l/2} \rho r^2 dr = 2\rho \left[\frac{r^3}{3} \right]_0^{l/2} = \frac{1}{12} ml^2$$

essendo ρ la densità lineare



- Momento rispetto al punto estremo A

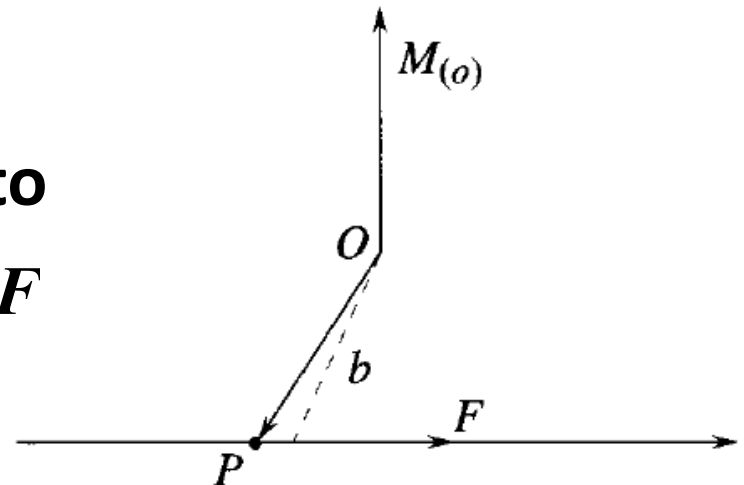
$$I_A = I_C + m \left(\frac{l}{2} \right)^2 = \frac{1}{3} ml^2$$

elementi di meccanica

Forze e momenti



- Sistema di forze agenti su un corpo rigido
 - Punto di applicazione delle forze può variare arbitrariamente lungo la retta di applicazione
- **Sistema equipollente**
 - Solo due forze, una delle quali applicata in un punto prefissato
 - Risultante + Coppia
- **Momento della forza rispetto ad un polo** $M_{(O)} = (P - O) \times F$



elementi di meccanica

Forze e momenti



- Cambio di polo

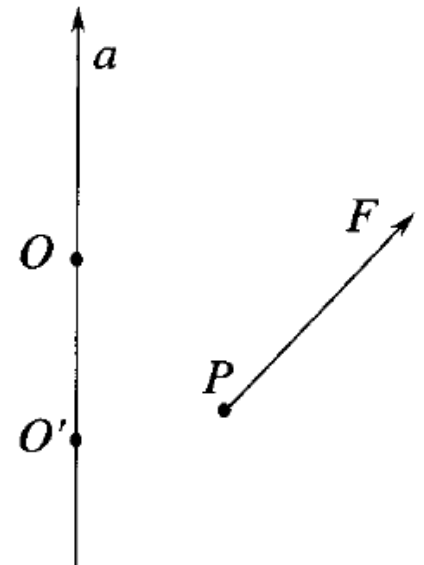
$$\mathbf{M}_{(O')} = \mathbf{M}_{(O)} + (\mathbf{O} - \mathbf{O}') \times \mathbf{F}$$

- **Momento della forza rispetto ad un asse**

$$\mathbf{M}_{(O')} \cdot \mathbf{a} = \mathbf{M}_{(O)} \cdot \mathbf{a} + (\mathbf{O} - \mathbf{O}') \times \mathbf{F} \cdot \mathbf{a} = \mathbf{M}_{(O)} \cdot \mathbf{a}$$

$\Rightarrow$ componente del momento della forza lungo l'asse è indipendente dal polo

$$\mathbf{M}_a = \mathbf{M}_{(O)} \cdot \mathbf{a}$$



elementi di meccanica

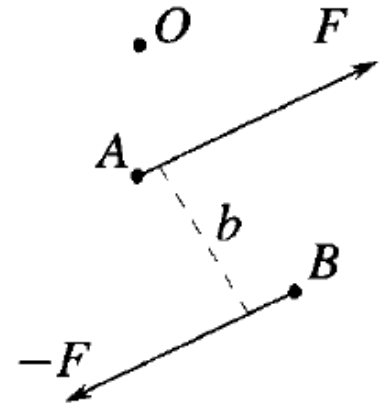
Forze e momenti



- Il momento risultante di un sistema di forze rispetto ad un polo è pari al momento del Risultante
- Il momento di una **coppia di forze** (Risultante nullo) è indipendente dal polo

$$\mathbf{M}_{(O)} = (\mathbf{A} - \mathbf{O}) \times \mathbf{F} + (\mathbf{B} - \mathbf{O}) \times -\mathbf{F} = (\mathbf{A} - \mathbf{B}) \times \mathbf{F}$$

- Massima riduzione di un sistema di forze
 - Risultante applicato in un punto prestabilito
 - Vettore momento rispetto al punto di applicazione del Risultante (ovvero il momento di una coppia equivalente)



elementi di meccanica

Equazioni di moto



- Vettore **quantità di moto**

$$\mathbf{Q} = \int_V \dot{\mathbf{p}} \rho dV = m \dot{\mathbf{p}}_C$$

dove $\dot{\mathbf{p}}_C$ è la velocità del centro di massa del corpo

- Vettore **momento della quantità di moto** (rispetto polo O)

$$\mathbf{\Gamma} = \int_V (\mathbf{P} - \mathbf{O}) \times \dot{\mathbf{p}} \rho dV$$

che nel caso di moto rotatorio intorno asse fisso:

$$\mathbf{\Gamma} = I \dot{\boldsymbol{\omega}}$$

con I momento di inerzia del corpo rispetto all'asse di rotazione

elementi di meccanica

Equazioni di moto



- **Equazioni cardinali del moto**

$$\frac{d\mathbf{Q}}{dt} = \mathbf{R} = m\ddot{\mathbf{p}}_C$$

$$\frac{d\mathbf{\Gamma}}{dt} = \mathbf{M}$$

- Ipotesi:
 - Corpo rigido libero
 - Momento espresso rispetto ad un polo fisso o al centro di massa

elementi di meccanica

Equazioni di moto



- **Moto vincolato:** a destra delle equazioni vanno aggiunte le relazioni vincolari
- **Moto rotatorio intorno ad un asse fisso**
 - Reazioni vincolari N_i nei punti di contatto O_i incognite
→ la seconda equazione cardinale cambia come segue:

$$\frac{d\mathbf{\Gamma}}{dt} = \mathbf{M} + (\mathbf{O}_1 - \mathbf{O}) \times \mathbf{N}_1 + (\mathbf{O}_2 - \mathbf{O}) \times \mathbf{N}_2 + \dots$$

- Nel caso in cui il polo è sull'asse di rotazione $\mathbf{a}$:

$$\frac{d\mathbf{\Gamma}}{dt} \cdot \mathbf{a} = \frac{d}{dt} (\mathbf{\Gamma} \cdot \mathbf{a}) = \mathbf{M} \cdot \mathbf{a}$$

elementi di meccanica

Equazioni di moto



- $\mathbf{\Gamma} \cdot \mathbf{a}$ è il momento della quantità di moto $\mathbf{\Gamma}_a$ rispetto all'asse (indipendente dal polo)
- $\mathbf{M} \cdot \mathbf{a}$ è il momento $\mathbf{M}_a$ del sistema di forze attive rispetto all'asse (indipendente dal polo)

$$\frac{d\mathbf{\Gamma}_a}{dt} = \mathbf{M}_a$$

*Non contiene reazioni
vincolari incognite!*

- Detto φ l'angolo di rotazione, ricordando che $\mathbf{\Gamma}_a = \mathbf{J}_a \dot{\varphi}$ ($\mathbf{J}_a$ è il momento di inerzia del corpo rispetto all'asse), si ottiene

$$\mathbf{J}_a \ddot{\varphi} = \mathbf{M}_a = \boldsymbol{\tau}$$

Coppia $\boldsymbol{\tau}$: momento rispetto all'asse di rotazione del sistema di forze attive agenti sul corpo



conversione elettromeccanica

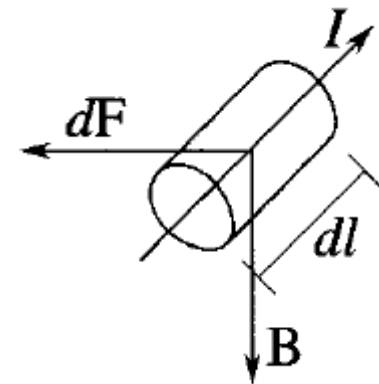
Generazione di coppia motrice

- **Legge di Lorentz:**

$$\mathbf{F} = q\mathbf{v} \times \mathbf{B}$$

- Forza agente su un elemento infinitesimo di un conduttore

$$d\mathbf{F} = I d\mathbf{l} \times \mathbf{B}$$



- Forza totale su un conduttore di lunghezza l

$$\mathbf{F} = \int_I I d\mathbf{l} \times \mathbf{B} \quad \xrightarrow[\text{Conduttore rettilineo}]{\mathbf{B} \text{ e } I \text{ uniformi}} \quad \mathbf{F} = I\mathbf{l} \times \mathbf{B}$$

conversione elettromeccanica

Generazione di coppia motrice



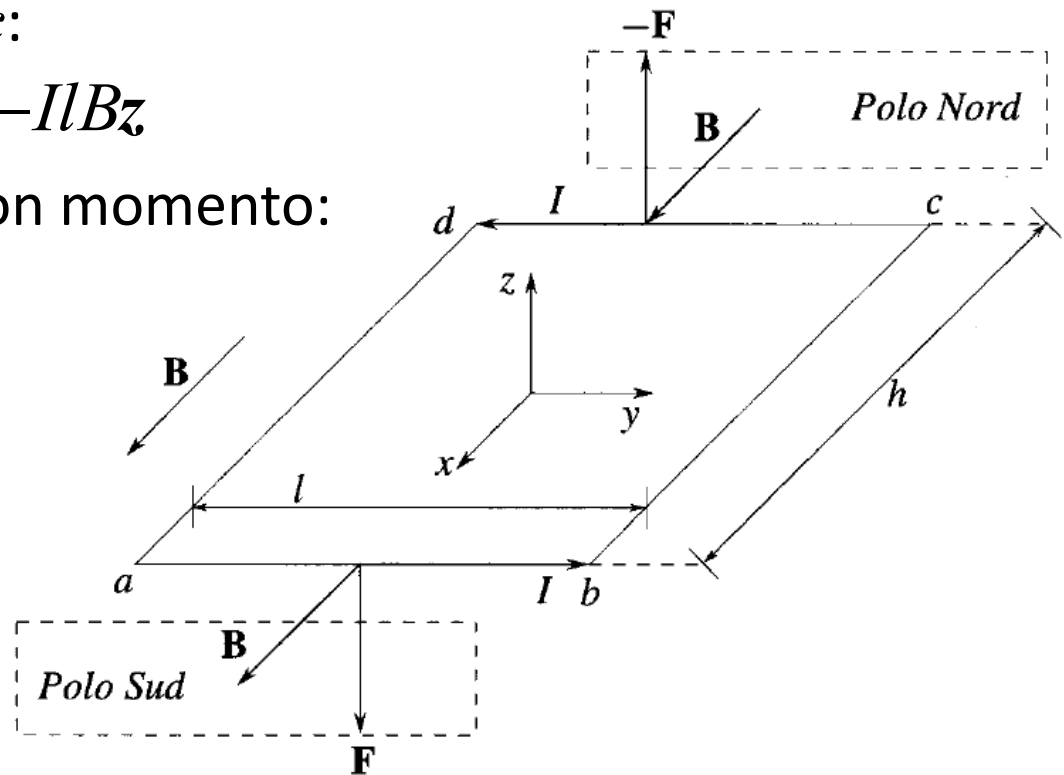
- Spira rigida rettangolare con $\mathbf{B}$ uniforme

- Forza sui lati ab e dc :

$$\mathbf{F} = IlB(\mathbf{y} \times \mathbf{x}) = -IlB\mathbf{z}$$

- Coppia sulla spira con momento:

$$\begin{aligned}\mathbf{M} &= (h\mathbf{x}) \times \mathbf{F} \\ &= -IhlB(\mathbf{x} \times \mathbf{z}) \\ &= IhlB\mathbf{y}\end{aligned}$$



conversione elettromeccanica

Generazione di coppia motrice

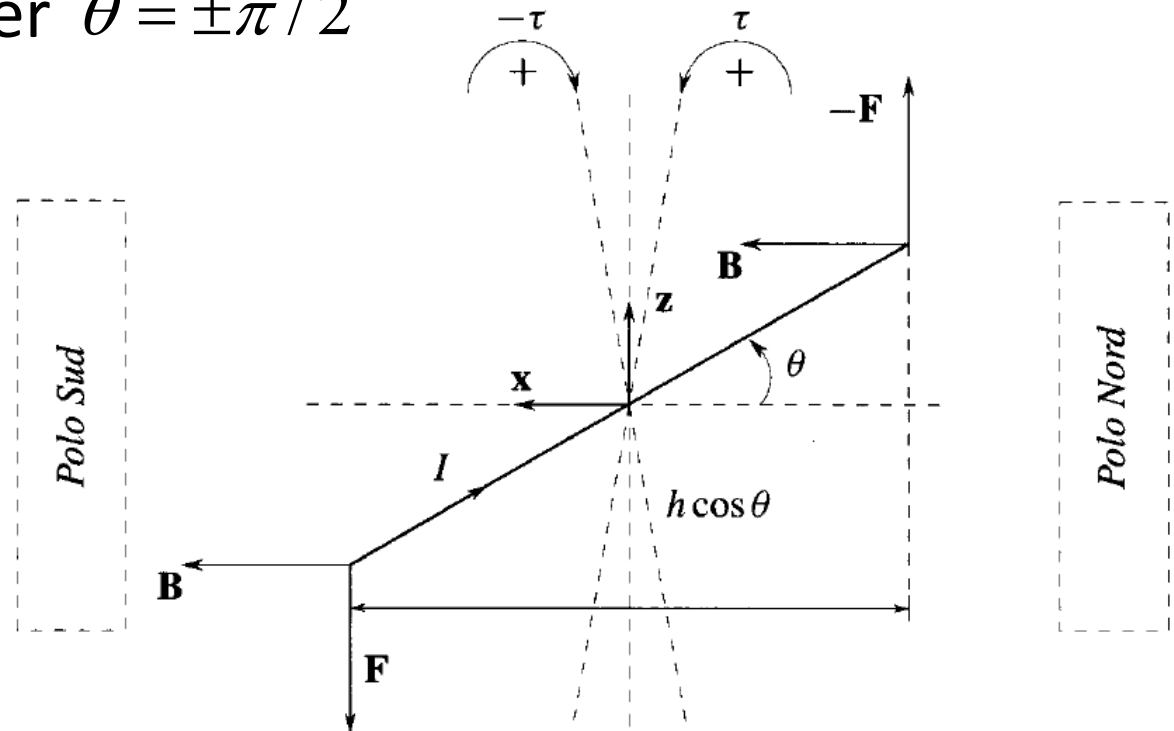


- Spira ruotata di un angolo θ

$$\mathbf{M} = I h l B \cos \theta \mathbf{y} = \tau \mathbf{y}$$

che si annulla per $\theta = \pm \pi / 2$

NOTA: i conduttori laterali ora, per simmetria, produrranno forze non nulle, ma uguali e contrarie, che quindi non daranno contributo alla coppia





conversione elettromeccanica

Generazione di forza controelettromotrice

- **Legge di Faraday** dell'induzione elettromagnetica:

In ogni circuito sottoposto all'azione di un campo magnetico viene indotta una f.e.m. uguale alla derivata rispetto al tempo del flusso magnetico attraverso il circuito, col segno cambiato

$$E = - \frac{d\Phi_m}{dt}$$

La f.e.m. indotta si oppone sempre al passaggio della corrente che ha generato il moto del circuito → **forza controelettromotrice**



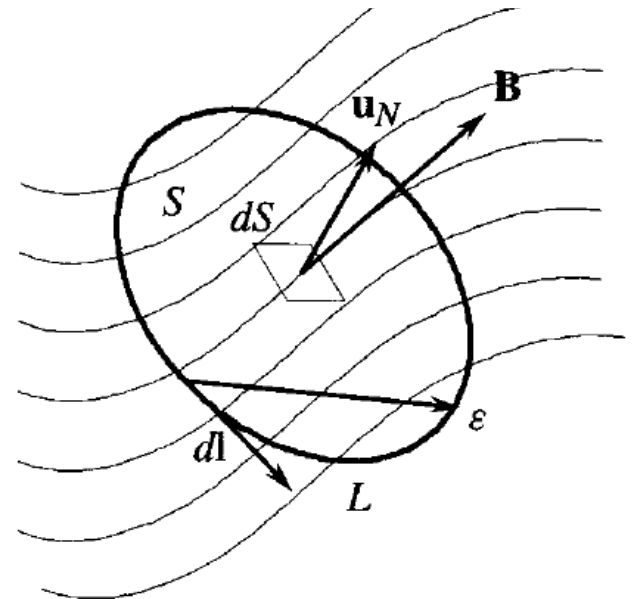
conversione elettromeccanica

Generazione di forza controelettromotrice

Detta L la lunghezza del circuito, il **campo elettrico** risulta

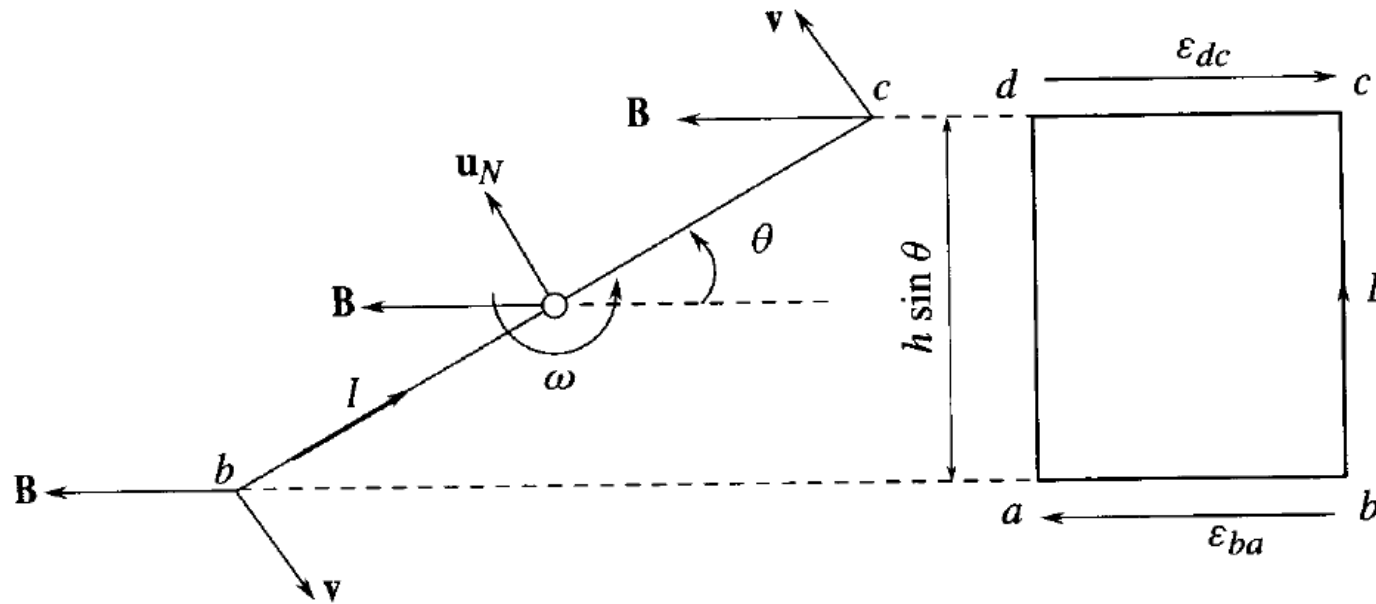
$$E = \oint_L \boldsymbol{\varepsilon} \cdot d\mathbf{l} \quad \text{da cui la L. Faraday} \quad E = \oint_L \boldsymbol{\varepsilon} \cdot d\mathbf{l} = -\frac{d}{dt} \int_S \mathbf{B} \cdot \mathbf{u}_N dS$$

dove S è una superficie avente il circuito L come contorno, $\mathbf{u}_N$ normale alla superficie dS assegnata in modo tale che $d\mathbf{l}$ ruoti in senso antiorario rispetto ad $\mathbf{u}_N$



conversione elettromeccanica

Generazione di forza controelettromotrice



- **Spira in rotazione:** flusso magnetico nella spira

$$\phi_m = \mathbf{B} \cdot \mathbf{u}_N S = \mathbf{B} \cdot \mathbf{u}_N h l = h l B \sin \theta$$

da cui la f.e.m.
$$E = -\frac{d\Phi_m}{dt} = -\omega h l B \cos \theta$$



conversione elettromeccanica

Generazione di forza controelettromotrice

- Quando il campo magnetico è costante nel tempo (eventualmente anche nello spazio) è conveniente calcolare la f.e.m. con la **legge del flusso tagliato**:

La f.e.m. indotta in un conduttore in moto relativo con velocità $\mathbf{v}$ rispetto ad un campo magnetico $\mathbf{B}$ costante nel tempo si può supporre dovuta ad un campo elettrico equivalente definito come:

$$\boldsymbol{\varepsilon} = \mathbf{v} \times \mathbf{B}$$



conversione elettromeccanica

Generazione di forza controelettromotrice

- Dalla circuitazione del campo elettrico indotto

$$E = \oint_L \boldsymbol{\varepsilon} \cdot d\mathbf{l} = -\varepsilon_{dc}l - \varepsilon_{ba}l$$

dove i lati ad e bc non danno contributi perché perpendicolari al campo elettrico

- Ricordando che $v = h\omega/2$ i moduli risultano

$$\varepsilon_{dc} = \varepsilon_{ba} = Bv \sin\left(\frac{\pi}{2} - \theta\right) = Bv \cos\theta = B \frac{h}{2} \omega \cos\theta$$

da cui si ricava l'espressione già trovata



conversione elettromeccanica

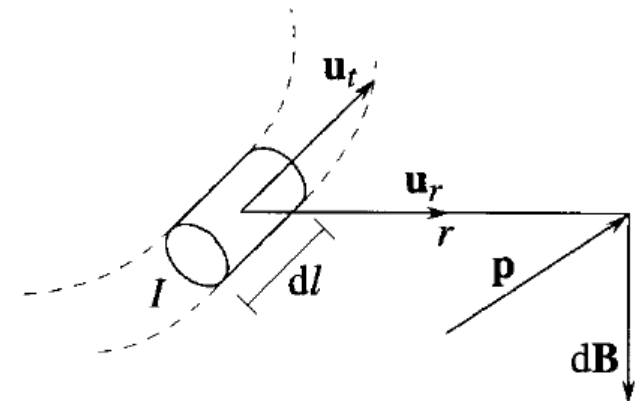
Principi generali di conversione elettromeccanica

- **Principio di conversione elettromeccanica:**
meccanismo di trasformazione dell'energia elettrica
in energia meccanica
- **Legge di Ampère-Laplace**

$$\mathbf{B}(\mathbf{p}) = \frac{\mu_0}{4\pi} \oint_L I \frac{\mathbf{u}_t \times \mathbf{u}_r}{r^2} d\mathbf{l}$$

$$\mathbf{p} = r\mathbf{u}_r$$

campo magnetico generato da una
spira di corrente *chiusa*





conversione elettromeccanica

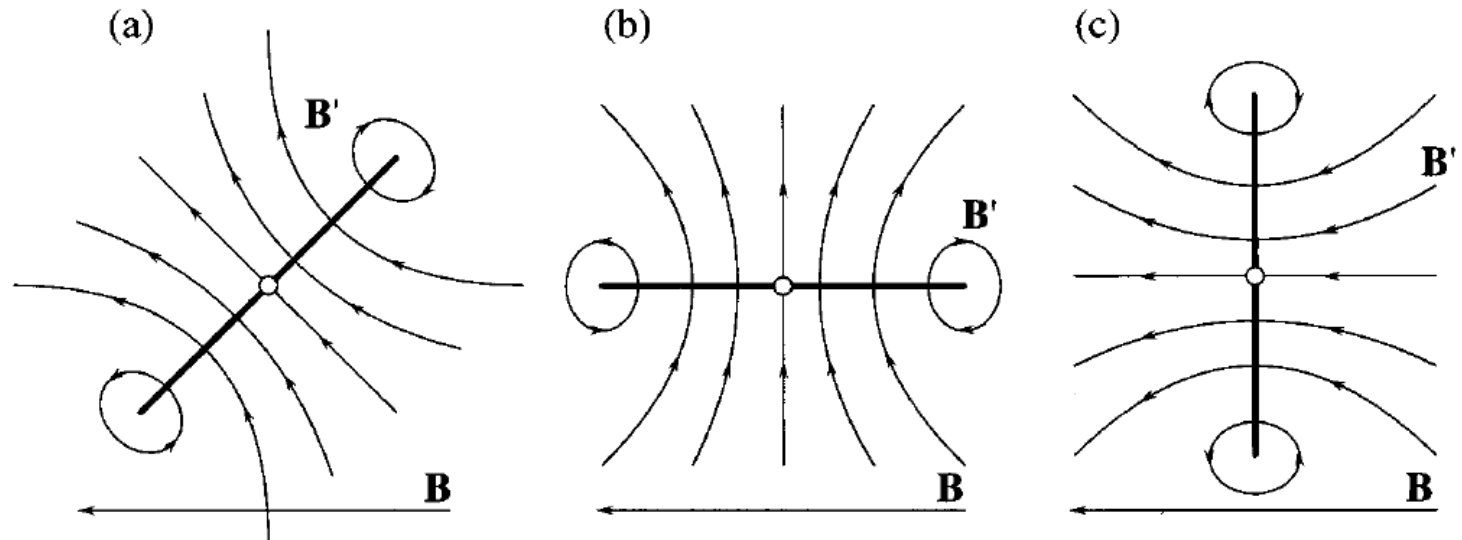
Principi generali di conversione elettromeccanica

- I campi magnetici possono essere generati da magneti permanenti e dalla circolazione di corrente
- La conversione elettromeccanica si attua mediante l'interazione di due campi magnetici secondo il **principio dell'allineamento**:

In una regione dello spazio sede di due campi magnetici si manifesta un'azione meccanica che tende ad allineare i campi stessi

conversione elettromeccanica

Principi generali di conversione elettromeccanica

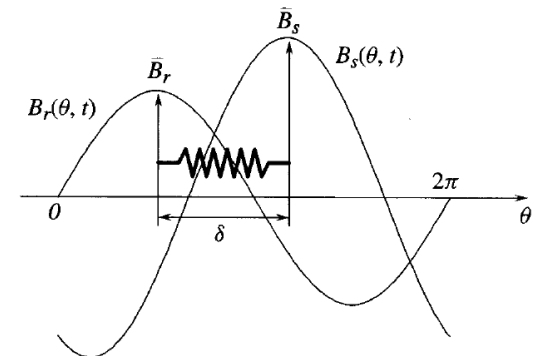
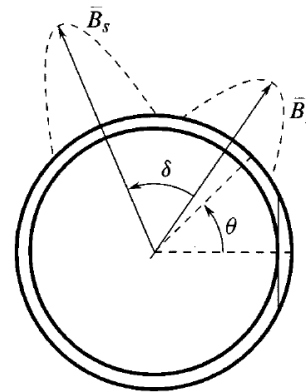


- (b) coppia di allineamento massima
- (c) coppia di allineamento nulla

conversione elettromeccanica

Principi generali di conversione elettromeccanica

- ... nei motori elettrici:
 - Campi magnetici generati nel traferro diretti radialmente
 - Intensità dipende dall'angolo di rotazione e dal tempo
 - Onde di intensità di campo magnetico
 - Interazione dei campi di statore e rotore genera la conversione elettromeccanica
 - Generazione di una coppia che tende ad allineare i campi
- $$\tau_m = k \bar{B}_r \bar{B}_s \sin \delta$$
- Quiete relativa $\rightarrow \delta = cost$





conversione elettromeccanica

Principi generali di conversione elettromeccanica

- **Principio di separazione dei motori**

- Coppia dipende separatamente dall'andamento *temporale* della corrente e da quello *spaziale* dei campi
 - Dipendenza spaziale → modalità realizzazione degli avvolgimenti e disposizione dei magneti permanenti
 - Dipendenza temporale → andamento imposto alle correnti negli avvolgimenti

$$B(\theta, t) = kf(\theta)g(i(t))$$



conversione elettromeccanica

Principi generali di conversione elettromeccanica

- ... riassumendo
 - La generazione di coppia nei motori elettrici è basata sull'interazione di due onde di intensità di campo magnetico, una generata da magneti o avvolgimenti posti sullo statore, l'altra generata da magneti o avvolgimenti posti sul rotore
 - Alimentando opportunamente gli avvolgimenti si cerca di mantenere uno sfasamento di 90° fra le due onde di intensità di campo magnetico, al fine di ottenere la massima generazione di coppia a parità di ogni altra condizione
 - Nel caso del motore a corrente continua la condizione di perpendicolarità viene mantenuta dal collettore, si ottiene pertanto una coppia intrinsecamente indipendente dalla posizione la cui ampiezza è proporzionale all'ampiezza della corrente di alimentazione



misura e sensori

Introduzione

- I **dispositivi di misura** rivestono un ruolo fondamentale nei *sistemi di controllo di processo*
- Vanno limitati i ritardi e il rumore di misura
- Elementi particolarmente critici sono:
 - Scelta del dispositivo (caratteristica intrinseca)
 - Localizzazione nel processo
 - Modalità di installazione
- Parametri fondamentali sono:
 - Precisione statica di misura
 - Risposta dinamica



misura e sensori

Introduzione

- I dispositivi di misura (*trasmettitore*) sono costituiti da:
 - **Elemento sensibile primario**
 - Fornisce un segnale o un parametro funzione del *misurando*
 - **Trasduttore**
 - Trasforma il segnale fornito dal sensore (in un segnale elettrico)
 - **Amplificatore**
 - Segnale di uscita normalizzato
- *Esempi:*
 - Resistore ($R = f(T)$) + Ponte di Wheatstone
 - Termocoppia: sensore e trasduttore coincidono
- **Nota:** terminologia (sensore, trasduttore, trasmettitore) spesso confusa!

proprietà generali dei sensori

Accuratezza



- **Accuratezza:** indica quanto una misura si avvicina al valore “vero” del misurando
 - **Valore vero:** non conoscibile → valore di riferimento ottenuto mediante una *misurazione esemplare*
 - La misura è influenzata da varie grandezze oltre al misurando: temperature, umidità, pressione, ecc.
- *Definizione ingegneristica:* massimo scostamento tra le misure (letture) fornite dal sensore e la curva di taratura sull'intero campo di misura



proprietà generali dei sensori

Accuratezza

- Viene fornito un unico valore (sintetico):
 - In percentuale del campo di misura (o fondo scala)

$$\varepsilon_f = 100 \cdot \frac{X_m - X_v}{X_{FS}}$$

dove X_m è il valore misurato più distante dal valore “vero” X_v

- In percentuale della misura

$$\varepsilon_a = 100 \cdot \frac{X_m - X_v}{X_v}$$

- Entrambe le definizioni dipendono da X_v
- ε_f è poco significativo se l’escursione X_{FS} è ampia



proprietà generali dei sensori

Accuratezza

- L'accuratezza viene anche espressa:
 - in valore assoluto, come massimo errore sul campo di misura
 - come funzione (tipicamente lineare) della misura
- La mancanza di accuratezza deriva da **errori sistematici**
 - Errori di calibrazione del sensore
- La **calibrazione** consente di far corrispondere l'uscita del sensore ai valori veri di riferimento
 - Va ripetuta periodicamente a causa di usura, sporcamento, ecc.



proprietà generali dei sensori

Precisione

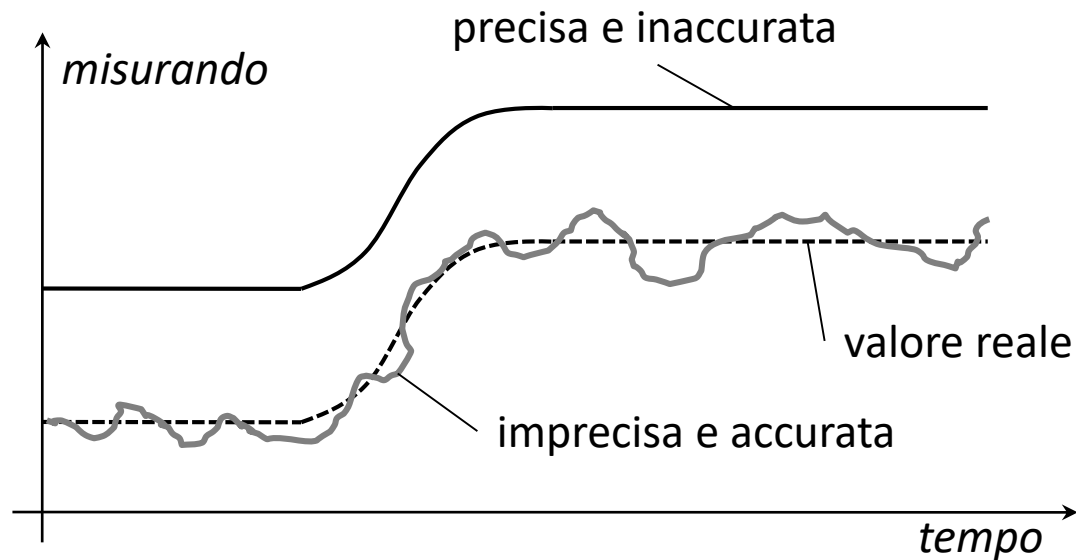
- La **precisione** (o **ripetibilità**) è riferita alla *ripetitività* (o *riproducibilità*) della misura
 - Dispersione di misure successive dello stesso misurando nelle stesse condizioni (stesso valore reale)
- *Definizione*: massimo scostamento tra una generica lettura e la migliore stima della misura
- Deriva da **errori casuali**
 - Può essere migliorata con medie di letture successive



proprietà generali dei sensori

Accuratezza vs. Precisione

- Nel controllo di processi la precisione è spesso più importante dell'accuratezza (più costosa)
- Confronto tra precisione e accuratezza:





proprietà generali dei sensori

Rangeability

- **Rangeability:** l'accuratezza e la precisione si applicano solo ad una parte del campo di misura
 - Nella parte rimanente non sono specificati (scadenti!)
- *Definizione:* rapporto tra l'estremo superiore (fondo scala) e quello inferiore, normalizzato all'unità, del campo di misura nel quale si applicano i dati di accuratezza e precisione
 - *Esempio:* sensore di portata con rangeability 20:1, con FS di 300 kg/s e accuratezza del 1% (della misura), registra, con tale accuratezza, portate comprese tra 15 e 300 kg/s



proprietà generali dei sensori

Caratteristica statica, sensibilità e risoluzione

- **Caratteristica statica:** la misura y fornita da un sensore è in generale una funzione (non lineare) del misurando u , $y = y(u)$, detta caratteristica statica
 - *Sensori lineari:* caratteristica statica lineare nell'ambito dell'accuratezza/precisione (campi di misura ristretti)
- **Sensibilità:** rapporto tra la variazione dell'uscita del sensore e la corrispondente variazione della misura

$$S(y) = \frac{\Delta y}{\Delta u}$$

- Sensori lineari $\rightarrow$ sensibilità = costante = **guadagno statico**



proprietà generali dei sensori

Caratteristica statica, sensibilità e risoluzione

- **Risoluzione:** minima variazione della variabile misurata che dà luogo a una variazione rilevabile dell'uscita

- Espressa come percentuale del campo di misura

$$R = 100 \cdot \frac{\Delta U_{\min}}{U_{\max} - U_{\min}}$$

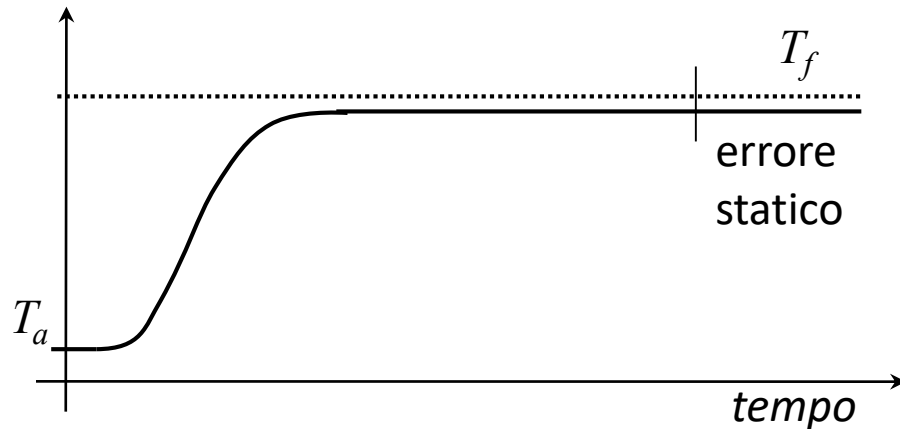
- Può essere diversa in punti diversi del campo di misura
 - Utile il valore medio e il minimo (*risoluzione massima*)
- *Soglia di sensibilità*: risoluzione intorno allo zero



proprietà generali dei sensori

Comportamento dinamico

- La risposta di un sensore sarà caratterizzata da un ritardo rispetto alle variazioni del misurando
- Esempio: *termocoppia*



La giunzione è incapsulata in un *pozzetto metallico* per proteggerla dalla corrosione, con capacità termica C , area e coefficiente di scambio A e γ , rispettivamente

$$C \frac{dT}{dt} = \gamma A (T_f - T) \quad \text{con} \quad \tau = \frac{C}{\gamma A} \approx 10\text{s}$$



proprietà generali dei sensori

Comportamento dinamico

- **Caratteristica dinamica:** f.d.t. per piccole variazioni intorno ad un punto di linearizzazione $\bar{y}$

$$F(s) = \frac{\Delta y}{\Delta u} \quad \text{con} \quad \lim_{s \rightarrow 0} F(s) = S(\bar{y})$$

- Spesso viene fornita una descrizione sintetica di $F(s)$
 - Banda passante, tempo di risposta, risonanze, ecc.
- Dinamica del sensore veloce rispetto al processo
 - Velocità di risposta è un parametro fondamentale per i sistemi di controllo di prestazioni elevate



proprietà generali dei sensori

Scelta dei sensori

- Fattori da considerare nella **scelta di un sensore**:
 - Campo (o Range, es. $200\div 600^{\circ}\text{C}$) e intervallo (o Span, es. 400°C) di misura
 - Accuratezza, precisione, sensibilità e risoluzione desiderate
 - Rangeability
 - Caratteristica dinamica
 - Livello di affidabilità
 - Costi (acquisto, installazione, manutenzione)
 - Problemi di installazione e pericolosità
 - Materiale di costruzione



proprietà generali dei sensori

Scelta dei sensori

- Scelta dipende dalla specificità del caso considerato
- Da considerare le normative nazionali/internazionali
- Campo di misura e **risoluzione complessiva** della misura si influenzano mutuamente nel caso di *conversione digitale*
 - *Esempio*: termocoppia con campo $0 \div 1000$ °C con conversione digitale a 12 bit ($0 \div 4095$) determina una perdita di risoluzione di 0.25°C
 - Se si riduce il campo di misura aumenta la risoluzione complessiva



sensori di temperatura

Termocoppia

- Le **termocoppie (TC)** sono i sensori di temperatura più diffusi in ambito industriale
- Sono costituite da due fili di metalli (o leghe) diversi, uniti ad un'estremità
- Muniti di apposita guaina protettiva costituiscono una *sonda* adatta ad essere immersa in un fluido o annegata in un metallo per rilevarne la temperatura
- Il loro principio di funzionamento è basato sul *fenomeno termoelettrico di Seebeck*

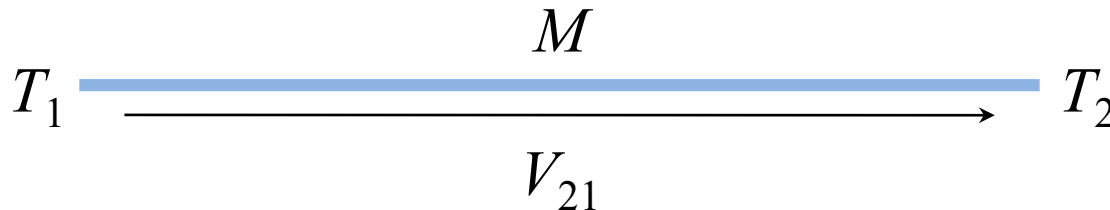


sensori di temperatura

Termocoppia: fenomeno termoelettrico

- **Fenomeno termoelettrico:** un tratto di filo metallico sottoposto a gradiente termico produce una differenza di potenziale, o *forza elettromotrice di Seebeck*

$$V_{21} = \int_{T_1}^{T_2} \sigma(T) dT$$



dove $\sigma(T)$ è una proprietà del metallo detta *coefficiente di Seebeck*

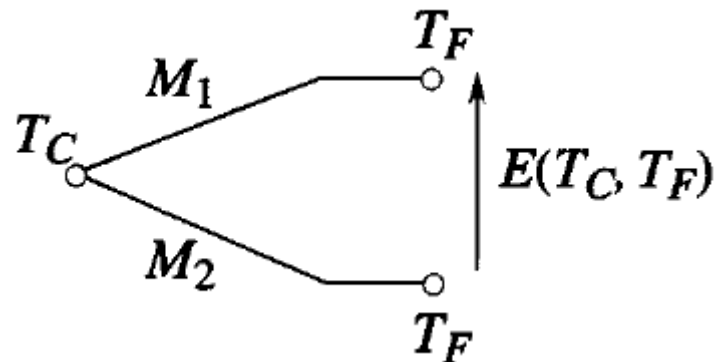


sensori di temperatura

Termocoppia: fenomeno termoelettrico

- Unendo due estremità di due fili metallici per una estremità, si manifesta una tensione funzione della temperatura della giunzione (**giunto caldo**) e di quella delle altre due estremità (**giunto freddo**)

$$E(T_C, T_F) = \int_{T_F}^{T_C} (\sigma_2(T) - \sigma_1(T)) dT$$





sensori di temperatura

Termocoppia: caratteristica statica

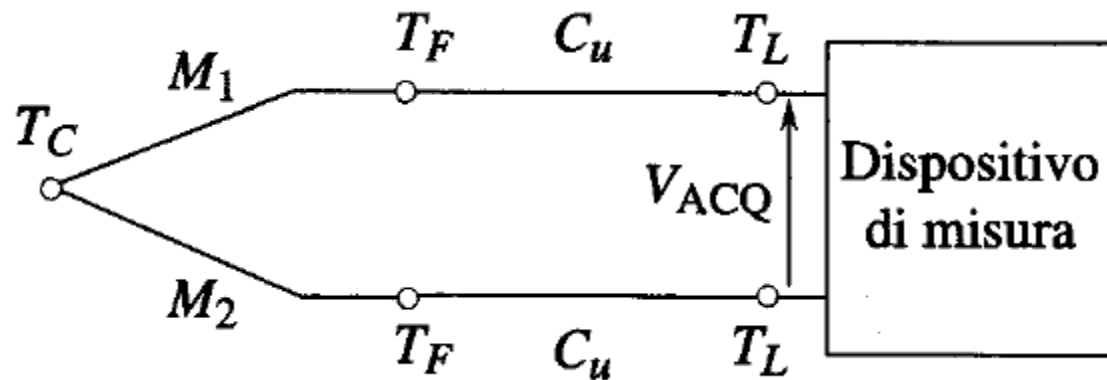
- Le termocoppie in commercio sono descritte, in funzione del tipo, dalla **caratteristica statica** $E(T_C, 0)$
 - Scegliendo opportunamente le leghe si ottengono caratteristiche statiche lineari per campi limitati
 - Per campi di misura molto ampi o accuratezze elevate è necessaria una *compensazione* (linearizzazione)
 - $E(T_C, 0)$ è fornita in forma tabellare con risoluzione di 1°C
 - La caratteristica statica è facilmente rilevabile in laboratorio



sensori di temperatura

Termocoppia: rilevazione della caratteristica statica

- La misura della $E(T_C, 0)$ avviene immergendo il giunto freddo in un bagno acqua-ghiaccio



- V_{ACQ} la differenza di potenziale misurata $\rightarrow E(T_C, 0) = V_{ACQ}$
- I fili di rame non influenzano la misura $\rightarrow$ stesso salto di temperatura, quindi stessa d.d.p.



sensori di temperatura

Termocoppia: compensazione del giunto freddo

- Nelle normali condizioni di funzionamento $T_F \neq 0$
- **Compensazione del giunto freddo:** risalire dalla differenza di potenziale acquisita V_{ACQ} con $T_F \neq 0$ alla $E(T_C, 0)$, necessaria per ottenere T_C
 - Realizzata via software (numerosi TC) o con appositi circuiti integrati (soluzioni dedicate) → stesso principio
- Compensazione software

$$E(T_C, T_F) = \int_0^{T_C} (\sigma_2(T) - \sigma_1(T)) dT - \int_0^{T_F} (\sigma_2(T) - \sigma_1(T)) dT$$
$$E(T_C, 0) \qquad E(T_F, 0)$$



sensori di temperatura

Termocoppia: compensazione del giunto freddo

$$E(T_C, T_F) = E(T_C, 0) - E(T_F, 0)$$

- Nota* T_F , dalla caratteristica statica si risale a $E(T_F, 0)$, quindi dalla lettura $V_{ACQ} = E(T_C, T_F)$ si ricava

$$E(T_C, 0) = V_{ACQ} + E(T_F, 0)$$

- Ancora utilizzando la caratteristica statica si ricava la temperatura desiderata T_C

(*) La misura di T_F viene eseguita raggruppando i cavi delle termocoppie omogenee nelle cosiddette *cassette di compensazione*, dove viene rilevata (o condizionata) la temperatura

- Uso di *cavi (compensati) di termocoppia*, omogenei con i materiali delle termocoppie, per collegare le TC alle cassette di compensazione → costosi



sensori di temperatura

Termocoppia: caratteristiche di misura

- Campi di misura delle TC: $-200 \div 2750^{\circ}\text{C}$
- Tensioni di uscita: $-10 \div 50\text{mV}$
- Sensibilità: $10 \div 50 \mu\text{V}/^{\circ}\text{C}$
- Poco costose $\rightarrow$ adatte ad uso in ambienti “difficili”
- Poco accurate ($0.1 \div 4^{\circ}\text{C}$ al variare della misura)
- Vanno linearizzate per intervalli di misura ampi
- Sonde protettive (*pozzetti*) $\rightarrow$ tempi di risposta lunghi



sensori di temperatura

Termocoppia: caratteristiche di misura

- Caratteristiche delle TC standard più diffuse

Tipo	Coppie metalli/leghe	Range normale [°C]	Range esteso [°C]
• J	ferro/costantana*	$-100 \div 750$	$-200 \div 1100$
• K	chromel**/alumel***	$0 \div 1100$	$-100 \div 1370$
E	chromel/costantana	$-150 \div 500$	$-200 \div 1000$
• T	rame/costantana	$-200 \div 300$	$-230 \div 400$
B	Pt80/Rh20/Pt94-Rh6	$600 \div 1650$	$100 \div 820$
S	Pt90-Rh10/Pt	$550 \div 1500$	$0 \div 1700$
R	Pt90-Rh13/Pt	$550 \div 1500$	$0 \div 1700$
N	Nicrasil-nichel	$0 \div 1300$	

*lega rame-nichel

**lega nichel-cromo

***lega nichel-alluminio-silicio

sensori di temperatura

Termocoppia: caratteristiche di misura



- Termocoppie più usate:

- Termocoppie **K** (0÷1100°C)

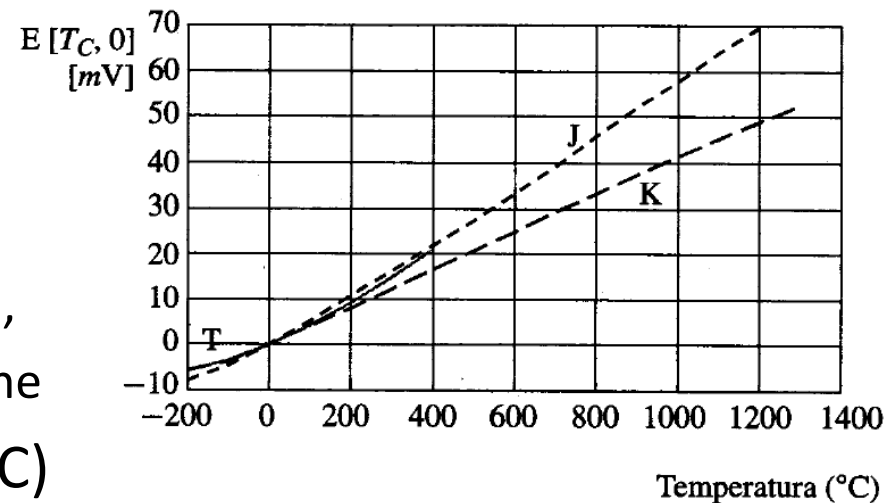
- Basso costo, sensibilità moderata, bassa accuratezza, alta resistenza alla ossidazione

- Termocoppie **J** (–100÷750°C)

- Basso costo, alta sensibilità, accuratezza moderata

- Termocoppie **T** (–200÷300°C)

- Costo contenuto, sensibilità moderata, alta accuratezza, adatte per basse temperature, non richiedono compensazione se usate in coppia per misure differenziali

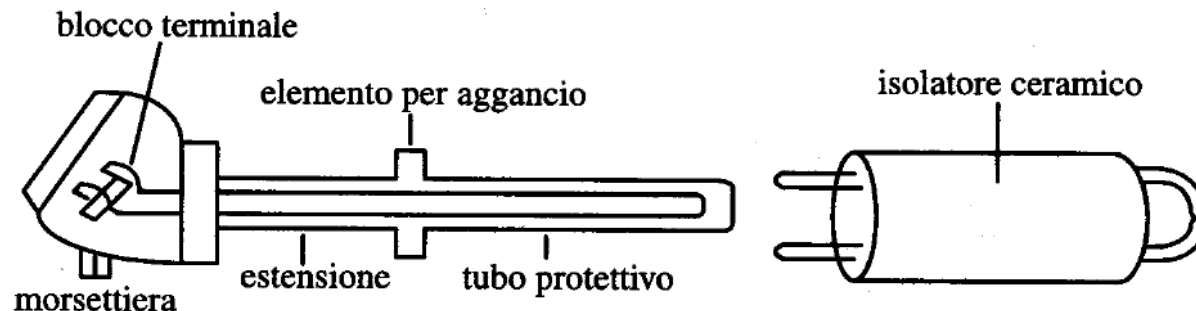


sensori di temperatura

Termocoppia: aspetti costruttivi



- Molto importante la qualità dei fili metallici utilizzati
 - Limiti di calibrazione molto accurati
 - Installati con molta cura
 - Fabbricati in diversi diametri
 - Fili sottili → pronta risposta
 - Fili spessi → maggiore resistenza alla corrosione e temperatura
 - Fili isolati tra loro e dalla guaina oppure *giunzione a massa*





sensori di temperatura

Termoresistenze

- **Termoresistenze** o **RTD** (Resistance Temperature Detector): sfruttano la dipendenza della resistenza elettrica dei metalli (*es. platino*) dalla temperatura
 - Bassa sensibilità
 - Elevata linearità
 - Per intervalli di misura estesi e elevata accuratezza per la caratteristica statica si utilizzano approssimazioni polinomiali del terzo ordine

$$R(T) = R_0(1 + \alpha_1 T + \alpha_2 T^2 + \alpha_3 T^3) \quad \text{con } R_0 = R(0)$$



sensori di temperatura

Termoresistenze

- Termoresistenze più diffuse:

RTD	Range normale
Pt: 100 Ω DIN (4376)	$-180 \div 800$ °C
Pt: 100 Ω JIS (C-1604)	$-180 \div 650$ °C
Ni: 120 Ω Ed7	$-45 \div 315$ °C
Cu: Ω	$-20 \div 250$ °C

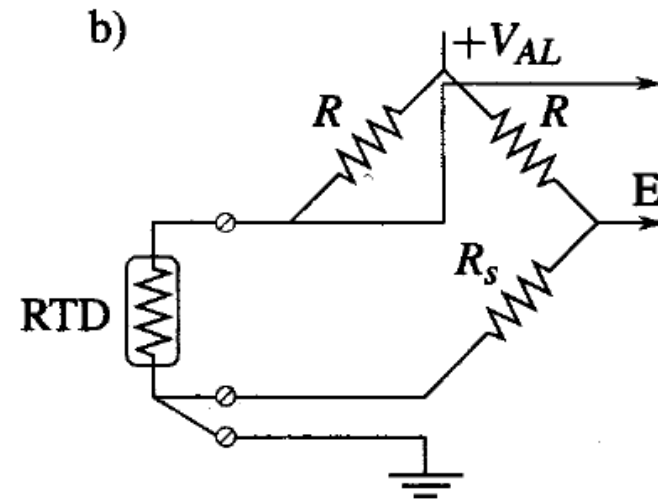
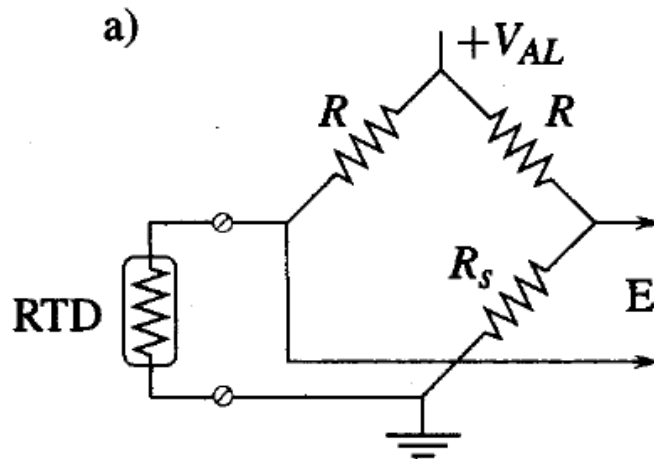
- *Esempio:* la RTD più usata è la **PT100**, con valore nominale 100 Ω a 0°C e sensibilità 0.4 Ω /°C

sensori di temperatura

Termoresistenze



- Trasduzione mediante **generatore di corrente**
- Trasduzione mediante **ponte di Wheatstone**



- b) **collegamento a 3 fili** per ripartire la resistenza dei cavi di collegamento (lunghi) tra il ramo della RTD e quello di R_s



sensori di temperatura

Termoresistenze

- Tensione di uscita dal ponte di Wheatstone

$$E(T) = V_{AL} \left(\frac{R_T(T)}{R + R_T(T)} - \frac{R_S}{R + R_S} \right)$$

- Considerata una temperatura di riferimento T_0 , scelta R_S per bilanciare il ponte a tale temperatura, risulta

$$R_S = R_T(T_0) \quad \text{e} \quad R_T(T) = R_S + \Delta R(T)$$

quindi se $R + R_S \gg \Delta R(T)$ risulta

$$E(T) \cong V_{AL} \frac{\Delta R(T)}{R + R_S}$$



sensori di temperatura

Termoresistenze

- **Autoriscaldamento:** la corrente è causa di dissipazione → sensore si porta ad una temperatura più alta di quella da misurare
 - Le correnti nel ponte vanno limitate a qualche mA
 - *Esempio:* Misura di temperatura nel campo 0÷500°C con una PT100, si sceglie $T_0 = 0^\circ\text{C}$ e $R_S = 100\Omega$ da cui $\Delta R(500) \cong 200\Omega$
Se $V_{AL} = 24\text{V}$, scegliendo $R = 24\text{k}\Omega$ si ottiene una corrente nella RTD di circa 1mA

La sensibilità corrispondente sarà

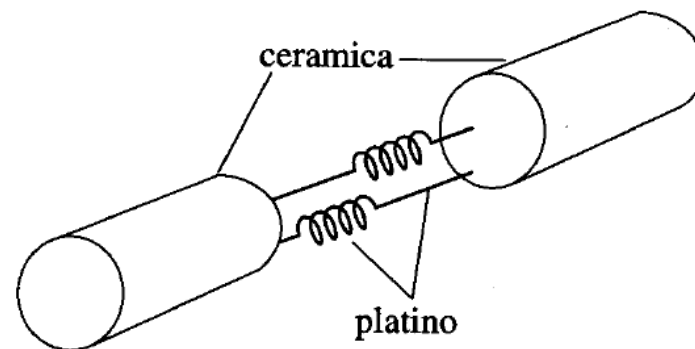
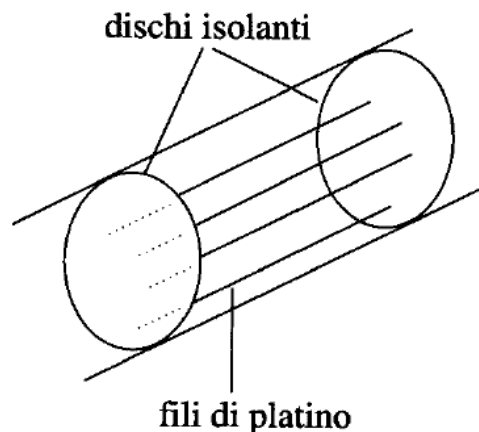
$$\frac{\Delta E}{\Delta T} = \frac{\Delta E}{\Delta R} \cdot \frac{\Delta R}{\Delta T} = 10^{-3} \text{ A} \cdot \frac{200\Omega}{500^\circ\text{C}} = 4 \cdot 10^{-4} \text{ V}/^\circ\text{C}$$

sensori di temperatura

Termoresistenze



- Realizzazione fisica:
 - Filo di lunghezza tale da avere un valore predeterminato di resistenza a 0°C, avvolto intorno ad un materiale isolante
 - Molta cura rivolta ad eliminare possibili cause di deformazioni e quindi di variazioni indesiderate della resistenza





sensori di temperatura

Termistori

- **Termistore:** dispositivo a semiconduttore o a ossido metallico avente resistenza elettrica funzione della temperatura
 - *Coefficiente di temperatura* $(\Delta R/R)/\Delta T$, ovvero la variazione percentuale della resistenza per una variazione di un grado centigrado della temperatura, alto (3÷5%)
 - Possibilità di risolvere il centesimo di grado
 - I termistori più diffusi hanno coefficiente negativi (NTC), ma ne esistono anche con coefficiente positivo (PTC)

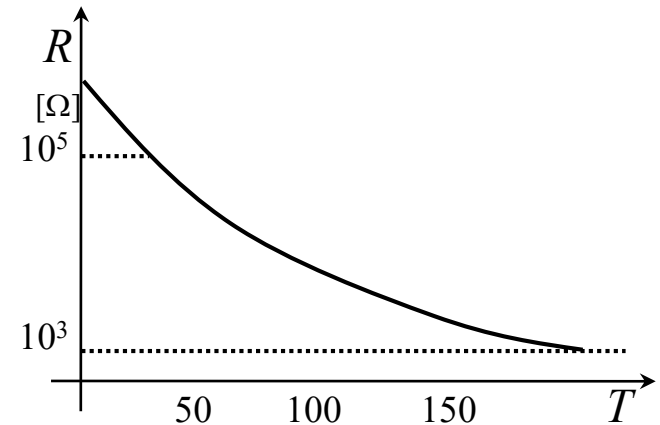
sensori di temperatura

Termistori

- *In figura*: caratteristica di un termistore NPT
- Accuratezza dell'ordine di 0.1°C
- Campo di misura ristretto (es. $-50 \div 250^{\circ}\text{C}$) per la forte variabilità della resistenza con la temperatura
- Poco lineari $\rightarrow$ compensazione:

$$\frac{1}{T} = A + B \ln(R) + C(\ln(R))^3$$

con T espressa in gradi Kelvin e i coefficienti A , B e C rilevati sperimentalmente mediante tre misure





sensori di temperatura

Termistori

- Scegliendo $|T_3 - T_1| \leq 100^\circ\text{C}$ e T_2 centrato a metà si ottiene uno scostamento massimo della curva approssimane da quella reale di $\pm 0.02^\circ\text{C}$
- Comportamento lineare si può ottenere accoppiando due termistori con coefficienti di temperatura opposti
- Condizionamento mediante ponte di Wheatstone
- Piccole dimensioni $\rightarrow$ risposta rapida
- Valori di resistenza elevati $\rightarrow$ poco sensibili alla resistenza dei cavi di collegamento
- Fragili $\rightarrow$ occorrono opportune protezioni



sensori di pressione

Introduzione

- Dispositivi usati per la **misura di pressione** di fluidi:
 - Basati sulla *misura dell'altezza di una colonna di liquido*
 - Manometri a colonna
 - Basati sulla *misura di deformazioni di un elemento elastico*
 - Tipo di elemento elastico
 - Modo con cui la deformazione è trasdotta in:
 - Spostamento → indicatori
 - Segnale elettrico → trasmettitori
- Piccoli impianti → indicatori locali (**manometri**)
- Misura di pressioni: *assolute, relative e differenziali*
 - Misura differenziale usata anche per misura di portata



sensori di pressione

Dispositivi basati sulla misura dell'altezza di una colonna di liquido

- **Manometri a colonna (o a U):** misurano in modo accurato la differenza tra due pressioni, trasducendola in altezza di colonna di liquido di densità nota
 - Liquidi utilizzati:
 - Acqua
 - Mercurio (13.5 volte più denso dell'acqua → pressioni elevate)

sensori di pressione

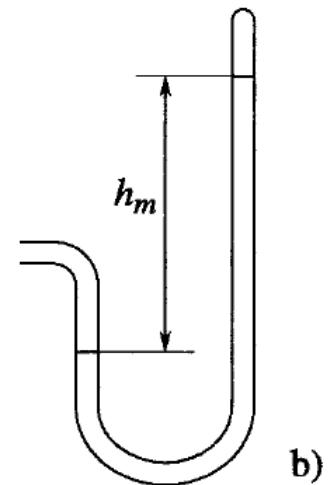
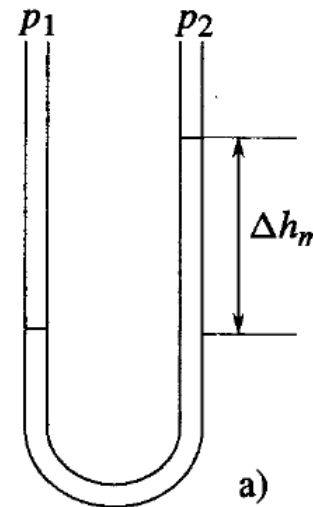
Dispositivi basati sulla misura dell'altezza di una colonna di liquido



- Manometri a **tubo aperto** a) in condizioni stazionarie:

$$p_1 - p_2 = \rho_m g \Delta h_m$$

con ρ_m densità del *liquido manometrico*, avendo trascurato la densità dei liquidi alla pressione p_1



- p_2 spesso è la pressione atmosferica
→ dall'altezza Δh_m si risale alla misura di pressione graduando la colonna in valori di p_1



sensori di pressione

Dispositivi basati sulla misura dell'altezza di una colonna di liquido

- Manometri a **tubo chiuso** b) contenenti mercurio sono utilizzati per misurare il grado di vuoto
 - **Barometro**: misura della pressione atmosferica
- Misure molto accurate su campi ristretti
 - Accuratezza di 0.5mm su
 - 10÷1500mm di Hg, corrispondenti a 0.01÷2 bar
 - 10÷2500mm H₂O , corrispondenti a 0.001÷0.25 bar



sensori di pressione

Dispositivi basati sulla misura di deformazione di un elemento elastico

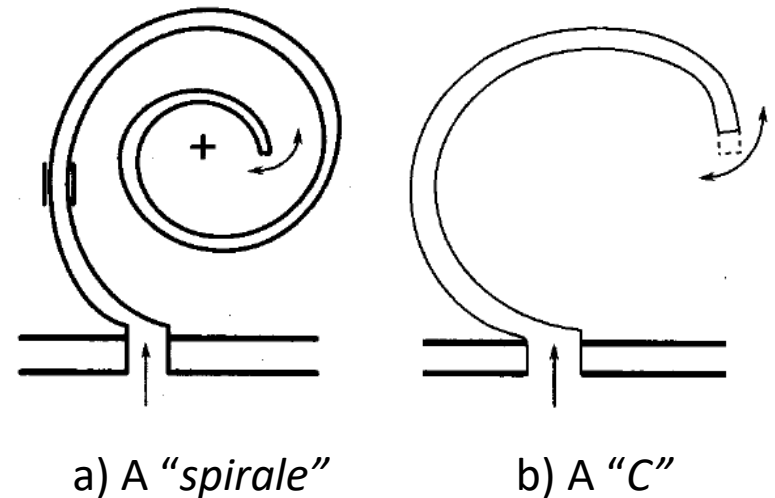
- Sensori costituiti da
 - un elemento che si deforma proporzionalmente alla pressione sotto misura
 - un dispositivo che trasduce la deformazione in
 - uno spostamento di un indicatore
 - un segnale elettrico
- Elementi deformabili:
 - Tubo di Bourdon, soffietti
- Trasduttori:
 - Membrane ceramiche, diaframmi

sensori di pressione

Tubo di Bourdon

- **Tubo di Bourdon**

- Usato per misure locali
- Tubo di sezione non circolare curvato
 - La pressione del fluido provoca la deformazione del tubo
- Deformazione trasdotta
 - nello spostamento di un indicatore con un meccanismo a leva
 - in un segnale elettrico (es. mediante potenziometri)
- In funzione del materiale, campi da 0.01 a 10000 bar

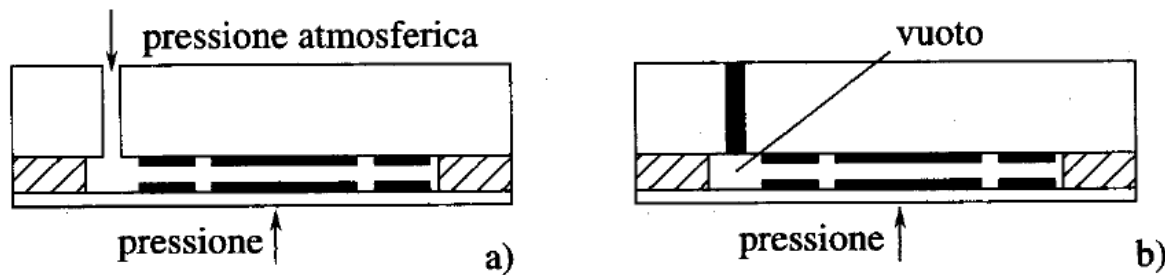


sensori di pressione

Celle di misura capacitave



- **Celle di misura capacitave**



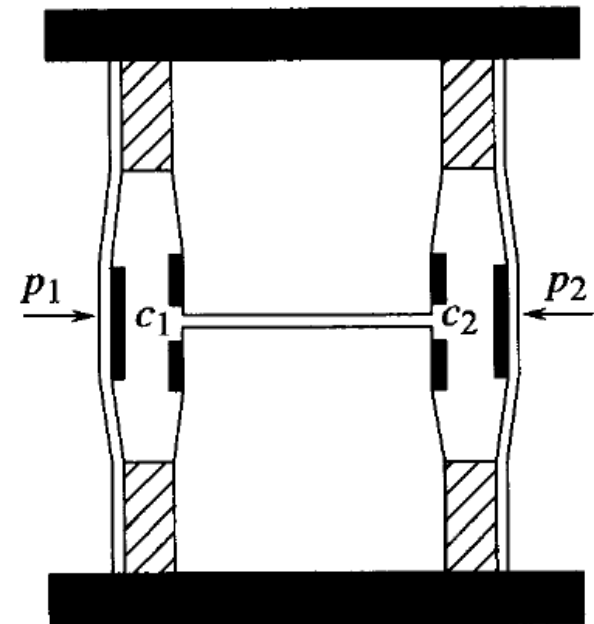
- La deformazione della membrana ceramica per effetto della pressione fa variare la distanza d tra le armature (lamine metalliche) di un condensatore
- La capacità varia con la relazione $C = \varepsilon_r \varepsilon_0 A/d$
- Misura di pressioni *relative* a) o *assolute* b)
- Trasduzione con ponte di Wheatstone alimentato in alternata, con raddrizzamento in uscita



sensori di pressione

Celle di misura capacitve

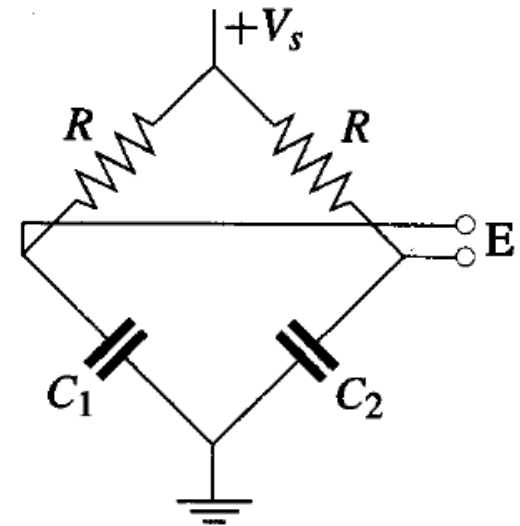
- Misura di *pressioni differenziali*
 - Celle ceramiche a doppio condensatore
 - Liquido dielettrico si sposta in ragione della differenza di pressione



sensori di pressione

Celle di misura capacitive

- Le due capacità sono inserite in due rami contigui di un ponte di Wheatstone
 - Raddoppio della sensibilità
 - Variazioni della densità del fluido dovute alla temperatura non influenzeranno la misura: stessa variazione di capacità → non provocano sbilanciamento





sensori di pressione

Trasduzione di una deformazione in un segnale elettrico

- **Estensimetri (Strain Gauge)**

- Costituiti da fili o lamine conduttrici metalliche
- Resistenza dipende dalla lunghezza e sezione:

$$R = \rho \frac{l_0}{A_0}$$

ρ resistività del metallo

- La lunghezza del conduttore varia in modo proporzionale alla forza di trazione applicata: volume costante →

$$V = A_0 l_0 = (l_0 + \Delta l)(A_0 - \Delta A)$$

$$R = \rho \frac{l_0 + \Delta l}{A_0 - \Delta A}$$



sensori di pressione

Trasduzione di una deformazione in un segnale elettrico

da cui

$$R = \rho \frac{l_0}{A_0} \left(1 + 2 \frac{\Delta l}{l_0} \right)$$

ovvero un *legame lineare* tra variazione di lunghezza e variazione di resistenza

$$\Delta R = 2R_0 \frac{\Delta l}{l_0}$$

– Legame resistenza-lunghezza: *Gain Factor* (~ 2)

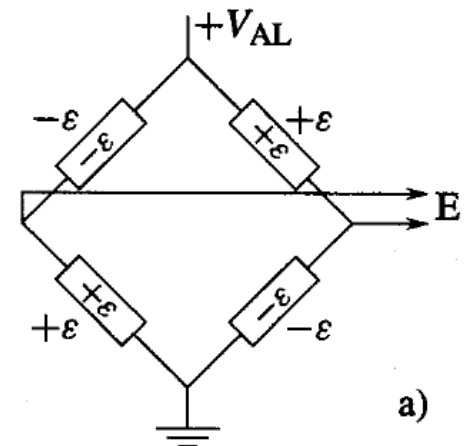
$$GF = \frac{\Delta R / R}{\Delta l / l}$$

- con leghe si ottengono *GF* maggiori → aumento della sensibilità

sensori di pressione

Trasduzione di una deformazione in un segnale elettrico

- Resistenza a riposo $\sim 100\Omega$ (es. 120Ω)
- Allungamento consentito $0.2\div 5\%$ della lunghezza a riposo
- Resistenza rilevata con ponte di Wheatstone
 - Compensazione delle variazioni di resistenza con la temperatura:
 - Due estensimetri su due lati contigui del ponte, uno soggetto a trazione ed uno soggetto a compressione $\rightarrow$ x2 la sensibilità
 - Quattro estensimetri ... x4 la sensibilità
- Mastici adesivi per fissaggio diretto all'elemento deformabile

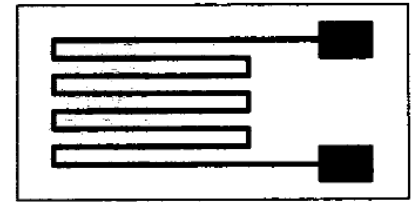




sensori di pressione

Trasduzione di una deformazione in un segnale elettrico

- Dimensioni: spessore $5\mu\text{m}$, lunghezza $\sim 1\text{mm}$, larghezza $\sim 0.1\text{mm}$
- Leghe: rame-nichel, nichel-cromo
- Strato di materiale ceramico isolante ($\sim 25\mu\text{m}$) tra lamina e elemento deformabile
- *Sensori di pressione* basati sugli estensimetri con campi di impiego da qualche bar fino a 10.000 bar
- Amplificatori integrati per portare il segnale in uscita dal ponte ($\sim \text{mV}$) a valori dell'ordine dei volt o mA
- Dispositivi molto veloci se collegati direttamente alla superficie (elevata banda)
- Ampia banda e costi ridotti (prodotti economici)



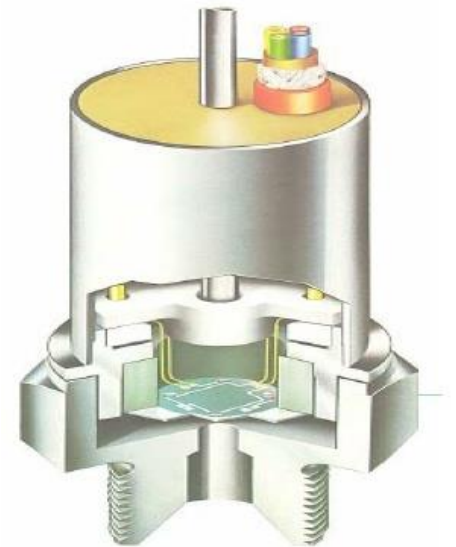


sensori di pressione

Trasduzione di una deformazione in un segnale elettrico

- **Trasduttori piezoresistivi**

- Si tratta di estensimetri evoluti: l'elemento deformabile è un cristallo di silicio (chip) sul quale sono realizzare resistenze estensimetriche mediante tecniche di diffusione
- La costruzione senza collanti e l'impiego dei cristalli di silicio consentono di ottenere prestazioni superiori in termini di rangeability e stabilità della misura



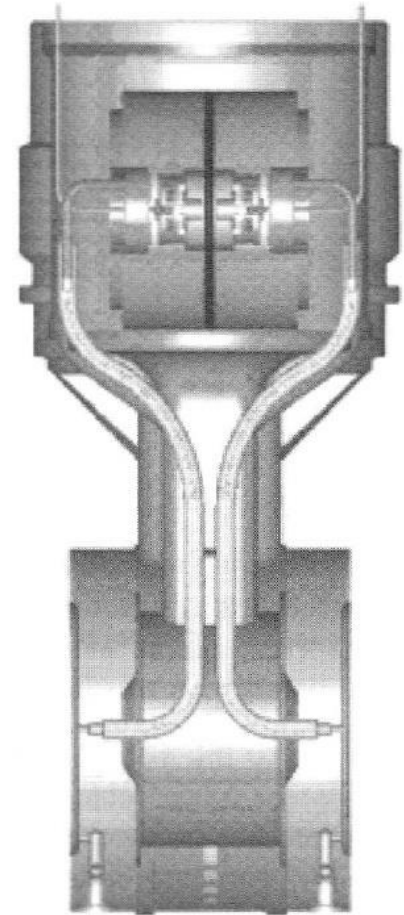


sensori di pressione

Trasduzione di una deformazione in un segnale elettrico

- **Trasduttori induttivi**

- *In figura* è illustrato un sensore induttivo di tipo differenziale
- Comprendono un diaframma metallico la cui deformazione per azione della pressione sotto misura fa variare la posizione di un nucleo di ferrite modificando così l'induttanza di una bobina





sensori di pressione

Trasduzione di una deformazione in un segnale elettrico

- La variazione dell'induttanza viene rilevata:
 - mediante un ponte di Wheatstone in alternata
 - inserendo l'induttanza in un circuito oscillatore
 - Variazione nell'induttanza → variazione frequenza di oscillazione, misurabile contando i massimi dei segnali su un intervallo di tempo prefissato



sensori di pressione

Trasduzione di una deformazione in un segnale elettrico

- **Risonatori su silicio**

- Risonatore meccanico a forma di *doppio diapason* realizzato in un monocristallo di silicio ed immerso in un campo magnetico permanente
- Oscillazioni alla frequenza di risonanza provocate mediante impulsi di corrente
- Frequenza di oscillazione dipende dalla deformazione trasversale provocata dalla pressione sotto misura
- Dalla frequenza di oscillazione si risale alla deformazione
- Elevate prestazioni (precisione, rangeability e stabilità della misura nel tempo)



sensori di portata

Introduzione

- I **sensori di portata (flussimetri)** si dividono in:
 - Sensori a strozzamento
 - Sensori a sezione variabile
 - Sensori rotanti
 - *Sensori basati sulle nuove tecnologie*
- Misurano la portata volumetrica
 - Sono detti **massici** quelli che misurano la portata massica
- Sono molto usati → ne esistono tante varietà
 - Influenzano notevolmente la qualità dei prodotti!

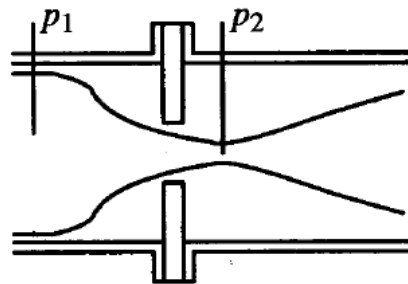
sensori di portata

Dischi forati, boccagli e venturimetri

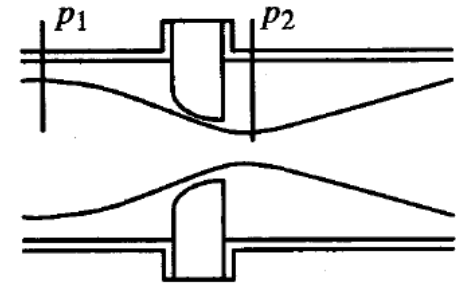


- **Dischi forati, boccagli e venturimetri**

- Provocano l'aumento della velocità del fluido in un tratto della tubazione imponendo una contrazione nella sezione di passaggio



a)

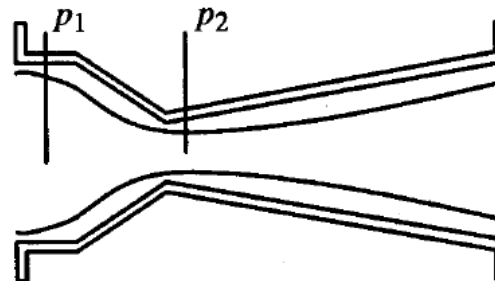


b)

a) Dischi forati

b) Boccagli

c) Venturimetri



c)



sensori di portata

Dischi forati, boccagli e venturimetri

- Misurando la caduta di pressione a monte del dispositivo e quella in corrispondenza della *vena contratta*, si può risalire alla portata utilizzando l'equazione di Bernoulli
 - Applicando l'equazione di Bernoulli ad una tubazione con un sensore a strozzamento disposto orizzontalmente:

$$\frac{v_1^2}{2} + \frac{p_1}{\rho} = \frac{v_2^2}{2} + \frac{p_2}{\rho}$$

da cui

$$v_2^2 - v_1^2 = \frac{2}{\rho}(p_1 - p_2)$$



sensori di portata

Dischi forati, boccagli e venturimetri

Considerando che la portata massica è $w = \rho A v$ si ottiene

$$w = \gamma A_2 \sqrt{\rho(p_1 - p_2)}$$

con

$$\gamma = \sqrt{\frac{2}{1 - \frac{A_2^2}{A_1^2}}}$$

- Correzione dovuta alle *perdite di carico* prodotte dalla strozzatura $\alpha v_2^2 / 2$ e dalla differenza tra la velocità media e la *distribuzione di velocità* nella sezione A_2 , corretta con un coefficiente ξ

$$\gamma = \xi \sqrt{\frac{2}{1 + \alpha - \frac{A_2^2}{A_1^2}}}$$



sensori di portata

Dischi forati, boccagli e venturimetri

- Si noti che $w = K \sqrt{p_1 - p_2}$ dove K va corretto in fase di taratura
- La sezione A_2 dipende dalla portata e andrebbe corretta
- Rangeability 3:1 e ridotta accuratezza (3-5% della misura)
- Per i *fluidi gassosi* va considerata la dipendenza della densità dalla temperatura e pressione (→ vanno misurate)

$$w \cong K \sqrt{p_1 - p_2} \sqrt{\frac{p_2}{T_2}}$$

detta correzione “**pi su t**” derivata dalla legge dei gas perfetti, con K che va corretto in fase di taratura



sensori di portata

Dischi forati, boccagli e venturimetri

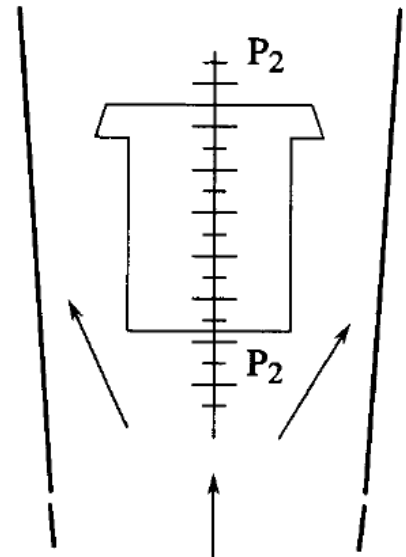
- Perdite di carico consistenti
 - Particolarmente elevate nei dischi forati
 - Meno rilevanti nei venturimetri, che però sono più costosi
- Sensori a strozzamento hanno una buona ripetibilità
 - Richiedono tarature frequenti a causa del deposito di materiale (incrostazioni) nelle vicinanze della strozzatura



sensori di portata

Sensori di portata a sezione variabile (rotametri)

- I **sensori di portata a sezione variabile** sono impiegati per misurare portate di fluidi e gas
 - Molto economici
 - Molto diffusi (industria chimica, farmaceutica e alimentare)
 - Usati spesso come indicatori locali
- Sono costituiti da un tubo di misura trasparente, conico e graduato, e da un galleggiante sagomato (*otturatore*), disposto internamente al tubo





sensori di portata

Sensori di portata a sezione variabile (rotametri)

- Vanno disposti in tratti di tubazione verticale
- Trasducono la portata in spostamenti dell'otturatore
- Vanno inseriti direttamente nella tubazione, compatibilmente con la pressione, altrimenti in un tratto di tubazione appositamente derivato
 - Per *pressioni molto elevate* si utilizzano modelli cosiddetti “**armati**” (fatti in acciaio con indicatore esterno attuato dall'otturatore per accoppiamento elettromagnetico)
- In condizioni stazionarie, la posizione del galleggiante dipende dall'equilibrio delle forze agenti sull'otturatore
 - Peso, forze delle pressioni sulle superfici superiore ed inferiore, eventuali forze dovute a molle di contrasto



sensori di portata

Sensori di portata a sezione variabile (rotametri)

- Si potrà quindi scrivere

$$A_0 p_1 = A_0 p_2 + g V_0 \rho_0$$

dove con il pedice 0 sono indicati i parametri del galleggiante (area, volume e densità)

- Applicando l'equazione di Bernoulli alle sezioni 1 e 2 si ottiene la portata alla sezione 2:

$$w = \gamma A_2 \sqrt{\rho [p_1 + \rho g h_1 - (p_2 + \rho g h_2)]}$$

dove

$$\gamma = \sqrt{\frac{2}{1 + \alpha - \frac{A_2^2}{A_{t1}^2}}}$$

A_{t1} sezione a monte
 A_2 sezione *di passaggio* a valle
 α perdite di carico



sensori di portata

Sensori di portata a sezione variabile (rotametri)

- Combinando le due relazioni precedenti:

$$w = \gamma A_2 \sqrt{\rho \frac{g V_0 (\rho_0 - \rho)}{A_0}}$$

avendo posto $h_2 - h_1 = V_0 / A_0$

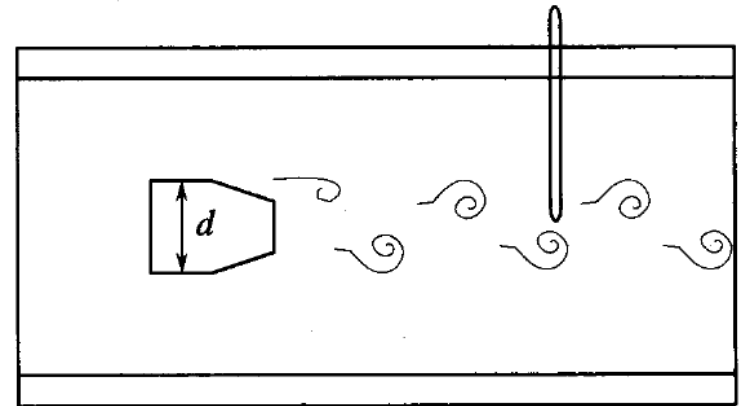
- Evidentemente variazioni della portata si riflettono sulla sezione di passaggio del fluido A_2 , essendo costanti tutti gli altri parametri, e quindi sulla posizione del galleggiante
- Se la densità è variabile (*fluidi gassosi*) andrà calcolata misurando temperatura e pressione (→ correzione “p su t”)
- Sono forniti “calibrati per la specifica applicazione”



sensori di portata

Sensori di portata a vortice (vortex)

- I **sensori a vortice (vortex)** misurano portate di gas, vapori e liquidi
 - Adatti per misure di fluidi puliti e a bassa viscosità (vapori e condense)
- Un corpo non smussato produce instabilità nel moto del fluido con vortici che si staccano in successione dal corpo estraneo (posizionato al centro del tubo)





sensori di portata

Sensori di portata a vortice (vortex)

- La frequenza di formazione dei vortici è direttamente proporzionale alla velocità del fluido: $f_v = K v$

$$w = \rho A \frac{f_v}{K}$$

- K costante per ampie variazioni (di Re) se il corpo perturbante è opportunamente sagomato
- La frequenza dei vortici è rilevata in differenti modi
 - Si utilizza un *termistore* riscaldato, la frequenza viene quindi rilevata in base alle oscillazioni della resistenza dovute al differente grado di raffreddamento provocato dai vortici
 - Si utilizza un *trasduttore piezoelettrico* per rilevare le variazioni locali di pressione provocate dai vortici



sensori di portata

Sensori di portata a vortice (vortex)

- I segnali forniti sono intrinsecamente digitali, costituite da sequenze di impulsi che possono essere facilmente contati
- Elevata accuratezza:
 - $\pm 0.75\%$ della portata per i fluidi
 - $\pm 1\%$ della portata per i gas
- Limiti di misura:
 - $0.2 \div 450$ Kg/s per l'acqua
 - $4 \div 3600$ Kg/s per l'aria
- Rangeability pari a 10:1, fino a 20:1
- Occorre che il moto sia molto turbolento ($Re > 30.000$)
- Non adatti per processi in cui le portate variano molto



sensori di portata

Sensori di portata a vortice (vortex)

- Sensore sempre pieno di fluido!
- Consigliata l'installazione di un tratto rettilineo prima del sensore, lungo 15 volte il diametro del sensore, ed uno dopo, lungo 5 volte il diametro del sensore
- Vanno compensate le variazioni di densità
- Perdite di carico importanti (↑energia per pompaggio)
- Possibili fenomeni di *cavitazione* ed *evaporazione*
- Molto diffusi nei processi, più costosi di quelli a strozzamento, ma offrono migliori prestazioni
- **Vortex massici**: acquisiscono anche l'ampiezza del segnale fornito dal sensore piezoelettrico (no stima della densità!)



sensori di portata

Sensori di portata elettromagnetici

- I **sensori di portata elettromagnetici** sono utilizzati in molti settori industriali
 - industria chimica, cartaria, farmaceutica, alimentare, impianti di potabilizzazione e trattamento reflui
- Il fluido sotto misura deve essere *elettricamente conduttivo* (conduttività $C = 5\mu\text{S/m}$)
 - Acque reflue, acidi, alcali, succhi, ecc.
- Funzionamento basato sulla legge di induzione elettromagnetico

sensori di portata

Sensori di portata elettromagnetici

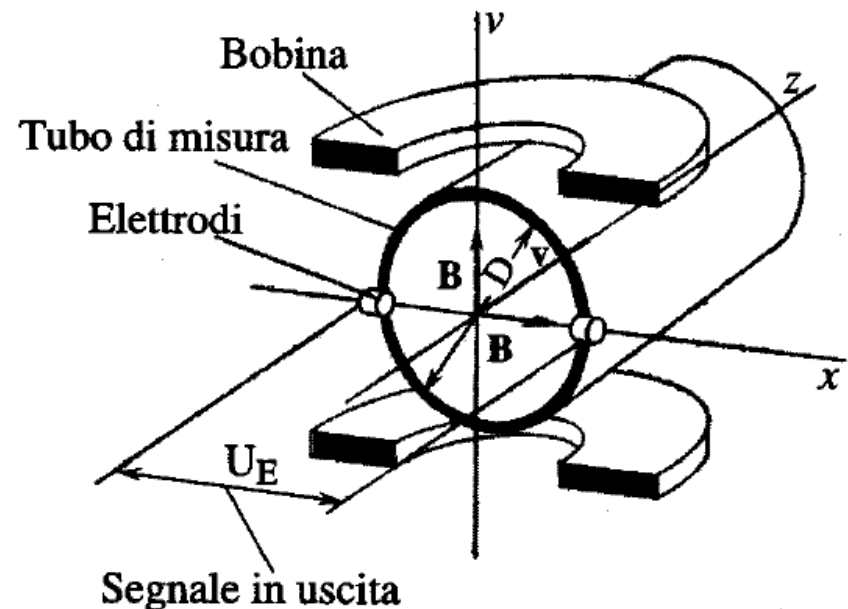
- Il fluido conduttivo viene sottoposto ad un campo magnetico trasversale: si genera una forza elettromotrice ortogonale alla direzione del campo e del moto del fluido

$$e = vdB$$

da cui si ricava

$$w = \rho A v = \rho A \frac{e}{Bd}$$

Nota: la relazione è stata ricavata per un conduttore rettilineo e andrebbe integrata sull'intera sezione del tubo





sensori di portata

Sensori di portata elettromagnetici

- La velocità del fluido nella sezione deve rimanere uniforme per avere un comportamento lineare
 - Sensore preceduto e seguito da tratti di condotta rettilinei lunghi almeno 3 e 2 volte il diametro, rispettivamente
- Molto accurati: 0.5% della portata
- Rangeability: 50:1, fino a 100:1 ma meno accurato
- Nessuna perdita di carico!
- Nessuna parte in movimento → robusti e affidabili
- Campo di misura: $0.005 \div 30000 \text{ Kg/s}$



sensori di portata

Sensori massici

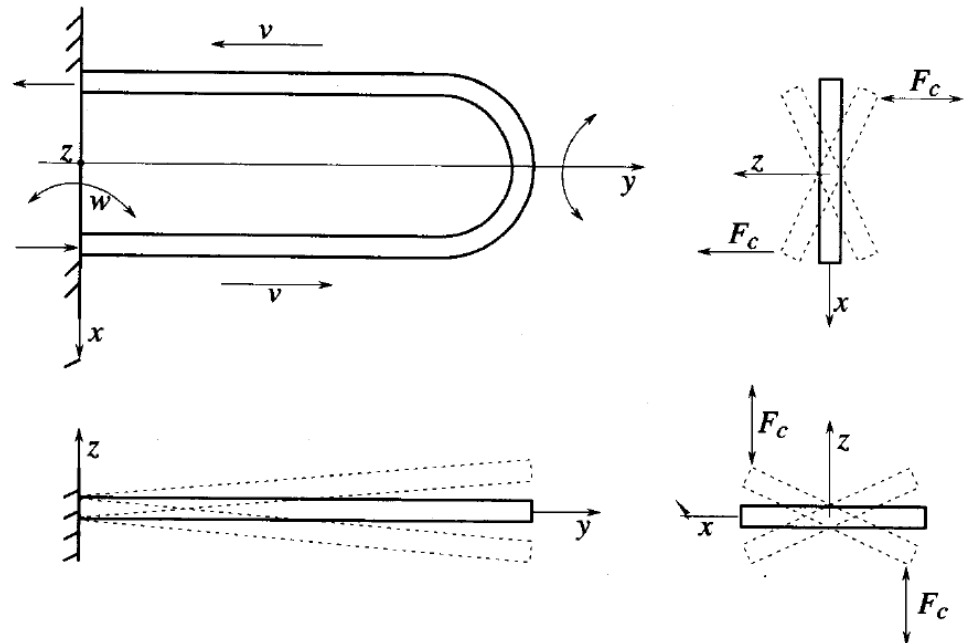
- I **sensori massici** forniscono direttamente la misura della portata massica → misura indipendente dalle variazioni di densità
 - Fluidi moderatamente densi
- Sfruttano l'accelerazione e la forza di Coriolis:
 - Su un corpo di massa m in moto con velocità $\mathbf{v}$ rispetto ad un sistema di riferimento in moto con velocità angolare ω si manifesta la forza di Coriolis

$$\mathbf{F}_c = 2m\omega \times \mathbf{v}$$

sensori di portata

Sensori massici

- Diverse realizzazioni commerciali basate su questo principio
- *Esempio*: sensore costituito da un tubo ad U
 - Tubo sollecitato nel tratto curvo con elettromagneti $\rightarrow$ si innesca una vibrazione alla frequenza di risonanza ω attorno all'asse x





sensori di portata

Sensori massici

- Fluido in moto con velocità $v \rightarrow$ si generano due forze di Coriolis nei tratti rettilinei dirette lungo z ma in direzioni opposte $\rightarrow$ tubo oscilla intorno all'asse y
- L'ampiezza delle oscillazioni è proporzionale al modulo delle forze di Coriolis (rilevabile con estensimetri)

$$F_c = 2m\omega v = 2\rho A l \omega v$$

quindi il segnale fornito dagli estensimetri è proporzionale alla portata massica

$$w = \rho A v$$

- La sensibilità è $\Delta F_c / \Delta w$ è proporzionale a ωl



sensori di portata

Sensori massici

- Molto accurati: 0.25% della portata
- Rangeability: 100:1
- Ripetibilità: 0.15% della portata
- Basse perdite di carico
- Campi di misura ristretti
- Molto costosi → da evitare se non ci sono vincoli sull'accuratezza



sensori di portata

Sensori rotanti

- I **sensori rotanti** sono costituiti da giranti, poste in rotazione dalla corrente del fluido, che assumono una velocità di rotazione proporzionale alla velocità del fluido e quindi alla sua portata volumetrica
 - Velocità di rotazione rilevata mediante emissione e conteggio su intervalli prefissati di impulsi magnetici
- Elevate perdite di carico
- Usura a causa della rotazione
- *Anemometri*: misurano velocità dei gas
 - Velocità comprese tra decimi e decine di metri al secondo



sensori di portata

Sensori rotanti

- Turbine di misura vengono usate per misurare portate volumetriche (massiche, nota la densità) di liquidi (si rileva la velocità di rotazione)
- Misuratori di correnti di fluidi in condotti aperti sono realizzati con gli stessi principi (*mulinelli a tazza*)
 - Basse velocità
 - Campo $0.1 \div 5$ m/s
 - Accuratezza 2%



sensori di livello

Introduzione

- **Misura di livello:** determinare la posizione, rispetto ad un piano di riferimento, dell'interfaccia tra due fluidi separati per azione della forza di gravità
 - Usata nei sistemi di regolazione
- *Sensori di livello*
 - Forniscono una misura che varia con continuità tra un minimo ed un massimo
- *Interruttori di livello*
 - Rilevano il superamento di una o più soglie



sensori di livello

Introduzione

- Si differenziano in base al *metodo di misura*:
 - Visuali
 - A galleggiante
 - Capacitivi
 - Conduttivi
 - Idrostatici
 - Radiometrici
 - A ultrasuoni
 - A microonde
 - A tasteggio
 - A vibrazione

sensori di livello

Sensori a principio idrostatico

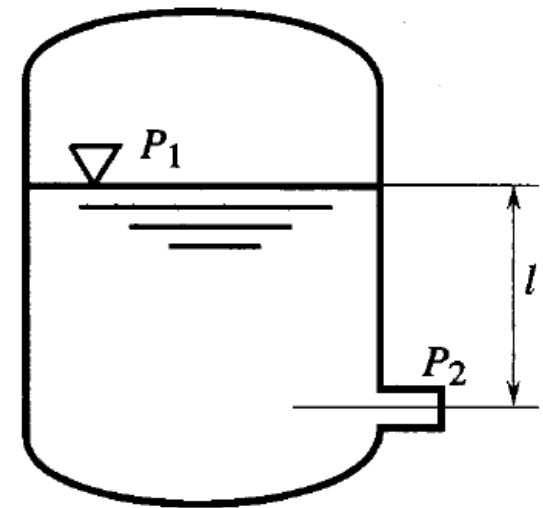


- **Sensori a principio idrostatico**

- Misurano la *massa per unità di superficie* $\rho_w l$ (densità per il livello del liquido) a partire dalle misure della pressione idrostatica sul fondo del serbatoio e a pelo libero

$$l = \frac{p_2 - p_1}{\rho_w g}$$

- Insensibili a schiume e depositi e a variazione di proprietà del fluido ad eccezione della densità





sensori di livello

Sensori a principio capacitivo

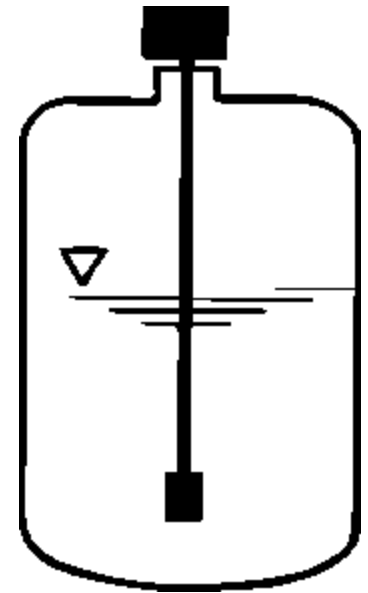
- I **sensori a principio capacitivo** misurano con continuità livelli di liquidi e solidi
 - Usati anche per la rilevazione del superamento di soglie
- Campo di misura: 0÷500 bar con temperature comprese tra -170° e 400°C
- Adatti ad installazioni in aree con pericolo di esplosione
- Versatili e caratterizzati da una grande varietà di soluzioni



sensori di livello

Sensori a principio capacitivo

- Una sonda inserita nel serbatoio forma con la sua parete metallica una capacità
 - Il valore di capacità dipende dalle superfici di sonda e parete, dalla distanza, dal tipo e natura dielettrica del fluido e dal livello
 - Il valore di capacità risulta proporzionale al livello del fluido nel serbatoio
 - Trasduzione con un ponte di Wheatstone





sensori di livello

Sensori a principio capacitivo

- Esistono *soluzioni a due elettrodi* (sonde)
 - Il livello di liquido che si interpone tra gli elettrodi ne determina la capacità
- Versioni semplificate riconoscono il superamento di uno o più livelli di riferimento
 - *Esempio*: livello basso, intermedio e alto con tre sonde di diversa lunghezza, ognuna relativa ad una soglia



sensori di livello

Sensori a ultrasuoni

- I **sensori a ultrasuoni** sono basati sulla misura del tempo impiegato da un'onda sonora per percorrere la distanza tra l'emettitore e la superficie riflettente del fluido e a tornare indietro (*tempo di volo*)
 - Il sensore emette in successione brevi impulsi ultrasonori che vengono riflessi dalla superficie del fluido
 - Il **tempo di volo** è proporzionale alla distanza percorsa e quindi al livello del fluido
 - In alcuni casi si misura lo *sfasamento* tra l'onda riflessa dalla superficie del fluido e dal fondo del serbatoio



sensori di livello

Sensori a ultrasuoni

- La **velocità del suono** dipende dalla *temperatura* del mezzo nel quale si propaga → per misure accurate è necessario misurarla e compensarne gli effetti
- Sono impiegati per misure continue di livello
- Sono costosi → uso di tipi più economici se possibili
 - Necessari quando:
 - La densità, la conducibilità e le proprietà dielettriche del fluido variano
 - Il sensore non deve entrare a contatto con il fluido (es. fluidi corrosivi)
 - Presenza di miscelatori
 - Non è possibile lo svuotamento del serbatoio per la taratura



sensori di livello

Interruttori di livello

- **Gli interruttori di livello** sono realizzati secondo diversi metodi, anche molto semplici
 - Basati sul principio capacitivo (già presentati)
 - Sistemi a vibrazione
 - Sistemi radiometrici
 - Sistemi basati sulla variazione di conduttività
- *Sistemi a vibrazione*
 - Costituiti da due aste metalliche e da una membrana metallica posti in vibrazione alla frequenza di risonanza
 - Quando la sonda entra in contatto con il fluido si ha uno smorzamento rilevabile delle vibrazioni → segnale digitale



sensori di livello

Interruttori di livello

- *Sistemi radiometrici*
 - Costituiti da un *emettitore di radioisotopi* e da un *rilevatore*, montati contrapposti all'esterno del serbatoio
 - Quando il fluido si interpone viene rilevato il raggiungimento del livello
 - Impiegati in condizioni dove altri sensori più semplici non possono essere impiegati
- *Sistemi a variazione di conduttività*
 - Con due sonde (o una più il serbatoio) si rileva la variazione di conduttività tra il fluido e il suo vapore o l'aria
 - Al raggiungimento del livello si rileva passaggio di corrente

sensori di posizione

Introduzione



- Sistemi di **misura della posizione** costituiti da:
 - Sensore di posizione che trasforma lo spostamento o l'angolo in un segnale elettrico
 - Elettronica di interpolazione del segnale
 - Sistema di conversione analogico/digitale
- *Principio fisico* di misura:
 - Fotoelettrico
 - Induttivo
 - Magnetico
 - Interferenziale

sensori di posizione

Introduzione



- Metodo di misura:
 - *Incrementale*
 - Si basa sul conteggio di impulsi a partire da un riferimento assoluto
 - *Assoluto*
 - Si basa sulla codifica diretta in termini di posizione assoluta
- I metodi assoluti hanno il vantaggio di fornire la misura direttamente all'accensione, mentre con i sistemi incrementali è necessario portare il sistema in una posizione di riferimento prestabilita

sensori di posizione

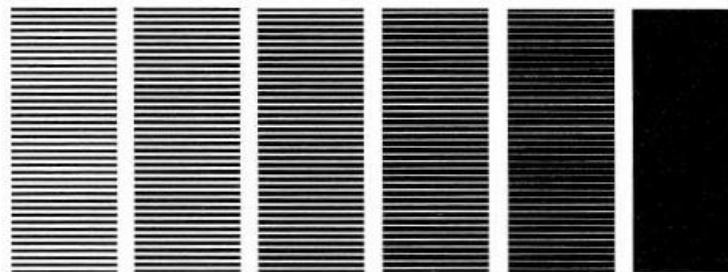
Metodo di misura incrementale: principio di misura fotoelettrico



Generazione dei segnali di scansione

- **Principio di misura fotoelettrico** (*principio di Moiré*):

Si fanno scorrere parallelamente due righe graduate di vetro con una divisione a reticolo, costituita da linee opache parallele intervallate da spazi intermedi trasparenti di eguale lunghezza → si rilevano oscillazioni periodiche di luminosità che si possono convertire, mediante sensori fotoelettrici, in corrispondenti segnali elettrici → **segnali di scansione**





sensori di posizione

Metodo di misura incrementale: principio di misura fotoelettrico

- Reticoli con divisioni molto fini ($\sim 10 \mu\text{m}$) $\rightarrow$ aumento della risoluzione
- Piccole imperfezioni delle divisioni non alterano la precisione o l'affidabilità perché la misura si ottiene per *integrazione* di un gran numero di impulsi
- Limite nell'ampiezza del periodo di divisione tale da non generare *fenomeni di diffrazione*
 - $\rightarrow$ periodo di divisione $\gg$ lunghezza d'onda della luce
- Per macchine di estrema precisione (passi $< 4 \mu\text{m}$) si utilizzano metodi di interpolazione interferenziali

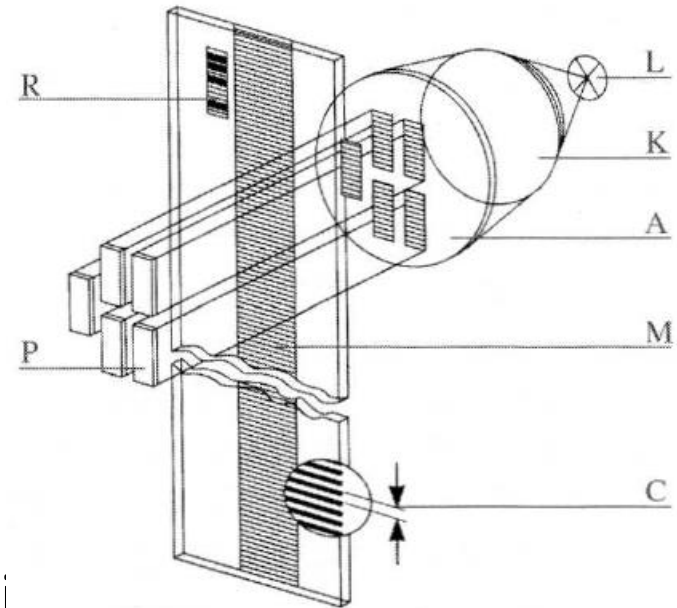
sensori di posizione

Metodo di misura incrementale: principio di misura fotoelettrico



- Metodo per trasmissione della luce

- **Disco di scansione A** a breve distanza dalla **riga graduata M**
- Fascio di luce parallelo prodotto da una **sorgente L** ed un **condensatore ottico K** colpisce, attraverso 4 *settori di scansione* del disco, la riga graduata e quindi i **fotoelementi P**

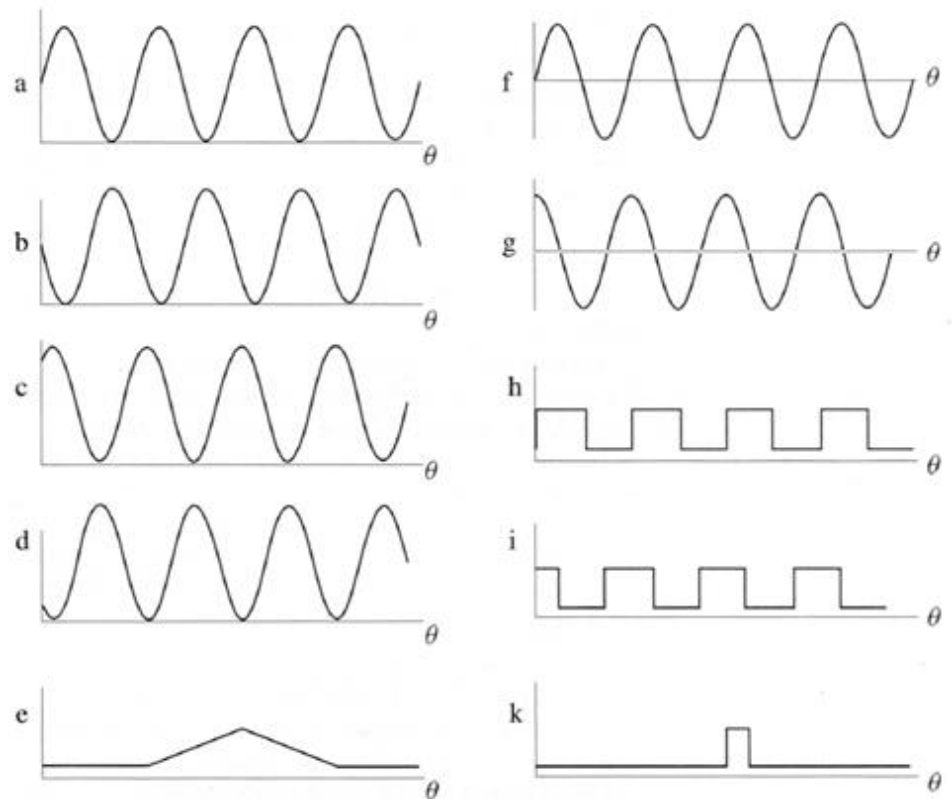


sensori di posizione

Metodo di misura incrementale: principio di misura fotoelettrico



- Il movimento della riga graduata produce nel fotoelemento un segnale di andamento sinusoidale a) con valor medio diverso da zero e una frequenza dipendente dalla velocità relativa
- Nel secondo settore di scansione la divisione è sfasata di mezzo periodo $\rightarrow$ si genera un segnale sinusoidale sfasato di 180° b)





sensori di posizione

Metodo di misura incrementale: principio di misura fotoelettrico

- Dalla differenza dei segnali a) e b) si ottiene un segnale sinusoidale V_s a valor medio nullo, di periodo L , denominato *segnale di scansione f)*
- Le altre due coppie settore/fotoelemento c) e d) generano un secondo segnale di scansione V_c sfasato di 90° g)
- Posto $\theta = 2\pi x/L$ si avrà

$$V_s = A \sin \theta$$

$$V_c = A \cos \theta$$

- I segnali di scansione sono convertiti in due segnali ad onda quadra, h) ed i) rispettivamente
- I fronti di salita (o discesa) generano impulsi di conteggio



sensori di posizione

Metodo di misura incrementale: principio di misura fotoelettrico

- Dall'ordine di successione degli impulsi provenienti dai due segnali di scansione si rileva la *direzione di movimento* → segnale di comando per un contatore bidirezionale
- La distanza tra due fronti successivi di h) ed i) corrisponde ad un passo di misura ($\frac{1}{4}$ del periodo di divisione)
- **Assolutizzazione della misura**
 - Le righe graduate presentano una o più **indici di riferimento R** (*reticolo campione*)
 - La scansione con un reticolo identico genera un singolo impulso e) trasformato in onda quadra k)
 - A tale impulso è associata una posizione assoluta



sensori di posizione

Metodo di misura incrementale: principio di misura fotoelettrico

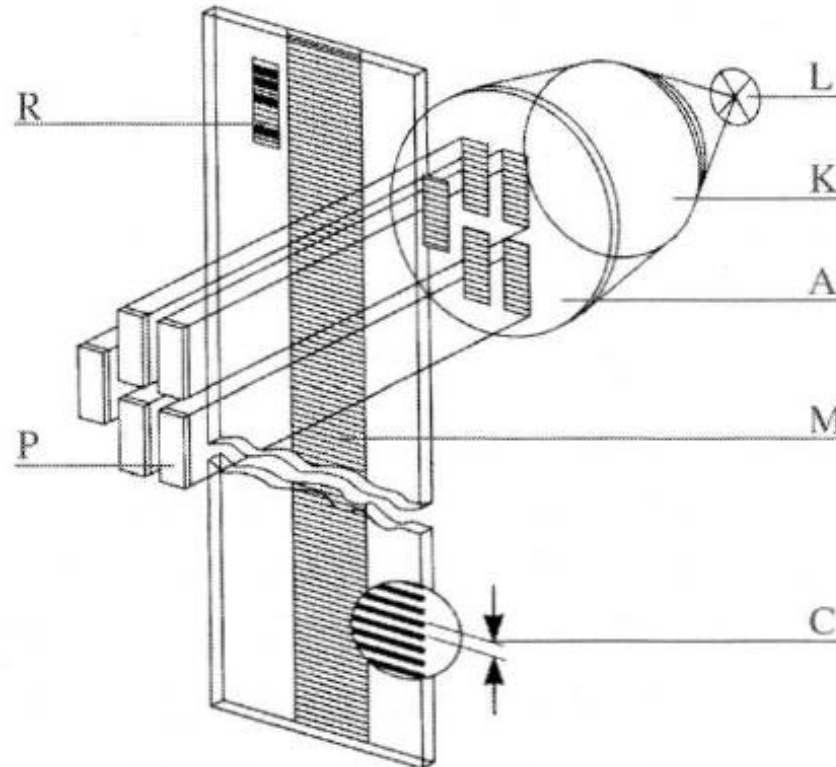
- Per ridurre gli spostamenti alla ricerca dell'indice di riferimento si utilizzano più indici a distanza variabile l'uno dall'altro
 - *Ad esempio*, si può variare di un periodo di divisione la distanza fra due indici successivi → il superamento di due indici successivi e la misura della relativa distanza consentono di risalire alla posizione assoluta



sensori di posizione

Metodo di misura incrementale: principio di misura fotoelettrico

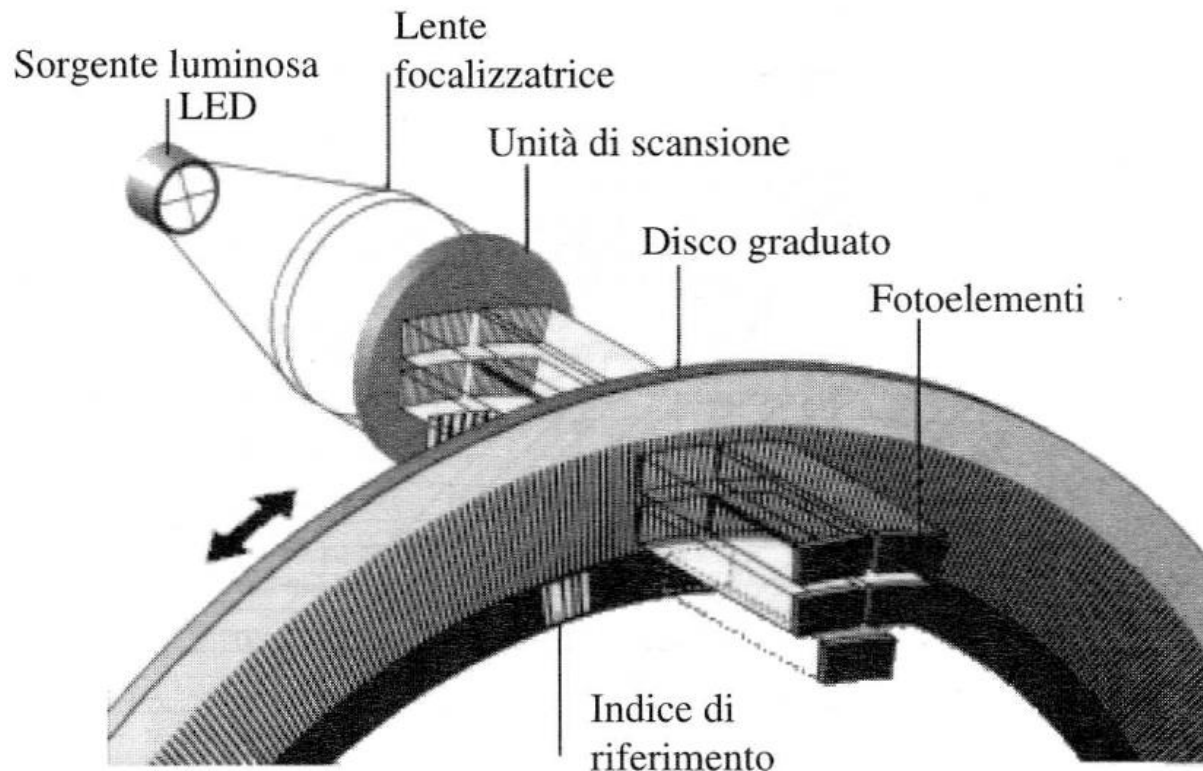
- Sistema di misura lineare
 - Riga ottica graduata in *moto traslatorio*



sensori di posizione

Metodo di misura incrementale: principio di misura fotoelettrico

- Sistema di misura angolare
 - Corona ottica circolare graduata in *moto rotatorio*





sensori di posizione

Metodo di misura incrementale: principio di misura fotoelettrico

Interpolazione dei segnali di scansione

- La risoluzione che si ottiene con la conversione diretta dei segnali di scansione in onda quadra è spesso insufficiente
- La frequenza dei segnali di interpolazione viene moltiplicata mediante un'**elettronica di interpolazione**

- **Metodo delle fasi ausiliarie**

- Sommando i segnali di scansione V_s e V_c con dei circuiti sommatori si ottengono n segnali (con n pari)

$$V_{s,k} = V_s \cos(k\pi / n) + V_c \sin(k\pi / n)$$

$$= V_s \sin(\theta + k\pi / n)$$

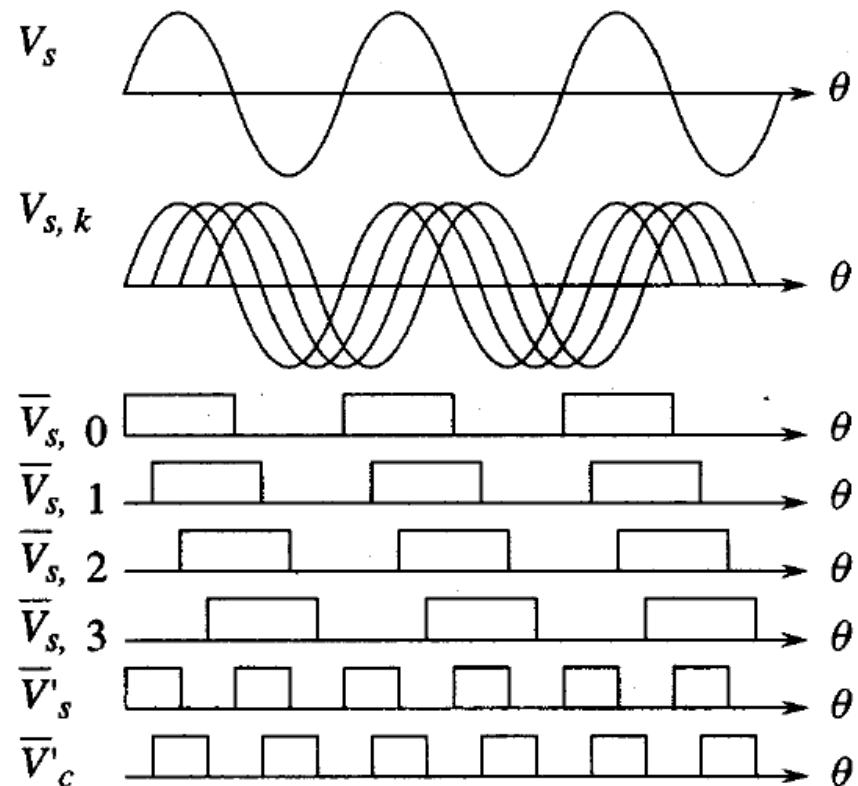
$$k = 0, \dots, n - 1$$

sensori di posizione

Metodo di misura incrementale: principio di misura fotoelettrico



- I segnali sono sfasati di $k\pi/n$ rispetto al segnale V_s
- I segnali sono quindi trasformati tramite comparatori in segnali ad onda quadra
- Con semplici circuiti logici si ottengono 2 treni di impulsi con frequenza $n/2$ volte quella di V_s sfasati di un quarto di periodo





sensori di posizione

Metodo di misura incrementale: principio di misura fotoelettrico

- La distanza tra due fronti successivi corrisponde ad un passo di misura
- Si possono ottenere riduzioni fino a $1/40$ ($n=20$) o a $1/100$ ($n=50$) del periodo di divisione
- **Metodi di interpolazione digitali**
 - Basati sull'utilizzo di microprocessori
 - Condizionamento dei segnali di scansione + conversione analogico/digitale
 - Si otterranno valori campionati dell'angolo

$$\theta^* = \arctan \frac{V_s^*}{V_c^*}$$

sensori di posizione

Metodo di misura incrementale: principio di misura induttivo

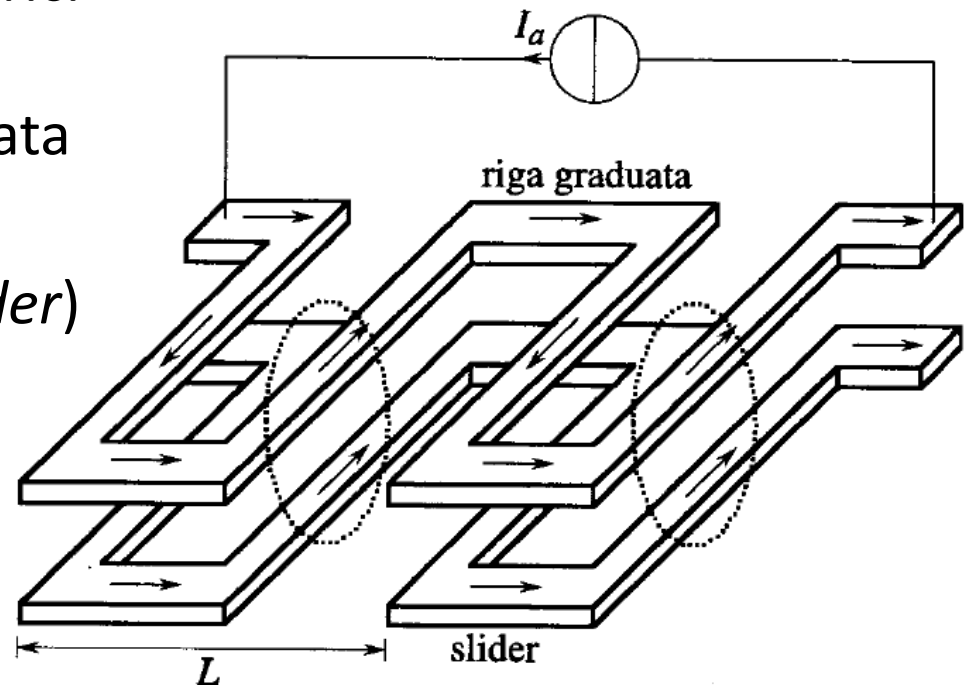


Generazione dei segnali di scansione

- *L'Inductosyn* utilizza il metodo induttivo

- Sistema di piste conduttrici a forma di greca che formano una riga graduata di lunghezza L
- Piastra di scansione (*slider*)
- La riga graduata è alimentata con una corrente alternata

$$I_a = I \sin \omega t$$





sensori di posizione

Metodo di misura incrementale: principio di misura induttivo

- Quando le piste sono sovrapposte perfettamente o sfasate di mezzo periodo, ai capi dello slider si rileva una tensione indotta di ampiezza massima, ma segno discorde nei due casi
- Quando le piste sono sfasate di un quarto di periodo la tensione indotta è nulla
- Sullo slider viene aggiunta un secondo sistema di piste sfasato rispetto al primo di un quarto di periodo
- Le mutue induttanze, posto $\theta = 2\pi x/L$, tra le piste saranno

$$M_s = M \sin \theta$$

$$M_c = M \cos \theta$$



sensori di posizione

Metodo di misura incrementale: principio di misura induttivo

- Le tensioni indotte ai capi dei circuiti di slider saranno

$$\tilde{V}_s = \frac{d(M_s I_a)}{dt}$$

$$\tilde{V}_c = \frac{d(M_c I_a)}{dt}$$

da cui, ponendo $A = MI\omega$

$$\tilde{V}_s = A \sin \theta \cos \omega t + (\dot{\theta} / \omega) A \cos \theta \sin \omega t$$

$$\tilde{V}_c = A \cos \theta \cos \omega t - (\dot{\theta} / \omega) A \sin \theta \sin \omega t$$

