

# Introduzione all'architettura dei Calcolatori

---

# Introduzione

---

- Che cos'è un calcolatore?
- Come funziona un calcolatore?
  - è possibile rispondere a queste domande in molti modi, ciascuno relativo a un diverso punto di vista
  - in questo corso, il punto di vista prevalente è quello del calcolatore come *macchina programmabile*, ovvero in grado di eseguire programmi

# Architettura dei calcolatori

---

- Che cos'è un calcolatore? Come funziona un calcolatore?
  - un calcolatore è un sistema
  - un **sistema** è un oggetto costituito da molte parti (componenti) che interagiscono, cooperando, al fine di ottenere un certo comportamento
- Studiare l'**architettura** di un sistema vuol dire
  - individuare ciascun componente del sistema
  - comprendere i principi generali di funzionamento di ciascun componente
  - comprendere come