- Mediante un sistema di interpolazione si ricava l'angolo θ e quindi lo spostamento relativo x all'interno di un periodo



sensori di posizione

Metodo di misura incrementale: principio di misura induttivo

Interpolazione dei segnali di scansione

- **Metodo ad inseguimento**

- I secondi termini delle espressioni dei segnali di scansione sono proporzionali alla velocità ed inversamente proporzionali alla pulsazione della corrente di eccitazione
- Trascurando questi termini per $\omega = 2\pi f$ con $f = 5 \div 20$ kHz

$$\tilde{V}_s \approx A \sin \theta \cos \omega t$$

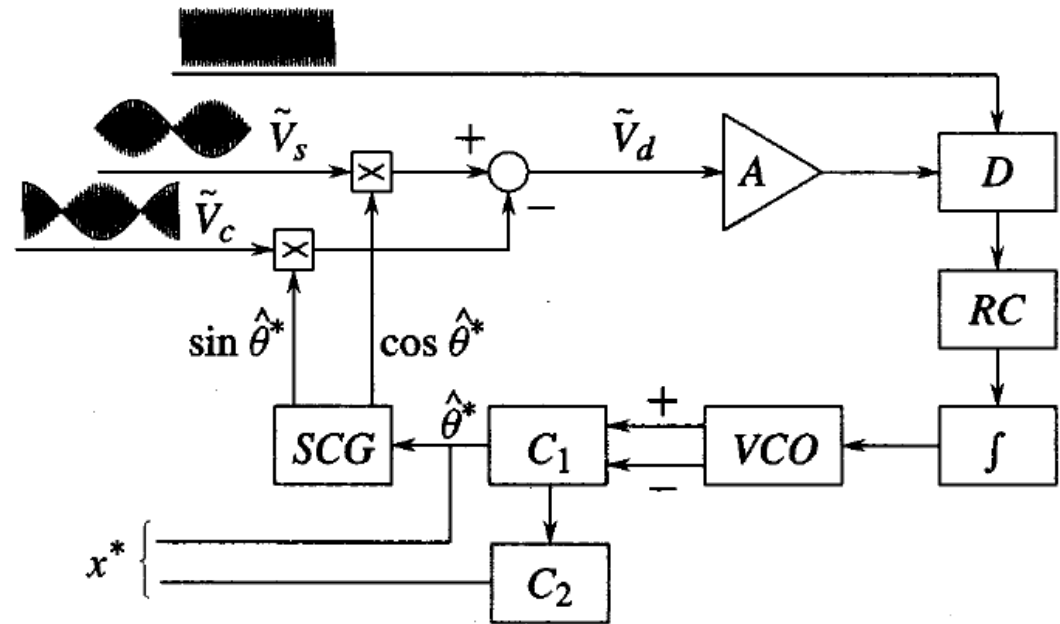
$$\tilde{V}_c \approx A \cos \theta \cos \omega t$$

sensori di posizione

Metodo di misura incrementale: principio di misura induttivo



- Con l'uso di un **generatore sin-cos** (SCG) e di **circuiti moltiplicatori**, dai segnali di scansione si genera il segnale differenza



$$\begin{aligned}\tilde{V}_d &= \tilde{V}_s \cos \hat{\theta}^* - \tilde{V}_c \sin \hat{\theta}^* \\ &= A \left(\sin \theta \cos \hat{\theta}^* - \cos \theta \sin \hat{\theta}^* \right) \cos \omega t \\ &= A \sin(\theta - \hat{\theta}^*) \cos \omega t\end{aligned}$$

** codifica digitale*



sensori di posizione

Metodo di misura incrementale: principio di misura induttivo

- Il segnale differenza viene amplificato (**amplificatore A**) e demodulato (**demodulatore sincrono D**)

$$V_e = \bar{A} \sin(\theta - \hat{\theta}^*)$$

- Il segnale viene quindi sottoposto ad una **rete di compensazione RC** (tipicamente una rete correttrice anticipatrice), integrato ed inviato ad un **convertitore tensione-frequenza VCO** (*Voltage Controlled Oscillator*)
- Il *treno di impulsi* in uscita avrà una frequenza proporzionale al segnale di errore
- Questo treno di impulsi viene contato da un **contatore bidirezionale** C_1 che produrrà il segnale desiderato



sensori di posizione

Metodo di misura incrementale: principio di misura induttivo

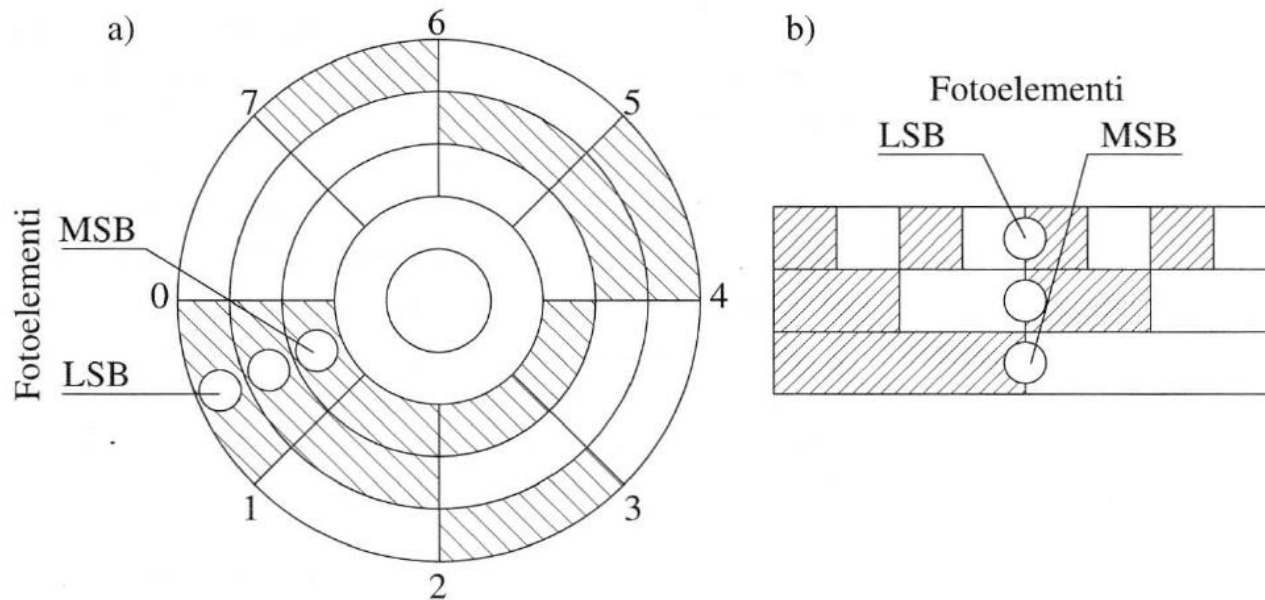
- La presenza di due integratori nell'anello, di cui uno è il contatore, assicura un *errore di inseguimento* della misura nullo per velocità di rotazione costanti
- La rete di compensazione avrà caratteristiche anticipatrici per assicurare la stabilità dell'anello
- L'*overflow* positivo o negativo del contatore C_1 attiva un secondo contatore C_2 che calcola il numero di periodi
- La misura complessiva dello spostamento sarà determinata dall'unione delle due uscite dei contatori
- **Metodo del picco** (poco accurato ma semplice)
 - Arcotangente dei picchi dei segnali $A\sin\theta$ e $A\cos\theta$

sensori di posizione

Metodo di misura assoluto: principio di misura fotoelettrico



- *Esempio: encoder assoluto a 3 bit, angolare a) e lineare b)*





sensori di posizione

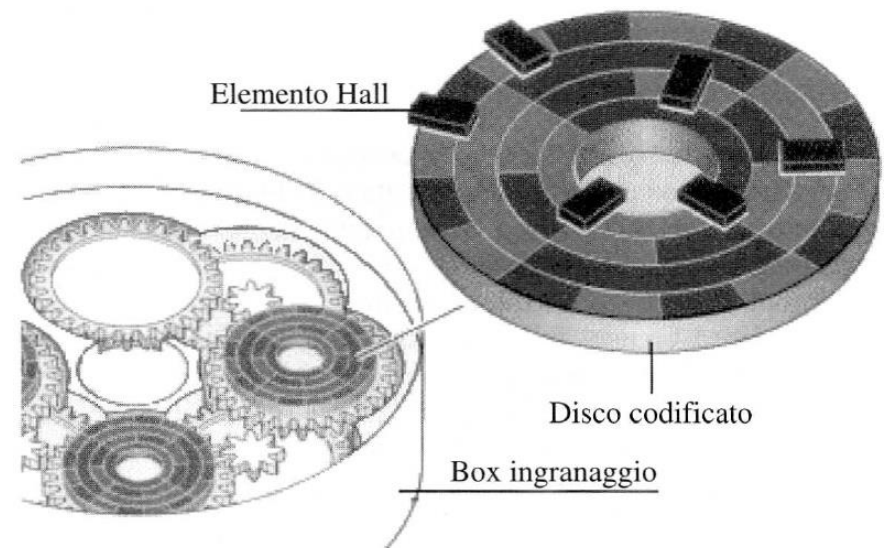
Metodo di misura assoluto: principio di misura fotoelettrico

- Su un disco viene realizzata una disposizione codificata di aree trasparenti ed opache, disposte su corone circolari
- In corrispondenza di ogni corona viene disposto un fotoelemento
- Le uscite dei fotoelementi rappresentano la codifica binaria della posizione assoluta (nell'ambito di un giro)
- N bit $\rightarrow N$ corone circolari $\rightarrow 2^N$ spazature nella corona circolare del LSB e 2 in quella del MSB (più interna)
- L'accuratezza e la risoluzione dipendono dal numero di corone circolari e dall'accuratezza delle tracce, che dipende dal diametro del disco

sensori di posizione

Metodo di misura assoluto: principio di misura fotoelettrico

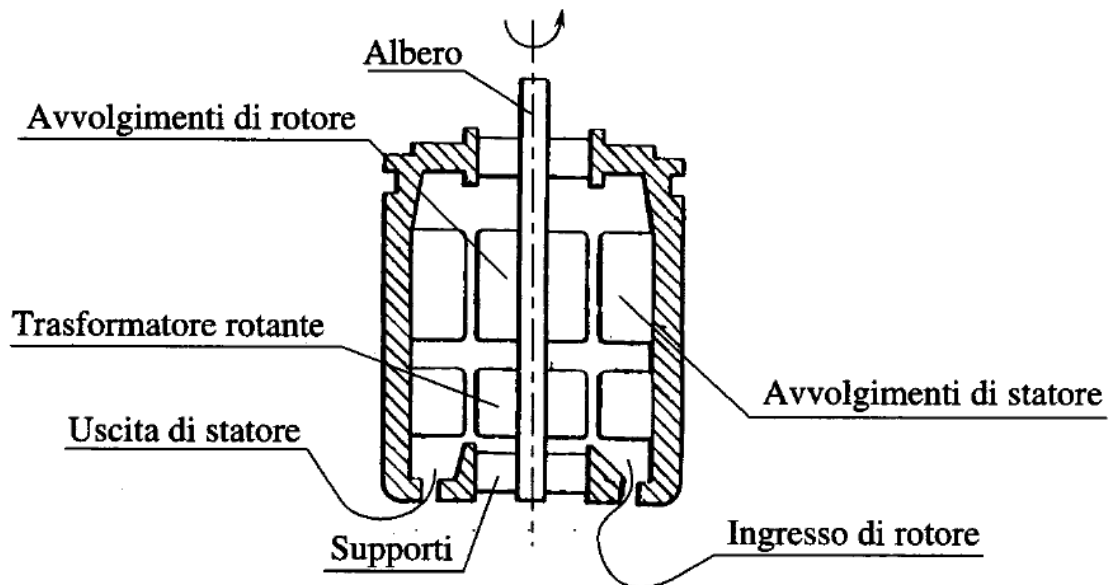
- Con un codice binario puro si avrebbero *variazioni simultanee* di più segnali → a causa di imprecisioni si avrebbero disallineamenti tra i fronti dei segnali → informazioni errate
- Con l'impiego del *codice Gray* si evitano le variazioni simultanee
- Per determinare anche il numero di giri si utilizzano supporti di misura multi-giro ad ingranaggi



sensori di posizione

Metodo di misura assoluto: principio di misura induttivo

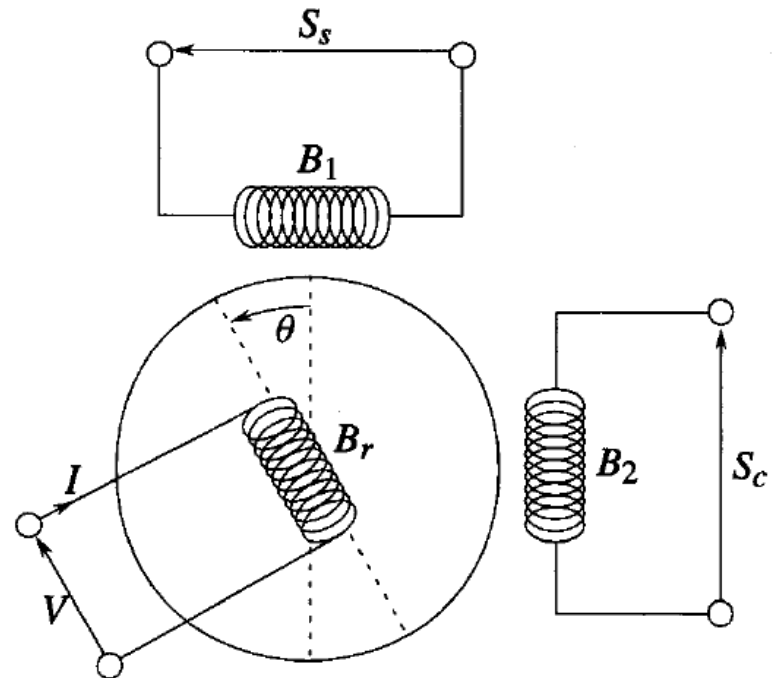
- Il *Resolver* utilizza il metodo induttivo
 - Il suo funzionamento si basa sull'accoppiamento elettromagnetico tra circuiti elettrici in modo da consentire la trasmissione continua della posizione angolare senza limiti di corsa



sensori di posizione

Metodo di misura assoluto: principio di misura induttivo

- Sul rotore è disposto un **avvolgimento di eccitazione** (B_r)
- Sullo statore sono disposti **due avvolgimenti in quadratura spaziale** l'uno con l'altro (B_1 e B_2)
- L'avvolgimento rotorico viene eccitato con una tensione sinusoidale alternata generata con l'impiego di *spazzole* o con un *trasformatore rotante*





sensori di posizione

Metodo di misura assoluto: principio di misura induttivo

- Il **trasformatore rotante** è costituito da due bobine concentriche rispetto all'asse di rotazione, una sul rotore e l'altra sullo statore
- La mutua induttanza degli avvolgimenti rotorico e statorici ha una dipendenza sinusoidale e cosinusoidale dall'angolo θ simile a quella già presentata in precedenza
- Si utilizza un **metodo di interpolazione ad inseguimento** simile a quello già presentato, che per il resolver prende il nome di *Tracking Resolver-to-Digital Converter* (TRDC)



sensori di posizione

Metodo di misura assoluto: principio di misura induttivo

- La **banda** dell'anello del TRDC può essere variata con apposite resistenze e va scelta:
 - Non troppo bassa, in modo da non introdurre sfasamenti rilevanti nell'anello di controllo di posizione
 - Non troppo alta, per non amplificare eccessivamente il rumore a frequenza superiori a quella di Nyquist → **aliasing durante il successivo campionamento**
- La banda assume valori pari a *diverse centinaia* di Hz
 - Per applicazioni in macchine utensili non sono sufficienti
- Se la *banda non è sufficiente* si campionano i segnali di scansione ad intervalli di alcune decine di μs e si ricava l'angolo con la *funzione arcotangente*



sensori di accelerazione

Accelerometri a massa sismica

- **Accelerometri a massa sismica**
 - Il contenitore è fissato al corpo sotto misura
 - Una massa (detta *sismica*) è collegata al contenitore mediante una molla
 - La posizione della massa è proporzionale all'accelerazione del contenitore *in una opportuna banda di frequenze*
 - La misura di accelerazione si ricava dalla misura della posizione relativa della massa rispetto al contenitore
 - L'accelerometro e l'elettronica di condizionamento sono oggetti di piccole dimensioni



sensori di accelerazione

Accelerometri a massa sismica MEMS

- **Accelerometri MEMS**

- Gli accelerometri MEMS (*Micro-Electro-Mechanical-System*) sono interamente realizzati in scala micrometrica in cristalli di silicio
- Il dispositivo di misura è di tipo capacitivo
 - Una o più capacità il cui valori dipendono dalla posizione della massa sismica
- Accelerometri MEMS ad 1, 2 o 3 assi hanno le dimensioni e l'aspetto di un piccolo circuito integrato (chip)
- Uscita analogica con segnali di alto livello o digitali su porta seriale ad alta velocità



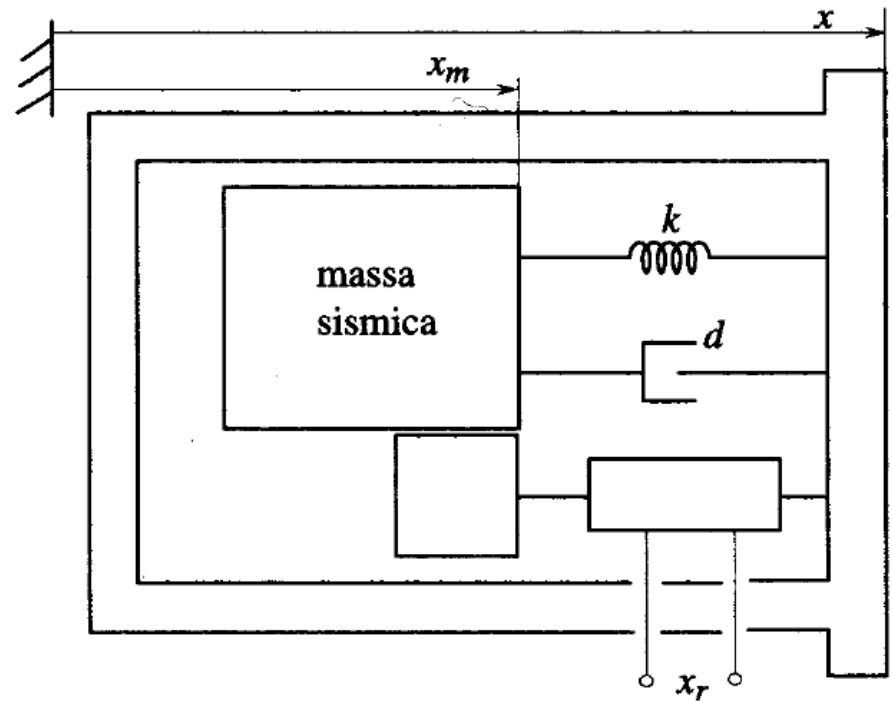
sensori di accelerazione

Accelerometri a massa sismica: principio di funzionamento

- Principio di funzionamento
 - La legge di Newton lega l'accelerazione del contenitore alla posizione relativa della massa

$$m\ddot{x}_m = f_s + f_d$$

dove m è la massa e x_m è la posizione assoluta della massa sismica rispetto ad un riferimento esterno fisso





sensori di accelerazione

Accelerometri a massa sismica: principio di funzionamento

- Il legame forza-deformazione sulla molla viene espresso con la legge lineare di Hooke

$$f_s = k(x - x_m)$$

$$f_d = d(\dot{x} - \dot{x}_m)$$

dove f_s e f_d sono le forze che esprimono l'azione della molla e dello smorzatore viscoso sulla massa, k la costante elastica della molla, d il coefficiente di attrito viscoso

- Posto $x_r = x - x_m$, si ottiene (x posizione assoluta conten.)

$$m\ddot{x}_m = kx_r + d\dot{x}_r$$

aggiungendo e sottraendo $m\ddot{x}$

$$m\ddot{x}_r + d\dot{x}_r + kx_r = m\ddot{x}$$



sensori di accelerazione

Accelerometri a massa sismica: principio di funzionamento

- Il legame tra l'accelerazione oggetto della misura e la posizione relativa della massa sismica è del secondo ordine

$$\frac{X_r}{\ddot{X}} = \frac{1}{s^2 + \frac{d}{m}s + \frac{k}{m}}$$

caratterizzata da una *pulsazione naturale*

$$\omega_n = \sqrt{k/m}$$

e da uno *smorzamento* (usualmente piuttosto piccolo)

$$\xi = d / 2\sqrt{1/(km)}$$

- Nella banda di frequenze inferiori a $0.5\omega_n$ la posizione relativa è una buona approssimazione dell'accelerazione



condizionamento e conversione dei segnali

Introduzione

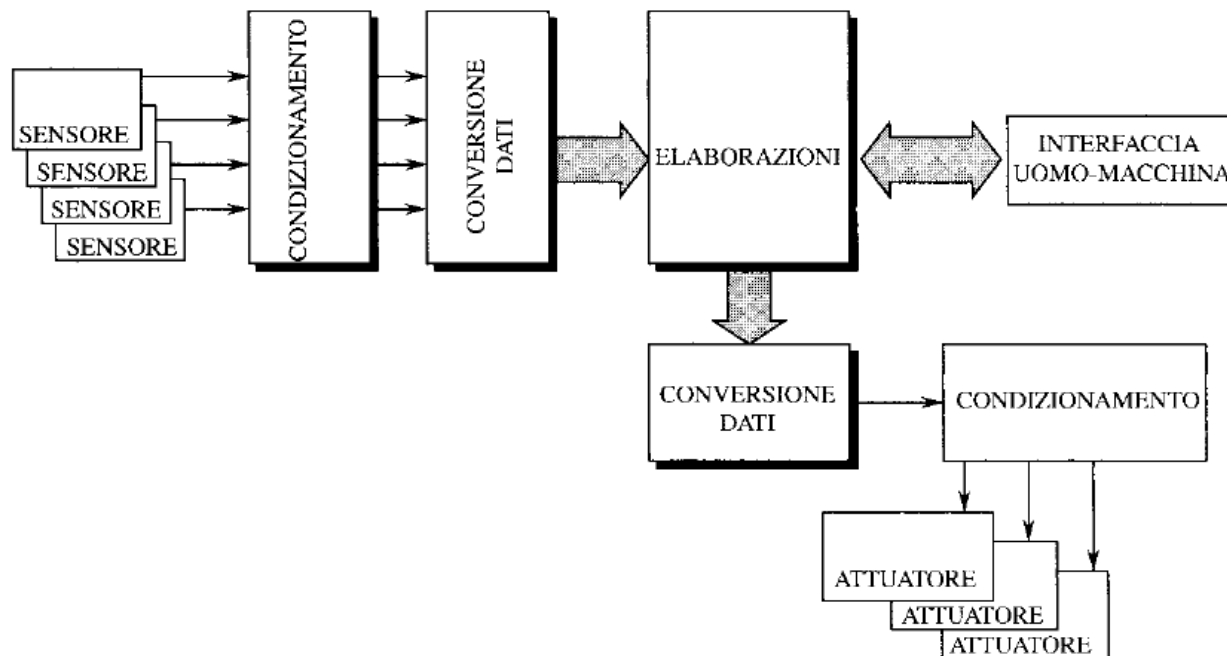
- I segnali forniti dagli elementi sensibili di misura richiedono specifici trattamenti (**condizionamento**) prima del campionamento e della conversione in valori numerici
- Il condizionamento è realizzato con *circuiti analogici*
- Segnali provenienti dai sensori sono di tipo:
 - **Analogico**: rappresentano grandezze fisiche che variano con continuità nel tempo
 - **Digitale**: rappresentano lo stato di componenti, superamento di soglie, raggiungimento di fine corsa, ecc.



condizionamento e conversione dei segnali

Introduzione

- Anche i *segnali di uscita* dai dispositivi di controllo sono classificati in analogici e digitali
- Schema di un sistema di controllo digitale:





condizionamento e conversione dei segnali

Introduzione

- Bisogna separare il segnale dal **rumore elettrico**
 - Dovuto in buona parte ai collegamenti tra l'elemento sensibile e i circuiti di condizionamento
- *Sorgenti di rumore:*
 - Interferenze dovute a circuiti adiacenti
 - Radiazioni elettromagnetiche di origine remota
- I *rimedi* riguardano:
 - Modalità di realizzazioni dei collegamenti
 - Caratteristiche elettriche dei segnali trasmessi
 - Cavi di trasmissione



ingressi analogici

Generalità

- **Campionamento:** conversione di un segnale in una sequenza numerica che rappresentano i valori assunti dal segnale in un insieme discreto di istanti di tempo
 - **Periodo di campionamento:** intervallo di tempo tra due campioni successivi
- **Teorema del campionamento (o di Shannon):** un segnale continuo a banda limitata ω_M è ricostruibile in modo univoco dai suoi valori campionati se la pulsazione di campionamento ω_S è maggiore di $2\omega_M$



ingressi analogici

Generalità

- **Formula di interpolazione di Shannon:** consente di ricostruire il segnale continuo a partire dalla sequenza dei suoi campioni
- **Pulsazione di Nyquist:** pari alla metà della pulsazione di campionamento $\omega_N = \omega_S / 2 \rightarrow f_N = \omega_N / (2\pi)$
- Disturbi agenti sul processo e sulla strumentazione di misura $\rightarrow$ segnale a *banda illimitata*
 - Disturbi alla *frequenza di rete* e suoi multipli associati:
 - a **emissioni elettromagnetiche** di utilizzatori elettrici
 - ad **anelli di massa**



ingressi analogici

Generalità

- **Aliasing:** il campionamento può causare la trasposizione in bassa frequenza (nella banda di controllo) del rumore in alta frequenza ($> f_N$)
- **Prefiltraggio:** necessario per eliminare il rumore in alta frequenza prima di effettuare il campionamento
- Forme di condizionamento:
 - *Filtraggio:* passa-basso, passa-banda e passa-alto
 - *Messa in scala:* Amplificazione ed attenuazione
 - *Adattamento di impedenza*
 - *Isolamento galvanico*



ingressi analogici

Generalità

- Condizionamento *attivo*
 - Amplificazione e isolamento galvanico
- Condizionamento *passivo*
 - Attenuazione, soppressione dei disturbi, conversione corrente-tensione e filtraggio
- **Conversione corrente-tensione** necessaria quando i segnali di ingresso giungono con l'*anello di corrente* (4÷20mA) usato per trasmissioni a distanza
- Schede di condizionamento vs. condizionamento sui trasmettitori (oggi prevalente)



ingressi analogici

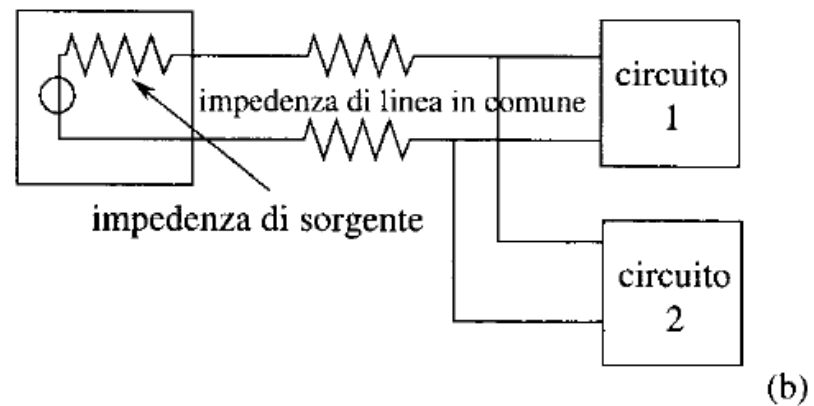
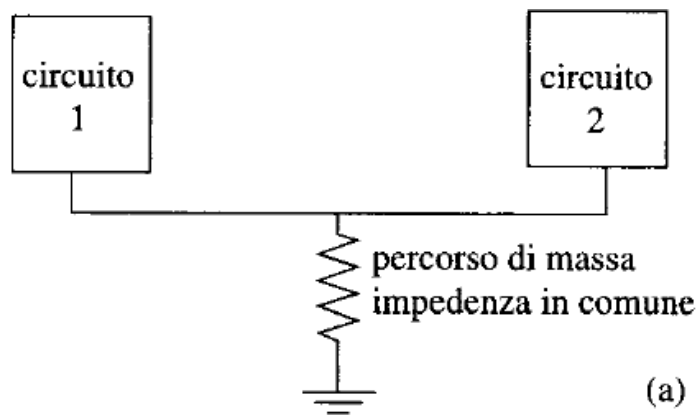
Accoppiamenti e interferenze tra circuiti e segnali

- **Accoppiamento di due circuiti:** generazione di segnali elettrici indesiderati in un circuito da parte dell'altro
 - Accoppiamento *condotto*
 - Dovuto a comunanza di percorsi o tratti di percorsi di massa e di linee di collegamento
 - Accoppiamento *capacitivo*
 - Veicolato dalle capacità parassite tra i cavi dei circuiti
 - Accoppiamento *induttivo*
 - Originato da una corrente elettrica che fluisce in un circuito e dal flusso magnetico generato che si concatena con l'altro circuito

ingressi analogici

Accoppiamento condotto

- **Accoppiamento condotto:** si manifesta quando correnti di due circuiti fluiscono attraverso una impedenza comune



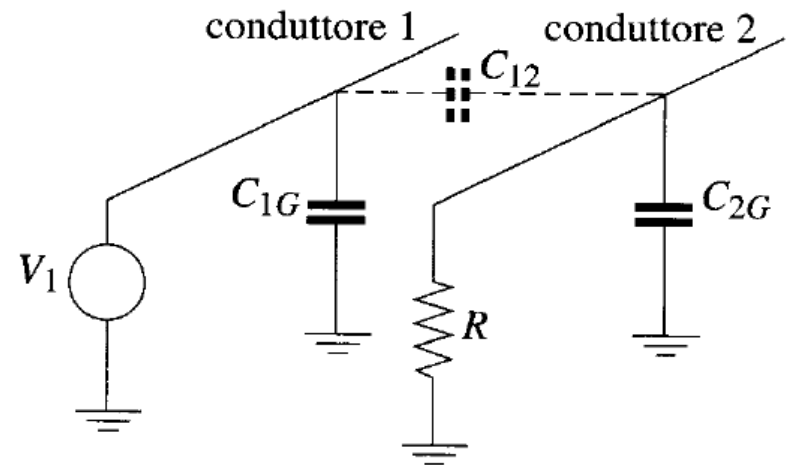
ingressi analogici

Accoppiamento capacitivo

- **Accoppiamento capacitivo:** veicolata dalle capacità parassite tra parti prossime di circuiti distinti e massa

- *Esempio:*

- C_{12} : capacità parassita tra i conduttori 1 e 2
- C_{1G} e C_{2G} : capacità parassite tra i conduttori 1 e 2 e massa



$$V_2(s) = \frac{R}{Z_{12}(s) + R} V_1 \cong \frac{R}{Z_{12}(s)} V_1 = sC_{12}RV_1$$

in ipotesi che $Z_{12} = 1/(sC_{12}) \gg R$

V_1 tensione del conduttore 1

R resistenza verso massa del conduttore 2



ingressi analogici

Accoppiamento capacitivo

- Il rumore in tensione generato per accoppiamento capacitivo è proporzionale alla frequenza e all'ampiezza del segnale sorgente del disturbo, all'impedenza verso massa del conduttore ricevitore e alla capacità parassita
- Le capacità parassite verso massa determinano la *saturation* con la frequenza del segnale sorgente

$$\left| \frac{V_2}{V_1} \right| \cong \frac{C_{2G}}{C_{12} + C_{2G}}$$

nell'ipotesi che $R \gg 1/(\omega(C_{12} + C_{2G}))$

ingressi analogici

Accoppiamento capacitivo



- Se il conduttore 2 fosse un circuito di acquisizione allora la sua l'impedenza di ingresso R non dovrà essere eccessivamente alta per ridurre gli effetti dei disturbi
- Le capacità parassite si riducono fortemente (in modo non lineare) con la distanza
- Per *combattere gli effetti* delle capacità parassite:
 - Si devono **allontanare i circuiti** e/o
 - **Schermare il circuito da proteggere** (circondare i conduttori con una sottile lamina metallica)



ingressi analogici

Accoppiamento induttivo o elettromagnetico

- Un circuito percorso da una corrente I_1 produce un campo *magnetico di induzione* proporzionale alla corrente
- Ad un circuito vicino si concatena un flusso magnetico $\Phi_{12} = M_{12}I_1$ dove M_{12} è la mutua induttanza tra i due circuiti
- Se il flusso concatenato è variabile nel tempo sul circuito si induce una forza elettromotrice indesiderata (rumore) pari a $V_d = -M_{12} \, dI_1 / dt$

ingressi analogici

Accoppiamento induttivo o elettromagnetico



- Se la corrente è sinusoidale

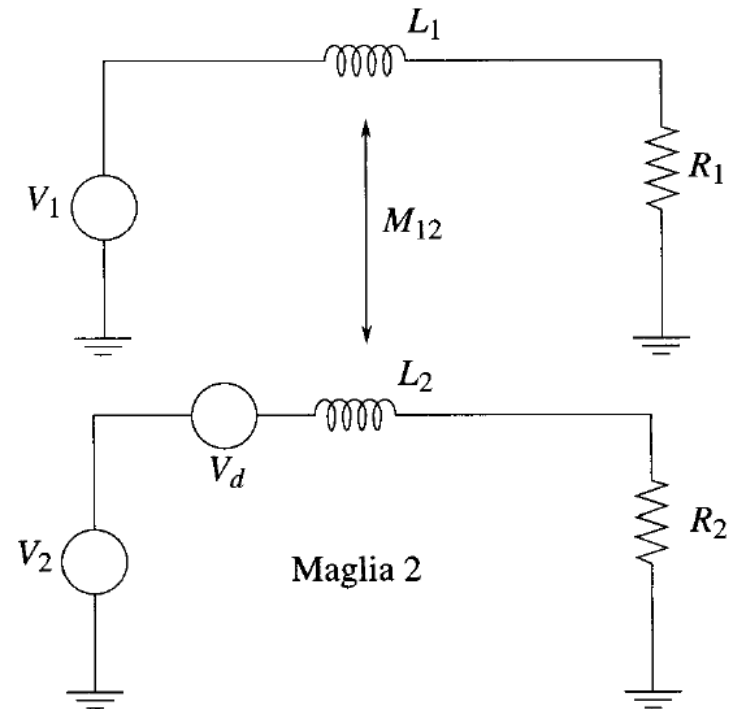
$$I_1 = A_1 \sin \omega t$$

la forza elettromotrice risulta

$$V_d = -M_{12}A_1\omega \cos \omega t$$

proporzionale alla frequenza

- Per minimizzare gli effetti dell'accoppiamento bisogna:
 - **Allontanare i circuiti**
 - Disporre i circuiti in modo da **ridurre l'area della maglia** costituita dal secondo circuito perpendicolare alle linee del campo generato dal primo





ingressi analogici

Accoppiamento induttivo o elettromagnetico

- Sorgenti di campi magnetici rilevanti e variabili a frequenze elevate sono gli *alimentatori di potenza*
 - Modulazioni di tensioni switching o PWM a frequenze di decine di kHz
- Trasmissioni a distanza → **cavi intrecciati** (*twistati*)
 - Campo netto in ogni direzione è nullo per ogni periodo di treccia (2 incroci e 2 anse) se percorsi dalla stessa corrente
- La schermatura non ha effetto sull'accoppiamento elettromagnetico
 - La lamina metallica (non ferromagnetica) non altera campi

ingressi analogici

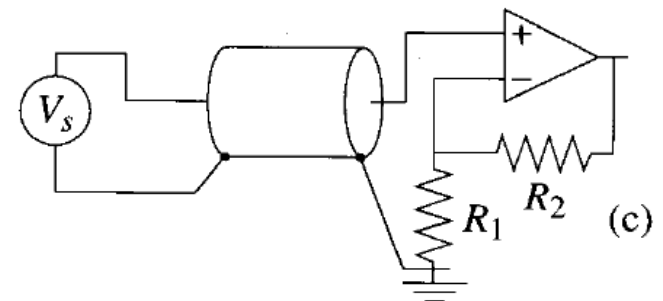
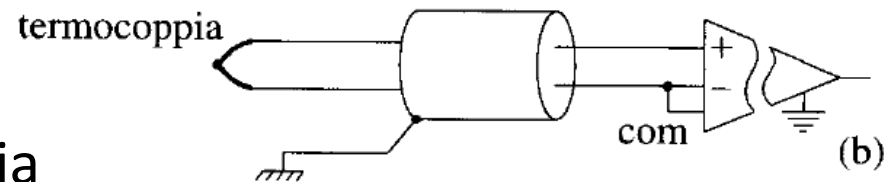
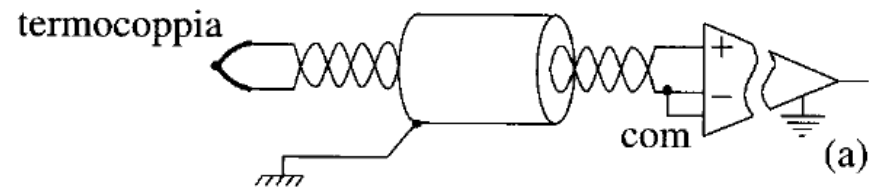
Cavi di collegamento

- Cavi usati nelle applicazioni di controllo sono:

- Cavi *twistati*
- Cavi *schermati*
- Cavi *twistati e schermati*

- *Esempio a)*

- Cavo twistato per collegare una termocoppia ad un amplificatore di isolamento, con schermo a massa ad una estremità



ingressi analogici

Cavi di collegamento



- *Esempio b)*
 - Doppino piatto schermato per la stessa applicazione del caso a) (soluzione poco usata)
- *Esempio c)*
 - Cavo coassiale, con calza esterna usata come filo di ritorno del segnale, per collegare un segnale di tensione e un circuito di amplificazione non invertente
 - La corrente nella calza è uguale e opposta a quella nel conduttore centrale e ne annulla il campo magnetico generato all'esterno dello schermo, evitando emissioni elettromagnetiche
 - Cavo coassiale da preferire al doppino schermato e twistato per $f > 100\text{kHz}$ → poco usato nelle applicazioni di controllo



ingressi analogici

Collegamenti di massa

- I **collegamenti di massa** sono causa di accoppiamento tra i circuiti che ne hanno parti in comune
- **Massa di segnale** (*signal ground*): nodo di riferimento rispetto al quale vengono misurate le tensioni negli altri nodi del circuito
- Le masse possono essere collegate a **terra** (*earth ground*):
 - Per non lasciare fluttuare il potenziale
 - Per ragioni di sicurezza

ingressi analogici

Collegamenti di massa



- Senza un collegamento diretto, la tensione tra la massa e la terra può raggiungere valori imprevedibili
- Il *collegamento di sicurezza* è necessario nelle apparecchiature in cui sono presenti tensioni pericolose per le persone in caso di contatto accidentale
- Nei sistemi di acquisizione e controllo di processo si considerano almeno due masse
 - La **massa del campo** (o dei sensori e degli attuatori)
 - La **massa dei circuiti di acquisizione**

ingressi analogici

Collegamenti di massa



- Se entrambe le masse sono connesse localmente a terra, si forma un collegamento tra le masse stesse attraverso un *percorso di terra*
 - Collegando le masse per realizzare il percorso di ritorno del segnale, si realizza un anello chiuso (**anello di massa** o *ground loop*)
 - La terra non è equipotenziale → nell'anello di massa nasce una circolazione di corrente
 - Le differenze di potenziale di terra, in impianti di grande dimensione, possono essere di diverse decine di volt e possono variare nel tempo (attacco di motori, ecc.)



ingressi analogici

Amplificatori per il condizionamento

- I convertitori A/D richiedono in ingresso tensioni di *alto livello* ($\sim V$)
 - Segnali di *basso livello* ($\sim mV$) vanno amplificati
- Amplificatori utilizzati nei sistemi di controllo:
 - Amplificatori *operazionali*
 - Amplificatori *da strumentazione*
 - Amplificatori *d'isolamento*
- **Amplificatori operazionali** sono impiegati in piccoli sistemi, con specifiche di prestazioni poco stringenti e in ambienti poco rumorosi



ingressi analogici

Amplificatori per il condizionamento

- **Amplificatori da strumentazione e gli amplificatori d'isolamento** sono impiegati in contesti rumorosi e per i quali sono richieste specifiche stringenti
 - Costi maggiori
 - Circuiti più complessi e di maggiori dimensioni



ingressi analogici

Collegamenti single-ended e differenziali

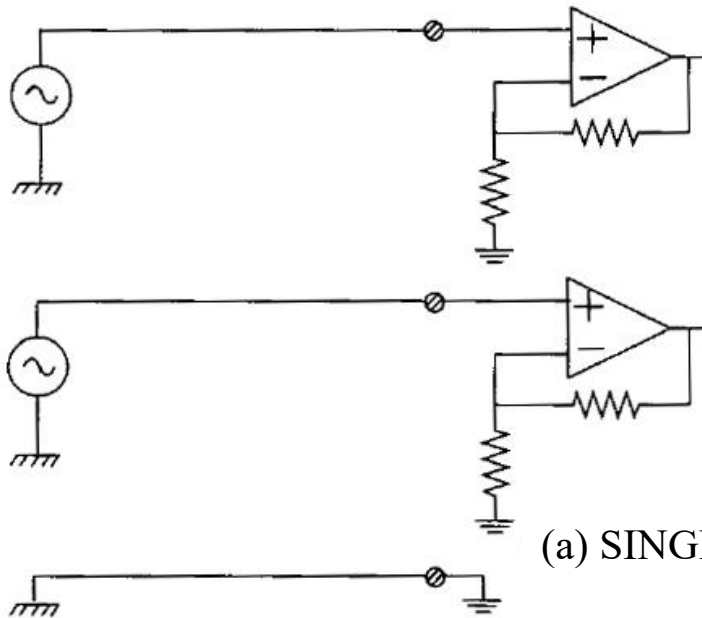
- Schede di acquisizione dati sono in grado di acquisire
 - n segnali *differenziali* → bifilari
 - $2n$ segnali *single-ended* → monofilari
- Alcuni segnali sono intrinsecamente differenziali
 - *Esempio*: uscita di un ponte di Wheatstone
- Costo della trasmissione e del condizionamento dei segnali differenziali è circa il doppio di quella single-ended → conviene utilizzare una trasmissione single-ended quando è possibile

ingressi analogici

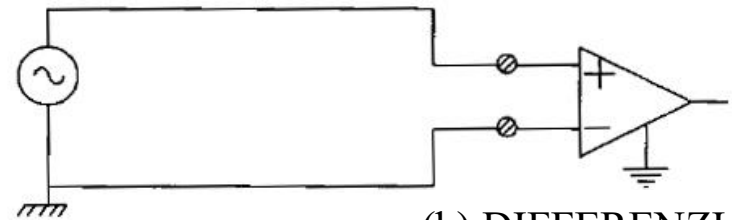
Collegamenti single-ended e differenziali



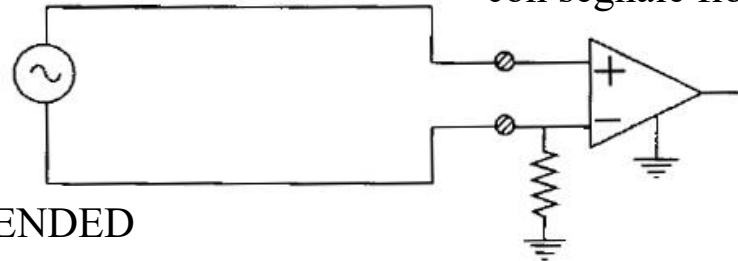
- Esempi di collegamenti differenziali e single-ended:



(a) SINGLE-ENDED



(b) DIFFERENZIALE
con segnale floating



(c) DIFFERENZIALE
con segnale a massa

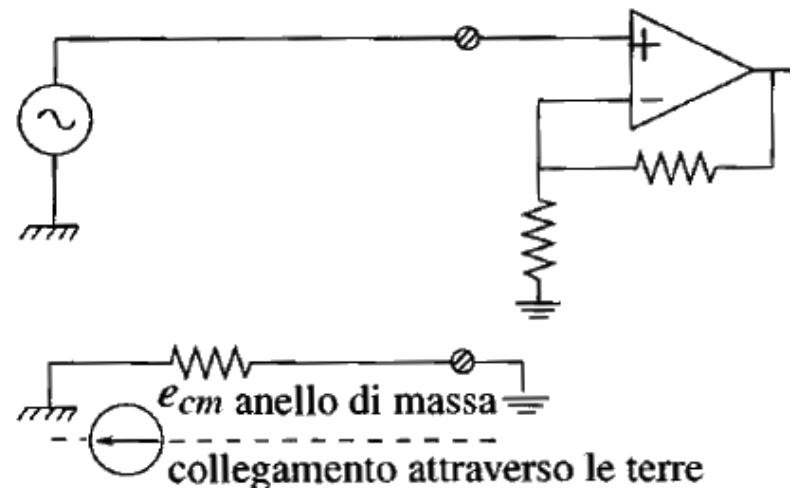
Massa di campo
con ritorno comune



ingressi analogici

Collegamenti single-ended e differenziali

- Le connessioni single-ended sono fortemente influenzate dalle **differenze di potenziale tra le masse** che si somma al segnale da acquisire
- La differenza di potenziale tra le masse è dovuta a
 - Differenza di potenziale dei punti di terra a cui sono collegate localmente le masse
 - Correnti degli altri trasduttori
 - Correnti estranee sul percorso di ritorno comune





ingressi analogici

Collegamenti single-ended e differenziali

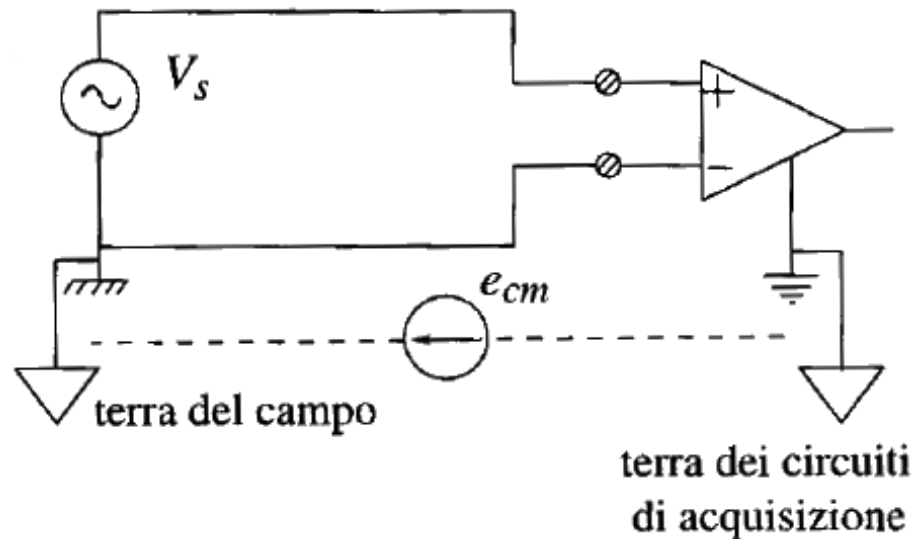
- Le connessioni single-ended sono utilizzabili quando la differenza di potenziale tra le masse è piccola rispetto al segnale da acquisire e alla risoluzione della catena di misura
 - In queste ipotesi non viene introdotto un errore di misura apprezzabile



ingressi analogici

Collegamenti single-ended e differenziali

- Nelle connessioni differenziali la differenza di potenziale tra le masse compare come una **tensione di modo comune** (e_{cm}) presente su entrambi i terminali dell'amplificatore





ingressi analogici

Collegamenti single-ended e differenziali

- La tensione di modo comune (e_{cm}) in uscita ad un amplificatore differenziale compare divisa per il fattore CMRR (*Common Mode Rejection Ratio*):

$$V_u = G_d (V_s + e_{cm} / \text{CMRR})$$

nelle ipotesi che $Z_{in} \gg R_1, R_2$ e $Z_{cm} \rightarrow \infty$

- *Esempio:* $V_s = 0.02\text{V}$, $e_{cm} = 4\text{V}$, $\text{CMRR} = 80\text{dB} \rightarrow$ errore 2%



ingressi analogici

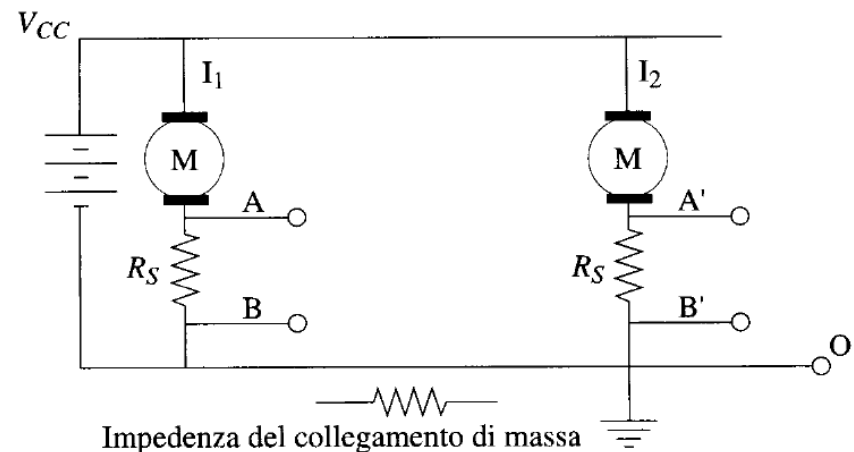
Collegamenti single-ended e differenziali

- **Collegamento a massa** dell'ingresso invertente dell'amplificatore per i collegamenti differenziali
 - È necessario per realizzare un percorso di ritorno a massa delle *correnti di polarizzazione* dell'amplificatore
 - In assenza di un apposito collegamento a massa, le correnti ritornano a massa attraverso le capacità parassite → le capacità si possono caricare nel tempo → aumenta la tensione di modo comune → quando la tensione di modo comune raggiunge valori prossimi a quelli dell'alimentazione l'amplificatore va in **saturatione**

ingressi analogici

Collegamenti single-ended e differenziali

- In alcuni casi il collegamento a massa non è praticabile → **amplificatori di isolamento**
 - Possono funzionare con la massa del circuito di ingresso flottante
- *Esempio di percorsi di massa comune*: acquisizione delle misure di corrente di due motori
 - Si rilevano le tensioni ai capi di due resistenze in serie agli avvolgimenti di statore





ingressi analogici

Collegamenti single-ended e differenziali

- Le misure differenziali sono V_{AB} e $V_{A'B'}$
- Le misure single-ended sono V_{AO} e $V_{A'O'}$
- A differenza di V_{AB} , la tensione V_{AO} dipende oltre che da I_1 , anche dalla d.d.p. sul percorso di ritorno dell'alimentazione
- Nel collegamento differenziale le tensioni V_B e $V_{B'}$ misurate rispetto alla massa del sistema di acquisizione appaiono come tensioni di modo comune

ingressi analogici

Isolamento galvanico



- **Isolamento galvanico:** due circuiti si dicono isolati galvanicamente se l'impedenza che li collega è superiore a $100\text{M}\Omega$ per le correnti continue
 - È impedita la circolazione di correnti in continua ed a *basse frequenze*
 - È impedita la circolazione di correnti alle frequenze di rete (50Hz) e i suoi primi multipli
 - Principali disturbi impiantistici e dell'ambiente manifatturiero

ingressi analogici

Isolamento galvanico



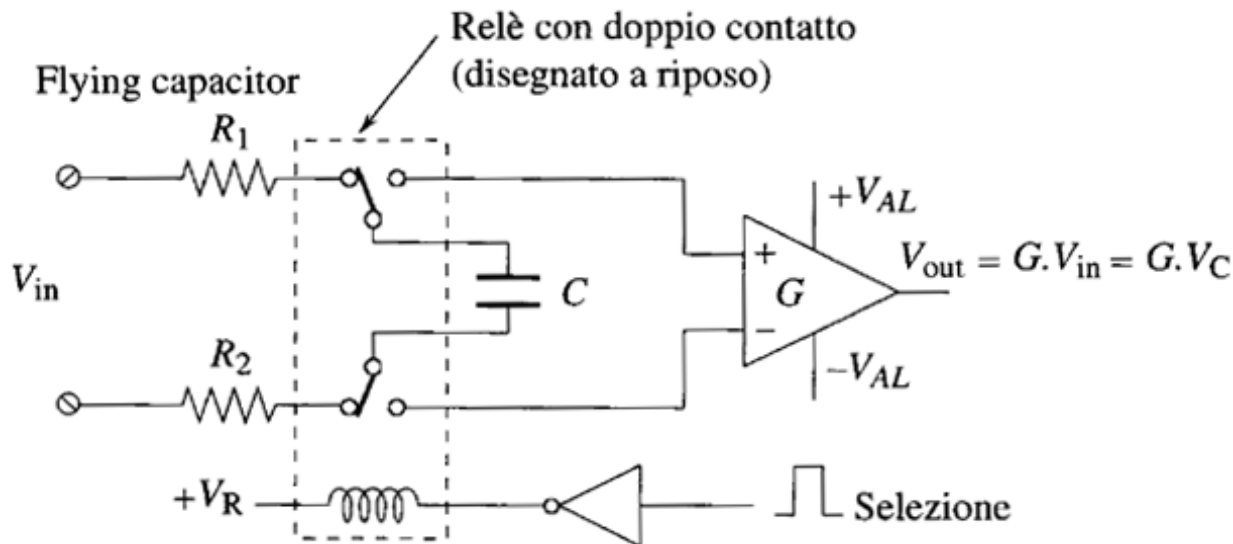
- Realizzazioni più comuni:
 - **Accoppiamento a trasformatore**, con modulazione e demodulazione delle componenti continue
 - Garantisce maggiore isolamento e affidabilità
 - Molto costoso
 - Impiego in grandi impianti, rischiosi e con requisiti stringenti di affidabilità (es. produzione di energia)
 - **Accoppiamento ottico** → coppia diodo LED-fototransistor
 - Soluzione economica
 - Impiego in piccoli e medi impianti con caratteristiche meno stringenti di affidabilità
 - Dimensioni molto contenute

ingressi analogici

Isolamento galvanico



- **Flying capacitor (capacità commutata):**
 - *Soluzione economica*
 - Componente elettromeccanico → *poco affidabile*
 - Realizza anche il *filtraggio* e il *campionamento*, ma a basse frequenze (lenta stabilizzazione dei contatti a relè ~ms)



ingressi analogici

Isolamento galvanico



- **Scopi funzionali** dell'isolamento galvanico:
 - Consentire il funzionamento integrato dei circuiti di misura e trasmissione e di quelli di acquisizione con riferimenti di tensione (masse) indipendenti
 - Sarà possibile acquisire segnali di piccola intensità sovrapposti ad elevate tensioni di modo comune
 - Riduzione degli effetti di disturbi su percorsi e anelli di massa
 - Interruzione delle correnti continue di bassa frequenza
 - La realizzazione reale di impianti complessi spesso sfugge al progettista → si creano facilmente, anche in modo accidentale, collegamenti e anelli di massa durante la messa in opera degli impianti

ingressi analogici

Isolamento galvanico



- **Sicurezza:**

- L'isolamento galvanico protegge i circuiti a valle (o a monte) e le persone:

- rispetto a differenze di potenziale elevate tra ingresso e uscita (es. dovute a guasti)
 - da contatti accidentali con alte tensioni
 - da scariche elettriche sulla sorgente del segnale o sulle linee di trasmissione (es. di origine atmosferiche)

- Molto impiegato nei *sistemi di monitoraggio medicali*

- Salvaguardia dei pazienti
 - Variazioni di potenziale di massa potrebbero danneggiare, in caso di contatto, le parti interne dell'organismo (molto delicate)



ingressi analogici

Segnali standard

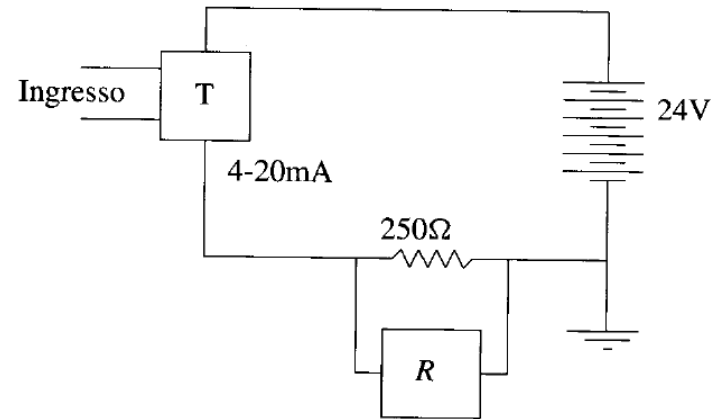
- Sistemi di controllo di processo e macchine di piccola e media dimensione sono realizzati con **componenti discreti**
 - Unità a se stanti che si scambiano segnali di misura e comando
- Per le misure si utilizzano trasmettitori con
 - Trasmissione in *corrente*
 - Trasmissione in *tensione*

ingressi analogici

Segnali standard



- **Trasmissione in corrente:** segnali in corrente nel campo $4 \div 20 \text{ mA}$
 - Segnale minimo pari a 4 mA
 - *zero vivo* → rilevazione di guasti
 - *alimentazione* dei circuiti trasmettitori senza ulteriori cavi
 - Normalmente il trasmettitore opera come un **generatore di corrente** comandato dal segnale di ingresso:
 - Compensazione dei disturbi in tensione
 - Compensazione degli effetti di variazione della resistenza di carico o dei cavi di collegamento



ingressi analogici

Segnali standard



- Consente l'invio a grandi distanze senza perdita di accuratezza
- La **massima distanza** di trasmissione dipende:
 - Dall'accuratezza richiesta
 - Dalle sorgenti di rumore e dalla loro distanza
 - Dall'ampiezza e dalla frequenza del rumore
 - Dall'efficacia degli schermi
- *Esempio*: segnali in corrente di banda 10Hz, cavi schermati, rumore di livello industriale medio, accuratezza desiderata dello 0.5% → distanze tra 300 e 1.500 m

ingressi analogici

Segnali standard



- **Trasmissione in tensione:** segnali standard $0\div 5\text{ V}$, $0\div 10\text{ V}$, $\pm 10\text{ V}$,
- Viene di norma utilizzata per
 - Applicazioni con distanze ridotte (es. laboratorio)
 - Sistemi di controllo di macchine automatiche
 - Collegamenti interni (o nei pressi) degli armadi di controllo
- Le massime distanze di trasmissione sono:
 - Per segnali da ± 1 a $\pm 10\text{ V} \rightarrow 15\div 90\text{ m}$
 - Per segnali da ± 0.01 a $\pm 1\text{ V} \rightarrow 1.5\div 30\text{ m}$
- Oggi si prediligono le **trasmissioni digitali**



ingressi analogici

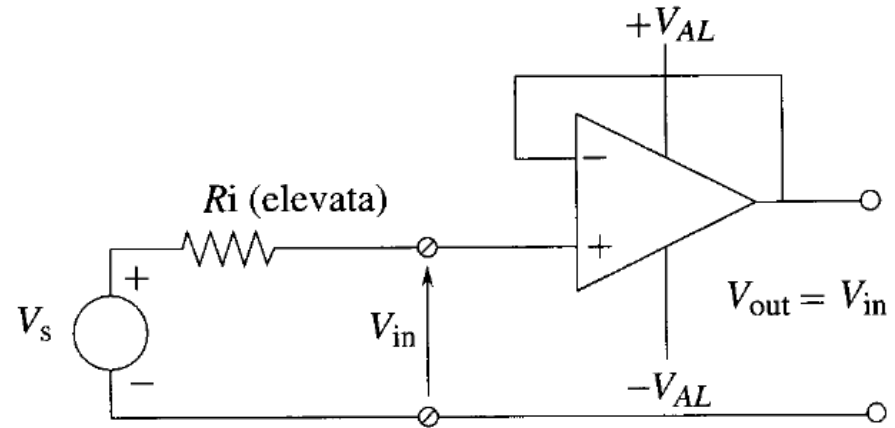
Circuiti di condizionamento: Adattamento d'impedenza

- Elementi sensibili e trasduttori sono modellabili come generatori *reali* di tensione
- Se il circuito di interfaccia assorbe corrente, il valore letto sarà affetto da errore a causa della caduta di tensione sulla resistenza interna del generatore
- È necessario che i circuiti di interfaccia esibiscano *impedenza di ingresso elevate* ($\sim 100 \text{ K}\Omega$)
- Per lo stesso motivo, è necessario che il circuito di interfaccia presenti *in uscita una bassa impedenza*

ingressi analogici

Circuiti di condizionamento: Adattamento d'impedenza

- Circuiti operazionali configurati come *inseguitore* o *buffer* presentano impedenze di ingresso elevate

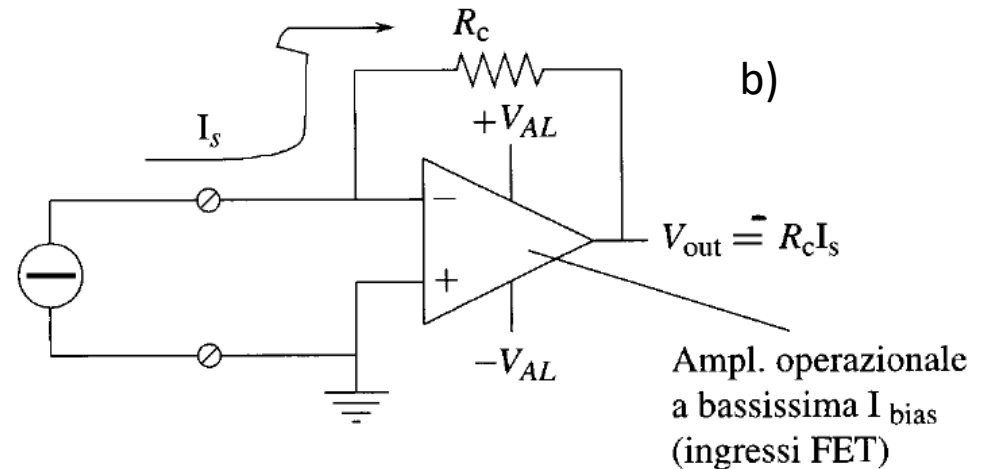
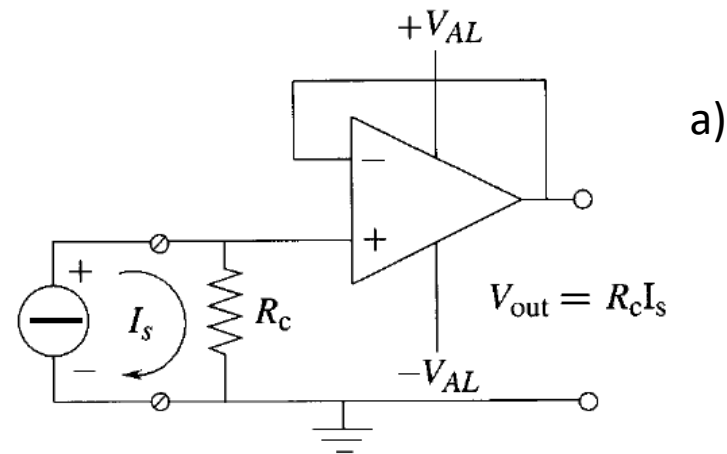


- L'uscita insegue l'ingresso entro i limiti definiti dalle tensioni di saturazione
- Impedenza di ingresso $\sim 100 \text{ M}\Omega$
- Impedenza di uscita molto minore di 100Ω

ingressi analogici

Circuiti di condizionamento: Conversione di corrente in tensione

- **Conversione corrente-tensione** mediante una resistenza calibrata ed un circuito inseguitore a)– *Esempio*: per un segnale 4÷20mA occorre una $R=250\Omega$ per ottenere 1÷5V





ingressi analogici

Circuiti di condizionamento: Conversione di corrente in tensione

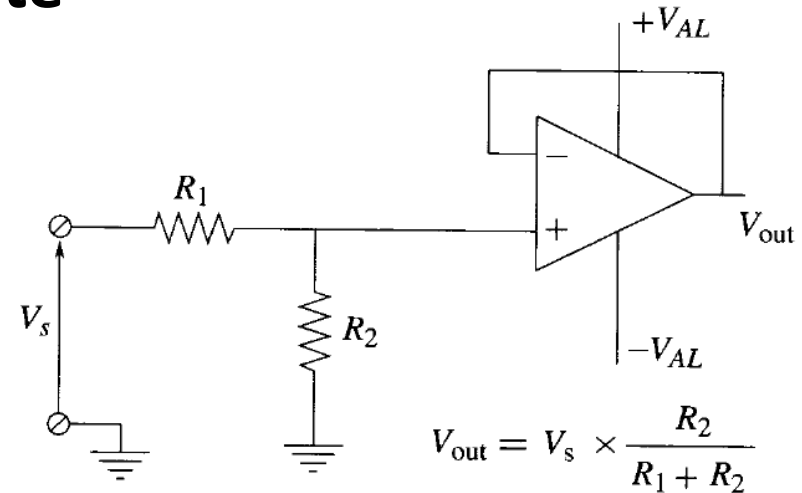
- La resistenza deve essere precisa, stabile ed avere un basso coefficiente di temperatura
- Correnti di basso livello (μA) non possono essere convertite con lo schema a) perché occorrerebbero resistenze di alcuni $\text{M}\Omega$
 - Resistenza di ingresso diventerebbe paragonabile alla resistenza del generatore reale di tensione
 - Si avrebbero errori di lettura delle correnti rilevanti
- Il circuito b) è adatto anche a correnti di basso livello
 - Amplificatore con bassa corrente di polarizzazione

ingressi analogici

Circuiti di condizionamento: Messa in scala dei segnali analogici



- Tensioni superiori a quelle accettate dai convertitori A/D devono essere **attenuate**
 - Uso di *partitori di tensione*
 - R_1 troppo alta abbassa la banda a causa della *capacità parassita* C_p in parallelo a R_2 altrimenti ...
 - Si introduce una capacità C_s in parallelo a R_1 tale per cui risulta $R_1 C_s = R_2 C_p$ nell'ipotesi che l'impedenza della sorgente sia piccola rispetto a $R_1 + R_2$
 - *Ipotesi*: impedenza di ingresso dell'amplificatore $\ll R_2$

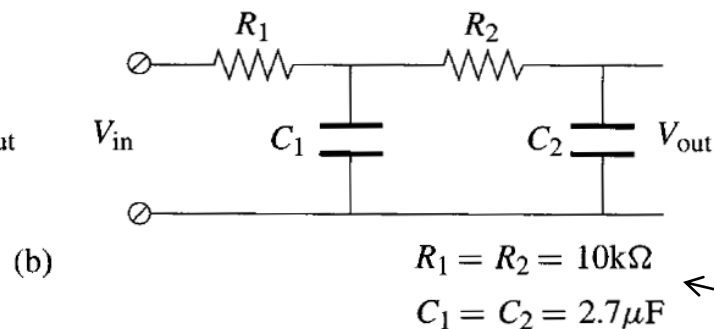
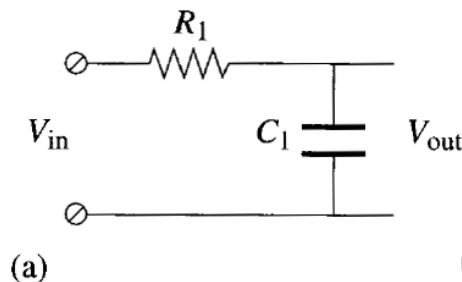


ingressi analogici

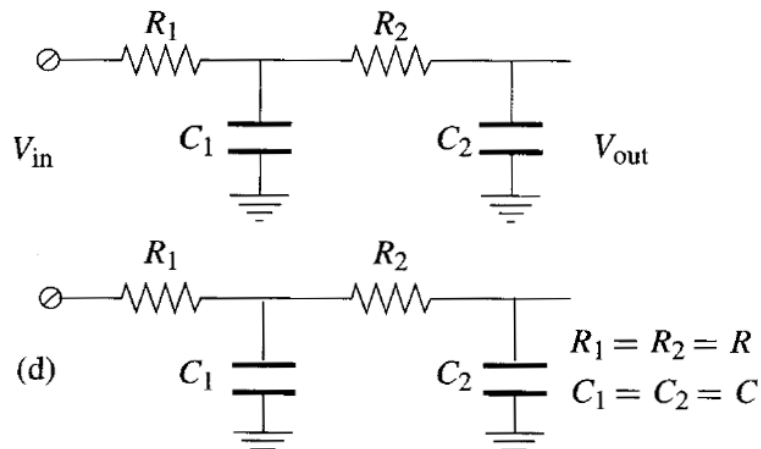
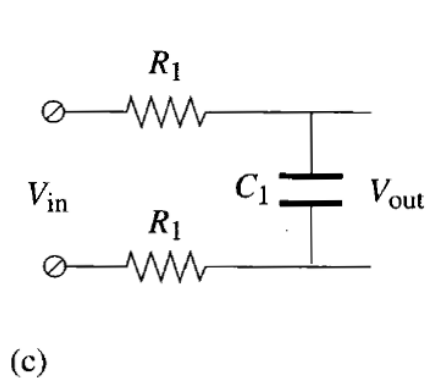
Circuiti di condizionamento: Filtraggio



- **Prefiltraggio** necessario per attenuare il rumore a frequenze superiori a quella di Nyquist



Attenuazione
di -6dB a 1Hz ,
 -52dB a 60Hz



ingressi analogici

Circuiti di condizionamento: Filtraggio



– Filtri passivi

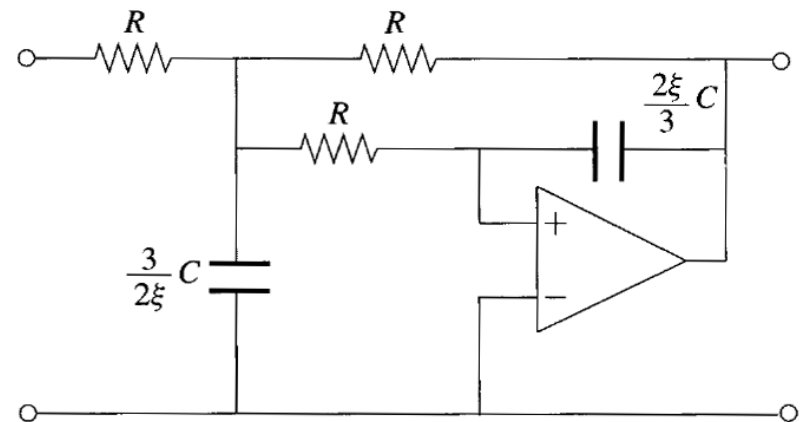
- a) Passa-basso a un polo per collegamenti single-ended
- b) Passa-basso a due poli per collegamenti single-ended
- c) Passa-basso a un polo per collegamenti differenziali
- d) Passa-basso a due poli per disturbi di modo comune nelle connessioni differenziali (rami bilanciati → valori R e C uguali)

– Frequenza di taglio scelta in base alla banda dei disturbi

– **Filtro attivi** migliori ma più costosi e ingombranti

$$G_f(s) = \frac{\omega_f^2}{s^2 + 2\xi\omega_f s + \omega_f^2}$$

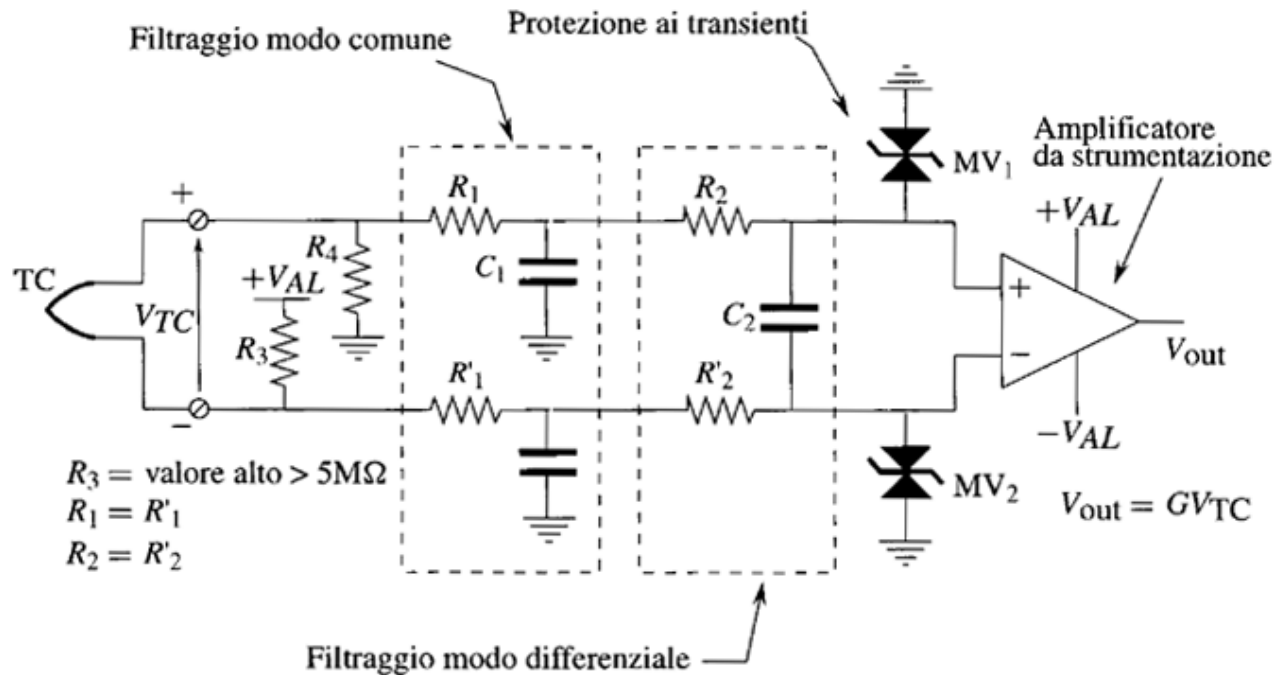
con $\omega_f = 1/(RC)$



ingressi analogici

Circuiti di condizionamento: Condizionamento dei segnali di termocoppie

- **Circuito di condizionamento di termocoppie**
 - Autodiagnostica (*fault detection*)
 - Protezione da eventi rovinosi accidentali





ingressi analogici

Circuiti di condizionamento: Condizionamento dei segnali di termocoppie

- Composto da
 - Un amplificatore differenziale
 - Un filtro differenziale
 - Un filtro di modo comune
 - Circuito per il riconoscimento di termocoppia aperta
 - Protezione da sovratensioni transitorie
- R_3 porta in saturazione l'amplificatore in caso di apertura della termocoppia
- R_4 fa scorrere a massa le correnti di polarizzazione
- *Varistori* proteggono l'amplificatore da tensioni impulsive elevate

ingressi analogici

Circuiti di condizionamento: Condizionamento dei trasduttori resistivi



- **Trasduzione resistenza-tensione** nei sensori resistivi (termoresistenze, estensimetri) mediante *ponte di Wheatstone* o *generatore di corrente*
 - Sensori resistivi hanno bassa sensibilità → aumentare la dinamica (escursione) del segnale di uscita
- *Esempio: generatore di corrente con Pt100*
 - Corrente limitata a 2mA per evitare l'autoriscaldamento
 - $I_{RTD} = 2\text{mA}$, sensibilità di $0.4 \Omega/^{\circ}\text{C}$ → uscita $0.8 \text{ mV}/^{\circ}\text{C}$
 - Per adattare il segnale ai convertitori A/D occorrerebbe amplificare di più di 100 volte



ingressi analogici

Circuiti di condizionamento: Condizionamento dei trasduttori resistivi

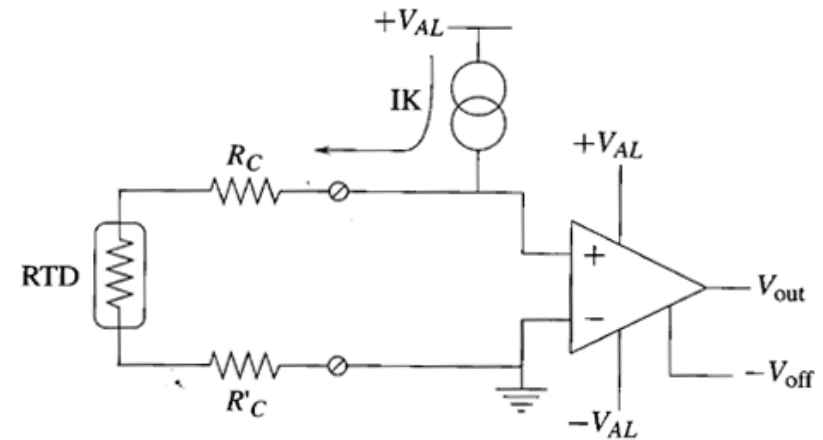
- Tensione a riposo (0°C) pari a 200mV $\rightarrow$ guadagno limitato a circa 10, per evitare la saturazione
- Convertitore a 12bit (ingresso $0\div 10\text{V}$, quantizzazione 2.44mV) $\rightarrow$ sensibilità 0.3°C
- Sistemi a ponte consentono di *azzerare la tensione a riposo* $\rightarrow$ si può aumentare il guadagno $\rightarrow$ aumenta la sensibilità
 - Maggior costo (resistori di alta precisione e stabilità)

ingressi analogici

Circuiti di condizionamento: Condizionamento dei trasduttori resistivi



- Soluzioni a generatore + **amplificatore con offset** pari alla tensione a riposo → aumenta la dinamica ma il costo è paragonabile a quello di un ponte di Wheatstone
- Errori dovuti alle resistenze dei cavi possono essere ridotti con una **soluzione a 4 fili**:
 - due fili usati per il pilotaggio in corrente ($I = 2\text{mA}$)
 - due fili usati per la lettura della tensione ($I =$ corrente di polarizzazione dell'amplificatore)

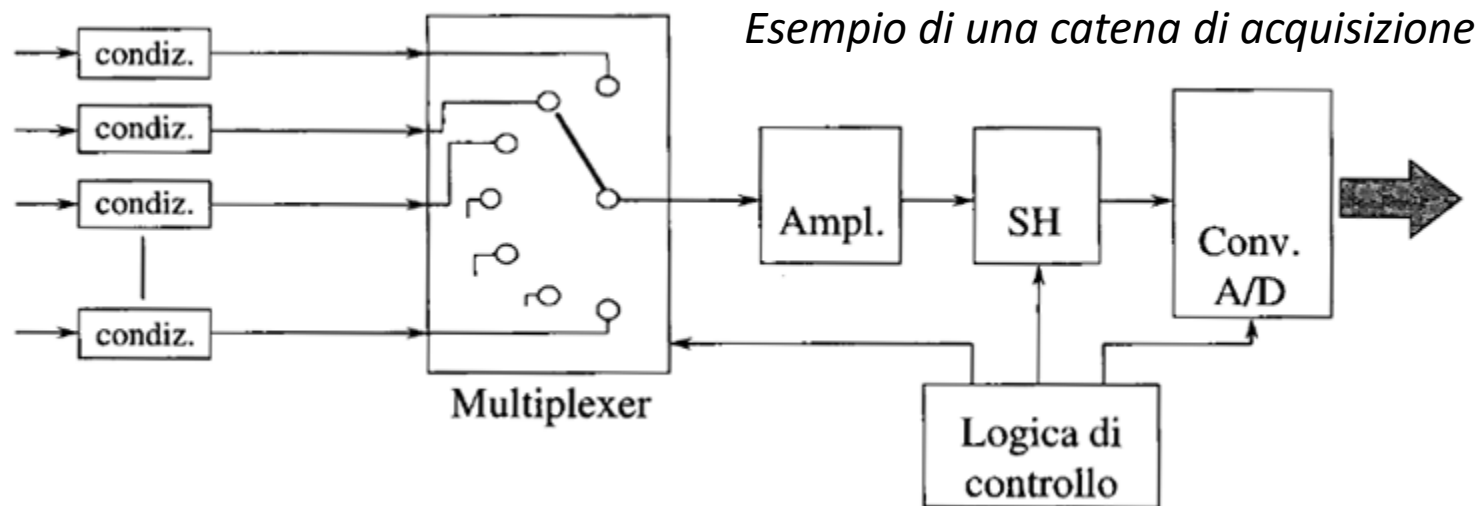


$$V_{\text{out}} = \left[IK \left(RTD + R_C + R'_C \right) G \right] - V_{\text{off}}$$

ingressi analogici

Multiplexer

- Un **multiplexer** è un deviatore a più posizioni che permette di utilizzare un solo convertitore A/D per l'acquisizione di più ingressi
 - Riduzione dei costi
 - Riduzione della velocità di acquisizione

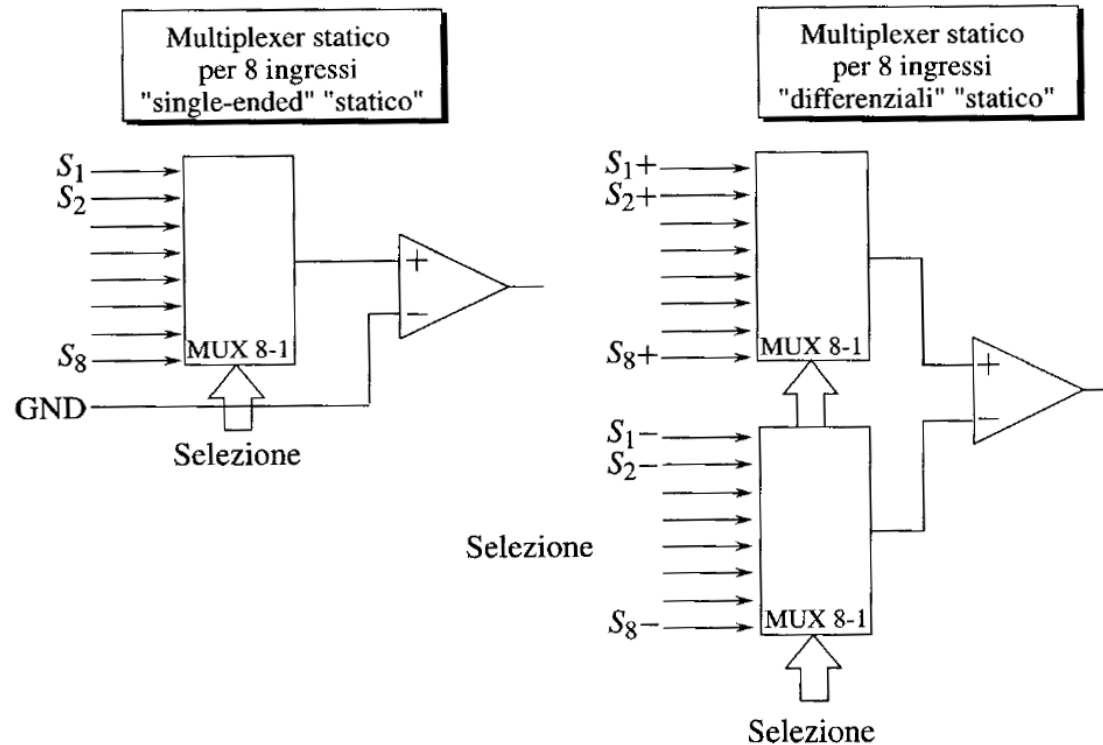


ingressi analogici

Multiplexer



- La riduzione dei costi dei convertitori A/D ne sta limitando la diffusione
- Componenti statici realizzati in tecnologia MOS





ingressi analogici

Multiplexer: Dimensionamento

- **Throughput**: velocità di acquisizione $v_a = n f_s$, dove n è il numero di ingressi e f_s la frequenza di campionamento → rappresenta la *frequenza di commutazione*
- T_{CO} è il **tempo di conversione totale** di un segnale, pari alla somma dei tempi di *commutazione* del multiplexer, di *assestamento* dell'amplificatore, di *campionamento* del campionatore-mantenitore (SH) e di *conversione* del convertitore A/D
→ deve risultare: $T_{CO} < 1/(n f_s)$
- **Skew**: canali acquisiti in istanti di tempo diversi
 - Se *skew* eccessivo per l'applicazione → n SH prima del multiplexer



ingressi analogici

Campionamento e mantenimento

- **Campionatore-mantenitore (*Sample & Hold, SH*):** acquisiscono il segnale di ingresso e lo conservano per il tempo necessario alla conversione
 - Alcuni convertitori richiedono un segnale di ingresso costante durante la conversione
 - I SH non sono necessari per i convertitori più veloci
 - La necessità va valutata in base alla variabilità del segnale in rapporto al periodo di campionamento e all'accuratezza
 - *Esempio:* segnale sinusoidale $\rightarrow$ in un intervallo t_a , la massima variazione si ha in corrispondenza del passaggio per lo zero:

$$\Delta V = t_a \frac{d}{dt} (A \sin \omega t) \Big|_{t=0} = t_a A \omega$$



ingressi analogici

Campionamento e mantenimento

- Massimo errore di conversione rapportato al valore picco-picco

$$\varepsilon = \frac{\Delta V}{2A} = \frac{t_a \omega}{2} = \pi f t_a$$

- Con riferimento ad un convertitore a n bit, l'errore di conversione ($lsb: 1/2^n$) deve essere al più pari ad ε , da cui si ricava il limite superiore per il tempo t_a da impiegare per la conversione

$$t_a \leq \frac{1}{2^n \pi f}$$

- Con $n=10, f=1\text{KHz}$ si ottiene $t_a=311\text{ns}$ → valore troppo piccolo per i convertitori commerciali!

→ **segnali a frequenze elevate richiedono i SH**, che essendo più veloci dei convertitori, ne riescono a rilevare correttamente i valori

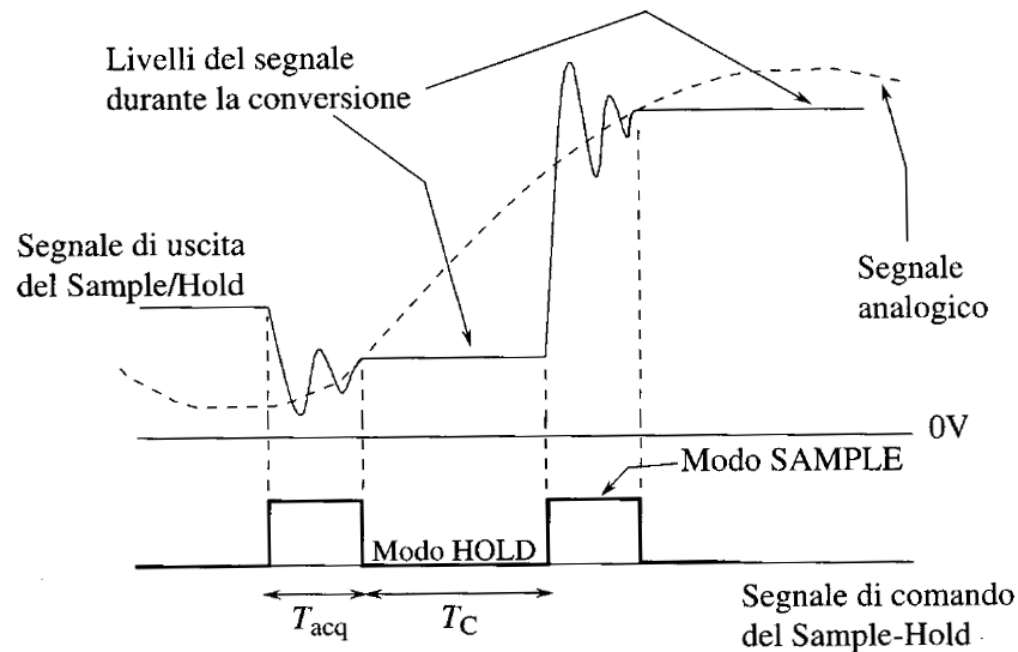
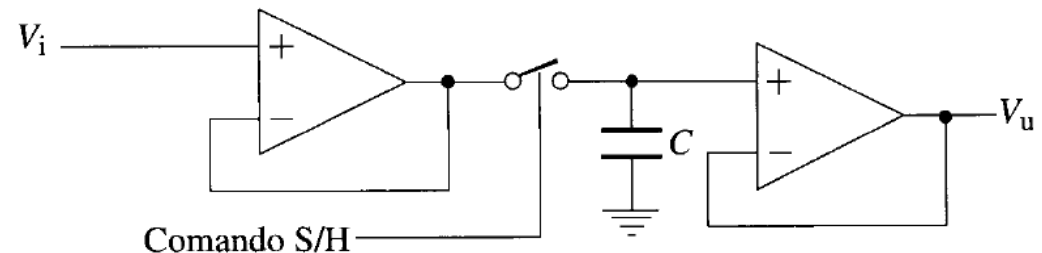
ingressi analogici

Campionamento e mantenimento

- Schema di principio

- Due fasi:

- Modo *sample*
- Modo *hold*



ingressi analogici

Campionamento e mantenimento



- Fattori che determinano le prestazioni dei SH:
 - Valore finito della corrente di carica del condensatore
 - Limita la velocità di carica del condensatore e quindi la velocità di variazione del segnale di ingresso (*slew-rate*)
 - Corrente di fuga
 - Calo della tensione di uscita nel tempo (*droop*) inversamente proporzionale alla capacità → bisognerebbe aumentare la C
 - Se aumenta la C peggiora lo *slew-rate* → Trade-off!
 - Occorrono capacità di ottima qualità (es. polipropilene) → basse perdite
 - Incertezza sul tempo di apertura dell'interruttore (*jitter*)
 - Tipicamente limitato ($\sim$ ps)



ingressi analogici

Campionamento e mantenimento

- Massima frequenza del segnale in ingresso che determina errori inferiori alla quantizzazione

$$f_{lsb} = \frac{1}{2^n \pi t_a}$$

- *Esempio*: convertitore a 8 bit, $t_a = 150 \mu s \rightarrow f_{lsb} = 8.28 \text{ Hz}$
molto inferiore al limite teorico $1/t_a = 6.6 \text{ KHz}$
- Con l'introduzione dei SH:
 - Si riduce la velocità teorica di conversione $1/(t_a + t_{SH})$, ma di pochissimo dato che $t_{SH} \sim 10 \text{ ns} \ll t_a$
 - La f_{lsb} cresce fino al valore limite secondo il teorema del campionamento (frequenza di Nyquist $f_N = 1/2t_a$)



ingressi analogici

Conversione A/D

- **Convertitore analogico digitale (ADC)** a n bit riceve in ingresso un segnale analogico e lo codifica in un segnale digitale a n bit
 - Ogni codice binario rappresenta il rapporto tra l'ampiezza del segnale di ingresso V_x e l'intera scala V_{fs} del convertitore, arrotondato all'intero più vicino

$$B_x = \text{Int} \left(\frac{V_x}{V_{fs}} 2^n + \frac{1}{2} \right) \quad \text{per} \quad V_x < V_{fs} \left(1 + \frac{1}{2^{n+1}} \right)$$

$$B_x = 2^n - 1 \quad \text{per} \quad V_{fs} \left(1 - \frac{1}{2^{n+1}} \right) \leq V_x < V_{fs}$$

dove $\text{Int}(x)$ rappresenta la parte intera di x



ingressi analogici

Conversione A/D

- Due segnali V_{x1} e V_{x2} tali che

$$\frac{|V_{x1} - V_{x2}|}{V_{fs}} 2^n < 1 \quad \Leftrightarrow \quad |V_{x1} - V_{x2}| < \frac{V_{fs}}{2^n}$$

possono dar luogo alla stessa uscita

- **Errore di quantizzazione:** $V_{fs}/2^n$
- **Risoluzione:** $100/2^n$
 - errore di quantizzazione espresso in percentuale del fondo scala
- *Convertitore ideale* → solo errore di quantizzazione
- **Livelli di quantizzazione:** valori in corrispondenza dei quali si ha in uscita la transizione tra due codici successivi

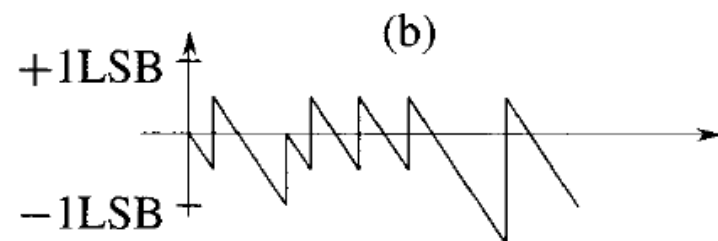
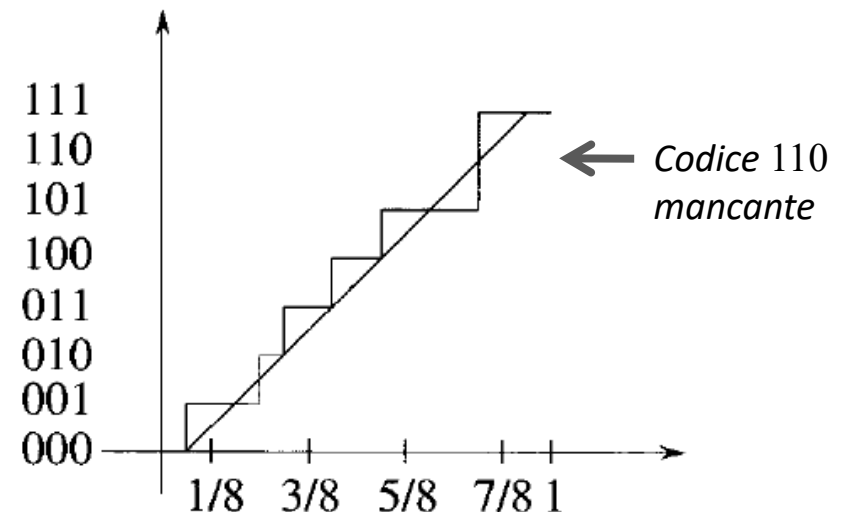
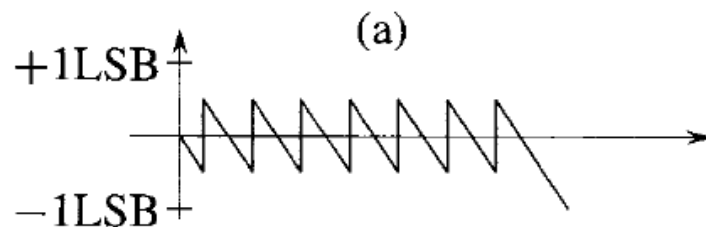
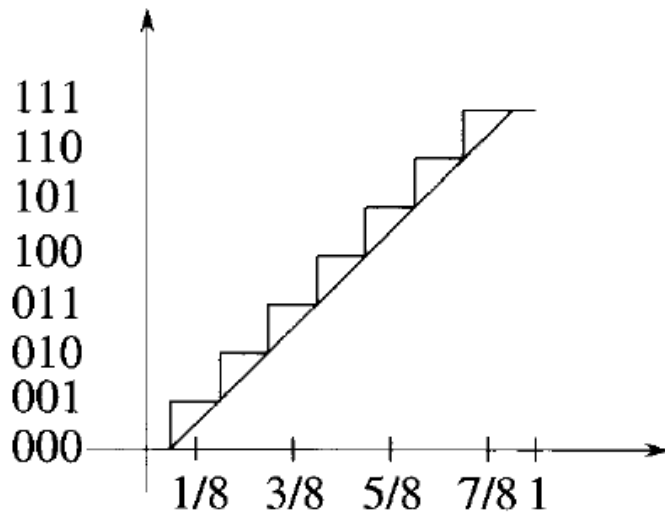
$$kV_{fs}/2^{n+1} \quad \text{per} \quad k = 1, 3, \dots, 2^{n+1}-3$$

ingressi analogici

Conversione A/D



- Relazione ingresso-uscita di un ADC a 3 bit: a) ideale, b) reale



ingressi analogici

Conversione A/D



- **Errore differenziale di linearità:** scostamento tra la distanza ideale ($V_{fs}/2^n$) e quella reale di due livelli di transizione successivi
 - Se l'errore differenziale di linearità è pari a 1 lsb ci sarà un *codice mancante* in uscita
- **Errore integrale di linearità:** differenza tra l'ampiezza effettiva del segnale di ingresso nei punti di transizione del codice e quella teorica
 - Pari alla sommatoria degli errori differenziali di linearità
- **Mancata monotonia:** in un certo campo di ampiezze dell'ingresso, al crescere (diminuire) delle ampiezze può corrispondere una diminuzione (aumento) del codice



ingressi analogici

Conversione A/D: risoluzione ed accuratezza

- La **risoluzione complessiva** è un parametro fondamentale per il dimensionamento di un sistema di acquisizione
 - Risoluzione del convertitore deve essere superiore a quella del sensore $\rightarrow n$ tale che $1/2^n < R$, dove R è la risoluzione del sensore espressa in “*per unità*” (p.u.)
 - Nella pratica si considera la relazione tra i reciproci: $[2^n]_{\text{dB}}$ deve superare il “*range dinamico*” del sensore $[1/R]_{\text{dB}}$ (inverso della risoluzione in dB)
 - *Esempio*: sensore con range dinamico di 80dB $\rightarrow$ ADC a 14bit, corrispondente a 84dB (12 bit corrispondono a 78dB)
 - Risoluzione ADC: per 12 bit $\rightarrow$ 0.025% del campo di tensioni in ingresso, per 16 bit $\rightarrow$ 0.0015%



ingressi analogici

Conversione A/D: risoluzione ed accuratezza

- Oltre alla risoluzione del convertitore vanno adeguatamente dimensionati anche l'amplificatore, il filtro e il campionatore
 - *Esempio*: una variazione del segnale di 1mV non è rilevata da un convertitore a 12 bit con fondo scala 10V (minimo valore apprezzabile $10/4096 = 2.44\text{mV}$)
 - amplificando il segnale di un fattore 10, la stessa variazione sarebbe apprezzata (in assenza di rumore), ma parte del segnale potrebbe andare fuori scala del convertitore
- **L'accuratezza** è uno dei parametri che definisce la qualità di una *catena di acquisizione dati*
 - Definisce l'**errore totale** di ogni lettura effettuabile

ingressi analogici

Tipi di convertitori A/D



- **Tipi di convertitori** di maggiore diffusione:
 - ad Approssimazioni Successive
 - a Integrazione
 - Paralleli (Flash)
 - Sigma-Delta
- La **frequenza di campionamento** (o “*velocità*”) è un parametro importante per la scelta del convertitore
 - I convertitori si differenziano in base al principio di funzionamento e quindi alla velocità

ingressi analogici

Tipi di convertitori A/D



- **Convertitori ad approssimazioni successive** molto usati per frequenze superiori a 100Hz (arrivano fino al MHz)
 - Necessitano che il segnale in ingresso si mantenga costante durante la conversione → SH
- Per le basse frequenze si utilizzano i **convertitori ad integrazione**
 - Alte risoluzione a costi bassi
 - Velocità: 3÷100 campioni al secondo
 - Accurati e lineari
 - Mediano la variazione del segnale di ingresso durante il tempo di campionamento → questo equivale a realizzare una operazione di filtraggio sul segnale campionato, di cui bisogna tener conto!

ingressi analogici

Tipi di convertitori A/D



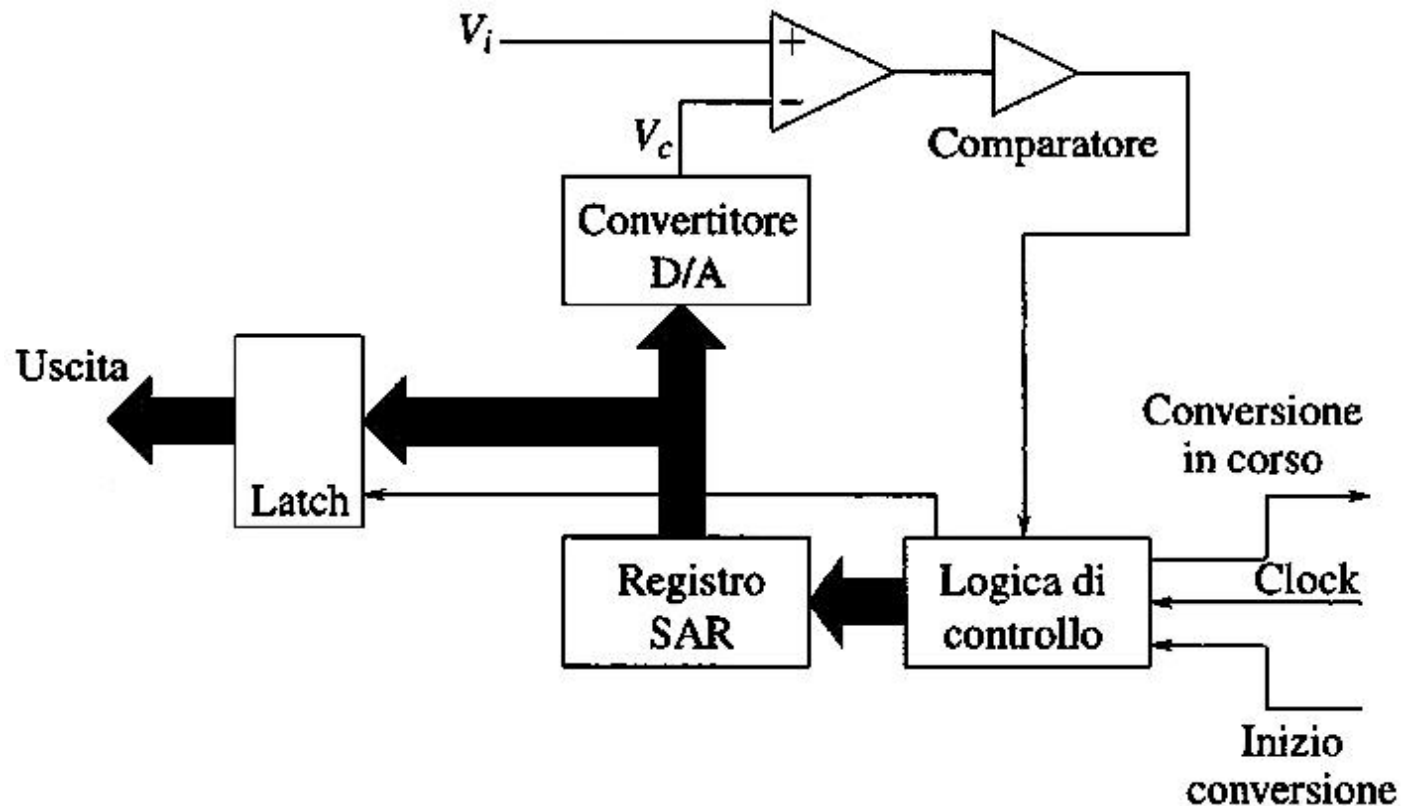
- **Convertitori Sigma-Delta** usati a basse frequenze
 - Accurati (più di quelli ad integrazione) e poco costosi
- **Convertitori paralleli** sono i più veloci
 - Molto costosi
 - La loro complessità ne limita la risoluzione a 8 bit

ingressi analogici

Convertitori A/D ad approssimazioni successive



- **Convertitore ad approssimazioni successive (ASAD)**





ingressi analogici

Convertitori A/D ad approssimazioni successive

- Funzionamento
 - Si basa sull'impiego di un convertitore digitale/analogico (D/A) che produce in uscita un segnale di ampiezza proporzionale al valore contenuto nel registro SAR (registro ad approssimazioni successive)
 - All'inizio del ciclo di conversione, il SAR pone ad 1 il bit più significativo del codice di uscita
 - La tensione di uscita V_c del D/A sarà pari alla metà del campo accettabile in ingresso
 - Se il comparatore rileva che la tensione $V_i \geq V_c$ allora il bit è mantenuto alto, altrimenti è abbassato



ingressi analogici

Convertitori A/D ad approssimazioni successive

- Successivamente è portato alto il bit successivo e si ripete il confronto
- Il procedimento è iterato e dopo n confronti si genera una tensione V_c uguale a V_i a meno dell'errore di quantizzazione
- Il dato numerico prodotto dal SAR rappresenta la codifica della tensione applicata in ingresso
- Prestazioni
 - *Risoluzione*: tra 8 e 16 bit
 - *Tempi di conversione*: $\sim \mu s$
 - *Ottimo rapporto costo-prestazioni*



uscite analogiche

Introduzione

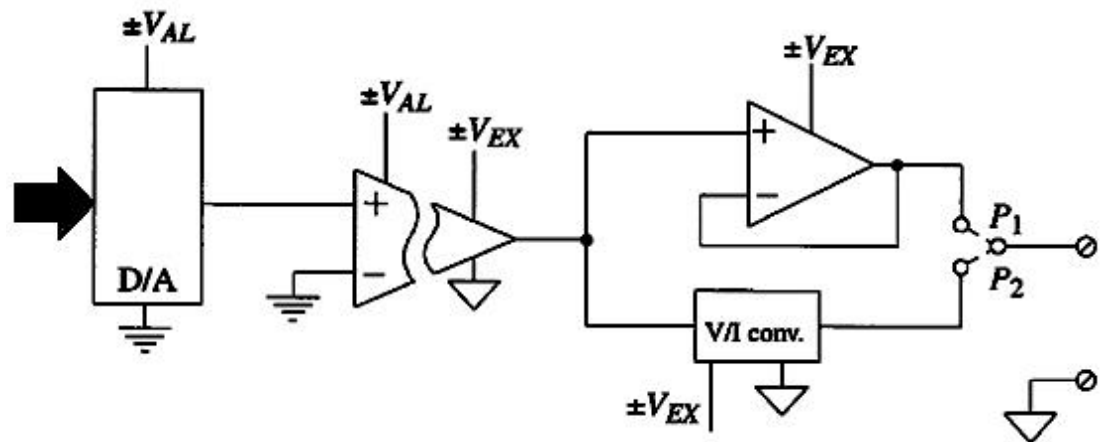
- Le **uscite analogiche** in un sistema di controllo provvedo a pilotare
 - Attuatori
 - Indicatori
 - Registratori a carta
 - ...
- Forniscono tensioni a livello alto ($\pm 1\text{V}$, $\pm 10\text{V}$) o segnali di corrente nel campo standard $4\div 20\text{mA}$
- Sono realizzati con convertitori D/A seguiti da circuiti amplificatori

uscite analogiche

Condizionamento



- **L'isolamento galvanico** spesso richiesto per le uscite
 - Negli impianti industriali le uscite analogiche possono essere collegate a svariati dispositivi
 - Possibilità di creare anelli di massa
 - Possibilità di generare scariche ambientali o originate da guasti
 - L'isolamento galvanico consente di proteggere cose e persone da questi tipi di problemi
 - *Esempio con amplificatore di isolamento*



uscite analogiche

Conversione D/A



- Principio di funzionamento dei **convertitori D/A**
 - Si utilizza la rappresentazione binaria del numero da convertire per aprire/chiusure interruttori elettronici e in questo modo generare una tensione ad esso proporzionale
 - Ampia gamma, realizzati con differenti tecnologie
 - Bipolari
 - CMOS
 - BiMOS
 - Realizzazioni in forma
 - Discreta
 - Monolitica

uscite analogiche

Conversione D/A a resistenze pesate

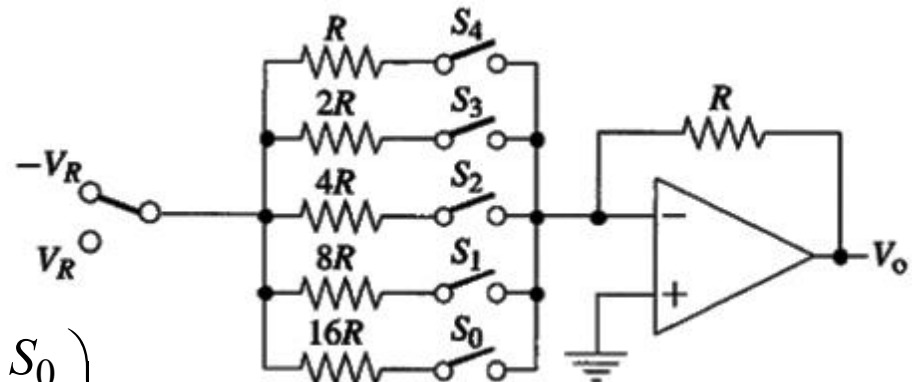


- **Convertitori D/A a resistenze pesate**

- *Esempio* a 5 bit con riferimento al numero binario ($\pm S_4 S_3 S_2 S_1 S_0$)

- Risulta

$$V_0 = \pm V_R \left(S_4 + \frac{S_3}{2} + \frac{S_2}{4} + \frac{S_1}{8} + \frac{S_0}{16} \right)$$
$$= \pm \frac{V_R}{16} \left(2^4 S_4 + 2^3 S_3 + 2^2 S_2 + 2^1 S_1 + S_0 \right)$$



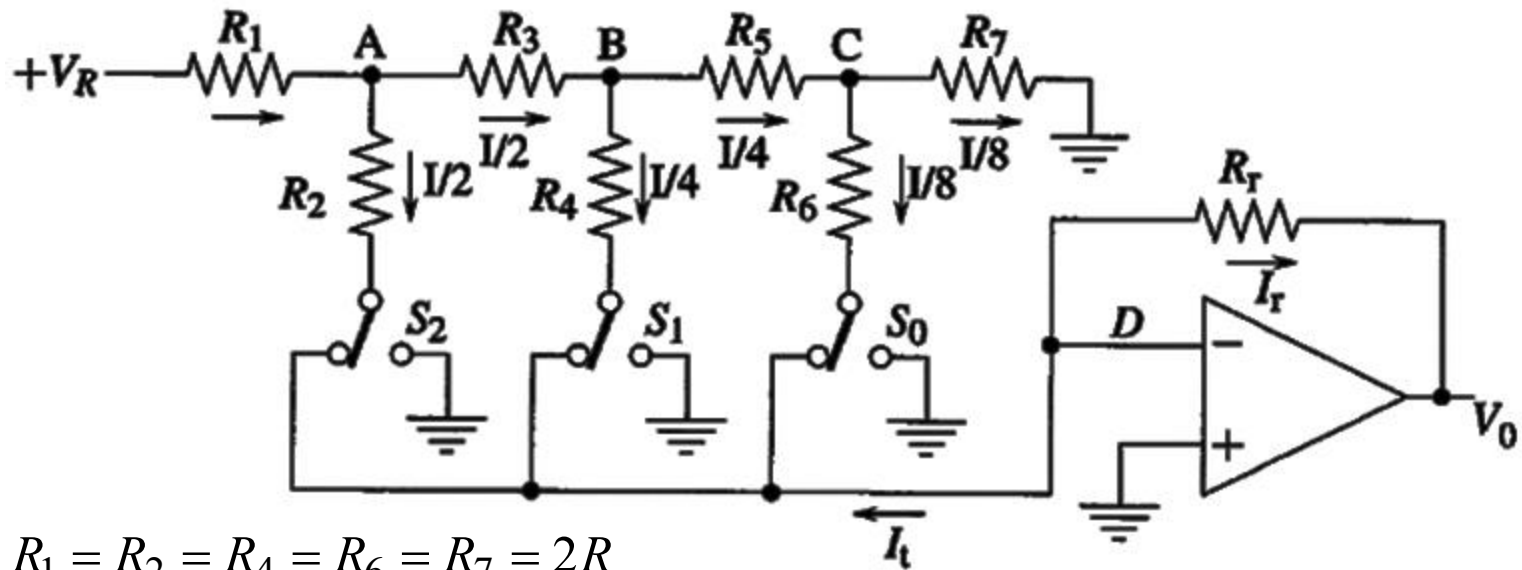
- Soluzione poco pratica per conversioni a molti bit → **eccessiva dispersione** dei valori delle resistenze → irrealizzabile sia in forma discreta che integrata
- Impedenza di ingresso al convertitore variabile → oscillazioni alim.



uscite analogiche

Conversione D/A a scala invertita

- Soluzioni con *reti a scala*
 - Basati solo su due valori di resistenza R e $2R$
- **Convertitore a scala invertita**



$$R_1 = R_2 = R_4 = R_6 = R_7 = 2R$$

$$R_3 = R_5 = R$$



uscite analogiche

Conversione D/A a scala invertita

- Converte il numero binario ($S_2S_1S_0$) in

$$V_0 = R_r i_t = -\frac{V_R}{24} \frac{R_r}{R} (2^2 S_2 + 2S_1 + S_0)$$

- Si noti che la corrente che entra nei nodi A , B e C vede due impedenze verso massa uguali, qualsiasi sia la posizione dei commutatori, e pari a $2R$, essendo il nodo D sull'ingresso invertente dell'operazionale una massa virtuale, per cui si divide in due parti uguali
- Ogni commutazione a 1 devia quindi verso il ramo di uscita una corrente di peso proporzionale al proprio
- La somma di tutte le correnti che fluiscono in R_r fornisce la tensione di uscita



uscite analogiche

Conversione D/A a scala

- L'impedenza di ingresso al convertitore è costante e pari a $3R$, eliminando oscillazioni della tensione di alimentazione
- Convertitori a scala sono usati per risoluzioni fino a 10 o 12 bit, mentre per risoluzioni più elevate si usano **tecniche miste**
 - tensione di riferimento generata con un convertitore a resistenze pesate, pilotato dai bit più significativi
- Sono diffusi convertitori dotati di amplificatore di uscita e di generatore di riferimento già integrati
- Alcune soluzioni presentano anche un registro in grado di memorizzare il valore digitale di comando



ingressi e uscite digitali

introduzione

- I **segnali digitali** (logici, booleani, discreti) sono i più diffusi nei sistemi di controllo
 - Normalmente sono forniti da interruttori meccanici o elettronici
- Segnale sotto forma di
 - *Informazione ohmica* (stato aperto/chiuso)
 - *Livello di tensione*
- Se necessario, il segnale deve essere tradotto in due livelli di tensione compresi in due bande $\rightarrow 0/1$



ingressi e uscite digitali

introduzione

- Sono inoltre necessarie operazioni di *filtraggio* e di *isolamento galvanico*
- Si dividono in due categorie principali:
 - *Segnali di stato*
 - *Segnali da contare*
- **Uscite** fornite come due possibili livelli di tensione
- Uscite realizzate con
 - **relay** elettromeccanici
 - **switch** allo stato solido
 - Transistor, SCR (*Silicon Controlled Rectifier*), TRIAC

ingressi e uscite digitali

introduzione



- **SCR** sono dispositivi allo stato solido che agiscono come interruttori molto veloci
 - Impiegati per la modulazione di correnti elevate
 - Si controlla solo l'accensione
 - Lo spegnimento interviene quando la tensione alternata che lo pilota passa per lo zero
- **TRIAC** sono simili agli SCR ma offrono la possibilità di comandare anche l'istante di spegnimento

ingressi e uscite digitali

Condizionamento



- **Condizionamento degli ingressi digitali**
 - *Adattamento*
 - *Isolamento galvanico*
- **Adattamento** di segnali digitali provenienti da interruttori alimentati con tensioni alto livello si realizza con partitori di tensione
 - Se l'interruttore non è alimentato (si dice “pulito”) si dovrà aggiungere una tensione per l'alimentazione del contatto
 - Nei contatti meccanici è necessario far scorrere correnti almeno di 20mA per impedirne l'ossidazione

ingressi e uscite digitali

Condizionamento

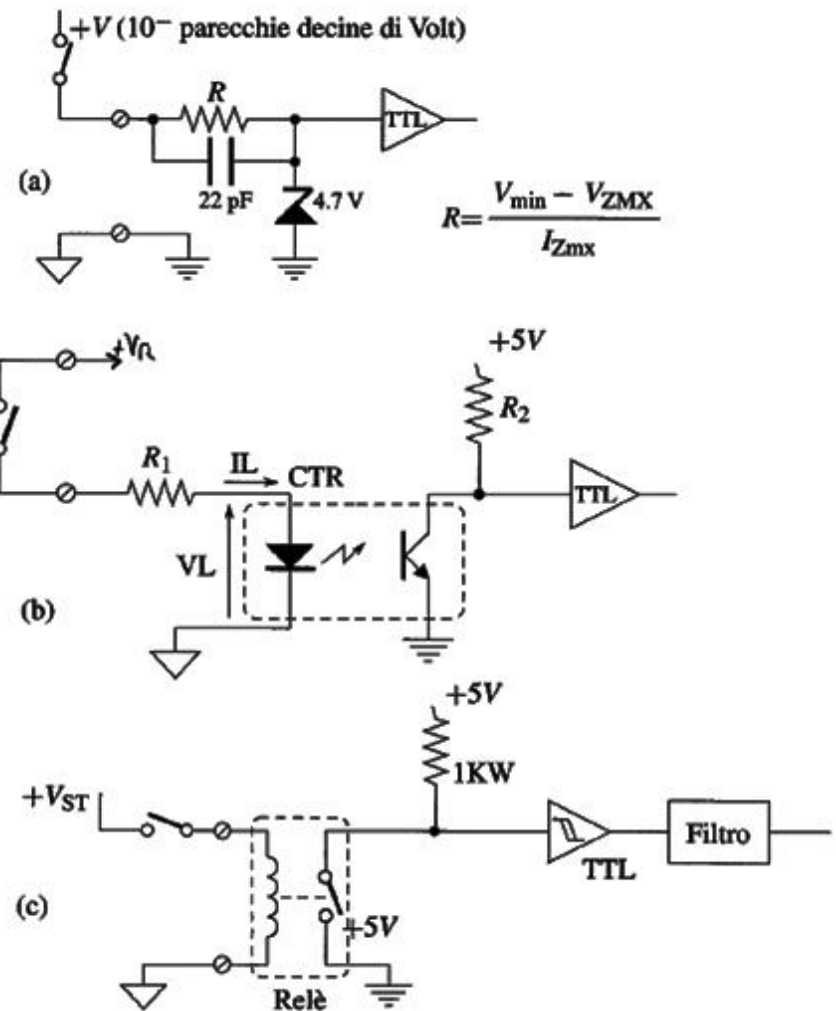


- Elevato livello di tensione di alimentazione per
 - Aumentare il rapporto segnale/rumore
 - Compensare eventuali cadute di tensione sui cavi
- Successivamente si deve ridurlo per renderlo compatibile con i circuiti di acquisizione

ingressi e uscite digitali

Condizionamento

- *Esempi con ingressi TTL*
 - a) Adattamento di un contatto alimentato
 - b) Alimentazione, adattamento e isolamento galvanico a fotoaccoppiatore di un contatto "pulito"
 - c) Adattamento e isolamento galvanico a relay di un contatto alimentato



ingressi e uscite digitali

Condizionamento



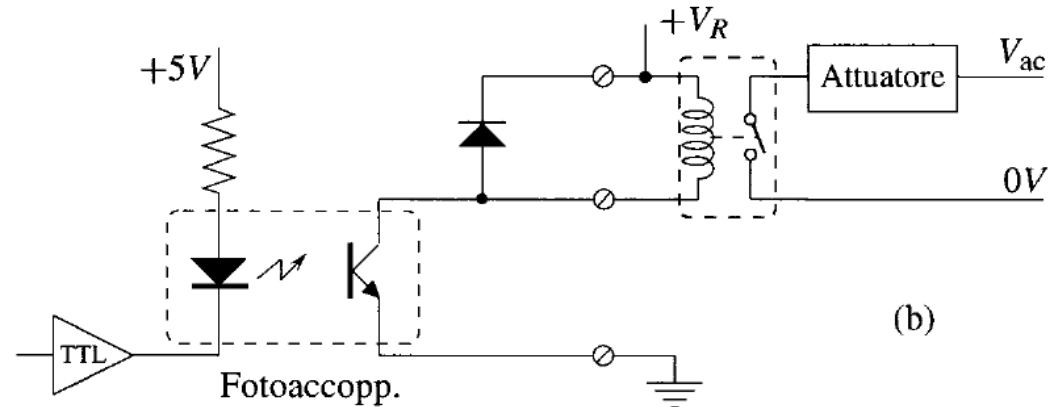
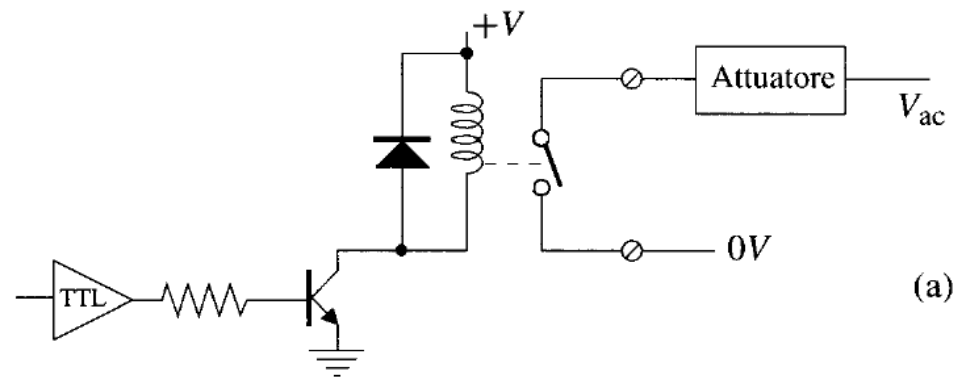
- **Isolamento galvanico** utile e spesso necessario per motivi di sicurezza

- Contatto accidentale con tensioni elevate
- Scariche accidentali

- *Esempi*

a) Isolamento a relay

b) Doppio isolamento a fotoaccoppiatore e relay



protezioni

Generalità



- **Protezioni** sono necessarie su tutti i punti di collegamento di un sistema di acquisizione
 - Tensioni elevate (*surge*) possono giungere sugli ingressi e sulle uscite danneggiando i relativi circuiti elettronici
 - Cause disparate:
 - Fulmini
 - Campi magnetici molto intensi
 - Scariche di elettricità statica
 - Contatti accidentali con linee ad alta tensione
 - Errori di connessione!

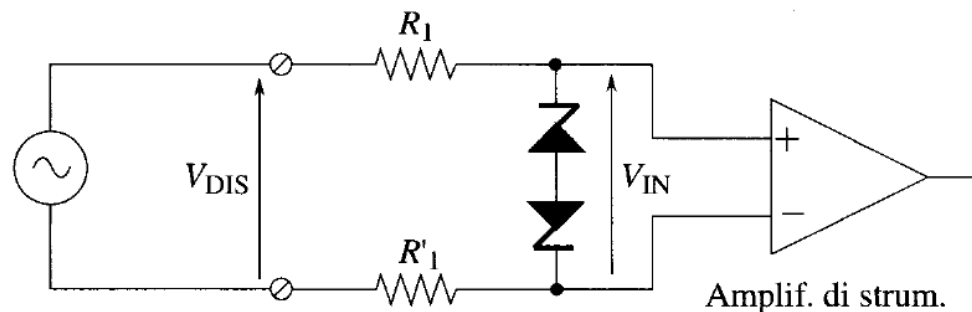
protezioni

Generalità

- *Esempi* basati su
 - Soppressori di transienti (varistori): b)
 - Diodi Zener: a)

- **Soppressori**

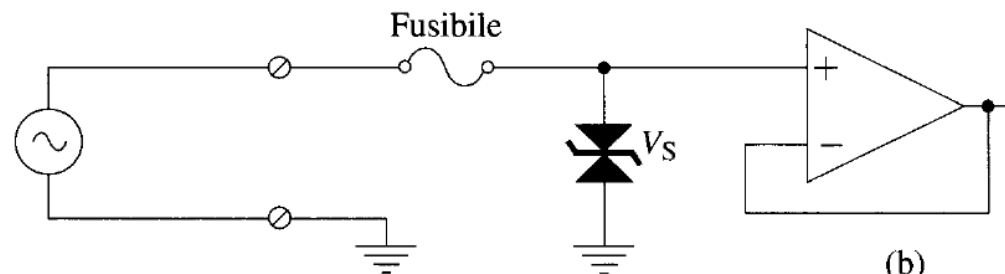
- Utili per protezione di tensioni impulsive
- Dimensionati in base al livello di energia da dissipare
- Veloci e poco costosi



$$V_{INMax} \ll V_{INMaxIA}$$

$$V_{INMax} = V_{Zener} + V_{direttazener} \cong V_{Zener} + 0.7$$

(a)



(b)

protezioni

Generalità



- **Diodi Zener** hanno comportamenti simili ai varistori ma possono dissipare energie inferiori
 - Con riferimento allo schema a), le due resistenze di egual valore limitano le correnti negli Zener
 - La potenza dissipabile dagli Zener e dalle resistenze determina la massima energia sopportabile senza rotture
- Sia i diodi Zener che i varistori esibiscono *correnti di fuga* che, se pur molto basse, vanno considerate quando si trattano segnali a basso livello

protezioni

Generalità



- **Fusibili** forniscono protezioni a basso costo
 - Devono essere ripristinati dopo il loro intervento → tempi lunghi rispetto al ripristino automatico
- Le protezioni si possono avvalere anche di **condensatori di tipo antinduttivo**
 - Usati per tagliare i disturbi ad alta frequenza
 - Possono essere usati sia su segnali analogici che digitali



attuatori per il controllo dei flussi

Introduzione

- **Modulazione del flusso** (portata) di fluidi liquidi (gassosi) in circuiti idraulici (pneumatici) diffusa nei processi industriali (chimici, termici, petroliferi, ...)
 - *Esempio*: regolazione del livello (di liquido) o di pressione (di gas) in serbatoi di accumulo o di processo, ...
- La modulazione viene attuata mediante:
 - **Valvole di regolazione**
 - **Pompe**
- La scelta e il dimensionamento delle valvole di regolazione influenzano le prestazioni del controllo

attuatori per il controllo dei flussi

Circuiti idraulici

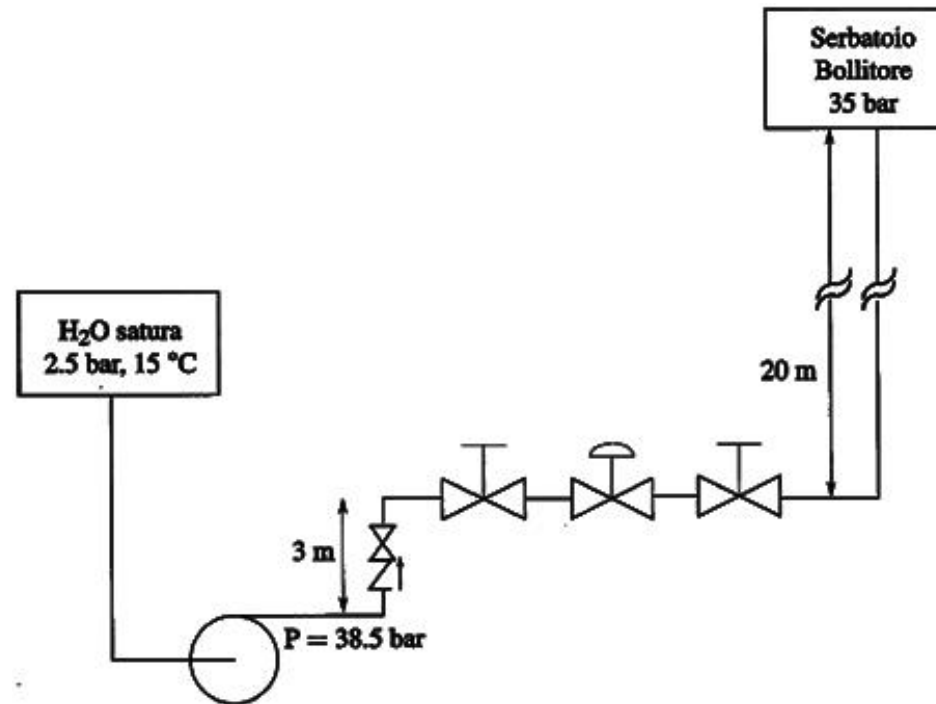


- *Esempio di circuito idraulico*

- Prelievo di acqua a bassa pressione da un serbatoio di accumulo per l'invio in un serbatoio bollitore ad elevata pressione

- Elementi del circuito:

- Tratti di tubazione
- Pompa
- Valvole (1 di non ritorno, 2 di isolamento, 1 di regolazione)





attuatori per il controllo dei flussi

Circuiti idraulici

- La **pompa** ha la funzione di innalzare la pressione del fluido ad un valore maggiore di quello del bollitore
- La **valvola di non ritorno** interviene ad impedire l'inversione del moto del fluido in caso di problemi alla pompa o di sovrappressione nel bollitore
- Le **valvole di isolamento** vengono chiuse manualmente quando si deve intervenire sulla valvola di regolazione (es. per la manutenzione)
- La **valvola di regolazione** è utilizzata per la *modulazione automatica* della portata del fluido



attuatori per il controllo dei flussi

Circuiti idraulici

- Tutte le valvole ed i tratti di tubazione, in particolare quelli a sezione ridotta e le curve, rappresentano delle **resistenze** al moto del fluido
 - La valvola di regolazione consente di modulare la portata del fluido variando la sezione di passaggio → agisce come una **resistenza variabile**



attuatori per il controllo dei flussi

Circuiti idraulici

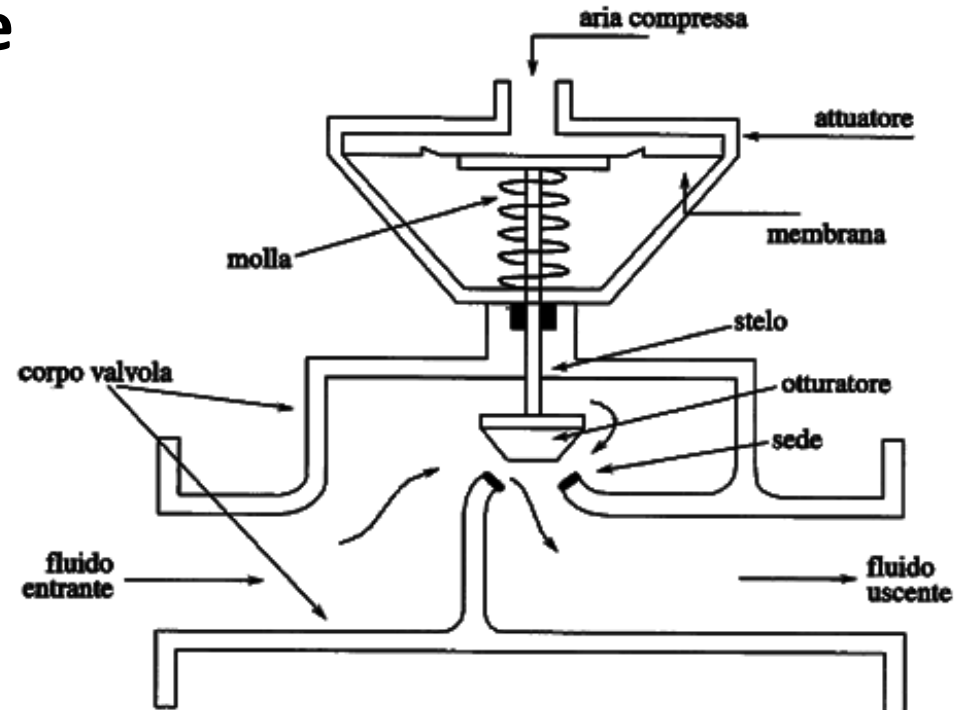
- Il moto di un fluido in un circuito può essere studiato con le **equazioni di Bernoulli**
 - Conservazione del carico idrostatico totale in un condotto (somma delle energie cinetica, potenziale e di pressione, al variare della quota lungo il condotto, a meno delle perdite)
 - La forza di gravità in presenza di dislivelli o differenze di pressione può determinare il moto del fluido → **pompe forniscono il carico idrostatico** per compensare:
 - Le perdite di carico nelle tubazioni e nei raccordi
 - Le perdite di carico nelle valvole
 - Dislivelli e salti di pressione sfavorevole

valvole di regolazione

Generalità

- Le **valvole di regolazione** consentono di *modulare con continuità* la portata di fluido nella tubazione in cui sono inseriti
- *Esempio*: valvola a globo con attuatore pneumatico

- Modulando la pressione dell'aria inviata all'attuatore si modifica la posizione dello stelo → varia la sezione di passaggio del fluido tra l'otturatore e la sua sede





valvole di regolazione

Generalità

- La variazione di sezione utile al passaggio di fluido tra l'otturatore e la sua sede provoca una **variazione delle perdite di carico** e quindi della portata transitante
- L'**attuatore** ha il compito di posizionare l'otturatore in funzione del segnale proveniente dal regolatore
 - Vanno scelti e dimensionati a parte
 - Sono impiegati tipi di attuatori:
 - Pneumatici
 - Oleodinamici
 - Elettromeccanici

valvole di regolazione

Definizioni



- **Otturatore**

- È un corpo mobile opportunamente sagomato, la cui posizione rispetto ad una apertura fissa determina l'area della sezione di passaggio del fluido transitante
- Può assumere qualunque posizione tra due estremi:
 - *Posizione di chiusura*: minima portata nella valvola
 - *Posizione di completa apertura*: massima portata nella valvola

- **Chiusura completa**

- Una valvola si definisce completamente chiusa quando con l'otturatore in posizione di chiusura la portata in essa transitante è nulla

valvole di regolazione

Definizioni



- **Corsa**
 - Spostamento dell'otturatore dalla posizione di chiusura
- **Corsa nominale**
 - Corsa corrispondente alla posizione di completa apertura
 - Coincide con la corsa massima
- **Corsa relativa (h)**
 - Rapporto tra la corsa e la corsa nominale
 - $h = 0 \rightarrow$ posizione di chiusura
 - $h = 1 \rightarrow$ posizione di completa apertura

valvole di regolazione

Definizioni



- **Trim**
 - Indica l'insieme degli organi interni della valvola, tra cui l'otturatore e la sua sede, che sono in diretto contatto con il fluido in essa transitante
 - Con la sua forma determina le caratteristiche di efflusso della valvola al variare della corsa
- **Diametro nominale (DN)**
 - È in numero che identifica la dimensione dell'interfaccia di una valvola e degli altri elementi di un circuito idraulico
 - Elementi con gli stessi DN e PN (pressione nominale) sono direttamente accoppiabili



valvole di regolazione

Cenni alla classificazione delle valvole

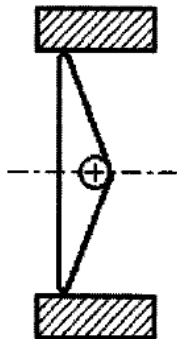
- *In base al tipo di movimento dell'otturatore:*
 - Valvole lineari
 - L'otturatore si sposta linearmente, traslando lungo un asse
 - La corsa è espressa in millimetri
 - Valvole rotative
 - L'otturatore ruota attorno ad un asse
 - La corsa è espressa in gradi
- *In base alla forma dell'involucro esterno (corpo):*
 - A globo, ad angolo, toroidale, a tre vie, split body, ecc.
- *In base alla forma dell'otturatore:*
 - A tappo, a pistone, a ghigliottina, a farfalla, a sfera, ecc.

valvole di regolazione

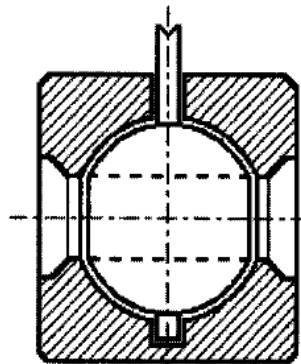
Cenni alla classificazione delle valvole



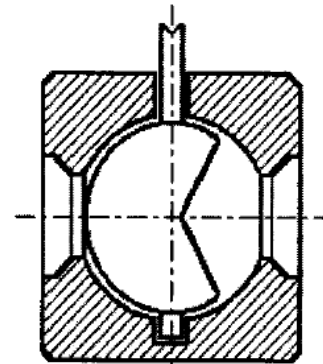
- Nel seguito si considereranno:
 - a) Valvole rotative con otturatore a disco eccentrico (a farfalla)
 - b) Valvole rotative con otturatore a sfera
 - c) Valvole rotative con otturatore a sfera con incavo a V



(a)



(b)



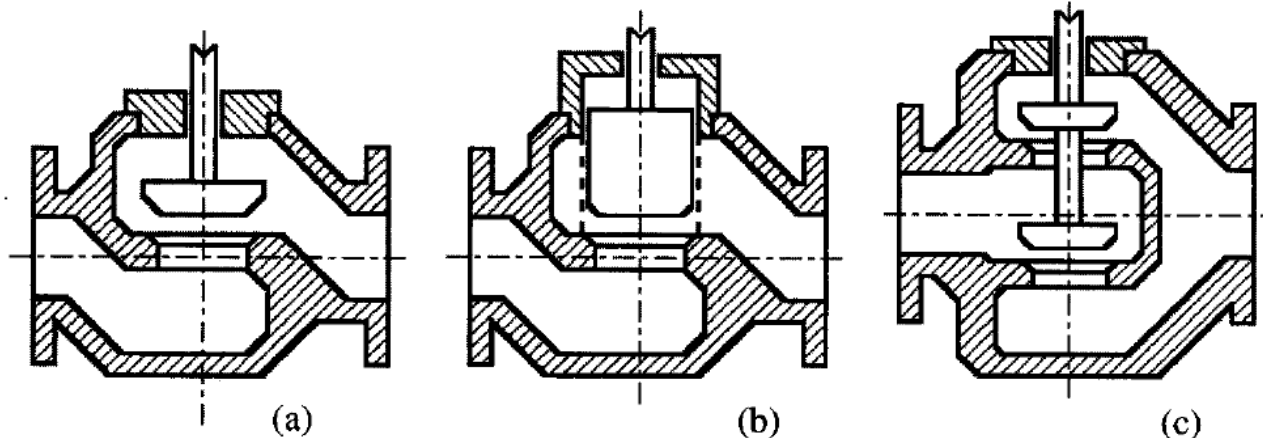
(c)

valvole di regolazione

Cenni alla classificazione delle valvole



- a) Valvole lineari a globo con *port** singola, con otturatore a tappo con profilo parabolico o a pistone
- b) Valvole lineari a globo con *port* singola, con otturatore a pistone guidato in una gabbia singola
- c) Valvola lineare a globo con *port* doppia, con otturatore a tappo con profilo parabolico oppure di tipo V-port



***Port:**
*l'apertura fissa
(alloggiamento
dell'otturatore)
nelle valvole
lineari*



valvole di regolazione

Caratteristiche intrinseche e coefficiente di efflusso C_v

- La portata di fluido transitante in una valvola dipende
 - **Pressione di ingresso** o a monte p_1
 - **Temperatura di ingresso** o a monte T_1
 - Si suppone trascurabile la variazione di temperatura all'interno della valvola $\rightarrow T_1 = T_2$
 - **Pressione di uscita** o a valle p_2
 - Pressione differenziale o **caduta di pressione** $\Delta p = p_1 - p_2 \geq 0$
 - **Pressione di vapore** del fluido p_v alla temperatura T_1
 - **Densità relativa** del fluido $G_f = \rho/\rho_0$, essendo ρ la densità del fluido nelle condizioni a monte e $\rho_0 = 1000 \text{ kg/m}^3$ la densità dell'acqua alle condizioni ambiente



valvole di regolazione

Caratteristiche intrinseche e coefficiente di efflusso C_v

- Il **coefficiente di efflusso** (o di portata) C_v , definisce le caratteristiche di portata proprie di una valvola, ovvero alle condizioni di riferimento

Definizione: Il C_v è la **portata volumetrica**, espressa in gpm (galloni al minuto, 1 gallone è pari a 3.8 litri), di acqua a temperatura compresa nell'intervallo $5\div 40^\circ\text{C}$ ($G_f = 1$) transitante attraverso la valvola quando ad essa è applicata una caduta di pressione statica $\Delta p = 1 \text{ psi} = 6.895 \text{ Pa}$



valvole di regolazione

Caratteristiche intrinseche e coefficiente di efflusso C_v

- Il C_v dipende
 - dalle **dimensioni** della valvola
 - *Esempio*: valvola da 1 pollice $\rightarrow C_v = 15$; 6 pollici $\rightarrow C_v = 400$
 - dalla **struttura interna** della valvola
 - Determinata sperimentalmente
 - dalla forma dei **tratti di condotta** nelle vicinanze della valvola
 - Cambi di sezione o di direzione nelle tubazioni nelle immediate vicinanze possono modificare i profili di velocità, e quindi di portata, nella valvola in maniera non facilmente prevedibile
 - Nelle rilevazioni sperimentali del C_v si predispongono a monte e a valle tratti di tubazione rettilinei 20 e 7 volte il diametro nominale



valvole di regolazione

Caratteristiche intrinseche e coefficiente di efflusso C_v

- Nelle installazioni si dovranno prevedere tratti di tubazione rettilinea a monte ed a valle almeno 6 e 3 volte il diametro nominale della valvola, affinché il C_v non si discosti troppo da quello nominale
- Il C_v è una **funzione monotona non decrescente** della corsa h dell'otturatore
 - $C_{v\ min} \rightarrow h = 0$ (posizione di chiusura)
 - $C_{v\ max} \rightarrow h = 1$ (posizione di completa apertura)



valvole di regolazione

Caratteristiche intrinseche e coefficiente di efflusso C_v

- Definizioni utili per definire il legame tra C_v ed h :
 - **Coefficiente di efflusso nominale** C_{vn} : valore di C_v alla corsa nominale ($C_{vn} = C_{v\ max}$)
 - **Coefficiente di efflusso relativo** $\phi = C_v / C_{vn}$
 - **Apertura**: valore di ϕ espresso in percentuale
 - **Caratteristica intrinseca (o inerente) di portata** $\phi(h)$: relazione che esprime il coefficiente di efflusso relativo ϕ in funzione della corsa relativa h , con Δp costante
 - **Rangeability intrinseca** $r = C_{v\ max} / C_{v\ min}$ ovvero è il rapporto tra il valore massimo ed il valore minimo del coefficiente di efflusso



valvole di regolazione

Caratteristiche intrinseche e coefficiente di efflusso C_v

- La **forma delle caratteristiche intrinseche** può essere modellata sagomando opportunamente l'otturatore
 - Il costruttore è tenuto (normativa IEC 534-2-4) a fornire la descrizione della caratteristica intrinseca della valvola in forma grafica oppure in forma tabellare
- Le caratteristiche di uso più comune sono:
 - Caratteristica *lineare*
 - Caratteristica *esponenziale* (o *equipercentuale*)
 - Caratteristica *quadratica* (o *parabolica*)
 - Caratteristica *ad apertura rapida* (o *quick opening*)

valvole di regolazione

Caratteristiche intrinseche e coefficiente di efflusso C_v



a) Caratteristica lineare

$$\phi = \phi_0 + \alpha h$$

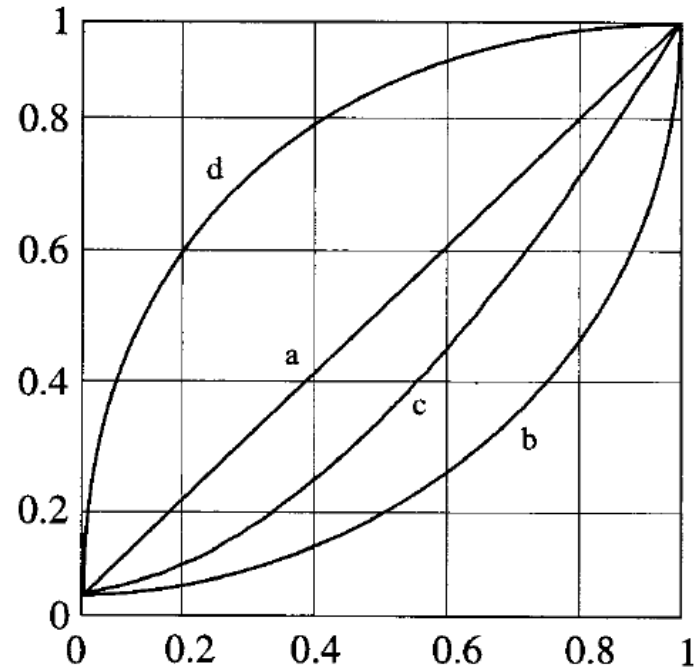
dove ϕ_0 e α sono costanti

b) Caratteristica esponenziale o equipercentuale

$$\phi = \phi_0 e^{\beta h}$$

dove ϕ_0 e β sono costanti

- *Nota:* non è possibile realizzare una caratteristica equipercentuale fino alla chiusura completa





valvole di regolazione

Caratteristiche intrinseche e coefficiente di efflusso C_v

- Guadagno della valvola è proporzionale all'efflusso relativo

$$\frac{d\phi}{dh} = \beta\phi$$

per cui ad eguali incrementi della corsa h corrispondono incrementi equipercettuali di ϕ

c) **Caratteristica quadratica o parabolica**

$$\phi = \phi_0 + \gamma h^2$$

dove ϕ_0 e γ sono costanti

d) **Caratteristica ad apertura rapida (quick opening)**

- È caratterizzata da una curva a derivata decrescente



valvole di regolazione

Caratteristiche intrinseche e coefficiente di efflusso C_v

- Costanti espresse in termini della rangeability

- Osservando che per

- per $h = 0$, $\phi(0) = C_{v \min} / C_{vn} = C_{v \min} / C_{v \max} = 1/r$

- per $h = 1$, $\phi(1) = C_{v \max} / C_{vn} = 1$

- si ottengono le seguenti espressioni:

- Caratteristica lineare

$$\phi = h + \frac{1}{r}(1 - h)$$

- Caratteristica equipercentuale

$$\phi = r^{h-1}$$



valvole di regolazione

Caratteristiche intrinseche e coefficiente di efflusso C_v

- Caratteristica parabolica

$$\phi = h^2 + \frac{1}{r}(1 - h^2)$$

- La **scelta del tipo di caratteristica** dipende dal legame tra la portata e le variabili correlate al circuito in cui la valvola è inserita
 - In particolare la scelta della valvola dipende dal legame esistente tra la portata e la pressione a monte e la pressione a valle, ovvero dalla loro differenza, che dipendono dal circuito in cui la valvola è inserita



valvole di regolazione

Dipendenza della portata dal salto di pressione

- La dipendenza della portata transiente in una valvola ed il relativo salto di pressione monte-valle si può ricavare applicando l'equazione di Bernoulli
 - Si suppongono uguali le sezioni di ingresso ed uscita ($A_1=A_2=A$)
 - Si suppone trascurabile la differenza di quota ($z_1=z_2$)
 - Ipotesi di fluido incomprimibile $\rightarrow$ velocità uguali nelle sezioni di ingresso e uscita per la conservazione della massa ($v_1=v_2=v$)
 - Indicando con $\alpha(h)$ il coefficiente di proporzionalità delle perdite di carico con la corsa relativa, si ottiene

$$\frac{p_1 - p_2}{\rho} = \alpha(h) \frac{v^2}{2}$$



valvole di regolazione

Dipendenza della portata dal salto di pressione

- da cui si può ricavare la portata volumetrica $q = Av$

$$q = A \sqrt{\frac{2}{\alpha(h)}} \sqrt{\frac{1}{\rho_0}} \sqrt{\frac{p_1 - p_2}{G_f}}$$

dove $G_f = \rho/\rho_0$ e $\rho_0 = 1000 \text{ kg/m}^3$

- I valori di $\alpha(h)$ non sono noti, ma possono essere legati a quelli di $C_v(h) \rightarrow$ ponendo $\Delta p = p_1 - p_2 = 1 \text{ psi}$ e $G_f = 1$ nella relazione precedente allora, per definizione, la portata volumetrica q coincide con $C_v(h)$:

$$C_v(h) = A \sqrt{\frac{2}{\alpha(h)}} \frac{1}{\sqrt{\rho_0}}$$



valvole di regolazione

Dipendenza della portata dal salto di pressione

da cui si può ricavare

$$q = N_0 C_v(h) \sqrt{\frac{\Delta p}{G_f}}$$

dove $N_0 = (\text{psi})^{-0.5}$ è un coefficiente unitario di dimensionamento

- La relazione precedente è l'equazione di efflusso di una valvola in unità volumetriche secondo la normativa ISA per il dimensionamento delle valvole di regolazione
- La corrispondente equazione in unità massiche è

$$w = N_1 C_v(h) \sqrt{G_f \Delta p}$$

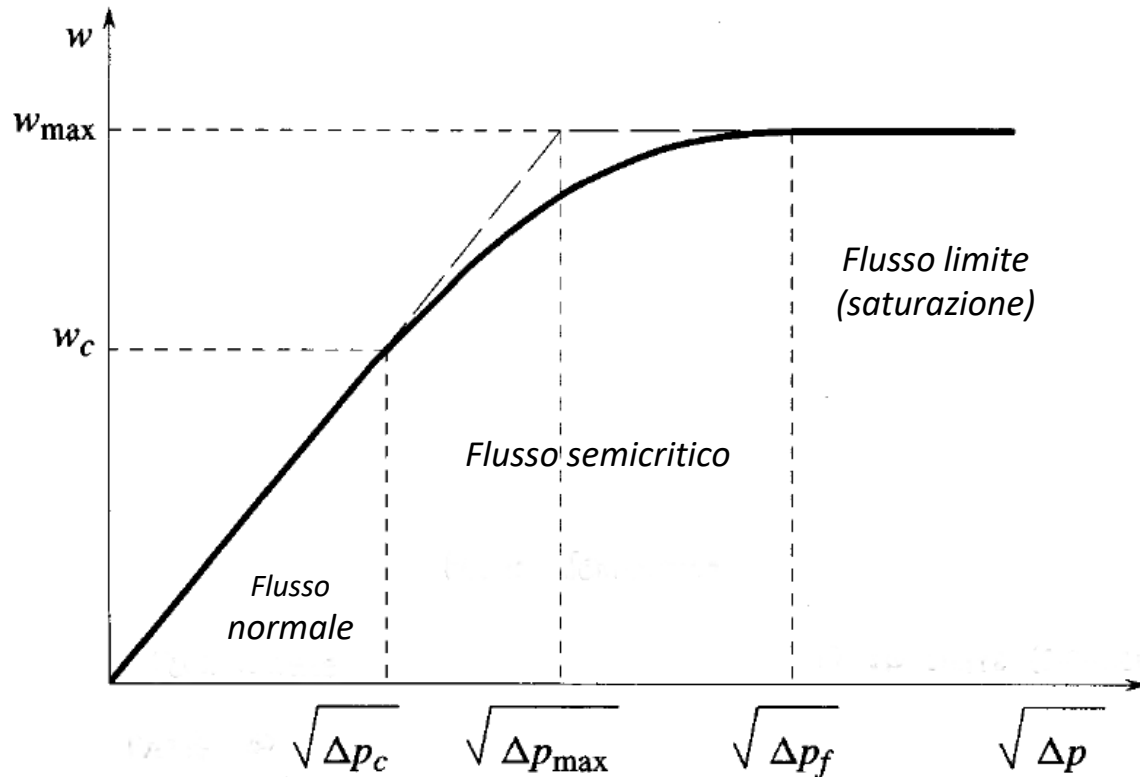
dove N_1 è un opportuna coefficiente dimensionale



valvole di regolazione

Dipendenza della portata dal salto di pressione

- Le espressioni ricavate della portata valgono per Δp inferiori ad un valore limite (*critico*):





valvole di regolazione

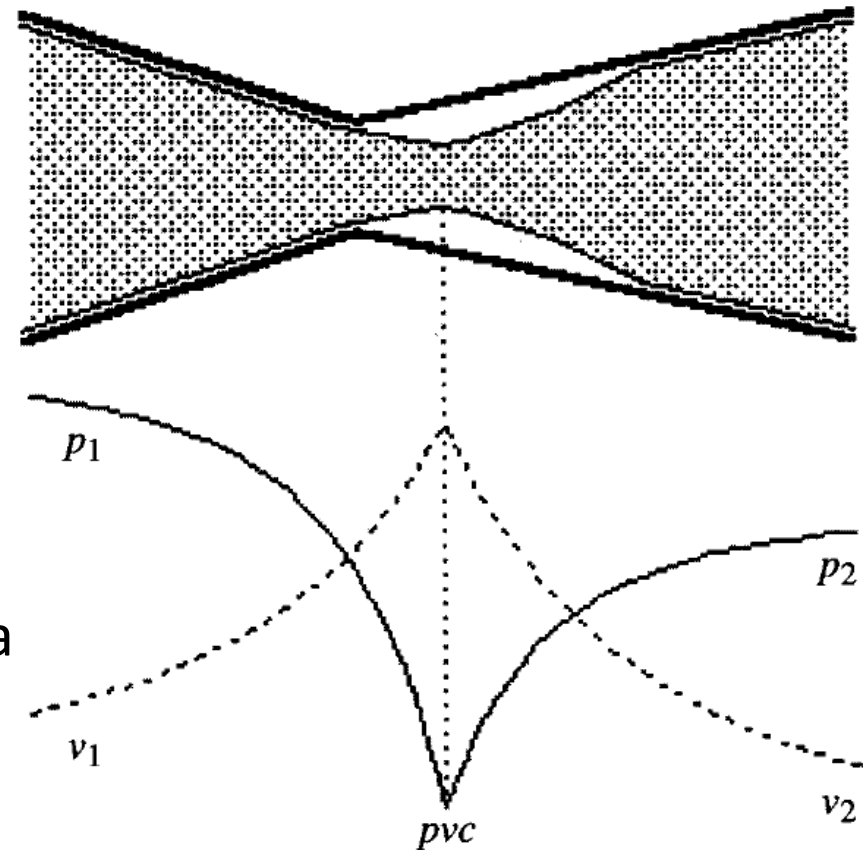
Dipendenza della portata dal salto di pressione

- L'andamento qualitativo della portata massica di un liquido transitante in una valvola in funzione di $\sqrt{\Delta p}$ riportato nel diagramma precedente è stato ottenuto nelle seguenti condizioni
 - Flusso nella valvola turbolento
 - Il liquido è un fluido newtoniano (viscosità funzione del solo stato fisico, es. pressione e temperatura)
 - La pressione a monte è mantenuta costante, mentre quella a valle è progressivamente diminuita
 - La corsa è mantenuta costante ($h = \text{cost}$)
- Si distinguono tre diverse zone di funzionamento
 - Una zona di *flusso normale*: $w \propto \sqrt{\Delta p}$
 - Una zona di *flusso semicritico*: $w < \alpha \sqrt{\Delta p}$
 - Una zona di *flusso limite (clogged flow)*, o di *saturazione*: $w = w_{\max}$

valvole di regolazione

Cavitazione e flashing

- Gli andamenti della seconda e terza zona possono essere spiegati introducendo i fenomeni della **cavitazione** e del **flashing**
 - Si consideri una valvola schematizzata come in figura
 - All'ingresso (*convergente*) il fluido è allo stato liquido e alla pressione p_1



valvole di regolazione

Cavitazione e flashing



- All'uscita (*divergente*) la pressione p_2 è sempre inferiore all'ingresso a causa delle perdite, in ipotesi di fluido incomprimibile e valvola orizzontale
- Lungo la convergente si riduce progressivamente la sezione → aumenta la velocità → si riduce la pressione (← Bernoulli)
- Lungo la divergente accade l'opposto
- La sezione di area minima della valvola (*strozzatura*) non corrisponde alla sezione dove è raggiunta la minima pressione, perché la vena fluida continua a contrarsi fino ad assumere la sezione minima (detta di ***vena contratta***) all'interno della zona divergente

valvole di regolazione

Cavitazione e flashing



- Nella sezione di vena contratta la pressione p_{vc} (*pressione di vena contratta*) è minima e la velocità è massima
- **Valvola reale**: l'equivalente della strozzatura è la sezione di passaggio del fluido compresa tra otturatore e sede
 - La sezione di passaggio dipende dalla corsa → anche i profili di pressione e velocità dipendono dalla corsa
- Se la pressione di vena contratta p_{vc} diventa inferiore alla pressione di vapore p_v all'interno della valvola si formano **bolle di vapore** → si presenterà il fenomeno della **cavitazione** oppure quello del **flashing**, a seconda del valore che la pressione p_2 assume rispetto a p_v :

valvole di regolazione

Cavitazione e flashing



– Se $p_2 > p_v \rightarrow$ **cavitazione**

- Le bolle di vapore collassano non appena vengono trasportate dalla corrente del fluido in un punto a pressione superiore a p_v
- Il ripetuto collasso delle bolle produce **onde di pressione** di intensità molto elevata accompagnata da **forte rumorosità**

– Se $p_2 \leq p_v \rightarrow$ **flashing**

- Le bolle di vapore raggiungono l'uscita della valvola, dove può essere presente una **miscela di liquido-vapore** oppure solo vapore
- Il flashing è accompagnato da **forte rumorosità**

– Entrambi questi fenomeni provocano **usura e danneggiamento** delle superfici metalliche a contatto con il fluido



valvole di regolazione

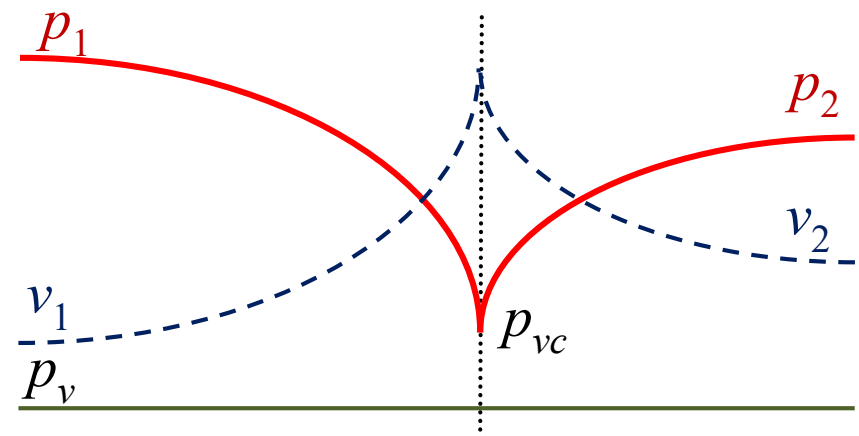
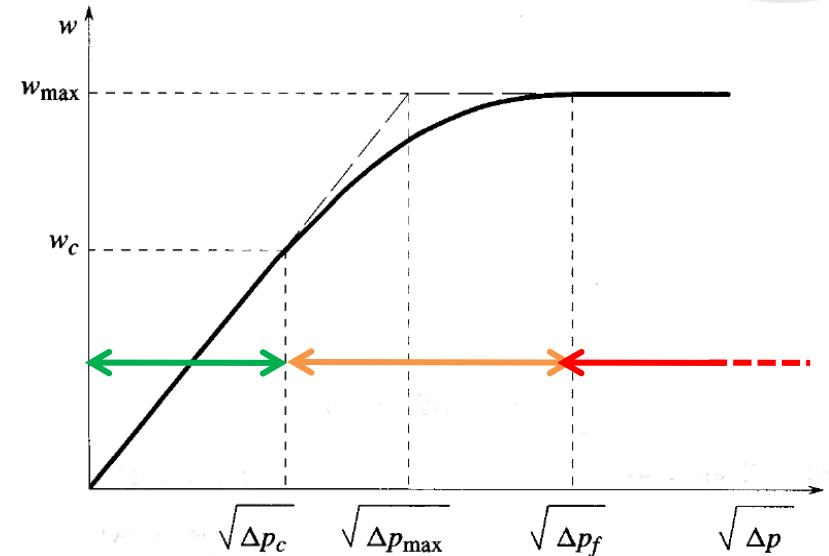
Cavitazione e flashing

- La **cavitazione deve essere evitata** per salvaguardare la valvola
 - *Trim anticavitazione*: valvole speciali studiate per ridurre gli effetti la rumorosità e gli effetti dannosi della cavitazione e del flashing
- Il **flashing non può essere evitato**: dipende dal fluido (p_v), dalle sue condizioni di ingresso alla valvola e dalla pressione di uscita, imposta dal circuito a valle
- Gli effetti negativi del flashing (rumore e danneggiamento) aumentano con la velocità del fluido
 - Si adottano **trim induriti** e si limita la velocità del fluido a 3.5 m/s

valvole di regolazione

Cavitazione e flashing

- Giustificazione della caratteristica di efflusso reale:
 - **Zona di flusso normale**
 - $p_{vc} > p_v$
 - Cavitazione assente
 - Andamento della portata in accordo all'equazione di Bernoulli
 - $0 \leq \Delta p < \Delta p_c$, dove Δp_c è il valore di Δp per cui risulta $p_{vc} = p_v$



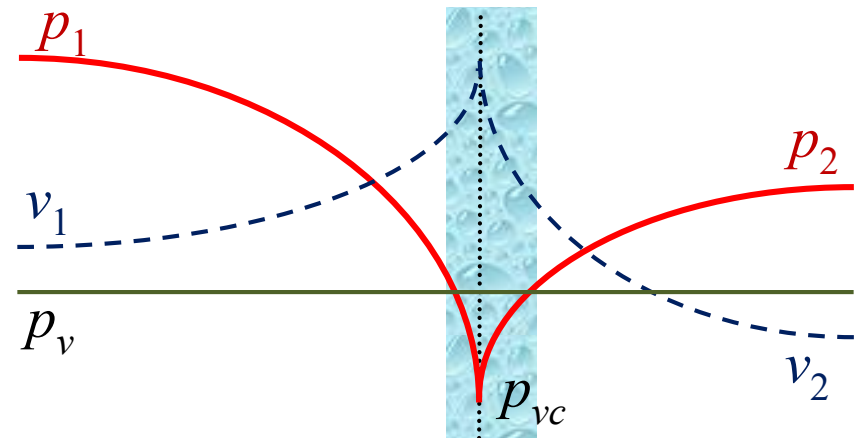
valvole di regolazione

Cavitazione e flashing



– Zona di flusso semicritico

- $p_{vc} \leq p_v$ e $p_2 > p_v$
- Cavitazione presente → evaporazione del fluido nella valvola → aumento della velocità del fluido → aumento delle perdite di carico → riduzione della crescita della portata massica con Δp
- $\Delta p_c \leq \Delta p < \Delta p_f$, dove $\Delta p_f = p_1 - p_v$
- Quando in tutta la sezione di vena contratta si ha vapore → la portata raggiunge w_{max} → velocità pari alla *velocità del suono* nel fluido stesso (v_{max})
- La p_{vc} non è più legata a p_2 ma dipende solo da p_v

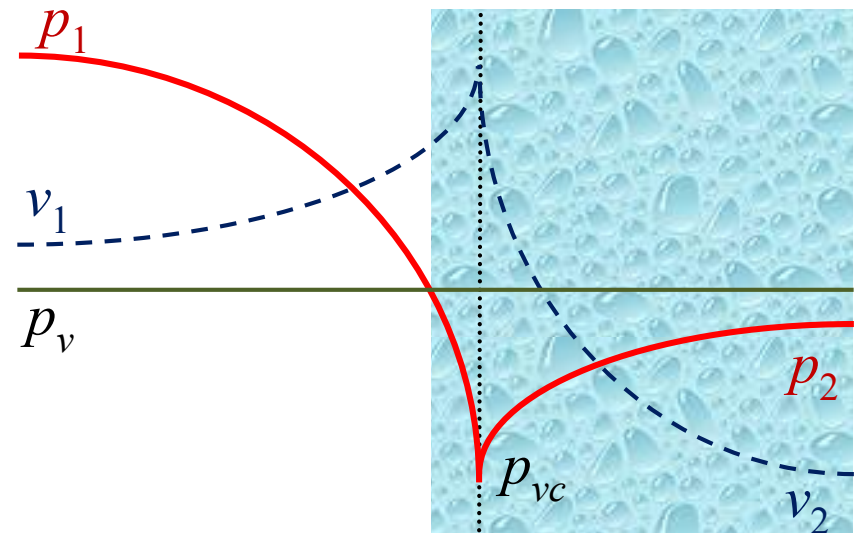


valvole di regolazione

Cavitazione e flashing

– Zona di flusso limite

- $p_{vc} < p_v$ e $p_2 \leq p_v$ ($\Delta p > \Delta p_f$)
- Se cresce Δp (diminuisce p_2) $\rightarrow$ velocità nella sezione di vena contratta rimane costante $\rightarrow$ portata costante ($w = w_{max}$) $\rightarrow$ in uscita dalla valvola si ha una miscela liquido-vapore con *titolo di vapore crescente* al diminuire di p_2





valvole di regolazione

Scelta e dimensionamento

- La **scelta di una valvola di regolazione** necessita di considerare diversi elementi per conseguire una buona combinazione di caratteristiche funzionali, di affidabilità, durata e costo
 - Vanno, ad esempio, considerare:
 - Condizioni di pressione e temperatura
 - Proprietà corrosive ed erosive del fluido
 - Tipo e capacità (taglia)
 - **La caratteristica di portata (intrinseca)**
 - Tipo e taglia dell'attuatore
 - Sicurezza e interfacciamento con il sistema di controllo
- } → materiale costruttivo



valvole di regolazione

Scelta della caratteristica intrinseca

- **Caratteristica installata:**
 - Le valvole sono usate per
 - *Regolare portate*
 - *Regolare livelli*
 - *Regolare temperature*
 - *Regolare pressioni*
 - La portata transitante in un valvola dipende dalla **corsa dello stelo h** e dal **salto di pressione monte-valle Δp**
 - *Quando la valvola viene inserita in un circuito, una variazione della portata determina una variazione del salto di pressione a causa delle variazioni di perdite nel circuito (valvola esclusa)*

valvole di regolazione

Scelta della caratteristica intrinseca



- ***Caratteristica installata***: relazione tra corsa e portata transitante in una valvola inserita in un circuito $\rightarrow w(h)$
 - Dipendenza del salto di pressione dalla corsa $\rightarrow$ la **caratteristica installata differisce dalla caratteristica intrinseca**
- **La scelta della caratteristica intrinseca della valvola deve essere operata in modo che la caratteristica installata abbia l'andamento desiderato**
 - In particolare, la caratteristica installata deve essere scelta con riferimento alle specifiche richieste dal sistema di controllo

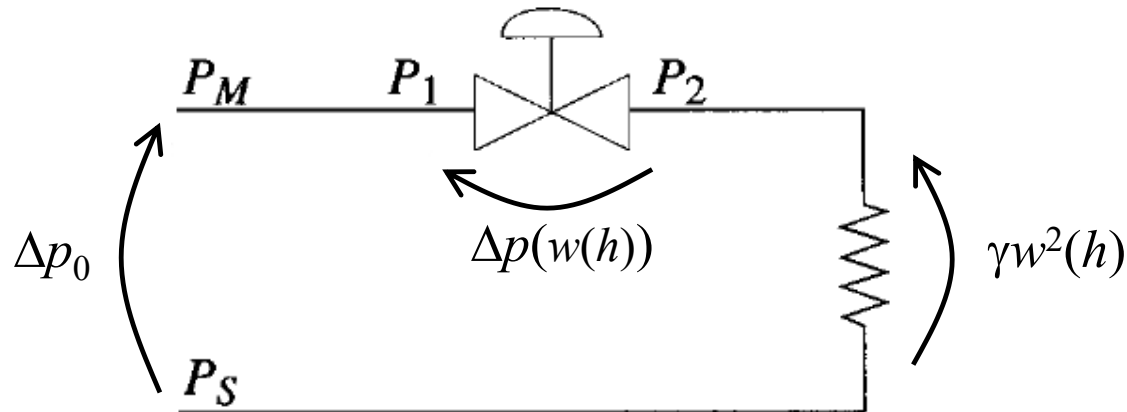


valvole di regolazione

Scelta della caratteristica intrinseca

- **Regolazione di portata**

- **Obiettivo: caratteristica istallata il più possibile lineare**
→ migliori prestazioni del sistema di controllo
- *Circuito di riferimento:*





valvole di regolazione

Scelta della caratteristica intrinseca

costituito da

- Una **valvola di regolazione**
 - Un **serpentino** (es. scambiatore di calore), in cui si suppongono concentrate tutte le perdite di carico del circuito
 - Una **pompa**, che fornisce il salto di pressione $\Delta p_0 = p_M - p_S = \text{cost}$, non raffigurata e supposta **ideale** → le perdite di carico della pompa si considerano parte di quelle del circuito
- Si evidenzia il **ruolo cruciale della caduta di pressione** sulla valvola in rapporto alla caduta totale sul circuito:
- Da un lato *dovrebbe essere elevata*, per aumentare l'efficacia di modulazione della portata
 - Dall'altro *dovrebbe essere bassa*, per contenere i costi di pompaggio



valvole di regolazione

Scelta della caratteristica intrinseca

- Perdite di carico nelle tubazioni e negli elementi di collegamento e raccordo sono funzione del quadrato della velocità del fluido → in *ipotesi di fluido incomprimibile*, anche funzione del quadrato della portata massica:

$$p_2 = p_S + \gamma w(h)^2$$

$$\Delta p(w(h)) = p_1 - p_2(w(h)) = p_M - p_S - \gamma w(h)^2 = \Delta p_0 - \gamma w(h)^2$$

da cui si ricava che la **caratteristica installa** è la *funzione implicita*

$$w(h) = N_1 C_v(h) \sqrt{G_f \Delta p(w(h))}$$





valvole di regolazione

Scelta della caratteristica intrinseca

Se si desidera conseguire $w(h) = Kh$, con $0 \leq h \leq 1$, deve risultare

$$C_v(h) = \frac{Kh}{N_1 \sqrt{G_f (\Delta p_0 - \gamma K^2 h^2)}}$$

quindi la **caratteristica intrinseca desiderata** risulta

$$\phi = \frac{C_v(h)}{C_{vn}} = \frac{\sqrt{\Delta p_0 - \gamma K^2}}{\sqrt{\Delta p_0 - \gamma K^2 h^2}}$$

- La **caratteristica intrinseca che linearizza la caratteristica installata** dipende dal peso relativo delle perdite di carico sul circuito (o sulla valvola) rispetto al totale

valvole di regolazione

Scelta della caratteristica intrinseca



- Se le perdite di carico sul circuito sono piccole rispetto a quelle complessive ($\Delta p_0 \gg \gamma K^2$) la caratteristica più adatta è quella lineare
- Se il peso relativo delle perdite di carico sul circuito diventano rilevanti al crescere di h la caratteristica più adatta è quella a guadagno crescente con la corsa, ovvero quella equipercentuale



valvole di regolazione

Scelta della caratteristica intrinseca

- Discriminazione tra la caratteristica lineare e quella equipercentuale
 - Basata sul parametro $V = \Delta p_n / \Delta p_0$, che esprime il rapporto tra il salto di pressione sulla valvola alla corsa nominale e quello totale sul circuito

Osservando che il coefficiente di proporzionalità delle perdite sul circuito si può esprimere come

$$\gamma = \frac{\Delta p_0 - \Delta p_n}{w_n^2}$$

dove w_n è la portata nominale

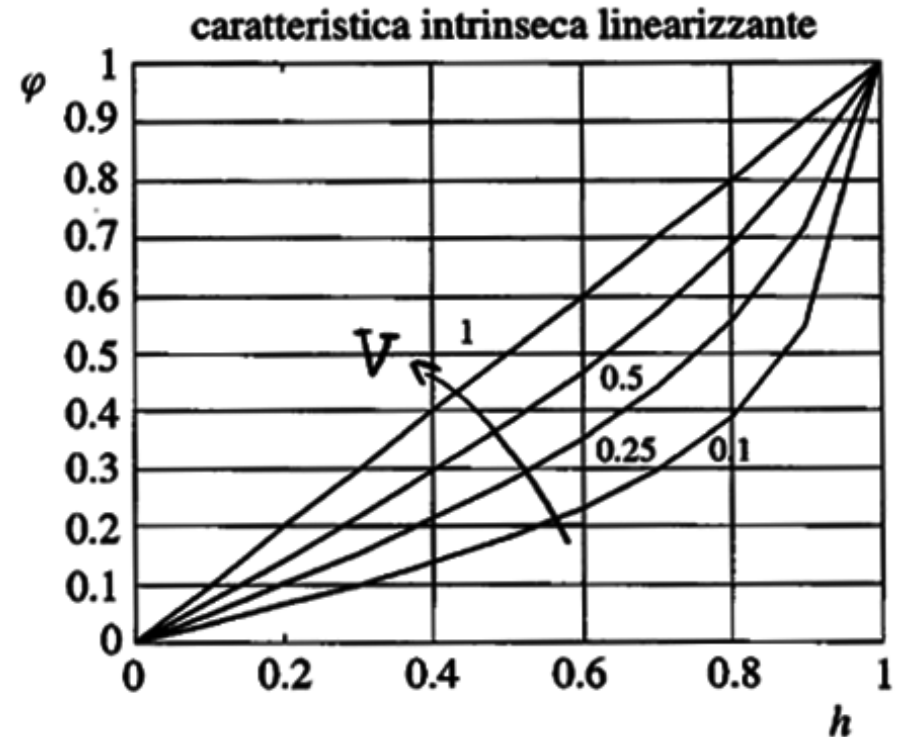
valvole di regolazione

Scelta della caratteristica intrinseca

La *caratteristica intrinseca* che linearizza quella installata si può esprimere in funzione di V come segue:

$$\phi = \frac{C_v(h)}{C_{vn}} = h \sqrt{\frac{V}{1 - (1 - V)h^2}}$$

il cui andamento è riportato in funzione di V nella figura a lato





valvole di regolazione

Scelta della caratteristica intrinseca

- Al variare di V la *caratteristica installa nel circuito di esempio* varia in modo differente a seconda che si scelga una caratteristica intrinseca lineare o equipercentuale

Dalla caratteristica installa in forma implicita,

$$w(h) = N_1 C_v(h) \sqrt{G_f \Delta p(w(h))}$$

quadrando e dividendo per il suo valore nominale, si ricava

$$\frac{w^2}{w_n^2} = \frac{C_v^2}{C_{vn}^2} \frac{\Delta p}{\Delta p_n}$$

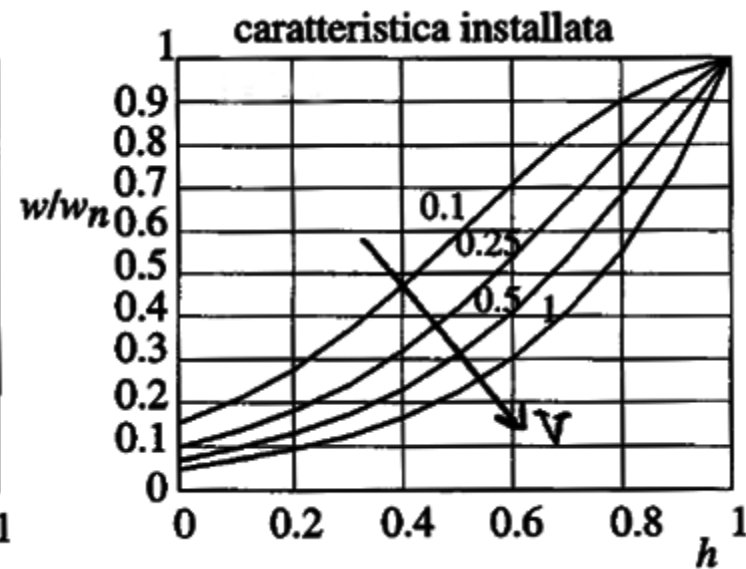
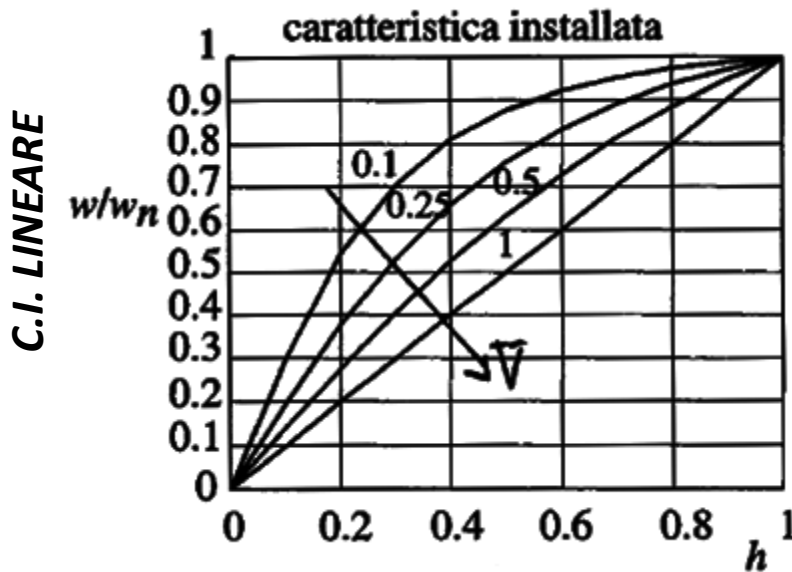
dove si è omessa la dipendenza da h

valvole di regolazione

Scelta della caratteristica intrinseca

Utilizzando l'espressione della $\Delta p(h)$ riferita al circuito divisa per Δp_n , si ottiene la caratteristica installata relativa in V

$$\frac{w}{w_n} = \frac{1}{\sqrt{1 - V + V \frac{C_{vn}^2}{C_v^2}}}$$





valvole di regolazione

Scelta della caratteristica intrinseca

- **Regole pratiche** per la scelta della caratteristica intrinseca:
 1. Se in condizioni nominali $V = \Delta p_n / \Delta p_0 > 0.4$ la caratteristica intrinseca lineare consente di ottenere una caratteristica installata più lineare
 2. Se in condizioni nominali $V = \Delta p_n / \Delta p_0 < 0.25$ si ottengono migliori risultati con la caratteristica equipercentuale
 3. Se $0.25 < V < 0.4$ le curve quadratiche, oppure le curve equipercentuali o lineari “modificate”, sono più adatte



valvole di regolazione

Scelta della caratteristica intrinseca

- Se V non è nota con sufficiente precisione si suggerisce di scegliere una caratteristica quadratica o modificata
- V elevato $\rightarrow$ aumento della caduta di pressione sulla valvola $\rightarrow$ aumento dei costi di pompaggio $\rightarrow$ molto diffuse le valvole equipercetuali
 - Per ottenere variazioni piccole della caduta sul circuito (caratteristica lineare) $\rightarrow$ sovradimensionamento delle tubazioni e della pompa $\rightarrow$ aumento dei costi di impianto e di esercizio
- **Valvole ad apertura rapida**
 - Sono usate in condizioni in cui è richiesto il **funzionamento in completa apertura o chiusura (*on/off*)** per avere la massima prontezza durante la fase di apertura della valvola

valvole di regolazione

Scelta della caratteristica intrinseca

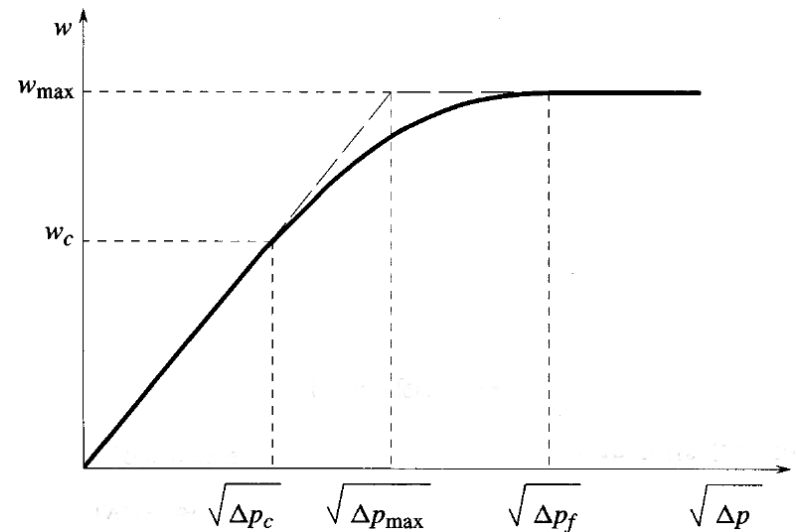


- Presenza di **elementi non lineari** nel circuito
 - *Esempio*: sensori di portata a strozzamento → forniscono un segnale proporzionale al quadrato della portata
 - Le regole precedenti non sono più valide
 - Si ricorre a **strumenti di calcolo e simulazione**
- Problema di **controllo di pressioni, livelli e temperature**
 - Uso di strumenti di calcolo e simulazione
 - *Tipicamente* si utilizzano valvole equipercettuali per le regolazioni di pressioni, e lineari per temperatura e livello

valvole di regolazione

Dimensionamento

- La normativa IEC sulle equazioni di dimensionamento delle valvole ignora la zona di funzionamento semicritico
 - Le equazioni di efflusso della zona semicritica non sono di semplice formulazione
 - La zona viene sostituita con i prolungamenti delle rette relative alla zone di flusso normale e limite
 - Punto in comune alle due rette:
 $\left(\sqrt{\Delta p_{\max}}, w_{\max} \right)$





valvole di regolazione

Dimensionamento

– Le **equazioni di efflusso approssimate** diventano:

- *Zona di flusso normale* ($0 \leq \Delta p < \Delta p_{max}$)

$$w = N_1 C_v \sqrt{G_f \Delta p}$$

- *Zona di flusso limite* ($\Delta p \geq \Delta p_{max}$)

$$w = w_{max}$$

dove Δp_{max} si ricava dalla seguente relazione:

$$\Delta p_{max} = F_L^2 (p_1 - F_F p_v)$$

in cui F_L è il *coefficiente di recupero* della valvola, determinato sperimentalmente e fornito dal costruttore, e F_F è il *coefficiente del rapporto della pressione critica per i liquidi*

valvole di regolazione

Dimensionamento



- Nota Δp_{max} si ricava w_{max} utilizzando l'equazione di efflusso

$$w_{max} = N_1 C_v \sqrt{\Delta p_{max} G_f} = N_1 C_v F_L \sqrt{(p_1 - F_F p_v) G_f}$$

che può essere **utilizzata per dimensionare la valvola nel caso in cui debba operare in condizioni di flusso limite**



valvole di regolazione

Dimensionamento: coefficiente di recupero F_L

- Il **coefficiente di recupero F_L** indica la capacità di una valvola di riconvertire l'energia cinetica posseduta dal fluido in vena contratta in energia meccanica di pressione
 - In vena contratta la pressione è minima ed è massima l'energia cinetica del fluido (è massima la velocità)
 - Nel tratto successivo l'energia cinetica si riduce (diminuisce la velocità) e si converte in energia di pressione (aumenta la pressione)
 - La conversione non è totale a causa delle **dissipazioni dovute agli attriti** interni alla valvola



valvole di regolazione

Dimensionamento: coefficiente di recupero F_L

- Il coefficiente di recupero F_L :
 - Dipende solo dalla geometria interna della valvola
 - Può essere ritenuto indipendente dalla corsa
 - Non dipende dalle proprietà del fluido
- Alcuni costruttori riportano il coefficiente $K_m = F_L^2$



valvole di regolazione

Dimensionamento: coefficiente di recupero F_L

- Il coefficiente di recupero può essere espresso come segue

$$F_L = \sqrt{\frac{p_1 - p_2}{p_1 - p_{vc}}}$$

poiché $p_{vc} < p_2$, risulta $F_L \leq 1$

- Quanto più F_L è prossimo all'unità tanto maggiore è l'attitudine della valvola a dissipare energia in attrito ($p_{vc} \rightarrow p_2$)
 - Valori bassi di F_L comportano che la zona di flusso semicritico si manifesti per salti di pressione bassi
- Valvole con F_L elevato possono, in alcuni casi, **evitare la cavitazione**: a parità di Δp la p_{vc} risulterà più elevata
 - Esistono valvole con trim speciali anticavitazione che hanno coefficienti di recupero prossimi all'unità



valvole di regolazione

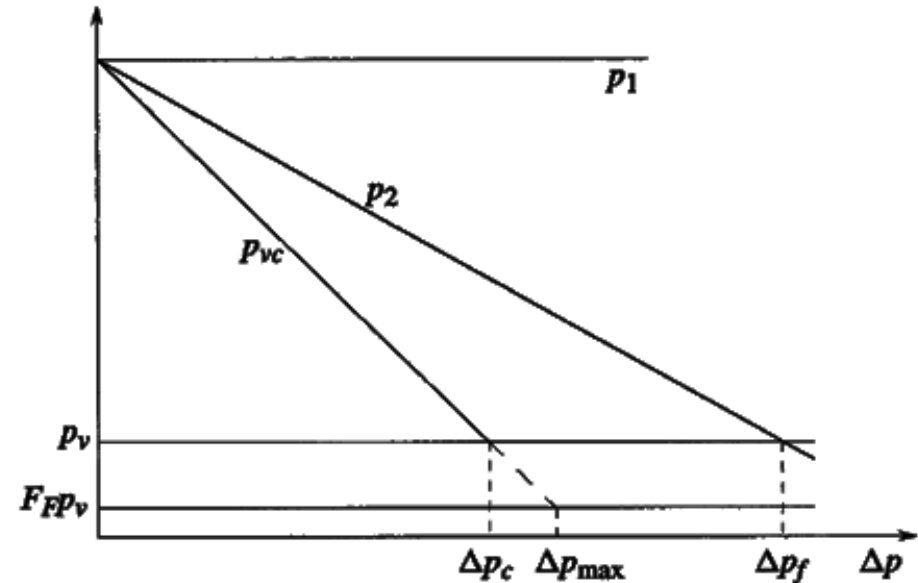
Dimensionamento: coefficiente di recupero F_L

- In base al valore di F_L le valvole possono essere suddivise in
 - Valvole a **basso recupero**
 - $F_L > 0.8$, in condizioni nominali
 - Caratterizzate da tortuosità del percorso compiuto dal fluido al loro interno, che sono causa di attriti e vortici che dissipano energia (*esempio*: valvole a globo)
 - Valvole ad **alto recupero**
 - $F_L < 0.8$, in condizioni nominali
 - Le valvole rotative sono nella grande maggioranza dei casi ad alto recupero (*esempio*: valvole rotative)

valvole di regolazione

Dimensionamento: coefficiente del rapporto della pressione critica per i liquidi F_F

- Il coefficiente del rapporto della pressione critica per i liquidi F_F può essere definito a partire dalla figura a lato, osservando quanto segue



- Nella zona di funzionamento normale ($\Delta p < \Delta p_c$) tra p_2 e p_{vc} esiste un legame lineare espresso da F_L
- Quando p_{vc} diventa pari a p_v in vena contratta il liquido inizia a vaporizzare e la relazione lineare non vale più



valvole di regolazione

Dimensionamento: coefficiente del rapporto della pressione critica per i liquidi F_F

- Quando p_{vc} diventa minore di p_v la portata non dipende più da p_2 ma da p_{vc} e raggiunge il valore massimo quando in tutta la sezione di vena contratta si ha vapore che si muove alla velocità del suono
- In questa condizione, la p_{vc} **da utilizzare per il calcolo della portata massima** si può ottenere da p_v per mezzo del coefficiente F_F ($p_{vc} = F_F p_v$), ovvero

$$F_F = \frac{p_{vc}}{p_v}$$

dove p_{vc} assume il valore relativo alla condizione di efflusso limite



valvole di regolazione

Dimensionamento: coefficiente del rapporto della pressione critica per i liquidi F_F

- Nell'ipotesi che il fluido, nelle condizioni di saturazione,
 - Sia un liquido fortemente disperso nel suo vapore
 - Possieda una velocità unica per i due stati fisici in equilibrio termodinamico tra loro

per il calcolo della F_F si può utilizzare la relazione:

$$F_F = 0.96 - 0.28 \sqrt{\frac{p_v}{p_c}}$$

in cui p_c è la *pressione termodinamica critica* (per l'acqua $p_c = 22.12$ MPa)

Nota: se le precedenti ipotesi non sono valide si deve ricorrere a relazioni di determinazione sperimentale



valvole di regolazione

Dimensionamento: coefficiente di incipiente cavitazione K_C

- Il **coefficiente di incipiente cavitazione** consente di determinare il valore della pressione differenziale a cui inizia a verificarsi il fenomeno di cavitazione
 - È rilevato sperimentalmente e fornito dal costruttore
 - È definito come segue:

$$K_C = \frac{\Delta p^*}{p_1 - p_v}$$

in cui Δp^* è il valore di pressione differenziale, con p_1 costante, al quale la valvola, con acqua, inizia a cavitare in misura tale che l'effettiva portata si discosti del 2% dal valore di efflusso teorico, predetto con Bernoulli



valvole di regolazione

Dimensionamento: coefficiente di incipiente cavitazione K_C

- Quando il valore di K_C non è noto, lo si può ricavare sperimentalmente tramite le seguenti relazioni:
 - Per valvole a globo: $K_C \approx 0.8F_L^2$
 - Per valvole rotative: $K_C \approx 0.6F_L^2 \div 0.8F_L^2$



valvole di regolazione

Procedura per il dimensionamento

- Ai fini del **dimensionamento di una valvola**, si considereranno noti
 - La portata nominale del fluido (w_n)
 - La caduta di pressione sulla valvola (Δp)
 - La temperatura (T)
 - La pressione a monte della valvola (p_1)Sarà compito del progettista reperire o calcolare tali dati dai disegni impiantistici



valvole di regolazione

Procedura per il dimensionamento

- Il **procedimenti di dimensionamento** prevede i seguenti passi:
 - 1) Si calcola del coefficiente di efflusso C_v con la relazione
$$C_v = \frac{w_n}{N_1 \sqrt{G_f \Delta p}}$$
 - 2) Si sceglie, a catalogo, una valvola con un C_v uguale o vicino a quello calcolato al punto 1)
 - Si **verifica**, utilizzando il coefficiente di recupero F_L per calcolare Δp_{max} ed eventualmente il coefficiente di incipiente cavitazione K_C per Δp_c , il punto di funzionamento sulla caratteristica di efflusso
 - Il coefficiente di incipiente cavitazione K_C sarà necessario nei casi più critici rispetto al problema della cavitazione (elevato scostamento della caratteristica di efflusso reale da quella approssimata)

valvole di regolazione

Procedura per il dimensionamento



- Una **indicazione conservativa**, rispetto a variazioni del *setpoint* che richiedano di aumentare rapidamente la portata, è di scegliere una valvola il cui coefficiente di efflusso sia pari a quello calcolato già alla corsa relativa del 70%, anziché quella nominale → valvole sovradimensionate, più costose
- Se il punto di funzionamento non è soddisfacente per la **presenza di condizioni di cavitazione** occorrerà cercare altre valvole con C_v simile ma diverso coefficiente di recupero F_L (più alto → aumento di perdite!)
- Per applicazioni in cui è richiesta una ampia escursione della portata, ad esempio fino ad un terzo del valore nominale (rangeability del processo 3:1), può essere necessario **verificare la linearità della caratteristica installata**



valvole di regolazione

Esempio di dimensionamento

- Si consideri un problema di dimensionamento caratterizzato dai seguenti punti:
 - Tipo di servizio: *ricircolazione d'acqua*
 - Tubazione (*piping*): 6"
 - Condizioni di lavoro:
 - $p_1 = 35$ bar
 - $\Delta p = 12.1$ bar
 - $T_1 = 28.3$ °C
 - $p_v = 0.0386$ bar
 - $\rho = 1000$ kg/m³
 - Portata nominale: $w_n = 300$ kg/s

valvole di regolazione

Esempio di dimensionamento



- Primo tentativo di dimensionamento:
 - 1) Calcolo del coefficiente di efflusso: $C_v = 358.9$ gpm
 - 2) Si sceglie una *valvola a sfera* da 6 pollici con caratteristica intrinseca esponenziale, rangeability $r = 20:1$, $C_{vn} = 940$ gpm, $K_m = 0.3$ ($F_L = 0.55$)
 - Dall'applicazione della caratteristica equipercentuale $\phi(h) = r^{h-1}$ si ottiene $\phi(0.7) = 0.407$ e $C_v(0.7) = 0.407 C_{vn} = 382.7$ gpm, corrispondente all'incirca al valore calcolato al passo 1)
- *Verifica per la cavitazione*: dall'espressione del coefficiente del rapporto della pressione critica per i liquidi si ottiene $F_F = 0.957 \rightarrow$ il Δp_{max} che identifica la zona di flusso limite risulta $\Delta p_{max} = 10.5$ bar $\rightarrow \Delta p > \Delta p_{max} \rightarrow$ **cavitazione!**

valvole di regolazione

Esempio di dimensionamento



- Secondo tentativo di dimensionamento:
 - 2) Si sceglie una *valvola a globo* con caratteristiche simili alla precedente, ma con coefficiente di recupero più elevato: $K_m = 0.6$ ($F_L = 0.77$)
 - *Verifica per la cavitazione*: $\Delta p_{max} = 21 \text{ bar} \rightarrow \Delta p < \Delta p_{max} \rightarrow$ valvola lavora nella zona di flusso normale
 - *Verifica più accurata*: si considera il coefficiente di incipiente cavitazione per la valvola scelta $K_C = 0.45 \rightarrow$ da cui risulta $\Delta p_c = 15.7 \text{ bar} > \Delta p = 12.1 \text{ bar} \rightarrow$ il punto di lavoro sarà dentro la zona di funzionamento normale



valvole di regolazione

Dimensionamento: verifica della velocità di uscita dalla valvola

- **Verifica della velocità di uscita dalla valvola**
 - La velocità non influenza il calcolo del C_v , ma influisce sul **rumore acustico** generato dal moto del fluido
 - Le normative impongono dei limiti al rumore
 - I valori massimi di velocità accettati dipendono dal particolare fluido e dalle dimensioni della valvola
 - Sono generalmente limitata ad alcuni metri al secondo per i fluidi e frazioni di Mach per gas e vapori surriscaldati (100÷120 m/s)
 - Per ridurre la velocità è necessario **aumentare il diametro delle valvole**



valvole di regolazione

Dimensionamento: flusso non turbolento

- Le equazioni di efflusso derivate in precedenza sono valide solo in ipotesi di moto del fluido nella valvola turbolento
 - In caso di flusso non turbolento ($Re < 4000$) predicono una portata superiore a quella reale
 - La normativa ANSI/ISA relativa al dimensionamento delle valvole di controllo prevede un apposito **coefficiente correttivo del numero di Reynolds**
 - va introdotto nelle equazioni di efflusso e consente di ottenere il valore di portata corretto anche in condizioni di moto non turbolento



valvole di regolazione

Dimensionamento: adattatori di dimensione (riduzioni)

- In alcuni casi, per adattare valvole con dimensioni nominali inferiori a quelle della condotta, si usano degli adattatori di dimensione detti **riduttori**
 - Sono elementi molto corti caratterizzati da una repentina variazione di sezione tra ingresso e uscita
 - Modificano i profili di pressione e di velocità del fluido → provocano perdite di carico aggiuntive
 - Particolarmente rilevanti se il diametro della tubazione è superiore al doppio di quello della valvola
 - Le normative prevedono appositi **coefficienti correttivi**, che consentono di utilizzare le equazioni di efflusso per l'insieme costituito da valvola e riduzioni



valvole di regolazione

Fluidi comprimibili

- Il moto dei **fluidi comprimibili** (gas, vapori, miscele liquido-vapore) è governato dalle equazioni di conservazione della massa
 - La loro densità dipende fortemente dalla pressione e dalla temperatura → il comportamento differisce molto da quello dei fluidi incomprimibili
 - La portata w di fluido *comprimibile* transitante in una valvola è funzione della radice della **pressione differenziale relativa** x , definita come il rapporto tra il salto di pressione e la pressione a monte $x = (p_1 - p_2) / p_1$



valvole di regolazione

Fluidi comprimibili

- Al fine di mantenere la struttura dell'equazione per i fluidi incomprimibili, la normativa ANSI/ISA prevede di esprimere la portata come:

$$w = N_6 C_v Y(x) \sqrt{x p_1 \rho_1}$$

dove N_6 è una costante dimensionale opportuna

- L'equazione vale per $0 \leq x \leq F_k x_T$, dove x_T è la *pressione differenziale relativa limite*, rilevabile sperimentalmente e fornita dai costruttori, e $F_k = (c_p/c_v)/1.4$ è il rapporto tra l'esponente dell'adiabatica del gas in questione e quello dell'aria (c_p e c_v sono i calori specifici a pressione e a volume costante)
- Y è detto *coefficiente di espansione* e vale $Y(x) = 1 - x / (3F_k x_T)$
 - x limitato $\rightarrow Y$ limitato: $2/3 \leq Y \leq 1$

valvole di regolazione

Fluidi comprimibili

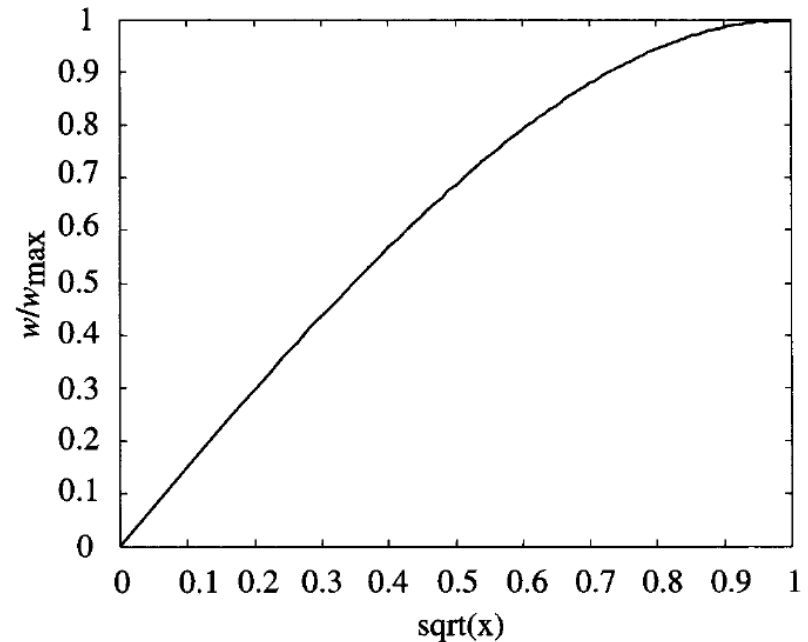


- La portata raggiunge il suo valore massimo per

$$x = F_k x_T$$

$$w_{\max} = N_6 C_v \frac{2}{3} \sqrt{F_k x_T p_1 \rho_1}$$

come si può apprezzare dal grafico a lato, tracciato per $F_k x_T = 1$, nell'ipotesi di tenere fisse le condizioni a monte e la corsa e facendo via via diminuire p_2





valvole di regolazione

Fluidi comprimibili

- La portata massima si raggiunge quando il fluido alla vena contratta raggiunge la massima la massima sezione trasversale alla velocità del suono e la pressione p_2 vale $p_{2CF} = p_1 (1 - F_k x_T)$
- Il valore p_{2CF} rappresenta l'estremo inferiore della zona di flusso limite → diminuendo ulteriormente p_2 la portata non aumenta più
- Anche la pressione di vena contratta raggiunge un valore limite che dipende solo da p_1 , secondo la relazione approssimata $p_1/p_{vc} \cong 2$



valvole di regolazione

Attuatori e servoposizionatori

- Gli **attuatori** hanno il compito di posizionare lo steso e l'otturatore della valvola in funzione del segnale proveniente dal regolatore
- Si utilizzano attuatori
 - Pneumatici
 - Oleodinamici
 - Elettromeccanici

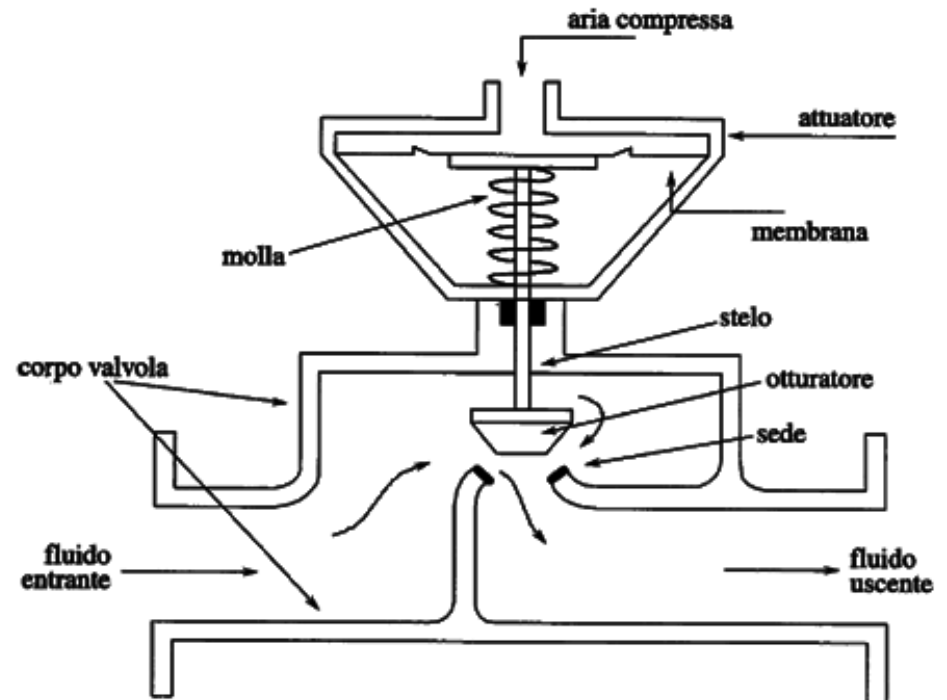
valvole di regolazione

Attuatori e servoposizionatori



- **Attuatori pneumatici a membrana**

- Sono i più diffusi sulle valvole di regolazione
- Semplicità di funzionamento e costo ridotto
- La membrana di gomma rinforzata è fissata al contenitore metallico in modo che la camera superiore (inferiore) sia a tenuta stagna
- L'attuatore in figura è di tipo *diretto*





valvole di regolazione

Attuatori e servoposizionatori

- Se la camera d'aria viene realizzata sullo stesso lato dello stelo l'attuatore viene detto *inverso*
- La pressione dell'aria sulla membrana determina una forza che la deforma spingendo verso il basso lo stelo e l'otturatore (verso l'alto nel caso di attuatori indiretti)
- La deformazione della membrana è contrastata dalla **molla**, in compressione sul lato inferiore, che spinge verso l'alto (opposto per gli attuatori indiretti)
- In *assenza di pressione* dell'aria la valvola si apre completamente (si chiude nel caso di attuatori indiretti)
 - Si dirà che la valvola funziona in modo *aria chiude* (nel caso opposto si direbbe *aria apre*)

valvole di regolazione

Attuatori e servoposizionatori



- Si dirà *flusso chiude* quando il moto del fluido tende a chiudere l'otturatore e *flusso apre* nel caso opposto
- L'attuatore a membrana deve essere **dimensionato** in modo da permettere la corsa dello stelo da 0 a 100%, a seguito di un segnale standard, per esempio 3÷15 psi, vincendo le resistenze dovute: agli attriti, alla gravita, alle inerzie in movimento, alla pressione assoluta del fluido e alla pressione differenziale monte-valle dell'otturatore
 - L'attuatore deve inoltre realizzare una spinta sulla sede per realizzare una **chiusura a tenuta**
 - Il primo passo per il dimensionamento è la stima di queste forze
 - Il secondo passo è la scelta di una **membrana di superficie adeguata**, essendo la forza proporzionale ad essa

valvole di regolazione

Attuatori e servoposizionatori



- Il segnale di pressione standard $3\div 15$ psi è ottenuto con appositi **convertitori I/P** (*corrente-pressione*)
 - Trasducono il segnale di corrente $4\div 20$ mA nel segnale pneumatico $3\div 15$ psi
- **Posizione *fail-safe***
 - Gli attuatori a membrana sono **intrinsecamente sicuri**
 - Disponendo opportunamente la molla di contrasto si può fare in modo che in assenza della pressione di controllo la valvola si porti nella posizione ritenuta sicura per l'impianto, di completa apertura o chiusura
- *Limiti*: corsa e spinta limitate, bassa velocità di intervento

valvole di regolazione

Attuatori e servoposizionatori



- **Attuatori pneumatici a pistone**
 - Un pistone sostituisce la membrana
 - Corse e spinte più elevati degli attuatori a membrana
 - Esistono pistoni a *doppio effetto*: le spinte per il moto in entrambe i sensi proviene dall'aria di comando (non ci sono molle in contrapposizione)
 - Richiedono l'impiego di un **servoposizionatore** o di un **sistema di controllo retroazionato di posizione** dello stelo
 - Sono molto costosi → poco diffusi

valvole di regolazione

Attuatori e servoposizionatori



- **Attuatori elettrici**

- Sono utilizzati per la **regolazione continua**
- Costituiti da
 - un motore elettrico
 - un riduttore di velocità (→ incrementa la coppia)
 - (eventuali) sensori di posizione e/o velocità
 - una unità elettronica di potenza (azionamento) e controllo
- Operano in due modi principali:
 - Il servomotore muove a velocità costante in apertura o in chiusura o rimane fermo → il regolatore determina la posizione modulando la durata dei tempi di moto e di fermo (regolazione a ciclo aperto)
 - Esiste un effettivo servomeccanismo di posizione (con sensori)

valvole di regolazione

Attuatori e servoposizionatori



- Sono indicati nei casi in cui sono richieste corse lunghe e spinte elevate
- Non sono *fail-safe*!
- Esistono **costruzioni antideflagranti**, ma poco utilizzate in ambienti a rischio esplosione (diffidenza)

valvole di regolazione

Attuatori e servoposizionatori



- **Attuatori oleodinamici**

- Producono spinte e velocità di intervento superiori a quelle degli altri attuatori
- Molto costosi → utilizzati con parsimonia
 - *Esempio:* impiegati nelle valvole di turbina di grossi gruppi di produzione di energia elettrica, dove in pochi decimi di secondo sono in grado di chiudere le valvole di intercettazione vapore in caso di blocco turbina
- Costituiti da
 - Una parte oleodinamica di attuazione
 - Una parte elettronica di controllo
- Esistono realizzazioni *fail-safe*



pompe

Grandezze fondamentali

- Grandezze di interesse per modellare le pompe:
 - **Portata volumetrica** q
 - **Portata massica** w
 - **Pressione del fluido alla flangia di aspirazione** p_e
 - **Pressione del fluido alla flangia di mandata** p_u
 - **Rendimento totale** η
 - Definito come il rapporto tra la potenza fornita dalla girante al fluido in uscita dalla flangia di mandata e la potenza trasferita all'albero della pompa del motore che l'aziona
 - **Velocità di rotazione dell'albero della pompa** n

pompe

Grandezze fondamentali



– Prevalenza totale $H = H_u - H_e$

- Espressa in metri di colonna di acqua
- H_u e H_e sono i **carichi totali** alle flange di mandata e di aspirazione:

$$H_u = \frac{v_u^2}{2g} + z_u + \frac{p_u}{\rho g} \quad H_e = \frac{v_e^2}{2g} + z_e + \frac{p_e}{\rho g}$$

dove z_u e z_e sono le quote e v_u e v_e le velocità medie del fluido riferite alla flangia di mandata e di aspirazione, rispettivamente, da cui si ricava l'espressione della prevalenza

$$H = H_u - H_e = \frac{v_u^2 - v_e^2}{2g} + (z_u - z_e) + \frac{p_u - p_e}{\rho g}$$

Nell'ipotesi spesso verificata che $z_u \cong z_e$ e $v_u \cong v_e$ risulterà

$$H \cong \frac{p_u - p_e}{\rho g}$$



pompe

Grandezze fondamentali

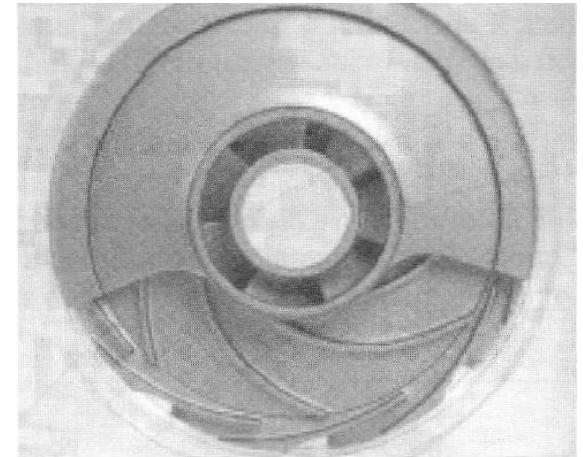
– **Condizioni nominali di funzionamento**

- Corrispondono al *punto di lavoro* con il massimo rendimento totale $\eta = \eta_{\max}$
 - Velocità di rotazione nominale n_0
 - Prevalenza nominale H_0
 - Portata volumetrica nominale q_0

pompe centrifughe

Generalità

- Le **pompe centrifughe** appartengono alla categoria delle pompe *fluidodinamiche*
 - Macchina operatrice idraulica in cui una corrente di fluido attraversa un elemento rotante munito di pale opportunamente sagomate (*girante*) → aumenta la pressione (e/o la velocità) → aumenta il carico totale
 - Sono di **costruzione modulare**
 - A *singolo stadio*
 - A più stadi identici disposti in serie → *pompa multistadio*

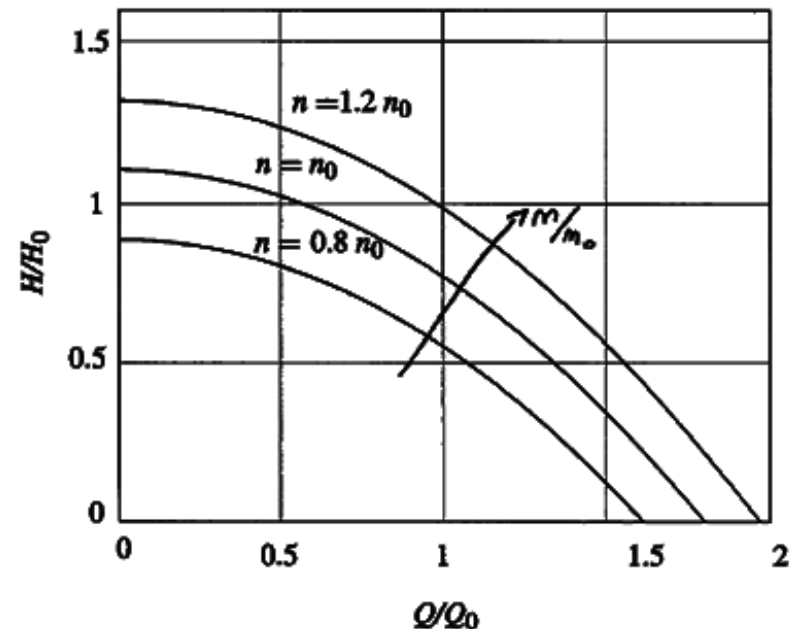


pompe centrifughe

Modello matematico



- **Modello matematico** di una pompa centrifuga:
 - Ipotesi di funzionamento normale e assenza di cavitazione
 - *Modello puramente algebrico*: si trascurano tutti i fenomeni di accumulo (volumi piccoli)
 - **Caratteristiche normalizzate prevalenza–portata** in corrispondenza di differenti velocità di rotazione
 - Le curve si possono approssimare con delle **parabole**



pompe centrifughe

Modello matematico



- Assumendo che il vertice si trovi sull'asse delle prevalenze:

$$H = An^2 + Bq^2$$

dove A e B sono costanti che si determinano imponendo il passaggio per due punti di una curva sperimentale prevalenza–portata, in funzionamento a giri costanti, fornita dal costruttore

- Scegliendo i punti sulla curva relativa alla velocità nominale ($n=n_0$) corrispondenti a
 - prevalenza H_M e portata nulla (vertice della parabola)
 - prevalenza H_0 e portata q_0 nominali (tipicamente H_0 è compreso tra il 75% e il 90% di H_M)

risulta:

$$A = \frac{H_M}{n_0^2} \quad B = \frac{H_0 - H_M}{q_0^2}$$

pompe centrifughe

Modello matematico



- L'incremento Δp di pressione si ricava dall'espressione della prevalenza:

$$\Delta p = p_u - p_e = A\rho g n^2 + B\rho g q^2$$

che in termini di portata massica si può esprimere come segue:

$$\Delta p = H_M g \rho \left(\frac{n}{n_0} \right)^2 + \frac{H_0 - H_M}{q_0^2} g \frac{w^2}{\rho} = k_{p1} \rho \left(\frac{n}{n_0} \right)^2 + k_{p2} \frac{w^2}{\rho}$$

in cui

$$k_{p1} = H_M g \quad k_{p2} = \frac{H_0 - H_M}{q_0^2} g$$

Nota: essendo $H_M > H_0$ risulta $k_{p2} < 0 \rightarrow$ al crescere della portata massica w si riduce il salto di pressione Δp

pompe centrifughe

Cavitazione



- La **cavitazione** in una pompa centrifuga si verifica quando nella *zona a pressione minima* la pressione scende al di sotto della pressione di vapore p_v
 - La **zona a pressione minima** è situata all'entrata del liquido sulle pale della girante (*occhio della girante*)
 - Le bolle di vapore che si generano nella zona a pressione minima vengono trasportate dal fluido verso zone a pressione superiore, dove collassano → *cavitazione*
 - Il collasso ripetuto delle bolle provoca **onde di pressione** di intensità molto elevata accompagnate da **rumore** e dall'**erosione** degli organi interni della pompa

pompe centrifughe

Cavitazione



- La verifica dell'occorrenza della cavitazione viene realizzata con il parametro **NPSH** (*Net Positive Suction Head*):

$$\text{NPSH} = \frac{p_e - p_{\min}}{\rho g} + \frac{v_e^2}{2g}$$

in cui $p_{\min}$ è la pressione nella zona a pressione minima, v_e è la velocità media del fluido alla flangia di aspirazione

→ considerato che in cavitazione si ha $p_{\min} \leq p_v$, allora

$$\text{NPSH} \geq \frac{p_e - p_v}{\rho g} + \frac{v_e^2}{2g} \Rightarrow \text{cavitazione}$$

dove p_v è la pressione di vapore alla temperatura media della zona a pressione minima $\approx T$ flangia di aspirazione

pompe centrifughe

Cavitazione



- Normalmente il *termine cinetico* $v_e^2/(2g)$ è trascurabile rispetto al *termine piezometrico*, si può quindi considerare il test semplificato

$$\text{NPSH} \geq \frac{p_e - p_v}{\rho g} \quad \Rightarrow \quad \text{cavitazione}$$

- Il parametro NPSH è fornito dai costruttori
- Per **evitare la cavitazione** l'installazione della pompa deve avvenire in modo tale che la pressione alla flangia di aspirazione sia superiore alla soglia di cavitazione in tutte le condizioni di funzionamento:

$$p_e \geq \rho g \text{NPSH} + p_v$$

pompe volumetriche

Generalità



- Le **pompe volumetriche** sono macchine operatrici idrauliche a funzionamento *ciclico*
 - In ogni ciclo di funzionamento un determinato volume è portato dalla bassa pressione della flangia di aspirazione all'alta pressione della flangia di mandata per effetto del movimento di uno o più elementi mobili
 - Pompe volumetriche **alternative**
 - Gli elementi mobili si muovono di moto alternativo
 - *Esempio*: pompe a stantuffo o a membrana
 - Pompe volumetriche **rotative**
 - Gli elementi mobili si muovono di moto rotativo
 - *Esempio*: pompe a ingranaggi, a lobi, a capsulismi, a palette

pompe volumetriche

Modello matematico



- **Modello matematico** di una pompa volumetrica
 - Si trascurano i fenomeni di accumulo → *modello algebrico*
 - Non si considererà il problema della *cavitazione* → non è generalizzabile per tutti i modelli di pompe volumetriche
 - Con riferimento al comportamento *ciclico* della pompa, la portata volumetrica può essere espressa come segue:

$$q = \eta_v k_1 n$$

in cui

- n è la velocità di rotazione dell'albero motore che aziona la pompa
- k_1 è una costante che dipende dalle caratteristiche costruttive della pompa



pompe volumetriche

Modello matematico

- η_v è il **rendimento volumetrico** (*grado di riempimento*)
 - È definito come il rapporto tra il volume di liquido effettivamente pompato a ogni ciclo e quello massimo occupabile dal fluido
 - Non è costante → *diminuisce all'aumentare sia della prevalenza sia della velocità di rotazione*
- Nel funzionamento normale il rendimento volumetrico η_v può ritenersi costante, quindi

$$q = k_p n$$

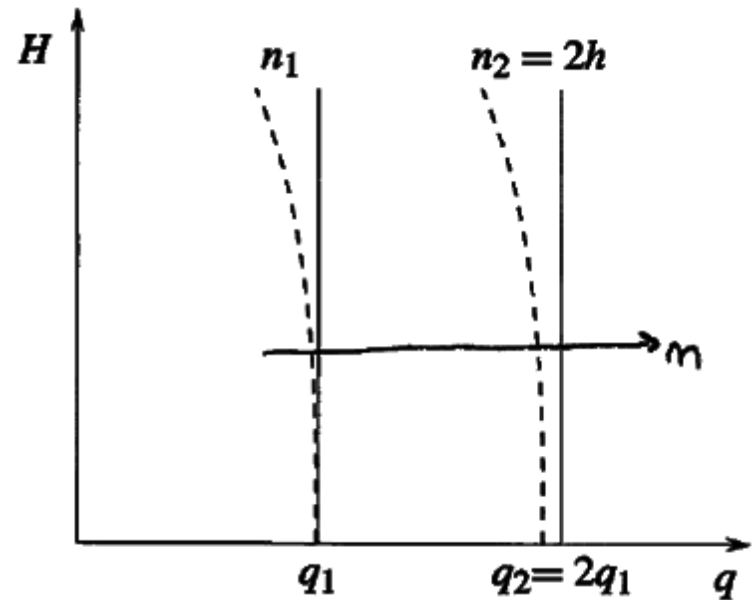
in cui si è posto $k_p = \eta_v k_1$, con η_v costante

pompe volumetriche

Modello matematico

- L'andamento qualitativo della caratteristica della pompa è riportato a lato, nel piano $H - q$
- La linea tratteggiata mostra l'**andamento reale**, determinato dalla progressiva riduzione di η_v con H e con n
- Il valore della **costante** k_p si può determinare dai valori della velocità nominale di funzionamento n_0 e dalla portata nominale q_0 :

$$k_p = \frac{q_0}{n_0}$$



pompe volumetriche

Modello matematico



- Il **funzionamento nominale** corrisponde a quello di *massimo rendimento* della pompa
- La portata massica risulta $w = \rho q$
- *Nota:* nelle pompe volumetriche alternative il **flusso di liquido non è continuo**, quindi i valori forniti dalle relazioni precedenti sono da intendersi come *valori medi* di un ciclo



schemi di collegamento e regolazione

Generalità

- Diversi **schemi di collegamento** di pompe e valvole possono essere realizzati allo scopo di **regolare la portata** di un circuito
 - Schemi di collegamento diversi sono caratterizzati da una diversa *efficienza*
 - *Esempio*: collegamento seriale (valvola in serie alla pompa)
 - Se la portata deve rimanere a valori molto inferiori a quelli nominali per lungo tempo si ha un notevole spreco di energia



schemi di collegamento e regolazione

Generalità

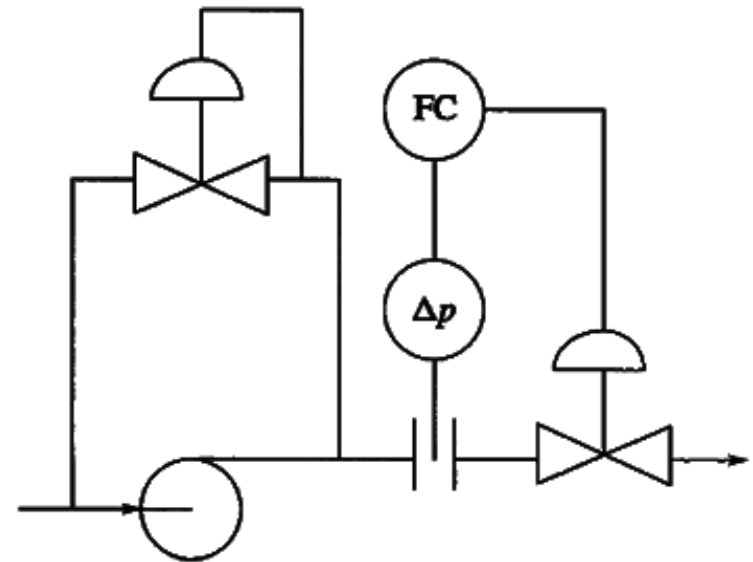
- Nella scelta dello schema di collegamento si devono considerare le **caratteristiche del motore** che aziona la pompa
 - Se si dispone di un **azionamento a velocità variabile**, la modulazione della portata può avvenire modulando la velocità del motore → si massimizza l'efficienza, si penalizzano la prontezza di risposta e la rangeability
- Le **pompe volumetriche** sono adatte quando sono richieste elevate pressioni di mandata
 - Il circuito a valle deve essere in grado di assorbire la portata nominale della pompa senza provocare eccessivi innalzamenti della pressione di mandata → danni alla pompa o alla linea, stallo del motore

schemi di collegamento e regolazione

Generalità

- Le pompe volumetriche vanno sempre protette con **valvole di scarico automatiche**

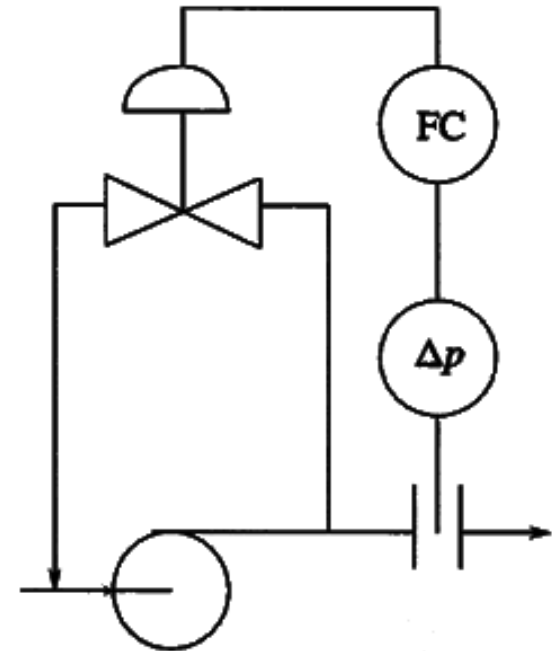
- Si introduce una *linea di bypass* dalla mandata all'aspirazione con una **valvola autoregolata**, la cui apertura viene comandata dalla pressione al suo ingresso → la pressione viene limitata ad una **soglia di taratura**
- Analoga protezione va inserita quando si vuole modulare la portata con una valvola in serie ad una pompa volumetrica



schemi di collegamento e regolazione

Generalità

- La **modulazione di portata** può essere realizzata mediante una linea di bypass con una valvola di regolazione comandata direttamente dal regolatore
- Le **pompe centrifughe** si utilizzano normalmente con una valvola in serie alla pompa, senza bypass di ricircolo
 - Soluzione particolarmente adottata per pompe di piccole dimensioni accoppiate a motori a velocità costante
 - Le **pompe centrifughe possono funzionare anche a portata nulla** ($H = H_M$) → non è necessario limitare la pressione alla mandata

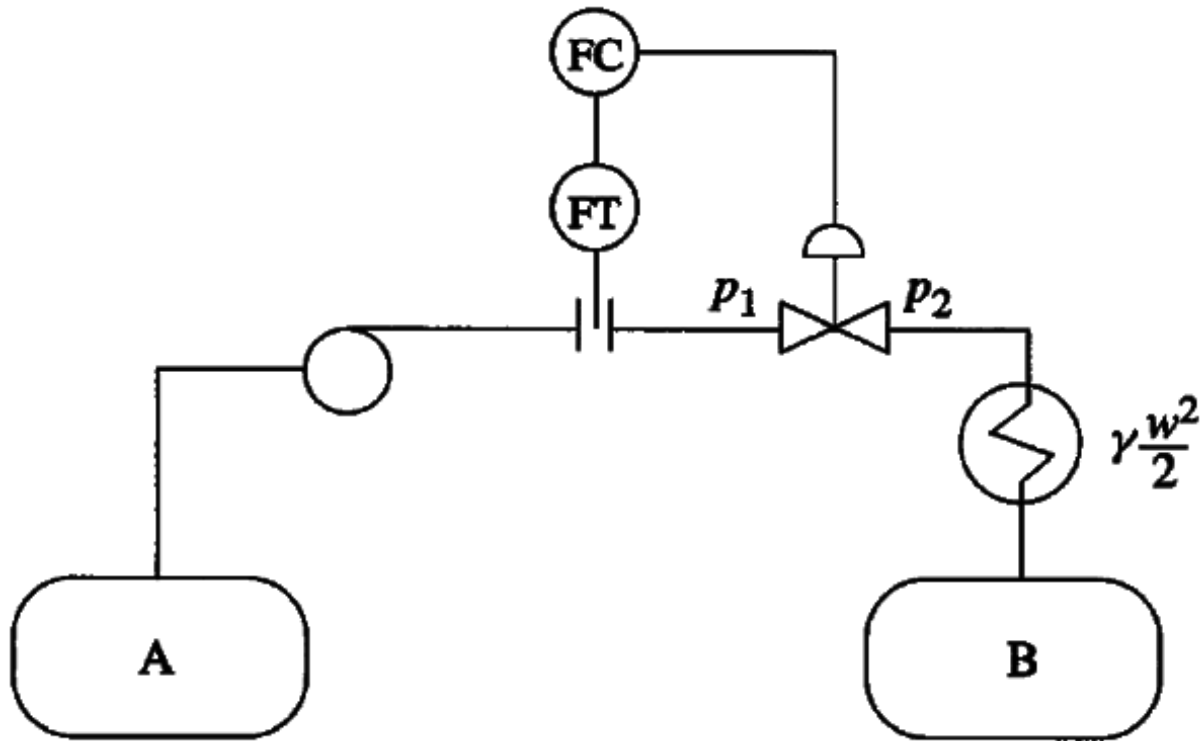




schemi di collegamento e regolazione

Schematizzazione del problema di controllo della portata

- **Esempio: regolazione di portata**





schemi di collegamento e regolazione

Schematizzazione del problema di controllo della portata

- Elementi presenti nel circuito idraulico:
 - Un **serbatoio di aspirazione**, alla pressione p_A
 - Un **serbatoio di scarico**, alla pressione p_B
 - Una **pompa centrifuga**
 - Una **valvola di regolazione**
 - Uno **scambiatore di calore** (*perdita di carico concentrata*)
- Si suppone inoltre che
 - La pompa è azionata da un motore a giri costanti
 - I serbatoi sono alla stessa quota, ma a pressioni diverse
 - Ipotesi di *fluido incomprimibile* → perdite di carico nello scambiatore proporzionali alla portata: $\gamma w^2/2$



schemi di collegamento e regolazione

Schematizzazione del problema di controllo della portata

- Le equazioni che esprimono portata e pressioni nel circuito sono espresse dal seguente sistema algebrico:

$$p_1 = p_A + k_{p1}\rho + k_{p2} \frac{w^2}{\rho}$$

$$p_2 = p_B + \gamma \frac{w^2}{2}$$

$$w = N_1 C_v(h) \sqrt{G_f(p_1 - p_2)}$$

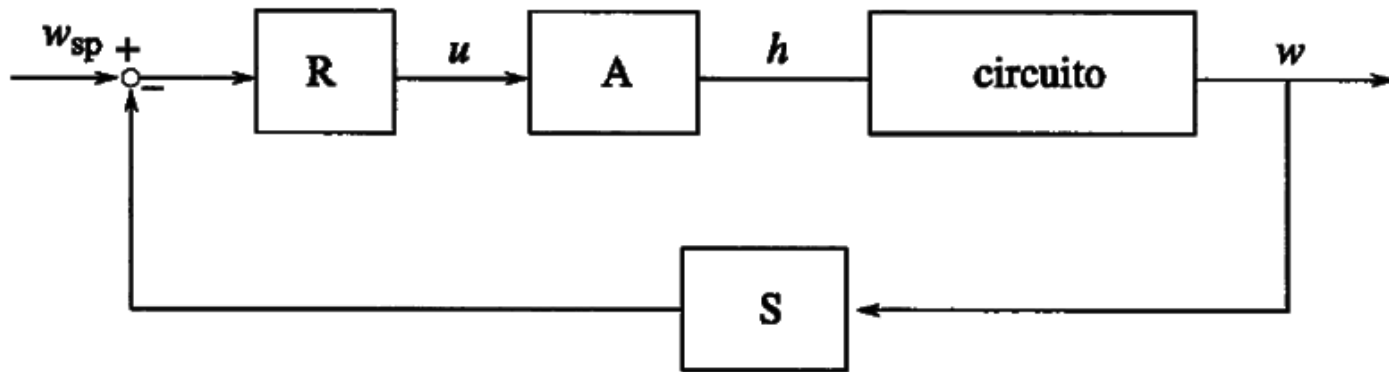
- si può ricavare la *relazione statica* $w(h)$
- Scegliendo opportunamente la caratteristica intrinseca, risulterà lineare: $w(h) = k h \rightarrow$ il processo algebrico da controllare può essere descritto con un **puro guadagno $\Delta w / \Delta h$ costante** e pari alla pendenza media della caratteristica installata: $\Delta w / \Delta h = w(1) - w(0)$

schemi di collegamento e regolazione

Schematizzazione del problema di controllo della portata



– Schema a blocchi dell'**anello di regolazione**:



- Il blocco A rappresenta il dispositivo che attua il movimento dell'otturatore della valvola, comandato dal regolatore
- Attuatore e processo hanno una banda elevata rispetto alla banda dell'anello di controllo → si trascurano le loro dinamiche → sistema puramente algebrico → impiego di **regolatori integrali**



attuatori per il controllo del moto

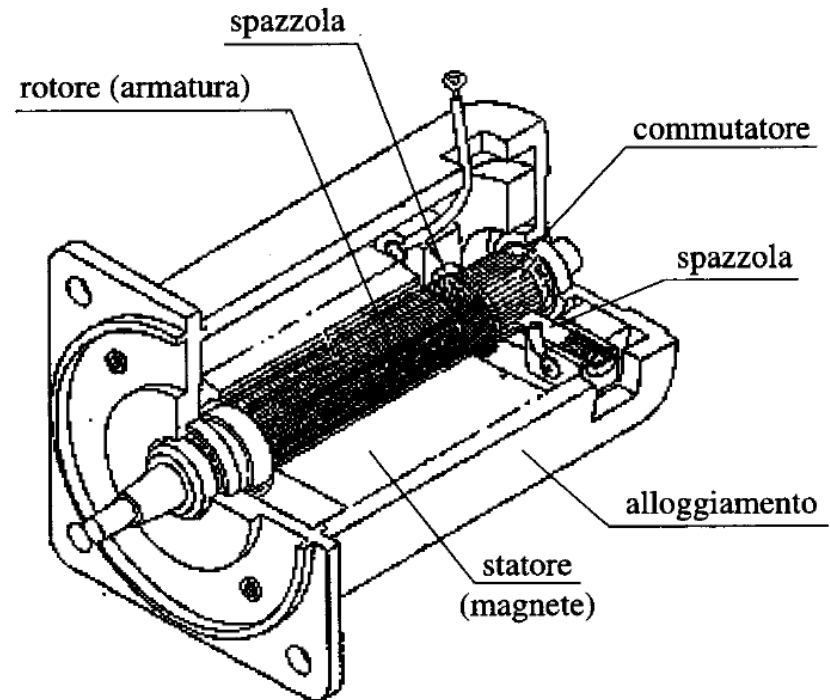
Introduzione

- **Attuatori per il controllo del moto**
 - Attuatori *elettrici*
 - Attuatori *oleodinamici*
 - Consentono il controllo della posizione relativa di organi meccanici in movimento
 - Utilizzano la **retroazione** della misura di posizione e di velocità
- **Non saranno considerati**
 - Motori elettrici asincroni → non consentono la regolazione di posizione a valori costanti
 - Motori passo-passo → non necessitano della retroazione sulla posizione (controllati ad anello aperto)

motori elettrici a magneti permanenti a corrente continua

Struttura del motore a corrente continua

- **Un motore a corrente continua è costituito da**
 - **Rotore:** un cilindro mobile di materiale ferromagnetico su cui sono avvolte le spire collegate a formare un circuito chiuso (**armatura**)
 - **Statore:** una parte fissa su cui vengono posti i magneti permanenti che generano il campo magnetico, comprensiva dell'alloggiamento

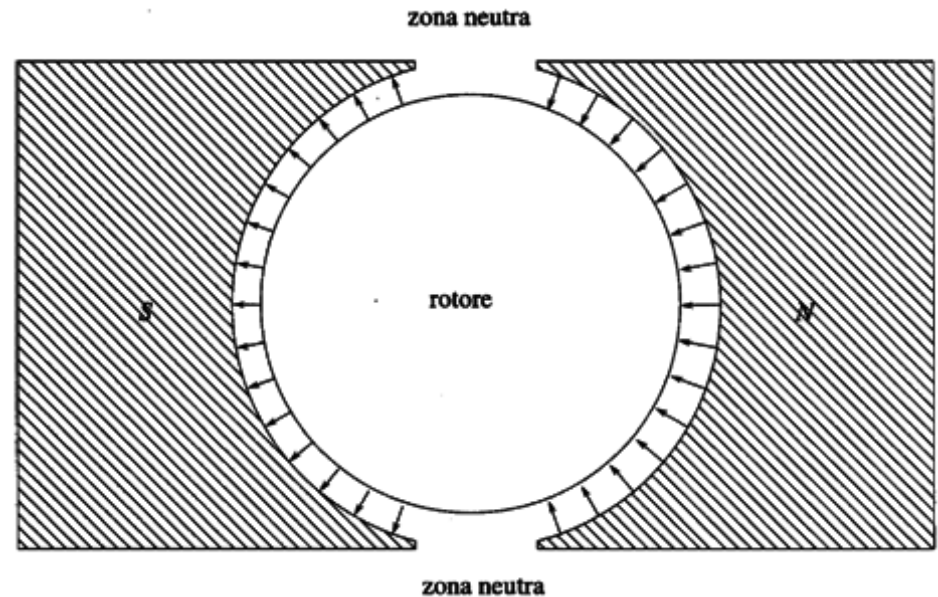


motori elettrici a magneti permanenti a corrente continua

Struttura del motore a corrente continua

– **Traferro:** spazio anulare di forma cilindrica compreso tra statore e rotore

- È sede di un campo magnetico le cui linee di forza, assumendo infinita la permeabilità magnetica del ferro, sono dirette in senso radiale
- *Nota:* i conduttori attivi sono disposti parallelamente all'asse del rotore, che individua quindi anche la direzione del campo elettrico equivalente indotto dal moto dei conduttori

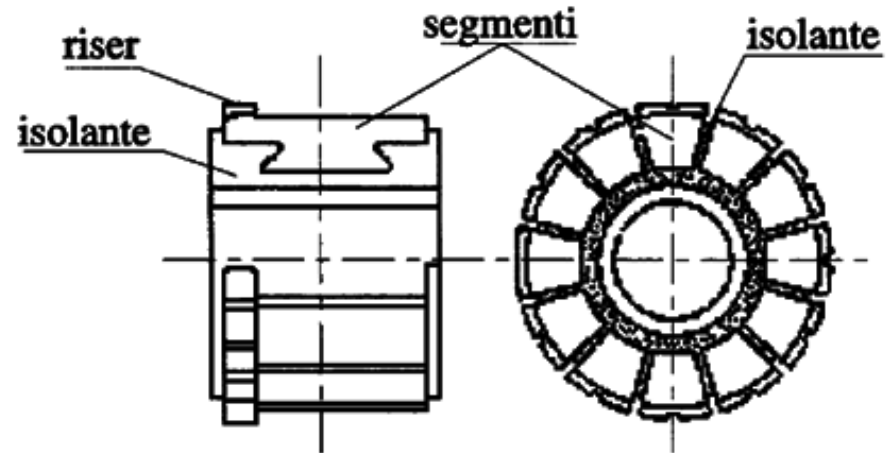


motori elettrici a magneti permanenti a corrente continua

Struttura del motore a corrente continua

- **Collettore (commutatore):**
costituito da segmenti di rame, isolati con mica o plastica, alle cui sporgenze (*risers*) sono collegati i terminali delle spire del circuito di armatura

- Il numero di segmenti coincide con il numero di spire
- Maggiore è il numero di segmenti → più uniforme risulta la coppia generate



motori elettrici a magneti permanenti a corrente continua

Struttura del motore a corrente continua

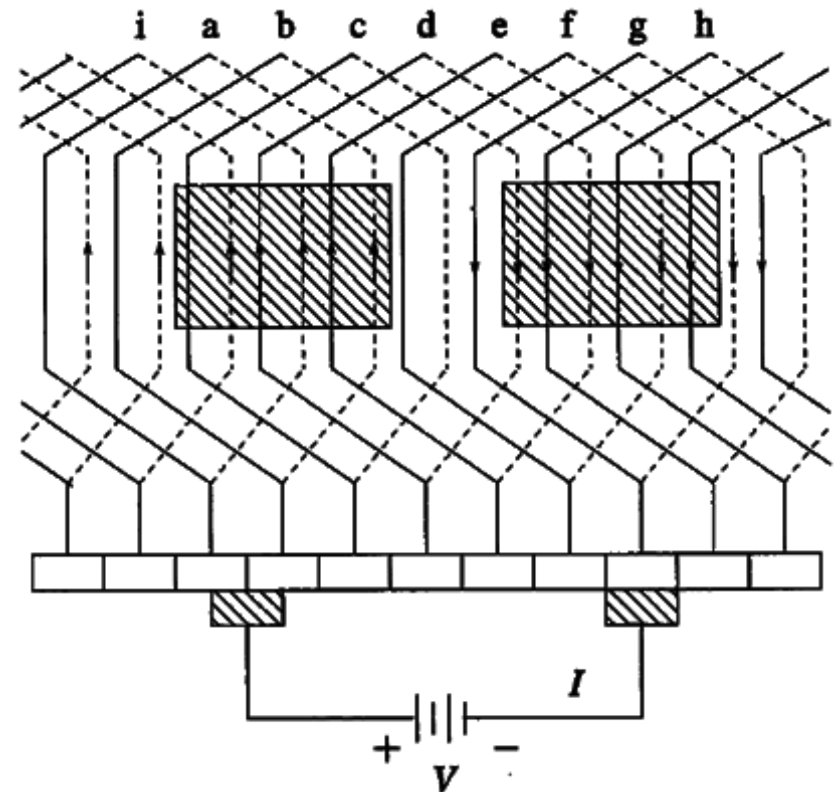
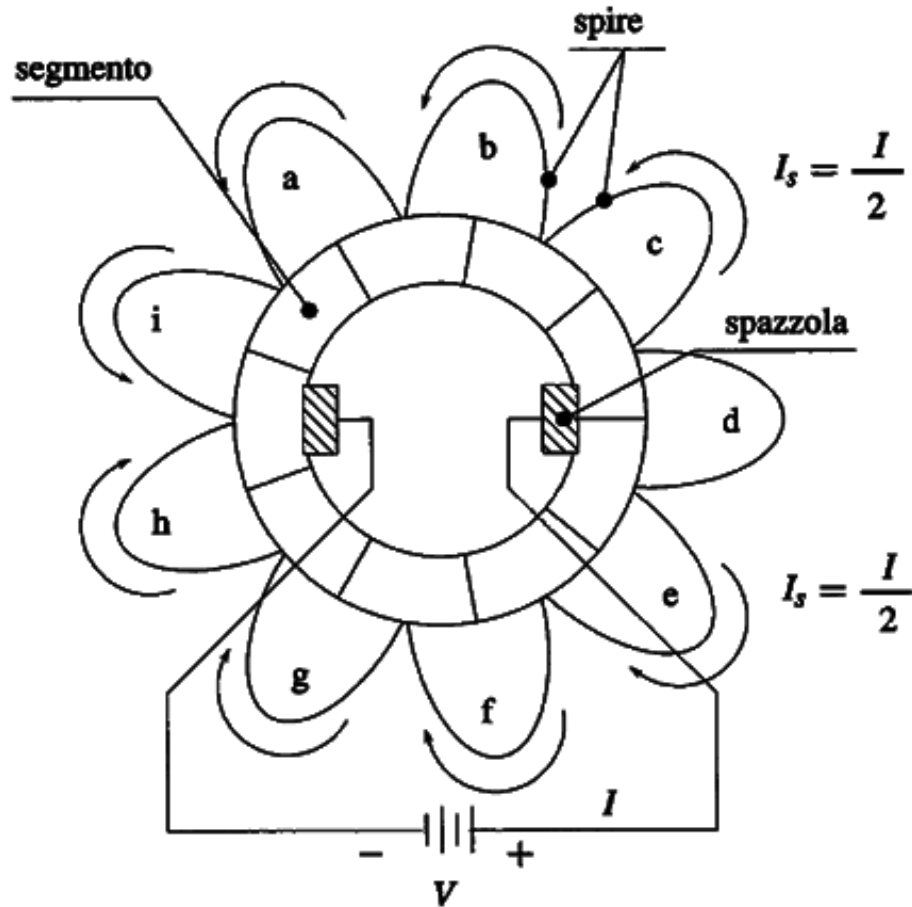


- **Spazzole:** vengono mantenute a contatto con il collettore durante la rotazione e consentono il collegamento con un circuito elettrico esterno che fornisce potenza ai circuiti di armatura
 - Ne sono usate una o più coppie
 - Sono realizzate in grafite o metallo prezioso
- Le aree dove non esiste flusso sono chiamate **zone neutre**, che si distinguono dalle aree dove il flusso esiste, chiamate **zone polari**
 - Le spazzole vengono normalmente poste nelle zone neutre
- Esistono diversi modi di avvolgere le spire e connettere i terminali ai risers

motori elettrici a magneti permanenti a corrente continua

Struttura del motore a corrente continua

Schema di collegamento a nove spire e sue connessioni:



motori elettrici a magneti permanenti a corrente continua

Struttura del motore a corrente continua

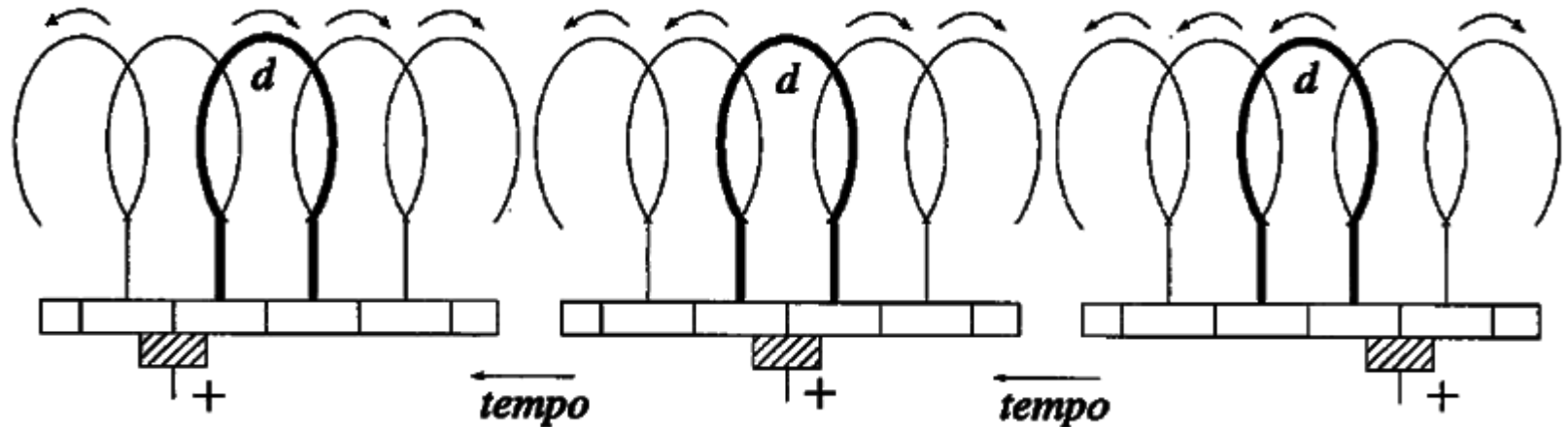


- Le spire vengono connesse ad anello l'una all'altra → la corrente d'armatura si ripartisce equamente in due circuiti
 - La corrente fluisce verso l'alto sotto il polo nord e verso il basso nella parte di spira sotto il polo sud
 - Il numero di conduttori in serie è circa la metà del totale
 - La corrente I_s che fluisce in ogni conduttore è pari alla metà della corrente impressa dal circuito di alimentazione $I_s = I_a / 2$
- Invertendo la polarità della sorgente di potenza la direzione della corrente e la coppia si invertono

motori elettrici a magneti permanenti a corrente continua

Struttura del motore a corrente continua

- Il sistema spazzole-collettore consente la commutazione del verso di percorrenza della corrente in una spira:



- Ogni spazzola è solitamente a contatto con due o più segmenti del collettore → una o più spire sono cortocircuitate da ogni spazzola

motori elettrici a magneti permanenti a corrente continua

Struttura del motore a corrente continua



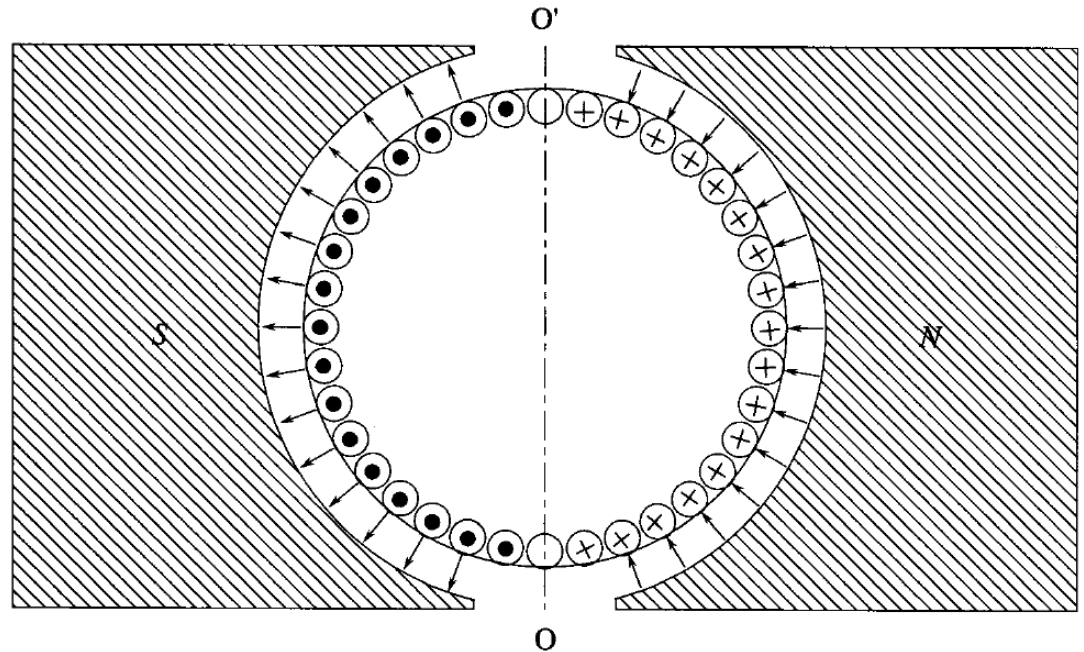
- Supponendo che il rotore ruoti in senso orario, la corrente scorre in senso antiorario nella spira d prima che questa venga cortocircuitata dalla spazzola connessa al morsetto positivo
- Dopo la cortocircuitazione il verso della corrente risulta invertito
- Nella spira sottoposta a commutazione, non essendo i conduttori di questa sottoposti al flusso del campo magnetico (zona neutra), non viene indotta alcuna forza elettromotrice

motori elettrici a magneti permanenti a corrente continua

Costante di coppia

- **Costante di coppia**

- La distribuzione della corrente di armatura in un motore con una coppia di poli e illustrato a lato
- La corrente fluisce nei conduttori a destra dell'asse OO' in verso entrante e nei conduttori alla sinistra in verso uscente
- *Nota:* il verso delle correnti non varia rispetto ai poli magnetici al variare dell'angolo per effetto del collettore

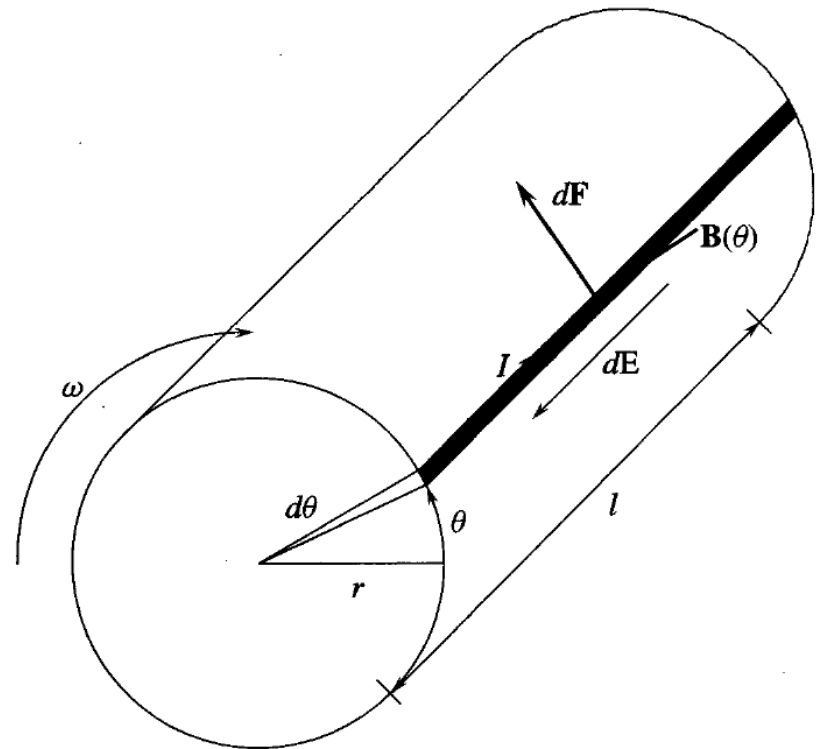


motori elettrici a magneti permanenti a corrente continua

Costante di coppia

- *Ipotesi*: conduttori uniformemente distribuiti lungo le generatrici del rotore → **densità angolare uniforme**
- Detto n il numero totale di conduttori, il numero dn di conduttori compreso in un angolo infinitesimo $d\theta$ sarà

$$dn = \frac{N}{2\pi} d\theta$$



motori elettrici a magneti permanenti a corrente continua

Costante di coppia



- Il **campo magnetico** al traferro $B(\theta)$ si assume *costante* in modulo, direzione e verso lungo ciascuna generatrice del rotore (di lunghezza l) e diretto in senso *radiale* rispetto all'asse del rotore → ciascun conduttore sarà soggetto ad una forza diretta in direzione tangenziale rispetto alla circonferenza esterna del rotore
- Il modulo della forza dF esercitato su dn conduttori adiacenti sarà dato da

$$dF = I_s l B(\theta) \frac{N}{2\pi} d\theta$$

motori elettrici a magneti permanenti a corrente continua

Costante di coppia

che si traduce in un contributo di coppia $d\tau$ all'asse del motore pari a

$$d\tau = \frac{N}{2\pi} I_s l B(\theta) r d\theta$$

dove r (*raggio di indotto al traferro*) indica la distanza di ogni conduttore dall'asse di rotazione

– La **coppia complessiva** τ_m è ottenuta per integrazione

$$\tau_m = 2 \int_{-\pi/2}^{\pi/2} d\tau = \frac{N}{\pi} I_s \underbrace{\int_{-\pi/2}^{\pi/2} l B(\theta) r d\theta}_{\phi_m}$$

motori elettrici a magneti permanenti a corrente continua

Costante di coppia

- Ricordando che il flusso magnetico ϕ_m che attraversa il rotore si calcola come:

$$\phi_m = \int_{-\pi/2}^{\pi/2} lB(\theta)r d\theta$$

e considerato che la corrente I_s che percorre ciascun conduttore è pari alla metà della corrente I impressa dal circuito esterno, si ha:

$$\tau_m = \frac{N\phi_m}{2\pi} I = K_t I$$

dove K_t è la **costante di coppia** del motore

motori elettrici a magneti permanenti a corrente continua

Costante di forza controelettromotrice



- **Costante di forza controelettromotrice**

- In accordo con la legge del flusso tagliato, la forza controelettromotrice complessiva indotta nei dn conduttori (collegati in serie) si calcola come

$$dE = \frac{N}{2\pi} \omega r B(\theta) l d\theta$$

dove ω indica la velocità angolare del rotore

Nota: la polarità di dE dipende dalla direzione del campo $B(\theta)$ e della velocità con cui i conduttori tagliano le linee di campo

motori elettrici a magneti permanenti a corrente continua

Costante di forza controelettromotrice



- La forza controelettromotrice E raccolta complessivamente ai morsetti del motore sarà quindi determinata integrando i contributi di tutti i conduttori *in serie* sotto uno stesso polo (pari alla metà del numero totale):

$$E = \frac{N}{2\pi} \omega \underbrace{\int_{-\pi/2}^{\pi/2} lB(\theta)r d\theta}_{\phi_m} = \frac{N\phi_m}{2\pi} \omega = K_e \omega$$

dove K_e è detta **costante di forza controelettromotrice**

- Si noti che $K_e = K_t$ **strutturalmente!**

motori elettrici a magneti permanenti a corrente continua

Modello dinamico



- **Modello dinamico**

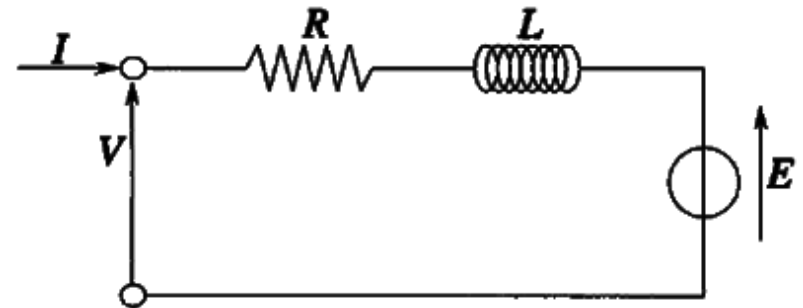
- La coppia generata da un motore a corrente continua è proporzionale alla corrente di armatura → per completare il modello del motore bisogna introdurre l'equazione che governa la *dinamica della corrente*
- La tensione applicata alle spazzole si ripartisce in tre contributi dovuti:
 - Alla caduta di tensione di tipo *resistivo* sugli avvolgimenti
 - All'*induttanza* degli avvolgimenti
 - Alla *forza contro elettromotrice* raccolta alle spazzole

motori elettrici a magneti permanenti a corrente continua

Modello dinamico

– Circuito equivalente di riferimento, dove

- R rappresenta la resistenza di armatura
- L rappresenta l'induttanza di armatura
- E la forza controelettromotrice raccolta alle spazzole



– Il **modello dinamico complessivo** riferito alle grandezze elettriche sarà definito dalle seguenti relazioni:

$$V = RI + L \frac{dI}{dt} + E$$

$$E = K_e \omega$$

$$\tau_m = K_t I$$

motori elettrici a magneti permanenti a corrente continua

Caratteristica coppia-velocità



- **Caratteristica coppia-velocità**

- Se la velocità ω è *costante* $\rightarrow$ grandezze elettriche costanti

$$V = RI + K_e \omega$$

$$\tau_m = \frac{K_t}{R} (V - K_e \omega)$$

- La seconda equazione rappresenta la *relazione coppia/velocità*

- La coppia decresce *linearmente* all'aumentare della velocità, con *pendenza* indipendente dalla tensione applicata e dalla velocità di rotazione $\rightarrow$ pendenza pari a $-K_t K_e / R$, dipende solo dalle caratteristiche elettriche e costruttive del motore

motori elettrici a magneti permanenti a corrente continua

Caratteristica coppia-velocità

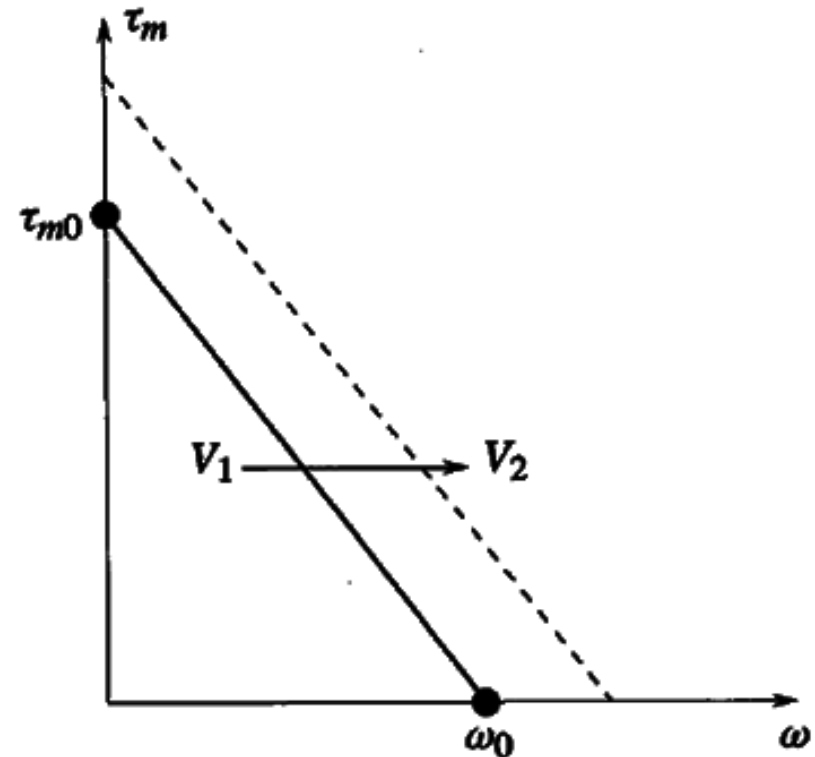
- Sulla *caratteristica coppia-velocità* possono essere individuate due importanti quantità:

- **Coppia di spunto** τ_{m0}

$$\tau_{m0} = \frac{K_t}{R} V$$

- **Velocità a vuoto** ω_0

$$\omega_0 = \frac{V}{K_e}$$



motori elettrici a magneti permanenti a corrente continua

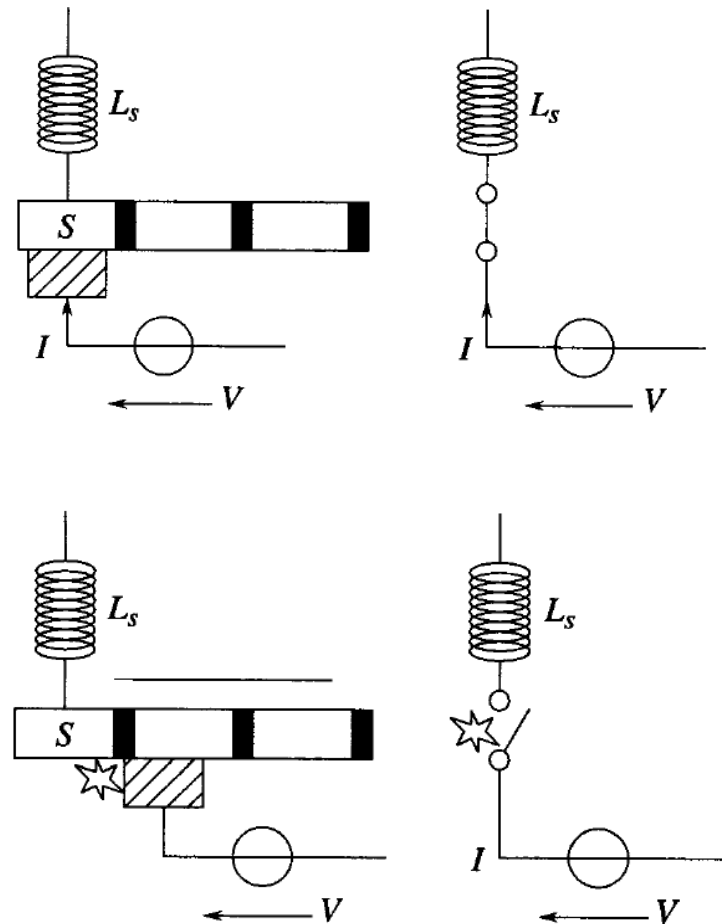
Pregi e difetti

- *Pregi:*
 - La coppia è legata alla corrente di armatura da una costante di proporzionalità
 - Il modello dinamico della corrente al variare della tensione impressa alle spazzole è lineare
 - la coppia può essere modulata regolando la corrente di armatura agendo sulla tensione di alimentazione
- *Difetti:*
 - L'utilizzo del collettore comporta problemi di *scintillamenti* e *usure*, legati proprio alla sua funzione specifica di dispositivo di commutazione della corrente nelle spire

motori elettrici a magneti permanenti a corrente continua

Pregi e difetti

- Il distacco fra una spazzola e una lamella può essere modellato mediante l'apertura di un interruttore
 - Una spira può essere modellata come una semplice induttanza L_s
 - All'apertura dell'interruttore si determinerà una **brusca variazione di corrente** in L_s
 - Si genererà una **sovratensione** pari a $L_s \, dI/dt$ in grado di perforare il dielettrico posto tra i morsetti dell'interruttore → **scarica elettrica (scintilla)**



motori elettrici a magneti permanenti a corrente continua

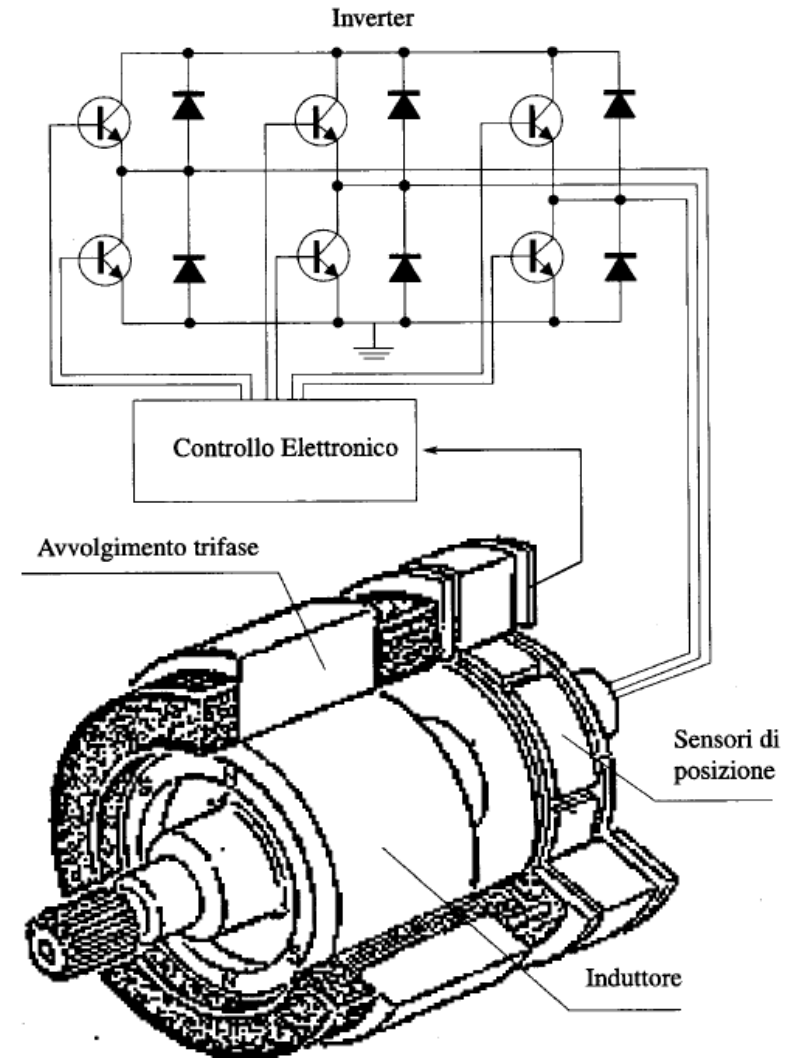
Pregi e difetti

- Al momento del distacco fra lamelle e spazzole si potranno originare delle **scintille**
 - Scintille di debole entità sono ben sopportate dalle spazzole e dai segmenti del collettore
 - Scintille di maggiore intensità e ripetute nel tempo possono provocare la **bruciatura delle spazzole** e la **volatilizzazione del metallo delle lamelle** → *usura progressiva del collettore*
 - Sarà necessario sostituire periodicamente le spazzole e rettificare il collettore
 - Le spire vengono progettate in modo da ridurre il più possibile l'induttanza
- *Alternativa* → Motori **brushless** (“senza spazzole”)
 - Non sono soggetti a questi problemi ma sono più complessi

motori elettrici a magneti permanenti brushless

Principio di funzionamento

- **Motore brushless** (senza spazzole) è costituito da
 - **Rotore:** su cui sono alloggiati i magneti permanenti (*esempio:* terre rare)
 - **Statore:** su cui vengono disposti gli avvolgimenti di fase (in genere tre)



motori elettrici a magneti permanenti brushless

Principio di funzionamento



- **Fasi:** sono alimentate alternativamente, in modo che il campo magnetico generato dalle relative correnti si mantenga sempre *ortogonale* e *sincrono* al campo generato dai magneti di rotore
- **Inverter:** realizza il sincronismo *commutando* le correnti negli avvolgimenti di statore in funzione della posizione angolare del rotore
 - Svolge la stessa funzione svolta dal sistema spazzole-collettore in un motore a corrente continua → **commutatore elettronico** anziché meccanico
- **Sensore di posizione:** rileva la posizione del rotore da fornire all'inverter

motori elettrici a magneti permanenti brushless

Principio di funzionamento



- *Nota:* la **commutazione elettronica** e la presenza di **diodi di ricircolo** consentono di eliminare il problema dell'usura legata alla scarica dell'energia elettromagnetica immagazzinata degli avvolgimenti

motori elettrici a magneti permanenti brushless

Principio di funzionamento

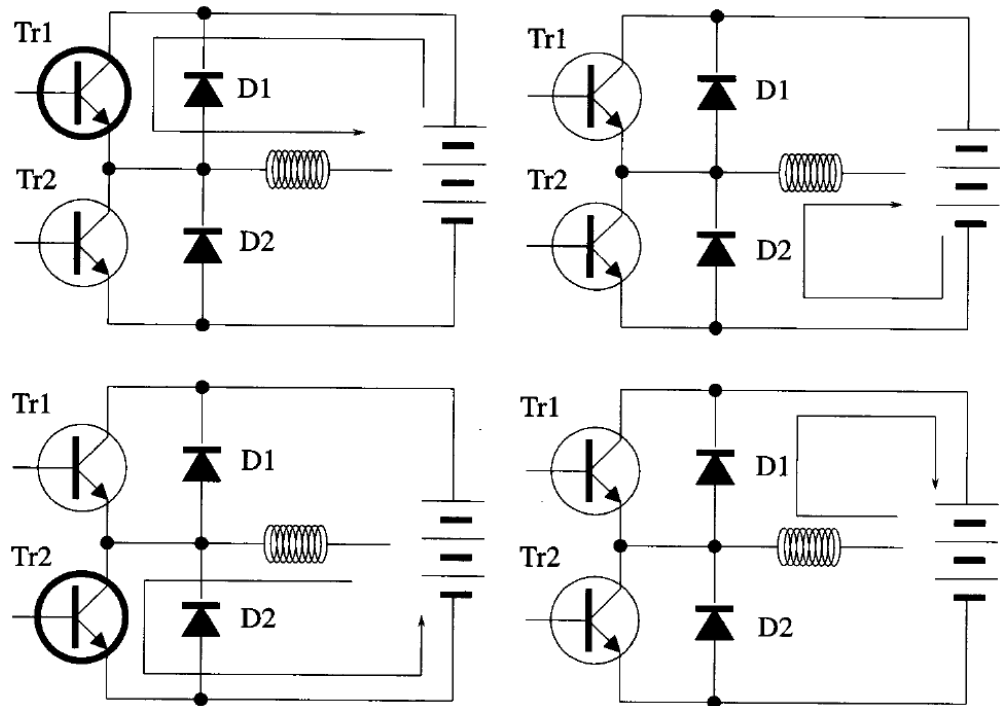


- **Funzionamento:**

- Le coppie **Tr1-D1** e **Tr2-D2** consentono di alimentare un avvolgimento nei due sensi

- I **diodi** consentono di evitare che la *sovratensione*

LdI/dt che viene a stabilirsi tra collettore ed emettitore al momento dell'interdizione dei transistor possa danneggiarli



motori elettrici a magneti permanenti brushless

Principio di funzionamento



- *Esempio:* quando il transistor **Tr1** viene spento, la corrente non si annulla di colpo ma può continuare a circolare attraverso il diodo **D2** fino ad annullarsi
- Un segmento del collettore nel motore a corrente continua equivale all'insieme di due transistor e due diodi (colonna) nell'inverter del motore brushless
 - Se questa equivalenza fosse mantenuta rispetto al numero di segmenti si otterrebbero dispositivi elettronici troppo complessi e costosi → si utilizzano **avvolgimenti bifase** o **trifase**
 - *Nota:* nei motori a corrente continua si devono utilizzare numerosi avvolgimenti per ridurre l'induttanza dei singoli circuiti, che altrimenti sarebbe troppo elevata e renderebbe impossibile la commutazione

motori elettrici a magneti permanenti brushless

Principio di funzionamento



- **Conversione elettromeccanica**

- È equivalente al caso del motore a corrente continua a patto di sostituire l'angolo meccanico θ con l'**angolo elettrico** α , proporzionale all'angolo meccanico tramite il numero di **coppie polari** p :

$$\alpha = p \theta$$

- La *derivata* rispetto al tempo dell'angolo elettrico definisce la **frequenza di commutazione**
 - Tanto più sono numerose le coppie polari p (in genere tre), tanto più alta è la frequenza di commutazione delle correnti negli avvolgimenti

motori elettrici a magneti permanenti brushless

Principio di funzionamento



- *Vantaggi**:
 - Migliore rapporto peso-potenza
 - Dimensioni molto ridotte
 - Bassa inerzia ed alta banda passante (piccole costanti di tempo elettriche)
 - Elevate accelerazioni meccaniche
 - Ottima affidabilità
 - Alta velocità di rotazione
 - Minima manutenzione
 - Idoneità al funzionamento anche in ambienti ostili

() Rispetto ai motori a corrente continua*

motori elettrici a magneti permanenti brushless

Principio di funzionamento



- *Svantaggi**:
 - Necessità di un sensore di posizione
 - Complessa logica di gestione della commutazione
- **Costo elevato**

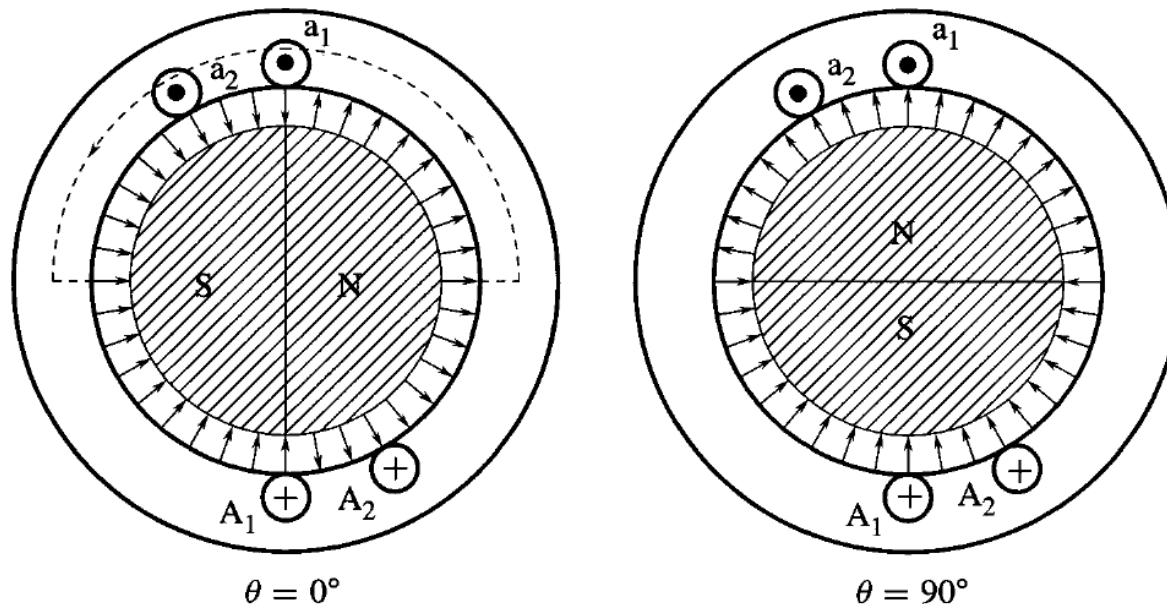
(*) *Rispetto ai motori a corrente continua*

motori elettrici a magneti permanenti brushless

Motore brushless trapezio



- **Motore brushless trapezio** costituito da



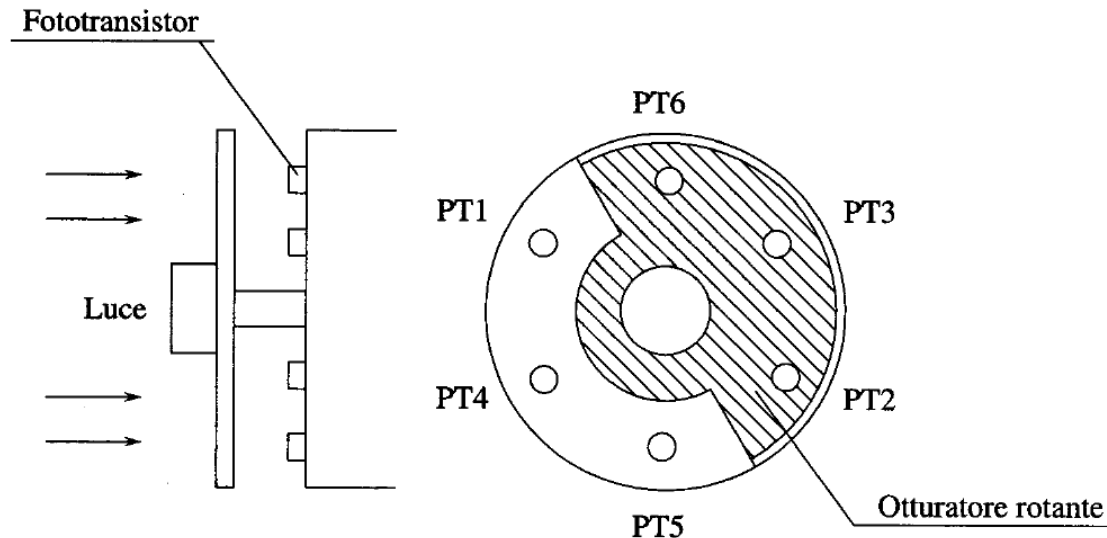
- **Rotore:** contiene un magnete permanente a 2 poli
- **Statore:** presenta 12 cave ed un avvolgimento trifase

motori elettrici a magneti permanenti brushless

Motore brushless trapezio



- Si hanno due cave per polo e per fase $\rightarrow$ 2 coppie polari ($p = 2$)
 - Ogni fase è costituita da due bobine adiacenti, ognuna a N spire, i cui assi formano un angolo di 30°
 - Ogni cava contiene N conduttori di una sola fase
- **Sensore di posizione:** la posizione angolare del rotore viene rilevata mediante 6 sensori ottici (*fototransistor*)

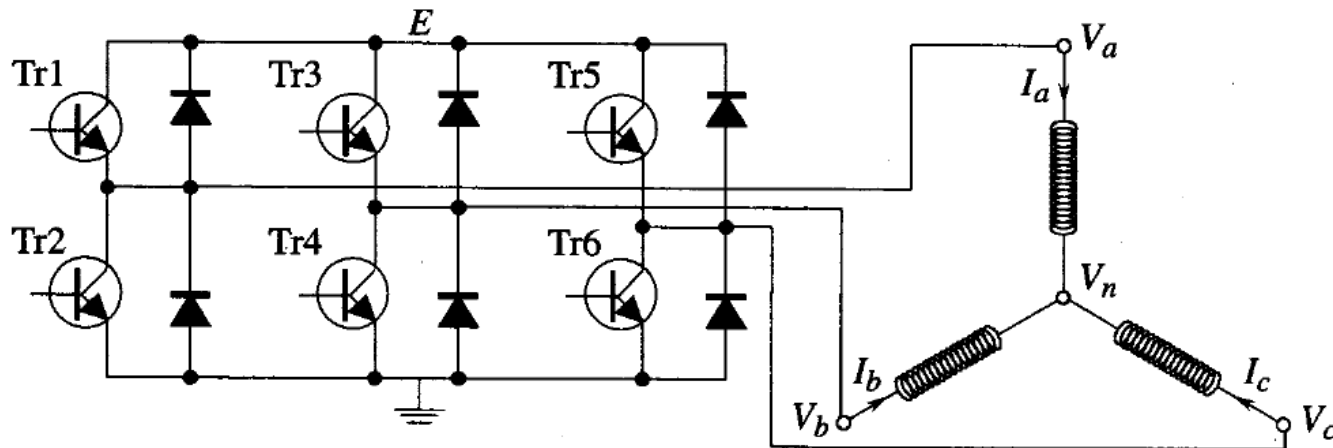


motori elettrici a magneti permanenti brushless

Motore brushless trapezio



- I sei fototransistor PT1-PT6 sono disposti a 60° fra loro
 - Sono sottoposti sequenzialmente, tramite un *otturatore* che ruota stabilmente al rotore, ad un fascio di luce, prodotto da una apposita sorgente luminosa
- **Inverter**: costituito da 6 transistor, la cui accensione è comandata dai segnali ottenuti dai fototransistor



motori elettrici a magneti permanenti brushless

Motore brushless trapezio



- **Coppia motrice**

- La potenza elettrica assorbita ai morsetti della macchina ed effettivamente convertita in potenza meccanica è data dal prodotto della corrente che circola negli avvolgimenti di statore per la forza controelettromotrice raccolta ai loro capi → nel caso di tre fasi distinte (a, b, c)

$$P_m = E_a I_a + E_b I_b + E_c I_c$$

Ricordando l'espressione della potenza meccanica:

$$P_m = \tau_m \omega$$

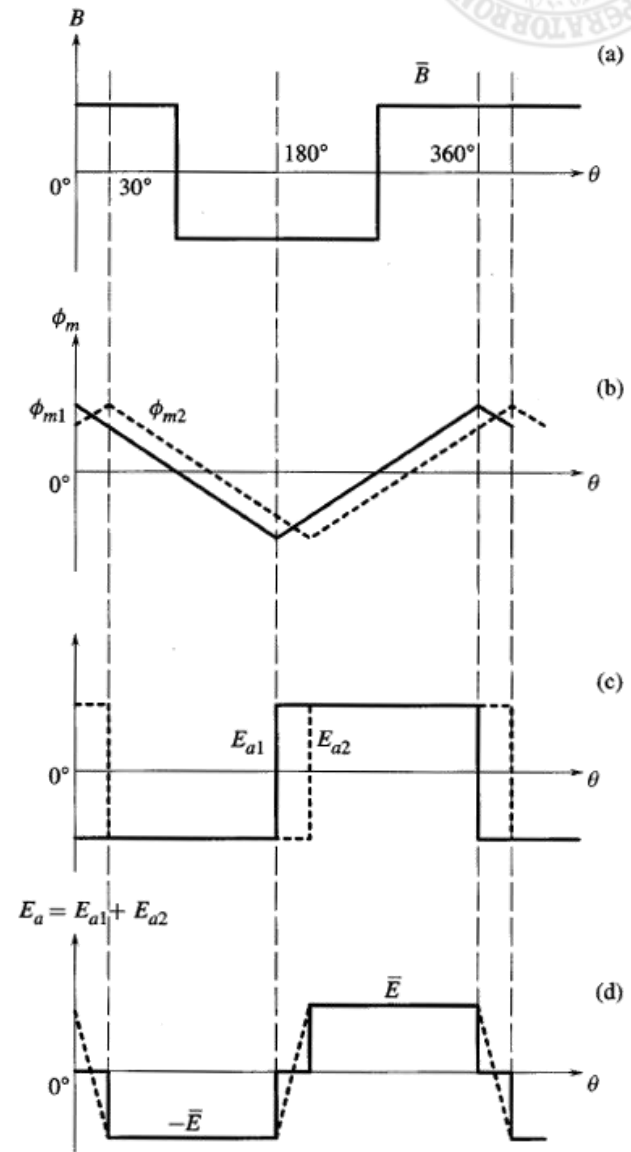
si ottiene

$$\tau_m = \frac{E_a I_a + E_b I_b + E_c I_c}{\omega}$$

motori elettrici a magneti permanenti brushless

Motore brushless trapezio

- Le **forze contro elettromotrici indotte** si possono calcolare derivando rispetto al tempo il flusso concatenato con ciascuna fase
 - Supponendo una forma ad *onda quadra* per la densità del flusso al traferro in funzione della posizione angolare del rotore, il flusso concatenato con la bobina a_1A_1 varia linearmente con la posizione del rotore
 - Il massimo ϕ_{max} del flusso concatenato (in valore assoluto) si ha per $\theta = 0^\circ$ (positivo) e per $\theta = 180^\circ$ (negativo)



motori elettrici a magneti permanenti brushless

Motore brushless trapezio



- Integrando il campo magnetico $B(\theta)$ lungo il traferro si ottiene:

$$\phi_{\max} = Nr l \int_{-\pi/2}^{\pi/2} B(\theta) d\theta = Nr l \bar{B} \pi$$

dove r è il raggio interno dello statore, l è la lunghezza assiale sia del rotore che dello statore e $\bar{B}$ è il valore assunto dal campo magnetico al traferro

- La forza controelettromotrice indotta nella bobina $a_1 A_1$ risulta

$$E_{a1} = - \frac{d\phi_{m1}}{dt} = - \frac{d\phi_{m1}}{d\theta} \frac{d\theta}{dt} = -\omega \frac{d\phi_{m1}}{d\theta}$$

- Esprimendo la derivata del flusso concatenato rispetto all'angolo in funzione del flusso concatenato massimo si ottiene per la forza controelettromotrice una forma ad onda quadra di ampiezza

$$\bar{E}_{a1} = \frac{2\phi_{\max}}{\pi} |\omega|$$

motori elettrici a magneti permanenti brushless

Motore brushless trapezio



- La forza controelettromotrice indotta nella bobina a_2A_2 ha la medesima ampiezza ma è sfasata di 30°
- Quando le due bobine vengono collegate in *serie* si ottiene una **forma trapezoidale** per la forza controelettromotrice:
Avvolgimenti *reali* $\rightarrow$ due fasi sempre in conduzione $\rightarrow$ la forza controelettromotrice assume una forma trapezoidale per ogni fase di ampiezza pari a

$$\overline{E}_{a1} = \frac{4\phi_{\max}}{\pi} |\omega|$$

- Dal profilo di forza controelettromotrice e dall'espressione della coppia motrice si deduce la forma d'onda da imporre alle correnti di fase al fine di ottenere una coppia indipendente dalla posizione angolare del rotore

motori elettrici a magneti permanenti brushless

Motore brushless trapezio

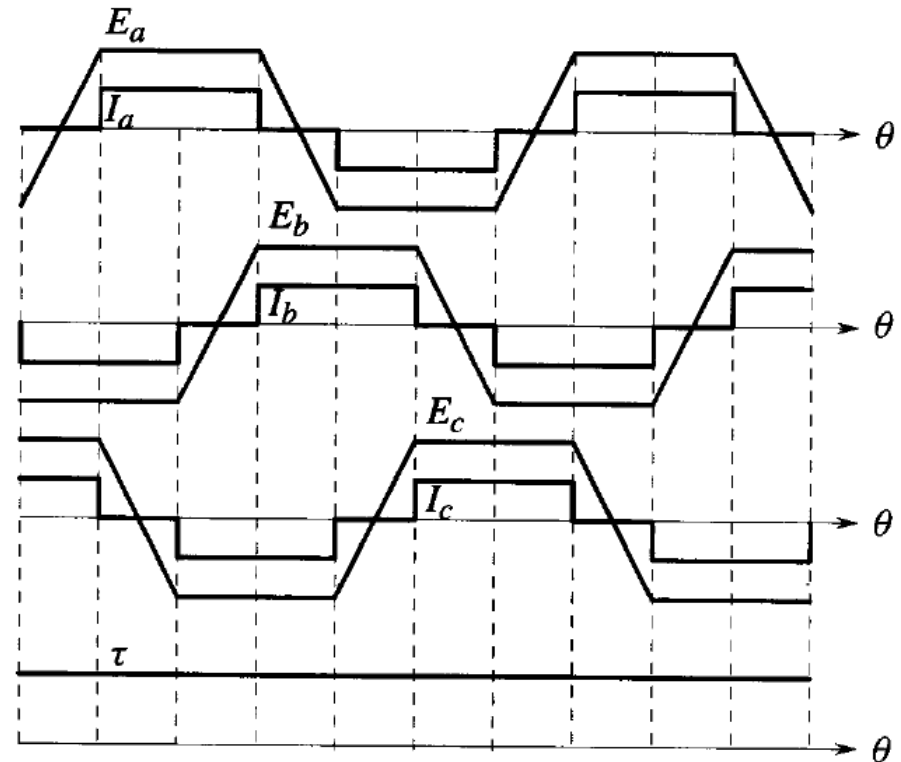


In ogni istante ci sono due fasi in conduzione
→ durante i 120° di conduzione di una fase, il **bilancio di potenza** si esprime come

$$\tau_m \omega = 2E_i I_i$$

con $i = a, b, c$, da cui si può ricavare l'espressione della **coppia**

$$\tau_m = \frac{2E_i I_i}{\omega} = 8 \frac{\phi_{\max}}{\pi} I = K_t I$$



motori elettrici a magneti permanenti brushless

Motore brushless trapezio



- Con riferimento ad un **collegamento a stella** delle fasi, cioè

$$I_a + I_b + I_c = 0$$

il modello dinamico del motore brushless trapezoidale relativo alle grandezze elettriche è definito dalle equazioni

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} R & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & R \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} + \frac{d}{dt} \left(\begin{bmatrix} L_a & M_{ab} & M_{ac} \\ M_{ba} & L_b & M_{bc} \\ M_{ca} & M_{cb} & L_c \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} \right) + \begin{bmatrix} E_a \\ E_b \\ E_c \end{bmatrix} + \begin{bmatrix} V_n \\ V_n \\ V_n \end{bmatrix}$$

dove V_i sono le tensioni applicate alle fasi (riferimento: massa dell'inverter), V_n è il potenziale del punto neutro (centro stella), R è la resistenza di fase, E_i è la forza contro elettromotrice indotta e L_i, M_{ij} sono rispettivamente le auto e le mutue induttanze di fase

motori elettrici a magneti permanenti brushless

Motore brushless trapezio



Ipotesi: riluttanze del motore costanti con l'angolo → **macchina isotropa** → L_i uguali tra loro, M_{ij} uguali tra loro → modello risulta

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = R \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} + L \frac{d}{dt} \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} + \begin{bmatrix} E_a \\ E_b \\ E_c \end{bmatrix} + \begin{bmatrix} V_n \\ V_n \\ V_n \end{bmatrix}$$

dove $L = L_i - M_{ij}$ (la mutua induttanza $M_{ij} < 0$)

Nota: il modello è costituito da tre equazioni differenziali e da una equazione algebrica di vincolo tra le correnti, imposta dal collegamento a stella → I_a, I_b, I_c, V_n incognite, V_a, V_b, V_c variabili esogene da usare per imporre la forma d'onda delle correnti

motori elettrici a magneti permanenti brushless

Motore brushless sinusoidale

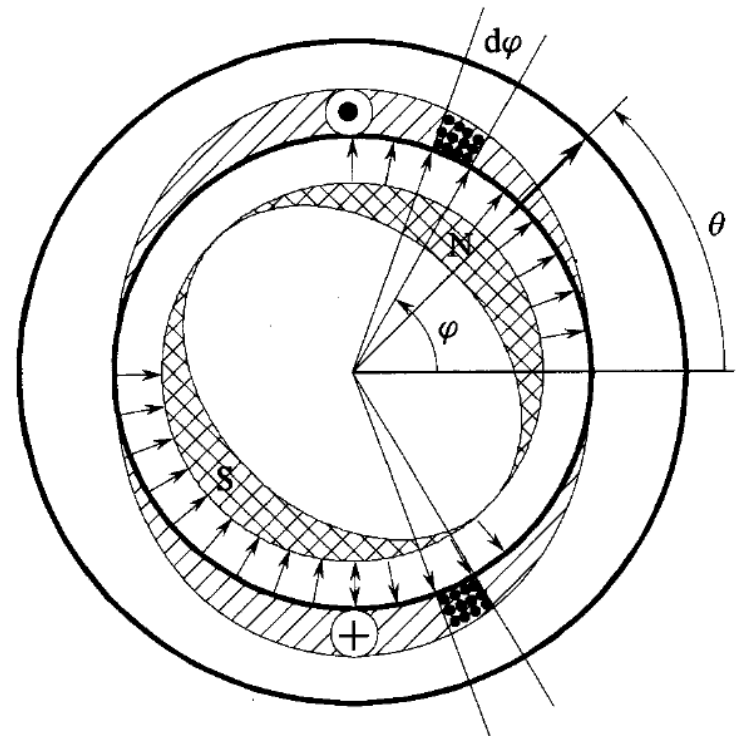


- **Motore brushless sinusoidale**

- Differisce da uno trapezio nella *funzione di forma* ottenuta per le forze controelettromotrici indotte
- In entrambi i casi, le forze controelettromotrici si possono esprimere come il prodotto della velocità angolare per una **funzione di forma** $K_i(\theta)$, cioè

$$E_i = \omega K_i(\theta)$$

- *Esempio*: struttura di un motore brushless sinusoidale (una sola fase)



motori elettrici a magneti permanenti brushless

Motore brushless sinusoidale



- Sagomando opportunamente i magneti permanenti posti sul rotore è possibile ottenere una **distribuzione sinusoidale del campo magnetico** → la direzione di massima ampiezza del campo magnetico ruota alla velocità di rotazione del rotore
- Si assume la direzione di massima ampiezza del campo come *asse di riferimento mobile* per la misura degli angoli si ha

$$B(\varphi, \theta) = \bar{B} \cos(\varphi - \theta)$$

dove l'angolo φ individua un generico punto lungo il traferro e l'angolo θ individua la rotazione del rotore

- Si assume una **distribuzione sinusoidale per i conduttori di ogni singola fase** → in un angolo infinitesimo $d\varphi$ sono contenuti un numero di conduttori dn pari a

$$dn = \frac{N_s}{2} \sin \varphi d\varphi$$

N_s è il numero di spire

motori elettrici a magneti permanenti brushless

Motore brushless sinusoidale



- Supponendo che i conduttori di ritorno siano individuati dall'angolo $-\varphi$, il flusso ϕ_m concatenato con la spira costituita dai dn conduttori risulta pari a

$$\phi_m(\varphi, \theta) = \int_{-\varphi}^{\varphi} B(\sigma, \theta) r l d\sigma = 2\bar{B}rl \sin \varphi \cos \theta$$

- La forza contro elettromotrice dE indotta nella spira costituita dai dn conduttori (in serie) sarà data da

$$dE = -\frac{d\phi_m}{dt} dn = \bar{B}rl \omega N_s \sin^2 \varphi \sin \theta d\varphi$$

- La **forza contro elettromotrice complessiva** risulta

$$E = \int_0^{\pi} dE = \omega \frac{\bar{B}rl N_s \pi}{2} \sin \theta = \omega K \sin \theta$$

Nota: Lo stesso risultato si poteva ottenere con la *legge del flusso tagliato*

motori elettrici a magneti permanenti brushless

Motore brushless sinusoidale



- Considerando il caso generale di p coppie polari e ricordando che le fasi sono sfasate di $2\pi/3$, la **funzione di forma** risulterà

$$K_a(\alpha) = pK \sin(p\theta) = pK \sin \alpha$$

$$K_b(\alpha) = pK \sin(p\theta - 2\pi/3) = pK \sin(\alpha - 2\pi/3)$$

$$K_c(\alpha) = pK \sin(p\theta - 4\pi/3) = pK \sin(\alpha - 4\pi/3)$$

- **Per ottenere una coppia costante rispetto alla posizione angolare del rotore è necessario imporre la stessa funzione di forma alle correnti di fase:**

$$I_a = I_a(\alpha) = I \sin \alpha$$

$$I_b = I_b(\alpha) = I \sin(\alpha - 2\pi/3)$$

$$I_c = I_c(\alpha) = I \sin(\alpha - 4\pi/3)$$

motori elettrici a magneti permanenti brushless

Motore brushless sinusoidale



La **coppia** corrispondente risulterà

$$\begin{aligned}\tau_m &= pKI \sin^2 \alpha + pKI \sin^2(\alpha - 2\pi/3) + pKI \sin^2(\alpha - 4\pi/3) \\ &= \frac{3}{2} pKI = K_t I\end{aligned}$$

- La dinamica delle grandezze elettriche risulterà descritta dalle stesse equazioni del motore brushless trapezio

motori elettrici a magneti permanenti brushless

Motore brushless sinusoidale



- ***Ripple di coppia***

- Nelle *macchine reali* la generazione di coppia non è dovuta solo alla coppia di allineamento, generata dall'interazione fra il campo magnetico dovuto ai magneti permanenti e quello generato dalle correnti di fase negli avvolgimenti
 - Variazione dell'autoinduttanza degli avvolgimenti di fase in funzione dell'angolo motore → **coppia di riluttanza**
 - Interazione fra il campo prodotto dai magneti permanenti e i denti delle cave presenti sullo statore → **coppia di cogging**

motori elettrici a magneti permanenti brushless

Motore brushless sinusoidale



- Transitori di commutazione delle correnti + imperfezioni delle forme d'onda di corrente e di forza controelettromotrice → **coppia di allineamento** dipendente dall'angolo $\theta \rightarrow \tau_m(\theta)$
- si generano dei **disturbi pulsanti di coppia (*ripple*)**
 - Insieme di pulsazioni indesiderate nella forma d'onda della coppia
 - Degradano la *regolarità del moto* del sistema, soprattutto alle *basse velocità*
 - Possono eccitare le *risonanze meccaniche* della struttura cui il motore è applicato



controllo di corrente

Amplificatori di potenza

- **Amplificatore di potenza**
 - Segnale fornito dai regolatori è un segnale a bassa potenza non compatibile con le esigenze della movimentazione
 - Si interpone uno stadio di amplificazione di potenza tra il regolatore ed il motore
 - Fornisce una tensione di *valore medio* proporzionale all'uscita del regolatore e adeguate correnti
- Esistono due categorie di amplificatori di potenza
 - Amplificatori *lineari* (classe A)
 - Amplificatori *switching* (classe D)

controllo di corrente

Amplificatori di potenza lineari



- **Amplificatore lineari**

- Consentono di applicare tensioni o correnti al motore proporzionali alla tensione di comando
- Sono i più **adatti ai fini del controllo**
 - Caratteristica lineare in un ampio range di valori dell'uscita
 - Larga banda
 - Bassa emissione di disturbi elettromagnetici
- **Potenza dissipata** può essere dello stesso ordine di grandezza di quella massima erogabile → limitati ad applicazioni che richiedono *bassi livelli di potenza*

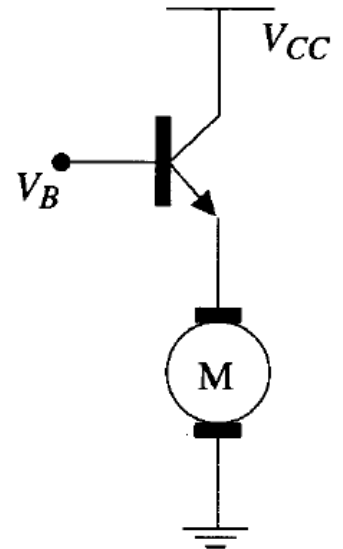


controllo di corrente

Amplificatori di potenza switching

- **Amplificatore switching**

- Modulano il *valore medio* della tensione applicata al carico (motore) e non il valore istantaneo
 - Modulano la durata degli intervalli di apertura e di chiusura del circuito di collegamento tra un alimentatore a corrente continua ed il carico
- Si utilizzano amplificatori **switching a transistor**
 - Necessità di regolare con continuità la velocità in range estesi (da zero a migliaia di giri al minuto)
 - Banda dei regolatori di corrente $200 \div 300 \text{ Hz} \rightarrow$ necessità di elevate frequenze di commutazione ($5 \div 30 \text{ kHz}$)





controllo di corrente

Amplificatori di potenza switching

- *Modulazione*: La tensione V_B è modulata in modo da far commutare il transistor alternativamente dallo stato di **interdizione** a quello di **saturazione** e viceversa → in uscita viene prodotta una forma d'onda a **treno di impulsi** di durata finita e variabile → il valore medio su un periodo è proporzionale alla durata dell'impulso
- **Dissipazione** sul transistor ridotta → corrente nel transistor nulla (interdizione) o caduta di tensione bassa (saturazione, $V_{CE} \cong 0.2V$)
- **Carico tipicamente passa basso** + utilizzo di elevate frequenza di commutazione → la forma d'onda è filtrata dall'induttanza degli avvolgimenti e dall'inerzia del rotore → comportamento lineare con ottima approssimazione

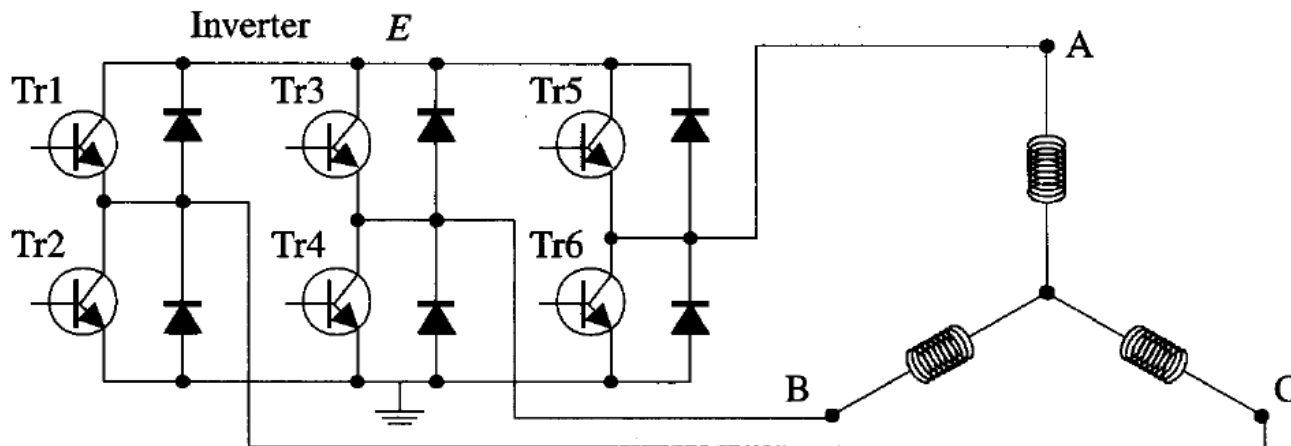
controllo di corrente

Amplificatori di potenza switching – Inverter



- **Inverter**

- Configurazione delle valvole più adottata al pilotaggio dei motori trifase → consente la circolazione di corrente in entrambi i sensi → è possibile invertire il moto del motore
- Ogni fase è alternativamente collegata alla tensione di alimentazione o a massa





controllo di corrente

Amplificatori di potenza switching – Inverter

- Classificazione degli Inverter in funzione di:
 - Caratteristiche delle *valvole* utilizzate
 - *Topologia* della parte di potenza
- Classificazione delle valvole più diffuse in base alla:
 - Modalità di commutazione: *forzata* o *non forzata*
 - Frequenza di commutazione
- Valvole a commutazione forzata
 - Comandati da un segnale esterno
 - *Tipologie di dispositivi*: IGBT, GTO, BJT, MOSFET
- Valvole a commutazione non forzata
 - Spegnimento quando si annulla la corrente
 - *Tipologie di dispositivi*: SCR e tiristori

controllo di corrente

Amplificatori di potenza switching – Inverter CSI



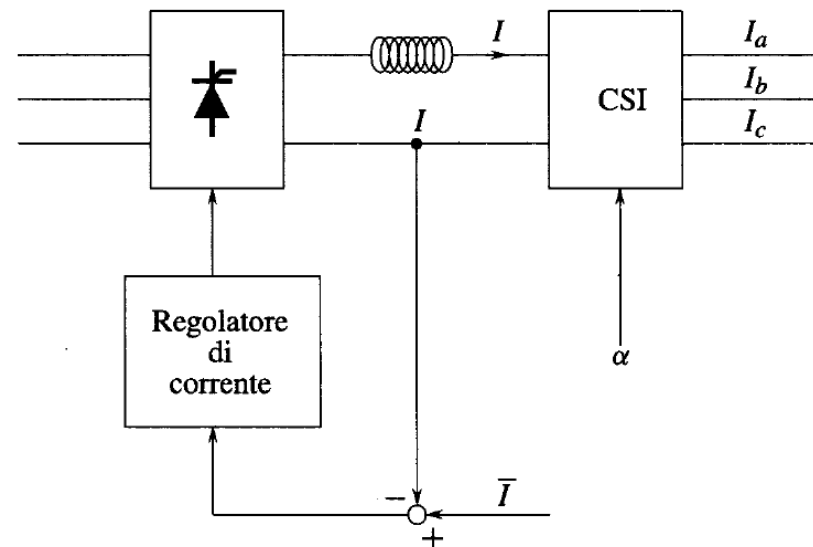
– Classificazione degli inverter in base alla *topologia* della parte di potenza:

- Inverter a *corrente* impressa
- Inverter a *tensione* impressa

– **Inverter a corrente impressa (CSI: *Current Source Inverter*)**

- La regolazione di corrente avviene esternamente all'inverter
- Azionamento costituito da:

- *Stadio di ingresso*:
convertitore statico
alimentato dalla rete trifase
a tensione e frequenza fisse
che alimenta il circuito
intermedio in corrente
continua





controllo di corrente

Amplificatori di potenza switching – Inverter CSI

- *Induttanza*: disaccoppia la rete a frequenza fissa dell'alimentazione da quella a frequenza variabile del carico
- *Stadio di uscita*: costituito da un convertitore statico, alimentato dal circuito intermedio, fornisce energia al carico a frequenza variabile
- Caratteristiche di un CSI
 - *Implementazione semplice*: necessita solo del regolatore di corrente del bus in continua
 - Utilizzato per motori brushless trapezoidale
 - Basta una bassa risoluzione della misura dell'angolo elettrico
 - *Aspetti critici*: ritardo di commutazione delle valvole
 - Non è possibile avere commutazioni istantanee delle correnti di fase a causa della tensione finita e dell'induttanza di fase → durante gli intervalli di commutazione tutte le fasi possono condurre → pulsazioni di coppia di ampiezza rilevante alle basse frequenze

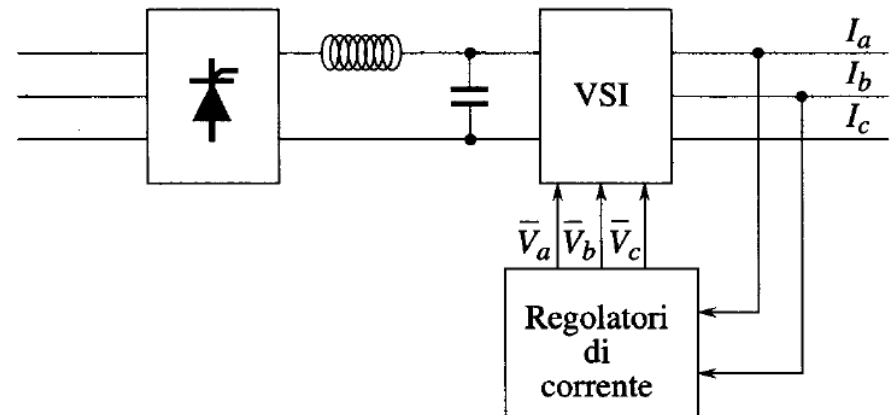
controllo di corrente

Amplificatori di potenza switching – Inverter VSI



– Inverter a tensione impressa (VSI: *Voltage Source Inverter*)

- Regolazione di corrente avviene grazie alla modulazione dei tempi di apertura delle valvole
- Composto da:
 - *Convertitore in ingresso*: genera l'alimentazione in continua
 - *Filtro LC*: stabilizza la tensione
 - *Stadio di conversione DC/AC*: pilotato dai regolatori di corrente





controllo di corrente

Progetto del regolatore di corrente

- **Progetto del regolatore di corrente** nel caso di *motori a spazzole e collettore*
 - Si può considerare la seguente funzione di trasferimento:

$$G_c(s) = \frac{1}{Ls + R}$$

che definisce il legame tra la tensione applicata agli avvolgimenti V e la corrente generata I

- Si può ricavare considerando la *forza contro elettromotrice* come *un ingresso esogeno (disturbo)*: approssimazione giustificata dal fatto che la velocità evolve secondo le dinamiche del sistema meccanico più lenta delle dinamiche della parte elettrica



controllo di corrente

Progetto del regolatore di corrente

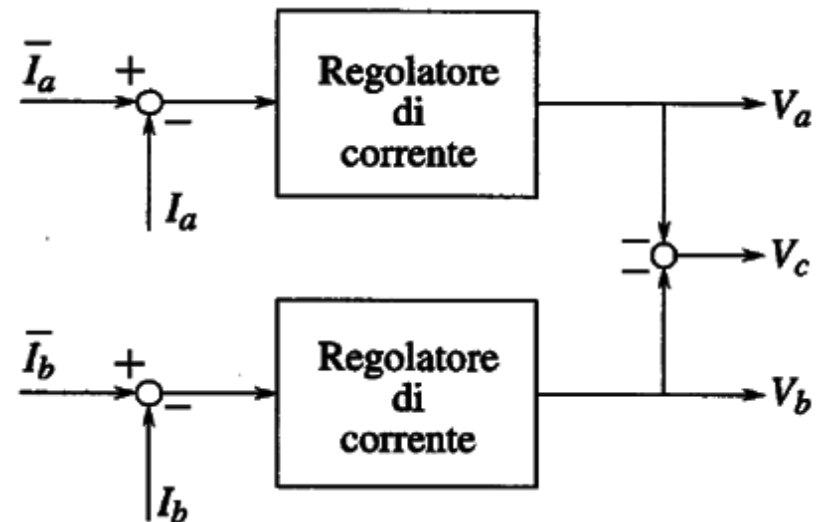
- Si assume che l'effetto della forza controelettrica sia rigettato dall'anello di controllo di corrente (viene considerato come un disturbo rientrante nella banda dell'anello di controllo)
- Sistema del primo ordine
 - Semplice da controllare → sufficiente un regolatore PI, tarato per cancellare l'unico polo del sistema e imponendo una banda di diverse centinaia di Hz



controllo di corrente

Progetto del regolatore di corrente

- **Progetto del regolatore di corrente** nel caso di *motori brushless trifase*
 - Collegamento a stella → correnti di fase non sono indipendenti → è possibile regolare (misurare) solo due correnti → terza corrente determinata dal vincolo a stella
 - Uscite dei due regolatori proporzionali alle tensioni di fase
 - Tensione della terza fase ottenuta come somma delle altre due cambiate di segno





controllo di corrente

Progetto del regolatore di corrente

- Questa tecnica è utilizzata nelle implementazioni del controllo di corrente di tipo *analogico*
- *Controllo digitale* → **Controllo vettoriale**: riformulazione del modello eliminando la dipendenza dall'angolo elettrico mediante una opportuna trasformazione di variabili



controllo di corrente

Controllo vettoriale

- **Controllo vettoriale**

- I motori a corrente continua con spazzole e collettore presentano un perfetto disaccoppiamento magnetico
 - *Asse polare*: individuato dai magneti permanenti
 - *Asse interpolare*: localizza il campo magnetico generato dalla corrente di armatura
- asse polare ed interpolare sempre **ortogonali**, indipendentemente dalla posizione del rotore o dalla tensione alle spazzole
- **Motori brushless** → ortogonalità deve essere imposta modulando opportunamente le correnti negli avvolgimenti
 - non esistono grandezze sullo statore che agiscono solo sulla coppia

controllo di corrente

Controllo vettoriale



- *Motori brushless sinusoidali* → per individuare tali termini si adotta una **trasformazione di coordinate** per rappresentare le grandezze di rotore (tensioni e correnti) rispetto ad un riferimento solidale al rotore stesso
 - Sistema di riferimento solidale al rotore: *asse diretto + asse in quadratura*
- Motori brushless a magneti permanenti
 - *Asse diretto* è l'asse del campo magnetico generato dall'azione dei magneti
 - All'*asse in quadratura* è possibile associare un **circuito fittizio** di alimentazione caratterizzato da una corrente proporzionale alla coppia → circuito fittizio svolge lo stesso ruolo del circuito di armatura nella macchina a corrente continua

controllo di corrente

Controllo vettoriale



– Trasformazione di Park

- Consente di ottenere un modello disaccoppiato del motore brushless

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \mathbf{T}(\alpha) \begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \quad \begin{bmatrix} I_d \\ I_q \end{bmatrix} = \mathbf{T}(\alpha) \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}$$

- Dove la matrice $\mathbf{T}(\alpha)$, funzione dell'angolo elettrico α , è pari a

$$\mathbf{T}(\alpha) = \sqrt{\frac{2}{3}} \begin{bmatrix} \cos \alpha & \cos(\alpha - 2\pi/3) & \cos(\alpha - 4\pi/3) \\ \sin \alpha & \sin(\alpha - 2\pi/3) & \sin(\alpha - 4\pi/3) \end{bmatrix}$$



controllo di corrente

Controllo vettoriale

- Applicando la trasformazione al modello del motore si ottiene:

$$V_d = RI_d + L \frac{dI_d}{dt} + p\omega LI_q$$

$$V_q = RI_q + L \frac{dI_q}{dt} - p\omega LI_d + \sqrt{\frac{3}{2}} pK\omega$$

- **Non c'è più dipendenza dall'angolo elettrico α**
... ma esiste **interazione** tra l'asse diretto e quello in quadratura
- Espressione della coppia nelle nuove variabili:

$$\tau_m = \sqrt{\frac{3}{2}} pKI_q$$

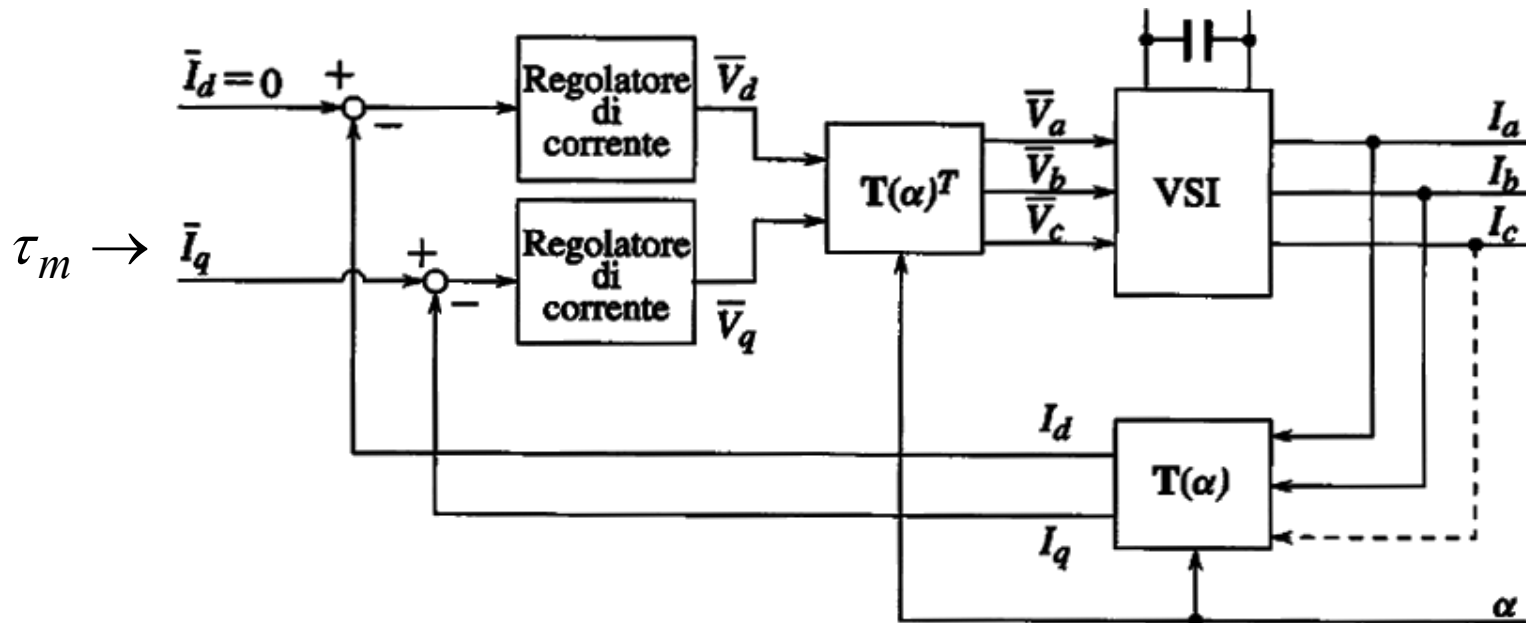
che **dipende dalla sola corrente in quadratura I_q**

controllo di corrente

Controllo vettoriale



- Regolazione $I_q \leftrightarrow$ regolazione della coppia τ_m
- **Corrente di asse diretto deve essere regolata a zero** $I_d \rightarrow$ ottimizza l'efficienza della macchina
 - I_d diversa da zero a parità di coppia determina un aumento della potenza elettrica assorbita a parità di coppia





controllo di corrente

Controllo vettoriale

- Il controllo vettoriale prevede:
 - La misura delle correnti di fase
 - La trasformazione delle grandezze elettriche (correnti) della macchina reale nelle variabili di asse diretto e in quadratura
 - L'applicazione della regolazione alle variabili fittizie
 - La trasformazione delle variabili di controllo definite sugli assi di rotore nelle corrispondenti variabili di statore, realizzata mediante la trasformazione (pseudo)inversa definita dalla matrice $\mathbf{T}(\alpha)^T$

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \mathbf{T}(\alpha)^T \begin{bmatrix} V_d \\ V_q \end{bmatrix} \quad \mathbf{T}(\alpha)^T = \sqrt{\frac{2}{3}} \begin{bmatrix} \cos \alpha & \sin \alpha \\ \cos(\alpha - 2\pi/3) & \sin(\alpha - 2\pi/3) \\ \cos(\alpha - 4\pi/3) & \sin(\alpha - 4\pi/3) \end{bmatrix}$$



controllo di corrente

Modulazione delle tensioni

- Modulazione delle *tensioni*
 - Controllo di fase e controllo vettoriale sono attuate mediante le tensioni di fase
 - Tensioni modulate mediante amplificatori switching che commutano tra interdizione e conduzione ad elevata frequenza
 - Il treno di impulsi generato viene filtrato dall'induttanza degli avvolgimenti → può essere approssimato al valore medio



controllo di corrente

Controllo PWM (Pulse Width Modulation)

- **Modulazione di durata di impulsi PWM**

- Modalità di modulazione tra le più diffuse
- Gli impulsi sono iniziati ad una distanza di t_M secondi $\rightarrow$ frequenza $f_M = 1/t_M$ ($\sim 5 \div 30$ kHz)
- Si consideri una velocità di rotazione desiderata ω_d con una coppia di carico $\tau_l \rightarrow$ occorrerà applicare una tensione V_d :

$$V_d = \frac{R}{K_t} (D_m \omega_d + \tau_l) + K_e \omega_d$$

ovvero occorrerà comandare un **tempo di accensione** t_{on}

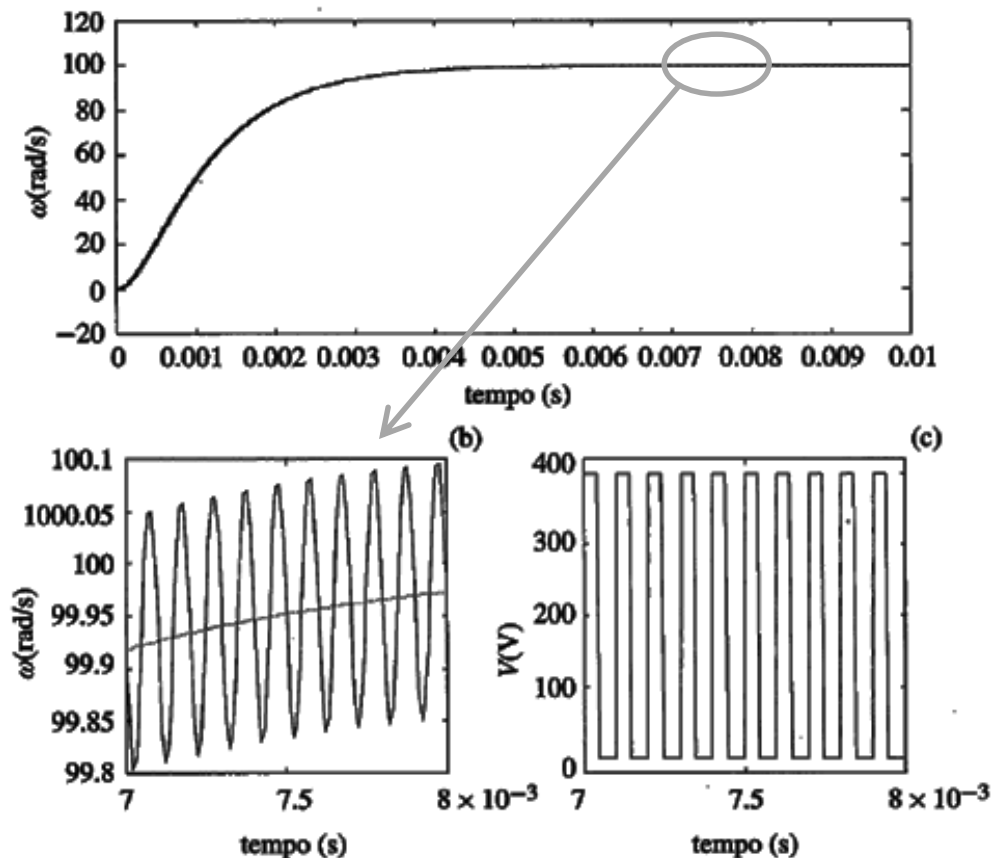
$$t_{on} = \frac{V_d}{V_{CC}} t_M$$

controllo di corrente

Controllo PWM (Pulse Width Modulation)

- *Esempio:* $\omega_d = 100 \text{ rad/s}$, $\tau_l = 1 \text{ Nm}$, $V_{CC} = 400 \text{ V}$, $f_M = 10 \text{ kHz}$, con parametri elettrici tipici di un motore, si ottengono i risultati in figura:

- Sono riportati gli andamenti temporali per il controllo lineare e PWM
- *Ondulazione di velocità* pari allo 0.1%



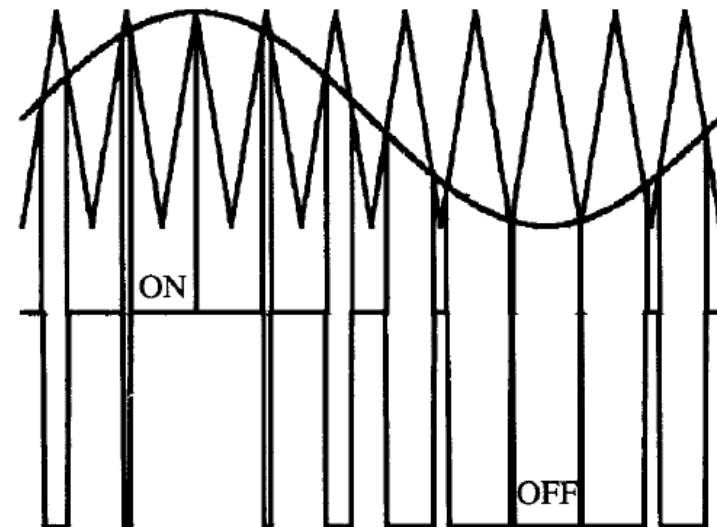
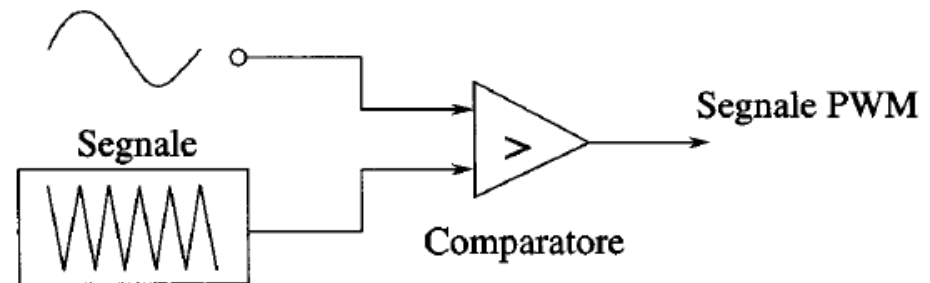
controllo di corrente

Controllo PWM – Metodo della sotto-oscillazione



- **Metodo della sottooscillazione**

- Consente di generare il *segnale PWM*
- Utilizza un oscillatore ed un comparatore
- *Esempio:* generazione di un segnale positivo
 - Il segnale PWM si porta al valore OFF ogni volta che il valore del segnale diventa superiore al segnale derivante dall'oscillatore



attuatori oleodinamici

Generalità



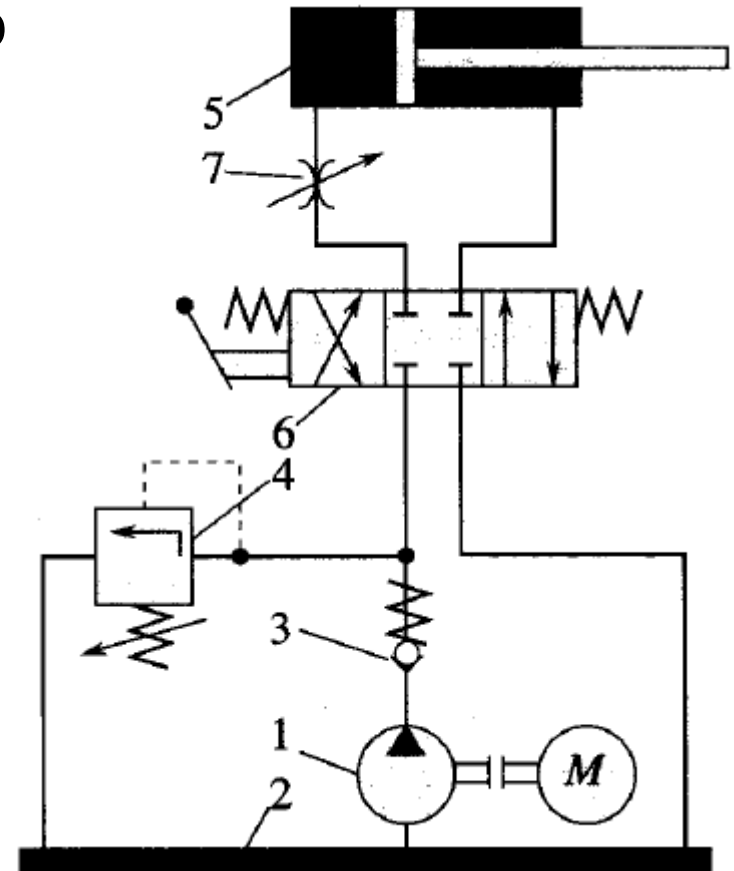
- **Attuatori oleodinamici**

- Generano la forza necessaria per il movimento del carico sfruttando *l'energia di pressione* posseduta da un fluido e fornita al circuito idraulico da una pompa azionata da un motore elettrico
- Pompe a portata variabile e valvole di comando e di regolazione consentono di modulare portate e pressione del fluido → *modulazione del flusso di energia*
- Possono sviluppare grandi forze con piccoli ingombri
- *Nota:* per semplicità di trattazione si considereranno solo attuatori lineari

attuatori oleodinamici

Circuito di base per l'attuazione oleodinamica

- **Cilindro attuttore** con circuito di alimentazione e controllo
 - La **pompa** (1), azionata da un motore elettrico o termico, aspira il liquido dal **serbatoio** (2) e lo convoglia nella tubazione di adduzione del fluido al **cilindro** (5)
 - il liquido fluisce e il pistone si muove fino ad incontrare una forza resistente esterna





attuatori oleodinamici

Circuito di base per l'attuazione oleodinamica

→ la pressione aumenta fino a creare una forza motrice maggiore di quella resistente del carico e quindi lo sposta
Il liquido che esce dal cilindro, dalla camera non connessa alla pompa, si scarica nel serbatoio di raccolta (2)

– Distributore a comando manuale (6)

- Interposto tra pompa e cilindro
- Consente di comandare il senso del movimento del pistone
 - Permette di connettere la pompa a una camera del cilindro e lo scarico all'altra, oppure di incrociare i collegamenti in modo da invertire il senso di spinta

– Valvola di blocco (3)

- Evita movimenti incontrollati del cilindro in caso di blocco della pompa → impedisce lo scarico del fluido attraverso la pompa



attuatori oleodinamici

Circuito di base per l'attuazione oleodinamica

- *Forza massima* sviluppabile dipende dalla pressione del fluido e dalla sezione del cilindro
- **Valvola limitatrice di pressione (4)**
 - Protegge l'impianto da sovrappressioni
 - *Funzionamento automatico*: una molla opportunamente tarata tiene premuto un elemento di tenuta (sfera o cono) sulla sede della valvola, mentre la pressione del circuito agisce sulla superficie dell'elemento di tenuta opposto alla valvola
Se la forza agente (pressione x superficie) diventa maggiore della forza esercitata dalla molla → l'elemento si solleva dalla sede → il fluido sfoga nel serbatoio attraverso la valvola stessa → la pressione si riduce fino a quando non ritorna sotto il valore che ha fatto aprire la valvola → la valvola si richiude



attuatori oleodinamici

Circuito di base per l'attuazione oleodinamica

– Valvola di controllo di portata (7)

- Consente di regolare la portata del fluido e quindi la velocità del movimento
- Consente di variare in modo continuo la sezione di passaggio del fluido e quindi la portata inviata al fluido
- *Funzionamento*: se la valvola riduce la sezione → si riduce la portata del fluido verso il pistone → si riduce la velocità di movimento del pistone

• Fluidi nei circuiti di attuazione oleodinamici

- Dipendono dalle esigenze specifiche dell'applicazione
- *Esempi*: oli minerali, fluidi sintetici ed ecologici, acqua e fluidi speciali (applicazioni aeronautiche)

attuatori oleodinamici

Cilindri



- **Cilindro oleodinamico**

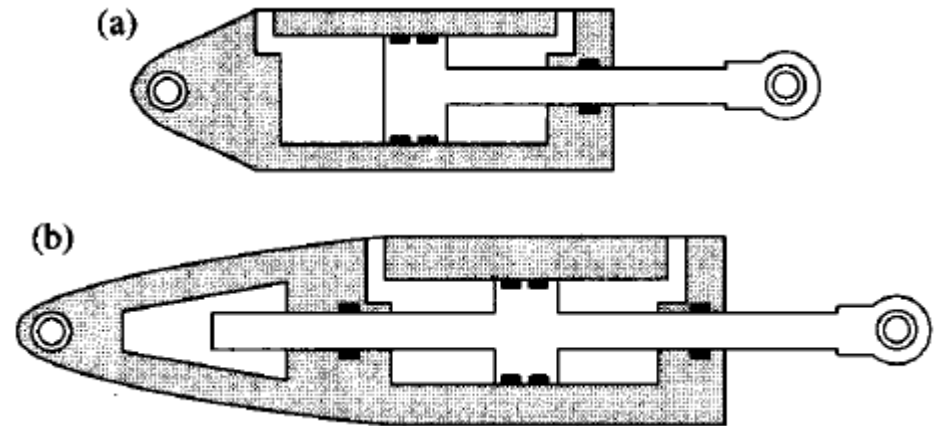
- Costituiti da un pistone libero di scorrere all'interno di una camicia cilindrica

- *Esempi: cilindri a doppio effetto*

- Entrambe le superfici del pistone sono utili per lo sviluppo di forze

- Entrambe le camere sono dotate di attacco di alimentazione

- **Cilindro a doppio stelo** b): secondo stelo funge da guida nel movimento di scorrimento → *cilindro sincrono*: le due aree utili anulari del pistone sono uguali → stessa forza nei due versi





attuatori oleodinamici

Cilindri

- Nel cilindro viene immesso fluido a pressione → si sviluppa una forza pari a $F = \Delta p A$, essendo:
 - Δp la differenza tra le pressioni del fluido sui due lati del pistone
 - A l'area del pistone, supposta uguale sui due lati
- Cilindro a doppio effetto offre
 - Forza costante lungo la corsa
 - Velocità costante lungo la corsa (dipendente dalla portata e dalla sezione del pistone)
 - Possibilità di generare forza di *trazione* e di *spinta* con lo stesso attuatore invertendo le alimentazioni

attuatori oleodinamici

Distributori



- **Distributore** (valvola di direzione)
 - Valvola (aperto-chiuso) che comanda l'avvio, l'arresto e la variazione del senso di flusso in un impianto oleodinamico
 - **Distributore a cursore**
 - Costituito da un corpo munito di cavità assiale di forma complessa, in cui è inserito un cursore con gole scorrevoli e pilotabile dall'esterno
 - Nel corpo sono ricavati canali anulari concentrici ed eccentrici rispetto al foro: le intersezioni tra foro cilindrico e canali anulari generano gli spigoli pilotanti del corpo che interagiscono con gli spigoli pilotati del cursore
 - Lo spostamento del cursore crea o interrompe i collegamenti tra i canali anulari del corpo, realizzando le posizioni di intersezione

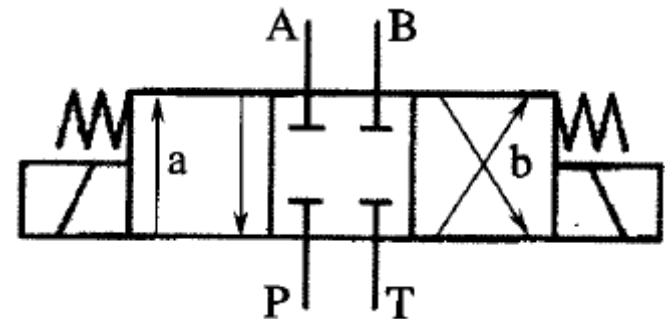
attuatori oleodinamici

Distributori



- *Esempio distributore 4/3:*
distributore con 4 attacchi
utili e 3 posizioni d'intersezione

- P è l'attacco di pressione, alla mandata della pompa
- T è l'attacco di ritorno al serbatoio
- A e B sono gli attacchi di utenza
- La posizione centrale è quella di riposo: assunta spontaneamente dagli elementi in assenza di azionamento esterno per effetto di *molle di centraggio*
 - *Posizione a:* attacco A → mandata P, attacco B → scarico T
 - *Posizione b:* attacco B → mandata P, attacco A → scarico T
 - *Posizione 0:* A e B isolati, alimentazione e carico scollegati



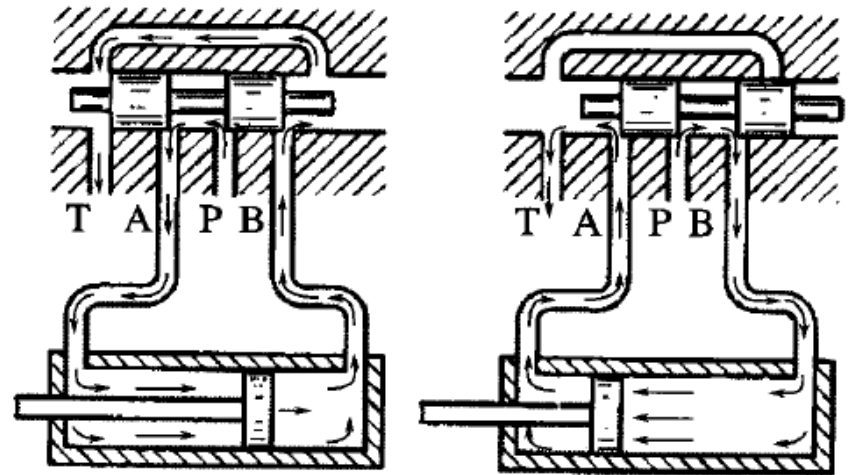
attuatori oleodinamici

Distributori



- *Esempio*: distributore 4/3 per pilotaggio di un cilindro a doppio effetto

- Nello schematizzazione a sinistra si determina una spinta sul pistone verso destra
- Nello schematizzazione a destra si determina una spinta sul pistone verso sinistra



- **Tacche di arresto**

- Assicurano l'esatto posizionamento longitudinale del cursore

- **Molle di ritorno**

- Portano il cursore nella posizione di riposo in assenza di comando

attuatori oleodinamici

Distributori



– Dispositivi di comando

- Azionamento manuali
- A pedale
- Meccanici
- Oleodinamici
- Pneumatici
- Elettromagnetici
 - Spinta generata da elettromagneti (solenoidi)
- Elettropilotati
 - Spinta generata da un dispositivo oleodinamico attivato da un impulso di comando generato da un elettromagnete
- *Nota:* nella schematizzazione del distributore 4/3 riportata in precedenza sono presenti due solenoidi (azionamento elettromagnetico diretto) e due molle di centraggio



attuatori oleodinamici

Valvole proporzionali e servovalvole

- Valvole proporzionali e servovalvole
 - Possiedono le funzioni proprie di commutazione dei distributori
 - Modulano con continuità la portata
- **Valvola proporzionale**
 - Simile ad un distributore, nella parte idraulica, ma dotato di un azionamento in grado di modulare con continuità la posizione del cursore e quindi la portata
 - Azionamento di tipo elettromagnetico diretto con motore coppia per le servovalvole e solenoide per le valvole proporzionali, oppure con uno stadio di preamplificazione oleodinamico



attuatori oleodinamici

Valvole proporzionali e servovalvole

- Servovalvole e valvole proporzionali sono l'anello funzionale di congiunzione tra circuito idraulico e circuito di controllo elettronico
 - Conferiscono agli impianti oleodinamici una facile programmazione dei cicli di lavorazione ed una grande flessibilità
 - La combinazione tra elettronica ed idraulica consente a segnali di piccola potenza il pilotaggio di rilevanti potenze idromeccaniche in pochi millisecondi
- **Servovalvola**
 - Distributore 4/3 con centro a flusso bloccato, provvisto di un azionamento elettromagnetico diretto
 - Regolazione con continuità della posizione del cursore, quindi della portata uscente

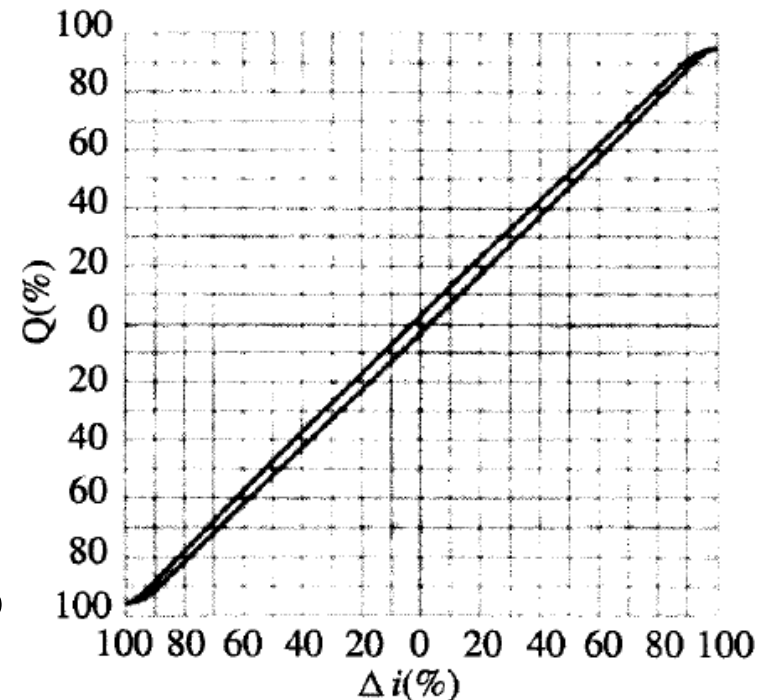
attuatori oleodinamici

Valvole proporzionali e servovalvole



– Esempio: caratteristica statica di flusso della servovalvola

- *Figura*: legame tra la corrente di comando e la portata idraulica transitante a caduta di pressione costante
- **Non linearità**: dovuta alla complessità della forma della cavità e delle sezioni di passaggio
- **Isteresi**: dovuta al comportamento elettromagnetico del solenoide e fenomeni di attrito
- **Banda morta**: dovuta alla necessità di assicurare la tenuta allargando la zona di riposo per il cursore (non presente in figura)

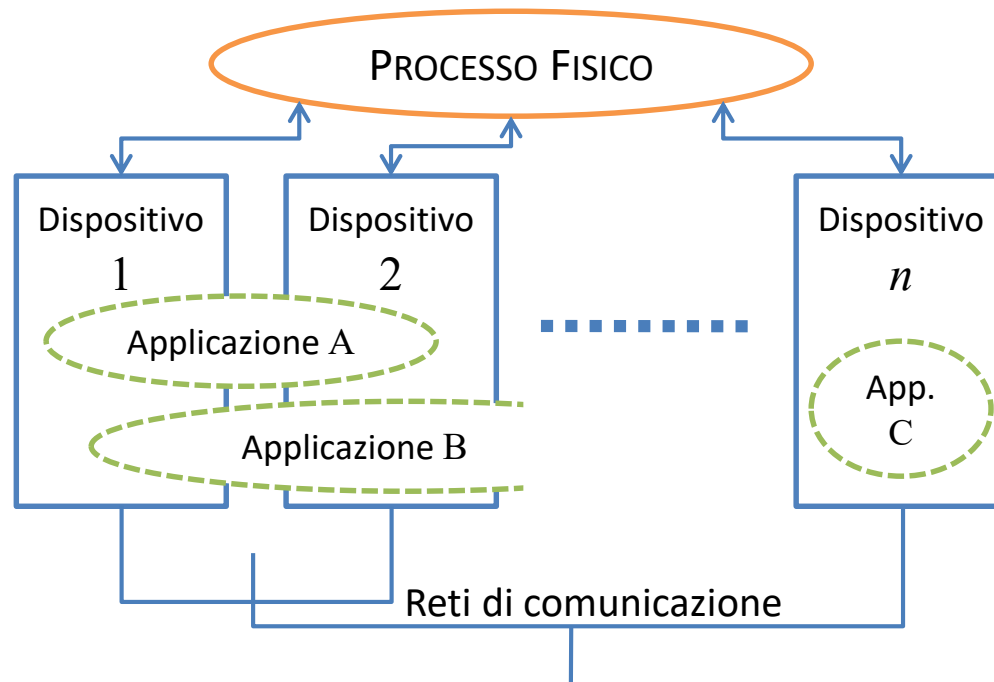


dispositivi per il controllo

Introduzione



- **Sistema di controllo di un processo industriale**
 - Una *definizione* è presente in una proposta di standard internazionale per l'automazione distribuita:



dispositivi per il controllo

Introduzione



- Insieme di **dispositivi** interconnessi e comunicanti tra loro attraverso una o più **reti di comunicazione**
 - Le reti di comunicazione possono essere organizzate secondo relazioni *gerarchiche*
- Le funzionalità espletate dal sistema sono modellate come **applicazioni**
 - Una applicazione può risiedere in un singolo dispositivo o essere distribuita tra diversi dispositivi
 - *Esempio*: chiusura di un anello di controllo, dove l'acquisizione delle misure è realizzata da un dispositivo, l'algoritmo di controllo da un altro e l'invio dei comandi per il controllo da un altro ancora

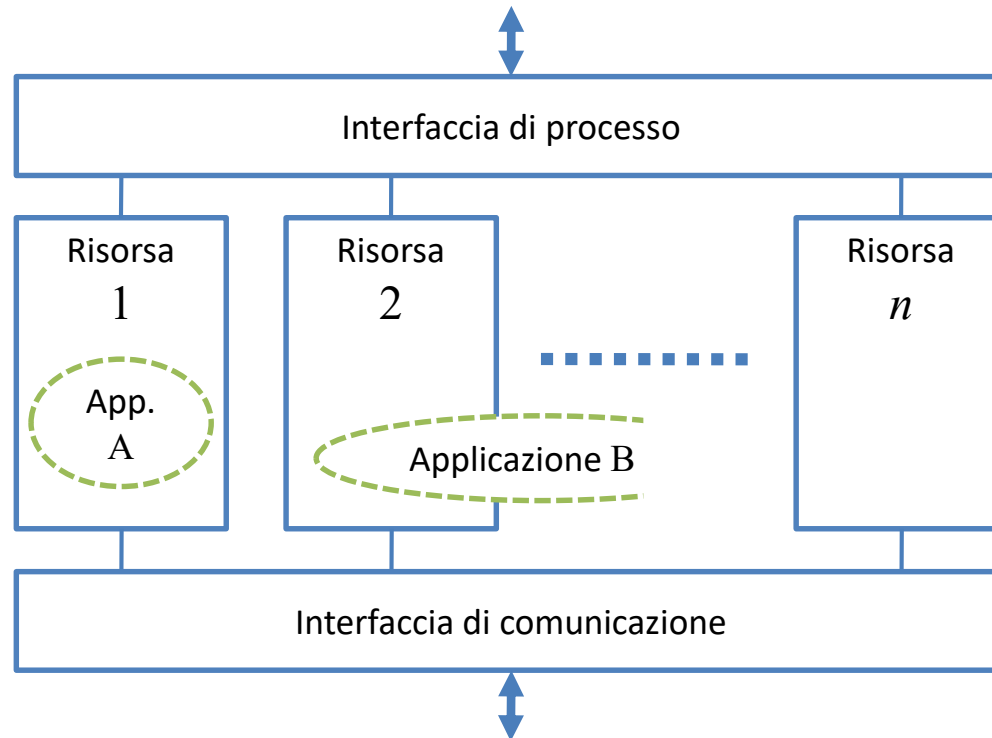
dispositivi per il controllo

Introduzione



- **Dispositivo**

- *Definizione*: entità fisica indipendente capace di realizzare una o più applicazioni, o parti di applicazioni



dispositivi per il controllo

Introduzione



- Un dispositivo è *limitato* dalle sue **interfacce**, ossia dai componenti hardware e software che gli permettono di comunicare con l'esterno
- Un dispositivo deve contenere almeno:
 - Una **risorsa**
 - Una **interfaccia** verso il processo (*interfaccia di processo*) e/o verso la rete di comunicazione (*interfaccia di comunicazione*)

dispositivi per il controllo

Introduzione



- **Risorsa**

- *Definizione*: suddivisione logica della struttura software (eventualmente hardware) di un dispositivo che abbia un controllo indipendente delle sue operazioni
- Una risorsa può essere creata, configurata, parametrizzata, fatta partire e cancellata senza condizionare altre risorse
 - Rientrano in questa definizione le **elaborazioni multitasking** su sistemi con uno o più processori
- *Funzioni* svolte da una risorsa:
 - Accettare dati e/o eventi dal processo e/o dalla rete di comunicazione
 - Processare i dati ricevuti



dispositivi per il controllo

Introduzione

- Restituire dati e/o eventi al processo e/o alla rete di comunicazione

Nota: le funzioni svolte da una risorsa devono essere eseguite secondo quanto specificato dall'applicazione che la sta utilizzando

– In una risorsa devono essere presenti:

- Una o più applicazioni locali, o parti di applicazioni distribuite, che processano dati ed eventi interni
- Funzioni che collegano i dati e gli eventi da e verso il processo e/o la rete di comunicazione con quelli interni
- Una *funzione di pianificazione* delle attività (ad esempio ciclica) per il coordinamento tra queste funzionalità

dispositivi per il controllo

Introduzione



- **Dati**

- I dati sono delle rappresentazioni di *fatti* o *concetti* espresse in una maniera formalizzata adatta alla comunicazione, all'interpretazione o al trattamento da parte delle risorse
 - *Esempio*: codice binario corrispondente ad una misura

- **Eventi**

- Gli eventi rappresentano l'occorrenza di particolari condizioni
 - *Esempi*: raggiungimento di una determinata temperatura, la segnalazione di una fotocellula

dispositivi per il controllo

Introduzione



- **Applicazione**

- Specifica le operazioni che devono essere svolte sui dati come conseguenza degli eventi
- Può essere distribuita tra molte risorse nello stesso o in differenti dispositivi
- Specifica le relazioni di causali con cui una risorsa determina le risposte a eventi interni, di processo o di comunicazione
- *Esempi*: esecuzione di operazioni, pianificazione di operazione, modifica di variabili, generazione di eventi addizionali, interazioni con le interfacce di processo e comunicazione

dispositivi per il controllo

Introduzione



- **Interfaccia di processo**

- Mette in relazione le risorse contenute nel dispositivo fisico con il processo fisico, comunicando con i sensori e gli attuatori
- È costituita da un insieme di dispositivi hardware e dal software per la loro gestione (driver)
 - *Esempi:* schede di ingresso/uscita analogiche, schede di ingresso/uscita digitali
- Le informazioni scambiate con il processo fisico sono presentate alle risorse come dati del processo e/o come eventi associati al processo

dispositivi per il controllo

Introduzione



- **Interfaccia di comunicazione**

- Mette in relazione le risorse con quelle appartenenti ad altri dispositivi per lo scambio di informazioni attraverso una rete di comunicazione
- È costituita da un insieme di dispositivi hardware e dal software per la loro gestione (driver)
 - *Esempi:* scheda di rete, scheda modem
- Consente di *presentare informazioni* (dati ed eventi) alla risorsa e *fornire servizi* aggiuntivi (supporto alla programmazione, configurazione del sistema, diagnostica)

dispositivi per il controllo

Introduzione



- **Tipologie di sistemi di controllo e misura**
 - Collegati direttamente a macchine da controllare
 - *Requisiti*: sistemi operativi real-time e multitasking, interfacce di processo molto sviluppate, costruzione robusta
 - *Esempio*: PLC (*Programmable Logic Controller*)
 - Interfaccia uomo macchina
 - *Requisiti*: buone capacità grafiche, capacità sviluppate per la comunicazione con altri dispositivi, non necessitano di una interfaccia di processo: (*esempio*: personal computer)
 - Controllo di macchine complesse
 - *Esempio*: robot manipolatori industriali
 - Gestione di basi di dati

dispositivi per il controllo

Requisiti di un dispositivo di controllo



- **Requisiti fondamentali di un dispositivo di controllo**
 1. Capacità di rispondere a stimoli provenienti dal loro esterno sotto forma di eventi o dati che arrivano da sensori
 2. Capacità di agire al loro esterno modificando il comportamento del processo fisico che controllano
- *Definizione:* un dispositivo di controllo è un sistema per l'elaborazione delle informazioni destinato al controllo diretto di processi fisici



dispositivi per il controllo

Requisiti di un dispositivo di controllo

- **Funzionalità di un dispositivo di controllo**
 - Controllo a ciclo chiuso di variabili fisiche
 - Regolazione ed asservimento
 - Calcolo dei valori di riferimento per tali variabili
 - Controllo logico/sequenziale
 - Gestione di allarmi e anomalie
 - Interfaccia operatore
 - Comunicazione con altri dispositivi
 - Trattamento di segnali di ingresso/uscita di differente natura (anche in numero elevato)



dispositivi per il controllo

Requisiti di un dispositivo di controllo

- **Modalità di esecuzione dei compiti**
 - *Periodicamente*, a intervalli di tempo assegnati
 - *Ciclicamente*, appena terminata l'esecuzione si deve ricominciare da capo
 - *Una sola volta*, all'occorrenza di eventi particolari
- Non tutte le funzionalità sono sempre presenti
 - Le funzionalità realmente implementate dipendono dalla complessità del dispositivo, ovvero dall'applicazione a cui è destinato
 - *Esempio*: Forno a microonde vs. Controllore robot manipolatore

dispositivi per il controllo

Requisiti di un dispositivo di controllo



- **Tempo reale**

- Interazione con il mondo fisico deve avvenire in tempo reale
- *Definizione*: un **sistema di elaborazione a tempo reale** deve rispondere in un modo certo ed entro tempi fissati a eventi esterni non prevedibili
- Ritardi nell'esecuzione di un processo sono considerati *malfunzionamenti* nei sistemi a tempo reale
 - *Esempi*: tutti i processi di controllo diretti

dispositivi per il controllo

Requisiti di un dispositivo di controllo



- Il requisito di tempo reale, da un punto di vista hardware, richiede:
 - Utilizzo di uno o più processori dotati di adeguata velocità di elaborazione → durata dell'esecuzione dei compiti entro i limiti
 - Tempi di esecuzione delle istruzioni noto (almeno valori massimi)
 - Accesso alla memoria e ai dispositivi di ingresso/uscita veloce, affidabile e deterministico
 - Garanzia di una tempificazione di riferimento certa
 - Presenza di funzioni di autodiagnosi
 - Presenza di ridondanze strutturali → possibilità di poter continuare a operare anche in presenza di malfunzionamenti

dispositivi per il controllo

Requisiti di un dispositivo di controllo



- Applicazioni di controllo possono essere:
 - *semplici* → assenza di un sistema operativo, tutto è affidato al programma scritto dall'utente
 - *complesse* → presenza di sistema operativo (se pur semplice) che si occupa della pianificazione dell'esecuzione dei processi (*scheduling*) e della gestione della comunicazione tra i processi
- **Sistema operativo a tempo reale**
 - Presenza di un meccanismo di assegnazione delle *priorità* → processi con una assegnata priorità non possono essere interrotti da processi a priorità inferiore
 - *Multitasking pre-emptive* → possibilità di interrompere qualsiasi processo per passare le risorse ad un processo che ne ha maggiormente bisogno



dispositivi per il controllo

Requisiti di un dispositivo di controllo

- Assenza di situazioni di *deadlock* tra processi → presenza di un meccanismo di acquisizione della priorità
 - Nel caso in cui un processo ad alta priorità è bloccato perché in attesa di una risorsa acquisita da un processo a bassa priorità, mentre ancora un altro processo è in esecuzione, il processo a bassa priorità deve acquisire la priorità del processo a priorità più alta che esso stesso blocca
- Meccanismo di sincronizzazione e comunicazione tra processi prevedibile → scambio di dati in tempi certi
- Tempi noti (valori massimi) necessari al sistema per sospendere un processo, lanciarne un altro e realizzare una chiamata di sistema
- Gestione degli eventi di malfunzionamento in maniera non brusca
 - *Esempio*: evitare che una divisione per zero interrompa necessariamente l'esecuzione di un processo



dispositivi per il controllo

Requisiti di un dispositivo di controllo

- Scalabilità → possibilità di scegliere le funzionalità del sistema operativo necessarie all'applicazione per non caricare inutilmente il dispositivo di controllo
- Gestione dei segnali di ingresso/uscita
 - Segnali di tipo analogico e digitale
 - Dispositivo dotato di apposite interfacce hardware e dei driver per pilotarle
 - Stesse considerazioni per le interfacce di comunicazione
- **Requisiti costruttivi**
 - Dispositivo in grado di funzionare in ambienti ostili (sporco, vibrazioni, interferenze elettromagnetiche)
 - Solidità e robustezza costruttive
 - Schermatura da campi elettromagnetici

controllori per applicazioni generiche

Introduzione



- **Architettura fisica di un dispositivo di controllo**
 - Molto varia, da un *singolo chip*, contenente tutto il necessario a realizzare un controllo, fino a un *sistema a bus*, dove risiedono più moduli specializzati
- **Architetture di un dispositivo di controllo per applicazioni generiche:**
 - *Monolitiche* → singolo chip o singolo circuito integrato
 - *Basate su bus* → combinazioni di moduli con differenti funzionalità interconnessi tramite un bus comune
 - *Basate su personal computer*

controllori per applicazioni generiche

Controllori monolitici



- **Controllori monolitici**

- Inglobano tutto quello che serve per realizzare le funzioni di controllo in un'unica scheda o in un unico circuito integrato
- Con riferimento ad una applicazione generica, un controllore monolitico deve possedere:
 - Un'unità di elaborazione che esegue i programmi utente e quelli di sistema operativo
 - Una memoria non volatile che contiene i programmi
 - I controllori monolitici non hanno memoria di massa
 - Una memoria (volatile) per la conservazione delle variabili
 - Un clock → segnali di temporizzazione necessari al controllo



controllori per applicazioni generiche

Controllori monolitici

- Dei dispositivi di interfaccia per l'acquisizione e la generazione di segnali analogici
 - Campionatori, convertitori D/A e A/D, ecc.
- Dei dispositivi di interfaccia per l'acquisizione e la generazione di segnali digitali
- Un sistema di gestione delle interruzioni

– **Microcontrollori**

- *System-on-chip*: tutti i componenti integrati un singolo chip
- Soluzioni circuitali specifiche → **istruzioni dedicate** alla manipolazione di bit, alla gestione degli ingressi e delle uscite, alla gestione veloce ed efficiente degli interrupt
- Presenti in modo diffuso in un gran numero di dispositivi
 - *Esempi*: elettrodomestici, calcolatori e periferiche, automobili, ecc.



controllori per applicazioni generiche

Controllori monolitici

- Adatti per applicazioni di controllo in cui la potenza di elaborazione necessaria non è elevata e vi è un numero ridotto di ingressi e uscite da gestire
- Disponibili in diverse taglie e varietà
 - Microcontrollori a 8 bit tra i più diffusi
 - Esistono ancora controllori a 4 bit
 - In aumento il mercato dei controllori a 16 e 32 bit
- Caratterizzati inoltre in base a
 - Tecnologia di integrazione
 - Tipo di architettura
 - Tipo e dimensione della memoria
 - Consumo
 - Numero e tipo di gestione degli interrupt
 - Numero e tipo di ingressi e uscite

controllori per applicazioni generiche

Controllori monolitici



– Sistema operativo

- Normalmente non previsto
- Utente si deve far carico, con la scrittura del programma, della gestione delle risorse
- Nei casi di necessità, si ricorre a sistemi operativi scalabili, dai quali si può utilizzare solo il pianificatore dell'esecuzione dei processi (*scheduler*) e il gestore della comunicazione inter-processo

– Sistema di sviluppo

- Basato su personal computer
- Prevista la programmazione in linguaggi assemblativi o compilatori per linguaggi ad alto livello come il C (*cross-compiler*)
- Disponibilità di *emulatori* per eseguire i programmi, anche passo per passo, collegabili direttamente al processo con una sonda

controllori per applicazioni generiche

Controllori monolitici



– Controllori single-board

- Costituiti da un'unica scheda elettronica dove trovano posto tutti i componenti prima descritti
- Maggiori prestazioni in termini di potenza di calcolo e numero di ingressi e uscite gestibili

– Unità di processo

- Processori CISC (*Complex Instruction Set Computer*)
- Processori RISC (*Reduced Instruction Set Computer*)
- DSP (*Digital Signal Processor*)

– Assente o limitata capacità di colloquiare con altri dispositivi, ridotte e semplici interfacce operatore

– Dimensioni ridotte (microcontrollori) → **sistemi embedded**



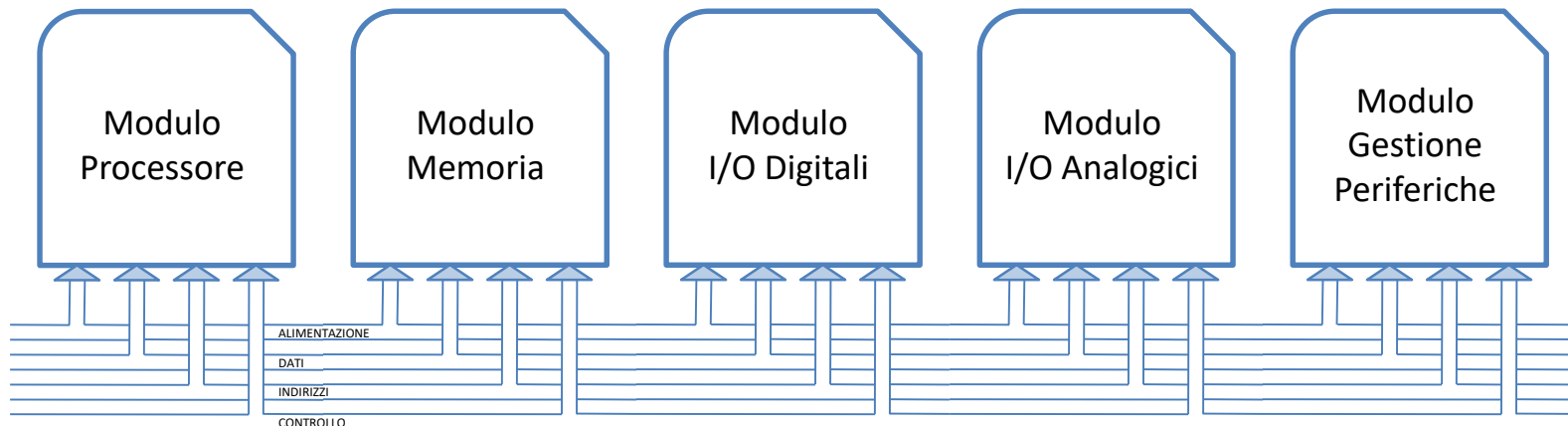
controllori per applicazioni generiche

Controllori con architettura a bus

- **Controllori con architettura a bus**

- Sono caratterizzati da:

- Elevate capacità di elaborazione
 - Possibilità di trattare un numero elevato di ingressi e uscite
 - Possibilità di comunicare attraverso reti informatiche
 - Possibilità di realizzare interfacce operatore molto sofisticate





controllori per applicazioni generiche

Controllori con architettura a bus

- Un bus è definito come la combinazione di tre concetti:
 - Un insieme di **linee elettriche**, raggruppate per funzioni che connettono tra loro varie schede o moduli
 - Un **protocollo** attraverso il quale i moduli possono usare le linee elettriche per comunicare tra loro
 - Un insieme di **caratteristiche elettriche e meccaniche** dei connettori che servono per collegare tra loro i moduli
- Schede collegate in parallelo → velocità di propagazione dei segnali elettrici, confrontata alla frequenza con cui variano, limita la lunghezza del bus a qualche decimetro



controllori per applicazioni generiche

Controllori con architettura a bus

- Collegamenti tipici realizzati da un bus sono:
 - Le *linee di indirizzo* → ogni scheda decodificherà il proprio indirizzo
 - Le *linee dati* (in alcuni casi in multiplex con quelle di indirizzo)
 - Le *linee di controllo* → handshake, clock, interrupt, arbitraggio bus
 - Le *linee di alimentazione*
 - Le *linee a disposizione dell'utente*
- **DMA** (*Direct Memory Access*)
 - Accesso alla memoria per il trasferimento di blocchi di dati senza far uso del processore
- **Arbitro del bus** nei sistemi a multiprocessore
 - Sempre la prima scheda, a rotazione o a priorità

controllori per applicazioni generiche

Controllori con architettura a bus



- Tra le *caratteristiche* di un bus si ricordano:
 - I dati elettrici e meccanici
 - Livelli di segnale, disposizione fisica dei conduttori, connettori
 - La velocità di trasmissione dei dati
 - Lo spazio indirizzabile
 - La lunghezza dei dati
 - Il numero delle linee di interrupt e come esse sono gestite
 - Bus sincrono (necessità di un clock) o asincrono (senza clock, più flessibile → occorrono dei segnali di sincronizzazione)
 - Il numero delle unità master del bus che sono consentite
 - Vincoli sul tipo di processore
 - Autoconfigurazione del bus all'inserimento di nuovi moduli



controllori per applicazioni generiche

Controllori con architettura a bus

- **Bus standard** → disponibilità di moduli di diversi produttori che possono essere connessi insieme per realizzare l'architettura desiderata
- Bus VME
 - Definizione fissata dalla norma IEEE 1014
 - Esistono varie implementazioni
 - Molto diffuso in ambito industriale
 - Ampia gamma di schede compatibili
 - Caratteristiche:
 - Linee di indirizzo da 16, 24, 32, 40 o 64 bit, nelle varie implementazioni
 - Linee dati da 8, 16, 24, 32 o 64 bit



controllori per applicazioni generiche

Controllori con architettura a bus

- Velocità massima di trasferimento fino a 500 MB/s
- 7 linee di interrupt
- 4 livelli di priorità per l'allocazione del bus
- Possibilità di trasferire blocchi da 256 word
- 64 linee definibili dall'utente
- Modo operativo asincrono, senza clock
- Può avere fino a 21 master del bus



controllori per applicazioni generiche

Controllori con architettura a bus

- Bus EISA (*Extended Industry Standard Architecture*)
 - Utilizzata in passato anche nei personal computer
 - Caratteristiche:
 - Linee indirizzo da 32 bit
 - Linee dati da 8, 16, 32 bit
 - Velocità massima di trasferimento da 33 MB/s
 - 6 linee di interrupt
 - Supporto del multi-master
 - Modo operativo sincrono, con clock



controllori per applicazioni generiche

Controllori con architettura a bus

– Bus PCI

- Utilizzata anche nei personal computer
- Caratteristiche:
 - Linee dati e di indirizzo da 32 bit
 - Velocità massima di trasferimento da 132 MB/s
 - Supporto del multi-master
 - Modo operativo sincrono, con clock

– Bus PC/104 e PC/104+

- Prevedono moduli di ridotte dimensioni con connettori passanti
 - Assenza di rack, moduli impilati → sistemi molto compatti (*sandwich*)
- Stesso bus dei PC (EISA per PC/104, PCI per PC/104+), ma differenti connettori e minore potenza per il pilotaggio delle linee
- Ridotte dimensioni e assorbimento di corrente → sistemi embedded



controllori per applicazioni generiche

Controllori con architettura a bus

– Architettura modulare

- Possibilità di selezionare l'hardware migliore per l'applicazione da realizzare
- **Moduli** tipici:
 - Modulo processore (CISC, RISC, DSP)
 - Moduli di ingresso e uscita analogici
 - Moduli di ingresso e uscita digitali
 - Moduli contatore ad alta velocità
 - Moduli di memoria di massa a stato solido
 - Moduli per comunicazione su rete informatica
 - Moduli per il controllo di periferiche (schermi, tastiere, ecc.)
- **Rack**: contengono i moduli meccanicamente e li uniscono elettricamente con un circuito stampato dotato di connettori

controllori per applicazioni generiche

Controllori con architettura a bus



– Sistema operativo

- Sistema più complesso → occorre un sistema operativo per gestire i processi elaborativi e le risorse del sistema

– Sistema di sviluppo

- È previsto l'uso di linguaggi di programmazione ad alto livello, come il C
- Esistono sistemi di sviluppo che implementano lo standard 61131-3 (descritto in seguito) → programmazione direttamente in uno dei cinque linguaggi di programmazione specifici per l'automazione, come il Ladder Diagram e l'SFC



controllori per applicazioni generiche

Controllori basati su personal computer

- Controllori basati su personal computer
 - Utilizzo di personal computer “general purpose” come dispositivi di controllo
 - Vantaggi rispetto ai dispositivi di controllo:
 - Costo minore
 - Minori professionalità necessarie per utilizzarli
 - Ampia varietà di fornitori hardware
 - Possibilità di scelta dei fornitori software
 - Manutenzione semplificata
 - Funzionalità di base molto ampie (*esempio*: interfaccia grafica, memoria di massa, interfacce di rete, ecc.)

controllori per applicazioni generiche

Controllori basati su personal computer



- Svantaggi rispetto ai dispositivi di controllo:
 - Interfaccia di processo limitata (poche schede installabili)
 - Non è robusto → non adatto all'impiego in ambienti ostili, come quelli in cui si trovano i processi da controllare
- Svantaggi superabili con l'uso di reti informatiche:
 - Il calcolatore può essere collocato lontano dal processo
 - È possibile collegare un elevato numero di schede di I/O remoto
→ penalizzazione dei tempi di rilevamento degli ingressi e di aggiornamento delle uscite



controllori per applicazioni generiche

Controllori basati su personal computer

- Sistema operativo a tempo reale
 - Necessari, vista la complessità dell'hardware
 - Esistono sistemi nativi a tempo reale, o apposite estensioni, basati su UNIX e su Windows
- Sistema di sviluppo
 - Programmazione con linguaggi ad alto livello
 - Esistono emulatori di controllori specializzati su piattaforma PC (*esempio*: Soft-PLC, Soft-CNC), programmabili secondo lo standard 61131-3



controllori specializzati

Controllori di macchine a controllo numerico

- **Controllori specializzati**
 - Sono controllori di ampia diffusione industriale che non richiedono la programmazione dell'algoritmo di controllo, ma semplicemente una sua configurazione
- **Controllori di macchine a controllo numerico**
 - *Macchina a controllo numerico*: macchinario che esegue lavorazioni e movimentazioni, con assi di rotazione e traslazione controllati, dotata di un linguaggio di programmazione ad alto livello orientato alla funzionalità che svolge



controllori specializzati

Controllori di macchine a controllo numerico

– *Esempio*: tornio numerico

- Assi sono quelli di rotazione del pezzo da tornire e quelli di movimentazione dell'utensile
- Linguaggio di programmazione prevede la possibilità di definire la traiettoria che l'utensile deve eseguire, attraverso la combinazione di segmenti di retta, di archi di assegnata curvatura, ecc.
 - Modello CAD 3D → generazione automatica delle traiettorie

– *Esempio*: braccio manipolatore robotico industriale

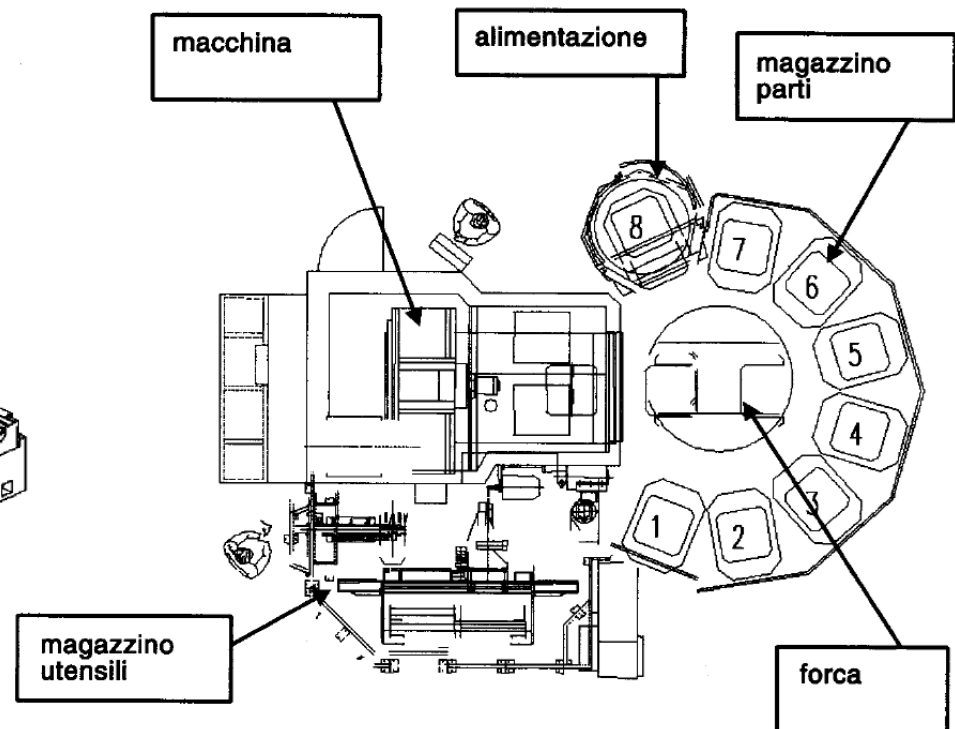
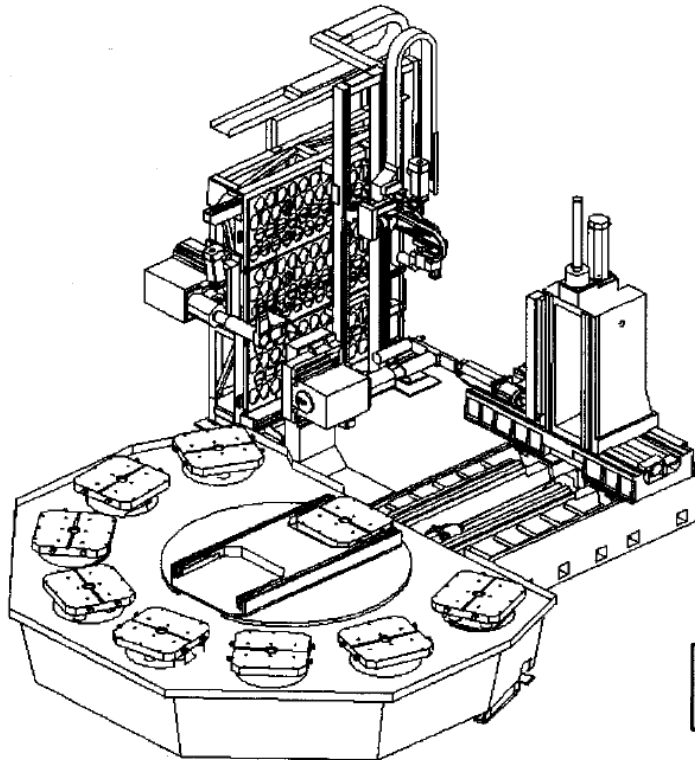
- Formato da più bracci connessi da giunti di rotazione o traslazione
- Il linguaggio di programmazione prevede la possibilità di scrivere istruzioni del tipo “vai dal punto A al punto B lungo una retta”, “realizza questo insieme di movimenti in maniera ciclica”, “chiudi la pinza”, ecc.

controllori specializzati

Controllori di macchine a controllo numerico



- *Esempio: illustrazione di una macchina a controllo numerico*



controllori specializzati

Controllori di macchine a controllo numerico



- Funzionalità realizzate dall'architettura di controllo
 - Controllare il movimento di ogni singolo asse
 - Generare i riferimenti per ogni singolo asse (leggi orarie)
 - Coordinare tra loro i movimenti degli assi
 - Eseguire istruzioni espresse nel linguaggio di programmazione
 - Gestire le logiche e le sequenze della macchina e di altri dispositivi ad essa connessa
 - Realizzare un'interfaccia uomo-macchina per consentire la supervisione del funzionamento e la modifica dei parametri, oltre che programmare e configurare la macchina
 - Comunicare con altri dispositivi per realizzare il coordinamento con altre macchine necessario alla produzione
 - Avere capacità di autodiagnostica



controllori specializzati

Controllori di macchine a controllo numerico

– Architettura a bus con moduli specializzati

- Un modulo per ogni asse (o per più assi) di movimentazione che si occupa del controllo di posizione e velocità dell'asse
- Un modulo dedicato all'interpretazione del programma e alla generazione delle traiettorie di riferimento per i singoli assi
- Un modulo per il controllo logico/sequenziale della macchina e di altri dispositivi ad essa connessi
- Un modulo per realizzare l'interfaccia operatore

– Programmazione

- Mediante linguaggi specifici
- Previsti programmi di *autoapprendimento* → posizionamento manuale + scrittura semiautomatica delle istruzioni da parte del controllore

controllori specializzati

Regolatori PID industriali



- Regolatori PID industriali
 - Regolatore PID → Proporzionale-Integrale-Derivativo
 - Sono i più usati in ambito industriale per mantenere variabili fisiche a valori costanti (*regolazione*)
 - Equazione di funzionamento nel tempo continuo:

$$u(t) = K_P e(t) + K_I \int_0^t e(\tau) d\tau + K_D \frac{de(t)}{dt} + u_B$$

dove u è il segnale di controllo, e è la differenza (*errore*) tra il valore di riferimento e il valore effettivo della variabile da controllare



controllori specializzati

Regolatori PID industriali

- Primo termine → *proporzionale*, definito da guadagno K_P
 - Secondo termine → *integrale*, definito dal guadagno K_I
 - Terzo termine → *derivativo*, definito dal guadagno K_D
 - Ultimo termine → *bias*, termine costante u_B
- Controllori dedicati disponibili in forma monolitica
- Si connettono direttamente ai sensori e attuatori sul processo
- Realizzazione reale
- Differisce da quella ideale rappresentata dall'equazione ideale
 - Modifiche per rendere il dispositivo più sicuro ed efficiente:
 - Controllo sui valori dell'errore e del segnale di riferimento
 - Meccanismi di *anti wind-up*
 - Trasferimento senza sobbalzi da controllo manuale ad automatico
 - ...

controllori specializzati

Regolatori PID industriali



- Algoritmo di controllo pre-programmato
 - Si devono assegnare solo i parametri di configurazione del controllore (K_p , K_I , ...)
 - In alcune realizzazioni è previsto un sistema di autodeterminazione dei parametri dell'algoritmo → *controllori auto-sintonizzanti*
- Comunicazione
 - In alcuni casi è prevista la possibilità di comunicare su rete informatica con altri dispositivi
 - Comunicazione dello stato del controllore
 - Ricezione di un nuovo riferimento
 - ...



controllori specializzati

Controllori per motori elettrici

- Controllori per motori elettrici
 - Controllo di motori è molto diffuso in ambito industriale
 - Attualmente è realizzato quasi esclusivamente con tecnologie digitali
 - Esistono controllori specializzati per differenti tipi di motori che contengono tutto il necessario per controllare e gestire un motore
 - Funzioni principali realizzate
 - Acquisire informazioni sulle variabili di stato elettriche e meccaniche necessarie al controllo attraverso misure dirette o stime
 - Implementare l'algoritmo di controllo del moto e la logica di gestione del motore

controllori specializzati

Controllori per motori elettrici



- Realizzare la conversione di potenza necessaria al corretto pilotaggio del motore
- Ricevere i comandi di moto da eseguire da altri dispositivi

– Funzioni secondarie realizzate

- Visualizzazione e comunicazione ad altri dispositivi dello stato e dei segnali di allarme
- Monitoraggio di variabili importanti
- Gestione di condizioni di allarme
- Esecuzione di procedure di test per la regolazione di parametri
- Autodiagnostica



controllori specializzati

Controllori per motori elettrici

- Si può realizzare un controllo di coppia o di velocità (*azionamenti elettrici*) oppure di posizione (*controllo d'asse*)
- Alcuni controlli non sono realizzati in retroazione
 - *Esempio*: controllo di velocità di un motore a passo
- Algoritmi di controllo pre-programmati
 - Necessaria solo la configurazione del controllore

controllore a logica programmabile

Introduzione



- **Programmable Logic Controller (PLC)**
 - Può essere considerato un dispositivo di controllo per applicazioni generiche, ma è caratterizzato da un alto grado di specializzazione → *controllo logico/sequenziale*
 - È il più diffuso dispositivo di controllo in ambito industriale
 - *Dispositivo modulare con architettura a bus*
 - Possibilità di trattare fino a migliaia di punti ingresso/uscita di varia natura (analogici/digitali, tensione/corrente, ecc.)
 - Dotato di sistemi operativi real-time multi-tasking proprietari
 - Sono dispositivi di costruzione molto robusta adatti ad applicazioni in contesti industriali (vibrazioni, sporco, disturbi elettromagnetici)

controllore a logica programmabile

Cenni storici



- **Cenni storici**

- Da sempre obbiettivo della tecnica è stata l'esecuzione di processi di lavorazione in modo automatico

- Prima della scoperta dell'elettricità → **controllori meccanici**

- Orologio a pendolo

- Orologio ad acqua

- Regolatore di velocità di Watt

- **Dispositivi elettromeccanici** (relè e temporizzatori)

- Facili da utilizzare, ma caratterizzati da molti svantaggi:

- » bassa velocità di elaborazione dei segnali

- » costo elevato

- » non riconfigurabili

- » progettazione, installazione, manutenzione lunghe e complesse

controllore a logica programmabile

Cenni storici



- L'avvento dell'elettronica (transistori, circuiti integrati) risolse molti dei problemi dei sistemi elettromeccanici
 - Mancava ancora una caratteristica essenziale: *flessibilità*
 - Possibilità di essere adattati in tempi brevi a svolgere nuove funzioni
- Avvento del calcolatore elettronico
 - La caratteristica di essere un **sistema programmabile** rivoluzionò anche il settore dell'automazione industriale
 - Nel 1968 nasceva il primo controllore a logica programmabile
 - La General Motors specificò le caratteristiche desiderate per una nuova generazione di controllori da destinare ai propri impianti:
 - » Facilità di programmazione e riprogrammazione, anche direttamente sul luogo di funzionamento
 - » Facilità di manutenzione → dispositivi modulari



controllore a logica programmabile

Cenni storici

- » Robustezza per consentirne il funzionamento in ambienti industriali (interferenze elettromagnetiche, polvere, vibrazioni)
- » Compattezza (dimensioni ridotte rispetto ai sistemi precedenti)
- » Costi ridotti
- Inoltre si richiedevano anche delle specifiche secondarie, tra cui:
 - » Possibilità di espandere la memoria
 - » Comunicazione con dispositivi di registrazione dei dati
 - » Possibilità di accettare segnali di media tensione alternata
- Il risultato fu lo sviluppo della prima generazione di PLC
 - *Sequenziatori riprogrammabili*

controllore a logica programmabile

Cenni storici



- La Allen Bradley introdusse il primo PLC basato su microprocessore (si trattava dell'8080) alla metà degli anni settanta
 - Da allora l'evoluzione dei PLC è continuata parallelamente a quella dell'informatica classica
- Gli attuali PLC di alta classe:
 - sono basati su sistemi multiprocessore
 - hanno la possibilità di connessione a reti informatiche
 - hanno la capacità di eseguire funzioni molto complesse
- I moderni PLC possiedono le caratteristiche dei calcolatori convenzionali ma sono adatti al loro utilizzo principale, il controllo di processi industriali

controllore a logica programmabile

PLC e sistema PLC



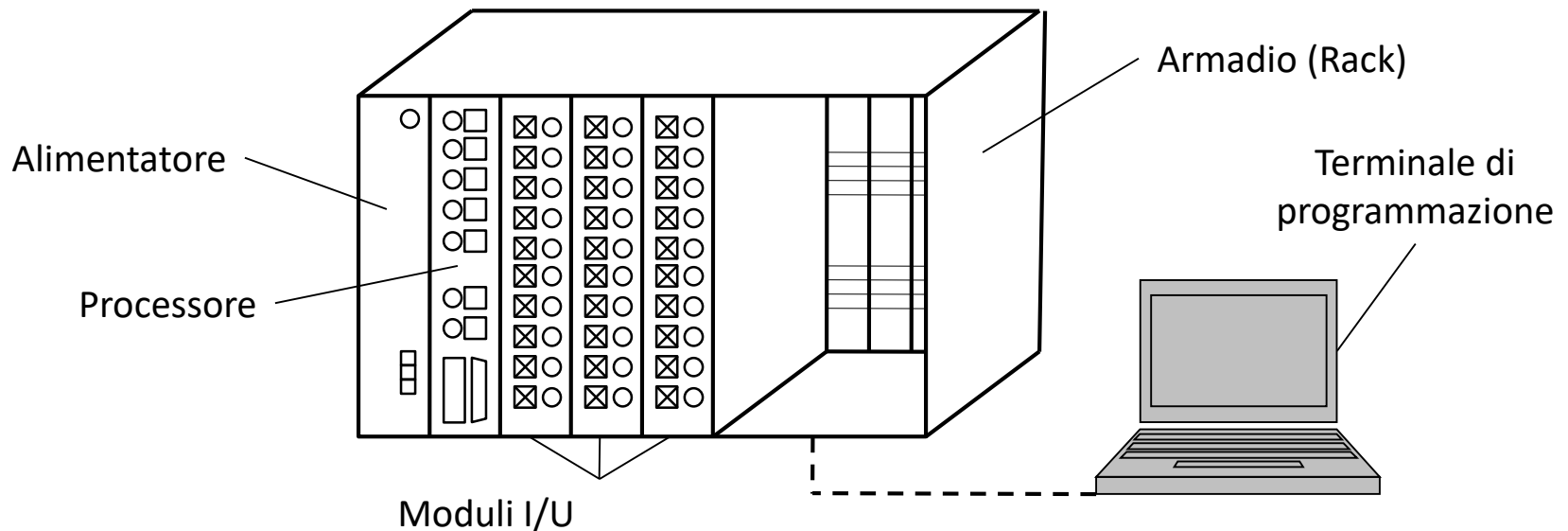
- **Standard 61131-1 CEI** (*Comitato Elettronico Internazionale*)
 - **PLC**: sistema elettronico a funzionamento digitale, destinato all'uso in ambito industriale, che utilizza una memoria programmabile per l'archiviazione interna di istruzioni orientate all'utilizzatore per l'implementazione di funzioni specifiche, come quelle logiche, di sequenziamento, di temporizzazione, di conteggio e di calcolo aritmetico, e per controllare, mediante ingressi e uscite sia digitali sia analogiche, vari tipi di macchine e processi

controllore a logica programmabile

PLC e sistema PLC



- **Sistema PLC:** configurazione realizzata dall'utilizzatore, formata da un controllore a logica programmabile (PLC) e dalle periferiche associate, necessaria all'automazione del sistema previsto
- Configurazione minima di un PLC:



controllore a logica programmabile

PLC e sistema PLC



- **Armadio (rack):** contiene e racchiude tutti gli altri moduli, assicurandone la connessione meccanica e il collegamento elettrico
 - Forma di un parallelepipedo, aperto su di un lato per permettere l'inserimento dei moduli che sono collegati elettricamente tra di loro grazie alla presenza sul lato opposto di un circuito stampato con dei connettori
- **Modulo processore:** è il vero e proprio PLC ed è costituito essenzialmente da una scheda a microprocessore con architettura simile a quella dei calcolatori convenzionali
 - Controlla e supervisiona tutte le operazioni eseguite all'interno del sistema
 - Esegue le istruzioni contenute nella memoria

controllore a logica programmabile

PLC e sistema PLC



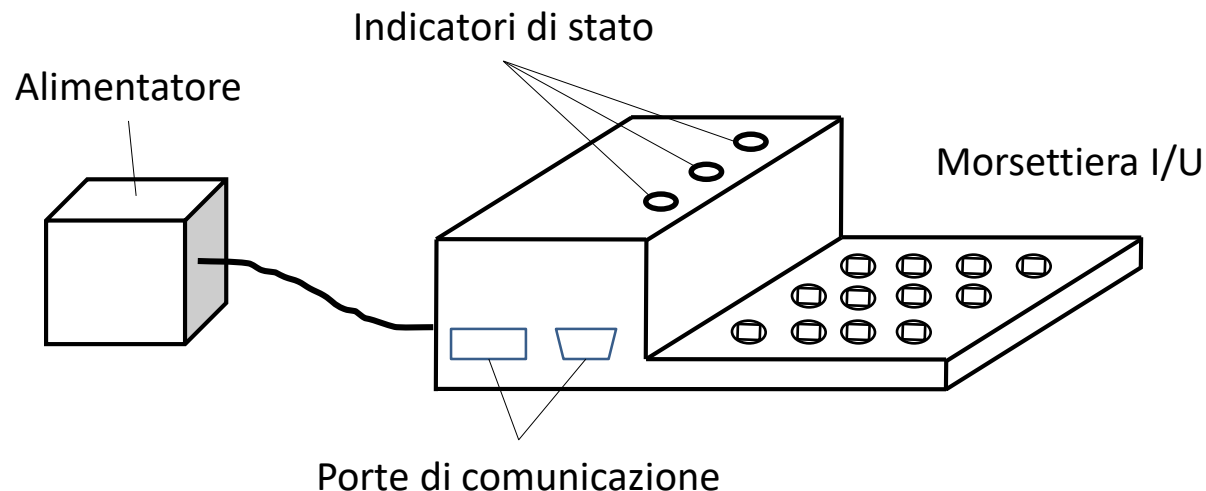
- **Moduli ingresso/uscita (I/U):** sono delle schede che permettono l'interfacciamento tra la microelettronica del PLC e del mondo esterno
 - Provvedono al *condizionamento dei segnali* e all'*isolamento*
- **Modulo alimentatore:** è una scheda che alimenta tutti gli altri moduli presenti nell'armadio
 - Connesso alla rete elettrica, fornisce una o più tensioni stabilizzate
- **Terminale di programmazione:** tipicamente si tratta di un personal computer e serve per la programmazione del PLC
 - Viene connesso al PLC tramite rete o una connessione seriale solo all'atto della programmazione

controllore a logica programmabile

PLC e sistema PLC



- Esistono PLC di piccole dimensioni di tipo *non modulari*
 - Sono controllori monolitici che non prevedono il rack
 - Racchiudono tutti o alcuni degli elementi prima descritti in una unica scheda



controllore a logica programmabile

Modulo processore



- Modulo processore
 - Racchiude una scheda con uno o più processori che eseguono i programmi del sistema operativo e quelli utente, e una memoria dove questi programmi sono conservati
 - Sono utilizzati **microprocessori** come unità di elaborazione centrale
 - Alcuni sono appositamente progettati per consentire l'interpretazione diretta di istruzioni sul singolo bit, molto diffuse nel controllo logico/sequenziale

controllore a logica programmabile

Modulo processore



- Alcune realizzazioni prevedono **tre microprocessori**:
 - Uno specializzato per operare sul singolo bit
 - Uno per le istruzioni di tipo aritmetico/logico
 - Uno dedicato alle comunicazioni con i moduli di ingresso/uscita e con i dispositivi esterni
- Modalità di funzionamento più diffusa è quella *ciclica*
 1. Aggiornamento dell'area di memoria appositamente riservata con i valori provenienti dagli ingressi fisici
 2. Esecuzione del programma/i utente
 3. Esecuzione dei programmi di gestione del sistema
 - Diagnostica
 4. Scrittura sulle uscite fisiche dei loro valori conservati nell'area di memoria appositamente riservata

controllore a logica programmabile

Modulo processore



- Il ciclo descritto prende anche il nome di “**ciclo a copia massiva degli ingressi e delle uscite**”
 - Ottimizza la comunicazione con i moduli di ingresso/uscita
 - Garantisce che i valori memorizzati degli ingressi restino invariati durante l’esecuzione dei programmi → coerenza
 - La lettura degli ingressi e la scrittura delle uscite è gestita completamente dal sistema operativo → non a carico dell’utente
 - Il sistema rimane “cieco” fino alla prossima lettura
 - Ritardo nella rilevazione del cambiamento degli ingressi al più pari al tempo di ciclo
 - Se un ingresso digitale varia due volte durante un ciclo di scansione, la variazione non viene e non verrà mai più rilevata
 - Durata del ciclo non prefissata → dipende dalla lunghezza e natura del programma → PLC non adatto per chiudere anelli di controllo

controllore a logica programmabile

Modulo processore



- Esistono specifiche istruzioni per realizzare l'**accesso immediato agli ingressi e alle uscite**
 - Sono utilizzati nei casi in cui non è tollerabile l'attesa della fine del ciclo per leggere un ingresso o scrivere una uscita
 - Allungano i tempi di esecuzione del programma → non è consigliabile l'abuso di tali istruzioni → normalmente destinate alla gestione delle condizioni di emergenza
- Gestione degli interrupt temporizzati o collegati allo stato dei segnali di ingresso → eccezione al normale ciclo a copia massiva

controllore a logica programmabile

Modulo processore



– Tempo di scansione (T_s)

- Descrive la velocità di elaborazione del modulo processore
- *Definizione*: tempo che intercorre tra due attivazioni successive della stessa porzione di programma applicativo nella modalità di funzionamento ciclico
- Da indicazioni sul tempo necessario ad eseguire il ciclo di funzionamento → dipende dal numero di ingressi e uscite da aggiornare e dalla dimensione e complessità del programma utente
- Viene indicato dal produttore come valore medio per programmi di media complessità
 - Varia da qualche unità a qualche decina di millisecondi per Kiloword (1024 parole) di programma a seconda della classe del PLC

controllore a logica programmabile

Modulo processore



– Tempo di risposta (T_r)

- *Definizione*: massimo intervallo di tempo che passa tra la rilevazione di un certo evento e l'esecuzione dell'azione di risposta programmata
- Tiene conto anche dei ritardi introdotti dai moduli di ingresso/uscita
- Relazione tra il tempi di scansione e il tempo di risposta, riferita al caso peggiore:

$$T_r \approx 2T_s + T_{i/u}$$

dove T_r è il tempo di risposta, T_s è il tempo di scansione, $T_{i/u}$ è il ritardo introdotto dai moduli di ingresso/uscita

controllore a logica programmabile

Modulo processore



– Sistema operativo

- Insieme di programmi di supervisione memorizzati in modo permanente nel PLC, dedicati a
 - Controllo delle attività
 - Elaborazione dei programmi utente
 - Comunicazione
 - Altre funzioni (esempio: *watchdog timer*, controlli di parità sulla memoria e sulle linee di comunicazione, controllo della tensione di alimentazione, controllo dello stato della batteria tampone, ecc.)
- *Nota:* le funzioni di controllo possono attivare degli indicatori di stato ed avviare delle routine di emergenza programmabili dall'utente

controllore a logica programmabile

Modulo processore



– Modalità operative di un PLC

- Attivabili attraverso una chiave hardware, sono:
 1. *Esecuzione*
 2. *Convalida*
 3. *Programmazione*
- Modalità di **esecuzione**
 - Programmi utente sono eseguiti
 - Ingressi ed uscite sono aggiornate
- Modalità di **convalida**
 - Programmi utente sono eseguiti
 - Aggiornamento delle uscite disabilitato
- Modalità di **programmazione**
 - Consente di caricare nella memoria il programma sviluppato

controllore a logica programmabile

Modulo processore



– Modalità di esecuzione dei programmi

- Ancora oggi esistono PLC nei quali i programmi vengono *interpretati*
 - Ogni istruzione è convertita in linguaggio macchina immediatamente prima di essere eseguita
- La diffusione di architetture basate su processori convenzionali e l'uso di personal computer come terminali di programmazione ha portato alla diffusione di PLC per i quali è prevista una fase di *compilazione* prima dell'esecuzione del programma

controllore a logica programmabile

Modulo processore



– Memoria

- Può tipicamente essere ripartita in 5 macro aree:
 - *Area sistema operativo*
 - » Riservata alla memorizzazione permanente del sistema operativo
 - » Di tipo non volatile
 - *Area di lavoro del sistema operativo*
 - » Riservata alla memorizzazione di dati intermedi da parte dei programmi del sistema operativo
 - » Deve permettere la lettura e la scrittura dei dati (memoria RAM)
 - *Area ingressi/uscite*
 - » Riservata alla memorizzazione degli stati degli ingressi e uscite
 - » Memoria di tipo RAM

controllore a logica programmabile

Modulo processore



- *Area utente*
 - » Riservata alla memorizzazione dei programmi utente
 - » Memoria di tipo RAM durante la programmazione, sostituibile con una di tipo PROM una volta realizzato e verificato il programma
- *Area dati utente*
 - » Riservata alla memorizzazione dei dati dei programmi utente
 - » Memoria di tipo RAM
- Aree di tipo RAM possono essere in parte o tutte alimentate da **batterie tampone** per evitare perdite di dati in caso di mancanza di alimentazione
 - Tecnologia CMOS per limitare il consumo
 - In caso di mancanza di alimentazione il sistema operativo e i programmi utente possono gestire la *modalità di ripristino* del funzionamento grazie ad appositi bit di stato

controllore a logica programmabile

Modulo processore



– Dati forniti dai costruttori

- Memoria (dichiarata)
 - Non sempre è chiaro a quali aree della memoria si riferisca (dovrebbe essere l'are utente, ma non sempre è così!)
 - Variano da mezzo Kiloword a qualche centinaio di Kiloword, con word di 8 o 16 bit, fino al Megaword per i PLC di ultima generazione
- Limiti sul numero e sul tipo di ingressi e uscite gestibili
- Limiti sul numero di strutture speciali (timer, contatori)
- Limiti sul numero di word a disposizione per altri dati
- Possibilità di espandere la memoria
- Numero di armadi di I/U gestibili direttamente
- Numero di armadi gestibili in modalità remota con i relativi tempi per la scansione



controllore a logica programmabile

Modulo processore

- Numero e qualità delle porte di comunicazione disponibili
 - Seriali, parallele, di rete
- Tipo di linguaggi supportati e complessità dell'insieme delle istruzioni
- Possibilità del multitasking
- Possibilità di gestire routine di interruzione
- PLC di sicurezza
 - Sono progettati per essere impiegati in applicazioni che richiedono gradi di sicurezza molto elevati (*esempio*: automazione di presse)
 - Prevedono una **ridondanza di unità elaborative**
 - Eseguono lo stesso programma e abilitano le uscite solo se c'è pieno accordo tra di loro

controllore a logica programmabile

Moduli ingresso/uscita



- Moduli ingresso/uscita
 - Sono i moduli con cui il PLC comunica con il processo fisico
 - Rilevano eventi e dati dai sensori
 - Comandano azioni agli attuatori
 - Devono realizzare l'interfaccia tra i livelli di tensione TTL o CMOS del PLC con le tensioni o correnti usate per la trasmissione dei segnali
 - Possibile connettere il PLC direttamente ai dispositivi sul campo
 - Ingressi e uscite spesso isolati galvanicamente (fotoaccoppiatori o trasformatori) dall'elettronica interna del PLC
 - Modularità e varietà
 - Possibile realizzare sistemi PLC su misura per l'applicazione



controllore a logica programmabile

Moduli ingresso/uscita

– Indirizzamento

- Dipende da dove i moduli sono allocati nell'armadio, o negli armadi, del PLC
- Sono previsti degli accorgimenti meccanici per evitare che un modulo venga installato in posizioni diverse da quelle previste

– Moduli di ingresso digitali

- Forniti di **circuiti di filtraggio** contro il rumore e i rimbalzi → introducono **ritardi** dell'ordine di qualche millisecondo nella rilevazione degli ingressi
- Forniti di indicatori di stato prima e dopo i circuiti di isolamento
- Sono riportate informazioni tecniche sul numero di ingressi gestibili, le tensioni di riferimento, il ritardo di segnalazione introdotto



controllore a logica programmabile

Moduli ingresso/uscita

– Moduli di uscita digitali

- Uscite normalmente protette da fusibili
- Realizzati mediante:
 - Transistori → per correnti continue
 - TRIAC o SCR → per correnti alternate
 - Relè → per correnti continue ed alternate

– Moduli di ingresso/uscita analogici

- Realizzano le conversioni digitali/analogiche o analogiche/digitali richieste per interfacciare direttamente segnali analogici con il PLC
- Trattano un'ampia gamma di segnali
- Intervalli di lavoro e caratteristiche filtranti programmabili
- Scalatura automatica in unità ingegneristiche del dato
- Sono optoisolati e possiedono indicatori di stato

controllore a logica programmabile

Moduli ingresso/uscita



- Valori tecnici riportati dai costruttori:
 - Valori dei segnali trattati
 - Possibilità di accettare ingressi single-ended o differenziali
 - La risoluzione di conversione
 - La rappresentazione dei dati fornita
 - La velocità di conversione
- Alcuni moduli di ingresso utilizzano un solo convertitore analogico/digitale ed un multiplexer per ridurre i costi
- Alcuni moduli di uscita possono far assumere un valore prefissato nel caso in cui dal PLC non arrivino nuove informazioni entro un tempo prefissato
- Considerata la loro diffusione e relativa complessità di gestione, esistono moduli specifici per l'utilizzo diretto di sensori di temperatura resistivi a metallo (RTD) e le termocoppie

controllore a logica programmabile

Modulo alimentatore



- Modulo alimentatore
 - Fornisce attraverso l'armadio l'alimentazione elettrica stabilizzata necessaria al funzionamento degli altri moduli
 - Deve assicurare una tensione costante anche in presenza di *microinterruzioni* o *fluttuazioni* della fornitura elettrica
 - È composto da
 - Un trasformatore
 - Un circuito rettificatore
 - Un filtro
 - Un circuito stabilizzatore
 - Un circuito per la protezione da sovracorrenti o cortocircuiti

controllore a logica programmabile

Modulo alimentatore



– Caratteristiche tecniche

- Massima potenza fornibile
- Possibilità di connessione in parallelo → aumentare la massima potenza o realizzare una ridondanza di alimentazione
- Possibilità di inviare al PLC una segnalazione di shutdown se l'alimentazione di ingresso scende sotto certi limiti → il PLC avvierà le procedure programmate prima dell'effettivo spegnimento
- Presenza di indicatori sul suo stato

– Dimensionamento

- Va fatto considerando la somma delle potenze richieste dai moduli previsti, aumentata di una certa percentuale per tener conto di eventuali espansioni future

controllore a logica programmabile

Armadio



- Armadio (Rack)
 - Deve contenere i moduli di un sistema PLC ed assicurarne la connessione elettrica/meccanica e la schermatura
 - La connessione elettrica è realizzata attraverso il **bus**: un insieme di linee elettriche, raggruppate per funzioni (linee indirizzo, linee dati, linee di controllo, linee di alimentazione) e dei protocolli associati, attraverso le quali un modulo può comunicare con gli altri
 - Si tratta di bus proprietari → impossibile utilizzare moduli di costruttori differenti
 - Caratteristiche di tipo meccanico (numero di slot, dimensioni di ingombro, modalità di fissaggio, ...)

controllore a logica programmabile

Terminale di programmazione



- Terminale di programmazione
 - Il PLC non prevede tastiere e schermi per la comunicazione con il programmatore → occorrono dispositivi particolari per la programmazione
 - Per i piccoli PLC si utilizzano ancora *terminali a tastiera*
 - Si connettono direttamente al PLC attraverso una porta di comunicazione (seriale)
 - Presentano all'operatore una tastiera ed un piccolo display a cristalli liquidi
 - Programmazione direttamente nella memoria del PLC

controllore a logica programmabile

Terminale di programmazione



- *Sistemi di sviluppo* basati su personal computer
 - Programmazione off-line
 - Presentano funzionalità di composizione dei programmi molto complesse
 - Capacità di memorizzazione permanente dei programmi sviluppati
 - Sono connessi al PLC direttamente o tramite una rete informatica
 - Con un solo personal computer è possibile programmare tutti i PLC connessi in rete
 - Sono previste funzioni di **monitoraggio** dell'esecuzione del programma e delle aree di memoria eseguibili anche durante il normale funzionamento del dispositivo

controllore a logica programmabile

Moduli speciali



- Moduli speciali
 - Moduli di I/U remoto
 - Utili quando il numero di ingressi e uscite è elevato e sono dislocati su aree di impianto estese
 - Si installano armadi di I/U sparsi nell'impianto, collegati al PLC attraverso un modulo di I/U remoto che provvede ad inviare lo stato degli ingressi e delle uscite dell'armadio in cui è montato attraverso una *linea seriale* o una *rete informatica*
 - Moduli per la connessione di rete
 - Gestiscono i protocolli di comunicazione per le diverse tipologie di reti informatiche che possono coinvolgere il PLC
 - Bus di campo, reti proprietarie, Ethernet

controllore a logica programmabile

Moduli speciali



– Moduli coprocessore

- Sono moduli contenenti un calcolatore convenzionale in grado di accedere direttamente alla memoria del PLC
- Possibilità di eseguire elaborazioni complesse
- Programmazione in linguaggi di alto livello (C o Basic)
- Possono comprendere una unità di memorizzazione di massa
- Possono comunicare direttamente con l'esterno attraverso interfacce classiche come la seriale RS-232, la parallela Centronics, lo standard PCMCIA

– Moduli PID

- Consente di disporre di alcuni anelli PID a cui il PLC fornisce solo i riferimenti
- Possono essere auto-sincronizzanti con cambio modalità neutro

controllore a logica programmabile

Moduli speciali



– Moduli servo

- Realizzano direttamente ed autonomamente l'asservimento di uno o più motori a passo, motori idraulici, motori in corrente continua con encoder incrementali

– Moduli encoder

- Realizzano le funzionalità necessarie per utilizzare uno o più encoder incrementali o assoluti
- Contengono contatori ad alta velocità

– Moduli interfaccia operatore

- Consentono l'interfacciamento di un operatore con il PLC mediante una tastiera ed un display alfanumerico
- Oggi poco diffusi e soppiantati da comuni PC connessi in rete

controllore a logica programmabile

Moduli speciali



– Moduli di backup

- Sono moduli che vanno inseriti in due PLC distinti e consentono di avere una funzionalità di backup
- *Sistema a massima disponibilità* → il modulo processore di riserva, attraverso tali moduli, viene informato sullo stato del modulo processore principale e lo può sostituire in tempi brevissimi, in caso di malfunzionamento di quest'ultimo, nella gestione degli ingressi e delle uscite
 - Eventualmente si può avere una duplicazione anche degli altri moduli che compongono il sistema
- *Sistema a sicurezza intrinseca*: prevede che i due moduli processore siano concordi nel decidere lo stato delle uscite prima che questa sia effettivamente applicata

– Moduli lettori di codici a barre, sistemi di visione, ...



controllore a logica programmabile

Classificazione dei PLC

- Classificazione dei PLC
 - Micro PLC
 - Trattano fino a 64 punti di ingresso/uscita, generalmente digitali
 - Memorie di 1 o 2 Kiloword
 - Non hanno una struttura modulare ad armadio (sistemi monolitici)
 - Possono prevedere delle espansioni degli I/U
 - Programmabili con un solo linguaggio di programmazione
 - Linguaggio a contatti
 - Insieme di istruzioni, funzioni e blocchi funzionali limitato
 - Utilizzati normalmente in sostituzione di logiche a relè in applicazioni come il controllo di macchine operatrici, di ascensori, di lavatrici

controllore a logica programmabile

Classificazione dei PLC



— Piccoli PLC

- Trattano da 64 a 512 punti di ingresso/uscita, in dominanza digitali
- Memorie fino a 4 Kiloword
- Hanno una struttura modulare ad armadio
- Possono avere capacità di connessione in rete e gestione di I/U remoto
- Programmabili con differenti linguaggio di programmazione
 - Insieme di istruzioni, funzioni e blocchi funzionali più completo



controllore a logica programmabile

Classificazione dei PLC

– Medi PLC

- Arrivano a trattare da 256 a 2048 punti di ingresso/uscita
- Memoria fino a qualche decina di Kiloword
- Struttura modulare ad armadio
- Gestiscono l'I/U remoto
- Sono arricchibili con moduli speciali
- Hanno elevate capacità di comunicazione in rete informatica
- Possibilità di programmazione molto evolute



controllore a logica programmabile

Classificazione dei PLC

— Grandi PLC

- Numerosi punti di ingresso/uscita (alcune migliaia)
- Memoria di centinaia di Kiloword
- Notevole capacità di trattamento delle informazioni
- Utilizzati come supervisori di celle automatizzate e come interfacciamento tra PLC di minori prestazioni e i calcolatori di gestione

controllore a logica programmabile

Linguaggi di programmazione



- Linguaggi di programmazione
 - Lo **standard IEC 61131** definisce la standardizzazione dei linguaggi di programmazione per i PLC
 - Lo standard è un *modello di riferimento* non vincolante
 - Esistono vecchi PLC ancora installati che non rispettano lo standard
 - I nuovi non sempre lo rispettano pienamente
 - Il linguaggio più diffuso per la programmazione dei PLC resta il **linguaggio a contatti** (*ladder diagram*)
 - Esistono diversi “dialetti” per i vari costruttori
 - Molto diffusa anche la **lista istruzioni**
 - Esistono anche in questo caso vari dialetti

controllore a logica programmabile

Linguaggi di programmazione



- Si stanno diffondendo anche il **testo strutturato** e l'**SFC** (*Sequential Functional Chart*)
- Solo pochi PLC, i più evoluti, attualmente prevedono la possibilità di essere programmati direttamente con l'SFC
 - L'SFC è il linguaggio migliore per la programmazione di algoritmi di controllo logico/sequenziale tipici dell'automazione industriale
 - Fortunatamente esistono metodi che consentono la *traduzione* (anche automatica) di programmi scritti in SFC in linguaggio a contatti, che è universalmente accettato da tutti i tipi di PLC



II PARTE

Controllo distribuito

Produzione integrata tramite elaboratori

Reti per l'automazione

Sistemi per il controllo di supervisione e l'acquisizione dati (SCADA)

Sistemi per l'esecuzione della produzione (MES) e controllo distribuito (DCS)

controllo distribuito

Generalità



- Controllo distribuito
 - Sistemi complessi di automazione industriale richiedono una notevole integrazione tra i vari sottosistemi che li compongono → **sistemi di controllo distribuito**
 - Necessaria la possibilità di *comunicazione* tra i dispositivi
 - L'informazione scambiata diviene uno dei fattori della produzione
 - L'informazione deve essere:
 - *Raccolta* → deve essere sempre aggiornata
 - *Pre-elaborata* → deve essere di qualità
 - *Aggregata o compressa* → deve essere nella giusta quantità
 - *Accumulata* → deve essere sempre disponibile
 - *Trasmessa* → deve essere disponibile dove serve
 - *Elaborata* → deve realizzare lo scopo per il quale è stata trasmessa

produzione integrata tramite elaboratore

Generalità



- Automazione industriale e controllo di processo
 - Presenza di celle di lavorazione automatizzate dedicate all'automazione di singole lavorazioni
 - Lavorazioni ripetitive e/o pericolose per l'uomo
 - Qualità superiore a costi ridotti per i prodotti
 - Scarsa integrazione tra le isole di lavorazione
 - Sono affrontati i problemi relativi alla singola lavorazione
 - Isole realizzate mediante dispositivi di controllo eterogenei tra loro, con protocolli di comunicazione spesso incompatibili
 - Modello di automazione comunque vantaggioso
 - Aumento di produttività, flessibilità di lavorazione, qualità uniforme del prodotto

produzione integrata tramite elaboratore

Generalità



- Soluzione migliorativa → *integrazione* dei sottosistemi
 - Necessaria una progettazione metodica dell'intero sistema informatico
 - Utilizzo di dispositivi di controllo standardizzati
 - Gestione dei flussi di informazione tra i vari dispositivi
 - Coordinamento tra i fattori produttivi, compreso l'uomo
 - L'uomo è impiegato in ruoli diversi: gestore, supervisore e manutentore dei sistemi automatizzati
 - Vantaggi
 - Migliore utilizzo delle risorse → accurata pianificazione dei processi produttivi, eventualmente eseguita in tempo reale
 - Massima flessibilità della produzione → adeguamento rapido per nuove operazioni

produzione integrata tramite elaboratore

Generalità



- Riduzione dei tempi di produzione
- Miglioramento della progettazione dei prodotti → necessità di fornire alle macchine informazioni non ambigue
- Identificazione, conservazione e riutilizzo delle informazioni relative ai prodotti → si evita di doverli reimpostare riducendo la possibilità di commettere errori
- Miglioramento dei controlli sulla produzione e della qualità degli stessi controlli → esempio, controllo estensivo invece che a campione
- Riduzione degli scarti di lavorazione
- Riduzione delle scorte di materie prime e dei magazzini per i prodotti finiti → produzione just-in-time

produzione integrata tramite elaboratore

Modello CIM



- **CIM** (*Computer Integrated Manufacturing*)
 - È un *modello di riferimento* funzionale per la realizzazione dell'automazione industriale e del controllo di processo basato sul *rilevamento*, il *coordinamento* e la *trasmissione* di **informazioni** tra i vari sottosistemi mediante l'utilizzo di *reti informatiche*

produzione integrata tramite elaboratore

Modello CIM



- Può essere rappresentata con una struttura piramidale a 6 livelli funzionali:



produzione integrata tramite elaboratore

Modello CIM



- Livello 1: *piano officina o campo*
 - Realizza le funzioni di misura e di comando sui processi produttivi
 - Dispositivi che la attuano sono quelli che interagiscono direttamente con i processi fisici (sensori ed attuatori)
- Livello 2: *sistemi di controllo*
 - Funzioni di controllo di macchine o processi e le funzioni di sicurezza
 - Sono impiegati dispositivi di controllo diretto (interagisco direttamente con i sensori e gli attuatori), come i PLC, i regolatori, i controllori di robot e di macchine a controllo numerico
- Livello 3: *supervisione di cella*
 - Necessità di coordinamento delle macchine e dei relativi sistemi di controllo delle celle di lavorazione automatizzate
 - Configurazione dei parametri dei sistemi di controllo e il coordinamento delle sequenze di attività da loro svolte, anche per realizzare un comportamento ottimale della cella



produzione integrata tramite elaboratore

Modello CIM

- Spesso è richiesto il coinvolgimento dell'operatore umano: devono essere presentate informazioni utili e necessarie in una forma intellegibile all'uomo che può decidere di impartire comandi o impostare dei riferimenti
- Dispositivi impiegati sono personal computer o dispositivi di controllo diretto come i PLC con maggiori capacità elaborative
- **Livello 4: *supervisione integrata* della produzione**
 - Viene gestita la base dati della produzione ed è realizzato il coordinamento tra le varie celle per realizzare l'intero processo produttivo
 - Interazione con gli operatori umani molto sofisticata
 - Pianificazione delle attività in funzione dello stato del sistema
 - Dispositivi utilizzati sono workstation o personal computer di classe superiore su cui sono eseguiti gli applicativi SCADA e MES

produzione integrata tramite elaboratore

Modello CIM



- Livello 5: *gestione stabilimento*
 - Sono integrati i vari comparti dello stabilimento, come la produzione, la logistica, l'amministrazione e la manutenzione
 - Pianificazione delle attività della produzione
 - Gestione delle risorse a livello aziendale
 - Pianificazione degli acquisti
 - Sono utilizzati i calcolatori che fanno parte del sistema informatico aziendale
- Livello 6: *gestione azienda*
 - Azienda può essere composta da più stabilimenti
 - Sono raccolte informazioni dai livelli inferiori per realizzare sistemi di *supporto alla decisione* che aiutino gli amministratori a pianificare i flussi fisici (materiali) e finanziari (investimenti) necessari al mantenimento dei livelli di produzione e al loro miglioramento

produzione integrata tramite elaboratore

Modello CIM



– Rappresentazione piramidale

- Rappresenta una organizzazione gerarchica
 - Ogni livello comunica direttamente con quello immediatamente superiore, da cui riceve comandi e cui fornisce informazioni, e con quello immediatamente inferiore, cui invia comandi e da cui riceve informazioni
- Rappresenta le differenti caratteristiche dei flussi di informazione tra i vari livelli
 - Partendo dal livello più basso e proseguendo verso l'alto sono necessarie via via minori quantità di informazioni scambiate, di maggiore qualità e con frequenze di aggiornamento inferiori

produzione integrata tramite elaboratore

Modello CIM



- Rappresenta nei livelli inferiori funzioni vicine all'impianto (necessaria conoscenza dei processi da automatizzare), nei livelli superiori funzioni più lontane che realizzano al supervisione e la pianificazione delle attività
 - Necessità di interazione con l'uomo è minima nei livelli inferiori (limitata a qualche segnalazione luminosa) ed è massima nei livelli superiori (interfaccia uomo-macchina molto sofisticata)
 - I dispositivi che implementano le funzioni di livello più basso devono avere tempi di risposta più piccoli e certi di quelli che implementano funzioni di livello più alto
- Livelli da 1 a 4 → automazione
- Funzioni di controllo e regolazione, anche se di alto livello

produzione integrata tramite elaboratore

Modello CIM semplificato



- Modello CIM solo di riferimento → modello di automazione industriale *semplificato* con soli due livelli
 - Livello 1: *automazione di basso livello*
 - Comprende tutti i dispositivi di controllo e di coordinamento
 - Funzioni previste dai livelli 1 e 2 e alcune del 3 del CIM
 - Livello 2: *automazione di alto livello*
 - Identifica i dispositivi di supervisione e acquisizione dati
 - Funzioni previste dal livello 4 e alcuni dei livelli 3 e 5 del CIM

reti per l'automazione

Introduzione



- Reti per l'automazione
 - Automazione integrata → necessario l'uso di **reti di comunicazione**
 - Reti informatiche consentono:
 - La comunicazione tra dispositivi diversi, sia per la tecnologia costruttiva che per funzionalità espletate
 - La possibilità di aggiungere o rimuovere dispositivi senza dover interrompere la produzione
 - Automazione usa le reti, non le progetta
 - Necessaria una profonda conoscenza delle caratteristiche di tutte le reti informatiche adatte all'automazione industriale per tutti i livelli del modello CIM

reti per l'automazione

Il modello OSI



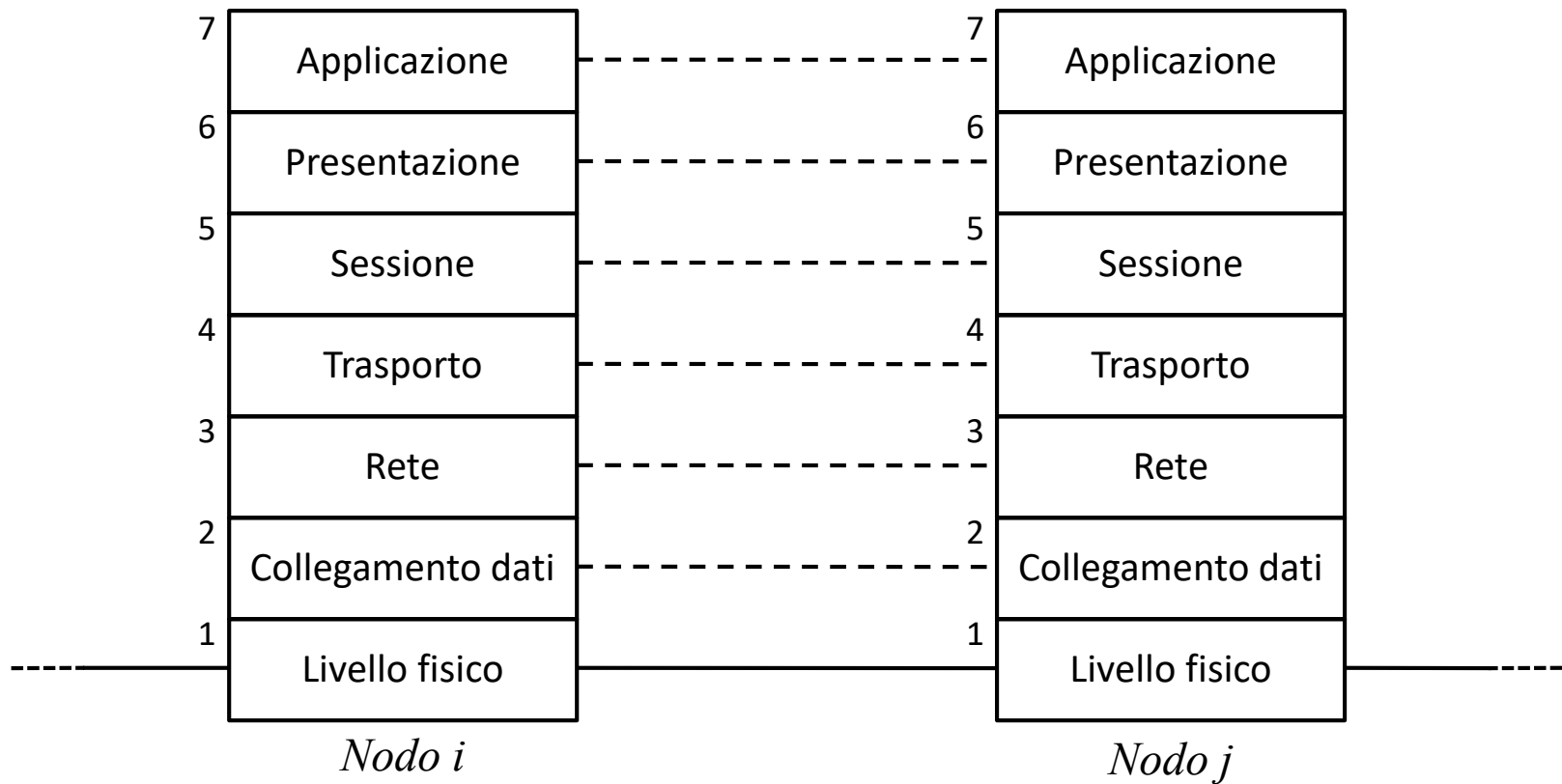
- **Modello OSI** (*Open System Interconnection*)
 - Si tratta di un modello di riferimento per le reti informatiche elaborato dall'ISO (*International Standards Organization*)
 - Non descrive nessuna rete informatica in particolare
 - In tale modello ogni nodo della rete è visto come una successione gerarchica di sette livelli
 - Ogni livello comunica direttamente solo con il livello immediatamente sottostante (richiedendo dei servizi) e immediatamente sovrastante (fornendo dei servizi) per mezzo di interfacce
 - Lo standard definisce i servizi che ogni livello deve fornire e quelli che può richiedere, non come questi sono implementati

reti per l'automazione

Il modello OSI



- Schematizzazione del modello OSI:



reti per l'automazione

Il modello OSI



- Livello 1: *livello fisico*
 - Si occupa di trasmissione di sequenze binarie tra due nodi
 - Comprende la definizione dei collegamenti meccanici ed elettrici tra i nodi e i driver software per le porte di comunicazione
 - Fissa parametri della rete come la velocità e le modalità di trasmissione dei bit
 - È l'unico livello in cui vi è una effettiva connessione tra i nodi
- Livello 2: *collegamento dati*
 - Realizzato via software, come tutti i livelli superiori
 - Viene verificato il corretto passaggio tra i nodi delle sequenze di bit organizzate in strutture standard (*frame*)
 - I frame sono composti in partenza, aggiungendo dei codici di controllo all'informazione da trasmettere, e verificati all'arrivo
 - In caso di errore è possibile recuperare la sequenza correggendole o richiedendone la ritrasmissione

reti per l'automazione

Il modello OSI



- Viene assicurata una trasmissione dei dati senza errori tra i nodi interconnessi
- Viene definito come i differenti nodi possono accedere al mezzo di trasmissione senza interferire tra loro (**metodo di accesso**)
- Livello 3: *livello di rete*
 - Garantisce un sentiero logico anche tra due nodi non direttamente connessi
 - Vengono definite le caratteristiche della rete dal punto di vista della sua gestione
 - Vengono implementate le funzioni di instradamento e di interconnessione tra i nodi
- Livello 4: *livello di trasporto*
 - Realizza le funzioni di trasporto dell'informazione indipendenti dalla struttura della rete

reti per l'automazione

Il modello OSI



- Assicura il trasferimento dei messaggi in maniera affidabile, accertandosi dell'integrità del messaggio ricevuto, composto da più sequenze di bit (pacchetti), ritrasmettendo in caso di necessità
- Viene realizzata l'interfaccia tra la rete e il software applicativo dei successivi tre livelli (area utente)
- Livello 5: *livello di sessione*
 - Gestisce lo scambio ordinario dei dati e il sincronismo tra i nodi → rende possibile la connessione remota
 - Comprende le funzioni necessarie a supportare lo scambio di informazioni tra differenti macchine → consente l'apertura, la gestione e la chiusura di un canale di comunicazione
- Livello 6: *livello di presentazione*
 - Realizza la codifica delle informazioni e la loro conversione → i dati binari in sequenza sono collegati al loro significato e rappresentati in forma di testi, figure o altro

reti per l'automazione

Il modello OSI



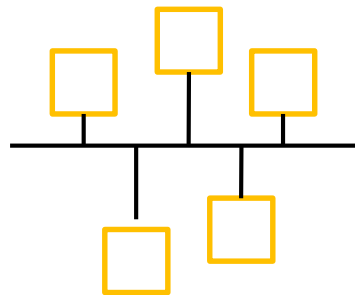
- *Livello 7: livello di applicazione*
 - Fornisce interfacce e servizi ai programmi applicativi come il trasferimento di file tra i nodi, la possibilità di operare su database distribuiti, la capacità di controllo remoto delle macchine
- Solo il *livello fisico* realizza la connessione fisica tra i nodi
 - Per i livelli superiori si può supporre l'esistenza di una *connessione virtuale*
 - Ogni livello può quindi interagire direttamente con il precedente ed il successivo sullo stesso nodo e virtualmente con l'omologo livello situato su altri nodi, attraverso i livelli sottostanti in ambedue i nodi
- Reti differenziate in base alla *topologia, mezzo trasmissivo e protocollo di accesso*



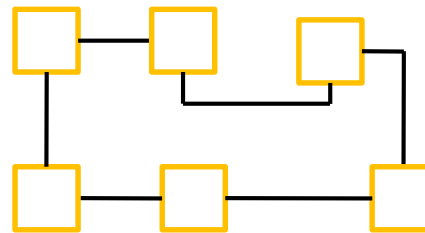
reti per l'automazione

Il modello OSI – Topologia

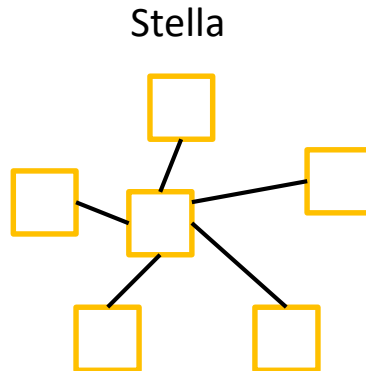
- Topologia
 - Definisce l'architettura fisica della rete, come i vari nodi sono collegati tra di loro



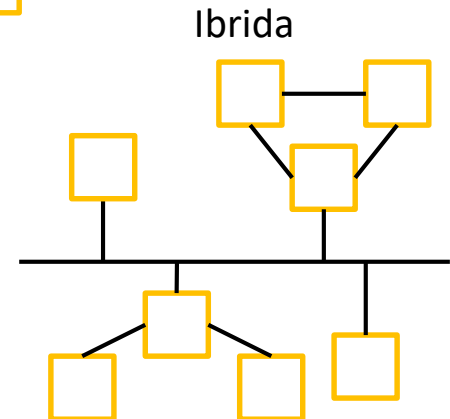
Bus



Ring



Stella



Ibrida

reti per l'automazione

Il modello OSI – Topologia



- Topologia *Bus* (anello aperto)
 - Non comporta problemi per l'aggiunta o la rimozione di nodi → le informazioni viaggiano in maniera indipendente
 - Esiste un unico instradamento possibile tra i nodi
- Topologia *Ring* (anello chiuso)
 - I messaggi attraversano i nodi → i nodi devono essere in grado di identificare a chi è stato trasmesso il messaggio ed eventualmente rispedirlo
- Topologia a *Stella*
 - Esiste un nodo primario connesso direttamente a tutti gli altri nodi → tutti i messaggi devono passare necessariamente per il nodo primario
- Tipologie *ibride* sono possibili

reti per l'automazione

Il modello OSI – Mezzo di trasmissione



- Mezzo di trasmissione
 - Definisce il supporto fisico attraverso cui passano le informazioni
 - Le sue caratteristiche influenzano le prestazioni della rete
 - *Doppino intrecciato (twisted pair)*
 - Formato da due conduttori isolati (eventualmente schermato)
 - Presenta una impedenza elettrica irregolare e una scarsa immunità ai disturbi
 - Massima distanza limitata a qualche centinaio di metri
 - Velocità di trasmissione limitata a qualche Mbit/s
 - Economico e facile da installare



reti per l'automazione

Il modello OSI – Mezzo di trasmissione

– *Cavo coassiale*

- Costituito da un conduttore centrale con uno schermo
- Risolve i problemi del doppino intrecciato
- Costo superiore e maggiore complessità di installazione
- Ampia larghezza di banda → può sostenere più canali di comunicazione modulati in frequenza

– *Fibra ottica*

- Permette la trasmissione di informazioni utilizzando segnali luminosi
- È il mezzo di trasmissione migliore, soprattutto in termini di velocità di trasmissione, della banda (possibilità di avere canali multipli), dell'immunità ai disturbi
- Molto costosa e l'installazione richiede personale specializzato

reti per l'automazione

Il modello OSI – Metodo di accesso



- Metodo di accesso
 - È una delle funzioni implementate al livello 2 del modello OSI (collegamento dati)
 - Definisce due aspetti dell'utilizzo della rete:
 - In ricezione → determinare a quali nodi è effettivamente destinato il messaggio e quale nodo lo ha generato
 - In trasmissione → evitare conflitti tra più nodi che vogliono contemporaneamente utilizzare il canale di trasmissione
 - Le interfacce di rete sono dotate di un indirizzo univoco a livello mondiale assegnato dal costruttore
 - Il metodo di accesso passerà alle altre funzioni di livello 2 il messaggio presente sul canale solo se riconosce il suo indirizzo

reti per l'automazione

Il modello OSI – Metodo di accesso



– Risoluzione dei conflitti in trasmissione

- Metodo di accesso *centralizzato o polling o master/slave*
 - Prevede l'esistenza di un nodo principale (*master*) che interroga sistematicamente gli altri nodi (*slave*)
 - Metodo deterministico e affidabile
 - Basse velocità di trasmissione
 - Non è assegnabile una priorità di accesso
 - Metodo semplice da gestire ed implementare
- Metodo di accesso a *gettone (token)*
 - Prevede l'esistenza di una particolare stringa di bit, il *token* o *gettone*, che viene passata in sequenza tra tutti i nodi della rete
 - » Quando una stazione riceve il token può trasmettere
 - » Il token non può essere trattenuto oltre un certo tempo



reti per l'automazione

Il modello OSI – Metodo di accesso

- Metodo di accesso deterministico → noto tempo massimo necessario affinché due nodi possano comunicarsi un messaggio
- Metodo di accesso *CSMA/CD (Carrier Sense, Multiple Access/Collision Detection)*
 - Ogni nodo, prima di trasmettere, ascolta la portante presente sul mezzo di trasmissione per essere sicuro che il canale sia libero
 - Questo non assicura che più nodi non cerchino di trasmettere contemporaneamente → durante la trasmissione il nodo rimane in ascolto per rilevare eventuali collisioni
 - » In caso di collisioni il nodo ritrasmette il messaggio dopo un periodo di tempo variabile determinato casualmente
 - Non è un metodo di accesso deterministico
 - Efficienza dipende dalla possibilità che si verifichino collisioni, ovvero dal numero dei nodi collegati

reti per l'automazione

Il modello OSI – Metodo di accesso



- Metodo di accesso *CSMA/BA (Carrier Sense, Multiple Access/Bit Arbitration)*
 - Se il nodo rileva libero il canale di trasmissione, inizia la trasmissione, potendo generare conflitti
 - Il meccanismo di risoluzione del conflitto non è distruttivo del messaggio → ogni trasmissione inizia con una fase di arbitraggio (sequenze particolari di bit) che permettono, in caso di conflitto, di riservare il canale al nodo a priorità maggiore
- Metodo di accesso a *divisione di tempo (time sharing)*
 - Ad ogni nodo viene concesso ciclicamente un intervallo temporale in cui può trasmettere le informazioni
 - Metodo che assicura il pieno determinismo
 - Di difficile implementazione → occorre sincronizzare temporalmente tutti i nodi della rete
 - Al nodo è concesso un intervallo anche se non deve trasmettere

reti per l'automazione

Il modello OSI – Metodo di accesso



- Metodo di accesso *produttore/consumatore*
 - È una variante del metodo a condivisione di tempo
 - Uno dei nodi è l'arbitro della rete
 - Vengono identificati i nodi che sono *produttori* di un certo tipo di informazioni e quelli che ne sono *consumatori*
 - L'arbitro della rete abilita la trasmissione di un certo tipo di informazione, abilitando il suo produttore a trasmettere e i suoi consumatori a ricevere
 - La sequenza con cui le informazioni sono abilitate a essere trasferite fissa anche i tempi di aggiornamento dell'informazione → rete deterministica
 - Non c'è necessità di trasmettere gli indirizzi → aumenta l'efficienza della rete
- Esistono metodi di accesso ibridi

reti per l'automazione

Standard per i livelli OSI



- *Rete locale o LAN (Local Area Network)*
 - Quando i primi due livelli del modello OSI sono gli stessi per tutti i nodi connessi
 - Esistono realizzazioni proprietarie e alcuni standard
 - Ethernet
 - Token bus
 - Token ring

reti per l'automazione

Standard per i livelli OSI



– Rete locale Ethernet

- Fu sviluppata alla fine degli anni settanta dalla Xerox e poi implementata da Digital e Intel (standard IEEE 802.3)
- Esistono diversi tipi: *Full Ethernet*
 - Topologia a bus
 - Cavo coassiale schermato
 - Velocità massima di trasmissione 10 Mbit/s
 - Distanza massima di collegamento 500 m per segmento (al più 5 segmenti connessi da ripetitori)
 - Numero massimo di nodi 1024
 - Rete lavora in banda base → un solo canale di trasmissione
 - Non è previsto un segnale di temporizzazione (clock)
 - Metodo di accesso CSMA/CD

reti per l'automazione

Standard per i livelli OSI



- Le informazioni sono codificate in formato Manchester (ogni bit è seguito dal suo negato)
 - » I nodi riceventi si sincronizzano sulla sequenza di bit in arrivo
 - » La cella (bit + suo negato) dura 100 ns → determina la velocità
- Un messaggio completo è chiamato frame ed è composto da un numero di byte variabile tra 72 e 1526
- Un frame contiene vari campi:
 - » Campo 2 (6 byte): rappresentanti l'indirizzo del destinatario o del gruppo di destinatari; viene esaminato da tutti i nodi per capire se il messaggio è diretto a loro
 - » Campo 3 (6 byte): rappresentanti l'indirizzo del mittente
 - » Campo 4 (2 byte): protocollo utilizzato nel campo dati
 - » Campo 5 (46÷1500 byte): contiene i dati
 - » Campo 6 (4 byte): controllo della corretta ricezione

reti per l'automazione

Standard per i livelli OSI



– Rete locale token bus

- Descritta dallo standard IEEE 802.4
- Topologia a bus
- Cavo coassiale o doppino intrecciato
- Velocità di trasmissione variabile da 1 a 10 Mbit/s
- Metodo di accesso a gettone che circola tra i nodi attivi
 - Possono esistere nodi passivi che rispondono ad una chiamata diretta da parte di quelli attivi
- Rete deterministica
 - Massimo ritardo quando un nodo deve aspettare che il gettone circoli in tutti gli altri nodi e vi permanga per tutto il tempo previsto

reti per l'automazione

Standard per i livelli OSI



– Rete locale token ring

- Introdotta dalla IBM
- Descritta dallo standard IEEE 802.5
- Metodo di accesso simile alla token bus, ma è prevista una topologia ad anello che fissa l'ordine di passaggio del gettone
- Velocità massima di trasmissione pari a 4 Mbit/s
- Onerosa da implementare

reti per l'automazione

Standard per i livelli OSI



- Standard per i livelli OSI 3 (livello di rete) e 4 (livello di trasporto)
 - TCP/IP (*Transmission Control Protocol/Internet Protocol*)
 - Insieme di protocolli per la gestione di rete
 - Protocolli sviluppati prima del modello OSI, ma data la loro diffusione (presenti nei sistemi UNIX) sono diventati uno standard *de facto*
 - DECNET
 - Sviluppato da Digital Equipment per i suoi sistemi VAX

reti per l'automazione

Integrazione tra reti



- Integrazione tra reti
 - Ponti (bridge)
 - Connettono le reti al livello 2
 - Consentono la segmentazione del traffico
 - Gestiscono localmente l'indirizzamento
 - Uniscono i vari segmenti in una unica rete
 - Prima di ritrasmettere un messaggio controllano se l'indirizzo appartiene o meno all'altro segmento
 - Instradatori (router)
 - Connettono la rete al livello 3
 - Stabiliscono come un pacchetto di dati debba essere instradato per giungere a destinazione scegliendo eventualmente il migliore tra più percorsi



reti per l'automazione

Integrazione tra reti

– Convertitori (gateway)

- Realizzano l'integrazione ai livelli 5, 6 e 7
- Traducono messaggi tra reti eterogenee

– Ripetitori

- Rigeneratori di segnale necessari per collegamenti molto lunghi

reti per l'automazione

Generalità



- Reti per l'automazione
 - Il modello di riferimento CIM richiede l'uso di reti informatiche per lo scambio di informazioni tra dispositivi dello stesso livello o di livelli differenti
 - Le esigenze di comunicazione sono differenti ai vari livelli:
 - Ai livelli inferiori → necessità di trasferire un notevole numero di informazioni semplici, da trasferire frequentemente entro intervalli di tempo piccoli e certi
 - Ai livelli superiori → esigenza di trasferire informazioni complesse entro intervalli di durata maggiore e non necessariamente determinata

reti per l'automazione

Generalità



- Caratteristiche delle comunicazioni tra i livelli CIM:



reti per l'automazione

Generalità



- **Dimensione** delle informazioni: aumenta dal basso verso l'alto ↑
 - *Esempio*: informazione generata da una fotocellula può essere rappresentata con un solo bit, mentre un piano di produzione può richiedere un megabyte
 - **Frequenza** di trasmissione delle informazioni: aumenta dall'alto verso il basso ↓
 - *Esempio*: un sensore di posizione va interrogato ogni 20 ms mentre un piano di produzione viene trasmesso una volta ogni 6 h
 - **Determinismo** della trasmissione: la necessità che la trasmissione avvenga entro tempi certi aumenta dall'alto verso il basso ↓
- Necessità di utilizzare diverse tipologie di reti informatiche
- Occorrono reti specializzate per il compito a cui sono destinate

reti per l'automazione

Generalità



- Si possono identificare 3 categorie di reti:
 - **Rete per le informazioni**
 - Assicura la comunicazione tra i dispositivi dedicati al controllo e quelli dedicati alla gestione dello stabilimento e dell'azienda (livelli 4, 5 e 6 della piramide CIM)
 - Deve garantire il trattamento di informazioni complesse, composte da molti byte
 - Le frequenze di invio sono basse
 - Non è necessario garantire trasferimenti in tempi certi
 - *Esempio* molto diffuso:
 - » Standard Ethernet per i livelli 1 e 2 dello standard OSI
 - » Uso di bridge per segmentare la rete → vengono uniti su di uno stesso segmento di rete i dispositivi che più necessitano di comunicare tra loro → aumenta l'efficienza della rete

reti per l'automazione

Generalità



- **Rete per il controllo**

- Deve assicurare la comunicazione tra i dispositivi dedicati al controllo e alla supervisione degli impianti (livelli 2, 3 e 4 della piramide CIM)
- Informazioni non molto complesse
- Frequenze di trasmissione più elevate
- Necessità di garantire trasmissioni entro tempi certi
- Diffusione di reti proprietarie
 - » Fornite dagli stessi costruttori dei dispositivi di controllo per integrare i loro dispositivi e i PC dotati di apposite schede
 - » Usualmente i livelli 1 e 2 sono basati su un metodo di accesso a token per assicurare il determinismo
 - » Software per la gestione integrato nel sistema fornito dal costruttore → l'utente deve solo configurare la rete ed utilizzare apposite istruzioni per l'invio e la ricezione dei messaggi

reti per l'automazione

Generalità



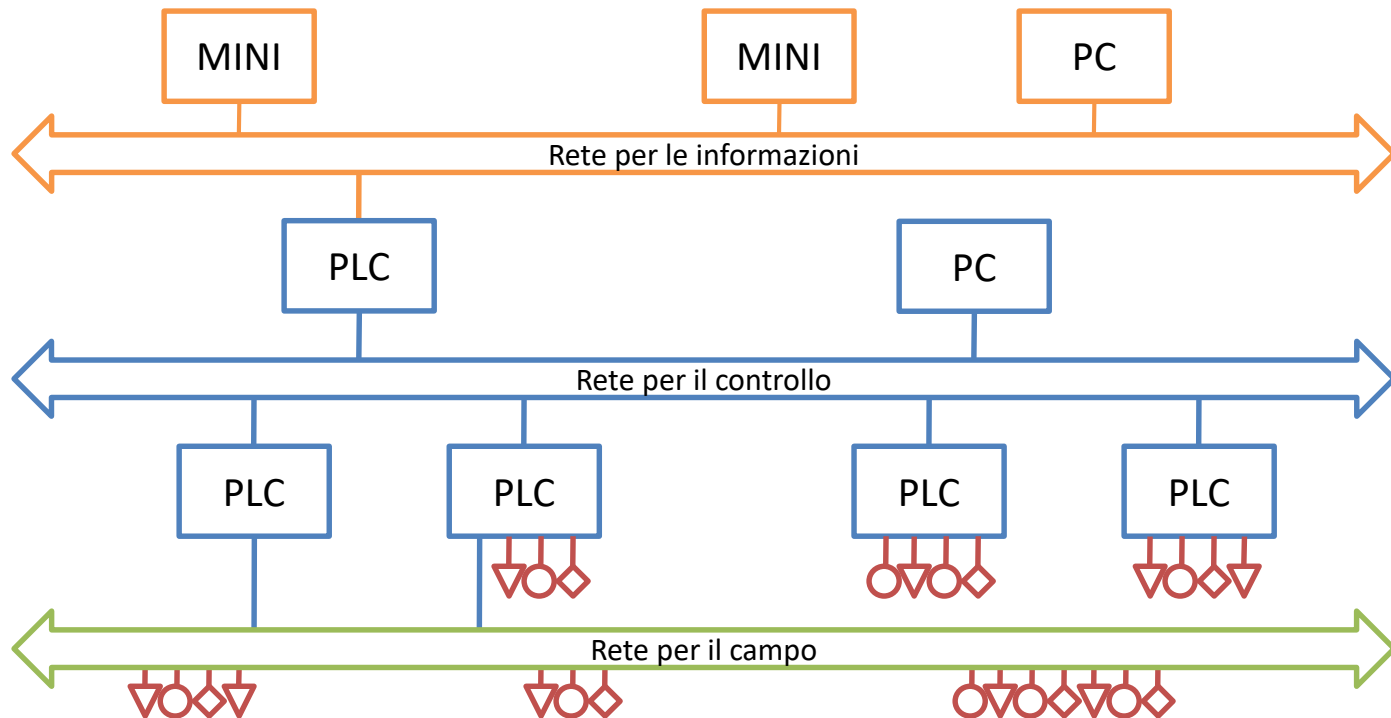
- *Tendenza*: uso di reti Ethernet come rete per il controllo
 - » Nodi connessi tramite switch, eventualmente dotati di funzionalità per la gestione del QoS (*Quality-of-Service*)
 - In ogni segmento è presente solo un dispositivo e lo switch → bassa probabilità di collisioni → lo switch gestisce l'instradamento tra i segmenti che gli fanno capo
- **Rete per il campo** (bus di campo)
 - Se pur di recente introduzione, si stanno diffondendo anche come reti per il controllo
 - Consentono di realizzare le comunicazioni tra dispositivi di controllo e sensori ed attuatori “intelligenti”, ovvero dotati di una interfaccia di rete e di un minimo di capacità elaborativa (livelli 1 e 2 della piramide CIM)

reti per l'automazione

Generalità



- Schema concettuale per una realizzazione di automazione integrata in cui sono evidenziate le tre tipologie di reti per l'automazione:



reti per l'automazione

Protocollo MAP



- Protocollo MAP
 - *Definizione*: protocollo per l'automazione della produzione (*Manufacturing Automation Protocol*)
 - Nato per uniformare e normalizzare il processo di produzione → MAP è un *tentativo di standard* per le reti informatiche per l'automazione che definisce ognuno dei livelli OSI
 - *Esempio*: la parte LAN del protocollo MAP (livelli 1 e 2) prevede una token bus a multifrequenza → consente di utilizzare più canali di comunicazione sullo stesso mezzo di trasmissione

reti per l'automazione

Protocollo MAP



- MMS: specifica per i messaggi di produzione (*Manufacturing Message Specification*)
 - Definita come lo standard per il livello 7 del MAP
 - Consente di poter utilizzare nello stesso stabilimento dispositivi diversi come tipo e/o fornitore
 - Si basa sul paradigma cliente/sergente → il dispositivo cliente richiede al dispositivo servente di eseguire un compito specifico (*esempio*: fornire dati, leggere un sensore, aprire una valvola, ecc.)
 - Il servente è rappresentato da un **VMD**: dispositivo di produzione virtuale (*Virtual Manufacturing Device*)
 - Elevato grado di astrazione rispetto ai dispositivi effettivamente esistenti

reti per l'automazione

Protocollo MAP



- Benefici: portabilità dell'applicazione sviluppata
 - L'MMS, isolata dal MAP, può essere utilizzata come livello 7 per altre reti (infatti è usata in alcuni bus di campo)
- Il tentativo di standardizzazione non ha avuto successo
 - Il protocollo MAP non si è diffuso → ritenuto eccessivamente complesso
 - Tentativi di realizzazione di MiniMAP → non è richiesta la realizzazione di tutti i 7 livelli OSI
 - » Il livello 7 è comunque reso conforme al MAP utilizzando la MMS

reti per l'automazione

Reti di campo



- **Reti di campo (bus di campo)**
 - Sono realizzazioni di reti informatiche per il collegamento tra i dispositivi di controllo e i sensori e gli attuatori presenti sul piano officina o campo
 - I sensori e gli attuatori diventano dei nodi di una rete informatica
→ dovranno essere dotati della capacità di elaborazione necessaria per comunicare mediante la rete
 - Caratteristiche
 - Messaggi brevi
 - Elevata frequenza nello scambio dei messaggi
 - Tempistica assicurata con rigore
 - Possibilità di invio diretto di messaggi a molti nodi contemporaneamente

reti per l'automazione

Reti di campo



- Le reti di campo sono reti “snelle” → sono definiti solo i livelli 1, 2 e 7 del modello OSI
 - Eventuali funzionalità necessarie dei livelli mancanti sono realizzate direttamente nei livelli previsti
- Vantaggi
 - Semplificazione delle architetture di controllo → reti facilmente espandibili e riconfigurabili
 - Riduzione del cablaggio → diminuzione dei costi di installazione e di manutenzione dei cavi
 - Possibilità di trasmettere informazioni più complesse e bidirezionali
 - I dispositivi intelligenti possono operare delle elaborazioni locali del segnale per linearizzarlo e convertirlo in unità ingegneristiche, possono segnalare il proprio stato e chiudere anelli di controllo locali

reti per l'automazione

Reti di campo



- Diminuzione della criticità dei tempi di risposta
 - Molte operazioni demandate direttamente al dispositivo, come la chiusura di anelli di controllo locali
- Possibilità di calibrare tutti i sensori e gli attuatori via software da un solo terminale connesso alla rete
- Maggiore robustezza delle trasmissioni
 - La trasmissione digitale è intrinsecamente meno sensibile ai disturbi di quella analogica e consente di implementare tecniche di riconoscimento e di correzione degli errori di trasmissione
- Ulteriore riduzione del cablaggio
 - Alcune realizzazioni prevedono che l'alimentazione passi sulle stesse linee dei dati
- Scarsa diffusione commerciale
 - Costo ancora elevato, non esiste uno standard unico affermato

reti per l'automazione

Reti di campo



– Classi di reti di campo:

- *Sensorbus* (bus di sensori) → operano al livello del bit
- *Devicebus* (bus di dispositivi) → operano al livello del byte
- *Fieldbus* (bus di campo) → operano al livello dei blocchi di byte

– Bus di sensori

- Realizzano solo i livelli OSI 1 e 2 (fisico e link dati)
- Lo scopo primario è la riduzione del cablaggio
- Lunghezza tipica dei messaggi trasmessi inferiore al byte
- Dispositivi già esistenti connessi al bus mediante dei multiplexer
 - Non è necessari utilizzare dispositivi intelligenti realizzati ad hoc
- Bus di sensori più diffusi: ASI e Seriplex

reti per l'automazione

Reti di campo



– Bus di dispositivi

- Consentono il trasferimento di messaggi di lunghezza fino a 16÷32 byte
- Realizzano le funzionalità dei livelli OSI 1 e 2 ed alcune del 7
- Possono esservi connessi dispositivi tradizionali mediante adattatori o dispositivi intelligenti basati o no su microprocessore
- Consentono la comunicazione di una semplice diagnostica
- Bus di dispositivi più diffusi: Interbus-S, DeviceNet, SDS

– Bus di campo

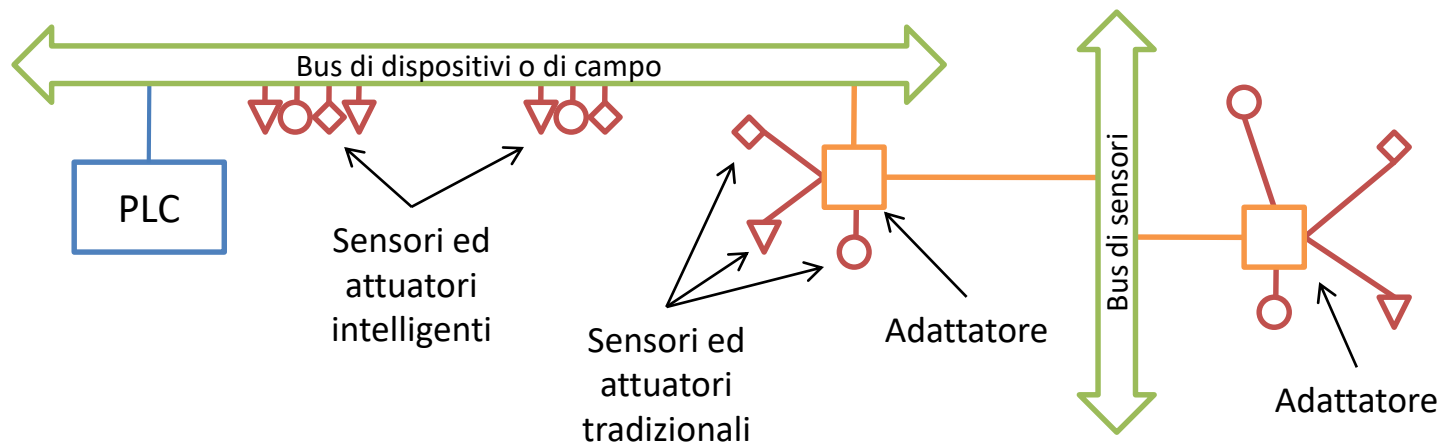
- Consentono la comunicazione di blocchi fino al migliaio di byte
- Realizzano le funzionalità dei livelli OSI 1, 2, 7 e di un livello aggiuntivo 8 (*livello utente*)

reti per l'automazione

Reti di campo



- I dispositivi sviluppati per la connessione ad un bus di campo includono algoritmi predefiniti che vengono configurati via rete per adattarsi alla specifica applicazione
- Bus di dispositivi più diffusi: Profibus, Foundation Fieldbus
 - È in corso un tentativo di unificare questi due standard
- Un impianto di controllo distribuito può impiegare tutte le categorie di reti di campo presentate, singolarmente o collegate tra di loro



reti per l'automazione

Reti di campo: ASI



– ASI (*Actuator Sensor Interface*)

- Si tratta di un bus di sensori
- Può connettere fino a 31 nodi
 - Ogni nodo può connettere fino a 4 dispositivi di ingresso/uscita binari → totale fino a 124 dispositivi
- Protocollo di comunicazione è basato sul chip ASI
 - Deve essere contenuto in ogni dispositivo connesso oppure in ogni nodo usato per connettere dispositivi tradizionali
- Protocollo di tipo master/slave a sequenza ciclica
 - Tutti gli slave vengono interrogati in successione e ricevono 4 bit di dati, utilizzati per il controllo delle uscite, e rispondono con 4 bit, utilizzati per rappresentare lo stato degli ingressi
 - Tempo di ciclo massimo inferiore a 5 ms



reti per l'automazione

Reti di campo: ASI

- Velocità di trasferimento di 167 Kbit/s
- Lunghezza massima 100 m
- Dispositivi che necessitano di più di 4 bit per comunicare lo fanno in cicli successivi, senza rallentare il tempo di ciclo
- Alimentazione a 24 V in corrente continua
 - Distribuita attraverso due conduttori non schermati e non intrecciati
 - Gli stessi conduttori trasportano anche l'informazione

reti per l'automazione

Reti di campo: Suriplex



– Suriplex

- Si tratta di un bus di sensori
- Può connettere fino a 512 dispositivi
- Configurazione master/slave o peer-to-peer
- Ogni dispositivo deve contenere un chip ASIC
 - Sono disponibili 32 funzioni logiche utilizzate per realizzare il protocollo di comunicazione
 - Dispositivi tradizionali possono essere collegati mediante degli adattatori che contengono il chip ASIC
- Cavi di alimentazione e di comunicazione separati
- Lunghezza dei collegamenti fino a 1500 m

reti per l'automazione

Reti di campo: InterBus-S



– InterBus-S

- Si tratta di un bus di dispositivi
- Può connettere fino a 256 dispositivi
 - Fino a 4096 ingressi digitali e 4096 uscite digitali o combinazioni di ingressi e uscite digitali e analogiche
- Velocità di trasmissione fino a 500 Kbit/s
- Lunghezza dei segmenti fino a 400 m
 - Massimo 32 segmenti
- Previsti diversi mezzi trasmissivi
 - Doppino, fibra ottica, trasmissione ad infrarossi
- Topologia ad anello
- Protocollo master/slave con verifica della correttezza dei messaggi con la tecnica CRC

reti per l'automazione

Reti di campo: Bus CAN



– DeviceNet e SDS

- Sono bus di dispositivi basati sul bus CAN
 - Il bus CAN fu originariamente sviluppato per collegare i dispositivi elettronici a bordo delle autovetture
 - Si tratta di un protocollo aperto
 - Messaggi di lunghezza variabile (fino a 8 byte)
 - Metodo di accesso di tipo CSMA/BA con priorità non distruttivo
 - » Messaggi mai perduti a causa di collisioni
 - Previsto un avanzato sistema per il controllo degli errori di trasmissione
 - Previsto l'impiego di 4 conduttori intrecciati e schermati
 - » Due per l'alimentazione e due per i dati



reti per l'automazione

Reti di campo: Bus CAN

- DeviceNet
 - Può gestire fino a 64 nodi
 - » Totale di 2048 punti di ingresso/uscita digitali
 - Velocità massima di 500 Kbit/s
- SDS
 - Può gestire fino a 64 nodi
 - » Possono diventare 126 utilizzando delle speciali tecniche di multiplexing degli indirizzi
 - Velocità massima di trasmissione fino a 1 Mbit/s

reti per l'automazione

Reti di campo: Foundation Fieldbus



– Foundation Fieldbus

- Basato sul modello OSI, di cui sono definiti i livelli 1, 2 e 7
- Viene definito un livello ulteriore: *livello utente*
 - Funzioni importanti, come i blocchi funzionali, servizi di descrizione dei dispositivi e servizi per la gestione della rete
- Livello 1 (livello fisico)
 - Utilizzo del bus H1
 - » Impiego di un bus a bassa velocità (H1), limitata a 31.25 Kbit/s
 - » Lunghezza massima di 1900 m
 - » Possibilità di collegare fino a 32 dispositivi non alimentati, 12 dispositivi alimentati direttamente dal bus, 6 dispositivi alimentati in modalità sicura (a bassa energia) per ambienti pericolosi

reti per l'automazione

Reti di campo: Foundation Fieldbus



- Utilizzo del bus HSE (*High Speed Ethernet*)
 - » Utilizzo di un bus ad alta velocità (HSE), basato sullo standard Ethernet
 - » Velocità fino a 2.5 Mbit/s
 - » Lunghezza massima di 750 m
 - » Fino a 127 dispositivi collegabili
- Le due soluzioni possono convivere attraverso l'utilizzo di un bridge
- Livello 2 (collegamento dati)
 - Il bus viene gestito da un dispositivo denominato *pianificatore attivo*
 - » Metodo di accesso deterministico e centralizzato che prevede due modalità di comunicazione: ciclica e aciclica
 - » Modalità ciclica: il pianificatore attivo controlla lo stato della rete e concede agli altri dispositivi, a istanti prefissati, il permesso di inviare messaggi

reti per l'automazione

Reti di campo: Foundation Fieldbus



- » Gli altri dispositivi possono ascoltare la rete e ricevere i messaggi a loro diretti (produttore/consumatore)
- » Una porzione di tempo viene riservata per comunicazioni acicliche sollecitate dal dispositivo pianificatore attraverso l'invio di un token specifico
- Livello 7 (livello applicazione)
 - Contiene la FMS (*Fieldbus Messaging Specification*)
 - » Derivazione dell'MMS del MAP
 - Codifica e decodifica i comandi del livello addizionale utente
 - Contiene il dizionario degli oggetti: consente ai dati di essere referenziati sia per la loro etichetta sia per la loro chiave identificativa

reti per l'automazione

Reti di campo: Foundation Fieldbus



- Livello addizionale 8 (livello utente)
 - Implementa la strategia di controllo distribuito del Fieldbus
 - Definisce il *modello software* con cui l'utente della rete interagisce
 - Sono previsti dei **blocchi funzionali** che sono delle funzioni predefinite per realizzare varie operazioni
 - » Operazioni di lettura e scrittura di dati
 - » Controllo del dispositivo
 - » Manipolazione dell'informazione
 - » Funzioni aritmetiche
 - L'utente può scrivere blocchi funzionali *ad hoc* per particolari applicazioni
 - I blocchi funzionali possono anche essere implementati nei dispositivi
 - » *Esempio*: valvola con blocco funzionale che realizza un PID

reti per l'automazione

Reti di campo: Foundation Fieldbus



- Esistono servizi di descrizione dei dispositivi che consentono ad un nodo di ricevere informazioni da altri nodi, come il nome del costruttore, le sue capacità diagnostiche, ecc.
- Sono compresi servizi per la gestione della rete
 - » Assegnazione degli indirizzi
 - » Pianificazione dell'esecuzione dei blocchi funzionali

reti per l'automazione

Reti di campo: Profibus



– Profibus

- È un bus di campo che consente sia il collegamento di un controllore master con dispositivi slave, sia comunicazioni peer-to-peer
- Si tratta di un insieme di reti, pensate per applicazioni diverse, con protocolli compatibili tra loro
- Livello 1 (livello fisico)
 - Previste diverse implementazioni
 - » RS485, RS485-IS, MBP e fibra ottica
 - Gestibili fino a 126 dispositivi
 - Velocità fino a 12 Mbit/s

reti per l'automazione

Reti di campo: Profibus



- Livello 2 (collegamento dati)
 - Prevede varie implementazione:
 - » DP-V0: metodo di accesso centralizzato in cui il master interroga ciclicamente i vari slave
 - » DP-V1: come il V0 con in più un metodo di accesso a token tra più master
 - » DP-V2: aggiunge al V1 un metodo di accesso a divisione di tempo del tipo produttore/consumatore tra gli slave
- Livello 7 (livello applicazione)
 - Implementa la specifica FMS
 - Offre vari servizi per la comunicazione dei dati
- Sopra il livello 7: *profili*
 - Specifiche definite dai costruttori o dagli utenti che riguardano proprietà, caratteristiche e comportamento dei dispositivi connessi

reti per l'automazione

Reti di campo: Profibus



- A seconda della combinazione degli elementi scelta nella rete si parlerà di diverse implementazioni del Profibus, tra cui:
 - Profibus-DP
 - » Orientato all'automazione industriale
 - » Livello 1: RS-485
 - » Livello 2: uno dei protocolli DP
 - » Profili: uno o più profili tipici dell'automazione industriale
 - Profibus-PA
 - » Orientato al processo
 - » Livello 1: MBP
 - » Livello 2: DP-V1
 - » Profilo specializzato per il controllo di processo

reti per l'automazione

Reti di campo: Profibus



- Motion Control with Profibus
 - » Orientando al collegamento di controllori per motori elettrici
 - » Livello 1: RS-485
 - » Livello 2: DP-V2
 - » Profilo specializzato (PROFIdrive)

sistemi per il controllo di supervisione e l'acquisizione dati

Generalità



- **Sistemi di supervisione e acquisizione dati**

- *Esempio*: circuito con interruttore e lampadina

- Indica la presenza di un oggetto, lo stato di accensione di un motore, l'occorrenza di un malfunzionamento,...
 - Se l'operatore non è vicino all'interruttore è necessario utilizzare un dispositivo di comunicazione per trasportare l'informazione
 - Se gli interruttori sono migliaia è necessario organizzare l'informazione per poterla inviare con un solo dispositivo di comunicazione (*esempio*: serializzando i dati)
 - L'operatore non può supervisionare lo stato di migliaia di interruttori simultaneamente → con l'uso di un PC si possono organizzare e storicizzare queste informazioni e visualizzarle in modo organizzato (quadri sinottici, serie storiche)

sistemi per il controllo di supervisione e l'acquisizione dati

Generalità

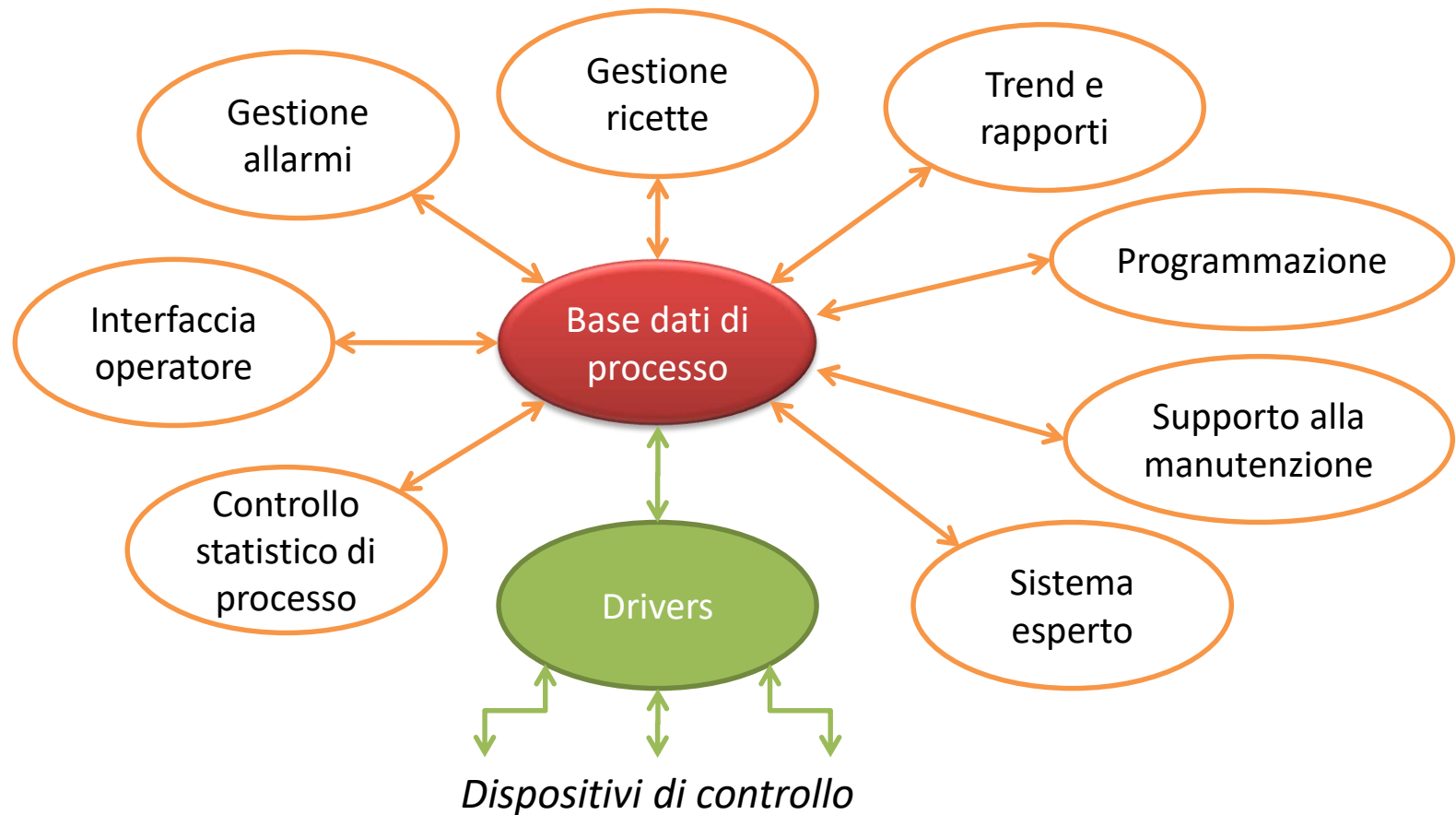
- L'operatore dovrebbe poter intraprendere delle azioni in risposta alle informazioni che riceve inviando comandi → la comunicazione deve essere bidirezionale
- L'uso di un PC può consentire di intraprendere delle azioni automatiche all'occorrenza di determinati eventi e lasciare all'operatore solo interventi in caso di eventi imprevisti o anomali

→ Sistema di supervisione e acquisizione dati (**SCADA**:
Supervisory Control And Data Acquisition)

- Sistema a *struttura modulare* con tecnologia client/server
- Nucleo della struttura è il modulo **base di processo** a cui tutti gli altri moduli fanno riferimento
- Molti degli altri moduli che possono comporre un sistema SCADA non sono necessariamente tutti presenti

sistemi per il controllo di supervisione e l'acquisizione dati

Architettura software modulare



sistemi per il controllo di supervisione e l'acquisizione dati

Base dati di processo

– Base dati di processo

- È il **nucleo** di ogni sistema di supervisione e acquisizione dati
- Necessario l'accentramento della gestione del flusso di informazioni anche per sistemi di controllo di piccole/medie dimensioni → centinaia/migliaia di punti di interfaccia di processo → metodo sistematico per processare i dati
- Insieme minimo di campi per un **record** della base dati si può così identificare:
 - Codice della variabile (TAG)
 - Classe di allarme
 - Tempo di aggiornamento
 - Valore grezzo
 - Valore convertito
 - Stato di allarme

sistemi per il controllo di supervisione e l'acquisizione dati

Base dati di processo



- Per le variabili analogiche vanno inoltre considerati:
 - Unità di misura
 - Intervallo ammissibile di variazione
- Per le variabili binarie
 - Stato esplicitato in testo
 - Valore di allarme
- *Esempio:* record per una variabile analogica di temperatura

Codice:	T25
Descrizione:	temperatura vasca
Tipo:	IAN
Indirizzo:	#2:35/36
Unità:	gradi Celsius
Val. minimo:	45.0
Val. massimo:	75.0
Campion. (s):	20

sistemi per il controllo di supervisione e l'acquisizione dati

Base dati di processo



Classe allarme: 2
Valore grezzo: 355.5
Valore conver.: 78.1
Stato allarme: 1

- Valori statici (vengono definiti all'atto della configurazione del DB):
 - » Il *codice* serve per identificare univocamente il record
 - » La *descrizione* serve per chiarire la natura delle variabili
 - » Il *tipo* mostra se la variabile è un ingresso oppure un'uscita e se essa è analogica o binaria (IAN: Ingresso ANalogico)
 - » L'*indirizzo* identifica l'apparecchiatura dalla quale la variabile viene prelevata (#2:35/36: unità 2, modulo 35, punto 36)
 - » Il campo *unità* riporta l'unità di misura
 - » Sono definiti i *valori minimo* e *massimo* che la variabile può assumere

sistemi per il controllo di supervisione e l'acquisizione dati

Base dati di processo



- » Il *tempo di campionamento* indica ogni quanto tempo la variabile viene aggiornata
- » La *classe di allarme* identifica la priorità dell'allarme e l'eventuale procedura automatica da eseguire se il valore della variabile supera i limiti
- Valori dinamici (aggiornato con la frequenza indicata dal tempo di campionamento):
 - » Il *valore grezzo* è quello comunicato dall'unità e può dover essere convertito in una certa unità di misura applicando una relazione matematica (scalatura, offset, linearizzazione, ...) per ottenere il *valore convertito*
 - » Lo *stato di allarme* indica se la variabile viola o meno i limiti

sistemi per il controllo di supervisione e l'acquisizione dati

Base dati di processo

- *Esempio:* record per un ingresso binario

Codice:	K010
Descrizione:	pompa scarico
Tipo:	IBI
Indirizzo:	#7:12/01
Stato 0:	ON
Stato 1:	OFF
Stato allarme:	OFF
Classe allarme:	3
Campion. (s):	2
Valore grezzo:	0
Val. conver.:	ON
Stato allarme:	0

sistemi per il controllo di supervisione e l'acquisizione dati

Base dati di processo



- La base di processo consente di **uniformare i dati** che contiene
 - Sono separati concettualmente i dati memorizzati da quelli disponibili, che possono essere di diverso tipo
- Gli altri moduli software accedono alla base dati senza conoscere dettagli su come la variabile è misurata e convertita
 - Si deve solo conoscere le modalità di rappresentazione interne alla base dati
- Se la procedura per l'acquisizione di una variabile cambia, le modifiche da apportare sono minime e interessano solo la base dati (*esempio*: sostituzione di un sensore con uno diverso)
- È possibile definire delle **variabili derivate** da quelle misurate
 - Possibile ottenere informazioni su dati come l'energia consumata o l'efficienza, che non possono essere misurate direttamente o sono troppo costose da misurare

sistemi per il controllo di supervisione e l'acquisizione dati

Base dati di processo

- Le variabili derivate sono aggiornate automaticamente dalla base dati di processo quando varia una delle variabili da cui dipendono
- La base dati deve essere aggiornata velocemente → viene contenuta nella *memoria centrale* e non su memoria di massa
 - Le sue **dimensioni sono fissate** all'atto della configurazione → l'arrivo di una nuova variabile cancella quello precedente
 - **La base dati di uno SCADA non è un archivio!**

sistemi per il controllo di supervisione e l'acquisizione dati

Modulo di comunicazione



– Modulo di comunicazione (*driver*)

- Si occupa della gestione delle comunicazioni con i vari dispositivi e del controllo del trasporto delle informazioni
 - *Esempi*: comunicazione via linee seriale, rete informatica
- Si occupa anche dell'interpretazione dei messaggi e del legame bidirezionale tra base dati del processo e dispositivo
- Almeno un modulo driver deve sempre essere presente
 - Consente allo SCADA di accedere in maniera contemporanea ai diversi dispositivi che sono connessi con il campo (RTU: *Remote Termial Unit*)
- Il suo funzionamento dipende dal tipo di comunicazione accettata dal dispositivi sottostante
 - Sono sviluppati driver appositi per connettere dispositivi di vari fornitori per i quali sono resi disponibili le modalità di comunicazione

sistemi per il controllo di supervisione e l'acquisizione dati

Modulo di comunicazione



- Sono forniti i moduli per la comunicazione con i dispositivi e le reti più diffuse
 - » Alcuni aziende indipendenti forniscono driver per dispositivi meno diffusi
- Il modulo driver consente anche la comunicazione con altri applicativi presenti sullo stesso elaboratore
 - » *Esempio*: in ambiente Windows può essere fornito di un driver DDE (*Dynamic Data Exchange*) per collegare fogli elettronici o basi dati tradizionali

sistemi per il controllo di supervisione e l'acquisizione dati

Interfaccia uomo-macchina



– Interfaccia uomo-macchina

- La presenza di operatori umani è necessaria anche in impianti con un elevato tasso di automazione
 - sono indispensabili sistemi di comunicazione uomo-macchina
- Il modulo di interfaccia operatore deve consentire:
 - *L'accesso alla base dati*
 - » Con tecniche di selezione delle variabili, di selezione dei campi, di ordinamento su campi specifici, di interrogazione, ...
 - *Una comunicazione efficiente con l'operatore*
 - » Possibilità di impostare pagine grafiche che rappresentano pannelli di comando e quadri sinottici dell'intero impianto o parti di esso (eventualmente organizzate in modo gerarchico)

sistemi per il controllo di supervisione e l'acquisizione dati

Interfaccia uomo-macchina

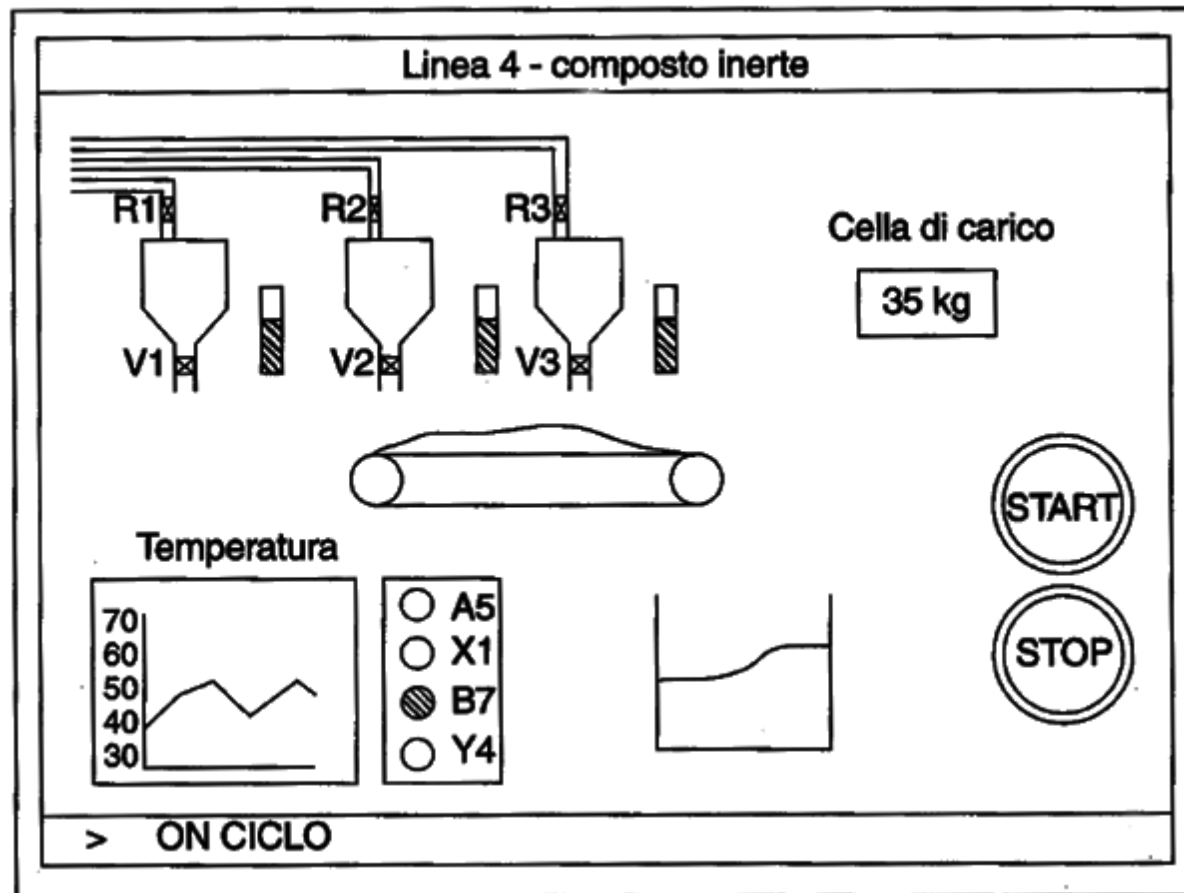


- Il **pannelli di comando** permettono all'operatore di inviare comandi all'impianto attraverso simboli grafici che rappresentano pulsanti, manopole, comandi a slitta, ecc.
- Nei **quadri sinottici** sono presenti simboli grafici caratterizzati da
 - *Elementi statici*: attributi grafici che non variano nel tempo (aspetto, colore, dimensione, ecc.)
 - *Elementi dinamici*: attributi grafici che variano a seconda del valore di certe variabili della base dati a cui sono associati
- I quadri sinottici sono organizzati e collegati tra di loro
 - L'operatore può passare con facilità dalla supervisione di una porzione di impianto all'altra
- È fondamentale la **coerenza** dei sinottici
 - Uso delle stesse convenzioni per rappresentare fenomeni simili

sistemi per il controllo di supervisione e l'acquisizione dati

Interfaccia uomo-macchina

- *Esempio:* quadro sinottico di un sottosistema di un impianto



sistemi per il controllo di supervisione e l'acquisizione dati

Interfaccia uomo-macchina



- I sistemi SCADA mettono a disposizione dell'operatore **una libreria grafica** di oggetti per la composizione dei quadri sinottici
- I quadri sinottici di ultima generazione sono di tipo *multimediale*
 - Presenza di immagini, registrate o in diretta, dell'impianto a cui sono sovrapposte le informazioni
 - Comunicazione attraverso sintetizzatori vocali
 - Riconoscimento di comandi vocali

sistemi per il controllo di supervisione e l'acquisizione dati

Gestione allarmi



– Gestione allarmi

- Gli allarmi gestiti da sistemi SCADA sono classificabili come segue:

– Allarmi a insorgenza

- » Vengono attivati dal verificarsi dell'evento a essi associato e poi permangono nel loro stato di attivazione fino ad una reinizializzazione del sistema

– Allarmi a insorgenza e riconoscimento

- » Dopo la loro attivazione possono essere disattivati quando l'operatore ne effettua il cosiddetto *riconoscimento*, comandandone esplicitamente la disattivazione

– Allarmi a insorgenza, riconoscimento e rientro

- » La disattivazione è possibile se l'operatore effettua il riconoscimento e la condizione di attivazione viene meno

sistemi per il controllo di supervisione e l'acquisizione dati

Gestione allarmi



- Il sistema permette di programmare la condizione che genera l'allarme e il tipo di allarme
 - Gli allarmi possono essere raggruppati in differenti **classi di priorità**
 - Nel caso di attivazione simultanea di più allarmi, i livelli di priorità definiscono quali debbano essere rilevati e serviti prima
- Ad ogni allarme o gruppo di allarmi con la stessa priorità è possibile associare una **sequenza di operazioni prefissate** che il sistema SCADA eseguirà automaticamente qualora l'allarme venga attivato
- L'insorgenza di un allarme viene comunicata all'operatore tramite l'interfaccia e memorizzata in maniera non volatile su supporto magnetico (**log degli allarmi**) per le analisi a posteriori degli eventi
 - Sono analizzati il numero, la frequenza e il tipo degli allarmi

sistemi per il controllo di supervisione e l'acquisizione dati

Gestione allarmi



- Gli allarmi possono essere **filtrati**
 - Riduzione degli allarmi a quelli effettivamente significativi a seconda delle condizioni operative
 - Viene realizzata definendo delle condizioni logiche che inibiscono il rilevamento di particolari condizioni di allarme
 - » *Esempio*: quando una valvola è chiusa non deve generare un allarme la condizione di assenza di flusso a valle della valvola

sistemi per il controllo di supervisione e l'acquisizione dati

Gestione ricette



– Gestione ricette (o *lotti* o *batch*)

- Gestisce delle sequenze di operazioni predefinite (*ricette*)
- Le ricette vanno fatte eseguire in conformità a *scadenze temporali* o al *verificarsi di eventi particolari* o alla *richiesta dell'operatore*
- L'uso delle ricette serve per impostare:
 - In un ambiente *multiprocesso*: i dati di impianto per ciascun processo
 - In un ambiente *multiprodotto*: i dati relativi ad ogni prodotto
- I dati impostati mediante ricette rappresentano i parametri da fornire alle macchine che eseguono effettivamente le varie fasi del processo o dei processi di lavorazione

sistemi per il controllo di supervisione e l'acquisizione dati

Tendenze e rapporti

– Tendenze e rapporti

- Il modulo tendenze e rapporti consente di tenere traccia dei **ritmi di produzione** sia per il passato (*serie storiche*) sia per il presente (*rappresentazioni grafica in tempo reale*)
- Le **serie storiche** sono usate per ricercare le cause di eventuali anomalie di funzionamento o può essere imposta per legge
- La **rappresentazione grafica in tempo reale** delle tendenze (*trend*) delle variabili consente di controllare l'evoluzione del processo
- Il modulo consente di selezionare le variabili da tracciare e gli intervalli di tempo di acquisizione
- I **rapporti** (*report*) sullo stato dell'impianto possono essere generati (in forma cartacea e/o informatica) in corrispondenza di scadenze temporali impostate o al verificarsi di un particolare evento (allarme o richiesta dell'operatore)

sistemi per il controllo di supervisione e l'acquisizione dati

Supporto alla manutenzione



– Supporto alla manutenzione

- Le azioni di manutenzione possono essere divise in due categorie:
 - *Manutenzione correttiva o straordinaria*
 - » Eseguita per riparare dispositivi che si guastano
 - *Manutenzione preventiva o programmata*
 - » Eseguita per mantenere l'impianto in condizioni ottimali di funzionamento
- **Manutenzione correttiva**
 - Consente all'operatore di individuare l'area dell'impianto e le possibili cause di malfunzionamento
 - Tiene traccia della frequenza e la tipologia di guasto di tutte le operazioni di manutenzione effettuata per analisi successiva

sistemi per il controllo di supervisione e l'acquisizione dati

Supporto alla manutenzione



- **Manutenzione preventiva**

- Molte operazioni di manutenzione devono essere eseguite con cadenze temporali fissate o dopo un certo numero di ore di funzionamento effettive
- Altre devono essere eseguite se si verifica un certo evento
- I sistemi SCADA forniscono in modo automatico i *piani di manutenzione* giornaliera da fornire alle squadre di manutentori
 - » I dispositivi da gestire possono essere migliaia → necessario un sistema automatizzato per gestire e coordinare la manutenzione

sistemi per il controllo di supervisione e l'acquisizione dati

Sistema esperto

– Sistema esperto

- È delegato alla decisione sulle modalità di comportamento del sistema in risposta a determinati eventi, nel caso in cui queste non siano state esplicitamente programmate
- È un modulo che deve poter “sostituire” l'operatore umano, che prende le sue decisioni sulla base dello stato dell'impianto e della sua esperienza
- Spesso il sistema esperto fornisce solo una serie di **indicazioni operative** all'operatore, al quale spetta la decisione finale
- Si tratta di una applicazione delle tecniche di *intelligenza artificiale*
 - Il funzionamento è basato su un insieme di regole codificate (base di conoscenza) e su un motore inferenziale in grado di inferire da tali regole il comportamento da tenere in situazioni non previste

sistemi per il controllo di supervisione e l'acquisizione dati

Controllo statistico



– Controllo statistico

- I principi alla base del controllo statistico di processo (CSP) furono stabiliti negli anni venti da Walter Shewhart
 - Oggi costituiscono la base del concetto di *qualità totale* nella produzione
 - Shewhart noto che ogni tipo di processo genera delle uscite i cui valori misurati presentano delle variazioni casuali intorno ad un valore medio
 - » La casualità è inevitabile e generata dallo stesso processo di misura e da parametri non controllabili
 - Una misura di tale casualità è data dalla **curva di distribuzione delle frequenze**
 - » Se invece di considerare i singoli campioni di misura si considerano le **medie di gruppi di campioni** esse si distribuiscono secondo una *distribuzione normale*

sistemi per il controllo di supervisione e l'acquisizione dati

Controllo statistico



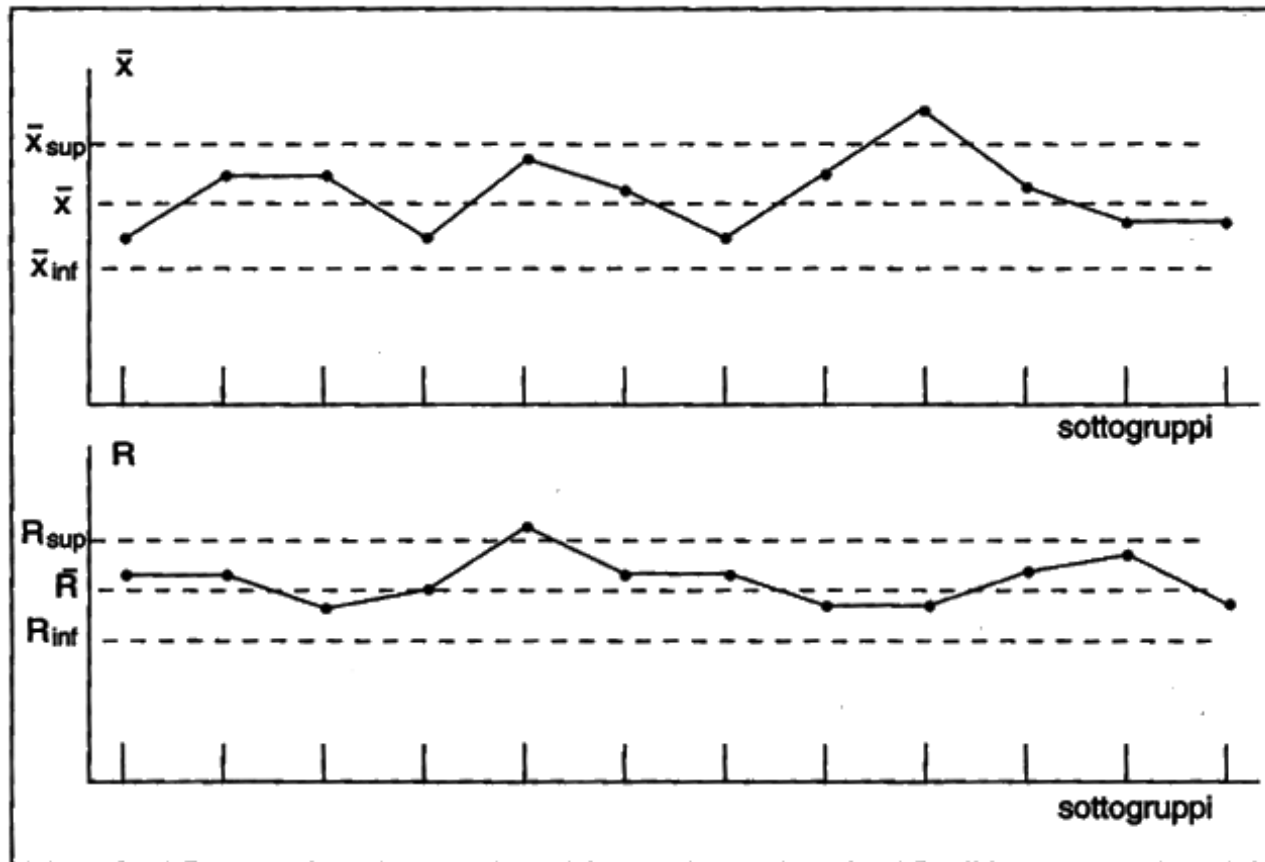
- Si tratta di una conseguenza del *teorema del limite centrale*
- Una distribuzione normale (forma a campana) è caratterizzata dalla sua *media* e dalla sua *varianza*
 - Se un processo è sotto controllo statistico, le medie e le varianze delle distribuzioni non devono cambiare
 - Se cambiano il processo è affetto da cause di **errore deterministico** (non casuale) che vanno identificate e corrette o compensate
- Il modulo statistico costruisce in tempo reale i *diagrammi di controllo* che rappresentano i dati statistici del processo rispetto al tempo in una forma adatta a verificare se il controllo statistico è mantenuto o meno
 - *Diagrammi delle medie*: valori medi di sottogruppi di misure
 - *Diagrammi degli intervalli*: intervalli di variazione nel sottogruppo
 - Dimensioni del sottogruppo tra 2 e 20 misure (5 è un valore diffuso)

sistemi per il controllo di supervisione e l'acquisizione dati

Controllo statistico



- *Esempio*: diagrammi delle medie e degli intervalli



sistemi per il controllo di supervisione e l'acquisizione dati

Controllo statistico



- Sul diagramma vengono indicati i valori medi delle medie e degli intervalli di variazione
- Sono rappresentati i **limiti di controllo** inferiore e superiore
 - » Si discostano dal valore medio di una quantità pari a tre volte la varianza stimata
 - » Esistono formule empiriche per il calcolo dei limiti:

$$\left\{ \begin{array}{l} \bar{x}_{\text{inf}} = \bar{\bar{x}} - A_2 \bar{R} \\ \bar{x}_{\text{sup}} = \bar{\bar{x}} + A_2 \bar{R} \\ R_{\text{inf}} = D_3 \bar{R} \\ R_{\text{sup}} = D_4 \bar{R} \end{array} \right.$$

dove le costanti sono determinate in funzione della dimensione del sottogruppo (per sottogruppi di 5 campioni: $A_2 = 0.577$, $D_3 = 0.0$, $D_4 = 2.114$)

sistemi per il controllo di supervisione e l'acquisizione dati

Controllo statistico



- Normalmente i punti devono essere distribuiti secondo una curva di distribuzione normale → il 97,73% dei campioni deve rientrare nei limiti di controllo
- L'indicazione che il processo potrebbe non essere sotto controllo statistico può essere data da alcune anomalie rilevabili dal diagramma (caratterizzate da una bassissima probabilità):
 1. La presenza di uno o più punti al di fuori dei limiti di controllo
 2. Almeno 9 punti successivi tutti sopra o sotto il valore medio
 3. Una sequenza di punti che si ripete sempre
 4. Almeno 7 punti successivi che danno una linea di tendenza di crescita o decrescenza
- Processo non sotto controllo statistico → esistono cause della variabilità dei campioni che non sono casuali → le cause si devono identificare ed eliminare o correggere
- Correlazioni tra variabili → ausilio all'identificazione delle cause

sistemi per il controllo di supervisione e l'acquisizione dati

Linguaggi di programmazione



– Linguaggi di programmazione

- I sistemi SCADA prevedono linguaggi di programmazione dedicati
- Esistono due classi di linguaggi:
 - *Linguaggi di tipo procedurale*
 - » Prevedono l'esplicitazione di tutte le istruzioni
 - » Sono vicini ai linguaggi dell'informatica classica
 - » Sono più complessi
 - » Offrono una maggiore flessibilità nella programmazione
 - *Linguaggi di tipo descrittivo*
 - » Permettono di trattare agevolmente i dati e le relazioni tra di essi
 - » Sono rivolti all'utente non necessariamente esperto di informatica
 - » Sono linguaggi più rigidi e limitati

sistemi per il controllo di supervisione e l'acquisizione dati

Architettura hardware

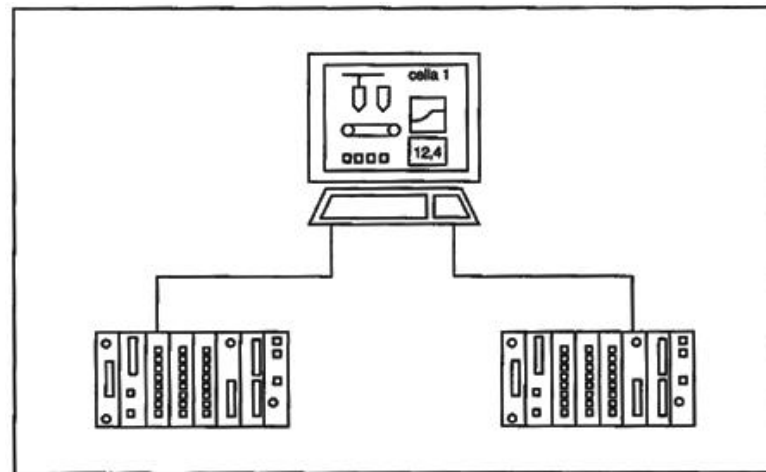
– Architettura hardware

- I sistemi SCADA di grande scala sono stati storicamente progettati nel dominio dei sistemi UNIX e VMS → uso di workstation adatte
- Recentemente si è passati rapidamente da sistemi monolitici (proprietary) a sistemi modulari e flessibili
 - Determinato dalla diffusione dei PC anche per il controllo in tempo reale
 - PC hanno raggiunto prestazioni notevoli, con architetture a 32 e 64 bit, affidabilità elevata e sono dotati di sistemi multitasking pre-emptive e di comunicazione interprocesso
 - Esistono diverse versioni di sistemi SCADA dedicati a sistemi operativi standard (come Windows) → riduzione dei costi di installazione e di sviluppo

sistemi per il controllo di supervisione e l'acquisizione dati

Architettura hardware

- Topologia dei sistemi SCADA
 - Sistemi *a connessione diretta*
 - » L'host su cui opera lo SCADA è connesso direttamente alla RTU o alle RTU tramite collegamenti dedicati (tipicamente seriali)
 - » Sono utilizzati per supervisionare piccole installazioni con poche RTU

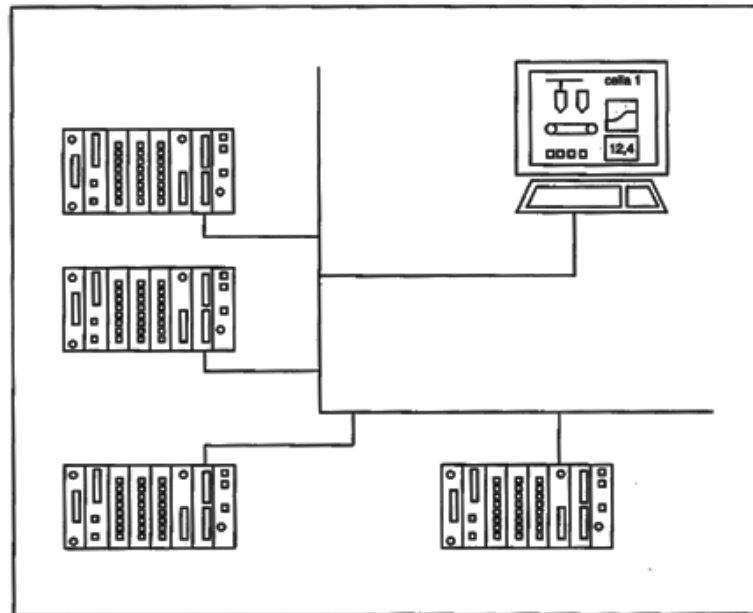


sistemi per il controllo di supervisione e l'acquisizione dati

Architettura hardware



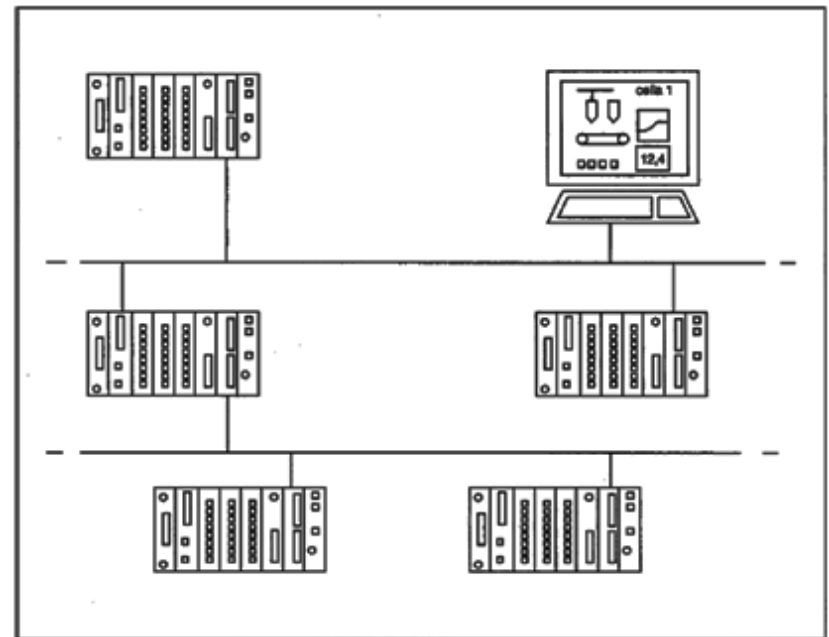
- Sistemi *connessi via rete*
 - » L'host su cui opera lo SCADA è connesso alla stessa rete (proprietaria) cui sono connesse le varie RTU
 - » Topologia adatta per medie installazioni



sistemi per il controllo di supervisione e l'acquisizione dati

Architettura hardware

- Sistemi *connessi via rete a più livelli*
 - » L'host su cui opera lo SCADA è connesso alle varie RTU attraverso vari livelli di rete *gerarchicamente organizzati*
 - » Le reti possono anche essere diverse come tipologia
 - » Sono adatti per grosse installazioni
 - necessità di monitorare parti del processo controllate da RTU di tipo diverso e di utilizzare sistemi specifici per l'acquisizione dei dati



sistemi per il controllo di supervisione e l'acquisizione dati

Architettura hardware

- I sistemi presentati possono comprendere sistemi su larga scala geografica
 - Parte delle RTU sono collegate al sistema SCADA attraverso modem o segnali in radiofrequenza
 - » *Esempio*: monitoraggio di dighe e bacini per la produzione di energia idroelettrica

sistemi per l'esecuzione della produzione (MES)

Sistemi MES



- **Sistemi per l'esecuzione della produzione (MES)**

- Livello CIM per la gestione dello stabilimento

- Sono considerate le *attività di pianificazione della produzione* e di *gestione delle risorse* dello stabilimento
 - **Attività effettuate off-line** a partire da valutazioni storiche o statistiche
 - *Esempio*: allocazione di componenti in un magazzino automatico in funzione della frequenza di prelievo/deposito
 - *Esempio*: pianificazione delle esecuzioni di lavorazioni su più macchine per massimizzare la capacità produttiva dell'impianto in base alla conoscenza a priori dei tempi di lavorazione delle macchine
 - In caso di **eventi non previsti** queste pianificazioni fatte a priori perdono di validità e devono essere rifatte

sistemi per l'esecuzione della produzione (MES)

Sistemi MES



– Livello CIM per la gestione integrata

- I sistemi SCADA (presenti in questo livello) conoscono lo stato dell'impianto in tempo reale

→ A questo livello è possibile implementare delle **funzioni di pianificazioni e gestione in tempo reale**, ovvero dipendenti in maniera continua dallo stato dell'impianto:

→ **MES** (*Manufacturing Execution System*)

- Sistemi che realizzano un ponte tra il mondo della produzione (controllo e supervisione) e il modo della gestione dello stabilimento (pianificazione delle attività, allocazione delle risorse)
- A partire dalle informazioni sullo stato dell'impianto, i sistemi MES **producono informazioni** che permettono *l'ottimizzazione delle attività di produzione* (dall'ordine alla consegna di prodotti finiti)

sistemi per l'esecuzione della produzione (MES)

Sistemi MES



- Usando dati attuali ed accurati i MES pianificano e dirigono le attività di produzione garantendo un rapido adeguamento alle variazioni delle condizioni di produzione
- Alcuni problemi che possono essere risolti con i MES sono:
 - Un processo critico per la produzione ha bisogno di manutenzione preventiva
 - Una macchina di produzione si è guastata improvvisamente
 - Un cliente richiede che ogni prodotto consegnato sia corredato da informazioni sulla produzione
 - Un cliente vuole sapere se, raddoppiando il suo ordine, la fornitura può ancora essere eseguita nei tempi previsti
 - Il costo di un prodotto dipende anche dall'effettivo tempo-uomo che è stato necessario alla sua produzione
 - Un cliente vuole sapere a che punto è il soddisfacimento di un suo ordine

sistemi per l'esecuzione della produzione (MES)

Sistemi MES



- Le funzioni principali svolte da un MES sono:
 - Pianificazione delle attività di produzione per migliorare le prestazioni dell'impianto data la capacità finita delle risorse
 - Allocazione delle risorse materiali e umane (chi deve fare cosa?) e loro monitoraggio (cosa stanno facendo adesso?)
 - Allocazione delle merci (dove deve andare? quando?)
 - Gestione della documentazione sulle merci, le risorse, i processi di produzione
 - Tracciabilità dei prodotti, in modo che si possa sempre risalire alla loro storia nell'impianto
 - Analisi delle prestazioni di produzione secondo metriche standardizzate o proprietarie
 - Gestione della attività di manutenzione per minimizzare l'impatto sulla produzione

sistemi per l'esecuzione della produzione (MES)

Sistemi MES



- I MES sono un **insieme di applicazioni software**
 - Tipicamente è utilizzata una tecnologia client/server e basi dati relazionali
 - Sono previste interfacce per la connessione ad altri sistemi di gestione e supervisione
 - Viene eseguito da uno o più calcolatori (PC, workstation, minielaboratori) connessi sulla rete per le informazioni



sistemi di controllo distribuito (DCS)

Sistemi DCS

- **Sistemi di controllo distribuito (DCS)**
 - *Controllo distribuito*: si intende un qualsiasi sistema di controllo per l'automazione industriale o controllo di processo costituiti da **più sistemi interconnessi**
 - **DCS** (*Distributed Control System*) si intendono sistemi caratterizzati dalle seguenti peculiarità:
 - Utilizzati principalmente nell'industria di processo
 - È prevalente il controllo a ciclo chiuso per la regolazione di variabili rispetto all'automazione/logica sequenziale
 - Sono sistemi proprietari
 - Un solo fornitore fornisce tutti i dispositivi di controllo e il software necessario



sistemi di controllo distribuito (DCS)

Sistemi DCS

- Sono sistemi chiusi
 - Non è possibile (o lo è con difficoltà) integrare altri dispositivi
 - Spesso anche le reti di comunicazione sono proprietarie
- Il software è altamente specializzato
 - Può prevedere modalità di programmazione ad altissimo livello
 - » *Esempio*: solo interconnessione grafica di moduli già sviluppati
 - Orientate soprattutto alla realizzazione di controlli a ciclo chiuso per la regolazione di variabili
- Comprendono uno o più dispositivi che realizzano il cosiddetto *Engineering System*
 - Permette una rapida configurazione del sistema
 - Assolve funzionalità tipiche di uno SCADA
 - Assolve (raramente) funzionalità tipiche di un sistema MES

sistemi di controllo distribuito (DCS)

Sistemi DCS



- Architetture con un alto grado di ridondanza
 - Sistemi completamente o parzialmente a **massima disponibilità**, capaci di continuare ad operare anche in presenza di malfunzionamenti



III PARTE

Programmazione dei Dispositivi di Controllo

Linguaggi di programmazione

Unità di organizzazione della programmazione

Diagramma funzionale sequenziale (SFC)

programmazione dei dispositivi di controllo

Introduzione



- **Programmazione dei dispositivi di controllo**
 - Moderni sistemi di automazione
 - Insieme di dispositivi (calcolatori) diversi tra loro differenziati in base alle funzionalità da espletare
 - Utilizzo di *linguaggi di programmazione comuni* con l'aggiunta di funzioni (librerie) speciali
 - *Esempio*: istruzioni per la movimentazione di un robot
 - Utilizzo di *linguaggi di programmazione specializzati* alle loro funzionalità
 - *Esempio*: istruzioni per la descrizione di profili da realizzare con un tornio a controllo numerico

programmazione dei dispositivi di controllo

Introduzione



- Controllore a logica programmabile (PLC)
 - Dispositivi di ampia diffusione, specializzato per l'automazione industriale (controllo logico/sequenziale), altamente flessibile
 - Non sono adottati linguaggi di programmazione comuni
- Esistono diversi linguaggi per la programmazione dei PLC
 - Spesso caratterizzati anche da vari “dialetti” diversi da macchina a macchina, a volte anche tra macchine dello stesso produttore
 - Convivono macchine di nuova generazione con macchine più vecchie
 - Non presentano alcune caratteristiche introdotte solo di recente



programmazione dei dispositivi di controllo

Introduzione

- Svantaggi degli attuali linguaggi di programmazione e della mancanza di una loro standardizzazione:
 - Utilizzo di linguaggi di basso livello orientati alla macchina
 - Programmi debolmente strutturati
 - Difficoltà di programmazione top-down o bottom-up
 - Difficoltà di manutenzione
 - Difficoltà di intervenire su programmi già sviluppati per estenderne le funzionalità
 - È assente la *data encapsulation*
 - Non si possono nascondere i dati su cui opera un modulo → altri moduli potrebbero erroneamente alterarli
 - Difficile riutilizzare moduli già sviluppati o trasportarli su altre macchine (*portabilità del software*)



programmazione dei dispositivi di controllo

Introduzione

- Dati debolmente strutturati
 - Complesso il trattamento e la comunicazione di dati complessi
- Difficile definire sequenze logiche o parallelismi
- Mancanza di determinismo nel tempo di esecuzione dei cicli di controllo
- Difficoltà nel definire operazioni aritmetiche
- Programmatori devono essere formati su diversi programmi e dialetti → elevati costi di formazione
- Non è possibile utilizzare diversi programmi contemporaneamente
 - Non si possono scegliere i linguaggi migliori a seconda delle funzionalità che si devono implementare

programmazione dei dispositivi di controllo

Introduzione



- **Standard per la programmazione dei dispositivi di controllo CEI 61131-3**
 - Nasce per superare i limiti e i problemi elencati
 - Frutto di **interessi commerciali** → non è esente da *ambiguità e incoerenze*
 - **I costruttori non devono necessariamente adeguarsi a tutto lo standard per dichiararsi conforme ad esso**
 - L'esistenza di uno standard di riferimento rappresenta comunque un notevole passo in avanti che caratterizzerà inevitabilmente lo sviluppo per le future generazioni di PLC

programmazione dei dispositivi di controllo

Introduzione



- Ciclo base di un controllore in retroazione
 1. Assunzione di informazioni sullo stato di variabili del processo
 2. Elaborazione delle informazioni acquisite per produrre i valori di controllo
 3. Invio dei valori di controllo al processo



programmazione dei dispositivi di controllo

Introduzione

- Modalità di esecuzione del ciclo di controllo
 - *Periodicamente*
 - A intervalli di tempo assegnati (*esempio*: ogni 20 ms)
 - *Ciclicamente*
 - Appena terminata l'esecuzione delle tre operazioni di riprende daccapo
 - *Una sola volta*
 - All'occorrenza di eventi particolari (*esempio*: raggiungimento di un certo livello, azionamento di un fine corsa, ecc.)
 - Sono imposti vincoli temporali sul ritardo massimo ammissibile nell'esecuzione delle operazioni necessarie

programmazione dei dispositivi di controllo

Introduzione



- Programmazione dei dispositivi di controllo
 - Faremo riferimento allo **standard CEI 61131**
 - CEI 61131, *Controllori Programmabili*, parte 3, Linguaggi di programmazione, del Comitato Elettronico Internazionale
 - Recepito in Italia nel 1996 dal Comitato Elettronico Italiano
 - Saranno presentati solo le parti che interessano gli aspetti puramente di programmazione
 - Lo standard descrive con estremo dettaglio anche aspetti quali il numero di bit da usare per rappresentare una variabile intera, ecc.
 - Sono di interesse gli aspetti che consentono di programmare i sistemi di controllo in modo moderno ed efficiente



programmazione dei dispositivi di controllo

Introduzione

- Lo standard è relativamente recente
 - Molti controllori programmabili già installati e ancora commercializzati non lo rispettano
 - Per questi dispositivi non sono previste le caratteristiche innovative descritte dallo standard
 - La progettazione del software per questi dispositivi dovrebbe comunque avvenire secondo i dettami dello standard → programmi **modulari, leggibili, strutturati**
 - Dal progetto si passerà all'implementazione in funzione dello specifico dispositivo a disposizione e del tipo di linguaggio da esso supportato utilizzando apposite **regole di traduzione** (automatiche)



variabili e tipi di variabili

Definizione

- **Variabili e tipi di variabili**
 - Sono il mezzo con cui è possibile *rappresentare i dati* all'interno del dispositivo di controllo
 - Lo standard prevede alcuni *tipi predefiniti* di variabili
 - Possono essere direttamente usati nella dichiarazione di variabili
 - **Definizione in forma testuale** indipendentemente dal linguaggio di programmazione utilizzato
 - *Eccezione*: nella definizione di funzioni e di blocchi funzionali le variabili di ingresso e di uscita possono essere definite in **forma grafica**



variabili e tipi di variabili

Tipi predefiniti

– *Numeri interi*

- Tipo INT
 - Rappresenta **numeri interi** nell'intervallo $[-32.768, 32.767]$
- Tipo UINT
 - Rappresenta **numeri interi non negativi** nell'intervallo $[0, 2^{16}-1]$
- Sono previsti altri tipi interi con diversi intervalli più o meno estesi

– *Numeri reali*

- Tipo REAL
 - Rappresenta **numeri reali** nell'intervallo $\pm 10^{\pm 38}$ con una precisione di una parte su 2^{23}
- Tipo LREAL
 - Rappresenta **numeri reali** in un intervallo più esteso con maggior precisione

variabili e tipi di variabili

Tipi predefiniti



– Variabili temporali

- Tipo TIME

- Rappresenta una **durata temporale** in unità che vanno dai giorni ai millisecondi
- Formato di rappresentazione $T\#\alpha d\beta h\gamma m\delta s\epsilon ms$ per indicare una durata di α giorni, β ore, γ minuti, δ secondi ed ϵ millisecondi
- *Esempi:*

- » $T\#2d5h30m$ per indicare una durata di 2 giorni, 5 ore e 30 minuti
- » $T\#5m100ms$ per indicare una durata di 5 minuti e 100 millisecondi
- » $T\#120s$ oppure $T\#2m$ per indicare una durata di 120 secondi

- Tipi DATE e TIME_OF_DAY

- Rappresentano la **data** e l'**ora**, rispettivamente



variabili e tipi di variabili

Tipi predefiniti

- Tipo DATE_AND_TIME
 - Rappresentano **data e ora**
- *Stringhe di caratteri*
 - Tipo STRING
 - Rappresenta una **stringa di caratteri**
- *Stringhe di bit*
 - Tipo BOOL
 - Rappresenta un **singolo bit** o una **variabile logica**
 - Tipo BYTE
 - Rappresenta una **stringa di 8 bit**
 - Tipo WORD
 - Rappresenta una **stringa di 16 bit**



variabili e tipi di variabili

Tipi predefiniti

- Tipo DWORD
 - Rappresenta una **stringa di 32 bit**
- Tipo LWORD
 - Rappresenta una **stringa di 64 bit**
- *Variabili generiche*
 - Tipo ANY
 - Rappresenta una variabile di **qualsiasi tipo**
 - Tipo ANY_NUM
 - Rappresenta un **qualsiasi dato numerico** (intero o reale)
 - Tipo ANY_INT
 - Rappresenta un **qualsiasi numero intero**
 - Tipo ANY_REAL
 - Rappresenta un **qualsiasi numero reale**



variabili e tipi di variabili

Tipi predefiniti

- Tipo ANY_DATE
 - Rappresenta un **qualsiasi variabile temporale** (date e/o ora)
- Tipo ANY_BIT
 - Rappresenta un **qualsiasi stringa di bit**



variabili e tipi di variabili

Non ambiguità

– Non ambiguità

- Requisito fondamentale per l'automazione è la **non ambiguità** dei valori che può assumere una variabile → all'atto della dichiarazione ad ogni variabile deve sempre essere assegnato un valore
- Valori predefiniti:
 - Variabili *numeriche*: valore zero
 - Variabili *stringa*: stringa nulla
 - Variabili *data*: 01/01/0001

variabili e tipi di variabili

Tipi derivati



- L'utilizzatore può definire dei **tipi derivati** in vari modi:
 - Per *equivalenza* a quelli definiti
 - Per *enumerazione*
 - Per *restrizione* di tipi già definiti
 - Definendo insiemi ordinati di più elementi dello stesso tipo (*array*)
 - Definendo insiemi di più elementi di tipo diverso (*strutture*)



variabili e tipi di variabili

Esempio

- *Esempio*, con valore di inizializzazione:

TYPE

```
impulsi : UINT(0..1000);  
stato : (fermo, funzionante, guasto, attesa);  
temperatura : REAL := 20.0;  
sensore_temperatura : STRUCT  
    valore : temperatura;  
    ultima_calibrazione : DATE;  
    intervallo_calibrazione : TIME;  
    valore_massimo : REAL := 100.0;  
    diagnostica : BOOL;
```

END_STRUCT

```
dati_termici_forno : ARRAY[1..10] OF  
    sensore_temperatura;  
dati_termici_insieme_forni : ARRAY[1..4,1..4]  
    OF dati_termici_forno;
```

END_TYPE



variabili e tipi di variabili

Esempio

- Mediante le parole chiave TYPE ed END_TYPE si sono introdotti:
 - Il tipo `impulsi`: equivale al tipo intero senza segno ma limitato ad appartenere all'intervallo [0, 1000]
 - Il tipo `stato`: ha i quattro possibili valori elencati, definito per rappresentare lo stato di macchine
 - Il tipo `temperatura`: equivalente al tipo numero reale, introdotto per migliorare la leggibilità, il cui valore iniziale è 20 e non 0
 - Il tipo `sensore_temperatura`: tipo strutturato (ottenuto utilizzando il costrutto `STRUCT...END_STRUCT`) composto da vari campi di tipo diverso che rappresenta i dati associati ad un ipotetico sensore di temperatura
 - Il tipo `dati_termici_forno`: è un vettore di elementi (parola chiave `ARRAY [...] OF`), ognuno del tipo `sensore_temperatura` supposto essere rappresentativo di 10 sensori di temperatura presenti in un ipotetico forno

variabili e tipi di variabili

Esempio

- Il tipo `dati_termici_insieme_forni`: che rappresenta una matrice di elementi, ognuno del tipo `dati_termici_forno`, la quale potrebbe essere rappresentativa di 16 forni collocati in un ipotetico stabilimento
 - » *Esempio*: 4 linee di produzione, ognuna delle quali è dotata di 4 forni
- L'introduzione di nuovi tipi migliora la **leggibilità** del software e consente di **organizzare i dati**
- L'uso di variabili strutturate consente di **comunicare più facilmente** un insieme di dati concernenti lo stesso concetto





variabili e tipi di variabili

Dichiarazione

– **Dichiarazione** delle variabili

- Le variabili vanno dichiarate all'inizio delle definizioni dei programmi, delle funzioni e dei blocchi funzionali
- Il costrutto per la dichiarazione delle variabili è VAR...END_VAR

– *Esempio:*

VAR

```
A, B : REAL;  
abilitazione : BOOL;  
conteggio : impulsi;  
stato_tornio : stato;  
termometro_7 : temperatura := 0.0;  
termocoppia_1, termocoppia_2 :  
    sensore_temperatura;  
forno_1 : dati_termici_forno;
```

END_VAR



variabili e tipi di variabili

Inizializzazione ed accesso

– Inizializzazione delle variabili

- L'inizializzazione delle variabili può anche avvenire all'atto della loro definizione
 - Il valore indicato sostituirà quello predefinito per il tipo

– Accesso alle variabili a più elementi

- Per *indice* nel caso di array
 - *Esempio*: `forno_1[7]` rappresenta il settimo elemento dell'array
- Per *campo* nel caso di strutture
 - *Esempio*: `Termocoppia_1.valore` rappresenta il campo `valore` della variabile strutturata `Termocoppia_1`



variabili e tipi di variabili

Variabili di ingresso e di uscita

– Variabili di ingresso e di uscita

- Si utilizzano i costrutti:
 - VAR_INPUT...END_VAR per le variabili di ingresso
 - » I loro valori devono essere forniti da unità di programmazione esterne rispetto a quella in cui sono definite
 - » **Non sono modificabili**
 - VAR_OUTPUT...END_VAR per le variabili di uscita
 - » I valori saranno scritti in variabili esterne all'unità di programmazione
 - » Sono ovviamente variabili modificabili
 - VAR_IN_OUT...END_VAR per le variabili di ingresso/uscita
 - » Fanno riferimento all'indirizzo di variabili esterne all'unità di programmazione e sono da essa modificabili

variabili e tipi di variabili

Visibilità



– **Visibilità** delle variabili

- La visibilità è **interna all'unità di programmazione** in cui sono dichiarate e non sono accessibili dall'esterno
- Fanno eccezioni le variabili definite di ingresso o di uscita
- **Variabili globali**
 - Dichiarate con il costrutto `VAR_GLOBAL...END_VAR`
 - Le variabili globali sono **visibili alle entità interne**
 - » Definizione a livello di programma → visibilità per le funzioni e i blocchi funzionali definiti nel programma
 - » Definizione a livello di configurazione → visibilità per le risorse e i programmi definiti nel dispositivo
 - Per usare una variabile globale definita in una entità esterna si deve dichiarare la variabile con il costrutto `VAR_EXTERNAL...END_VAR`
 - Possono essere definite solo a livello di programma, risorsa e configurazione



variabili e tipi di variabili

Variabili a rappresentazione diretta

– Variabili a rappresentazione diretta

- Consentono di riferirsi direttamente a specifiche locazioni di memoria
- Notazione: %ABxxx
 - A è il prefisso di locazione:
 - » I locazioni di memoria riservate agli ingressi
 - » Q locazioni di memoria riservate alle uscite
 - » M locazioni di memoria generica
 - B è il prefisso di taglia:
 - » X (o mancante) per rappresentare un bit
 - » B per rappresentare un byte (8 bit)
 - » W per rappresentare una word (16 bit)
 - » D per rappresentare una doppia word (32 bit)
 - » L per rappresentare una word lunga (64 bit)



variabili e tipi di variabili

Variabili accessibili

- Il codice XXX dipende dal dispositivo
 - » È formato da numeri interi eventualmente separati da punti
 - » Definisce l'indirizzo della locazione cui si riferisce

– Variabili accessibili

- Sono definite con la parola chiave `VAR_ACCESS...END_VAR`
- Possono essere indirizzate da programmi remoti
- Si possono riferire solo alle variabili di ingresso e di uscita di un programma, alle variabili globali e alle variabili a rappresentazione diretta
- Si può specificare che una variabili è accessibile in sola lettura o in sola scrittura da parte di un programma remoto
- Modalità di accesso dipendono dalle implementazioni
- **Sono una astrazione delle funzionalità di comunicazione**



variabili e tipi di variabili

Attributi e commenti

– **Attributi**

- Alle parole chiave di definizione delle variabili si possono far seguire i seguenti attributi:
 - RETAIN: sono variabili che devono essere conservate in mancanza di alimentazione (se consentito dal dispositivo)
 - CONSTANT: indica che si tratta di costanti, sempre uguali alla loro inizializzazione (quindi non modificabili)
 - AT: indica che la variabile deve essere allocata in memoria in una particolare locazione
 - » Notazione identica a quella per la definizione delle variabili a rappresentazione diretta

– **Commenti**

- Sono racchiusi tra la coppia di simboli (* e *)
 - *Esempio:* (* questo è un commento *)

linguaggi di programmazione

Standard 61131-3



- Linguaggi previsti dallo **standard IEC 61131 – parte III**
 - Testo strutturato
 - Linguaggio a contatti
 - Diagramma a blocchi funzionali
 - Lista istruzioni
 - Sequential Functional Chart
 - Verrà presentato ed approfondito successivamente

linguaggi di programmazione

Testo strutturato



- **Testo strutturato**

- È il linguaggio di programmazione per i dispositivi di controllo più vicino ai linguaggi informatici attuali, molto simile al *Pascal*
- È un linguaggio di programmazione **testuale di alto livello** sviluppato per applicazioni di automazione



linguaggi di programmazione

Testo strutturato

– Assegnazione

- È programmata con il simbolo :=

- *Esempi:*

```
A := 2.5;  
abilitazione := TRUE;  
conteggio := 350;  
stato_tornio := funzionante;  
termometro_7 := 30.0;  
termocoppia_1.valore := 554.0;  
termocoppia_1.ultima_calibrazione := 2004-03-30;  
termocoppia_1.intervallo_di_calibrazione :=  
    T#120d;  
termocoppia_2 := termocoppia_1;  
forno_1[4] := termocoppia_1;
```

- Il carattere ; rappresenta il **terminatore** di un'istruzione



linguaggi di programmazione

Testo strutturato

– *Operatori aritmetici*

- + somma
- - differenza
- * prodotto
- / divisione
- MOD resto di una divisione intera
- ** elevazione a potenza

– *Operatori sul bit*

- AND (oppure &) per la funzione prodotto logico
- OR per la funzione di somma logica
- XOR per la funzione di somma logica esclusiva
- NOT per la negazione



linguaggi di programmazione

Testo strutturato

– *Operatori relazionali*

- < minore
- <= minore o uguale
- > maggiore
- >= maggiore o uguale
- = uguale
- <> diverso

– *Parenthesi*

- Le parentesi (e) consentono di cambiare l'ordine di esecuzione delle operazioni con le note regole sulla precedenza delle valutazioni

linguaggi di programmazione

Testo strutturato



– Istruzioni condizionate

- IF ... THEN ... END_IF
 - Per eseguire istruzioni solo se è verificata una condizione logica
- IF ... THEN ... ELSE ... END_IF
 - Per eseguire istruzioni diverse a seconda del valore di una condizione logica
- IF ... THEN ... ELSEIF ... THEN ... ELSE ... END_IF
 - Per definire delle decisioni multiple, usando più ELSEIF
- CASE ... OF ... ELSE ... END_CASE
 - Per definire decisioni multiple che dipendono dal valore di una variabile intera

linguaggi di programmazione

Testo strutturato



- *Esempi:*

```
IF abilitazione & (conteggio<100) THEN
    conteggio := 100;
    stato_tornio := fermo;
END_IF
```

```
IF abilitazione THEN
    termocoppia_1.valore := termocoppia_2.valore;
ELSE
    termocoppia_1.valore := 0.0;
END_IF
```




linguaggi di programmazione

Testo strutturato

```
IF conteggio<100 THEN
  A := 1.0;
ELSEIF conteggio<500 THEN
  A := 2.0;
ELSEIF conteggio<800 THEN
  A := 3.0;
ELSE (* conteggio maggiore o uguale a 800 *)
  A := 4.0;
END_IF

CASE conteggio OF
  1 : stato_tornio := fermo;
  2, 3, 4, 5 : stato_tornio := funzionante;
  6..50 : stato_tornio := attesa;
ELSE
  stato_tornio := guasto;
END_CASE
```

linguaggi di programmazione

Testo strutturato



– Cicli

- `FOR ... TO ... BY ... DO ... .. END_FOR`
 - Per l'esecuzione di cicli basati sul valore di una variabile intera, la quale viene incrementata di un certo passo da un valore iniziale ad uno finale
- `WHILE ... DO ... .. END_WHILE`
 - Per l'esecuzione di cicli fintantoché un'espressione logica rimane vera (la verifica viene fatta prima dell'esecuzione delle istruzioni del ciclo)
- `REPEAT ... .. UNTIL ... END_REPEAT`
 - Per l'esecuzione di cicli fintantoché un'espressione logica rimane vera (la verifica viene fatta dopo dell'esecuzione delle istruzioni del ciclo)
- `EXIT`
 - Per uscire dai cicli in maniera immediata



linguaggi di programmazione

Testo strutturato

- *Esempi:*

```
FOR conteggio=10 TO conteggio=6 BY -1 DO
    forno_1[conteggio] := forno_1[conteggio-5];
    IF forno_1[conteggio].valore = 100.0 THEN
        EXIT;
    END_IF
END_FOR
```

```
WHILE termometro_7 > 100.0 DO
    conteggio := conteggio + 1;
END_WHILE
```

```
REPEAT
    conteggio := conteggio - 1;
UNTILE termometro_7 < 30.0;
END_REPEAT
```

linguaggi di programmazione

Testo strutturato



– *Terminazione anticipata*

- La parola chiave RETURN consente di interrompere anticipatamente l'esecuzione di funzioni e blocchi funzionali

linguaggi di programmazione

Linguaggio a contatti



- **Linguaggio a contatti (Ladder Diagram)**

- Traduzione letterale: *diagramma a scala*, dalla forma che assume il programma
- È il **più diffuso** linguaggio di programmazione per i controllori a logica programmabile
- Cenni storici
 - Le prime applicazioni di automazione industriale venivano realizzate utilizzando dispositivi elettromeccanici
 - Relè, temporizzatori, contatori, sequenziatori a cilindro
 - Avvento di controllori a programma memorizzato → sviluppo di un linguaggio di programmazione vicino alla tecnologia elettromeccanica → poteva essere compreso e usato dai tecnici dell'epoca



linguaggi di programmazione

Linguaggio a contatti

- Prime istruzioni disponibili rappresentavano:
 - Contatti normalmente aperti di un relè
 - Contatti normalmente chiusi di un relè
 - La bobina di eccitazione di un relè
 - Il temporizzatore
 - Il contatore
- La **forma a scala** assunta dal programma deriva dalla logica a relè
 - Due linee verticali (montanti della scala) rappresentanti l'alimentazione
 - Linee orizzontali (pioli della scala, *rung*) che alimentano una bobina se una certa combinazione di contatti abilita il flusso di energia

linguaggi di programmazione

Linguaggio a contatti



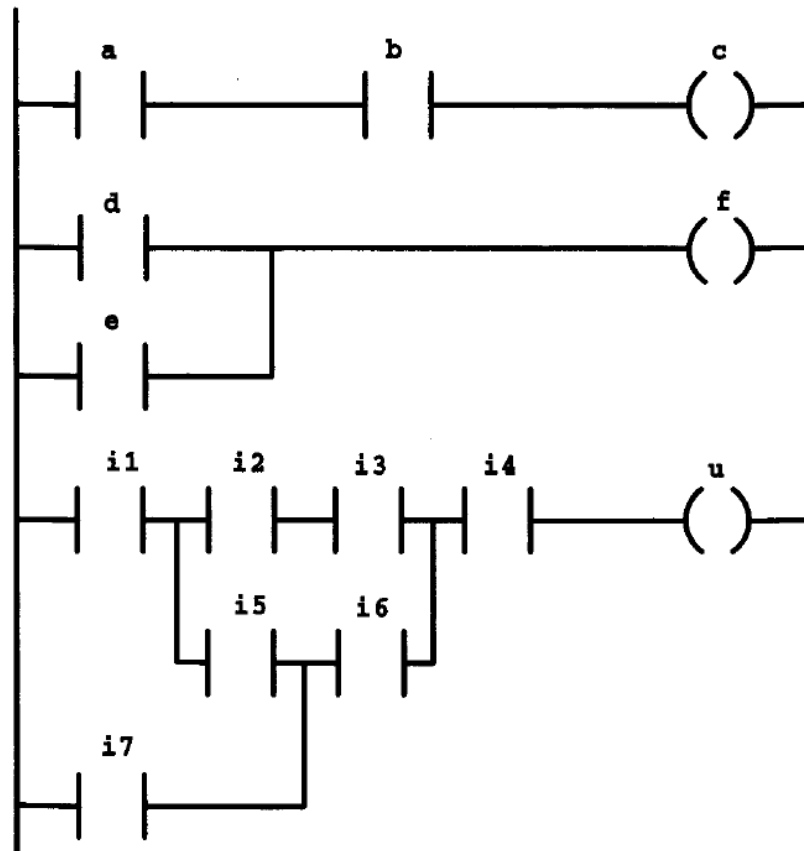
- I **contatti** possono essere associati:
 - Agli ingressi digitali provenienti dal processo, ovvero allo stato dei bit di memoria che li rappresentano
 - A condizioni interne del dispositivo, rappresentate da bit di memoria
- La **bobina** può essere associata ad un bit di memoria
 - Tramite il suo valore può comandare una uscita digitale o variare una condizione interna (un bit di memoria)
- Il linguaggio a contatti è particolarmente adatto a definire operazioni logiche booleane
 - La forma grafica è di notevole ausilio alla comprensione del significato delle istruzioni programmate
- **Flusso di energia** può andare sempre e solo *da sinistra a destra senza possibilità di inversione*
- **L'esecuzione delle istruzioni** avviene *dall'alto verso il basso e da sinistra verso destra*

linguaggi di programmazione

Linguaggio a contatti



- *Esempio*: rete in linguaggio a contatti





linguaggi di programmazione

Linguaggio a contatti

- Sono implementate le funzioni booleane
 - $c = a \text{ AND } b$
 - » La bobina C viene alimentata solo se sono chiusi entrambi i contatti a e b, permettendo i flussi di energia
 - $f = d \text{ OR } e$
 - » La bobina f viene alimentata solo se viene chiuso almeno uno dei contatti d e e, permettendo i flussi di energia
 - La bobina u può essere alimentata attraverso le sequenze di contatti $i1-i2-i3-i4$, $i1-i5-i6-i4$, $i7-i6-i4$
 - » Non può essere alimentata dalla sequenza $i7-i5-i2-i3-i4$

– Barre di alimentazione

- Una rete in linguaggio a contatti è delimitata a sinistra e a destra da due linee verticali che rappresentano le barre di alimentazione
 - La barra di destra può anche essere omessa

linguaggi di programmazione

Linguaggio a contatti



- Il **contatto** è un elemento che consente il flusso di energia verso destra in funzione dello stato del bit a cui è associato
 - I contatti sono *elementi di valutazione* dello stato di variabili booleane (bit) e servono per permettere o no il flusso di energia nel rung dove sono installate
 - I contatti disponibili sono:
 - **Contatto normalmente aperto** - - | | - -
 - » Il contatto si chiude se la variabile booleana associata diventa vera, consentendo il flusso di energia nel rung in cui è installato
 - » Se il valore della variabile è falso il contatto rimarrà aperto, impedendo il flusso di energia



linguaggi di programmazione

Linguaggio a contatti

- **Contatto normalmente chiuso** - - | / | - -
 - » Il contatto si chiude se la variabile booleana associata diventa falsa, consentendo il flusso di energia nel rung in cui è installato
 - » Se il valore della variabile è vero il contatto rimarrà aperto, impedendo il flusso di energia
- **Contatto a fronte di salita** - - | P | - -
 - » Si chiude solo quando il valore della variabile booleana associata passa, tra due successive esecuzioni della stessa istruzione, da falso a vero, permettendo il flusso di energia
- **Contatto a fronte di discesa** - - | N | - -
 - » Si chiude solo quando il valore della variabile booleana associata passa, tra due successive esecuzioni della stessa istruzione, da vero a falso, permettendo il flusso di energia

linguaggi di programmazione

Linguaggio a contatti



- Le **bobine** sono *elementi per operare* sul bit
 - Se alimentate dal flusso di energia, agiscono sul valore della variabile booleana associata
 - Le bobine disponibili sono:
 - **Bobina semplice** -- () --
 - » Se alimentata assegna alla variabile booleana associata il valore uno (vero), altrimenti il valore zero (falso)
 - **Bobina negata** -- (/) --
 - » Se non è alimentata assegna alla variabile booleana associata il valore uno, altrimenti il valore zero
 - **Bobina a memorizzazione SET o aggancio bobina** -- (S) --
 - » Se alimentata assegna il valore uno alla variabile booleana associato
 - » Tale valore viene conservato anche quando la bobina non è più alimentata

linguaggi di programmazione

Linguaggio a contatti



- **Bobina a memorizzazione RESET o sgancio bobina - - (R) - -**
 - » Se alimentata assegna il valore zero alla variabile booleana associato
 - » Tale valore viene conservato anche quando la bobina non è più alimentata
- **Bobina a fronte di salita - - (P) - -**
 - » Assegna il valore uno alla variabile booleana associata solo quando la sua alimentazione, tra due successive esecuzioni, passa da assente a presente
 - » Verrà generato un *impulso rettangolare* di durata pari all'intervallo tra le due successive esecuzioni
- **Bobina a fronte di discesa - - (N) - -**
 - » Assegna il valore uno alla variabile booleana associata solo quando la sua alimentazione, tra due successive esecuzioni, passa da presente ad assente



linguaggi di programmazione

Linguaggio a contatti

- **Bobina a ritenuta** - - (M) - - , - - (SM) - - , - - (RM) - -
 - » Corrispondono, rispettivamente, a - - () - - , - - (S) - - , - - (R) - - , ma la variabile booleana associata deve essere mantenuta in caso di mancanza dell'alimentazione
 - » Equivalente a utilizzare la parola chiave RETAIN nella definizione della variabile associate agli elementi non a ritenuta

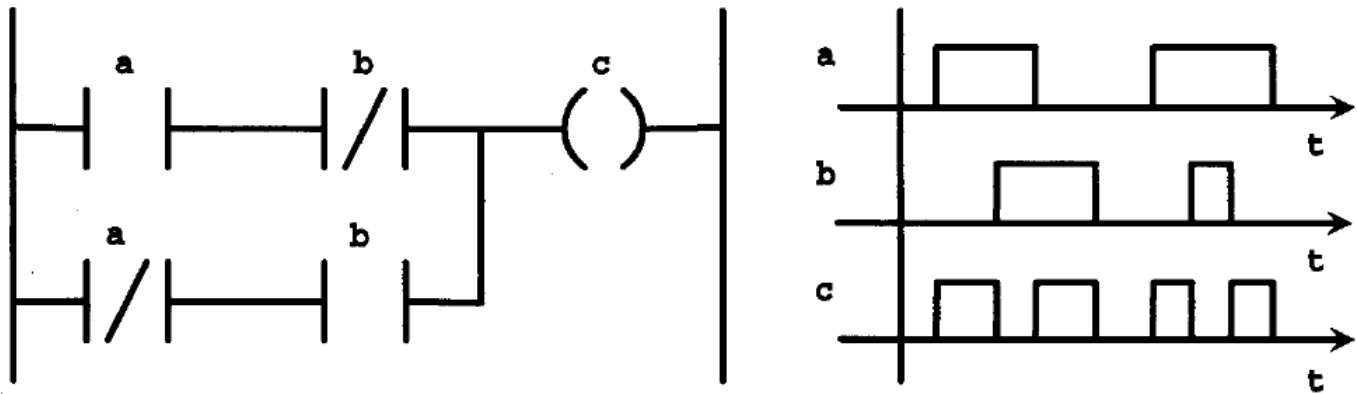
linguaggi di programmazione

Linguaggio a contatti



– Esempio: funzione XOR

- Rete in linguaggio a contatti con la relativa cronologia per i valori delle variabili booleane interessate



- Espressione equivalente in testo strutturato:
$$c := (a \text{ AND NOT}(b)) \text{ OR } (\text{NOT}(a) \text{ AND } b);$$

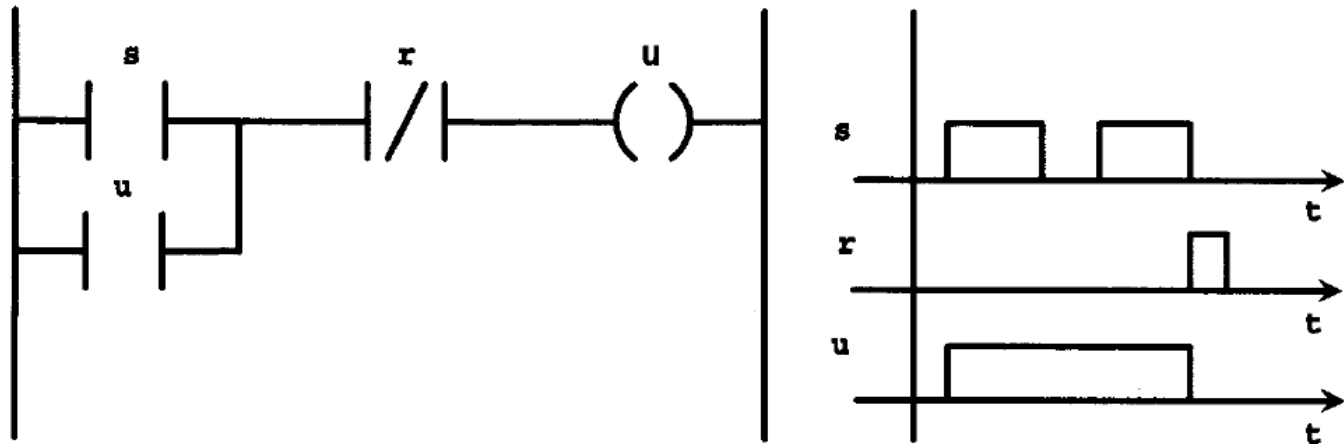
linguaggi di programmazione

Linguaggio a contatti



– Esempio: flip-flop SET/RESET

- Rete in linguaggio a contatti che *memorizza lo stato di una variabile booleana* attraverso l'**autoalimentazione della bobina**
 - La rete deve essere eseguita in modalità ciclica o periodica



- Espressione equivalente in testo strutturato:
$$u := \text{NOT}(r) \text{ AND } (s \text{ OR } u);$$
 - La U al secondo membro è relativa all'esecuzione precedente

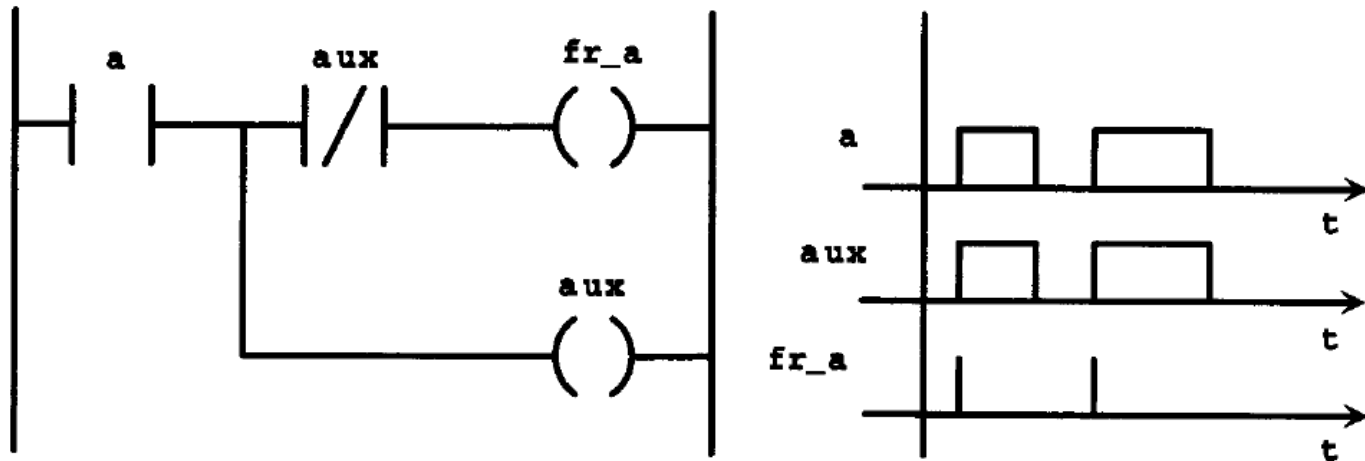
linguaggi di programmazione

Linguaggio a contatti



– Esempio: riconoscimento di un fronte di salita

- Il circuito è equivalente ad un *contatto a fronte di salita*
- Rete in linguaggio a contatti



- Espressione equivalente in testo strutturato:
fr_a := NOT(aux) AND a;
aux := a;

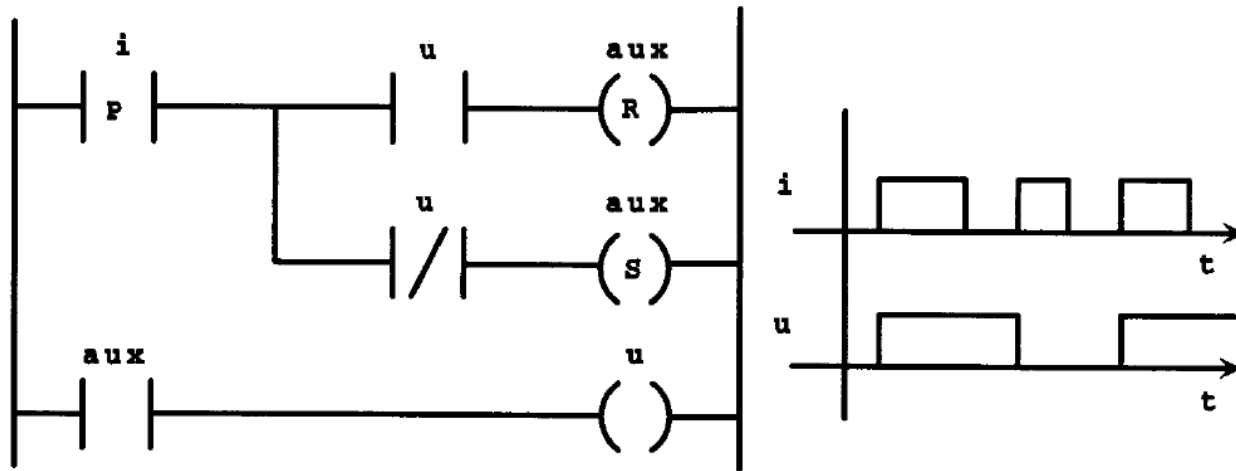
linguaggi di programmazione

Linguaggio a contatti



– *Esempio*: flip-flop a commutazione

- Rete in linguaggio a contatti





linguaggi di programmazione

Linguaggio a contatti

- Espressione equivalente in testo strutturato:

```
fr_i := NOT(i_precedente) AND i;  
i_precedente := i;  
IF fr_i THEN  
    IF u THEN  
        u := FALSE;  
    ELSE  
        u := TRUE;  
    END_IF  
END_IF
```

- Le prime due istruzioni realizzano il contatto sul fronte
- Non occorre una variabile ausiliaria grazie all'uso di una istruzione condizionata per l'assegnazione del valore alla variabile u

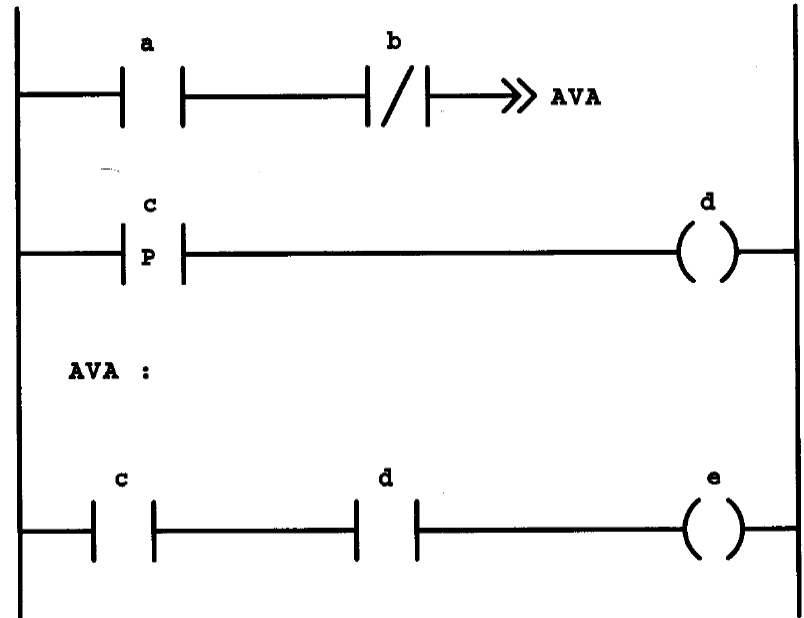
linguaggi di programmazione

Linguaggio a contatti



– Controllo di flusso

- Realizzato con il *salto*
- Si ottiene ponendo in un rung il costrutto - - ->>XXXX
 - xxxx è una stringa (*label*) che indica dove si deve saltare
 - La stringa è ripetuta prima del rung dove si deve saltare
 - *Esempio*: se *a* è vera e *b* è falsa si salta l'esecuzione del rung successivo





linguaggi di programmazione

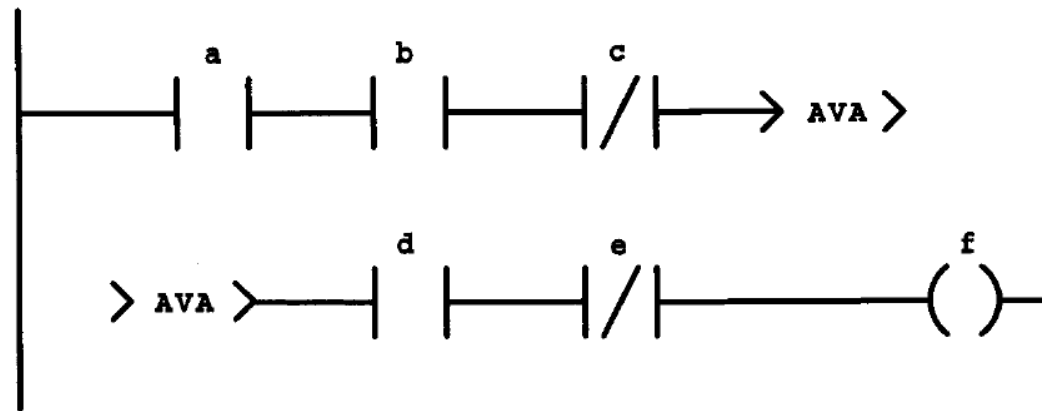
Linguaggio a contatti

– Istruzione di terminazione

- L'istruzione - -<RETURN>- - è utilizzata per la terminazione di funzioni e di blocchi funzionali

– Istruzione di connessione

- Per spezzare rung troppo lunghi in più rung viene utilizzata la coppia di istruzioni - ->yyyy> e >yyyy>- -, dove yyyy è una stringa (*label*)
- *Esempio:*



linguaggi di programmazione

Diagramma a blocchi funzionali



- **Diagramma a blocchi funzionali** (*Functional Block Diagram*)
 - Si tratta di un **linguaggio grafico**
 - Prevede la costruzione di *reti di componenti* in analogia a un circuito elettronico
 - Il **flusso di segnale** scorre sempre *da sinistra verso destra*
 - Particolarmente adatto per rappresentare graficamente:
 - Funzioni in logica booleana
 - Operazioni aritmetiche
 - Consente di costruire programmi complessi connettendo opportunamente blocchi funzionali e funzioni, eventualmente sviluppati in altri linguaggi



linguaggi di programmazione

Diagramma a blocchi funzionali

- Composizione di una **rete** in diagramma a blocchi funzionali
 - Non esistono *elementi intrinseci* del linguaggio
 - Le uscite di funzioni o blocchi funzionali sono connesse all'ingresso di altre
 - L'ingresso di una funzione o di un blocco funzionale può anche essere direttamente il valore di una variabile o un valore costante
 - Le uscite possono anche essere valori di variabili
- **Negazione** di una variabile booleana
 - Può essere indicata con un *cerchietto* posto prima dell'ingresso del blocco grafico in cui entra la linea del segnale

linguaggi di programmazione

Diagramma a blocchi funzionali



– Anelli chiusi

- È consentito l'utilizzo dell'uscita di una funzione come ingresso di un'altra funzione che la precede a sinistra nell'anello
- Un anello chiuso è interpretato in questo modo:
 - Il valore di uscita calcolato viene conservato e utilizzato come valore di ingresso alla successiva valutazione della rete
- Il valore dell'uscita può dipendere dall'ordine con cui sono valutati i blocchi che compongono l'anello
 - Bisogna fare attenzione al modo con cui l'ordine di valutazione è realizzato nel particolare dispositivo che si utilizza

– Connettori grafici

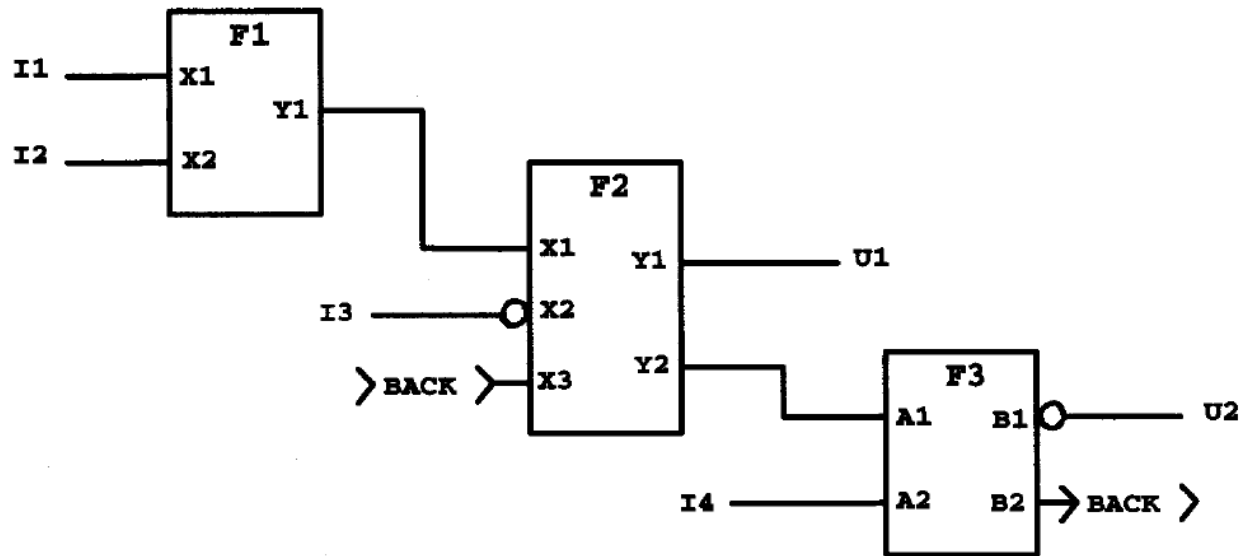
- Sono ammessi connettori grafici, simili a quelli del linguaggio a contatti ($-->yyyy>$ e $>yyyy>--$), per trasferire il flusso del segnale da una parte all'altra della rete

linguaggi di programmazione

Diagramma a blocchi funzionali



- *Esempio*: rete in diagramma a blocchi funzionali
 - Si considerano tre funzioni o blocchi funzionali F1, F2, F3
 - Si noti il ciclo chiuso tra una delle uscite di F3 a uno degli ingressi di F2 realizzato mediante il connettore grafico BACK



linguaggi di programmazione

Lista istruzioni



- **Lista istruzioni**

- Si tratta di un linguaggio di *basso livello* composto da sequenze di istruzioni
 - È un linguaggio di tipo **assemblativo**
- Ogni **istruzione** è composta da un operatore e da un operando
 - L'operatore può essere accompagnato da un **modificatore**
 - Ogni istruzione è scritta su un rigo diverso
 - L'istruzione può essere preceduta da una **etichetta**, composta da un nome seguita dal simbolo :

linguaggi di programmazione

Lista istruzioni



– Operatori

- Tutti gli operatori fanno sempre riferimento ad un **accumulatore**, oltre che all'operando indicato
- Gli operatori predefiniti sono:
 - LD per assegnare all'accumulatore il valore dell'operando
 - » L'operando può essere un valore assoluto o una variabile di qualunque tipo
 - ST per porre il valore dell'accumulatore nella variabile indicata dall'operando
 - S per porre una variabile booleana a 1, nel caso in cui il valore dell'accumulatore sia pari a uno
 - R per porre una variabile booleana a 0, nel caso in cui il valore dell'accumulatore sia pari a uno
 - AND oppure &, OR e XOR per realizzare operazioni logiche su variabili booleane

linguaggi di programmazione

Lista istruzioni



- ADD, SUB, MUL, DIV per realizzare le quattro operazioni aritmetiche
- GT, GE, EQ, NE, LE, LT per realizzare operazioni di comparazione (rispettivamente: maggiore, maggiore o uguale, uguale, diverso, minore o uguale, minore)
- JMP per saltare all’etichetta indicata nel campo operando
- CAL per realizzare la chiamata di un blocco funzionale
- RET per realizzare il ritorno da una funzione o da un blocco funzionale
- Chiamata di una funzione
 - Viene effettuata semplicemente indicandone il nome
 - Il risultato sarà posto nell’accumulatore

linguaggi di programmazione

Lista istruzioni



- **Modificatori**
 - **N** indica una negazione booleana dell'operando
 - » È utilizzabile con LD, ST, operatori booleani
 - **(** indica che la valutazione dell'operatore deve essere fatta solo quando si trova la corrispondente parentesi chiusa **)**
 - » È utilizzabile con gli operatori booleani, aritmetici e di comparazione
 - **C** indica che l'esecuzione dell'operatore è condizionata al fatto che il valore corrente dell'accumulatore sia 1 (o 0 se si usa anche il modificatore N)
 - » È utilizzabile per gli operatori JMP, CAL e RET

linguaggi di programmazione

Lista istruzioni



- *Esempio: funzione XOR*

```
LD      a    (* Metti il valore di a nell'accumulatore *)
ANDN    b    (* AND dell'accumulatore con il negato di b *)
OR(     b    (* OR dell'acc. col risultato dell'espressione *)
ANDN    a
)
ST      c    (* Conserva il valore dell'accumulatore in c *)
```

- *Espressione equivalente in testo strutturato:*

```
c := (a & NOT(b)) OR (b & NOT(a))
```



linguaggi di programmazione

Lista istruzioni

- *Esempio: flip-flop a commutazione*

```
LD      i                (* Queste istruzioni valutano *)
ANDN    i_precedente     (* il fronte di salita      *)
JMPCN   ava             (* Salta se non c'è il fronte di salita *)
LD      u                (* Queste istruzioni assegnano alla *)
STN     u                (* variabile u il suo valore negato *)
ava:    LD      i
        ST      i_precedente
```

unità di organizzazione della programmazione

Introduzione



- Organizzazione del software
 - Lo standard prevede alcuni **elementi comuni** che prescindono dall'effettivo linguaggio (o insieme di linguaggi) utilizzato per la programmazione
 - Le *variabili*, di cui si è data ampia trattazione in precedenza
 - Le *funzioni*, forniscono valori in uscita dipendenti dai soli valori in ingresso
 - I *blocchi funzionali*, forniscono in uscita valori dipendenti dai valori in ingresso e dallo stato attuale delle variabili interne
 - Le uscite dipendono dagli ingressi attuali, da quelli precedenti e dallo stato iniziale del blocco
 - I *programmi*, sono l'insieme logico degli elementi e dei costrutti di programmazione necessari al corretto trattamento dei dati

unità di organizzazione della programmazione

Introduzione



- **Unità di organizzazione della programmazione**
 - **POU: Program Organization Units**
 - Sono rappresentate, nel loro complesso, dalle *funzioni*, dai *blocchi funzionali* e dai *programmi*
 - Una POU è composta da tre elementi:
 - La *definizione* del tipo di POU e del suo nome
 - Nel caso di funzioni, anche del tipo di dato associato al nome
 - La *parte dichiarativa* delle variabili e dei loro attributi
 - Il *corpo* con le istruzioni nei linguaggi dello standard

unità di organizzazione della programmazione

Introduzione



- Oltre alle POU lo standard prevede:
 - I *compiti* (o *task*), sono gli elementi che controllano l'esecuzione dei programmi e blocchi funzionali su base periodica, ciclica o al verificarsi di eventi particolari
 - Le *risorse*, che corrispondono a entità in grado di eseguire programmi
 - La *configurazione*, l'elemento del linguaggio corrispondente a un dispositivo che comprende una o più risorse

unità di organizzazione della programmazione

Funzioni



- **Funzioni**

- Sono unità di organizzazione della programmazione *riusabili*
- A partire dai valori in ingresso calcolano **un solo dato in uscita**
 - Il dato di uscita può essere composto da più elementi
- Il dato in uscita è rappresentato dal **nome della funzione** stessa, che può essere usato direttamente nelle espressioni
- Il risultato **non deve dipendere** dallo stato di variabili interne → agli stessi ingressi devono corrispondere le stesse uscite

unità di organizzazione della programmazione

Funzioni



– **Definizione** delle funzioni:

- in forma *testuale*
- in forma *grafica*

– Definizione in forma testuale

- È costruita secondo il seguente schema:

```
FUNCTION nome_funzione : tipo (* tipo della *)
                                (* funzione   *)
```

```
VAR_INPUT
```

```
    ...; (* definizione delle variabili di *)
        (* ingresso                      *)
```

```
END_VAR
```

```
... (* altre definizioni di variabili *)
```

unità di organizzazione della programmazione

Funzioni



```
...      ( * corpo della funzione che deve * )  
          ( * prevedere l'assegnazione di    * )  
          ( * un valore a nome_funzione     * )
```

```
END_FUNCTION
```

- Il **corpo** della funzione può essere scritto in uno dei linguaggi dello standard tranne il Sequential Functional Chart
- Non possono essere definite variabili di uscita, di ingresso/uscita, a rappresentazione diretta, esterne, globali, accessibili, né a ritenuta

unità di organizzazione della programmazione

Funzioni

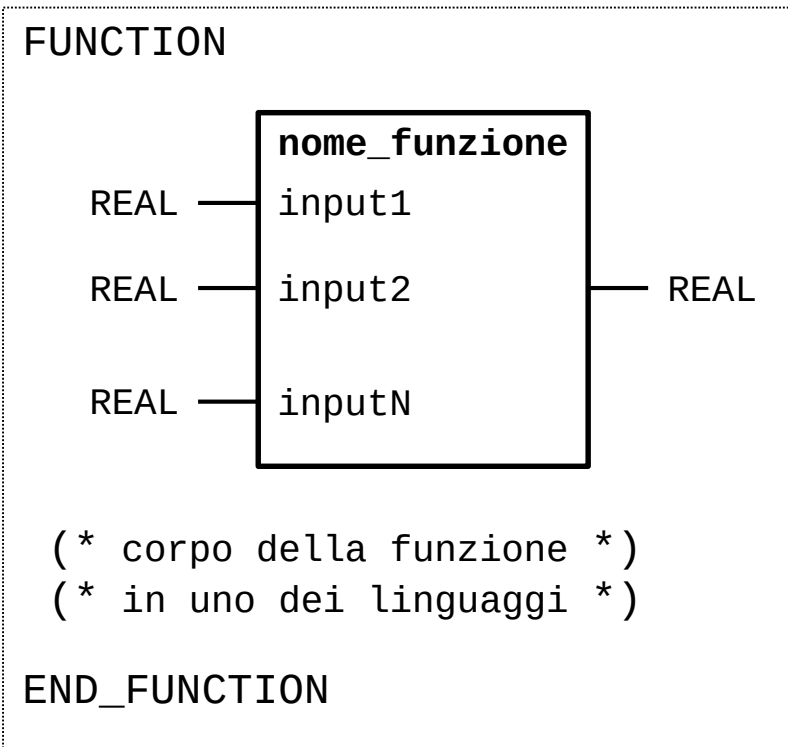


– Definizione in forma grafica

- Si ottiene sempre utilizzando le parole chiave

FUNCTION...END_FUNCTION

- La *definizione* dell'interfaccia esterna viene realizzata con un **blocco grafico rettangolare**
 - A sinistra nel blocco si pongono i parametri di ingresso
 - A destra nel blocco si pongono i parametri di uscita (uno nel caso di funzioni)
- Vanno quindi dichiarate le *variabili* interne e scritto il *corpo* in uno dei linguaggi possibili



unità di organizzazione della programmazione

Funzioni



- L'uso di funzioni nei linguaggi grafici (linguaggio a contatti o diagramma a blocchi funzionali) prevede:
 - **Ingresso implicito booleano EN**
 - Consente di abilitare l'esecuzione della funzione
 - **Uscita implicita booleana ENO**
 - Diventa vera quando la funzione è stata eseguita senza errori
- Utilizzo tipico nelle *catene di funzioni*
 - Bisogna assicurarsi che il blocco precedente abbia terminato prima di iniziare l'elaborazione del blocco corrente
 - È sufficiente collegare l'uscita **ENO** della prima all'ingresso **EN** della seconda

unità di organizzazione della programmazione

Funzioni



– Sovraccarico

- Alcune funzioni possono essere *sovraccaricate*
 - possono essere usate con variabili di ingresso di tipo diverso
 - *Esempi*: funzioni numeriche tipo addizione, sottrazione, ecc.

– Estensione

- Alcune funzioni possono essere *estendibili*
 - possono avere un numero di ingressi non specificato
 - *Esempi*: funzioni numeriche tipo addizione, sottrazione, ecc.

– Una funzione non può richiamare programmi o blocchi funzionali, né se stessa (non è consentita la *ricorsione*)



unità di organizzazione della programmazione

Funzioni predefinite

- Lo standard prevede alcune **funzioni predefinite**
 - Funzioni di *conversione di tipo di variabile*
 - Sintassi generica nella forma `TIP01_TO_TIP02(variabile)`
 - *Esempio*: `y := INT_TO_REAL(x)`
 - Funzione di *troncamento* TRUNC
 - Funzioni *numeriche*
 - Valore assoluto ABS
 - Radice quadrata SQRT
 - Logaritmo naturale LN e decimale LOG
 - Esponenziale naturale EXP
 - Seno SIN e arcoseno ASIN
 - Coseno COS e arcoseno ACOS
 - Tangente TAN e arcotangente ATAN

unità di organizzazione della programmazione

Funzioni predefinite



- Funzioni *aritmetiche standard*, applicabili a variabili numeriche generiche (ANY_NUM)
 - Addizione ADD o + e moltiplicazione MUL o * (estendibili)
 - Sottrazione SUB o – e divisione DIV o /
 - Divisione intera MOD
 - Assegnazione MOVE o :=
 - Elevamento a potenza EXPT o **
- Funzioni di *selezione*, sovraccaricate a tutti i tipi di dati
 - Selezione tra due SEL
 - Multiplexer (selezione tra molti) MUX
 - Massimo estendibile MAX
 - Minimo estendibile MIN
 - Saturazione LIM



unità di organizzazione della programmazione

Funzioni predefinite

- Funzioni di *comparazione*, sovraccaricate ed estendibili
 - Maggiore GT o $>$
 - Maggiore o uguale GE o $>=$
 - Minore LT o $<$
 - Minore o uguale LE o $<=$
 - Uguale EQ o $=$
 - Diverso NE o $<>$, non estendibile

unità di organizzazione della programmazione

Funzioni predefinite



- Funzioni applicabili a *stringhe di bit* (ANY_BIT)
 - Spostamento a sinistra SHL e a destra SHR di N bit con riempimento di zeri
 - Rotazioni circolari di N bit a sinistra ROL e a destra ROR
 - Prodotto logico estendibile AND o &
 - Somma logica estendibile OR o ≥ 1 nei linguaggi grafici
 - OR esclusivo estendibile XOR o $\geq 2K+1$ nei linguaggi grafici
 - Negazione NOT, applicabile solo ad una variabile booleana

unità di organizzazione della programmazione

Funzioni predefinite



- Funzioni sulle *stringhe di caratteri*
 - Lunghezza LEN
 - Estrazione di N caratteri a sinistra LEFT, a destra RIGHT o in mezzo MID
 - Concatenazione estendibile CONCAT
 - Inserzione di una stringa in un'altra INSERT
 - Cancellazione di caratteri da una stringa DELETE
 - Sostituzione di caratteri da una stringa REPLACE
 - Presenza di una sottostringa FIND
- Funzioni relative a operazioni su variabili di tipo temporale

unità di organizzazione della programmazione

Funzioni – Esempi



– Esempio 1

- Definizione funzione utente `soglia_satura` in forma testuale:

```
FUNCTION soglia_satura : REAL
  VAR_INPUT
    dato, lim_soglia, lim_sat : REAL;
  END_VAR
  IF ABS(dato) < lim_soglia THEN
    soglia_satura := 0.0;
  ELSE
    soglia_satura :=
      MIN(MAX(dato, -lim_sat), lim_sat);
  END_IF
END_FUNCTION
```

} Corpo

- *Note:* si noti l'uso del testo strutturato per la definizione del corpo della funzione e l'impiego di funzioni predefinite

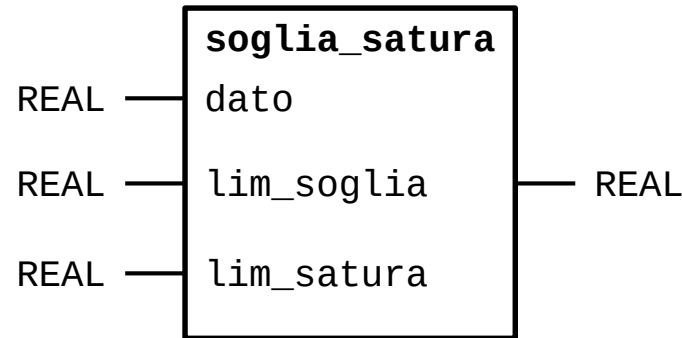
unità di organizzazione della programmazione

Funzioni – Esempi



- Definizione in forma grafica:

FUNCTION



(* corpo della funzione *)
(* in uno dei linguaggi *)

END_FUNCTION

unità di organizzazione della programmazione

Funzioni – Esempi



– Esempio 2

- Definizione funzione `correggi_sensore` in forma testuale:

```
FUNCTION correggi_sensore: WORD (* codifica *)  
                                (* BCD *)  
  
    VAR_INPUT  
        correggi : BOOL;  
        ingresso : WORD; (* codifica BCD *)  
        offset : INT;  
    END_VAR  
    IF correggi THEN  
        correggi_sensore :=  
            INT_TO_BCD(BCD_TO_INT(ingresso) -  
                offset);  
    END_IF  
END_FUNCTION
```

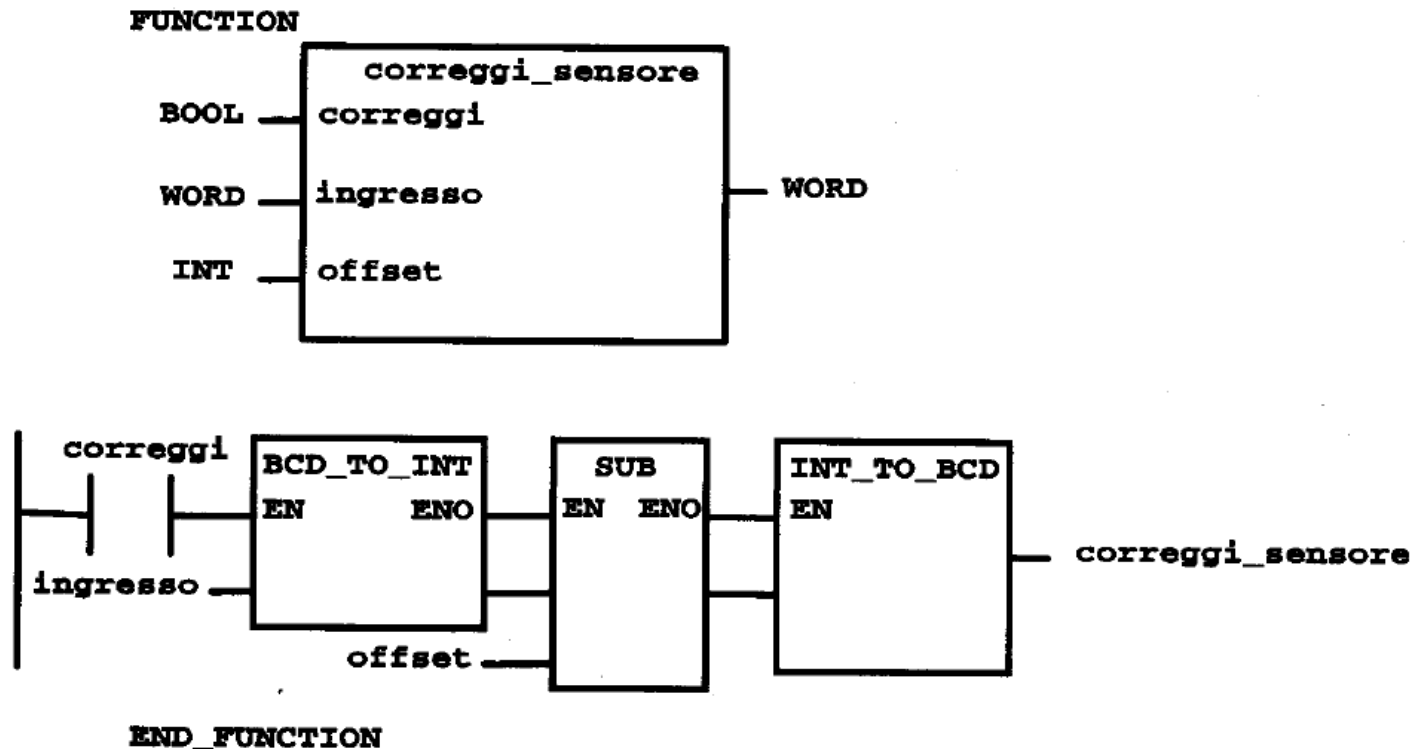
– *Nota*: si noti come il corpo sia espresso in testo strutturato

unità di organizzazione della programmazione

Funzioni – Esempi



- Definizione in forma grafica:



- *Note:* si noti come il corpo sia espresso in linguaggio a contatti e l'uso di funzioni predefinite e delle variabili implicite EN e ENO

unità di organizzazione della programmazione

Blocchi funzionali



- **Blocchi funzionali**

- Sono unità di organizzazione della programmazione nelle quali i valori delle uscite dipendono dai valori degli ingressi e dallo **stato** delle variabili interne
 - **I valori delle variabili interne e di uscita persistono tra due esecuzioni successive del blocco funzionale**
- Un blocco funzionale può contenere la chiamata ad altri blocchi funzionali o funzioni
- Non è consentita la *ricorsione*
 - un blocco non può chiamare se stesso

unità di organizzazione della programmazione

Blocchi funzionali



– Definizione in forma testuale

FUNCTION_BLOCK esempio

VAR_INPUT

...; (* definizione delle variabili di ingresso *)

END_VAR

VAR_OUTPUT

...; (* definizione delle variabili di uscita *)

END_VAR

...

(* altre definizioni di variabili *)

...

(* corpo del blocco funzionale *)

END_FUNCTION_BLOCK

- *Nota:* il corpo può essere scritto in uno dei linguaggio previsti dello standard, compreso il Sequential Functional Chart



unità di organizzazione della programmazione

Blocchi funzionali

– Definizione in forma grafica

- Si ottiene utilizzando le parole chiave
`FUNCTION_BLOCK...END_FUNCTION_BLOCK`
- L'interfaccia esterna viene definita mediante un blocco rettangolare
 - A sinistra si dispongono i parametri di ingresso
 - A destra si dispongono i parametri di uscita
- Si provvede quindi alla definizione delle variabili interne e alla scrittura del corpo in uno dei linguaggi dello standard

unità di organizzazione della programmazione

Blocchi funzionali



– Istanza

- Per poter utilizzare i blocchi funzionali, all'atto della loro definizione viene creata una *istanza* del blocco
 - Nome identificativo e struttura dati univoca
 - L'utilizzatore può accedere solo alle variabili di ingresso e di uscita
 - » Le variabili interne non sono accessibili
- La definizione di una istanza avviene in modo analogo alla definizione delle variabili
 - Corrisponde a tutti gli effetti alla definizione di una struttura dati
- La visibilità di una istanza ha le stesse proprietà di quella di una comune variabile
- Una istanza può essere data in ingresso a un'altra POU proprio come le variabili comuni

unità di organizzazione della programmazione

Blocchi funzionali



- *Esempio*: istanze del blocco funzionale esempio

VAR

istanza_1, istanza_2 : esempio;

END_VAR

- È possibile utilizzare l'attributo RETAIN (VAR RETAIN) per indicare che i valori devono essere conservati in caso di mancanza di alimentazione
- È possibile utilizzare il costrutto VAR_GLOBAL per indicare che la visibilità dell'istanza del blocco funzionale deve avere una visibilità globale

unità di organizzazione della programmazione

Blocchi funzionali predefiniti



– Data la loro importanza, lo standard prevede alcuni **blocchi funzionali predefiniti**

- Flip-Flop SET/RESET

- I flip-flop consentono di memorizzare una condizione logica
- Versione con priorità del SET e corpo in testo strutturato:

```
FUNCTION_BLOCK SR (* priorità del SET *)  
  VAR_INPUT  
    S1, R : BOOL;  
  END_VAR  
  VAR_OUTPUT  
    Q1 : BOOL;  
  END_VAR  
  Q1 := S1 OR (Q1 AND NOT(R));  
END_FUNCTION_BLOCK
```

unità di organizzazione della programmazione

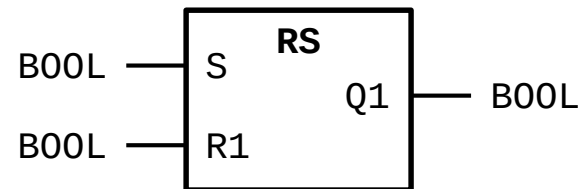
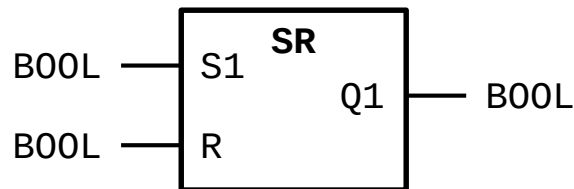
Blocchi funzionali predefiniti



- Versione con priorità del RESET e corpo in testo strutturato:

```
FUNCTION_BLOCK RS (* priorità del RESET *)  
  VAR_INPUT  
    S, R1 : BOOL;  
  END_VAR  
  VAR_OUTPUT  
    Q1 : BOOL;  
  END_VAR  
  Q1 := NOT(R1) AND (Q1 OR S);  
END_FUNCTION_BLOCK
```

- Definizione grafia dei blocchi funzionali SR e RS :

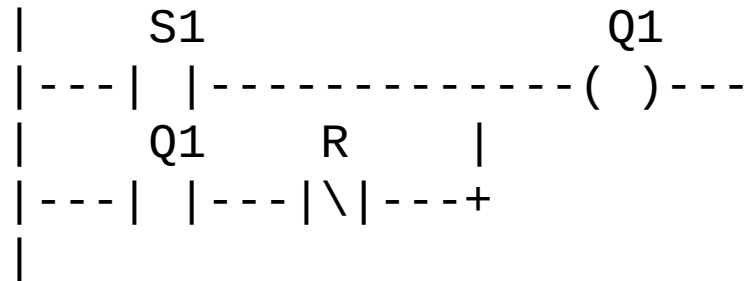




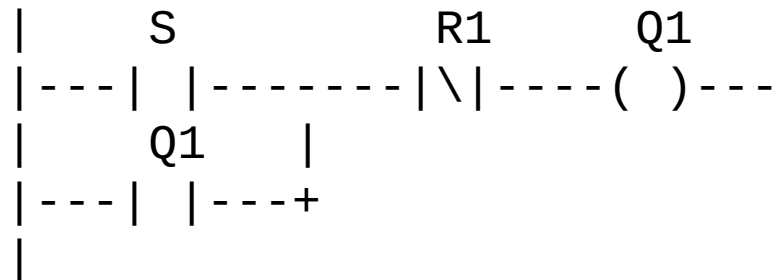
unità di organizzazione della programmazione

Blocchi funzionali predefiniti

- Corpo del flip-flop SR in linguaggio a contatti:



- Corpo del flip-flop RS in linguaggio a contatti:



unità di organizzazione della programmazione

Blocchi funzionali predefiniti



- Rilevatori di fronte

- Rilevatore del fronte di salita con corpo in lista istruzioni:

```
FUNCTION_BLOCK R_TRIG (* fronte di salita *)
  VAR_INPUT
    CLK : BOOL;
  END_VAR
  VAR_OUTPUT
    Q : BOOL;
  END_VAR
  VAR RETAIN
    AUX : BOOL := 0;
  END_VAR
  LD      CLK      (* corpo in lista istruzioni *)
  ANDN    AUX
  ST      Q
  LD      CLK
  ST      AUX
END_FUNCTION_BLOCK
```

unità di organizzazione della programmazione

Blocchi funzionali predefiniti



- Rilevatore del fronte di discesa con corpo in linguaggio a contatti:

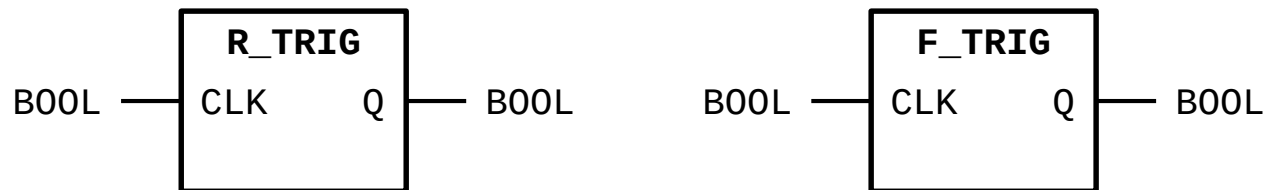
```
FUNCTION_BLOCK F_TRIG (* fronte di discesa *)
  VAR_INPUT
    CLK : BOOL;
  END_VAR
  VAR_OUTPUT
    Q : BOOL;
  END_VAR
  VAR RETAIN
    AUX : BOOL := 1;
  END_VAR
  |
  |                                     (* corpo in linguaggio a contatti *)
  |   CLK           AUX           Q
  |---|\|---+---|\|---( )---
  |               |               AUX
  |               +------( )---
  |
END_FUNCTION_BLOCK
```

unità di organizzazione della programmazione

Blocchi funzionali predefiniti



- *Nota*: in risposta alla prima chiamata si può avere il riconoscimento di un fronte se la variabile di ingresso è alta o bassa, rispettivamente
- *Nota*: il valore della variabile ausiliaria è mantenuto anche in assenza di alimentazione
- Definizione grafica dei blocchi funzionali R_TRIG e F_TRIG



- I riconoscitori di fronte possono essere **utilizzati implicitamente** sulle variabili di ingresso di altri blocchi funzionali:
 - » Le variabili vanno definite con il qualificatore R_TRIG (fronte di salita) e F_TRIG (fronte di discesa)
 - » Per la definizione grafica sulle linee di ingresso si possono usare i **simboli grafici** > (fronte di salita) e < (fronte di discesa)

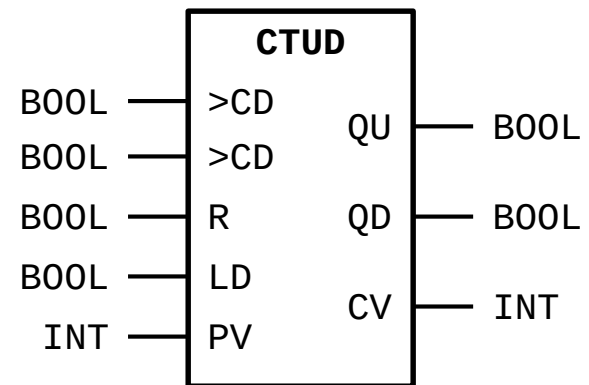
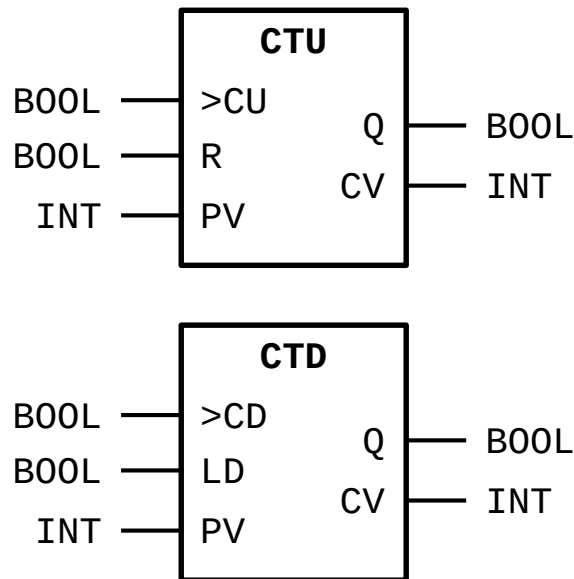
unità di organizzazione della programmazione

Blocchi funzionali predefiniti



- Contatori

- I contatori sono blocchi funzionali che segnalano il raggiungimento di un valore di conteggio
- Esistono il contatore a *incremento* (CTU), a *decremento* (CTD) e il *bidirezionale* (CTUD)
- Definizioni grafiche (si noti l'uso di *variabili implicite a fronte*)



unità di organizzazione della programmazione

Blocchi funzionali predefiniti



- Significato delle variabili di ingresso e di uscita:
 - » CU ingresso sui cui fronti di salita si incrementa il contatore
 - » CD ingresso sui cui fronti di salita si decrementa il contatore
 - » R è il reset (porta a zero il valore di conteggio)
 - » PV è il valore di conteggio di ingresso
 - » LD carica il contatore con il valore PV
 - » Q segnala che il contatore ha finito il conteggio
 - Ha raggiunto il valore di conteggio nel caso di contatore a incremento
 - Ha raggiunto il valore zero nel caso di contatore a decremento
 - » QD segnala che il contatore bidirezionale ha raggiunto lo zero
 - » QU segnala che il contatore bidirezionale ha raggiunto il valore PV
 - » CV è il valore raggiunto dal contatore

unità di organizzazione della programmazione

Blocchi funzionali predefiniti



- Comportamento del contatore a incremento che viene impostato al valore PV:
 - » Ad ogni fronte di salita CU incrementa il valore CV partendo da zero
 - » Quando arriva il valore impostato (PV) mette l'uscita Q a 1 e **non conta più**
 - » Il contatore può essere resettato con l'ingresso R
 - » Definizione del contatore a incremento:

```
FUNCTION_BLOCK CTU (* contatore a incremento *)  
  VAR_INPUT  
    CU := BOOL R_TRIG; (* definizione implicita *)  
                                (* del fronte *)  
    R := BOOL;  
    PV := INT;  
  END_VAR
```

unità di organizzazione della programmazione

Blocchi funzionali predefiniti



```
VAR_OUTPUT
  Q := BOOL;
  CV := INT;
END_VAR
IF R THEN
  CV := 0;
ELSEIF CU AND (CV<PV) THEN
  CV := CV + 1;
END_IF
Q := (CV = PV);
END_FUNCTION_BLOCK
```


unità di organizzazione della programmazione

Blocchi funzionali predefiniti



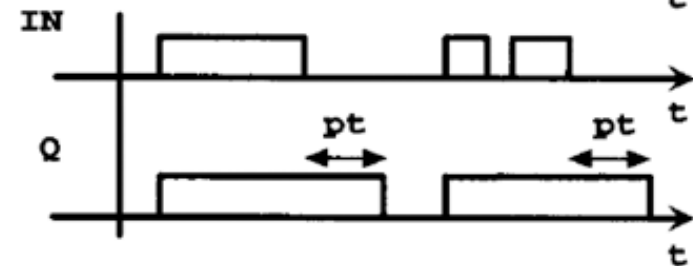
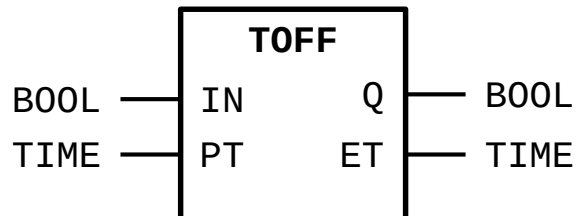
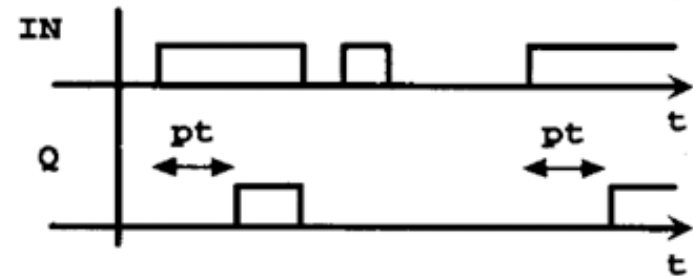
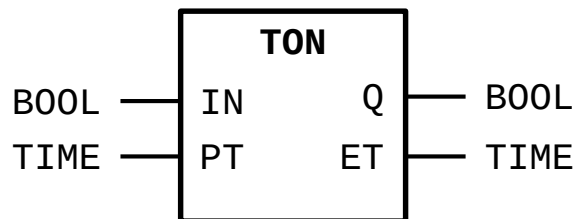
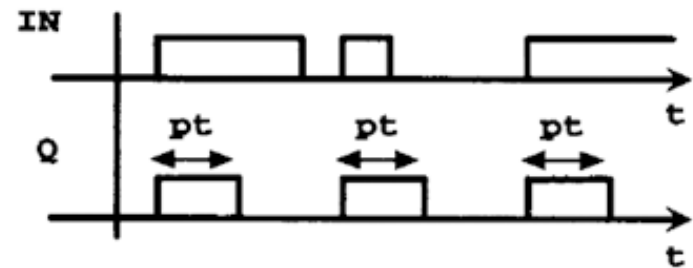
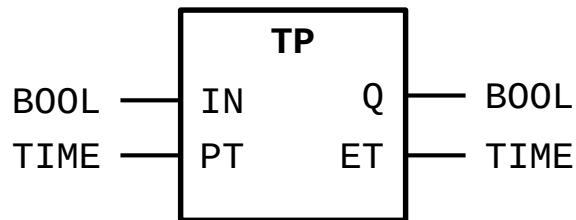
- Nella particolare implementazione deve essere previsto un valore massimo per il valore di conteggio PV
 - » Il limite può essere superato ponendo i **contatori in cascata**
 - » *Esempio*: contatori a incremento in cascata:
 - Uscita QU del primo collegata all'ingresso CU del secondo e come ingresso di reset R del primo
 - L'uscita QU del secondo diventerà alta quando i fronti di salita sull'ingresso CU del primo avranno raggiunto un numero pari al prodotto dei valori di conteggio PV

unità di organizzazione della programmazione

Blocchi funzionali predefiniti



- Timer (temporizzatori)
 - Definizione grafica dei *timer ad impulsi* (TP), *on-delay* (TON) e *off-delay* (TOF) e relativi cronogrammi:





unità di organizzazione della programmazione

Blocchi funzionali predefiniti

- Significato delle variabili di ingresso e di uscita:
 - » IN ingresso che fa partire le funzioni di temporizzazione
 - » PT è il valore di tempo da contare
 - » Q è l'uscita booleana su cui il temporizzatore agisce
 - » ET è il tempo trascorso
- *Timer ad impulsi*
 - » Genera una finestra rettangolare di durata prestabilita sulla sua uscita quando l'ingresso diventa vero
- *Timer on-delay* (ritardo di accensione)
 - » Alza la sua uscita dopo che è trascorso il tempo impostato da quando l'ingresso è diventato alto
 - » Se l'ingresso si abbassa, il timer si resetta
- *Timer off-delay* (ritardo di spegnimento)
 - » L'uscita diventa alta quando l'ingresso è vero e si abbassa dopo il tempo impostato quando l'ingresso diventa falso

unità di organizzazione della programmazione

Blocchi funzionali predefiniti



- Il massimo limite di conteggio temporale può essere superato:
 - » **Timer in cascata**
 - Il valore di conteggio risultante sarà pari alla somma dei singoli tempi di conteggio
 - » **Contatore in cascata ad un timer**
 - Il valore di conteggio risultante sarà pari al prodotto del valore impostato per il temporizzatore e il valore impostato per il conteggio



unità di organizzazione della programmazione

Blocchi funzionali definiti dall'utente

– *Esempio*: blocco funzionale definito dall'utente per la gestione di una risorsa comune (semaforo)

- Segnala se una risorsa è già occupata
- Definizione testuale, con corpo in testo strutturato:

```
FUNCTION_BLOCK semaforo
  VAR_INPUT
    richiesta, rilascio : BOOL;
  END_VAR
  VAR_OUTPUT
    impegnata : BOOL;
  END_VAR
  VAR
    aux : BOOL := 0;
  END_VAR
```



unità di organizzazione della programmazione

Blocchi funzionali definiti dall'utente

```
    impegnata := aux;  
    IF richiesta THEN  
        aux := 1;  
    ELSEIF rilascio THEN  
        impegnata := 0;  
        aux := 0;  
    END_IF  
END_FUNCTION_BLOCK
```

- Definizione grafica: →

FUNCTION_BLOCK



```
VAR  
    aux : BOOL := 0;  
END_VAR
```

(* corpo in uno dei linguaggi *)

END_FUNCTION_BLOCK



unità di organizzazione della programmazione

Blocchi funzionali definiti dall'utente

- *Esempio*: blocco funzionale definito dall'utente per la generazione di un'**onda quadra** su una variabile booleana quando un'altra variabile booleana è vera

- Si assume come valore iniziale il valore falso
- Definizione testuale:

```
FUNCTION_BLOCK ondaquadra
  VAR_INPUT
    abilita : BOOL;
    durata_ON, durata_OFF : TIME;
  END_VAR
  VAR_OUTPUT
    uscita : BOOL;
  END_VAR
  VAR
    timer1, timer2 := TON; (* sono usate due istanze *)
  END_VAR
  (* di timer on-delay *)
```



```

...
(* corpo del blocco funzionale *)
...
END_FUNCTION_BLOCK

```

-



unità di organizzazione della programmazione

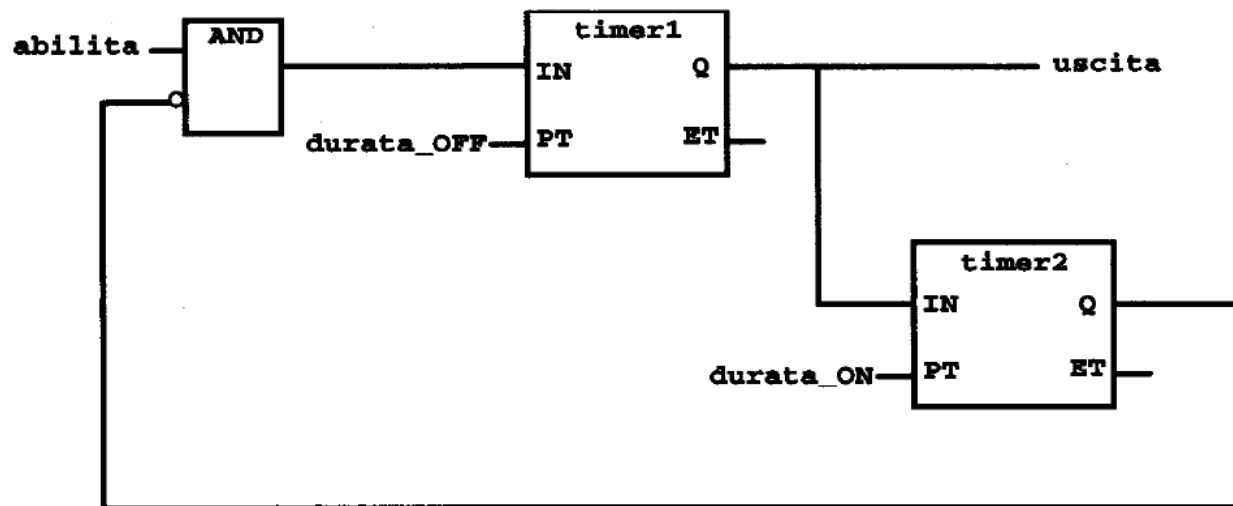
Blocchi funzionali definiti dall'utente

- Corpo in testo strutturato:

```
timer1(IN := NOT(timer2.Q) AND abilita, PT :=  
        durata_OFF);  
timer2(IN := timer1.Q, PT := durata_ON);  
uscita := timer1.Q;
```

- *Nota:* nel testo strutturato sono effettuate due chiamate ai blocchi funzionali di temporizzazione

- Corpo in diagramma a blocchi funzionali:





unità di organizzazione della programmazione

Blocchi funzionali definiti dall'utente

- Corpo in lista istruzioni:

```
LDN    timer2.Q
AND    abilita
ST     timer1.IN
LD     durata_OFF
ST     timer1.PT
CAL    timer1
CAL    timer2(IN := timer1.Q, PT := durata_OFF)
LD     timer1.Q
ST     uscita
```

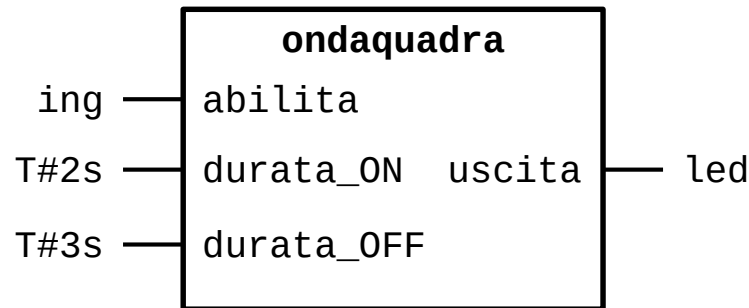
- *Nota:* sono stati utilizzati due diversi modi per assegnare i parametri ai blocchi funzionali



unità di organizzazione della programmazione

Blocchi funzionali definiti dall'utente

- Chiamata del blocco ondaquadra in testo strutturato:
`ondaquadra(abilita := ing, durata_ON := T#2s,
 durata_OFF := T#3s, uscita := led);`
- Chiamata del blocco in forma grafica:





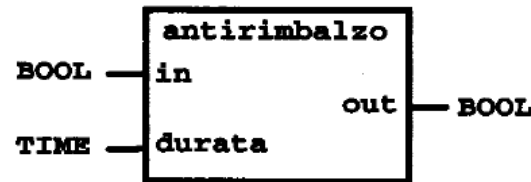
unità di organizzazione della programmazione

Blocchi funzionali definiti dall'utente

– Esempio: antirimbalzo

- Si vuole valutare il valore (vero o falso) di una variabile booleana solo se è rimasto costante per un certo tempo
- Definizione grafica con corpo in linguaggio a contatti:

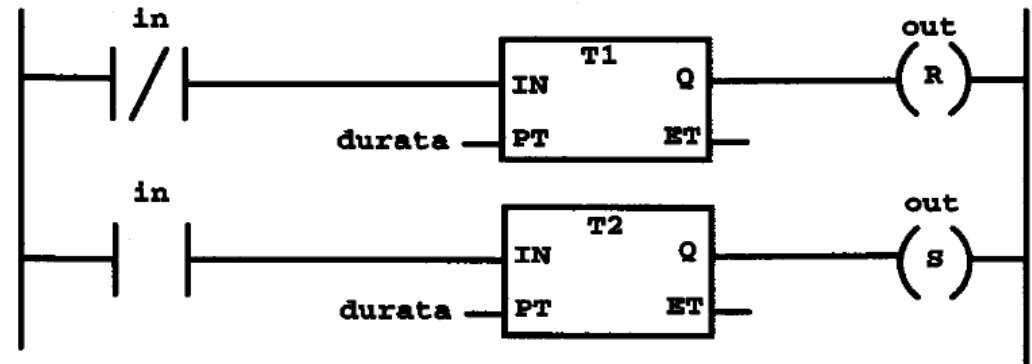
FUNCTION_BLOCK



VAR

T1, T2: TON;

END_VAR



END_FUNCTION_BLOCK

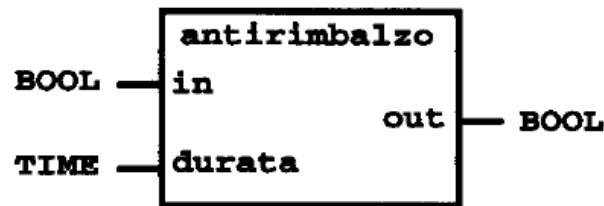
unità di organizzazione della programmazione

Blocchi funzionali definiti dall'utente



- Definizione grafica con corpo in diagramma a blocchi funzionale:

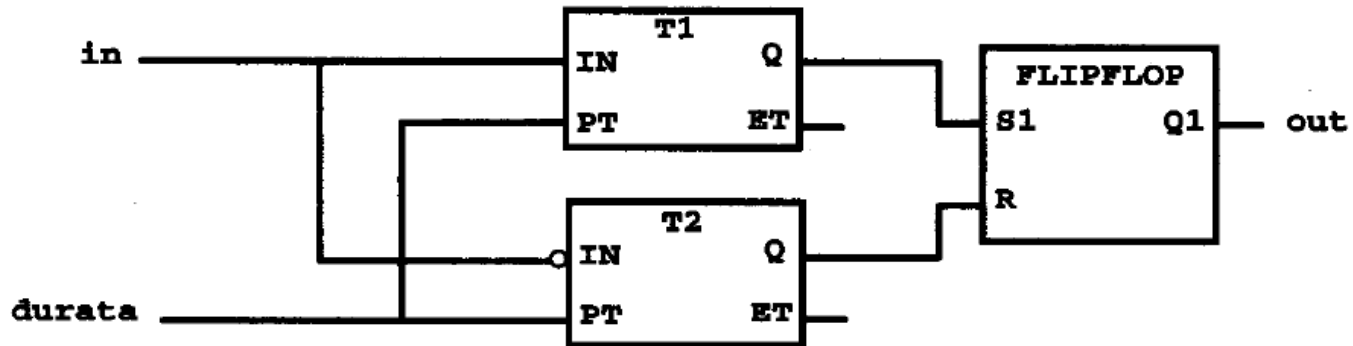
FUNCTION_BLOCK



VAR

T1,T2: TON; FLIPFLOP:SR;

END_VAR



END_FUNCTION_BLOCK

unità di organizzazione della programmazione

Programmi



- **Programmi**

- Rappresenta l'insieme logico di elementi e costrutti dei linguaggi di programmazione necessari per il controllo di una macchina o di un processo
- Può essere visto come un grande blocco funzionale con alcune caratteristiche fondamentali:
 - Può avere definizioni di variabili direttamente rappresentate in memoria → consente di adattarsi al dispositivo
 - Può avere definizioni di variabili globali, che saranno accessibili da blocchi funzionali interni tramite la definizione `VAR_EXTERNAL`
 - Può definire variabili di accesso indirizzabili da altri programmi remoti tramite la definizione `VAR_ACCESS`



unità di organizzazione della programmazione

Programmi

- Non può contenere una istanza di se stesso
 - Istanze di programmi possono essere dichiarate solo a livello di risorsa
- I programmi **sono le uniche POU che possono accedere alle variabili rappresentative degli ingressi e delle uscite fisiche** del dispositivo
- I programmi potranno poi passare le variabili rappresentative degli ingressi e delle uscite fisiche alle altre POU in essa contenute

unità di organizzazione della programmazione

Programmi



– Definizione

```
PROGRAM nome_programma
  VAR_INPUT
    ...;
    (* definizione delle variabili di ingresso *)
  END_VAR
  VAR_OUTPUT
    ...;
    (* definizione delle variabili di uscita *)
  END_VAR
  ...
  (* altre definizioni di variabili *)
  ...
  (* corpo del programma *)
END_PROGRAM
```

- Il corpo può essere scritto in uno dei linguaggi dello standard

unità di organizzazione della programmazione

Compiti



- **Compiti** (*task*)
 - È un elemento capace di far eseguire un programma o un blocco funzionale in maniera periodica o al verificarsi di eventi
 - Per definire un compito si possono utilizzare:
 - Il parametro SINGLE di tipo BOOL
 - Indica la variabile booleana il cui fronte di salita rappresenta il verificarsi dell'evento che causa un'**unica esecuzione** del programma o del blocco funzionale
 - Il parametro INTERVAL di tipo TIME
 - Indica la durata temporale del ciclo per un compito di tipo **periodico**
 - In caso di mancato uso di uno dei due parametri precedenti si definisce un compito **ciclico continuo**

unità di organizzazione della programmazione

Compiti



- Il parametro **PRIORITY** di tipo **UINT**
 - Indica la priorità del compito utilizzato dallo schedatore del dispositivo per gestire i vari compiti
 - **Il valore 0 indica la priorità massima**
 - Se non viene definito la priorità è quella minima
- L'algoritmo di schedulazione dipende dal particolare dispositivo
 - Può essere *pre-emptive* (task a priorità minore possono essere interrotti da quelli a priorità maggiore) oppure *non pre-emptive*
 - Normalmente per i dispositivi di controllo è richiesta la disponibilità di uno schedatore pre-emptive

unità di organizzazione della programmazione

Compiti



- *Esempi:*

```
TASK ciclo_controllo(INTERVAL:=T#20ms,PRIORITY:=0);  
TASK allarme(SINGLE:=flag_allarme,PRIORITY:=2);  
TASK background;  
TASK ciclo_refresh(INTERVAL:=T#10s,PRIORITY:=5);
```

dove:

- `ciclo_controllo` definisce un compito periodico da eseguire ogni 20 ms a priorità massima
- `allarme` definisce un compito che va eseguito solo una volta quando la variabile booleana `flag_allarme` passa dal valore falso a quello vero
- `background` definisce un compito ciclico alla minima priorità
- `ciclo_refresh` definisce un compito periodico da eseguire ogni 10 ms

unità di organizzazione della programmazione

Compiti



- Per assegnare un programma o un blocco funzionale a un particolare compito (task) si usa la parola chiave **WITH**
 - In maniera grafica l'assegnazione al compito si ottiene scrivendo il nome del compito (task) all'interno del blocco
 - I programmi e i blocchi funzionali non esplicitamente associati ad un compito (task) vengono eseguiti in maniera ciclica continua con la minima priorità

unità di organizzazione della programmazione

Risorse



- **Risorse** (*resource*)

- Corrispondono ad entità capaci di eseguire programmi
- La definizione inizia con la parola chiave **RESOURCE**, seguita dal nome identificativo, dalla parola chiave **ON** e dal tipo di processore su cui dovrà essere caricata, e termina con la parola chiave **END_RESOURCE**
 - Vanno definite eventuali variabili globali, variabili ad accesso remoto, compiti e programmi che la compongono
 - Nel definire i programmi si assegnano anche le loro variabili di ingresso e di uscita
 - Vanno collegati agli indirizzi di memoria corrispondenti a ingressi e uscite fisiche

unità di organizzazione della programmazione

Risorse



- A questo livello viene creata una **istanza dei programmi**
 - Una istanza viene definita come per i blocchi funzionali
 - Viene collegata alle variabili su cui deve operare
 - Viene collegata ad un compito (task)→ diventa un programma *run-time*
- La definizione della risorsa **non prevede la scrittura di istruzioni** ma semplicemente la **definizione di relazioni** tra gli elementi menzionati

unità di organizzazione della programmazione

Configurazione



- **Configurazione**

- Corrisponde alla definizione di tutto il software che deve essere caricato in un dispositivo di controllo
 - Contiene risorse, variabili globali e percorsi di accesso
- Lo scopo della configurazione è:
 - Definire il tipo di risorse elaborative a disposizione
 - Allocare i programmi sulle risorse
 - Indicare la modalità e la priorità di esecuzione dei programmi
 - Definire le variabili globali accessibili da tutti i programmi

unità di organizzazione della programmazione

Configurazione



- La definizione prevede l'uso della parola chiave **CONFIGURATION** con il nome assegnato
 - Vanno definite le variabili globali
 - Saranno visibili da tutti gli elementi che compongono la configurazione
 - Vanno definite le variabili ad accesso remoto
 - Si definiscono le risorse che compongono la configurazione
 - Sono rappresentative dei processi a disposizione
 - Contengono i compiti e i programmi da assegnare a ogni processore
 - La definizione viene chiusa dalla parola chiave **END_CONFIGURATION**



unità di organizzazione della programmazione

Configurazione

– Esempio:

```
CONFIGURATION controllore_cella
  VAR_GLOBAL
    A : REAL; (* queste variabili sono visibili in *)
    B : REAL; (* tutta la configurazione *)
  END_VAR
  RESOURCE processore_1 ON Pentium
    VAR_GLOBAL
      flag_allarme : BOOL; (* visibile in tutta la *)
                          (* risorsa *)
    END_VAR
    TASK ciclo(INTERVAL:=T#20ms,PRIORITY:=0);
    TASK allarme(SINGLE:=flag_allarme,PRIORITY:=2);
    PROGRAM tornio : controllo_macchina(IN1:=...,IN2:=...,...,
      OUT1:=...,OUT2:=...,...) WITH ciclo;
    PROGRAM all_tornio : gestione_allarmi(IN1:=...,IN2:=...,
      ...,OUT1:=...,OUT2:=...,...) WITH allarme;
  END_RESOURCE
```

unità di organizzazione della programmazione

Configurazione



```
RESOURCE processore_2 ON AB486
  PROGRAM cella : coordinamento (IN1:=..., IN2:=..., ...,
    OUT1:=..., OUT2:=..., ...) ;
END_RESOURCE
END_CONFIGURATION
```

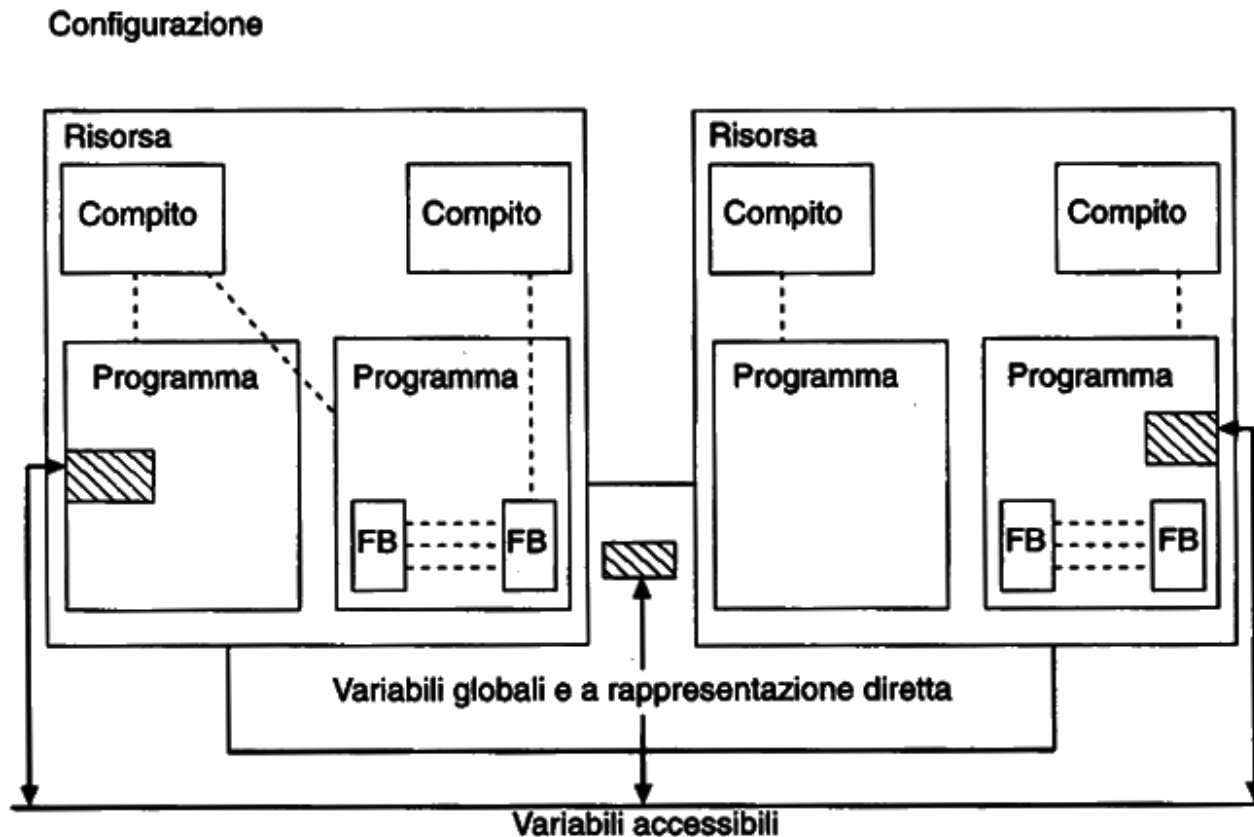
dove si è supposto che il dispositivo abbia due processori,
corrispondenti alle risorse, individuati dalle sigle Pentium e AB486



unità di organizzazione della programmazione

Relazioni tra le POU

- Relazioni tra le PUO





unità di organizzazione della programmazione

Relazioni tra le POU

– *Note:*

- È possibile eseguire un blocco funzionale con criteri specificati da un compito (task) diverso dal quello del programma che contiene il blocco
- È possibile definire variabili accessibili da parte di programmi remoti

diagramma funzionale sequenziale

Introduzione



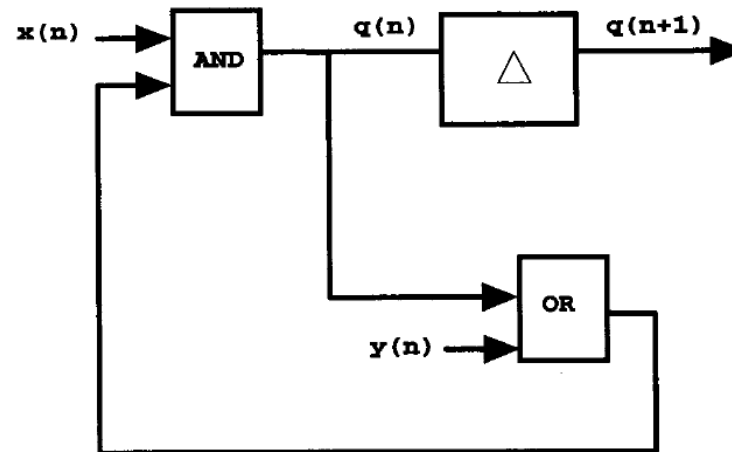
- **Sequential functional chart (SFC)**
 - Linguaggio innovativo adatto per la scrittura di algoritmi per il **controllo logico/sequenziale**
 - Nasce come risultato di una apposita commissione istituita nel 1975 in Francia
 - *Scopo*: cercare un mezzo per descrivere i sistemi complessi di automazione industriale ad *eventi discreti*
 - **Sistema ad eventi discreti** sono dotati di uno spazio di stato discreto, non continuo, e la cui evoluzione dipende dal verificarsi o no di condizioni particolari, gli *eventi*
 - Tecniche utilizzate in precedenza:
 - Equazioni descrittive del comportamento degli *automi a stati finiti*



diagramma funzionale sequenziale

Introduzione

- Rappresentazioni circuitali di tali equazioni



- I grafi di evoluzione dello stato

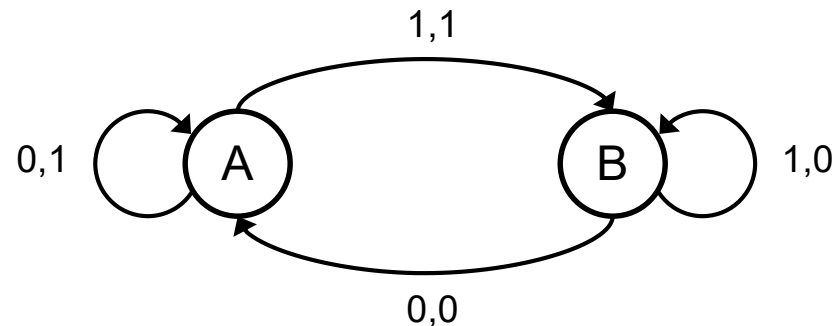




diagramma funzionale sequenziale

Introduzione

- Le descrizioni testuali del comportamento
- Sistemi di varia natura sviluppati autonomamente da aziende del settore
- *Risultato*: definizione del *GRAPHe de Coordination Etapes-Transitions* (**GRAFCET**)
 - Adottato dal CEI nel 1988 nello standard internazionale 848 come linguaggio per la descrizione di sistemi di automazione industriale
 - Si tratta di una semplificazione delle *Reti di Petri*, uno strumento grafico più generale per la rappresentazione e l'analisi di sistemi a eventi discreti
 - **La programmazione di un sistema di controllo per l'automazione industriale coincide con il descriverne il comportamento desiderato**
 - Il GRAFCET è stato incluso con il nome di **Sequential Functional Chart (SFC)** tra i linguaggi di programmazione dello standard 61131-3



diagramma funzionale sequenziale

Introduzione

- Non è un linguaggio uguale agli altri nelle caratteristiche e negli scopi
 - » Il suo utilizzo principale è la programmazione di algoritmi di controllo logico/sequenziale
 - » Consente una semplice segmentazione degli algoritmi di controllo in **fasi** e **transizioni** tra le fasi
 - » Il comportamento di un algoritmo scritto in SFC **dipende dallo stato precedente** → può essere utilizzato per la programmazione di blocchi funzionali e di programmi, non di funzioni
- Elementi di base del Sequential Functional Chart:
 - **Fase** (o tappa o passo) con le eventuali **azioni** associate
 - **Transizione** con la **condizione** associata
 - **Arco orientato** che connette fasi e transizioni



diagramma funzionale sequenziale

La fase

– La fase

- *Definizione*: **condizione invariante del sistema**, modificabile solo dall'occorrenza di un determinato evento, il quale genera una transizione che porta il sistema in una nuova fase
- Una fase può essere, in un determinato istante di tempo, **attiva** o **inattiva**
- Si definisce **condizione** dell'SFC l'insieme delle fasi attive
- Se una fase è attiva il comportamento del blocco funzionale o del programma descritto tramite l'SFC è definito dalle azioni associate a quella fase
- *Rappresentazione grafica*: una fase è rappresentata da un **rettangolo** al cui interno viene scritto il *nome* della fase
 - Il nome deve essere univoco nel contesto in cui è definita la fase



diagramma funzionale sequenziale

La fase

- Il rettangolo può essere connesso ad altri elementi, le transizioni, attraverso connettori sul bordo superiore o inferiore
- *Definizione testuale*: si utilizza il costrutto
`STEP nome_fase: ... END_STEP`
dove al posto dei puntini vanno poste le azioni associate alla fase
- *Fasi iniziali*: sono le fasi attivate all'inizio dell'esecuzione
 - Si identificano graficamente mediante due linee verticali poste nel rettangolo
 - La definizione testuale avviene con il costrutto
`INITIAL_STEP nome_fase: ... END_STEP`
 - Lo standard richiede che ogni grafo SFC debba avere **una sola fase iniziale**
- **Un blocco funzione o un programma possono essere costituiti da più grafi SFC non connessi**



diagramma funzionale sequenziale

La fase – Variabile segnalatrice (Marker)

- **Variabile segnalatrice o marker della fase**

- La definizione di una fase con il nome `nome_fase` implica la definizione della variabile booleana `nome_fase.X` ad essa associata, la quale assumerà i valori logici:
 - $\left\{ \begin{array}{ll} \mathbf{1} & \text{se la fase è attiva} \\ \mathbf{0} & \text{se la fase è inattiva} \end{array} \right.$
- Il valore di questa variabile è disponibile, per la connessione grafica, sulla destra del rettangolo che rappresenta la fase
- La variabile segnalatrice viene inizializzata a 1 per le fasi iniziali e al valore 0 per tutte le altre fasi
- Per indicare graficamente le fasi attive si può disegnare un pallino all'interno della fase



diagramma funzionale sequenziale

La fase – Variabile timer

- **Variabile timer**
 - La definizione di una fase con il nome `nome_fase` implica la definizione della variabile di tipo `TIME` dal nome `nome_fase.T` che rappresenta la durata dell'ultima attivazione della fase:
 - » Fase attiva → rappresenta il tempo trascorso dalla sua attivazione
 - » Fase inattiva → rappresenta la durata della sua ultima attivazione
 - Questa variabile è inizializzata al valore `T#0s` per tutte le fasi
- Le variabili segnalatrici e timer non possono essere modificate dall'utente ma solo utilizzate
 - Devono **essere variabili in sola lettura** per l'utente
 - Sono **variabili locali** all'unità di programmazione in cui l'SFC è definito

diagramma funzionale sequenziale

La transizione



– La transizione

- Viene indicata graficamente con una barretta trasversale posta sull'arco orientato
- Rappresenta la condizione che potrebbe far cambiare lo stato delle fasi attive
- Ad ogni transizione deve essere associata la sua **condizione** espressa come funzione booleana di variabili booleane, di equazioni che si riportano a valori booleani o di asserzioni
- Lo standard prevede varie possibilità per la definizione delle condizioni associate alle transizioni:
 - Espressione in testo strutturato posta alla destra della barretta
 - Rete in linguaggio a contatti posta alla sinistra della barretta oppure a essa collegata tramite un connettore



diagramma funzionale sequenziale

La transizione

- Rete in diagramma a blocchi funzionali posta alla sinistra della barretta oppure a essa collegata tramite un connettore
- Assegnando un nome alla transizione posto alla destra della barretta e definendola a parte con il costrutto:

```
TRANSITION nome_transizione  
    := ... ; (* condizione associata *)  
END_TRANSITION
```

- Utilizzando il costrutto

```
TRANSITION FROM fasi_precedenti  
    TO fasi_successive  
    := ... ; (* condizione associata *)  
END_TRANSITION
```

che implicitamente definisce anche la topologia del grafo indicando come le fasi sono collegate alle transizioni

diagramma funzionale sequenziale

Gli archi orientati



– Gli archi orientati

- Gli archi orientati collegano tra loro le fasi stabilendone la sequenza
- Sono interrotte dalle barrette delle transizioni che determinano le condizioni da soddisfare affinché si possano avere le attivazioni e le disattivazioni delle fasi
- L'orientamento degli archi va sempre dal bordo inferiore di una o più fasi al bordo superiore di una o più fasi
- Le frecce sugli archi non sono indispensabili ma sono consigliate per aumentare la leggibilità dei grafi nelle situazioni in cui l'orientamento risulterebbe ambiguo
 - *Esempio*: archi orientati dal basso verso l'alto



diagramma funzionale sequenziale

Gli archi orientati

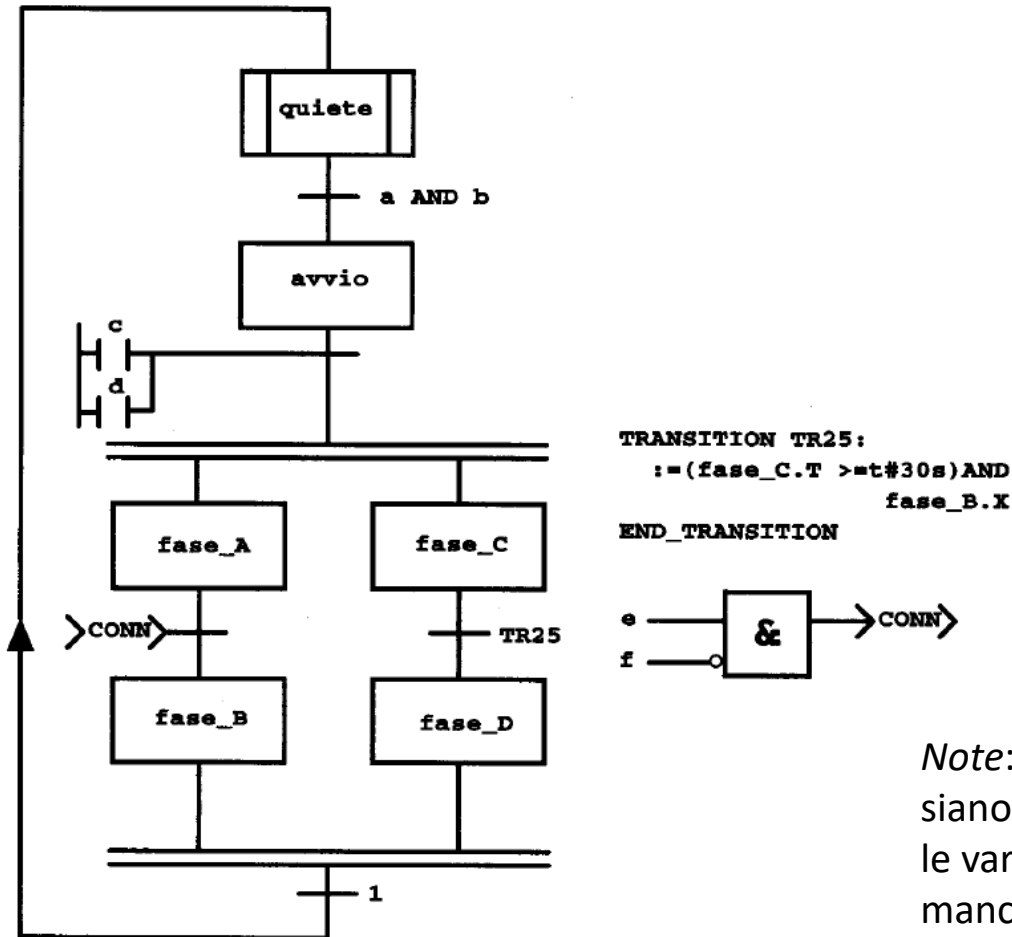
- Se più fasi convergono in una stessa transizione (*sincronizzazione*) o se ad una transizione succedono più fasi (*parallelismo* o *concorrenza*) è consigliabile utilizzare una **doppia linea orizzontale** per mettere in evidenza la fine e l'inizio di sequenze che devono evolvere in parallelo
 - In caso di definizione testuale dell'SFC, l'orientamento viene fissato con i costrutti che definiscono le transizioni del tipo
TRANSITION FROM fasi_precedenti
TO fasi_successive
- ❖ **Regole per la composizione di un SFC:**
- 1) **Due fasi non possono essere connesse direttamente, vi deve sempre essere una transizione tra loro**
 - 2) **Due transizioni non possono essere connesse direttamente, vi deve essere almeno una fase tra di loro**

diagramma funzionale sequenziale

Gli archi orientati



- *Esempio:* corpo di un blocco funzionale o di un programma in SFC



Note: si suppone che siano state definite tutte le variabili utilizzate; mancano le azioni



diagramma funzionale sequenziale

Gli archi orientati

- Definizione testuale corrispondente al grafo precedente:

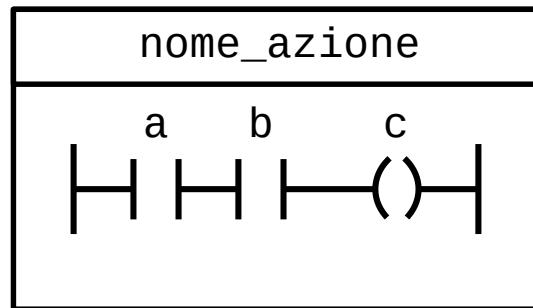
```
INITIAL_STEP quiete : END_STEP
STEP avvio : END_STEP
STEP fase_A : END_STEP
STEP fase_B : END_STEP
STEP fase_C : END_STEP
STEP fase_D : END_STEP
TRANSITION FROM quiete TO avvio := a AND b;
  END_TRANSITION
TRANSITION FROM avvio TO fase_A, fase_C := c OR b;
  END_TRANSITION
TRANSITION FROM fase_A TO fase_B := e AND NOT(f);
  END_TRANSITION
TRANSITION FROM fase_C TO fase_D := fase_b.X AND
  (fase_C.T GE T#30s);
  END_TRANSITION
TRANSITION FROM fase_B, fase_D TO quiete := 1;
  END_TRANSITION
```

diagramma funzionale sequenziale

Le azioni

– Le azioni

- Le azioni possono essere associate ad ogni fase
- *Definizione*: in forma grafica indicandone il *nome* ed il *corpo* (istruzioni che determinano cosa deve essere fatto),



oppure in forma testuale con il costrutto:

ACTION nome_azione

...; (* corpo dell'azione *)

END_ACTION

diagramma funzionale sequenziale

Le azioni



- Ogni **variabile booleana** può essere essa stessa un'azione
 - Se l'azione viene eseguita, il valore della variabile booleana associata viene portato ad 1, altrimenti viene portato a 0
 - Il nome dell'azione coincide con quello della variabile booleana e non è presente il corpo dell'azione
- Rappresentazione grafica completa

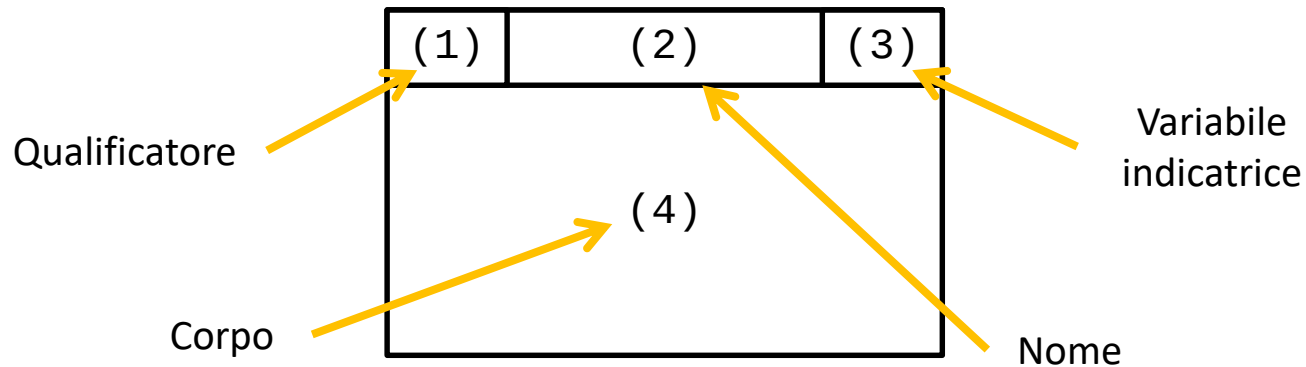


diagramma funzionale sequenziale

Le azioni



- Campo (1): *qualificatore dell'azione*
 - » Assente o uguale a **N**, per qualificare l'azione come **non memorizzata**, da eseguire fintanto che la fase è attiva e ancora una volta quando viene disattivata
 - » Uguale a **P**, per qualificare l'azione come **impulsiva**, da eseguire una volta quando la fase è attivata
 - alcune implementazioni dello standard possono assumere che l'azione venga eseguita un'altra volta quando la fase è disattivata
 - » Uguale a **D**, insieme a una costante di tipo TIME, per qualificare l'azione come **ritardata nel tempo**, cioè eseguita dopo la durata indicata, se la fase è ancora attiva, e sino alla fine della attivazione, e poi ancora per una volta
 - » Uguale a **L**, insieme a una costante di tipo TIME, per qualificare l'azione come **limitata nel tempo**, cioè eseguita per la durata indicata, o per la durata dell'attivazione della fase se minore, e poi ancora per una volta



diagramma funzionale sequenziale

Le azioni

- » Uguale a **S** oppure **R** per indicare il **set e il reset di un'azione memorizzata**, cioè che se settata in una fase permane in esecuzione anche dopo la disattivazione della fase fino a quando non viene esplicitamente resettata, e poi ancora per un ciclo di scansione
- » Uguale a **DS**, insieme a una costante di tipo TIME, per qualificare l'**azione come ritardata e memorizzata**, cioè memorizzata dopo la durata indicata se la fase è ancora attiva
- » Uguale a **SL**, insieme a una costante di tipo TIME, per qualificare l'azione come **memorizzata e limitata nel tempo**, cioè eseguita per il tempo indicato, anche se la fase si disattiva
- Campo (2): *nome dell'azione*
- Campo (3): *variabile indicatrice*
 - » Variabile booleana che viene posta a 1 nell'esecuzione dell'azione per indicare la fine dell'esecuzione dell'azione
 - Ha scopi di documentazione e può essere omessa

diagramma funzionale sequenziale

Le azioni



- Campo (4): *corpo dell'azione*
 - » Descrizione di ciò che deve essere fatto utilizzando uno dei linguaggi definiti dallo standard compreso l'SFC
 - » Il campo può essere omesso se l'azione viene definita a parte
- Il blocco dell'azione deve essere connesso alla fase a cui è associato (ad una fase possono essere associate più azioni)
 - *Graficamente*: con un collegamento a destra del simbolo della fase che stabilisce una relazione con la variabile segnalatrice della fase

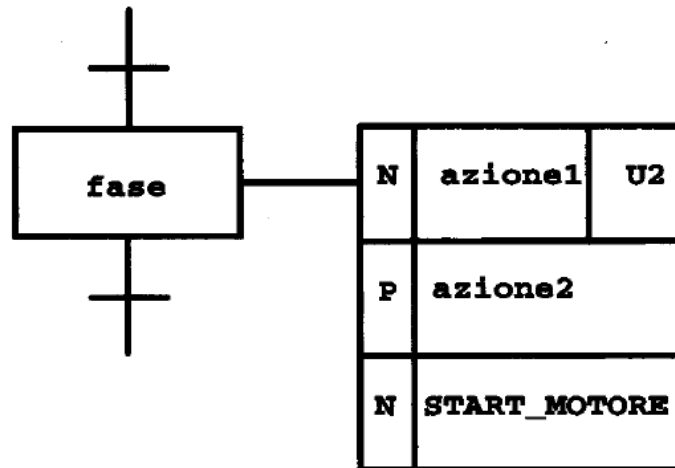




diagramma funzionale sequenziale

Le azioni

- *Testualmente*: come nel seguente esempio:

```
STEP nome_fase:  
    variabile_booleana(N);  
    altra_variabile_booleana(P);  
    AZIONE_COMPLESSA(L, T#30s, ind);  
END_STEP
```

dove è stato supposto che dovessero essere messe al valore 1 due variabili booleane (una per tutta la durata dell'attivazione della fase, l'altra è invece impulsiva) e che dovesse essere eseguita un'azione, definita a parte, come limitata nel tempo (durata 30 secondi), che prevede anche una variabile indicatrice `ind`

- Ad ogni azione è associata una **variabile booleana implicita** `nome_azione.Q` che è vera quando l'azione deve essere eseguita



diagramma funzionale sequenziale

Le azioni

- Il corpo delle azioni può essere scritto in uno qualsiasi dei linguaggi previsti dallo standard, anche l'SFC
 - È possibile sviluppare un programma in SFC con una metodologia **top-down** → si individuano le **macro-fasi** in cui passa il sistema → nelle azioni si sviluppa il dettaglio delle sequenze previste
 - Fasi senza azioni possono essere qualificate come **fasi di attesa**
 - Il sistema vi permane, senza fare nulla, sino a che non si realizzano le condizioni per passare in un'altra fase
 - I blocchi delle azione possono essere usati anche negli altri linguaggi grafici previsti dallo standard
 - *Linguaggio a contatti*: il blocco viene attivato se è alimentato
 - *Linguaggio a blocchi funzionali*: il blocco viene attivato se il suo ingresso booleano diventa vero
- e la variabile indicatrice dell'azione può essere utilizzata per segnalare il completamento dell'azione



diagramma funzionale sequenziale

Regole di evoluzione

- **Regole di evoluzione**

- La *condizione* di un SFC è l'insieme delle sue fasi attive
 - Un SFC può cambiare condizione attraverso il **superamento delle transizioni**
- Definizioni:
 - Una *transizione* è detta **abilitata** se tutte le fasi a monte di essa sono attive
 - Una *transizione* è detta **superabile** se è abilitata e la condizione a essa associata risulta essere vera
- *Regola di evoluzione: se una transizione è superabile essa viene effettivamente superata: tutte le fasi a monte vengono disattivate e tutte le fasi a valle vengono attivate*

diagramma funzionale sequenziale

Regole di evoluzione



- Le operazioni di disattivazione e attivazione si susseguono nell'esatto ordine indicato
 - La loro durata è legata alla particolare implementazione ed è data dal tempo che intercorre tra due valutazioni successive del grafo
- *Ambiguità di comportamento*: può capitare che transizioni distinte risultino superabili nello stesso istante → la regola afferma che se più transizioni diventano superabili nello stesso istante esse sono tutte superate contemporaneamente
- *Fase instabile* (nel senso delle macchine asincrone): la condizione associata alla transizione di uscita risulta già vera quando la fase viene attivata → la regola afferma che le azioni associate siano comunque eseguite prima della disattivazione della fase
 - La durata di attivazione di una fase non può essere nulla



diagramma funzionale sequenziale

Confronto con le Reti di Petri

- **SFC vs Reti di Petri**
 - Analogie
 - Entrambe le tecniche si basano sui concetti di fase e transizione
 - Hanno una rappresentazione grafica molto simile
 - L'evoluzione è causata per entrambe dal verificarsi di eventi esterni
 - Principali differenze
 - Lo stato di una fase
 - Nell'SFC è booleano (può essere o attiva o inattiva)
 - Nelle Reti di Petri lo stato di una fase è, in generale, un numero intero che rappresenta il numero di attivazioni della fase

diagramma funzionale sequenziale

Confronto con le Reti di Petri



- Transizioni simultaneamente superabili
 - In un SFC tutte le transizioni simultaneamente superabili devono effettivamente essere tutte contemporaneamente superate
 - Nelle Reti di Petri se più transizioni sono superabili, esse sono superate in una sequenza che potrebbe persino non comprenderle tutte
- Le condizioni di transizione
 - In un'SFC possono dipendere dallo stato delle singole fasi attraverso l'uso delle variabili segnalatrici della fase
 - Ciò non è previsto nelle Reti di Petri in cui un comportamento simile si potrebbe realizzare solo introducendo delle fasi e delle transizioni fittizie



diagramma funzionale sequenziale

Strutture classiche di programmazione

- Strutture classiche di programmazione
 - **Sequenza** semplice: alternanza tra fasi e transizioni in serie
 - Descrive semplici strutture di comando sequenziali (seriali)
 - **Scelta** (o **divergenza**) tra più attività: quando una fase è seguita da più transizioni
 - *Esempio*: se la fase A è attiva, sarà attivata la fase B se la condizione cond1 è vera e cond2 è falsa, la fase C se cond1 è falsa e cond2 è vera
 - cond1 e cond2 esprimono la scelta tra due possibilità

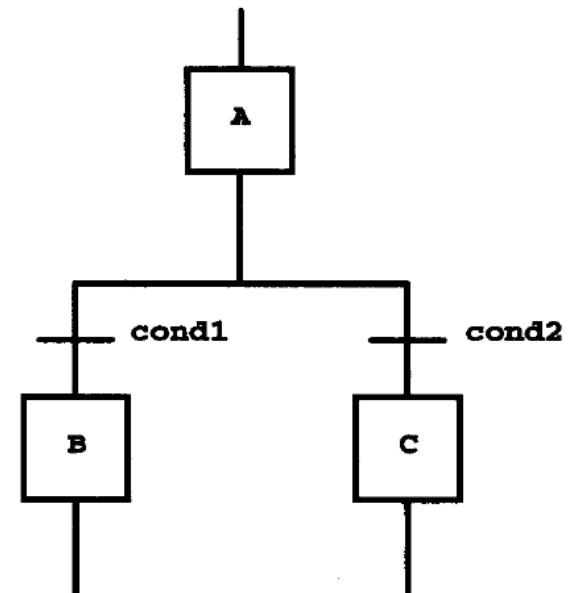


diagramma funzionale sequenziale

Strutture classiche di programmazione



- **Scelta mutuamente esclusiva**
 - Le condizioni di scelta devono sempre essere mutuamente esclusive: al più una sola condizione alla volta può risultare vera
 - **Mutua esclusione naturale**: le condizioni di scelta non sono mai vere contemporaneamente per la loro natura
 - » *Esempio*: presenza dello stesso oggetto a sinistra e a destra
 - **Mutua esclusione imposta**: la mutua esclusione viene realizzata nella costruzione delle condizioni, assegnando anche una **priorità** di superabilità
 - » Lo standard prevede che si possano **numerare** in ordine di priorità i rami della divergenza oppure che la priorità sia **imposta agendo sulle condizioni** delle transizioni



diagramma funzionale sequenziale

Strutture classiche di programmazione

» *Esempio*: condizioni di superamento modificate con l'aggiunta di priorità

$\text{cond1}' := \text{cond1};$

$\text{cond2}' := \text{cond2} \ \& \ (\text{NOT} \ \text{cond1});$

$\text{cond3}' := \text{cond3} \ \& \ (\text{NOT} \ \text{cond2}) \ \& \ (\text{NOT} \ \text{cond1});$

$\text{cond4}' := \text{cond4} \ \& \ (\text{NOT} \ \text{cond3}) \ \& \ (\text{NOT} \ \text{cond2}) \ \& \ (\text{NOT} \ \text{cond1});$

- *Caso particolare*: **salto di sequenza**

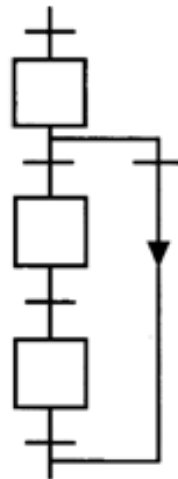


diagramma funzionale sequenziale

Strutture classiche di programmazione



- **Convergenza** di più attività: quando più sequenze terminano nella stessa fase attraverso transizioni diverse

- *Esempio:* la fase f9 sarà attivata se saranno superate la transizione T15 e/o la transizione T23
- **La convergenza è la chiusura naturale della scelta**
- *Caso particolare: ciclo di una sequenza*

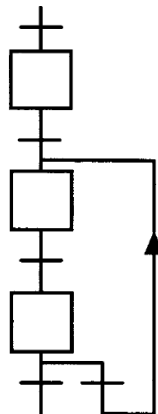
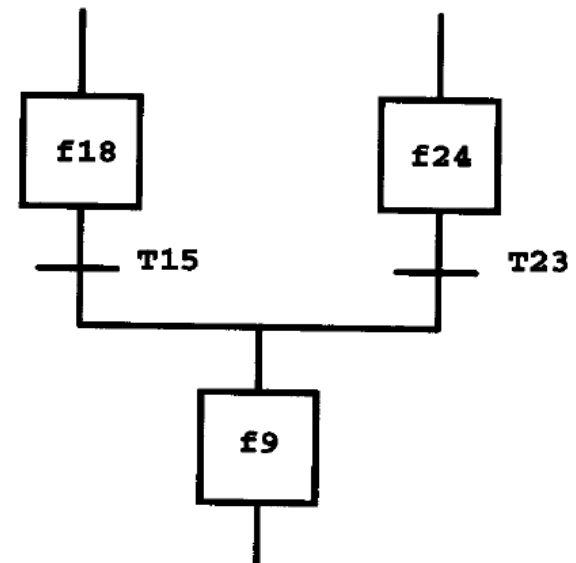




diagramma funzionale sequenziale

Strutture classiche di programmazione

- **Parallelismo (o concorrenza)** tra più attività: quando una transizione è seguita da più fasi

- *Esempio:* se la transizione T1 diventa superabile saranno attivate contemporaneamente le fasi f11, f21 e f31 dando luogo a diverse sequenze che evolveranno in modo indipendente le une dalle altre

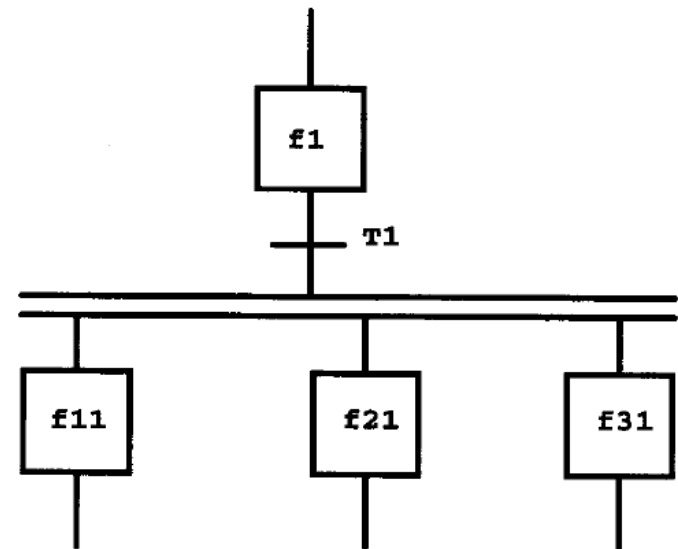


diagramma funzionale sequenziale

Strutture classiche di programmazione



– **Sincronizzazione** tra più attività: quando più fasi precedono la stessa transizione

- Condizione affinché la transizione si superata è che tutte le sequenze parallele siano terminate e che quindi le fasi finali siano tutte attive
- *Esempio:* le fasi finali f15, f29 e f34 sono sincronizzate dalla transizione T33
- **La sincronizzazione è la chiusura naturale di un parallelismo**

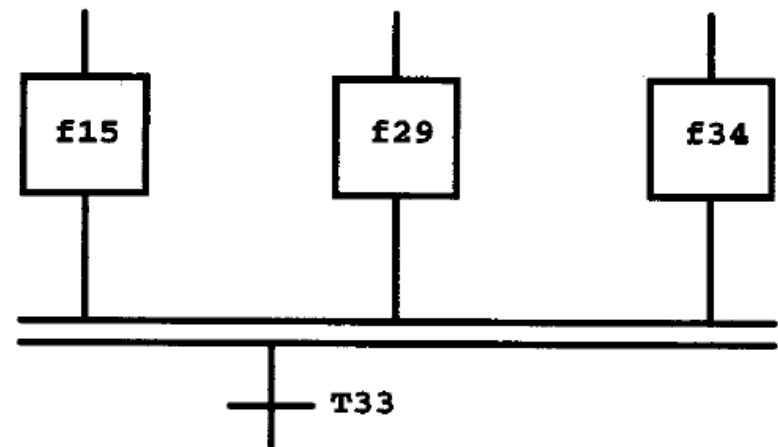


diagramma funzionale sequenziale

Strutture classiche di programmazione



– Sincronizzazione locale tra più sequenze

- *Esempio:* la sequenza di destra, una volta raggiunta la fase f47, deve attendere che la sequenza di sinistra superi la transizione T5 prima di poter procedere oltre

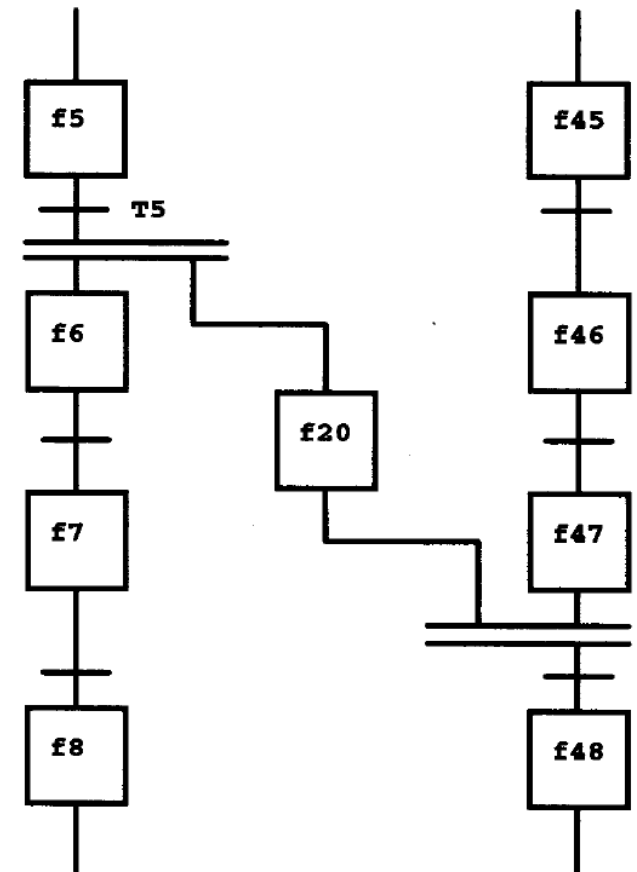




diagramma funzionale sequenziale

Strutture di programmazione da evitare

- **Strutture di programmazione da evitare**
 - Struttura errata: scelta con sincronizzazione

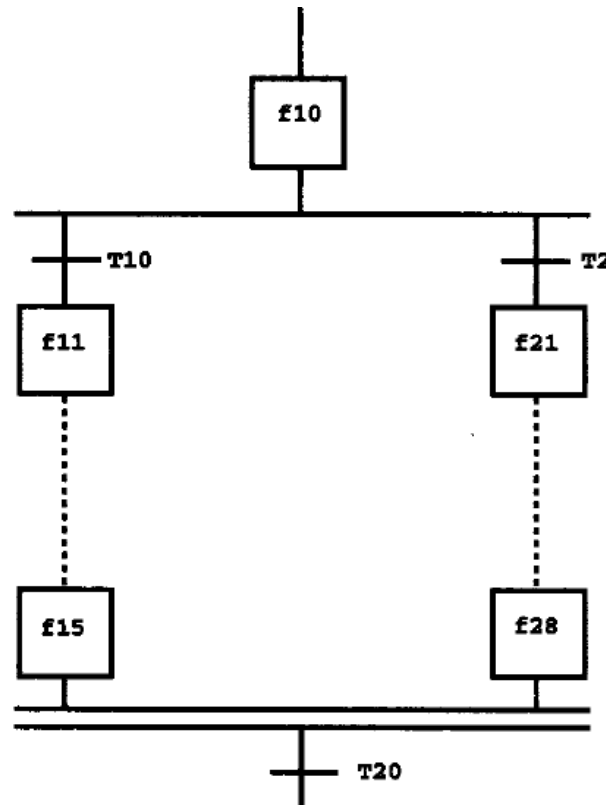




diagramma funzionale sequenziale

Strutture di programmazione da evitare

- Struttura ambigua: **parallelismo con convergenza**

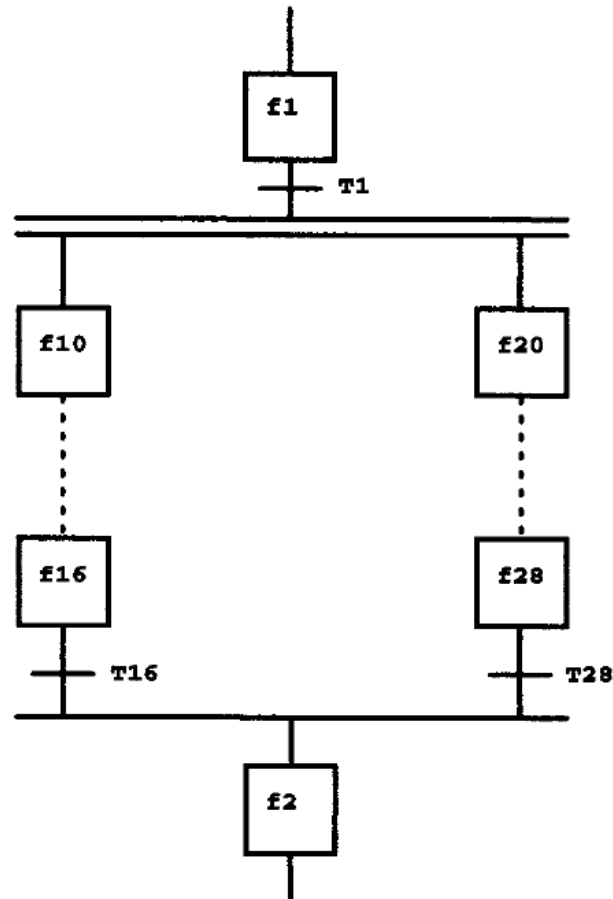




diagramma funzionale sequenziale

Vantaggi nel progetto del controllo

- **Vantaggi dell'SFC nel progetto del controllo**
 - L'SFC ha un grande potere espressivo
 - È il miglior linguaggio per la programmazione del controllo logico/sequenziale per sistemi ad eventi discreti
 - Il controllo si realizza attraverso sequenze di attività, la cui evoluzione dipende da condizioni logiche
 - Il controllo logico sequenziale di sistemi ad eventi discreti non è peculiare della sola automazione industriale
 - *Esempi:* sistema semaforico, stazione ferroviaria, accesso ad un canale di comunicazione, accesso a basi dati, coda di un server, ecc.
 - **Programmare in SFC corrisponde a descrivere, secondo la sua sintassi, il comportamento desiderato del sistema**

diagramma funzionale sequenziale

Vantaggi nel progetto del controllo



- SFC per la descrizione delle **specifiche funzionali**
 - L'individuazione delle specifiche funzionali per un impianto di automazione è propedeutico alla progettazione degli algoritmi di controllo che lo devono realizzare
 - Devono essere espresse in un linguaggio che non permetta ambiguità
 - L'SFC può utilmente essere impiegato per descrivere le specifiche funzionali → le azioni da eseguire e le condizioni da valutare sono scritte nel *linguaggio naturale*
 - *Esempi*: “apri la valvola”, “vai a destra”, “temperatura maggiore di 20°C”, “fine corsa raggiunto”, ecc.
 - La non ambiguità dell'SFC consente di individuare eventuali **specifiche non chiaramente definite**



diagramma funzionale sequenziale

Vantaggi nel progetto del controllo

- L'SFC prodotto è **autodocumentante**
 - Descrive esattamente quello che il sistema deve fare, le fasi attraverso cui il sistema deve passare e le ragioni per cui potrebbe dover cambiare fase
- Dall'SFC funzionale si può progettare l'SFC operativo con un approccio top-down
 - Vanno specificate le effettive azioni che devono essere eseguite in corrispondenza di ogni fase e le effettive condizioni da verificare
 - Una fase individuata a livello funzionale potrebbe essere realizzata con più fasi intermedie, secondo una programmazione top-down
 - » L'SFC può essere utilizzato anche nella definizione della azioni
- **Debugging** del programma
 - Se il programma si blocca in una fase si deve solo capire perché la transizione di uscita a quella fase non è stata superata



diagramma funzionale sequenziale

Vantaggi nel progetto del controllo

- Efficienza implementativa
 - I programmi scritti in SFC sono altamente efficienti in quanto in ogni valutazione del grafo devono essere effettivamente **eseguite solo le azioni associate alle fasi attive e valutate solo le transizioni in uscita a queste fasi**

diagramma funzionale sequenziale

Esempi di programmazione



- Esempi di programmazione in SFC
 - Gli esempi proposti sono per un puro scopo didattico
 - Esempi reali dovrebbero prevedere la gestione di condizioni di emergenza, il passaggio da modalità automatica a manuale e viceversa, ecc.
 - Le soluzioni proposte sono solo una delle possibili soluzioni per i problemi considerati
 - Vi sono sempre più soluzioni allo stesso problema, che possono differire per la chiarezza, la lunghezza, l'occupazione di memoria, ecc.
 - Per brevità, sarà presentato solo il corpo dei programmi
 - Sono omesse le fasi di definizione delle variabili e il loro collegamento agli ingressi e alle uscite fisiche del dispositivo

diagramma funzionale sequenziale

Esempi di programmazione



– Il carrello automatico

- Il carrello alla richiesta di un operatore deve spostarsi dalla sx alla dx di un binario, essere caricato mediante il ribaltamento di un serbatoio e riportarsi a sx per lo scarico del materiale

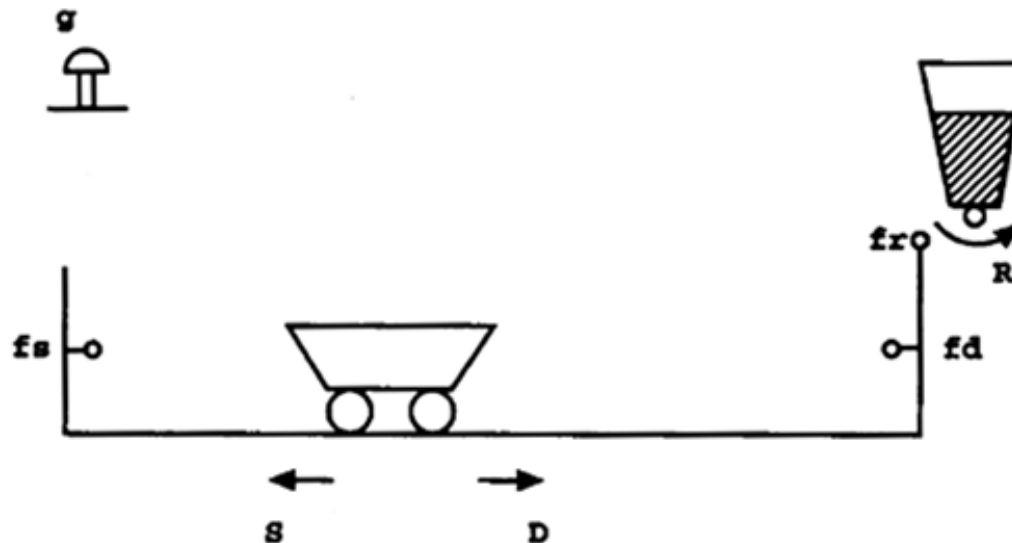




diagramma funzionale sequenziale

Esempi di programmazione

- Segnali di ingresso di tipo digitale:
 - Finecorsa per il carrello: f_s e f_d
 - Finecorsa per il serbatoio: f_r
 - Segnale proveniente dal pulsante di attivazione del ciclo: g
- Segnali di comando di tipo digitale:
 - Moto del carrello: S e D
 - Ribaltamento del serbatoio: R



diagramma funzionale sequenziale

Esempi di programmazione

- SFC funzionale

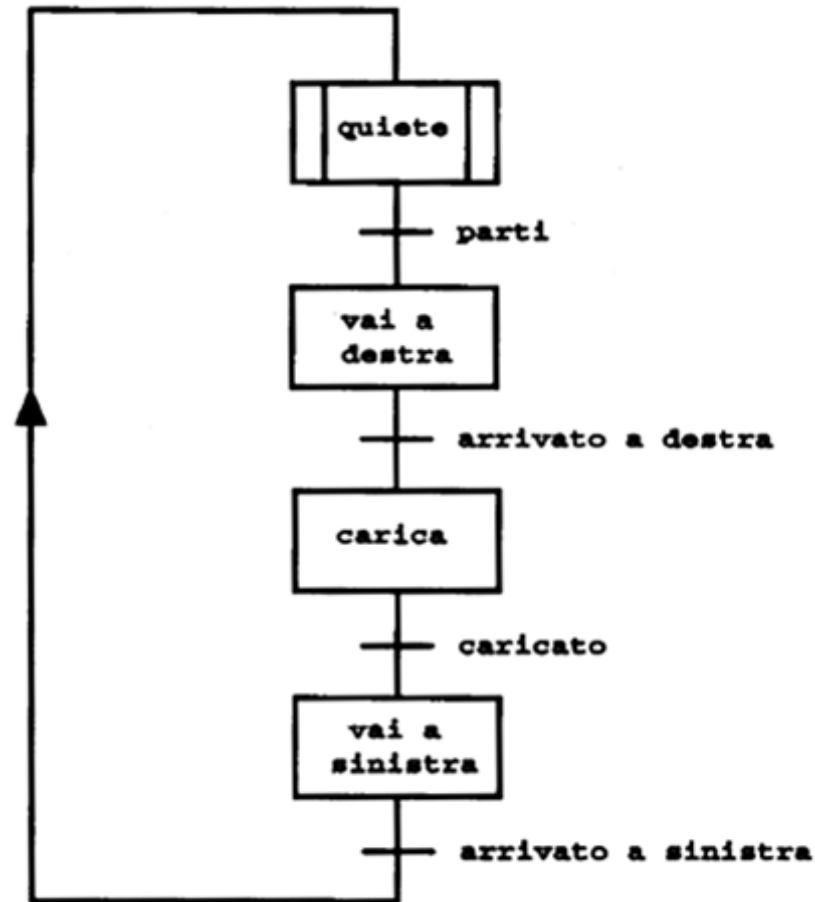


diagramma funzionale sequenziale

Esempi di programmazione



- SFC di controllo

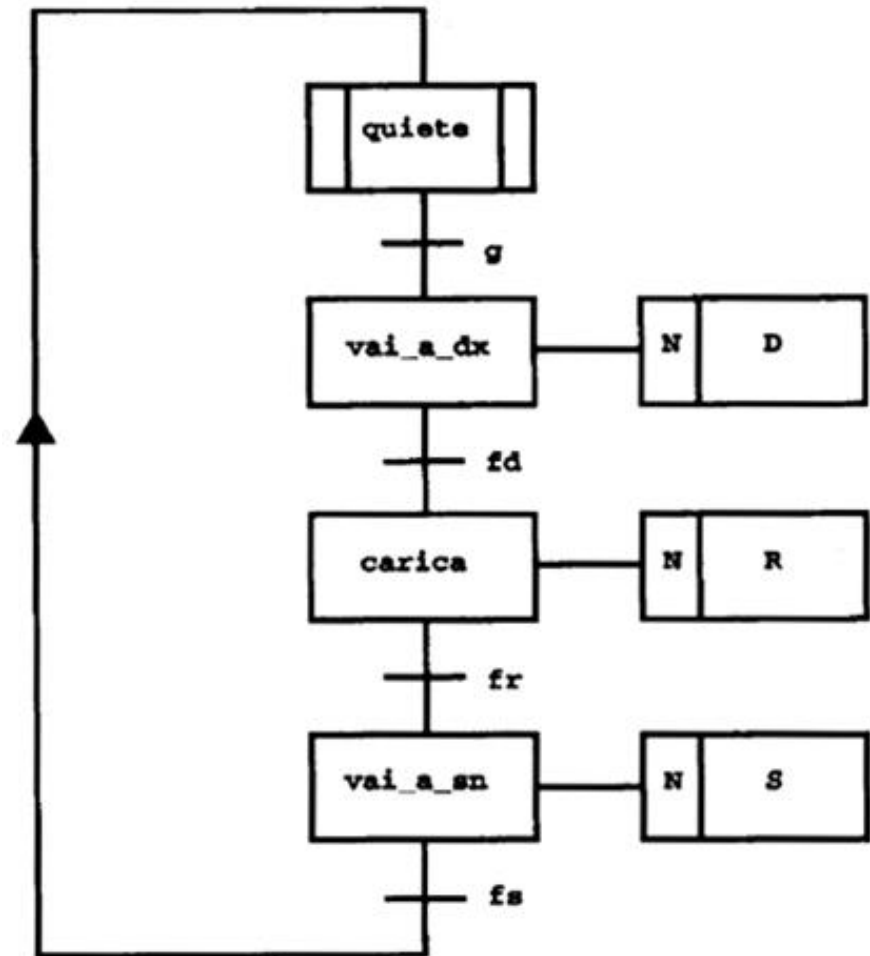
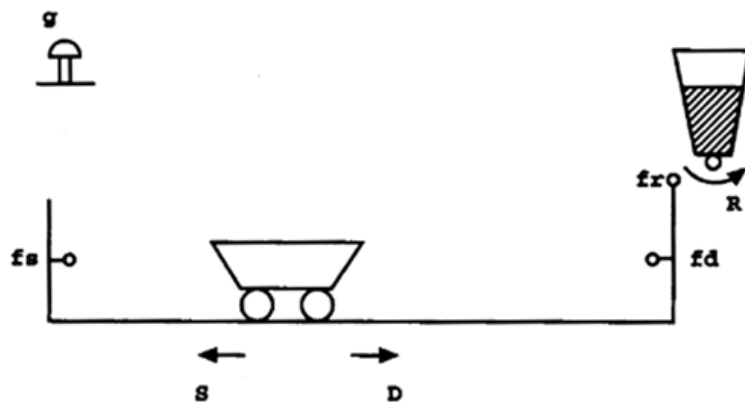


diagramma funzionale sequenziale

Esempi di programmazione



– Il sistema di irrigazione

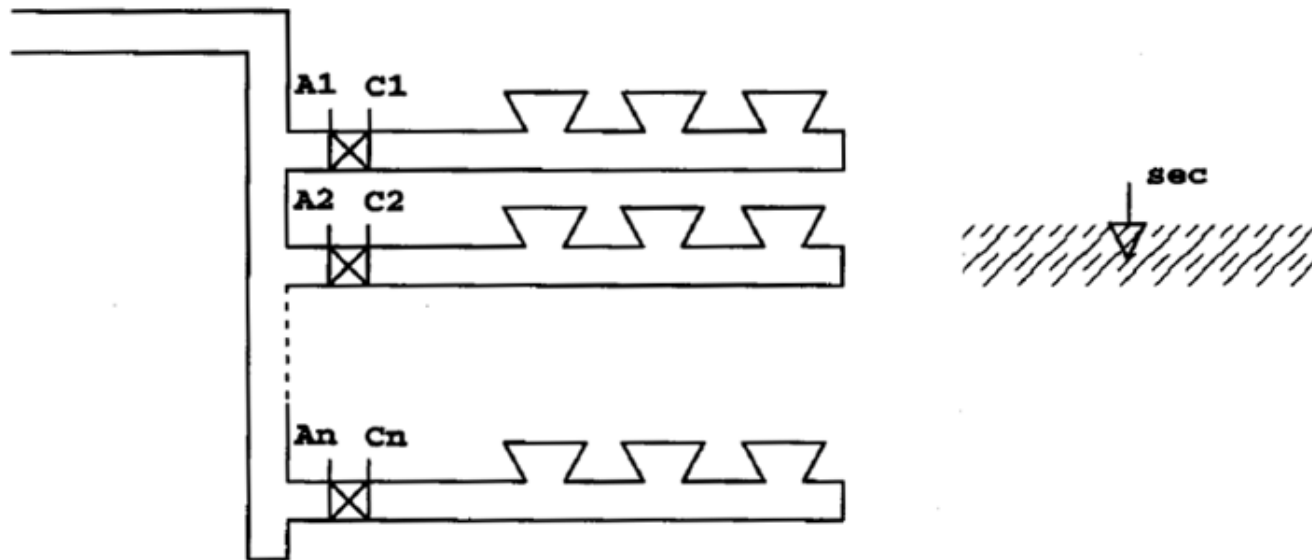




diagramma funzionale sequenziale

Esempi di programmazione

- La partenza del ciclo di irrigazione viene data da un sensore, SEC, di umidità che rileva la secchezza del suolo
- Si deve comunque prevedere un ciclo di irrigazione ogni tre giorni al massimo
- La rete irrigatrice è costituita da n rami, ciascuno comandato da un'elettrovalvola, i quali devono essere alimentati in sequenza data la limitatezza della portata d'acqua disponibile
- L' i -esima elettrovalvola è comandata in apertura dal segnale A_i che deve durare 400 ms e in chiusura dal segnale C_i che deve durare 200 ms
- Per ogni ramo deve essere prevista un'ora di annaffiatura

diagramma funzionale sequenziale

Esempi di programmazione

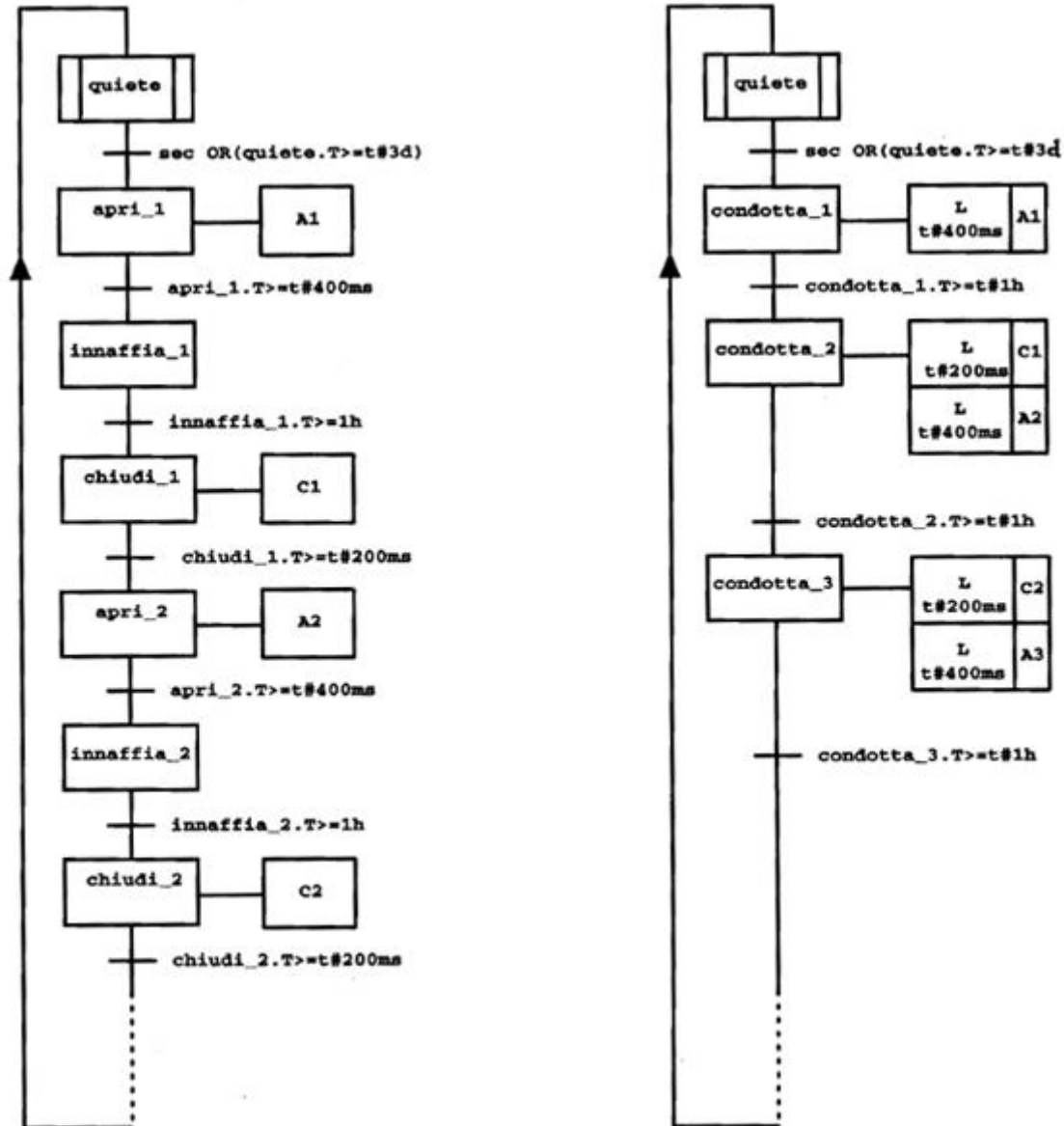


diagramma funzionale sequenziale

Esempi di programmazione



– Il trapano automatico

- La stazione di trapanatura automatica illustrata in figura ha a disposizione i seguenti segnali di ingresso, di tipo digitale:

- p, pulsante di avvio ciclo
- a, finecorsa della slitta alto
- m, contatto di media corsa
- b, finecorsa della slitta basso
- x, contatto che segnala l'altezza del pezzo
- y, contatto che segnala la presenza del pezzo

- I segnali di comando digitali sono:
 - S, che comanda la salita del trapano
 - D, che comanda la discesa del trapano

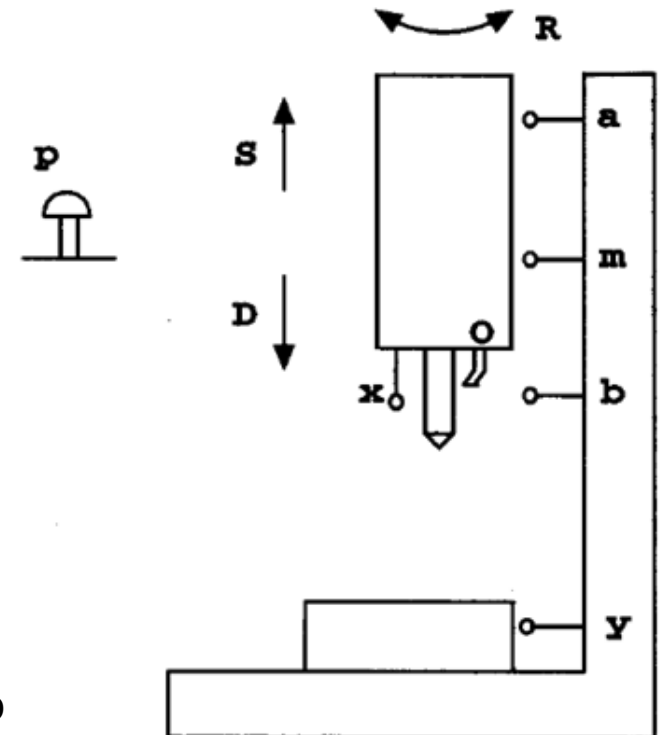




diagramma funzionale sequenziale

Esempi di programmazione

- R, che attiva la rotazione del trapano
- O, che attiva l'oleatrice
- Il contatto di altezza pezzo, x, segnala se il pezzo è da considerarsi basso, quando diventa vero dopo che lo è diventato il contatto di media corsa m, o alto, quando accade l'inverso
- Il ciclo inizia quando l'operatore preme il pulsante di partenza, se è presente un pezzo
 - Se il pezzo è basso il trapano viene messo in rotazione e fatto scendere sino al finecorsa basso; poi viene fatto risalire sino al finecorsa alto, dove viene fermata la rotazione
 - Se il pezzo è alto, il trapano, messo in rotazione, deve scendere fino al contatto medio e poi risalire; quindi deve essere attivata la pompa oliatrice per 2 secondi; infine il trapano deve scendere sino al finecorsa basso e poi risalire ed essere fermato
- Un nuovo ciclo può avviarsi solo se il pezzo trattato viene rimosso e sostituito con un altro

diagramma funzionale sequenziale

Esempi di programmazione

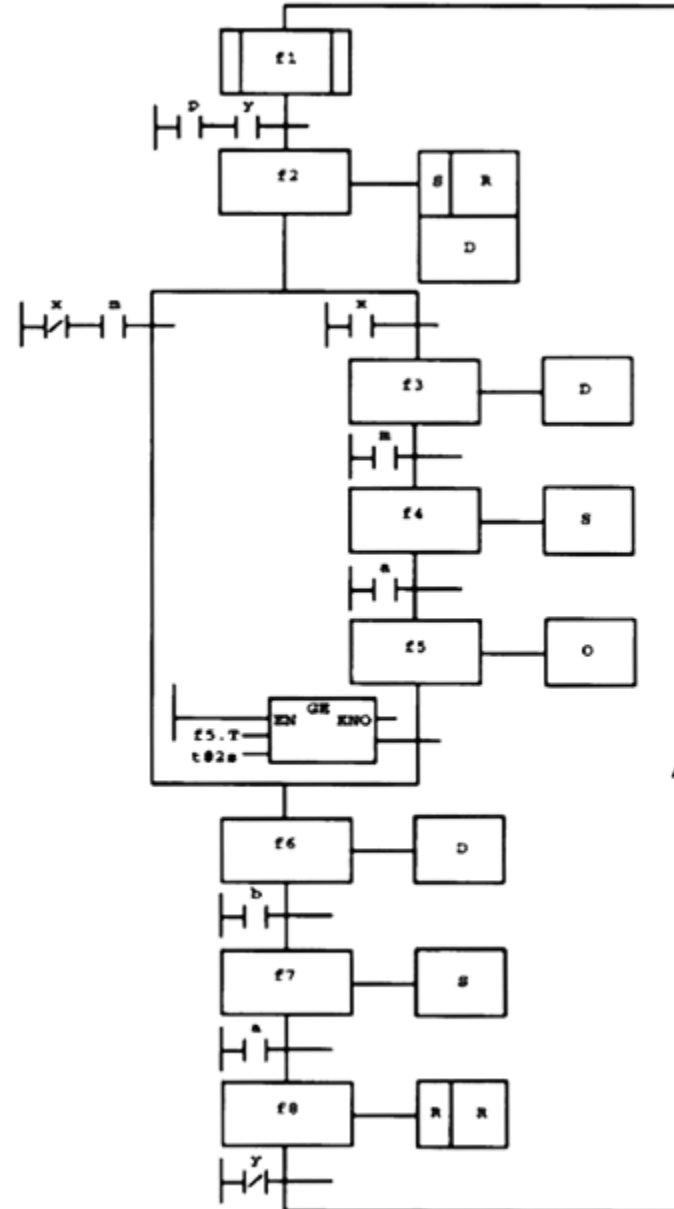
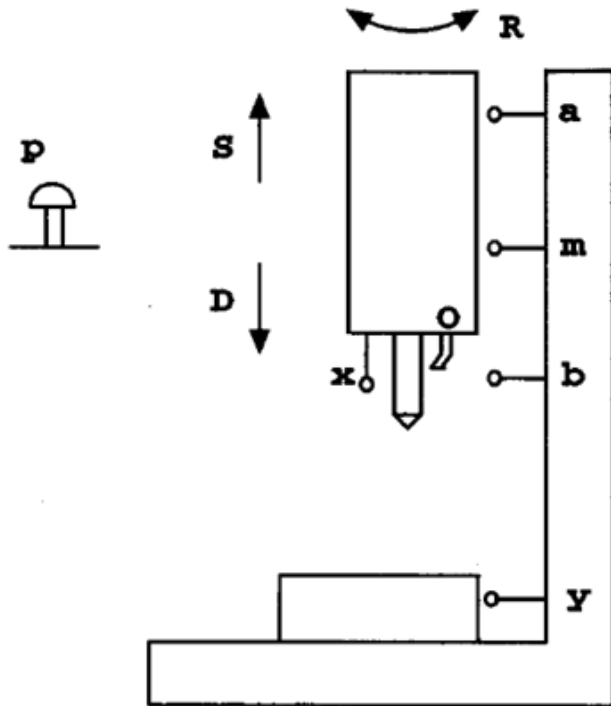


diagramma funzionale sequenziale

Esempi di programmazione



– Il distributore automatico di bibite

- Il distributore automatico di bibite illustrato in figura ha a disposizione i seguenti segnali di ingresso, di tipo digitale:
 - moneta, della gettoniera
 - aperto, indica che il vano di prelievo è stato aperto
- I segnali di comando digitali sono:
 - BLOCCA, per bloccare il vano di prelievo
 - SBLOCCA, per sbloccare il vano di prelievo
 - SA, comando del pistoncino superiore
 - SB, comando del pistoncino inferiore

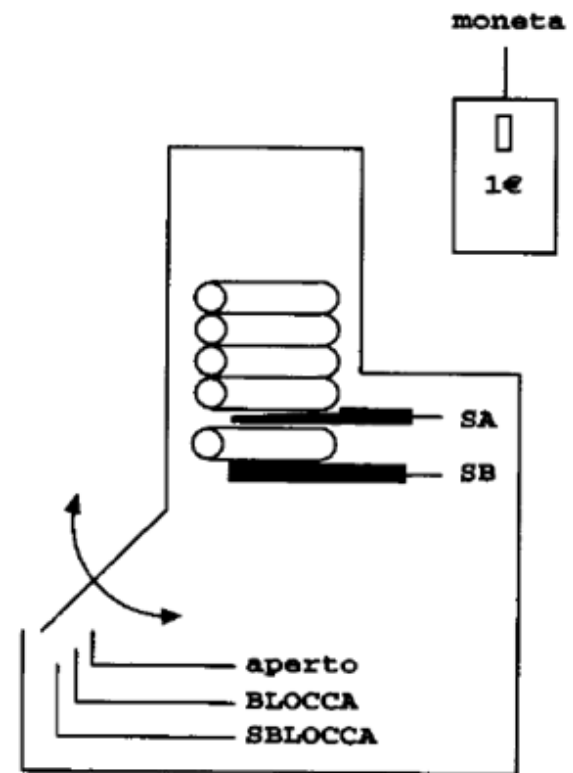


diagramma funzionale sequenziale

Esempi di programmazione



- Se il segnale di comando è inviato ai pistoncini, questi vanno a dx, altrimenti una molla li riporta a sx
- Si vuole realizzare l'automatismo in modo che, all'inserimento di una moneta, una sola bibita sia fatta cadere nel vano e sia possibile prelevarla

diagramma funzionale sequenziale

Esempi di programmazione

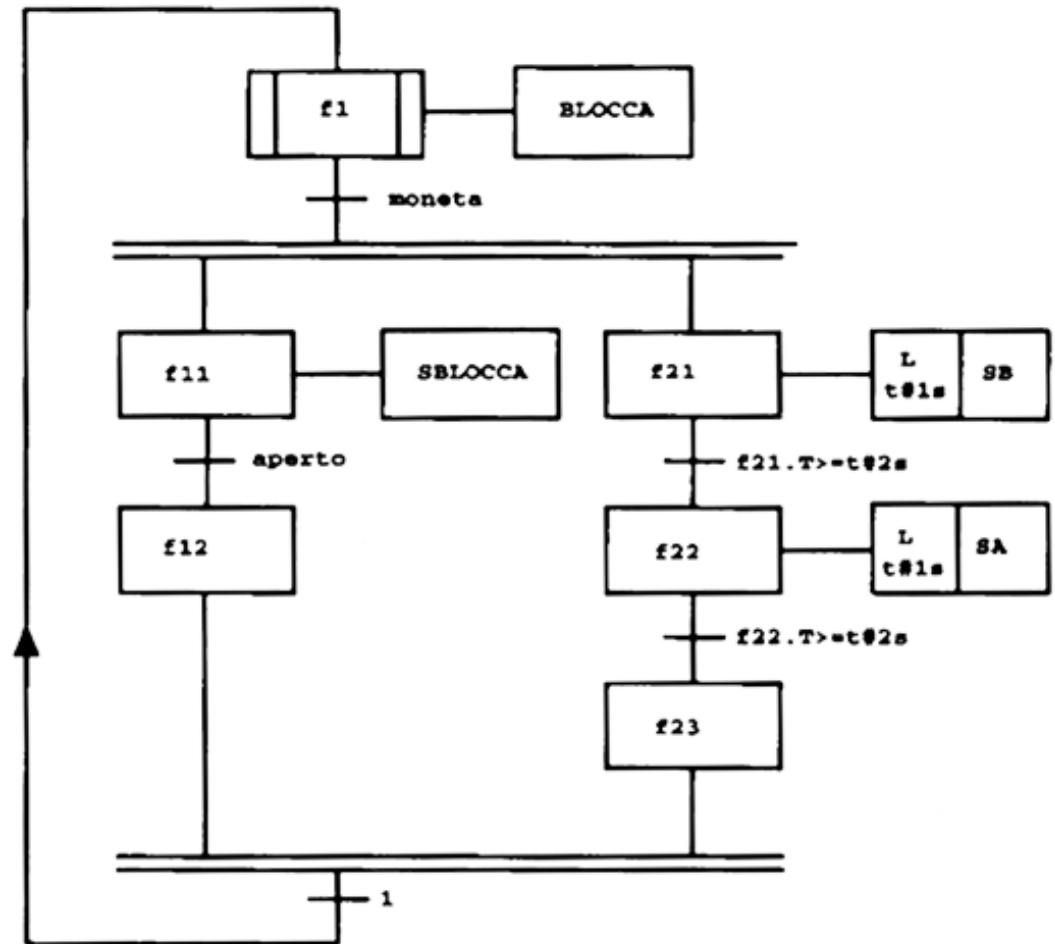
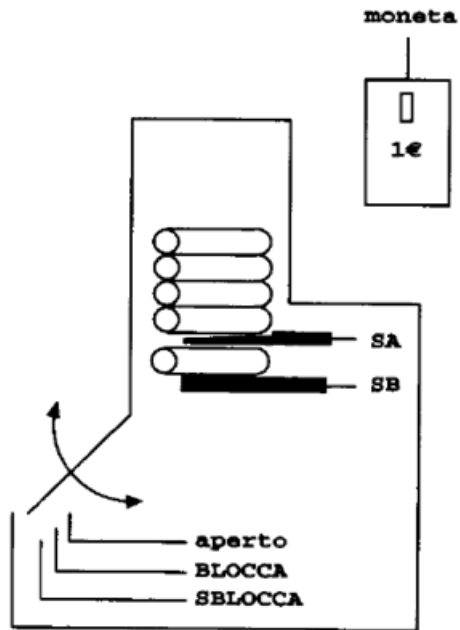


diagramma funzionale sequenziale

Esempi di programmazione



– L'accesso al parcheggio

- Si consideri il sistema per l'entrata a pagamento in un parcheggio schematizzato in figura

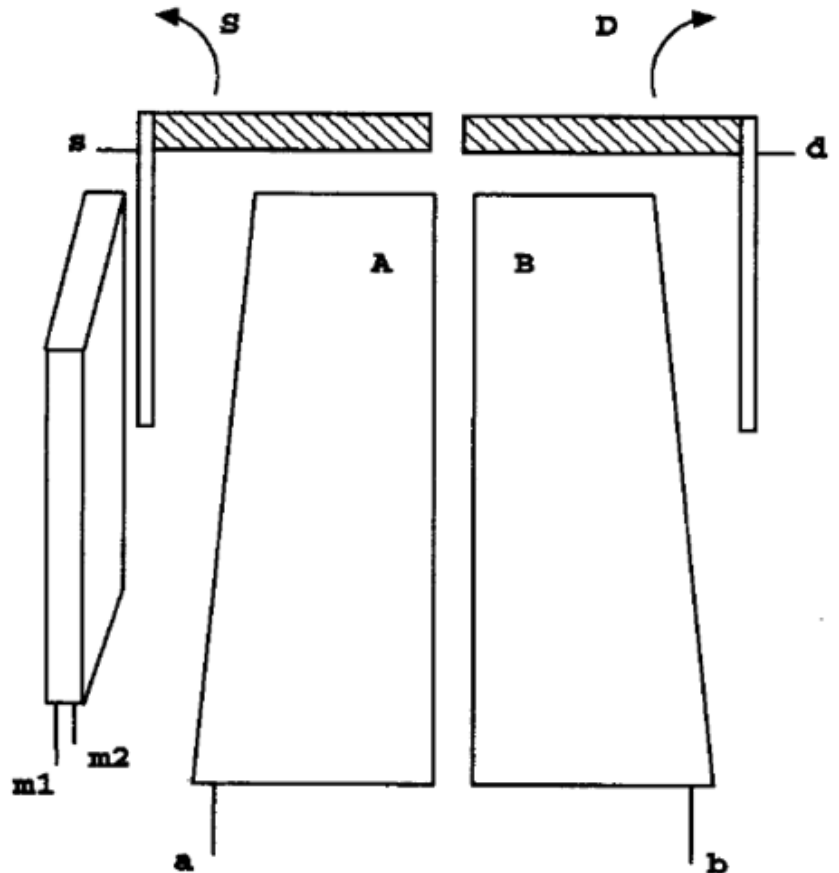




diagramma funzionale sequenziale

Esempi di programmazione

- La barriera di ingresso è composta da due barre: la barra di sinistra si può aprire da sola per far passare un motoveicolo mentre le due barre si devono aprire insieme per far passare un'automobile
- Se gli attuatori delle due barre non sono alimentati, le barre si richiudono con un meccanismo a molla
- Sulla sinistra vi è una gettoniera con due feritoie per il passaggio di monete da 1 e 2 euro, sul pavimento due placche con celle di carico per rilevare la presenza di un motoveicolo (segnale dalla sola placca A) o di un'automobile (segnali dalle placche A e B)
- Per far aprire la sola barra di sinistra un motoveicolo deve essere sulla sola placca A e deve essere inserita (almeno) una moneta da 1 euro; la barriera si richiude quando il veicolo non è più sulla placca A



diagramma funzionale sequenziale

Esempi di programmazione

- Per far aprire entrambe le barre, un'auto si deve portare sulle placche A e B e devono essere inserite due monete da 1 euro oppure una da 2 euro (almeno); la barriera si richiude quando non vi è più la presenza dell'auto sulle placche
- Si assume che, se un veicolo si appoggia prima sulla placca A, esso deve impegnare la placca B entro 1 secondo per essere considerato un'auto, altrimenti è considerato un motoveicolo
- Le barre devono essere entrambe chiuse prima di poter trattare un nuovo veicolo



diagramma funzionale sequenziale

Esempi di programmazione

- Il sistema di controllo ha a disposizione i seguenti segnali di ingresso, di tipo digitale:
 - a, presenza del veicolo sulla placca A
 - b, presenza del veicolo sulla placca B
 - m1, passaggio moneta da 1 euro
 - m2, passaggio moneta da 2 euro
 - s, barra di sinistra chiusa
 - d, barra di destra chiusa
- I segnali di comando, di tipo digitale, sono:
 - S, apri (e mantieni aperta) la barra di sinistra
 - D, apri (e mantieni aperta) la barra di destra

diagramma funzionale sequenziale

Esempi di programmazione

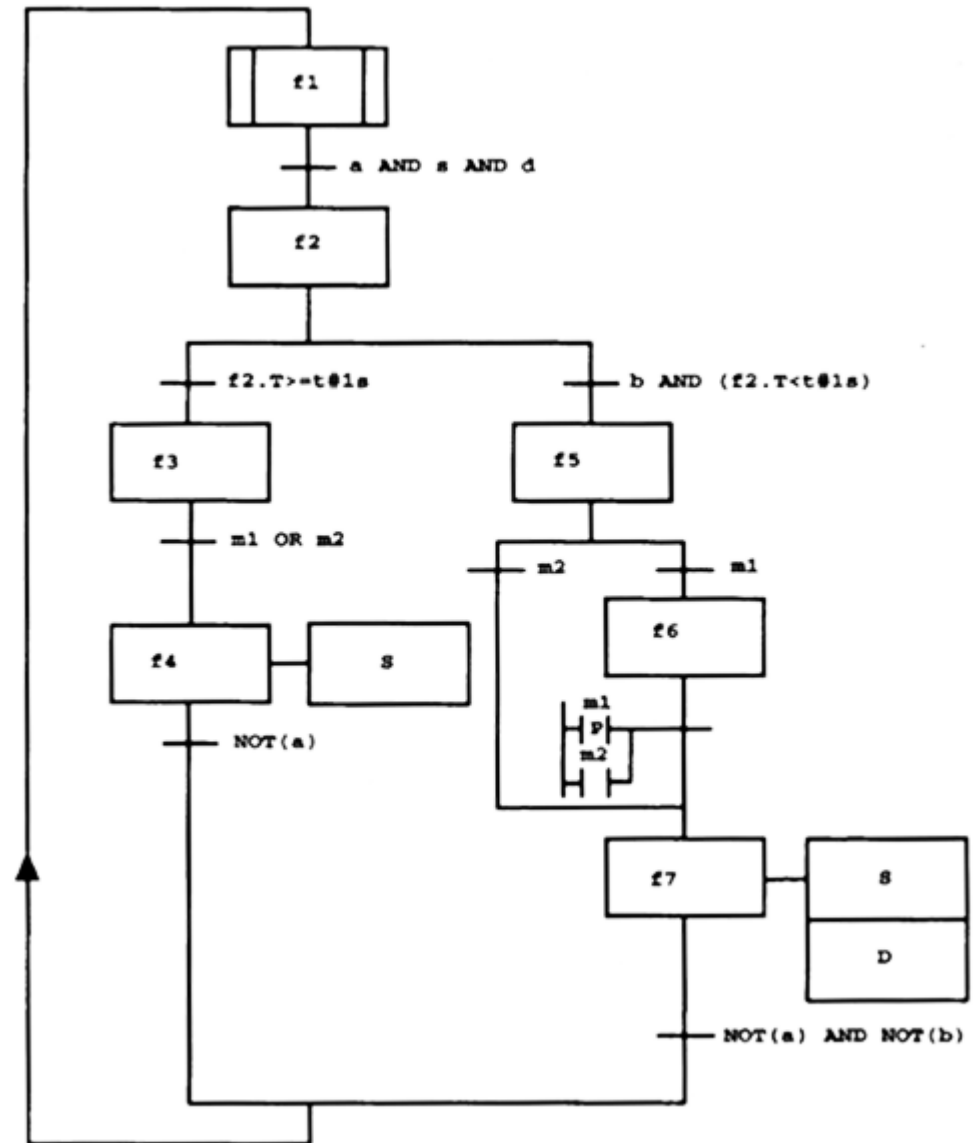
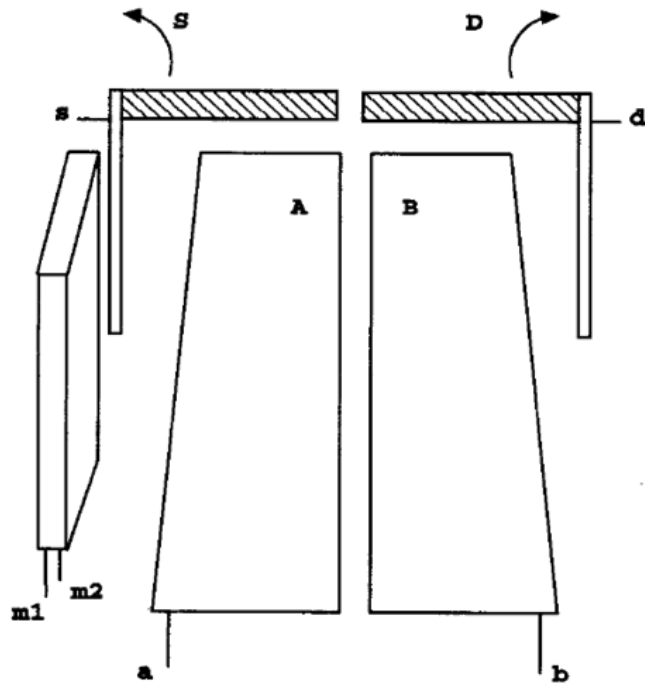


diagramma funzionale sequenziale

Esempi di programmazione



– Il carroponte a elettrocalamita

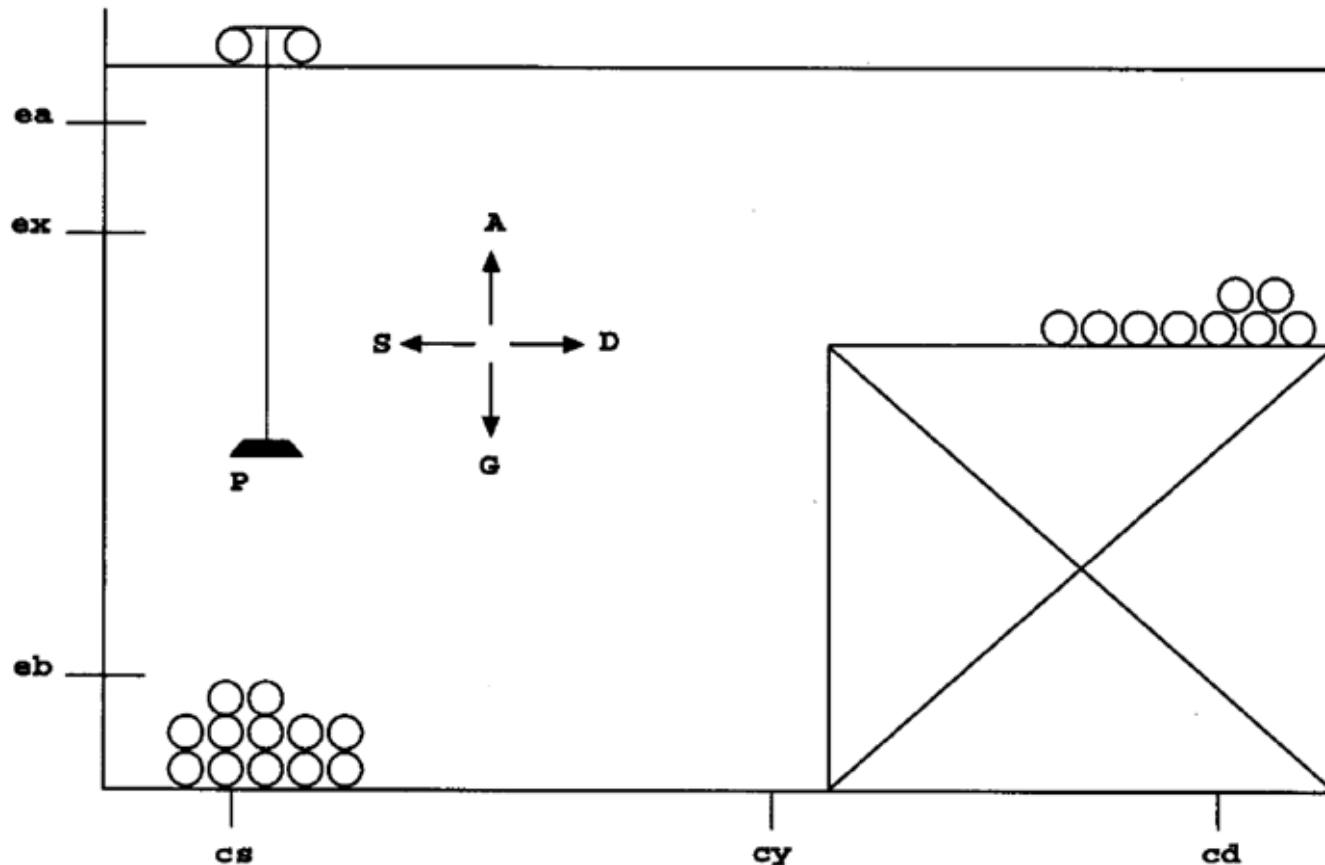




diagramma funzionale sequenziale

Esempi di programmazione

- I segnali di ingresso disponibili, di tipo digitale, sono:
 - ea, presenza elettrocalamita in alto
 - eb, presenza elettrocalamita in basso
 - ex, presenza elettrocalamita ad altezza x
 - cs, presenza carrello a sinistra
 - cd, presenza carrello a destra
 - cy, presenza carrello nella posizione y
 - vai, comando di partenza ciclo (non riportato nella figura)
- I segnali di comando, di tipo digitale, sono:
 - D, carrello va a destra
 - S, carrello va a sinistra
 - G, fai scendere l'elettrocalamita
 - A, fai salire l'elettrocalamita
 - E, attiva l'elettrocalamita



diagramma funzionale sequenziale

Esempi di programmazione

- All'inizio l'elettrocalamita è in alto a sinistra
- Alla partenza deve prendere un carico ferroso in basso e poi spostarsi in alto a destra dove lo rilascia, prima di riportarsi nella configurazione iniziale
- Deve essere evitata la zona proibita il cui confine è segnalato dal segnale cy

diagramma funzionale sequenziale

Esempi di programmazione

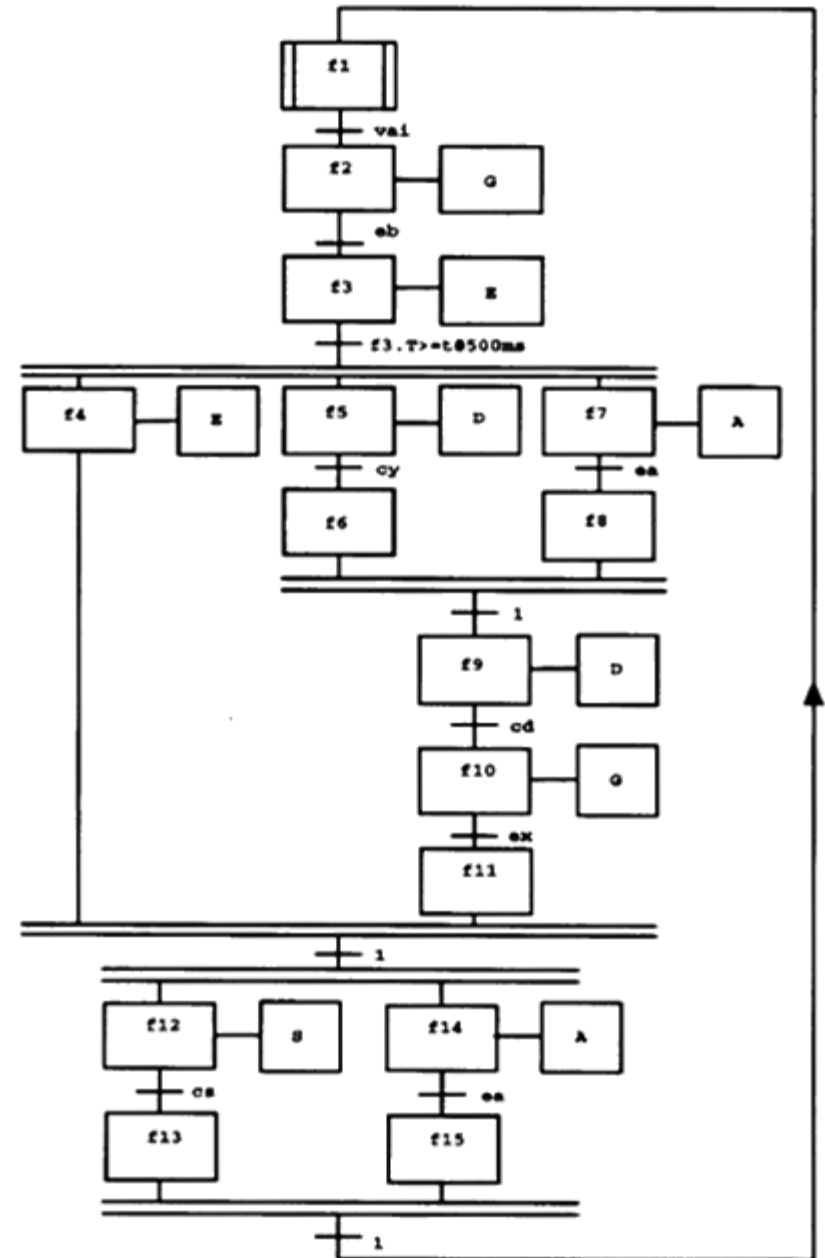
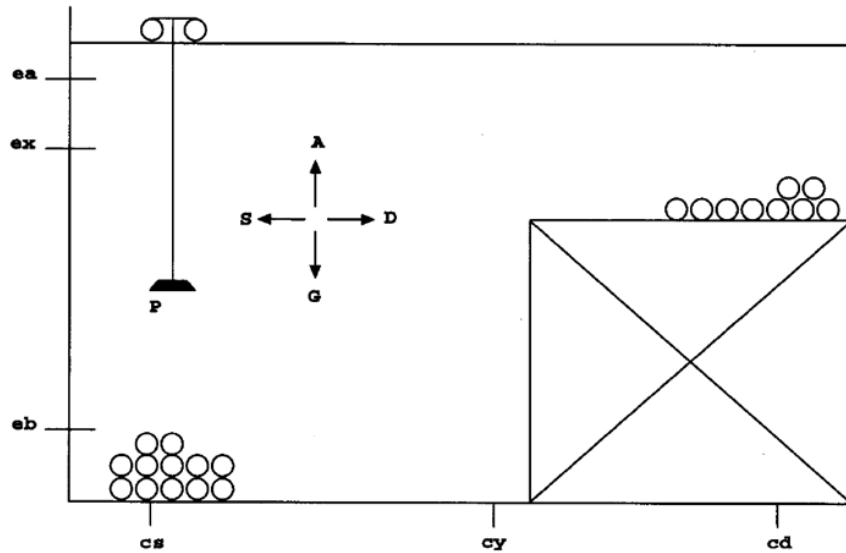


diagramma funzionale sequenziale

Esempi di programmazione



– Il sistema semaforico

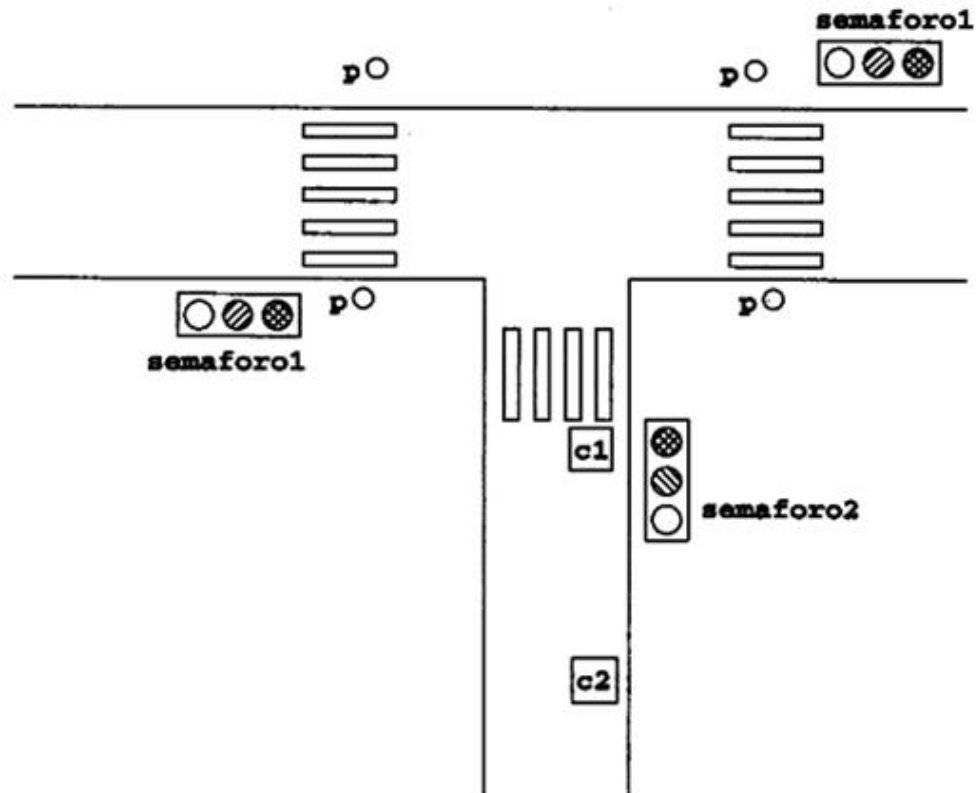




diagramma funzionale sequenziale

Esempi di programmazione

- I segnali di ingresso disponibili, di tipo digitale, sono:
 - p, pulsante richiesta pedonale
 - c1, c2, rilevatori presenza continua auto
- I segnali di comando, di tipo digitale, sono:
 - V1, luce verde semaforo 1
 - G1, luce gialla semaforo 1
 - R1, luce rossa semaforo 1
 - V2, luce verde semaforo 2
 - G2, luce gialla semaforo 2
 - R2, luce rossa semaforo 2
- Il semaforo 1 è posto su una strada di grande traffico e deve essere normalmente verde

diagramma funzionale sequenziale

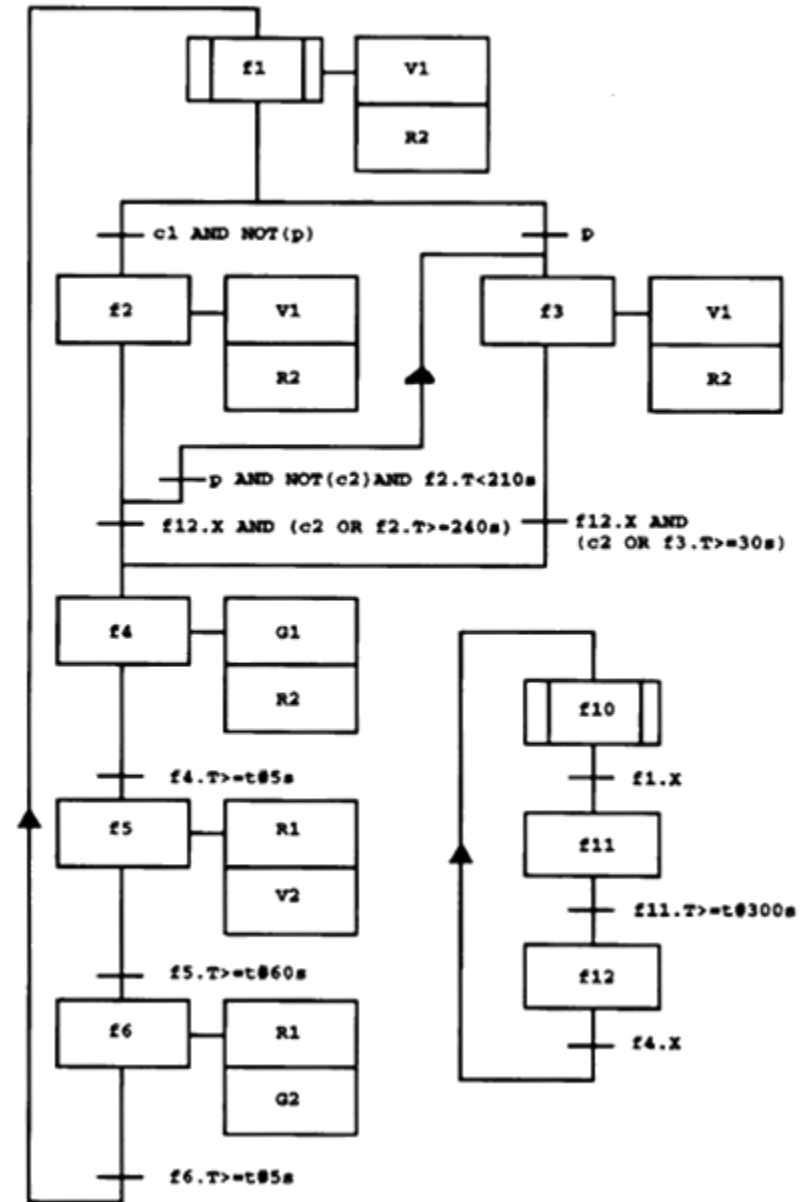
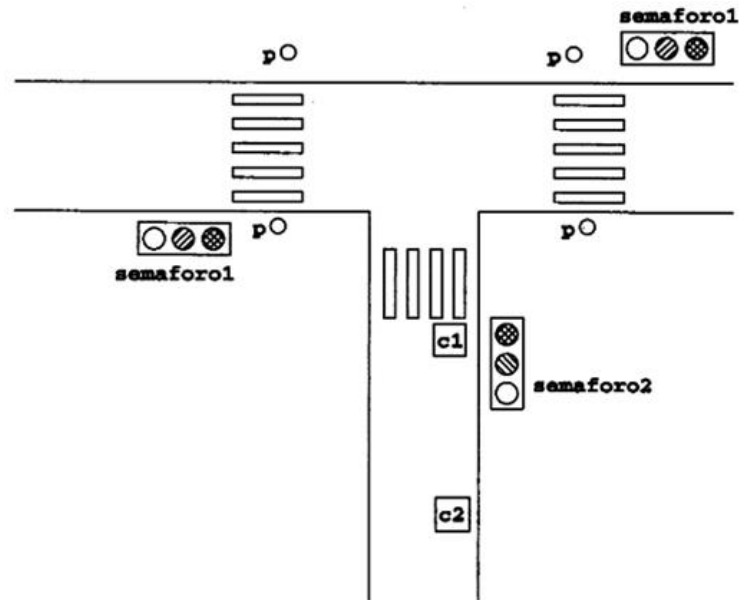
Esempi di programmazione



- Il ciclo semaforico, descritto più avanti, si deve attivare:
 1. Solo se il verde del semaforo 1 è stato presente per almeno 5 minuti
 2. Immediatamente, se entrambi i rilevatori C1 e C2 segnalano la presenza continua di auto (coda al semaforo 2)
 3. Dopo 30 secondi, se vi è la richiesta da parte di un pedone
 4. Dopo 240 secondi, se il rilevatore C1 segnala la presenza continua di un'auto
- L'ordine di presentazione delle quattro specifiche è anche quello di priorità che deve essere rispettato nel loro soddisfacimento
- Il ciclo semaforico è quello usuale, composto, per ogni semaforo, da una successione verde-giallo-rosso-verde
 - Il giallo deve durare 5 secondi mentre il verde del semaforo 2 deve durare 60 secondi

diagramma funzionale sequenziale

Esempi di programmazione



traduzione del diagramma funzionale sequenziale

Introduzione



- Traduzione del Sequential Functional Chart
 - L'SFC è uno degli aspetti più interessanti dello standard
 - Consente di scrivere con facilità gli algoritmi di controllo logico/sequenziale per sistemi ad eventi discreti
 - Semplice individuare fasi e transizioni che definiscono il comportamento desiderato del processo da automatizzare
 - Rappresenta le specifiche funzionali del sistema di controllo
 - Fornisce una documentazione implicita del programma e ne facilita una sua eventuale manutenzione
 - Esistono dispositivi di controllo che non prevedono l'SFC tra i possibili linguaggi di programmazione
 - Vecchi dispositivi o piccoli e molto semplici

traduzione del diagramma funzionale sequenziale

Introduzione



- Occorre una **procedura di traduzione** del programma sviluppato in SFC in modo che:
 - Sia effettivamente implementabile su una qualsiasi macchina per l'elaborazione delle informazioni
 - Sia semplice e dettagliata, in modo da produrre direttamente un codice eseguibile senza ulteriori sforzi progettuali
 - Presenti un legame biunivoco tra le istruzioni tradotte e le fasi e le transizioni dell'SFC di partenza, in modo da
 - Conservare la proprietà di autodocumentazione
 - Consentire la manutenzione sull'SFC di partenza e limitare le modifiche sul codice tradotto alle sole parti coinvolte, senza dover tradurre tutto il codice da capo
 - Far corrispondere a piccole modifiche sull'SFC piccole modifiche sul codice tradotto

traduzione del diagramma funzionale sequenziale

Introduzione



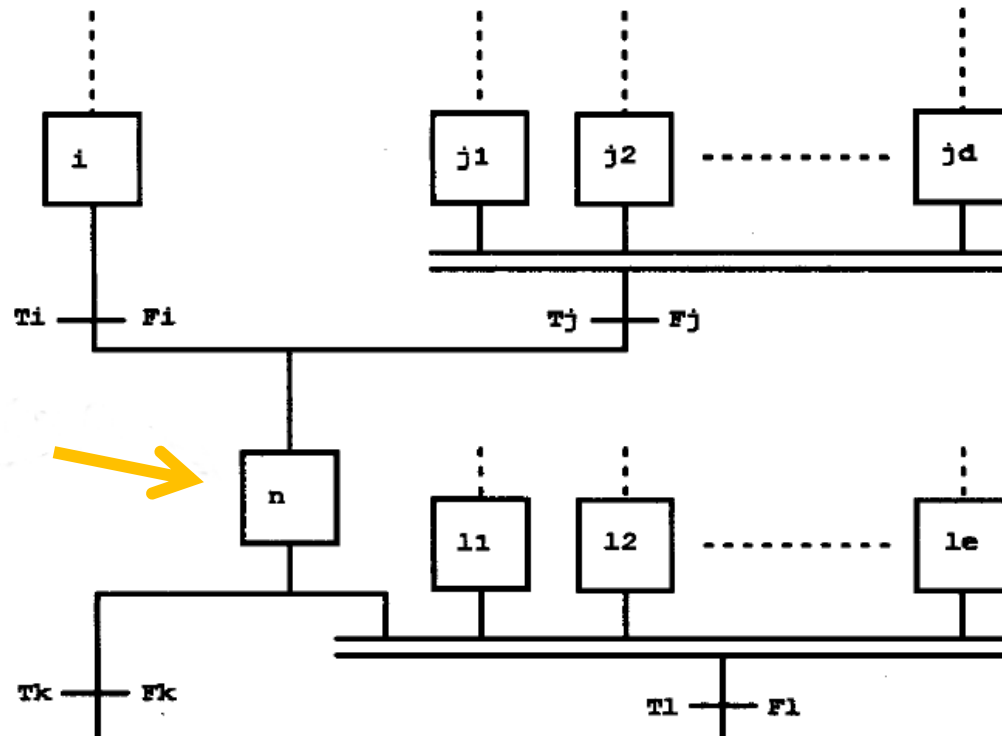
- Per sviluppare una procedura di traduzione è necessario formalizzare il comportamento dell'SFC, ovvero le regole di evoluzione del suo stato
 - Tali regole possono essere espresse mediante **equazioni booleane** oppure con un **algoritmo di evoluzione**
 - Entrambe le soluzioni assicurano la possibilità di realizzare un SFC tramite una macchina per l'elaborazione delle informazioni

traduzione del diagramma funzionale sequenziale

Equazioni booleane equivalenti



- Equazioni booleane equivalenti
 - Si consideri la fase generica di un SFC



traduzione del diagramma funzionale sequenziale

Equazioni booleane equivalenti



la quale può essere connessa

- A monte a delle fasi isolate tramite una transizione (come la fase i e la transizione T_i) oppure alla transizione di sincronizzazione di sequenze parallele (come le fasi $j_1, \dots, j_d$ e la transizione T_j)
- A valle essa può avere delle transizioni dirette (come la T_k) oppure chiudersi su una stessa transizione con altre fasi (transizione T_l e fasi $l_1, \dots, l_e$)

– Le regole di evoluzione affermano che:

- una fase diventa attiva se una delle transizioni a monte è superabile $\rightarrow$ l'equazione booleana che descrive l'attivazione della fase n ovvero che fornisce lo stato del suo segnalatore è

$$n.X := \sum_i (i.X \cdot F_i) + \sum_j \left[F_j \cdot \prod_h j \cdot x_h \right]$$

traduzione del diagramma funzionale sequenziale

Equazioni booleane equivalenti



dove con il simbolo F sono state indicate le condizioni associate alle transizioni

- Una fase viene disattivata se essa è attiva e se risulta superabile una delle transizioni a valle → l'equazione booleana che descrive la disattivazione della fase n è

$$n.X := n.X \cdot \text{NOT} \left[\sum_k F_k + \sum_l \left(F_l \cdot \prod_h l \cdot X_h \right) \right]$$

- L'equazione booleana complessiva che descrive l'attivazione e la disattivazione della fase è data dalla OR delle due precedenti equazioni
- Equazioni relative alle azioni saranno funzione dello stato dei segnalatori delle fasi in cui esse sono presenti

traduzione del diagramma funzionale sequenziale

Equazioni booleane equivalenti

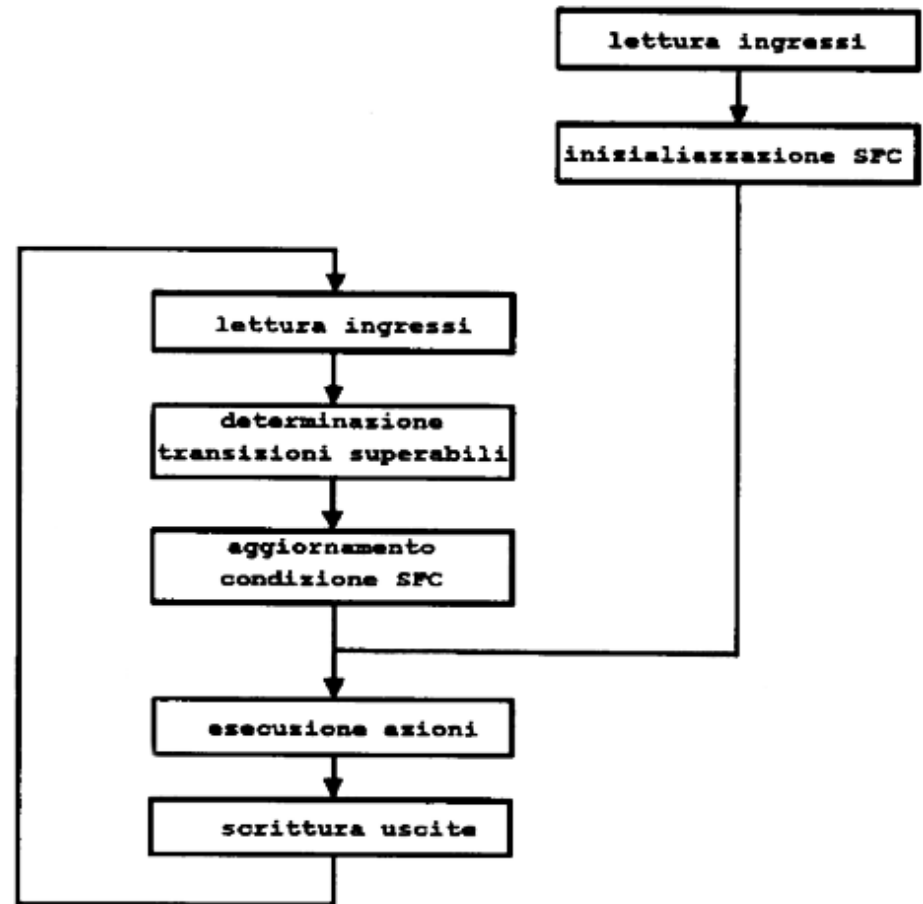


- Vanno quindi aggiunte solo le procedure per la lettura degli ingressi e la scrittura delle uscite per conseguire la realizzazione dell'SFC
- *Svantaggio di questo approccio*: le equazioni relative alla condizioni sulle transizioni risultano confuse all'interno delle equazioni delle fasi
 - Si perde il requisito della facilità della documentazione e manutenzione
 - *Esempio*: la modifica della condizione associata ad una sola transizione comporterebbe la modifica di tutte le equazioni di evoluzione delle fasi connesse a quella transizione

traduzione del diagramma funzionale sequenziale

Equazioni booleane equivalenti

- Algoritmo di evoluzione
 - Si tratta della rappresentazione in forma algoritmica delle regole di evoluzione
 - Prevede una fase di inizializzazione ed un ciclo iterativo



traduzione del diagramma funzionale sequenziale

Equazioni booleane equivalenti



– Fase di inizializzazione

- È necessaria la lettura degli ingressi, in quanto l'esecuzione delle azioni potrebbe essere condizionata dallo stato degli ingressi
- Sono attivate tutte le fasi iniziali
- Sono eseguite tutte le (eventuali) azioni associate alle fasi iniziali

– Il ciclo iterativo prevede

- L'aggiornamento dei valori degli ingressi fisici
- La determinazione delle eventuali transizioni superabili (quelle con fasi a monte tutte attive e condizioni associate vere)
- La determinazione della nuova condizione dell'SFC attraverso l'attivazione e disattivazione delle fasi
- L'esecuzione delle azioni associate alle fasi
- L'aggiornamento dei valori delle variabili di uscita

traduzione del diagramma funzionale sequenziale

Equazioni booleane equivalenti



- Il ciclo da realizzare richiede:
 - La lettura degli ingressi fisici
 - La determinazione delle uscite di controllo attraverso
 - la valutazione della superabilità delle transizioni
 - la determinazione della nuova condizione
 - l'esecuzione delle azioni
 - La scrittura delle uscite fisiche
- Si tratta di un classico ciclo di controllo che andrebbe associato ad un compito ciclico continuo
 - La realizzazione dell'algoritmo prevede la realizzazione di tale ciclo o l'utilizzo di meccanismi preesistenti nel dispositivo impiegato
 - *Esempio:* controllori a logica programmabile

traduzione del diagramma funzionale sequenziale

Codifica dell'algoritmo di evoluzione in linguaggio a contatti



- Codifica dell'algoritmo di evoluzione in linguaggio a contatti
 - Si presenterà la codifica in linguaggio a contatti perché è attualmente il linguaggio più diffuso per la programmazione di PLC
 - Nel caso di utilizzo di un PLC si sfrutterà il fatto che il ciclo è già presente
 - È prevista la lettura degli ingressi e la scrittura delle uscite fisiche
 - Le variabili temporali sono direttamente implementate (blocchi funzionali di temporizzazione)

traduzione del diagramma funzionale sequenziale

Codifica dell'algoritmo di evoluzione in linguaggio a contatti



– Passi del processo di codifica:

- Si associa ad ogni fase una variabile booleana che rappresenta il segnalatore della fase
 - Indica lo stato della fase (attiva o inattiva)
 - Si utilizza il nome della fase
 - Questi valori saranno utilizzati come condizioni per l'esecuzione delle azioni corrispondenti e per la valutazione delle condizioni delle transizioni
- Si associa ad ogni transizione una variabile booleana che funga da segnalatore della transizione
 - Indica se quella transizione è superabile o meno
 - Si numerano le transizioni e si indicheranno tali valori con Tr seguito dal numero della transizione

traduzione del diagramma funzionale sequenziale

Codifica dell'algoritmo di evoluzione in linguaggio a contatti



- La determinazione delle transizioni superabili corrisponderà a valutare i valori dei segnalatori delle transizioni in dipendenza del valore delle fasi a monte e del valore della condizione associata
- Saranno utilizzati per determinare quali fasi vanno attivate e quali disattivate, aggiornando i valori dei segnalatori
- Il programma in linguaggio a contatti che codifica l'SFC originario sarà composto da quattro sezioni:
 1. La sezione di inizializzazione
 2. La sezione di valutazione delle transizioni
 3. La sezione di aggiornamento della condizione
 4. La sezione di esecuzione delle azioni

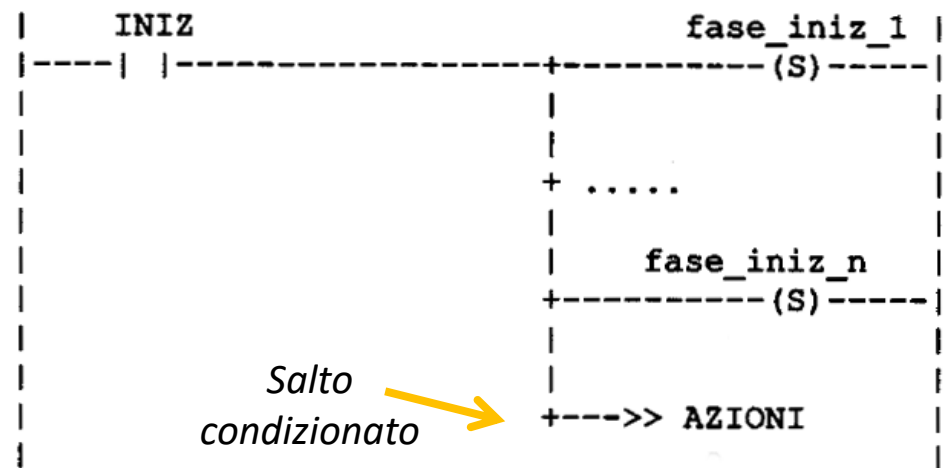
traduzione del diagramma funzionale sequenziale

Codifica dell'algoritmo di evoluzione in linguaggio a contatti



– Sezione di inizializzazione

- Va eseguita una sola volta all'inizio dell'esecuzione del programma
- La sua funzione è quella di porre a 1 il valore dei segnalatori delle fasi iniziali previste dall'SFC
- Si utilizzeranno delle bobine a memorizzazione associate ai segnalatori delle fasi iniziali
- Si suppone che il dispositivo di controllo metta a disposizione un bit di stato INIZ il cui valore è vero solo durante il primo ciclo
- *Esempio:*



traduzione del diagramma funzionale sequenziale

Codifica dell'algoritmo di evoluzione in linguaggio a contatti



– Sezione di valutazione delle transizioni

- Deve valutare se tutte le transizioni sono superabili o meno
- Viene aggiornato lo stato dei segnalatori di tutte le transizioni
- La sezione prevede una istruzione per ogni transizione che ne esprime la superabilità

```
|      (* valutazione delle transizioni *)      |
|
|  fase i      condizione                      Tri  |
|-----| |-----| |-----| |-----| |-----|
|
|  fase j      condizione                      Trj  |
|-----| |-----| |-----| |-----| |-----|
|
|  fase k fase l  condizione                      Trk  |
|-----| |-----| |-----| |-----| |-----|
|
|
```

traduzione del diagramma funzionale sequenziale

Codifica dell'algoritmo di evoluzione in linguaggio a contatti



– Sezione di aggiornamento della condizione

- Deve disattivare le fasi a monte delle transizioni superabili ed attivare le fasi a valle
- Sarà composta da una istruzione per ogni transizione per disattivare le fasi a monte (nel caso in cui sia superabile) e **successivamente** da una istruzione per ogni transizione per attivare le fasi a valle (nel caso in cui sia superabile)

– Sezione di esecuzione delle azioni

- È dedicata all'esecuzione delle azioni associate a fasi attive
- Sarà preceduta dall'etichetta AZIONI : cui punta il salto condizionato utilizzato nella sezione di inizializzazione
- Si userà lo stato delle variabili segnalatrici delle fasi per condizionare l'esecuzione delle azioni associate a tali fasi

traduzione del diagramma funzionale sequenziale

Codifica dell'algoritmo di evoluzione in linguaggio a contatti



- Traduzione delle variabili temporali associate alle fasi
 - Possono essere utilizzate nell'SFC nelle condizioni di transizione e nelle azioni limitate o ritardate nel tempo (qualificatori L e D)
 - Nel linguaggio a contatti sono realizzate tramite i blocchi funzionali di temporizzazione on-delay (TON)
 - Saranno abilitati nelle fasi in cui devono contare il trascorrere del tempo come fossero delle azioni

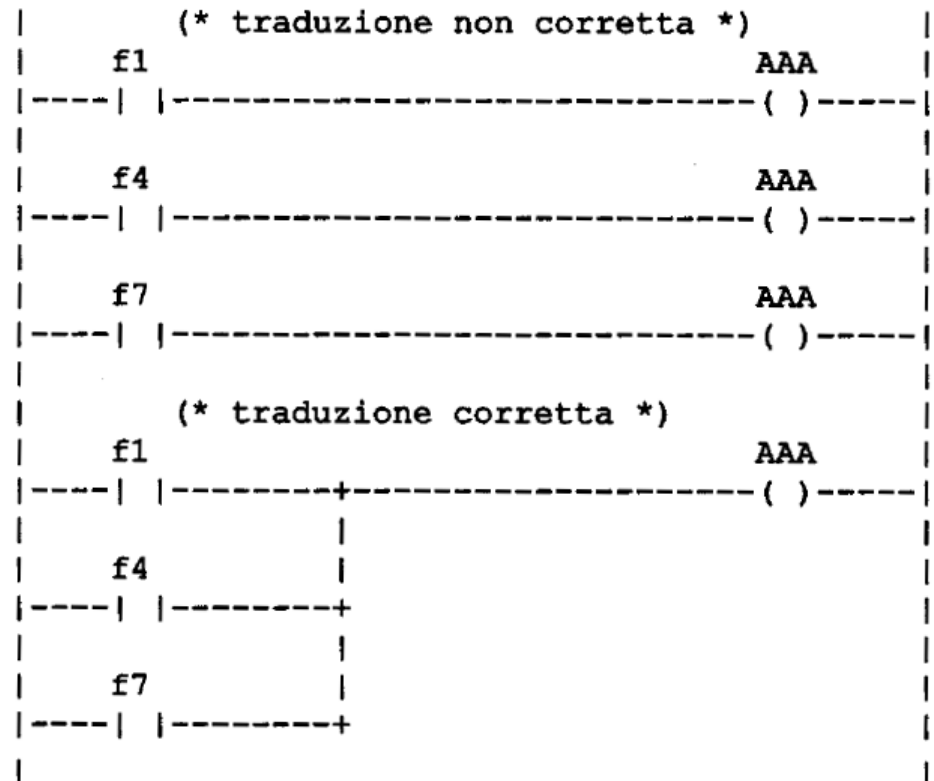
traduzione del diagramma funzionale sequenziale

Codifica dell'algoritmo di evoluzione in linguaggio a contatti



- Problemi legati all'interpretazione del linguaggio a contatti
 - Si supponga che in più fasi distinte debba essere posta ad uno una variabile booleana:

- Se si usasse un rung diverso per ogni fase si avrebbe che la variabile subirebbe alla fine il solo effetto dell'ultimo rung valutato (si veda l'esempio a lato)



traduzione del diagramma funzionale sequenziale

Codifica dell'algoritmo di evoluzione in linguaggio a contatti



- La corretta traduzione prevede che la variabile (in generale l'azione) sia collegata all'**OR logico delle variabili segnalatrici** delle fasi da cui dipende
- SFC costituito da grafi non connessi
 - La costruzione delle quattro sezioni rimane invariata anche nel caso di più grafi non connessi
 - Le sezioni relative ai singoli grafi andranno poi messe insieme nella corretta sequenza per ottenere lo stesso effetto dell'SFC di partenza
 - Nel caso di traduzione di blocchi funzionali, ogni blocco funzionale dovrà essere tradotto singolarmente, producendo codici tradotti distinti

traduzione del diagramma funzionale sequenziale

Codifica dell'algoritmo di evoluzione in linguaggio a contatti



– Relazioni biunivoche

- La codifica presentata introduce relazioni biunivoche tra le fasi, transizioni e azioni dell'SFC e rung del programma tradotto
 - Eventuali modifiche all'SFC possono essere riprese direttamente sul codice tradotto
 - Malfunzionamenti del programma possono essere corretti osservando i valori dei segnalatori per ricavare le fasi dell'SFC in cui si sono verificati i problemi

traduzione del diagramma funzionale sequenziale

Codifica dell'algoritmo di evoluzione in linguaggio a contatti



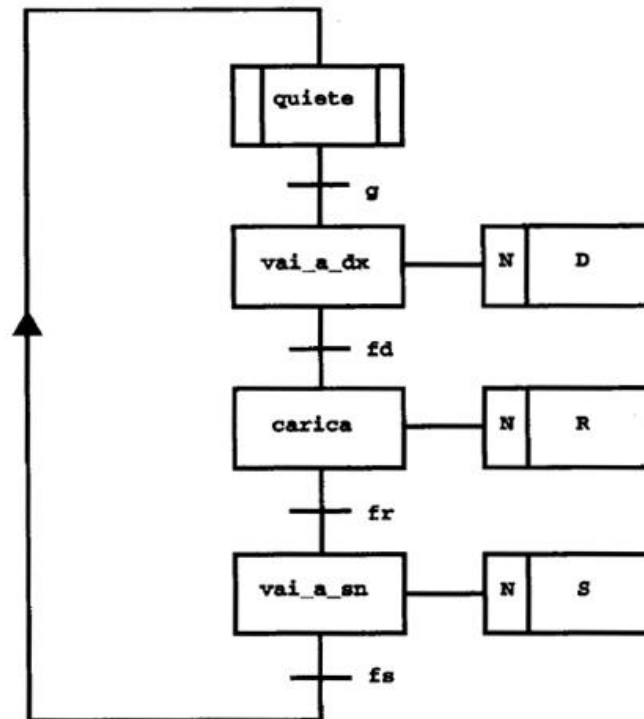
- Lunghezza del programma in linguaggio a contatti
 - È certamente molto più lungo di quello che produrrebbe un programmatore esperto nell'uso del linguaggio a contatti
 - Normalmente non è un limite, considerate le attuali dimensioni delle memorie dei dispositivi di controllo
 - Ad un aumento di lunghezza però non corrisponde un aumento dei tempi di elaborazione
 - Utilizzando i segnalatori delle fasi e delle transizioni come contatti all'inizio dei rung si evita di dover valutare le restanti parti dei rung

traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti



- Esempi di traduzione di SFC in linguaggio a contatti
 - Il carrello automatico
 - SFC da tradurre

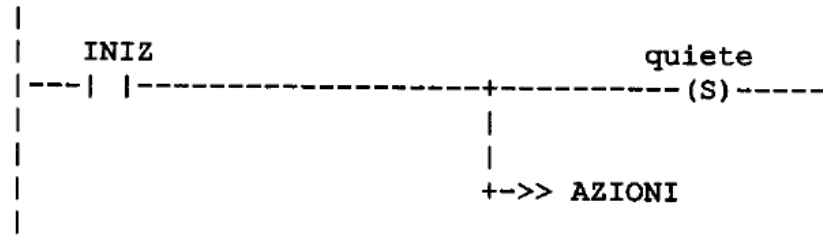


traduzione del diagramma funzionale sequenziale

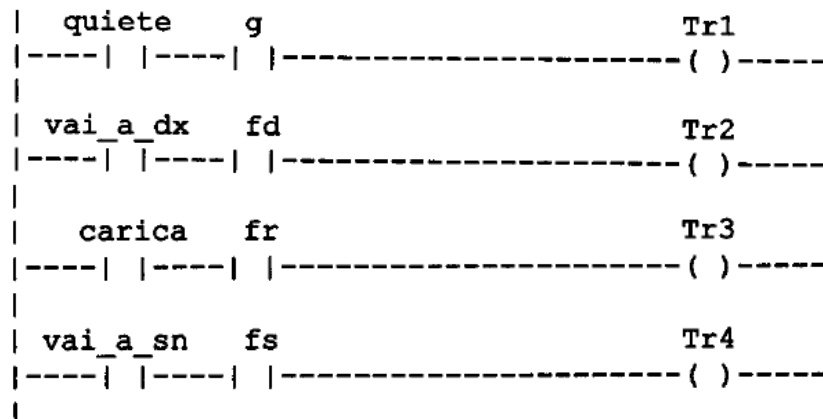
Esempi di traduzione di SFC in linguaggio a contatti



- Sezione di inizializzazione



- Sezione di valutazione delle transizioni

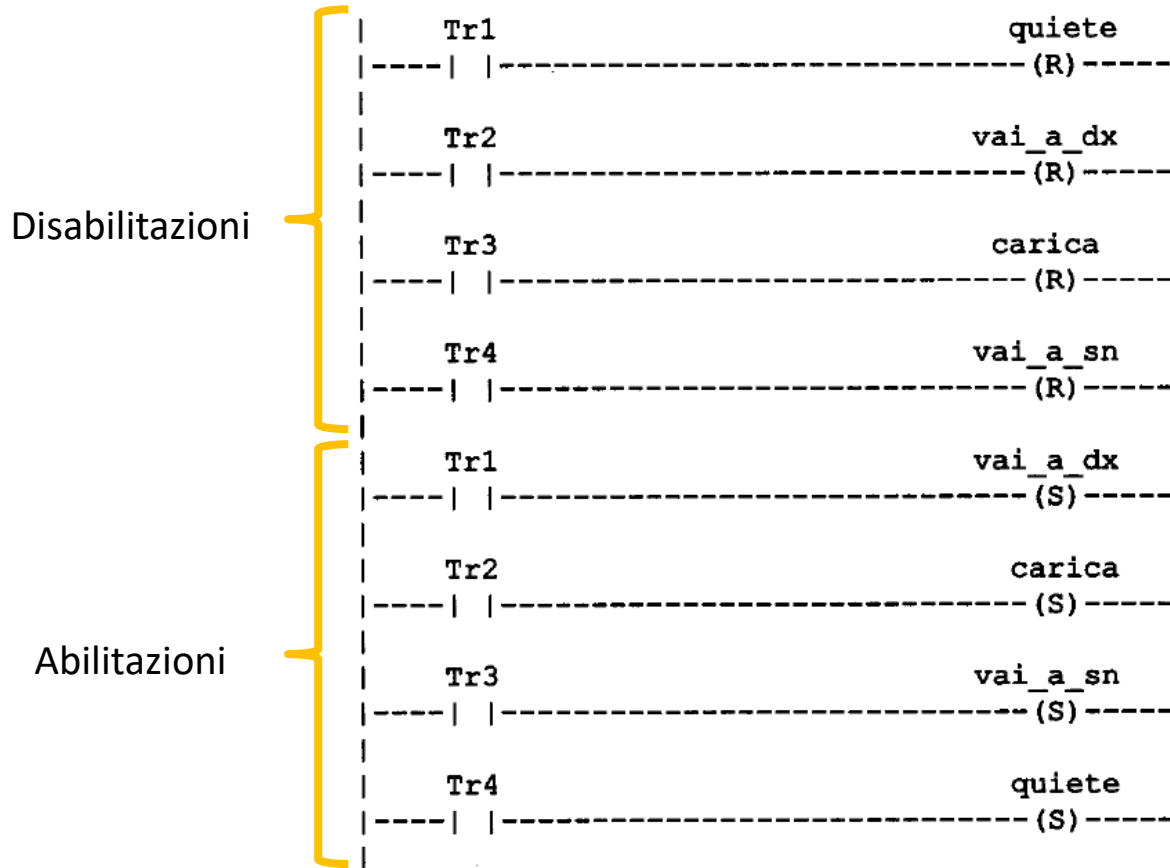


traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti



- Sezione di aggiornamento delle condizioni



traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti



- Sezione di esecuzione delle azioni

```
|  AZIONI:  |
|           |
| vai_a_dx          D  |
|-----| |----- ( )-----|
|           |
| carica            R  |
|-----| |----- ( )-----|
|           |
| vai_a_sn          S  |
|-----| |----- ( )-----|
|           |
```

- ## traduzione del diagramma funzionale sequenziale

traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti



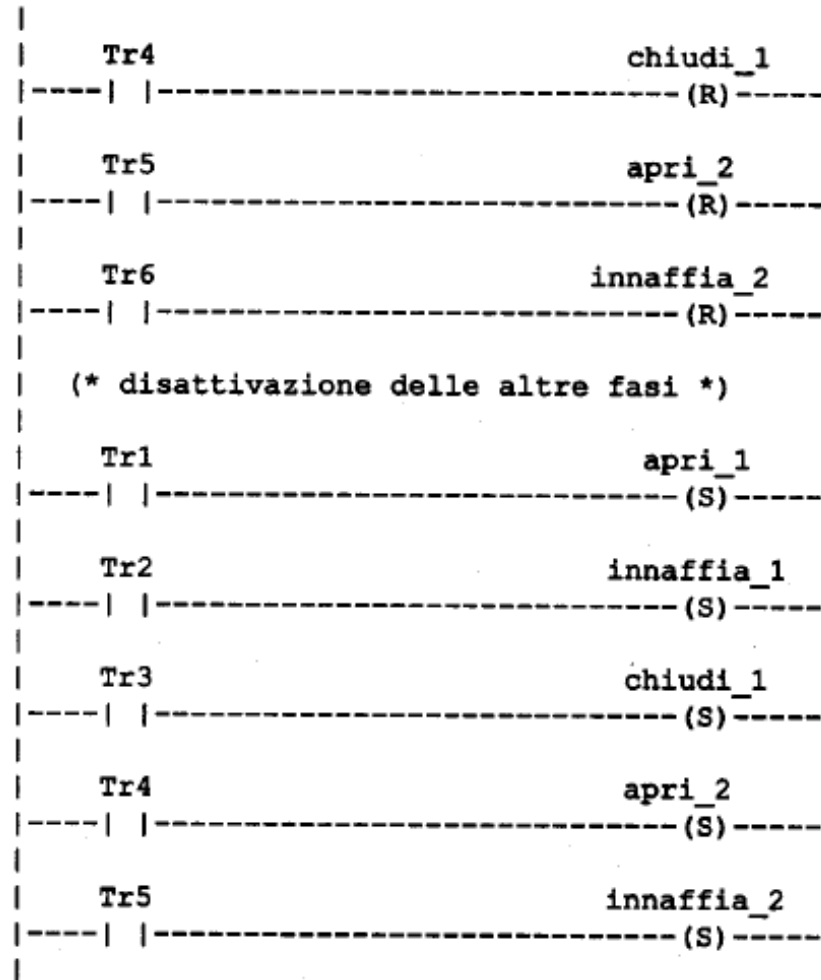
```
|
| chiudi_1 timer4.Q                               Tr4 |
|-----| |-----| |-----| ( ) -----|
|
| apri_2 timer5.Q                                 Tr5 |
|-----| |-----| |-----| ( ) -----|
|
| innaffia_2 timer6.Q                             Tr6 |
|-----| |-----| |-----| ( ) -----|
|
| chiudi_2 timer7.Q                               Tr7 |
|-----| |-----| |-----| ( ) -----|
|
|
| (* valutazione altre transizioni *)
|
```

- Sezione di aggiornamento della condizione

```
|
| Tr1                                             quiete |
|-----| |-----| (R) -----|
|
| Tr2                                             apri_1 |
|-----| |-----| (R) -----|
|
| Tr3                                             innaffia_1 |
|-----| |-----| (R) -----|
|
```

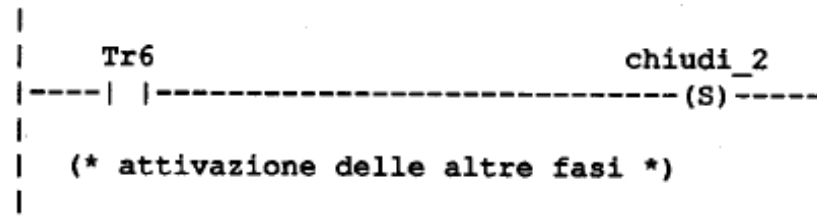

traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti

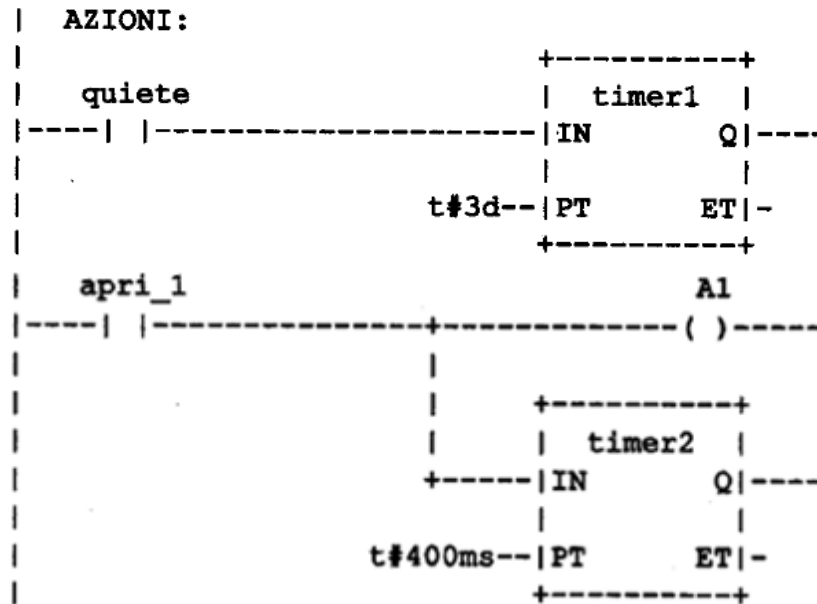


traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti

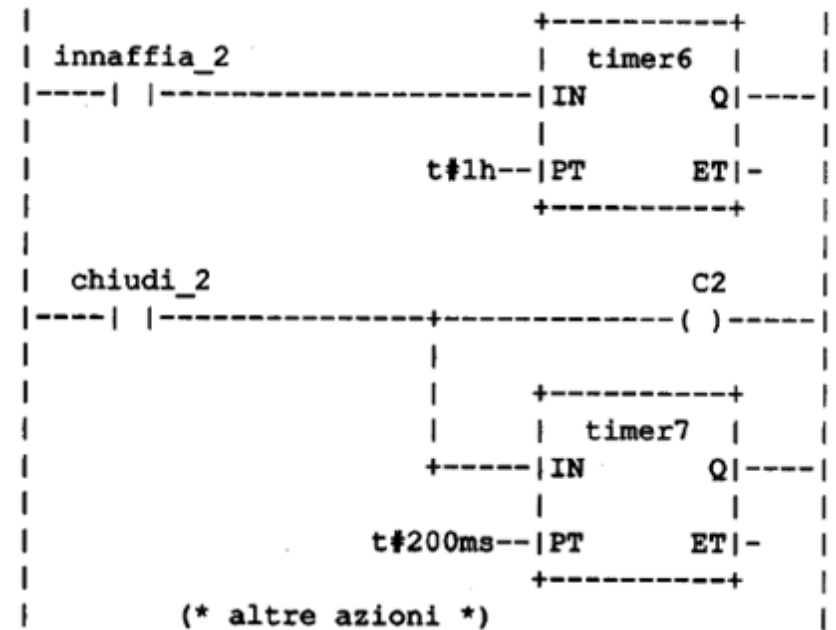
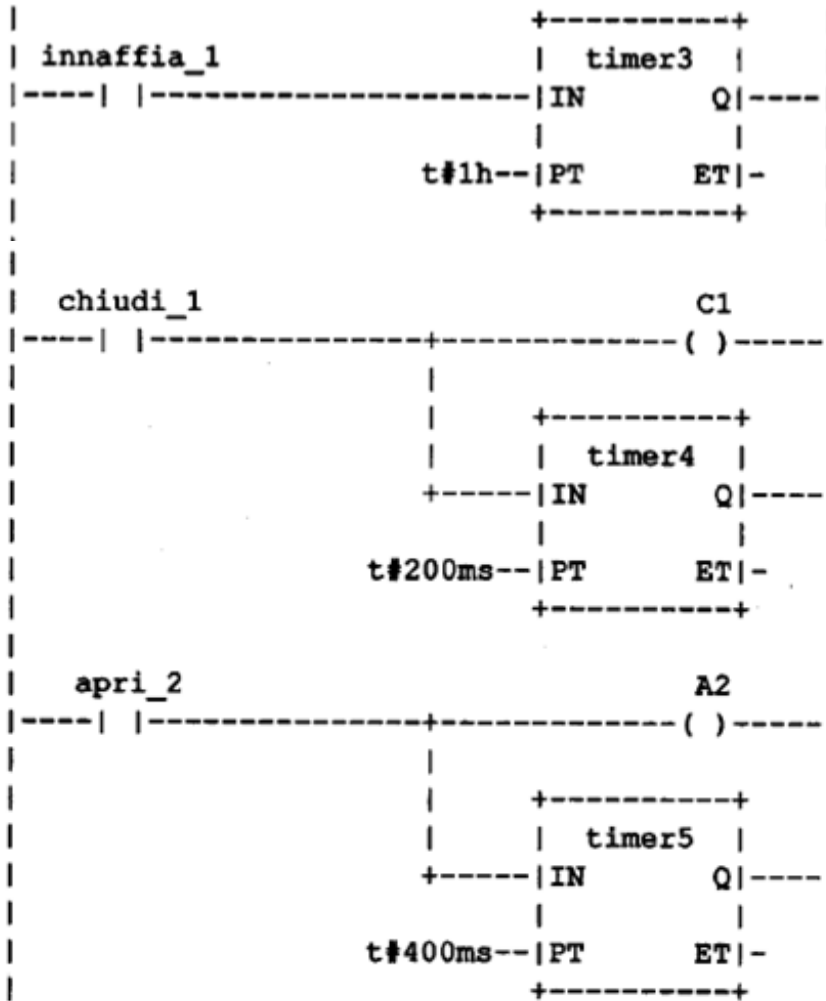


- Sezione di esecuzione delle azioni



traduzione del diagramma funzionale sequenziale

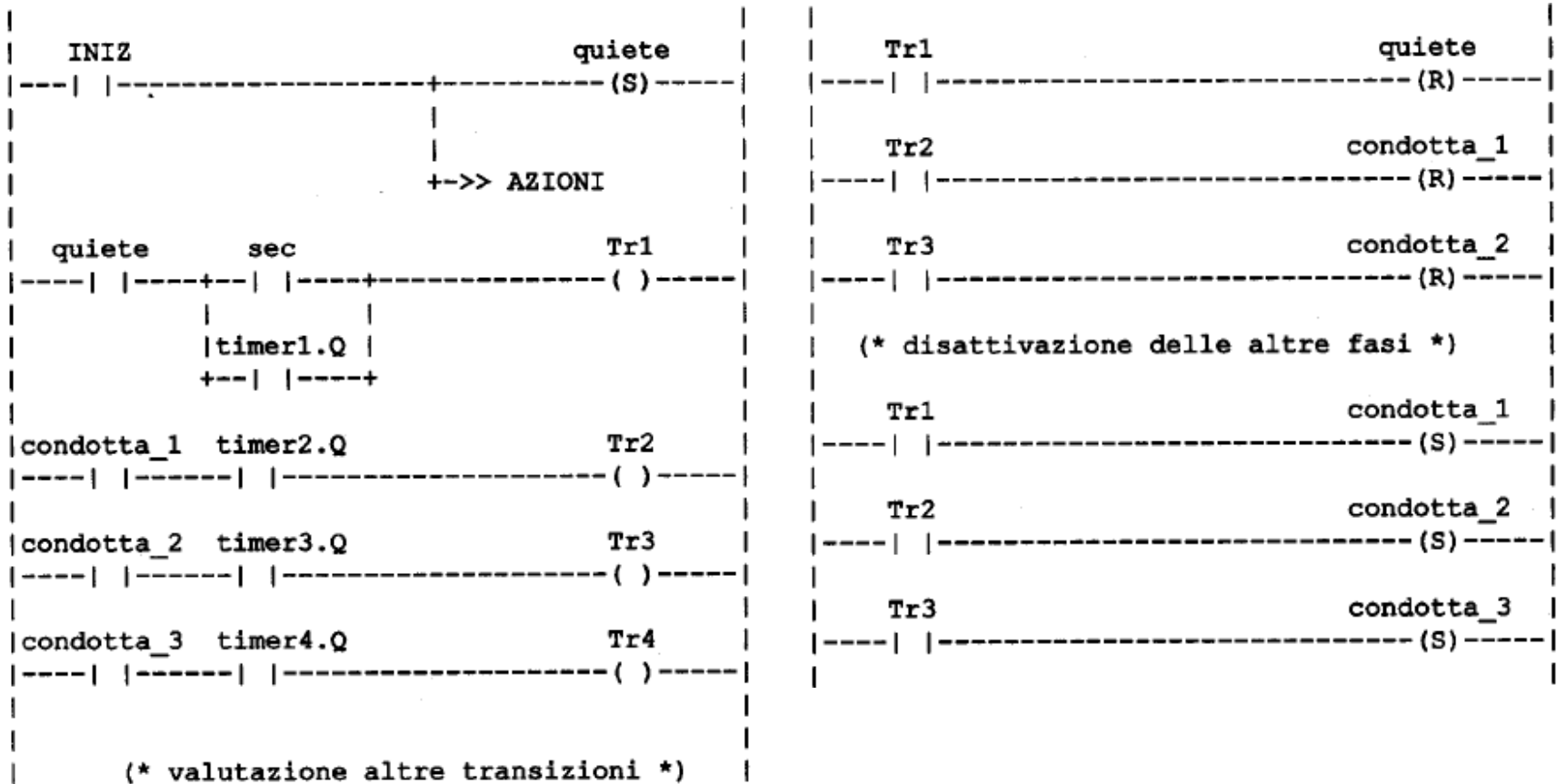
Esempi di traduzione di SFC in linguaggio a contatti



traduzione del diagramma funzionale sequenziale

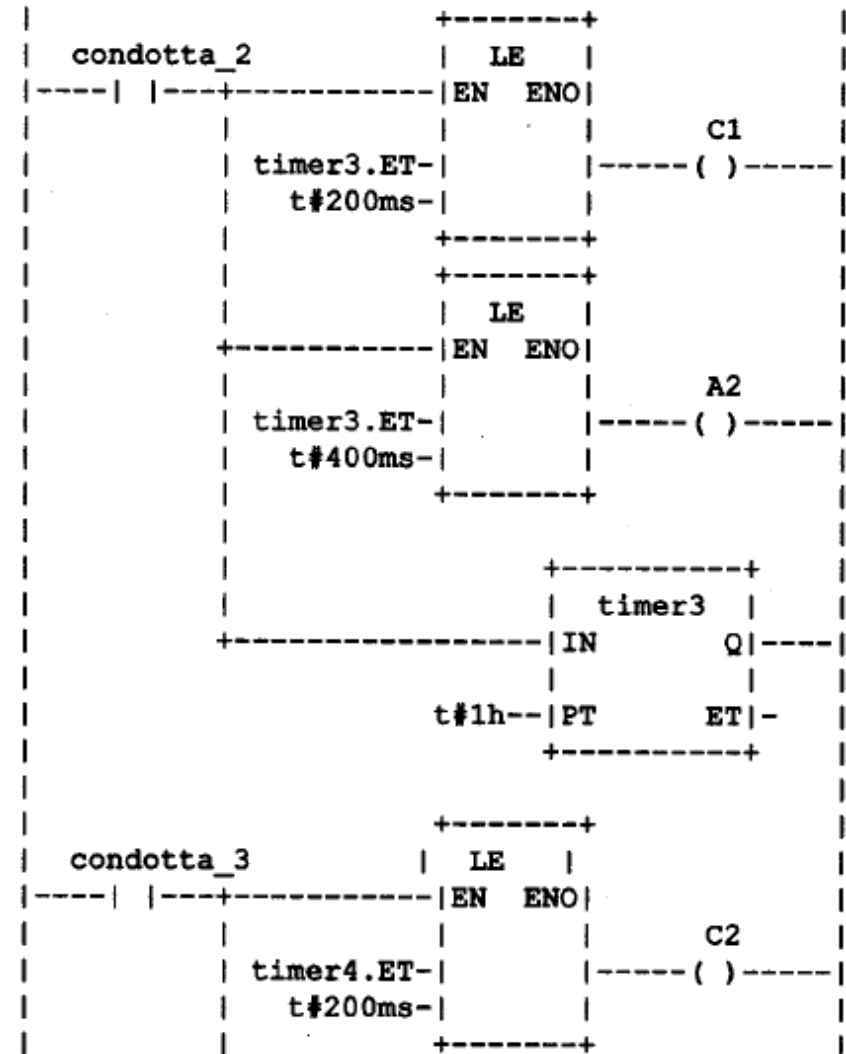
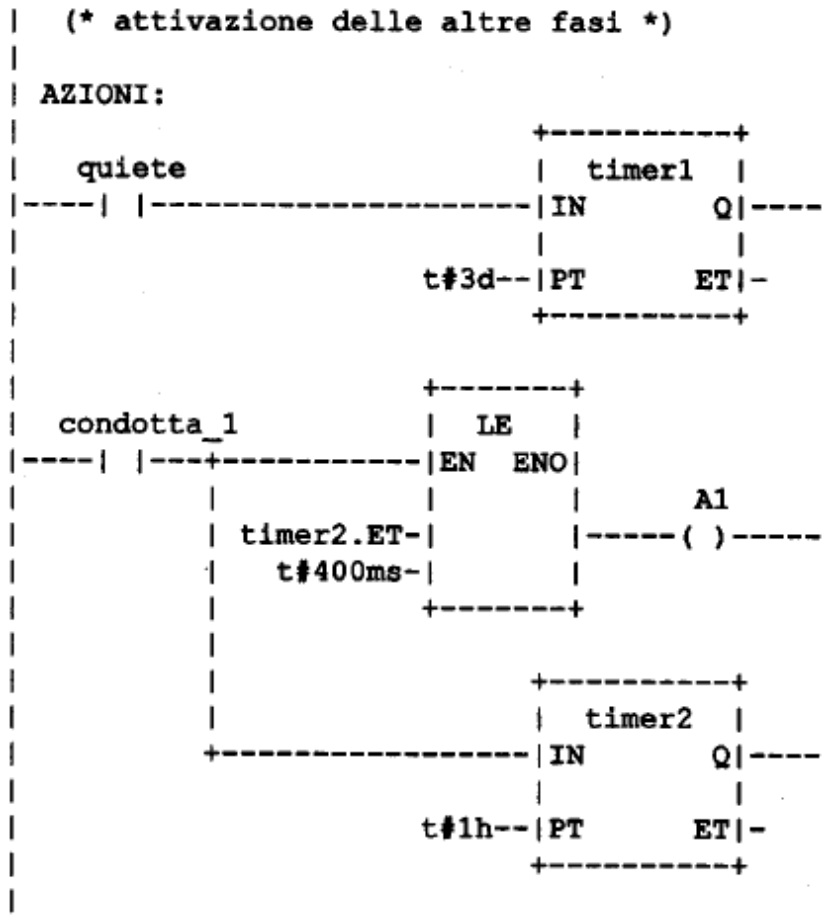
Esempi di traduzione di SFC in linguaggio a contatti

- Traduzione del programma alternativo



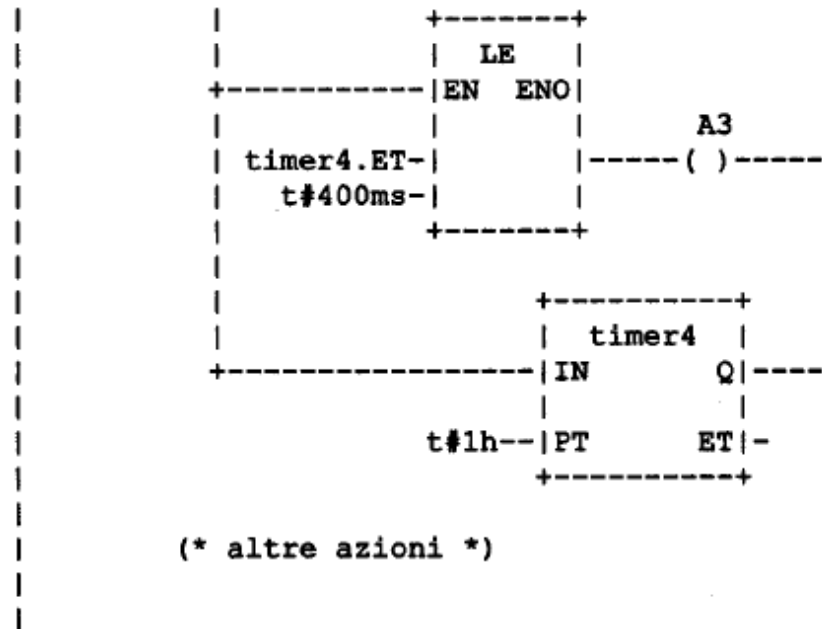
traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti



traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti

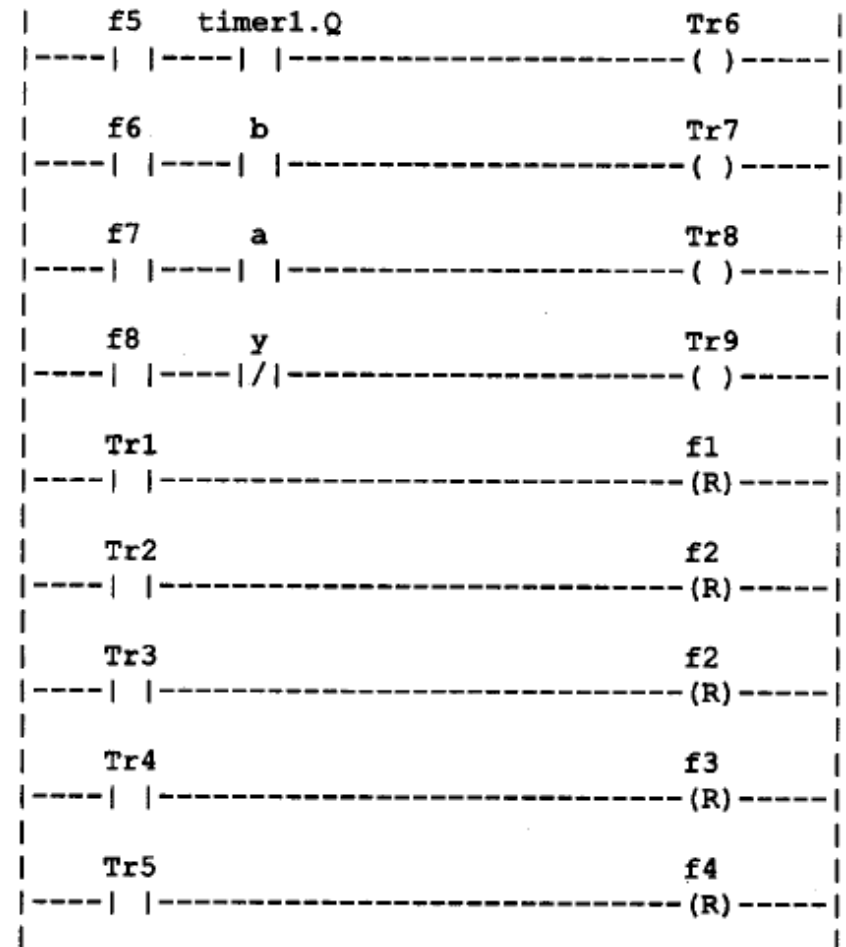
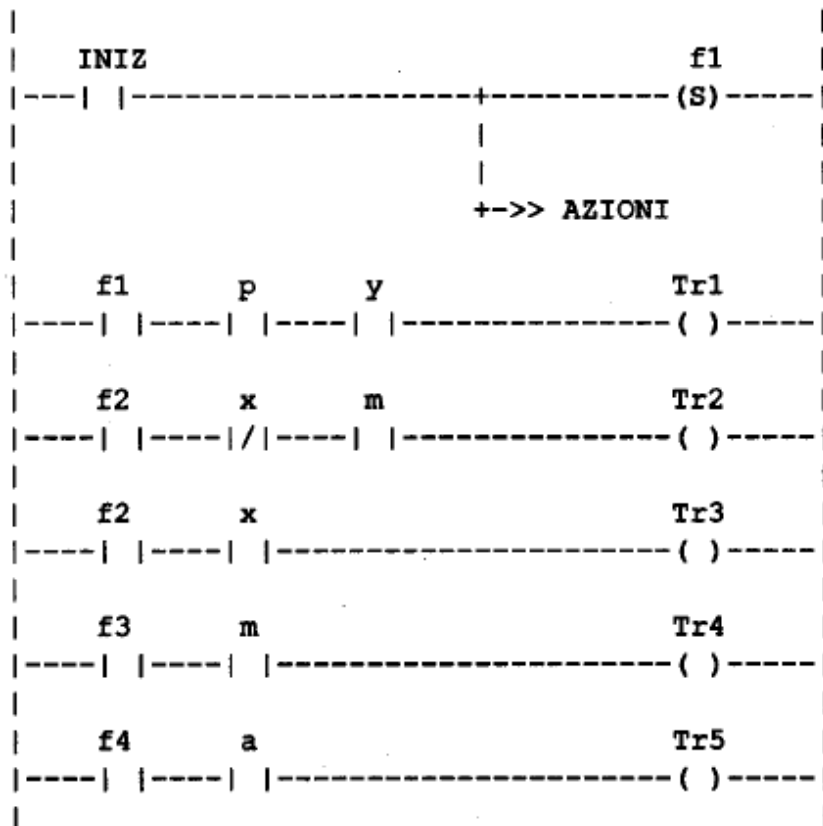


traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti

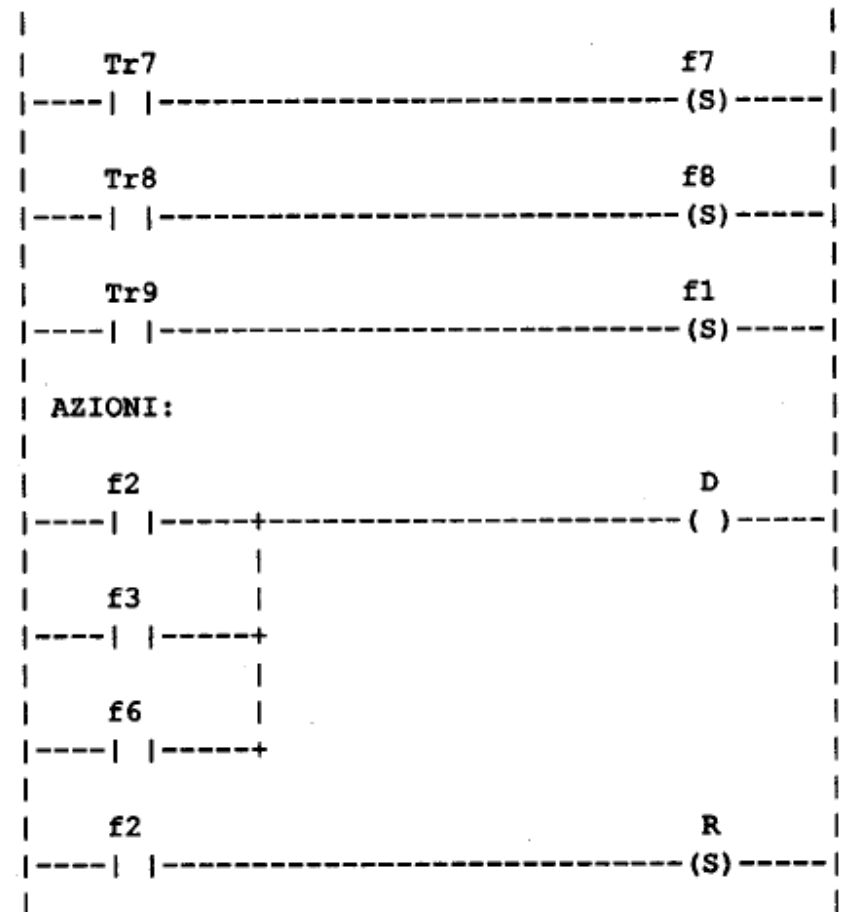
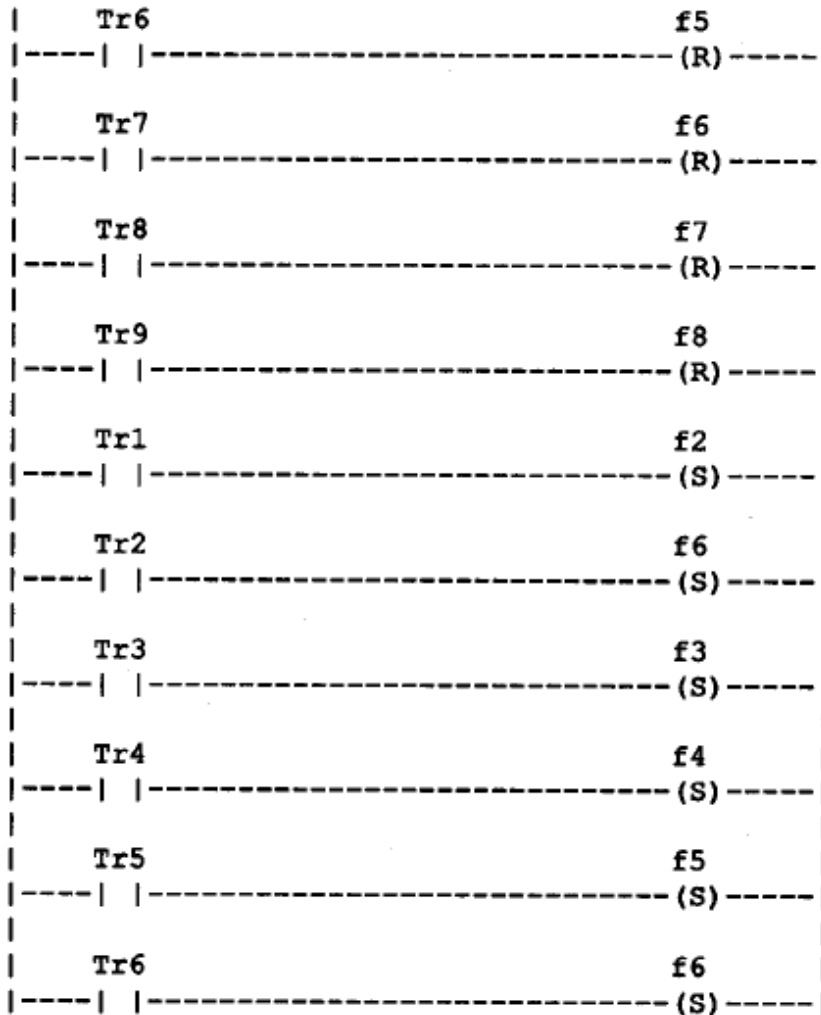


– Il trapano automatico



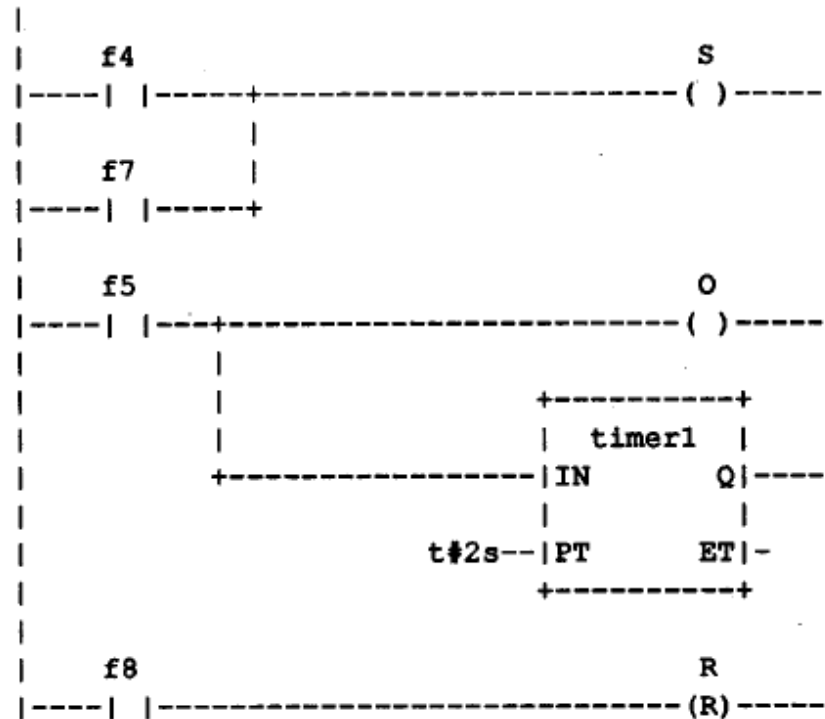
traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti



traduzione del diagramma funzionale sequenziale

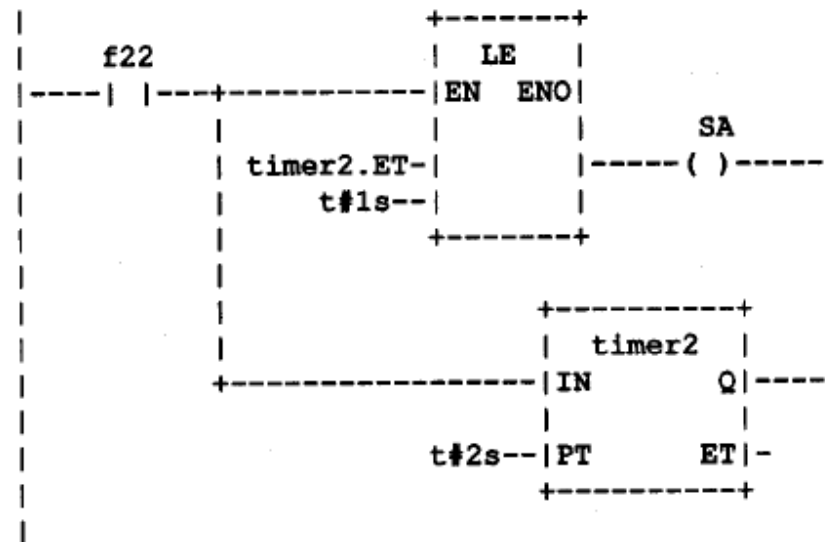
Esempi di traduzione di SFC in linguaggio a contatti





traduzione del diagramma funzionale sequenziale

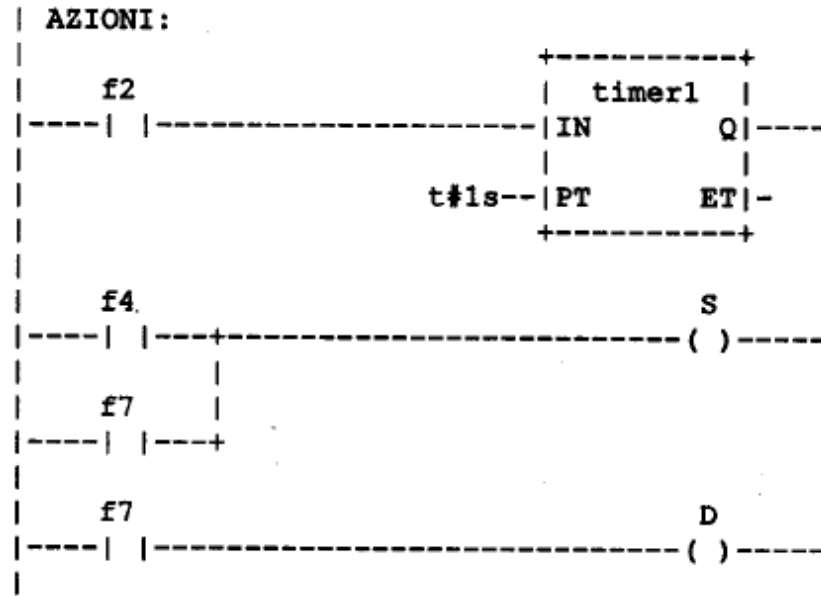
Esempi di traduzione di SFC in linguaggio a contatti





traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti

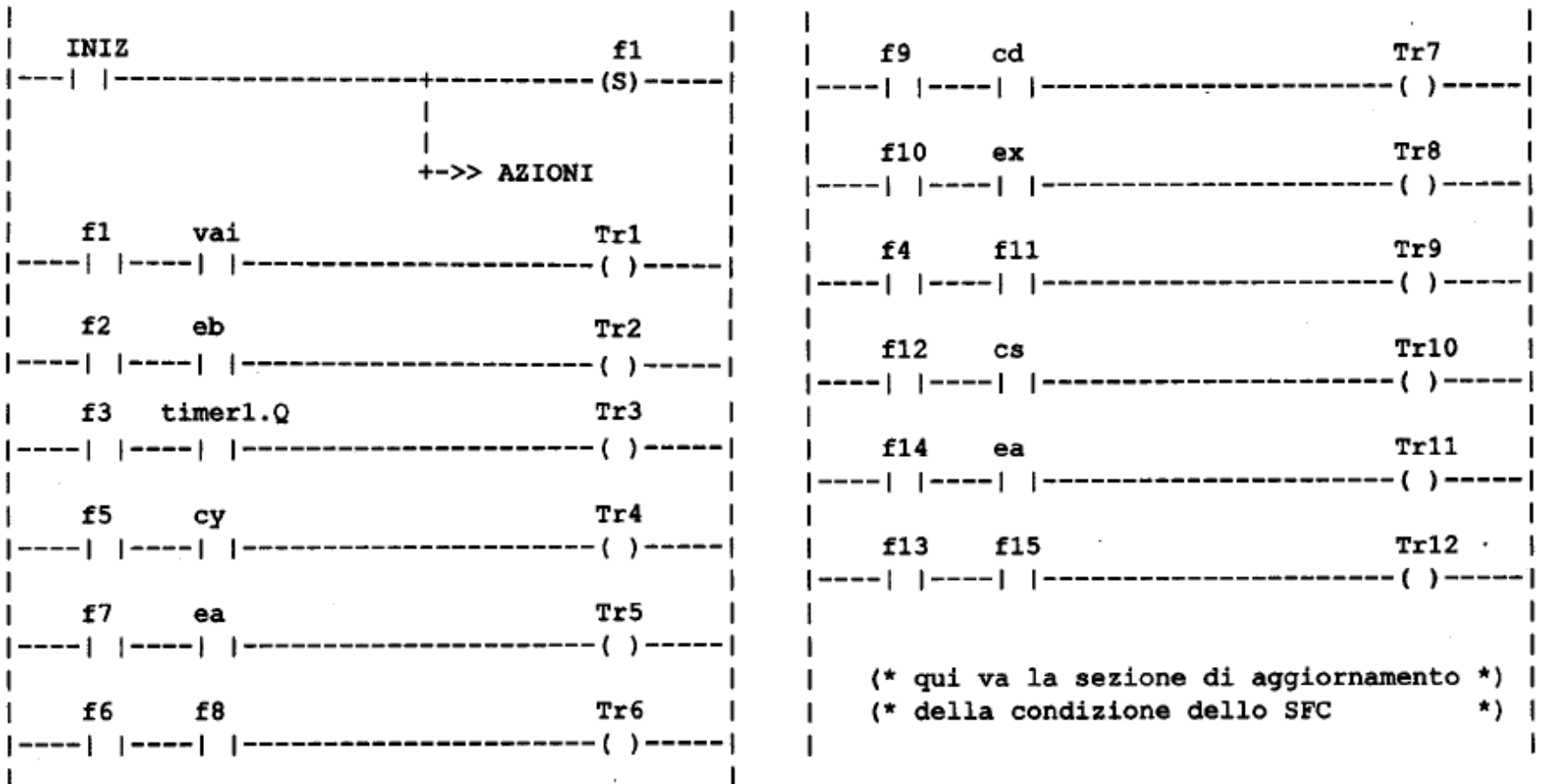


traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti

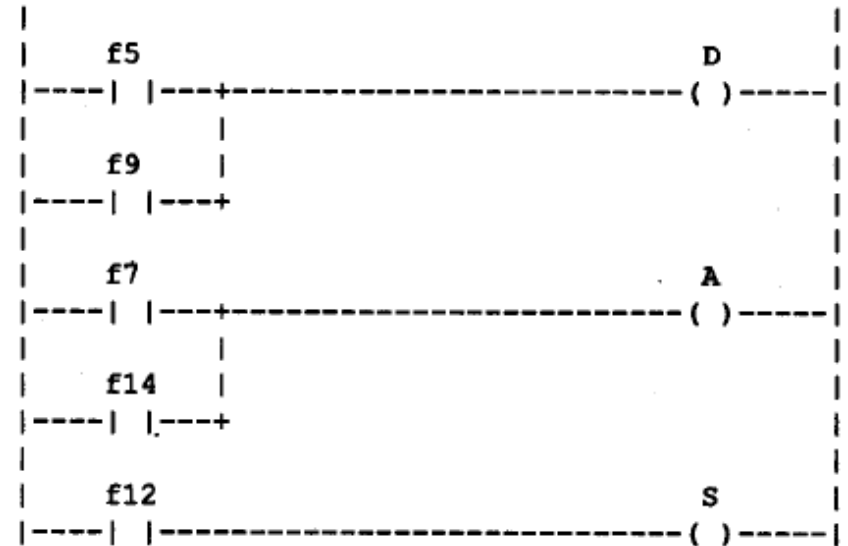
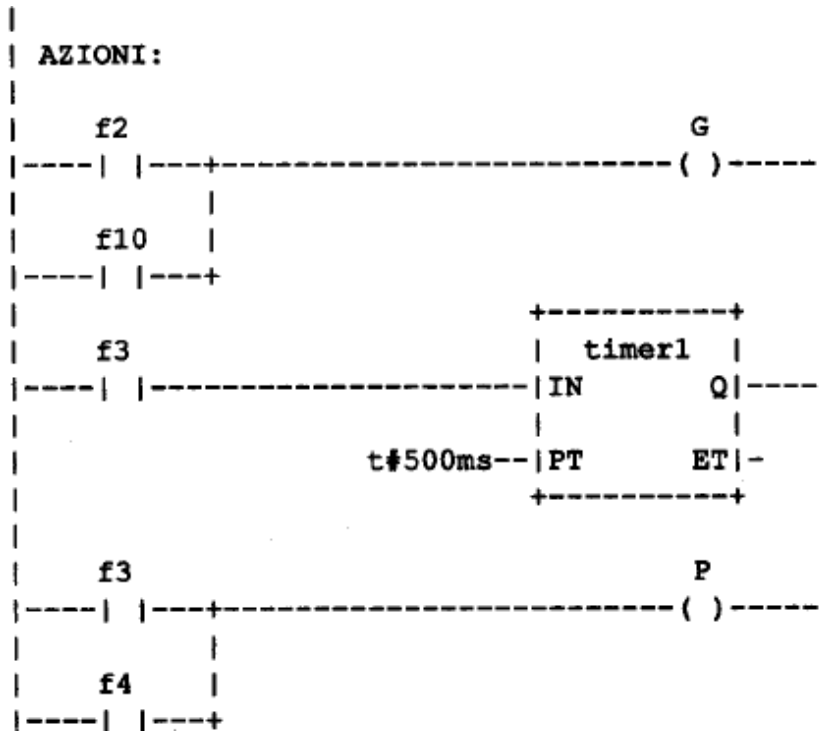


– Il carro ponte a elettrocalamita



traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti

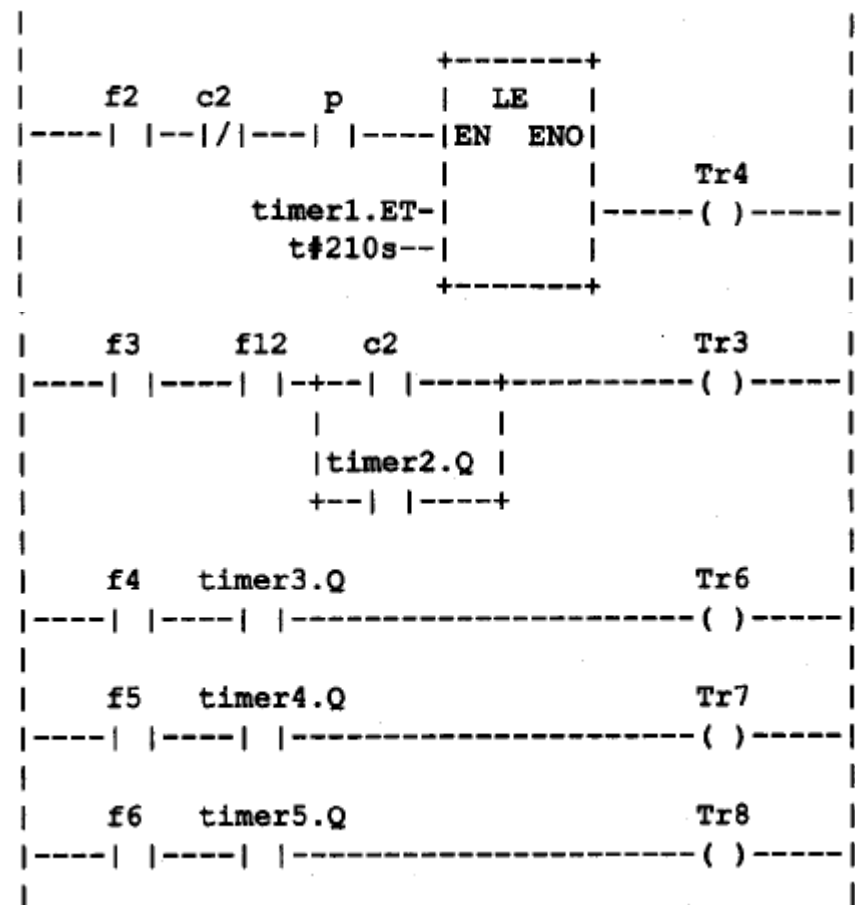
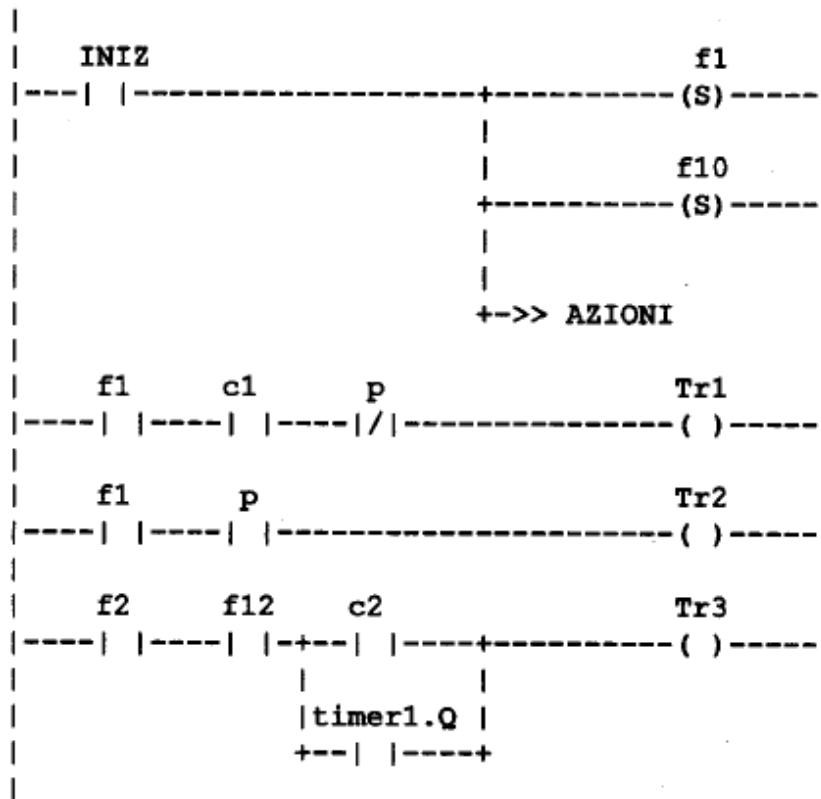


traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti

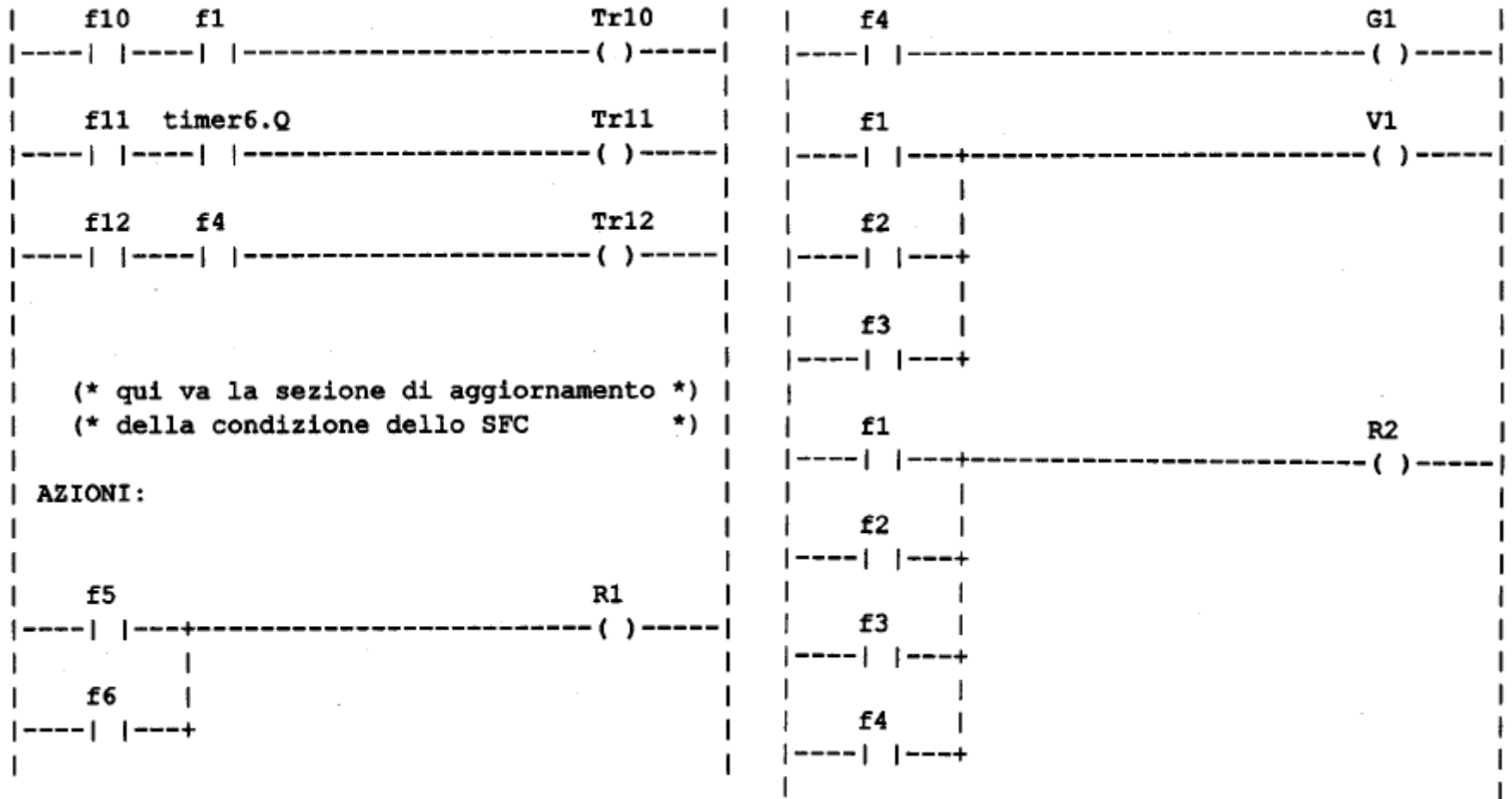


– Il sistema semaforico



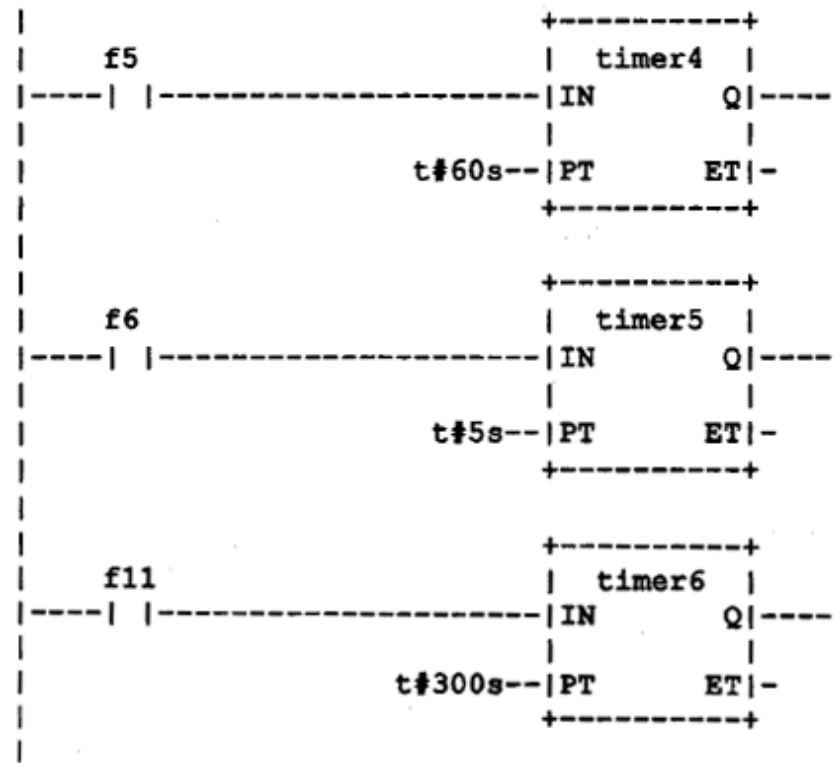
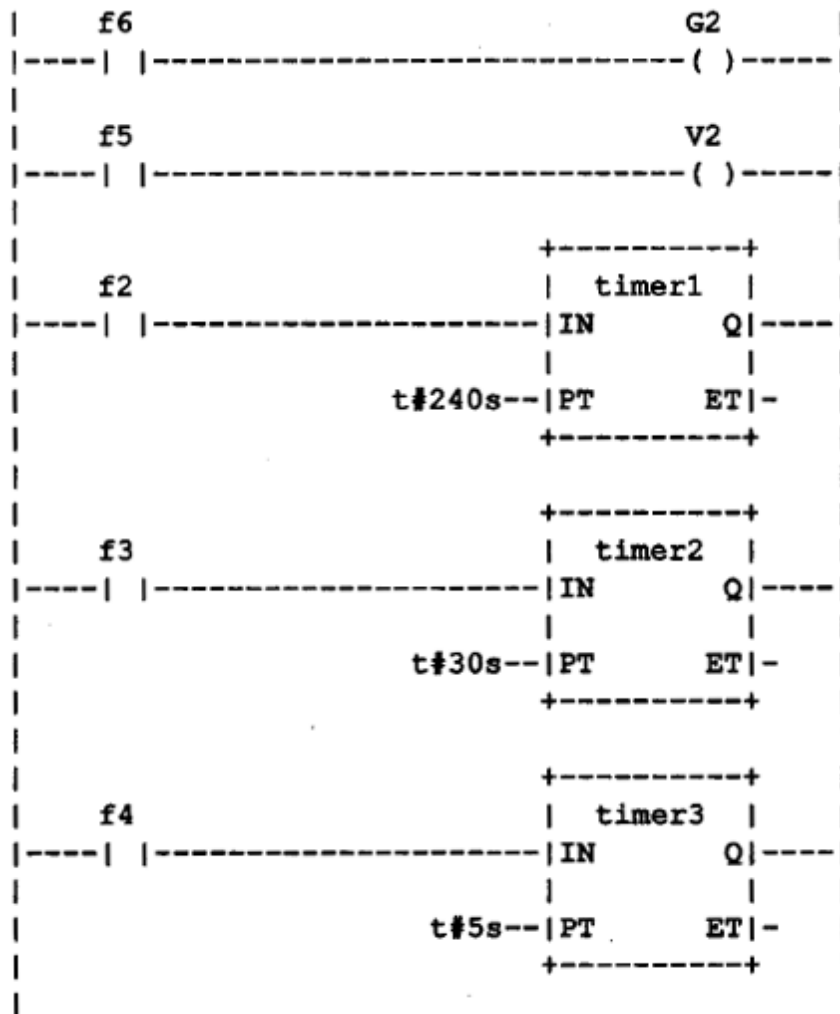
traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti



traduzione del diagramma funzionale sequenziale

Esempi di traduzione di SFC in linguaggio a contatti



macroazioni

Introduzione



- **Macroazione**

- Azione operata da un SFC che ha effetti sulla condizione di un altro SFC
 - Agisce modificando le variabili segnalatrici dello stato di un SFC
 - È un concetto di recente introduzione **non previsto dallo standard**
- Consente di introdurre delle **gerarchie** tra gli SFC
 - L'SFC che esegue la macroazione avrà una posizione nella gerarchia superiore a quello che viene influenzato dalla macroazione
- Si definiranno due macroazioni:
 - Sospensione
 - Forzatura

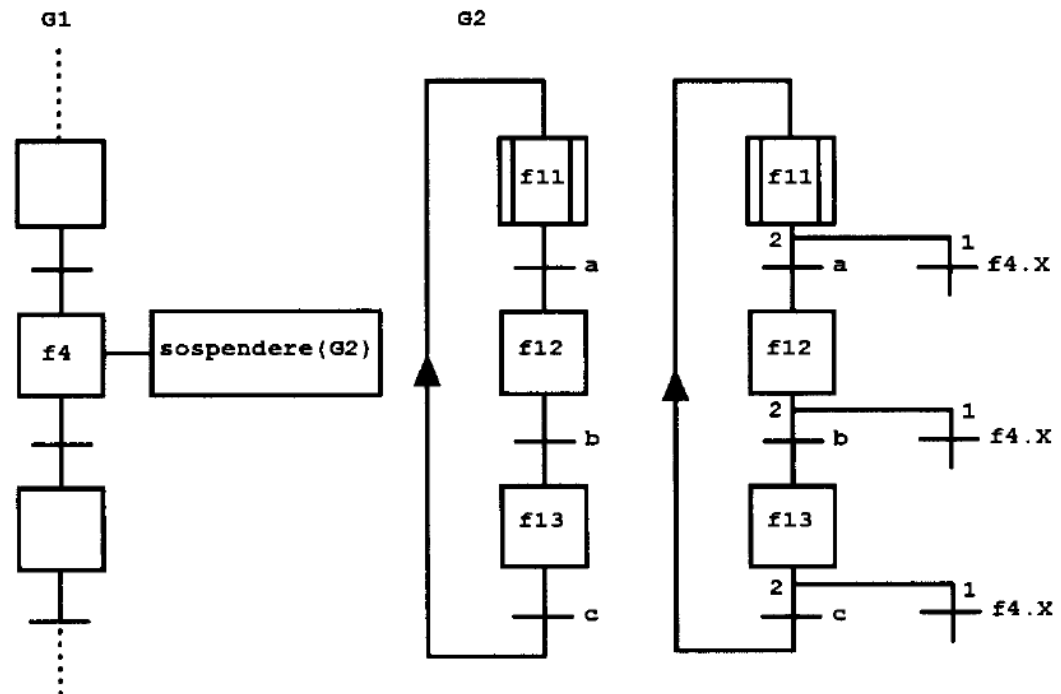
macroazioni

Sospensione

– Sospensione

- Consente di rendere inattive tutte le fasi di un altro SFC
- *Esempio:* SOSPENDERE (nome_SFC)

Nota: equivale ad aggiungere una transizione superabile con la massima priorità in uscita a tutte le fasi dell'SFC che deve essere sospeso



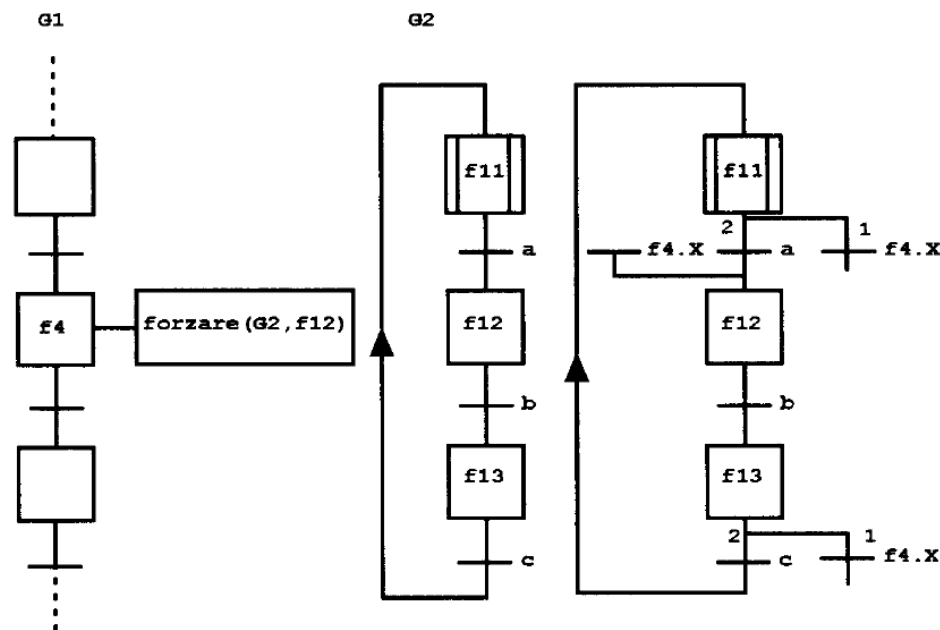
macroazioni

Forzatura

– Forzatura

- Consente ad un SFC di livello gerarchico superiore di forzare quelli di livello inferiore ad assumere una certa condizione (l'insieme delle fasi attive) a prescindere dalle regole di evoluzione
- *Esempio* : FORZARE (nome_SFC, fase_1_da_attivare, ..., fase_n_da_attivare)

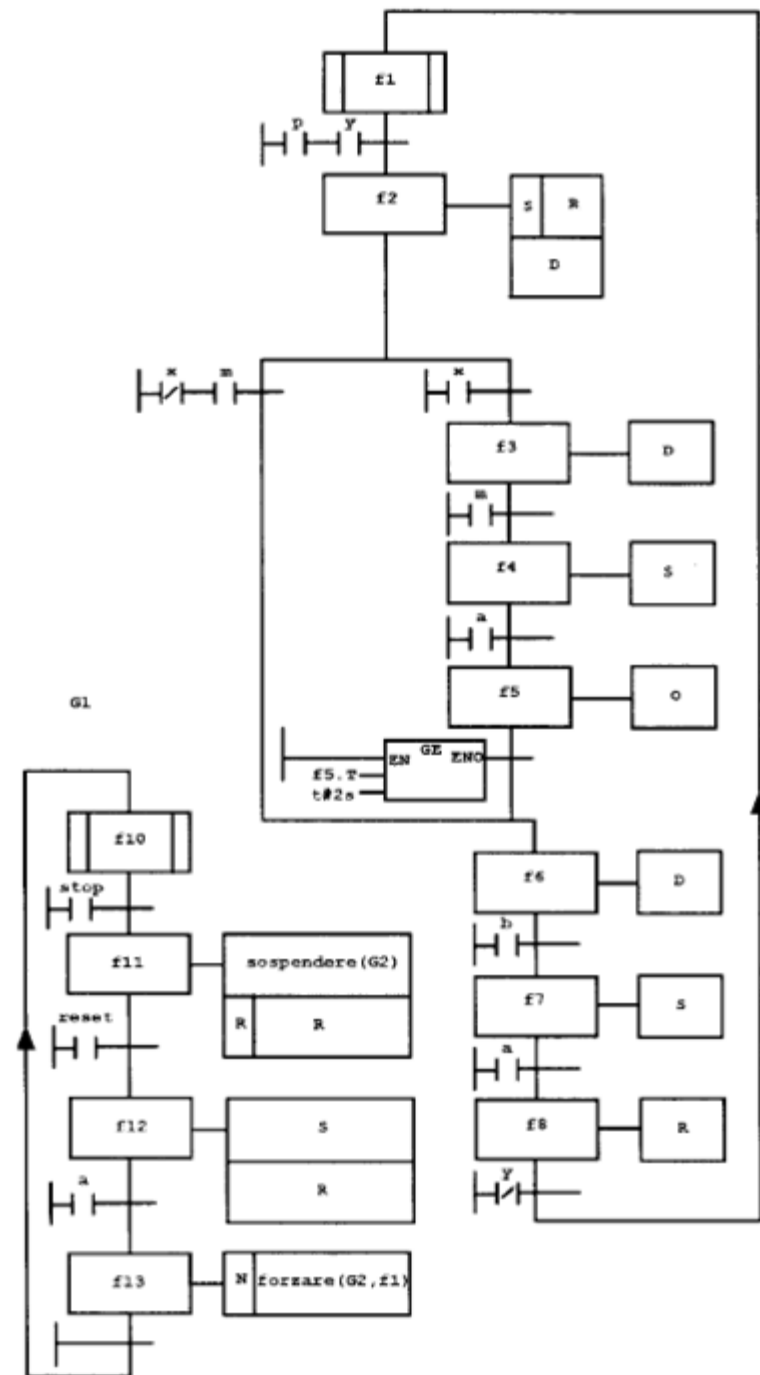
Nota: equivale ad aggiungere una transizione superabile con la massima priorità in uscita a tutte le fasi che non devono diventare attive ed una in entrata a tutte quelle che lo devono diventare



macroazioni

Esempio – Trapano automatico

- *Esempio*: trapano automatico con segnale di emergenza
 - Si considera il problema del controllo del trapano automatico già presentato a cui si vuole aggiungere la gestione di un segnale di arresto di emergenza
 - L'operatore ha a disposizione in segnale stop per bloccare il trapano ed un segnale reset per rimettere il sistema nelle condizioni iniziali
 - *Nota*: si noti come le **azioni a memorizzazione** siano state **disattivate esplicitamente**





macroazioni

Traduzione in linguaggio a contatti

– Traduzione delle macroazioni in linguaggio a contatti

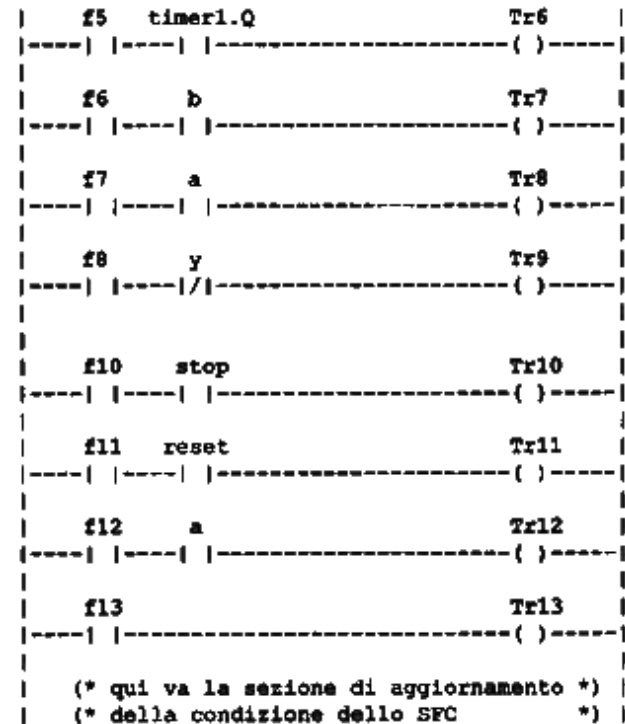
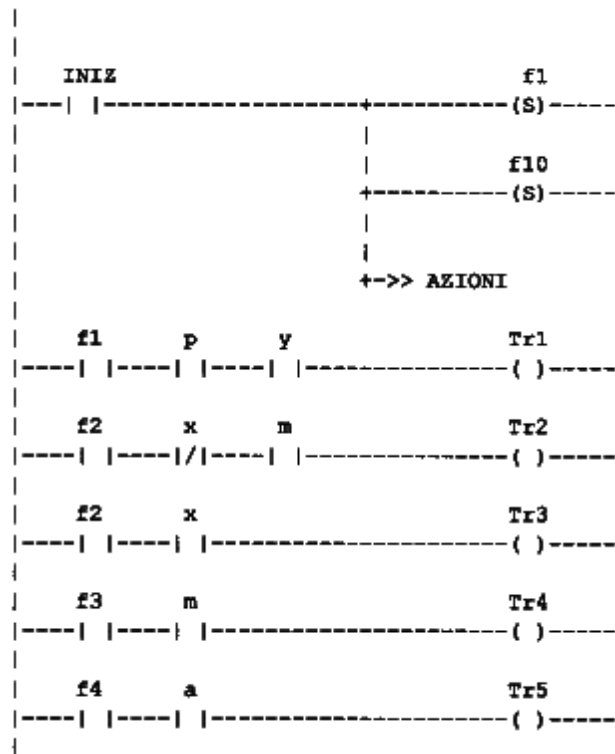
- La traduzione delle macroazione va fatta come per tutte le azioni nella sezione di esecuzione delle azioni
- **La codifica della macroazione deve precedere quella delle azioni degli SFC di livello gerarchico inferiore**
 - In questo modo l'esecuzione della macroazione annulla l'eventuale evoluzione della condizione dell'SFC controllato avutasi nella sezione di aggiornamento della condizione



macroazioni

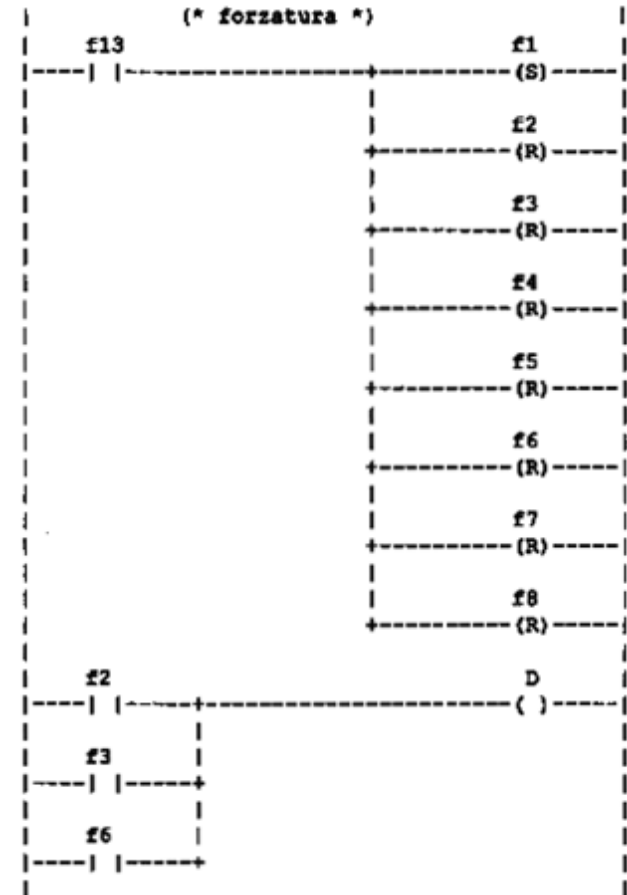
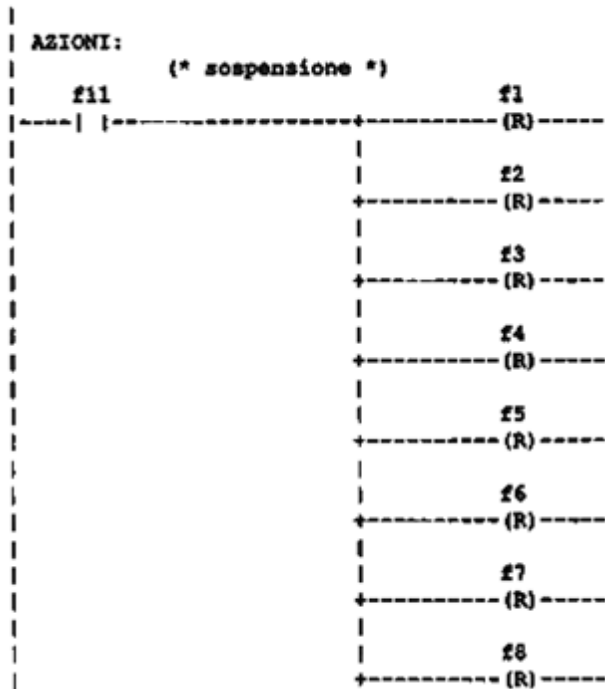
Traduzione in linguaggio a contatti

- *Esempio*: traduzione dell'SFC di controllo del trapano automatico con segnale di emergenza



macroazioni

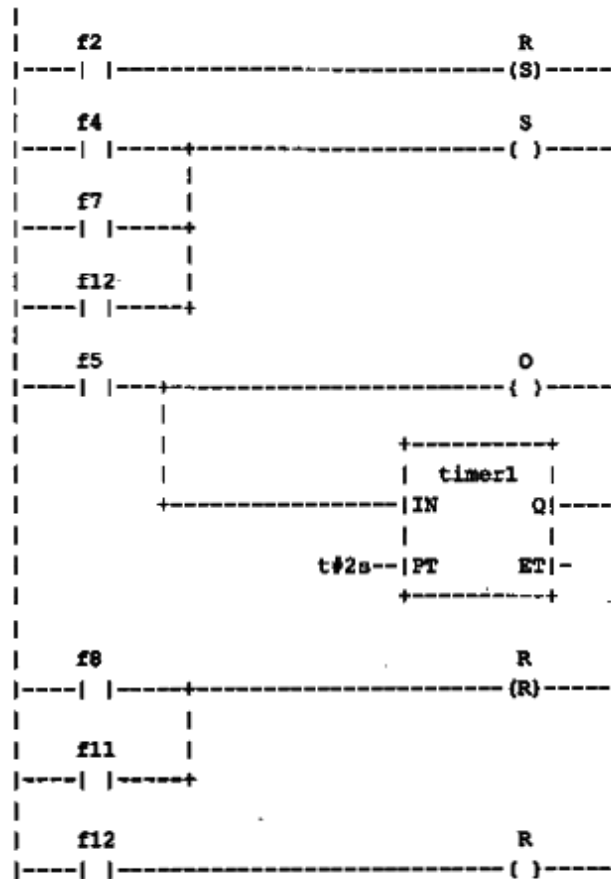
Traduzione in linguaggio a contatti





macroazioni

Traduzione in linguaggio a contatti

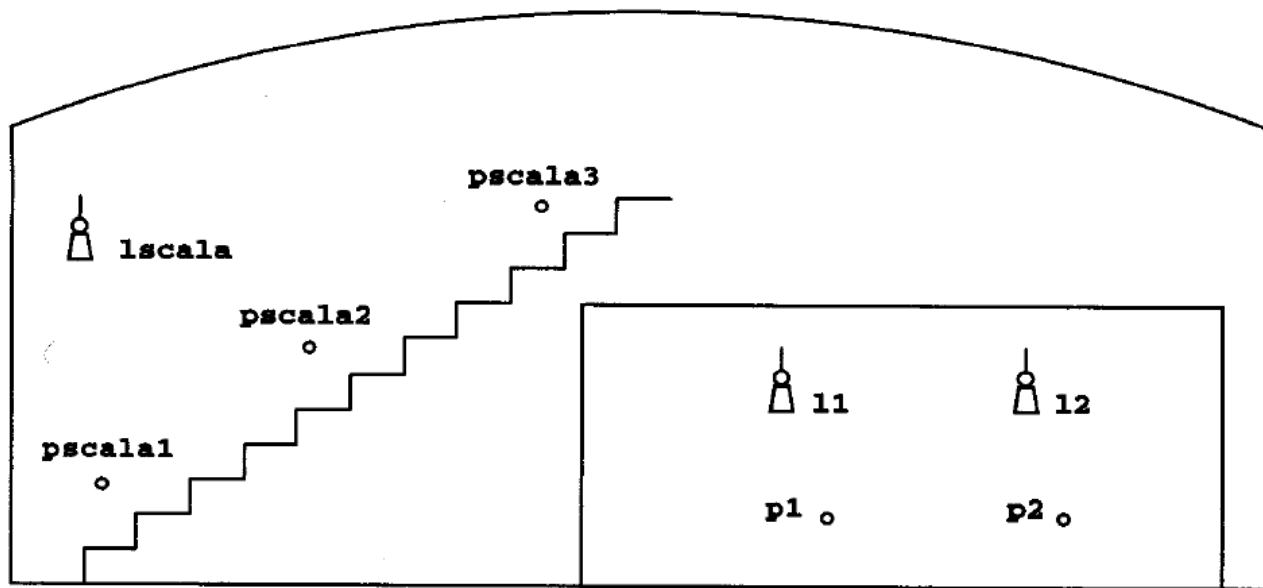


esempi di programmazione

Introduzione



- Esempi di programmazione
 - Controllo sistema di illuminazione



- Si vogliono controllare i sistemi di illuminazione di due capannoni identici, composta da una scala e un ambiente di lavoro



esempi di programmazione

Introduzione

- Segnali di uscita, di tipo digitale:
 - `lsca la`, controlla l'accensione delle luci delle scale
 - `l1`, `l2`, controllano le luci nell'ambiente di lavoro
- Segnali di ingresso, di tipo digitale:
 - `pscala1`, `pscala2`, `pscala3`, pulsanti di comando per le luci della scala
 - `p1`, `p2`, pulsanti di comando per le luci dell'ambiente di lavoro
- Specifiche:
 - le luci della scala si devono accendere alla pressione di uno dei pulsanti di accensione e rimanere accese per 60 secondi
 - se durante l'accensione delle luci della scala viene premuto di nuovo uno dei pulsanti il conteggio del tempo deve ripartire
 - se viene premuto il pulsante `p1` si deve accendere l'impianto `l1` nell'ambiente di lavoro

esempi di programmazione

Introduzione



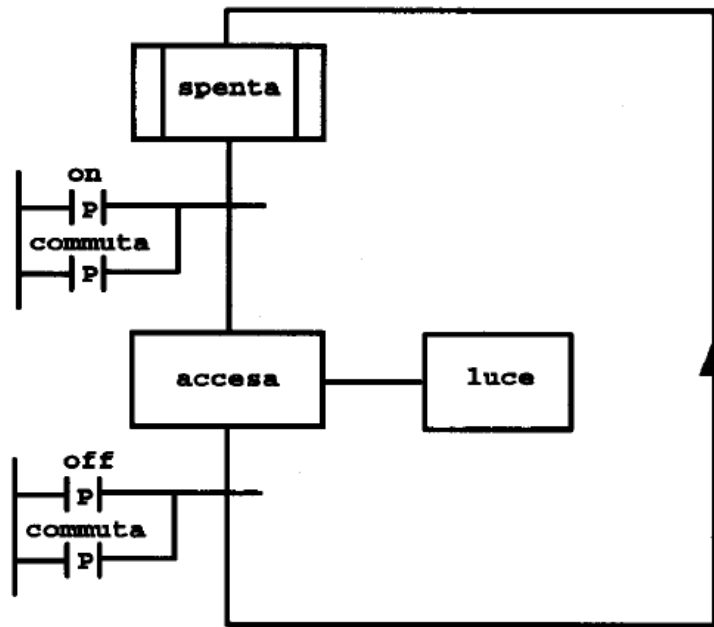
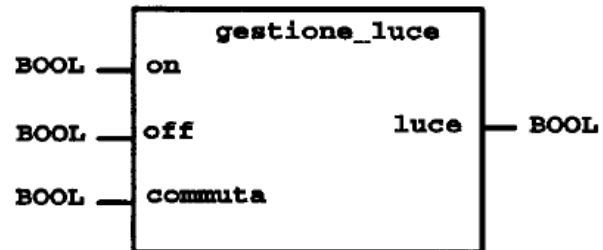
- se viene premuto il pulsante p2 si deve accendere l'impianto l2
- se uno dei due pulsanti p1 o p2 viene premuto due volte entro mezzo secondo, si devono accendere tutti e due gli impianti l1 e l2
- se uno dei due pulsanti p1 o p2 viene tenuto premuto per più di mezzo secondo, si devono spegnere tutti e due gli impianti l1 e l2
- Soluzione proposta:
 - Si utilizzeranno i blocchi funzionali per separare il problema in **funzionalità elementari**, ovvero in parti più piccole → **POU riusabili**
 - Gestione di un impianto luminoso generico (gestione_luci):
 - » Blocco funzionale in grado di accendere (porre allo stato logico 1), spegnere (porre allo stato logico 0) o cambiare lo stato di una uscita
 - » Fronte di salita su on → Luce viene posta nello stato 1
 - » Fronte di salita su off → Luce viene posta nello stato 0
 - » Fronte di salita su commuta → Luce cambia stato

esempi di programmazione

Introduzione



.FUNCTION_BLOCK



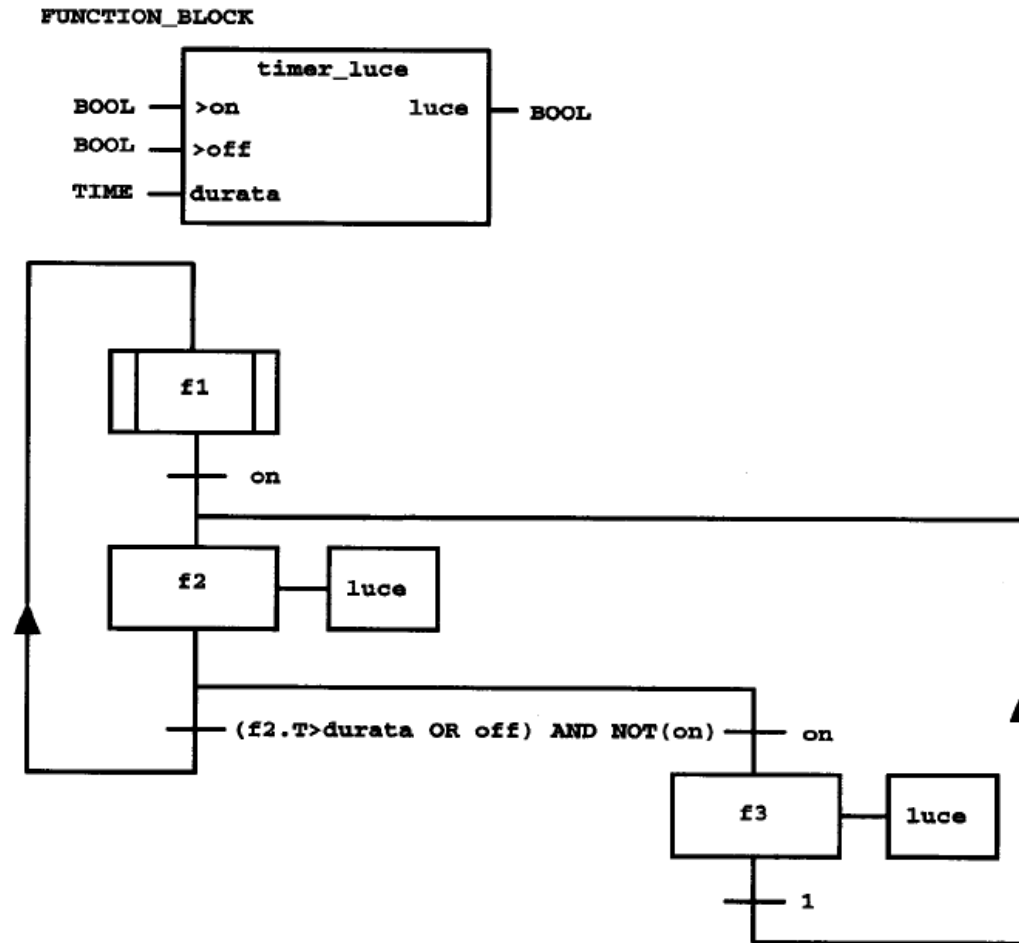
END_FUNCTION_BLOCK

esempi di programmazione

Introduzione



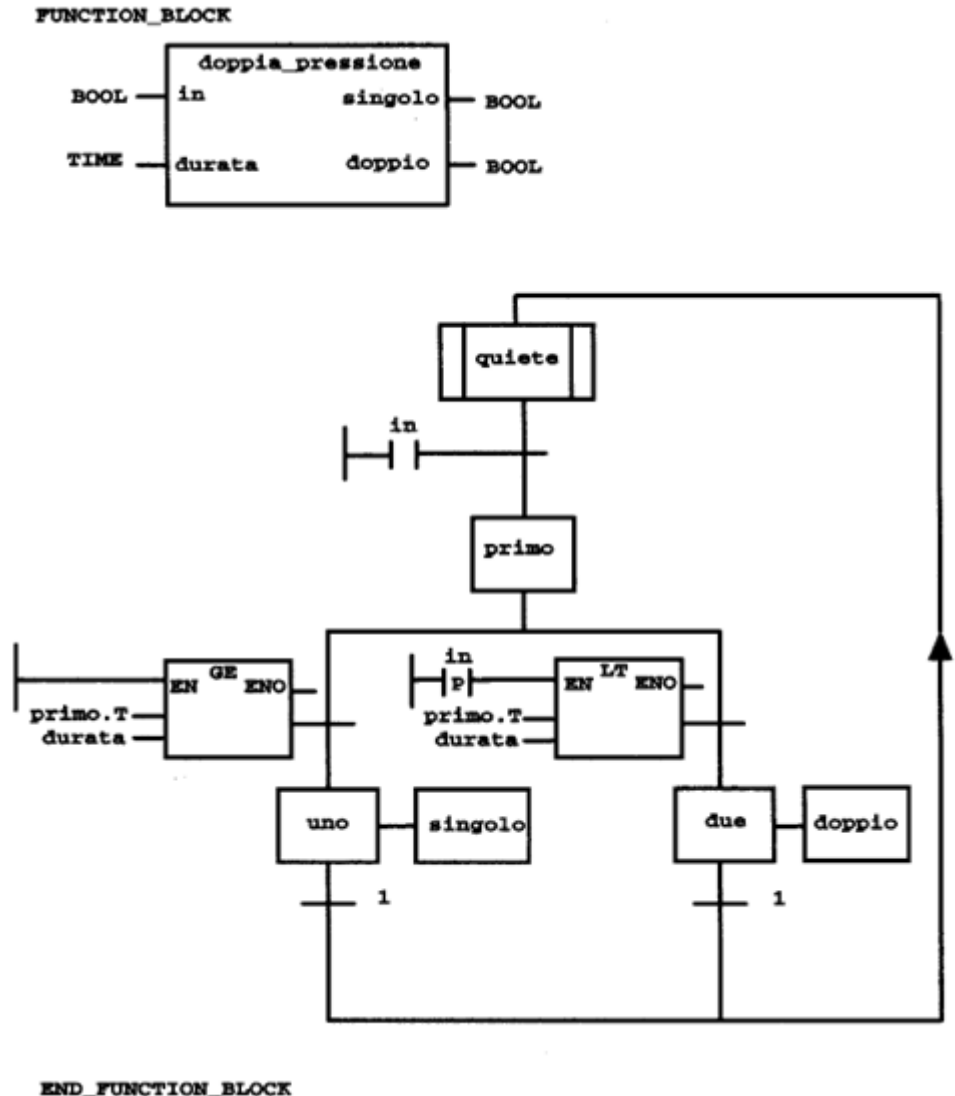
- Accensione temporizzata di un impianto luminoso (timer_luce)



esempi di programmazione

Introduzione

- Riconoscimento di una singola o una doppia pressione su un pulsante (doppia_pressione)
 - » durata indica l'intervallo di tempo a partire dalla prima pressione entro cui deve arrivare la seconda

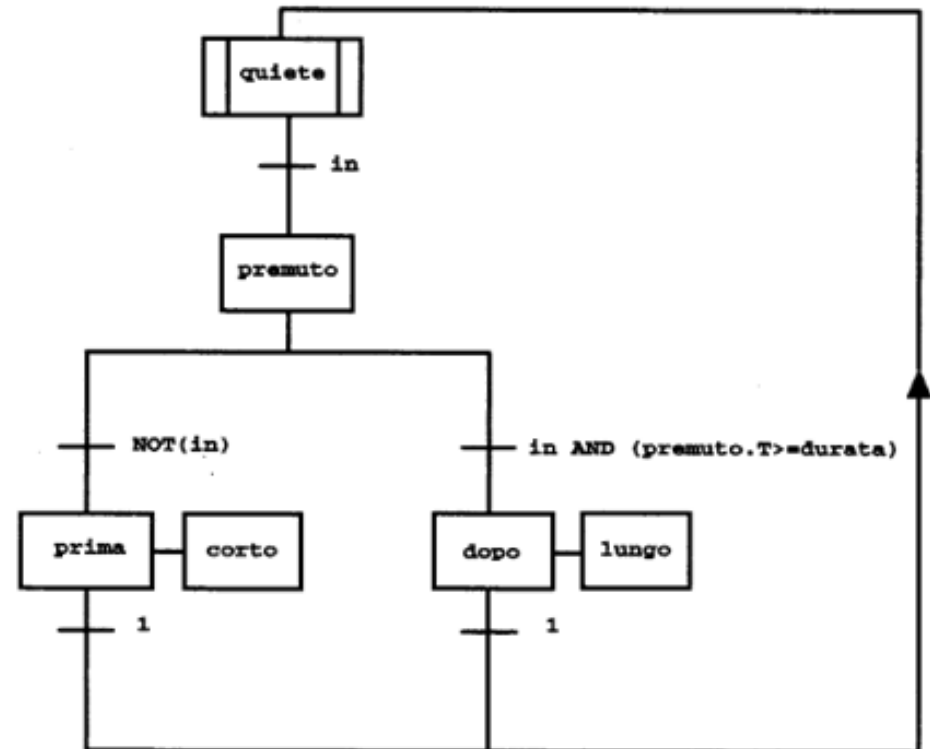
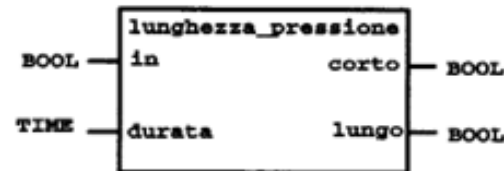


esempi di programmazione

Introduzione

- Riconoscimento di una pressione continua prolungata su un pulsante (lunghezza_pressione)
 - » durata indica l'intervallo di tempo che discrimina se la pressione è di durata breve (uscita corto) o lunga (uscita lungo)

FUNCTION_BLOCK



END_FUNCTION_BLOCK

esempi di programmazione

Introduzione

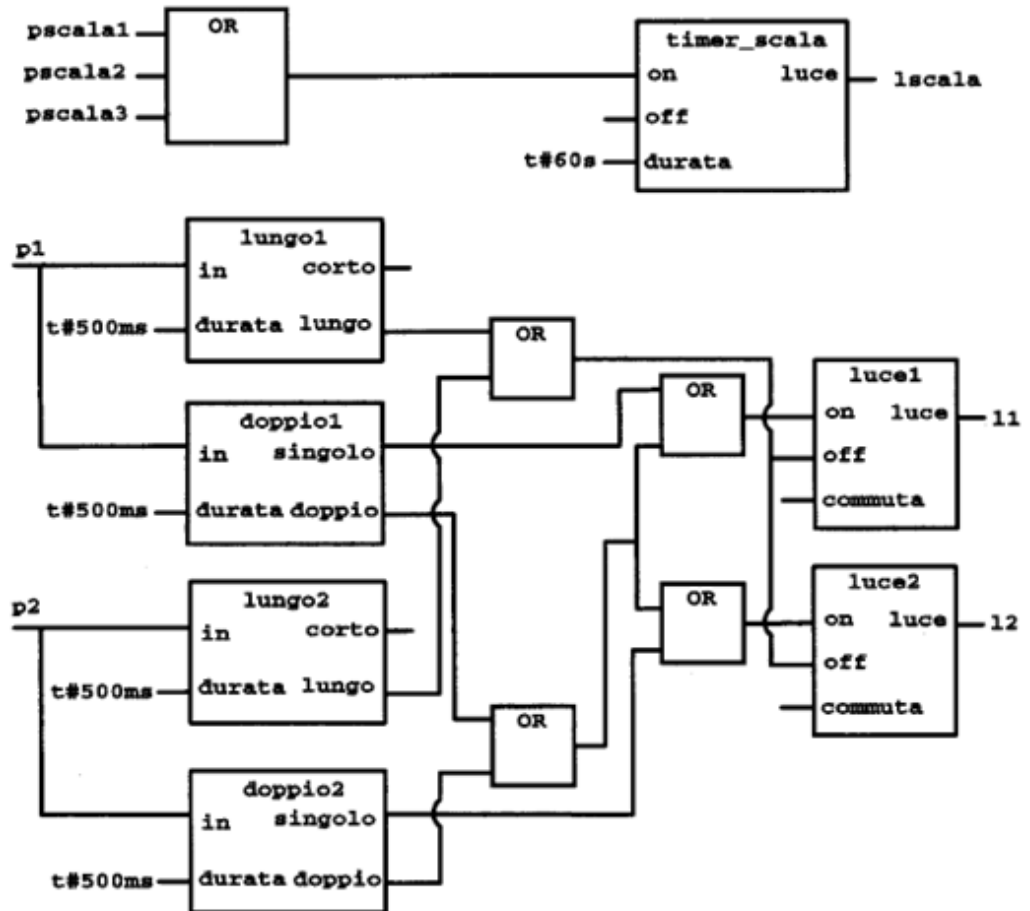
- Programma di gestione di un generico capannone (gestione_luci)

PROGRAM gestione_luci

VAR_INPUT p1,p2,pscala1,pscala2,pscala3: BOOL; END_VAR

VAR_OUTPUT l1,l2,lscala: BOOL; END_VAR

VAR timer_scala: timer_luce; luce1,luce2: gestione_luce;
lungo1,lungo2: lunghezza_pressione;
doppio1,doppio2: doppia_pressione; END_VAR



END_PROGRAM



esempi di programmazione

Introduzione

– Risorse e configurazione

- » Si suppone di disporre di una sola risorsa di elaborazione con la sigla PLC586:

```
CONFIGURATION controllore_impianti_luce
  RESOURCE controllore ON PLC586
    PROGRAM capannone1 : gestione_luci(
      p1:=%I3.4, p1:=%I3.5, pscala1:=%I3.6,
      pscala2:=%I3.7, pscala3:=%I3.8,
      lscala:=%Q1.1, l1:=%Q1.2, l2:=%Q1.3 );
    PROGRAM capannone2 : gestione_luci(
      p1:=%I2.4, p1:=%I2.5, pscala1:=%I2.6,
      pscala2:=%I2.7, pscala3:=%I2.8,
      lscala:=%Q4.1, l1:=%Q4.2, l2:=%Q4.3 );
  END_RESOURCE
END_CONFIGURATION
```

- » *Nota:* due istanze dello stesso programma perché i capannoni sono identici, ma collegati a differenti ingressi e uscite fisiche

esempi di programmazione

Introduzione



- » Si suppone di disporre di due risorse di elaborazione con la sigla PLC486 e PLC586:

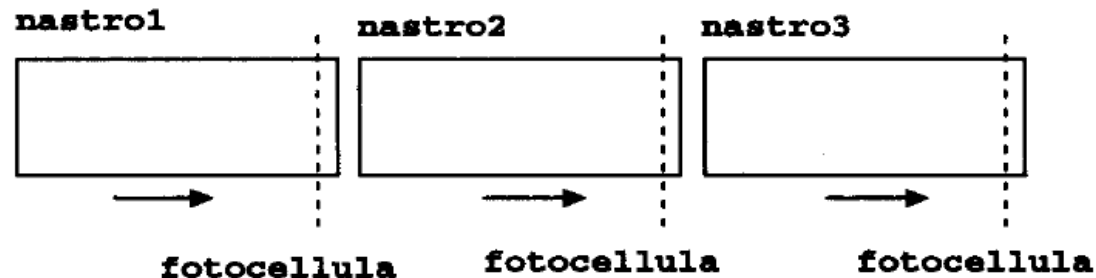
```
CONFIGURATION controllore_impianti_luce
  RESOURCE primo_capannone ON PLC586
    PROGRAM capannone1 : gestione_luci(
      p1:=%I3.4, p1:=%I3.5, pscala1:=%I3.6,
      pscala2:=%I3.7, pscala3:=%I3.8,
      lscala:=%Q1.1, l1:=%Q1.2, l2:=%Q1.3 );
  END_RESOURCE
  RESOURCE secondo_capannone ON PLC486
    TASK periodo (INTERVAL:=T#10ms);
    PROGRAM capannone2 : gestione_luci(
      p1:=%I2.4, p1:=%I2.5, pscala1:=%I2.6,
      pscala2:=%I2.7, pscala3:=%I2.8,
      lscala:=%Q4.1, l1:=%Q4.2, l2:=%Q4.3 )
    WITH periodo;
  END_RESOURCE
END_CONFIGURATION
```

esempi di programmazione

Introduzione



– Sistema di nastri trasportatori



- Dei pacchi vengono posati sul primo nastro e devono essere trasportati alla fine del terzo nastro
 - Per ogni nastro, si dispone dei seguenti ingressi digitali:
 - photo, fotocellula posta alla fine del nastro
 - p_dopo, indica che il dispositivo successivo è pronto ad accettare un pacco
- e delle seguenti uscite digitali:
- ava, controlla il movimento del nastro

esempi di programmazione

Introduzione



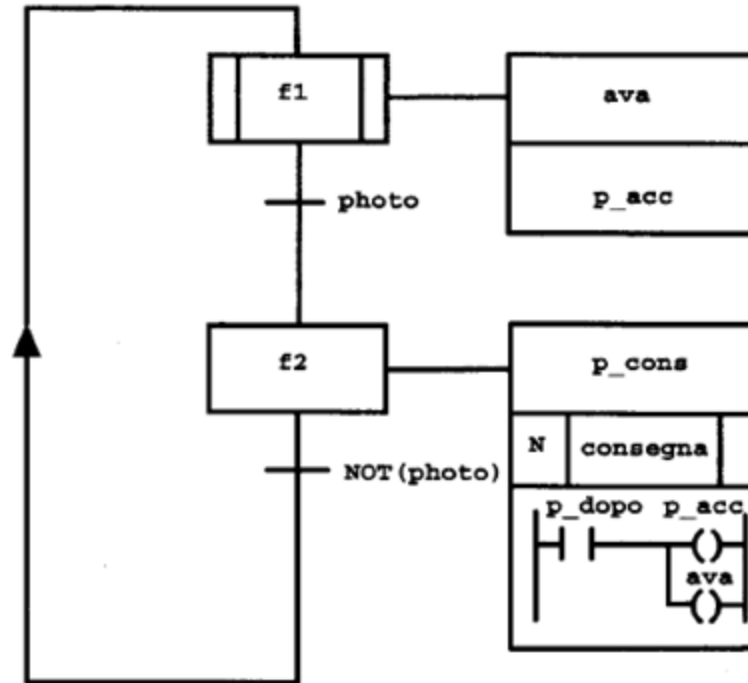
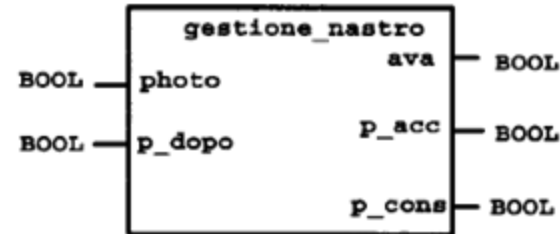
- p_acc, indica al dispositivo precedente che il nastro è pronto per accettare un altro pacco
- p_cons, indica al dispositivo successivo che il nastro è pronto a consegnare un pacco arrivato alla sua estremità
- Soluzione proposta:
 - Considerata la modularità del sistema, è conveniente definire un blocco funzionale che gestisca il singolo componente per poi connetterli al livello di programma
 - Gestione di un nastro trasportatore (gestione_nastro)
 - » Il nastro è normalmente in movimento e si ferma solo se un pacco arriva alla sua estremità ma il dispositivo successivo non è pronto ad accettarlo
 - » Ogni qualvolta il nastro è in movimento può accettare un pacco dal nastro che lo precede

esempi di programmazione

Introduzione



FUNCTION_BLOCK



END_FUNCTION_BLOCK

esempi di programmazione

Introduzione



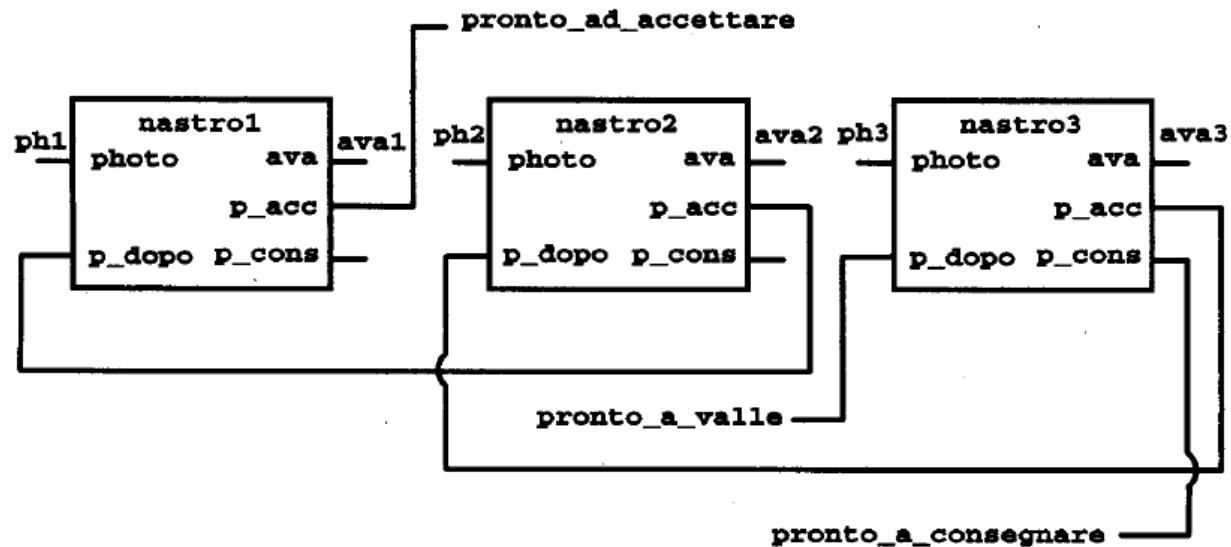
- Programma per la gestione dell'intero sistema:

```
PROGRAM tre_nastri
```

```
VAR_INPUT ph1,ph2,ph3,pronto_a_valle: BOOL; END_VAR
```

```
VAR_OUTPUT ava1,ava2,ava3, pronto_ad_accettare,  
           pronto_a_consegnare: BOOL; END_VAR
```

```
VAR nastro1,nastro2,nastro3: gestione_nastro; END_VAR
```



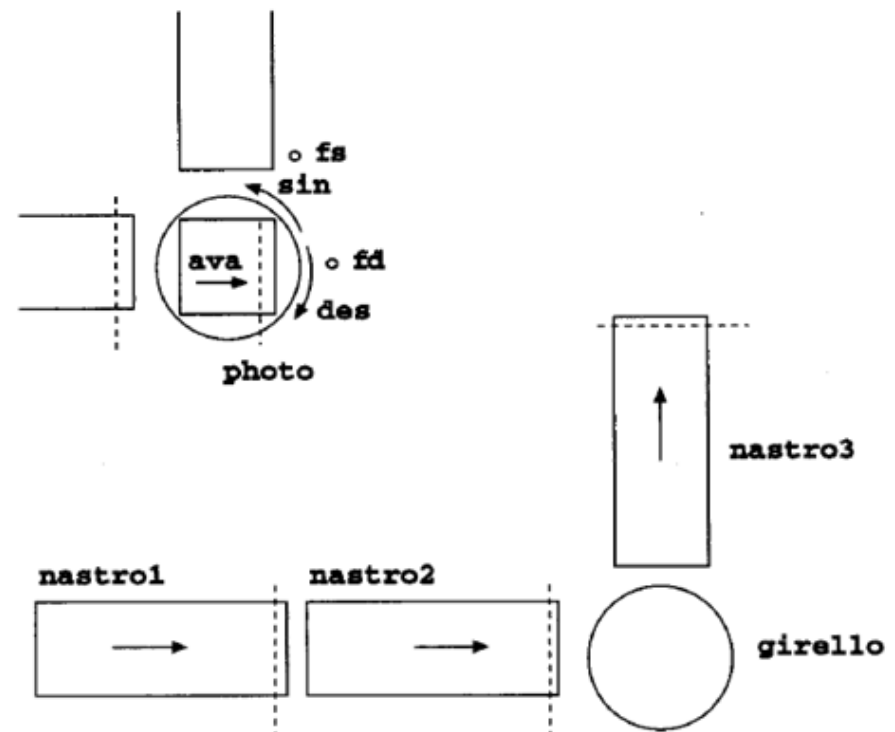
```
END_PROGRAM
```


esempi di programmazione

Introduzione

– Sistema di nastri trasportatori e girello automatico

- Il sistema è composta da tre nastri trasportatori e da un girello automatico descritto in figura
- Per la gestione dei nastri si utilizzerà il blocco già sviluppato in precedenza





esempi di programmazione

Introduzione

- Per il girello, si dispone dei seguenti ingressi digitali:
 - photo, fotocellula posta alla fine del girello che indica che il pacco è stato completamente trasferito
 - fs, finecorsa di rotazione antiorario del girello
 - fd, finecorsa di rotazione orario del girello
 - p_dopo, indica che il dispositivo a valle del girello è pronto ad accettare un pacco
 - p_prima, indica che il dispositivo a monte del girello è pronto a consegnare un pacco

e delle seguenti uscite digitali:

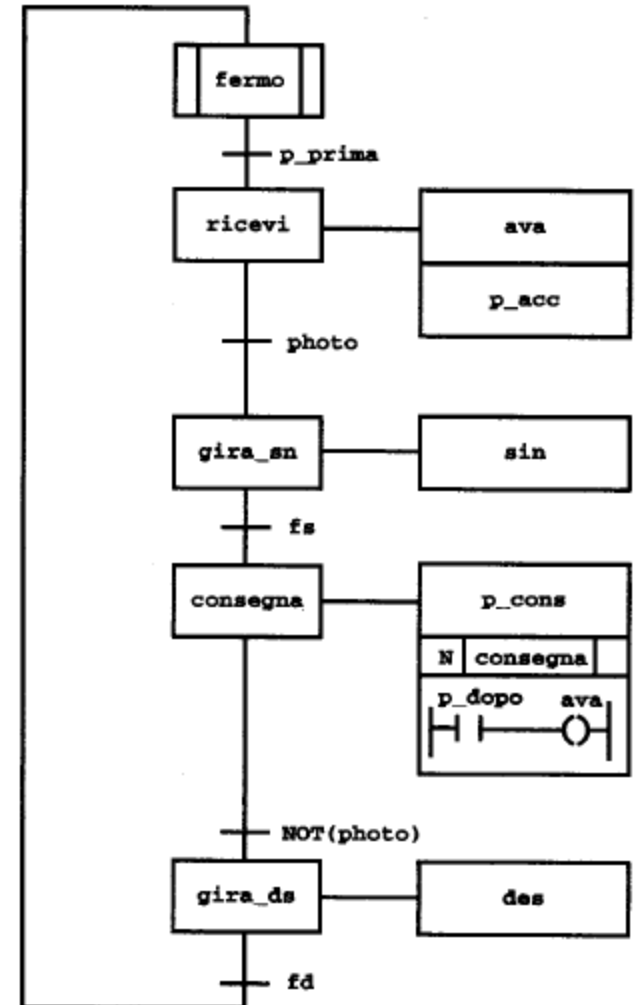
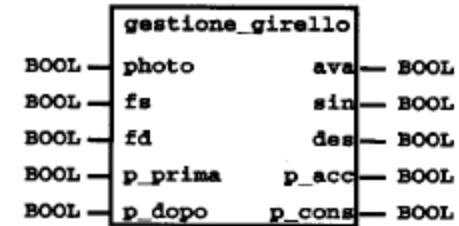
- ava, controlla il movimento dei rulli posti sul girello
- sin, controlla la rotazione del girello in senso antiorario
- des, controlla la rotazione del girello in senso orario

esempi di programmazione

Introduzione

- p_acc, indica che il girello pronto ad accettare un pacco dal dispositivo a monte
- p_cons, indica che il girello è pronto a consegnare un pacco al dispositivo a valle
- Gestione del girello (gestione_girello)
 - Il girello è normalmente in quiete
 - I rulli vengono messi in movimento quando il dispositivo a monte è pronto a consegnare un pacco
 - Dopo la consegna, il girello deve ruotare in senso antiorario di 90° e consegnare il pacco se possibile prima di ritornare nella posizione di attesa

FUNCTION_BLOCK



esempi di programmazione

Introduzione

- Programma di gestione del sistema completo (nastri_girello)

```
PROGRAM nastri_girello  
  
VAR_INPUT ph1,ph2,ph3,phg,fsg,fdg,pronto_a_valle: BOOL; END_VAR  
  
VAR_OUTPUT ava1,ava2,ava3,avag,sing,desg,pronto_ad_accettare,  
           pronto_a_consegnare: BOOL; END_VAR  
  
VAR nastri1,nastro2,nastro3: gestione_nastro;  
    girello: gestione_girello;  
END_VAR
```

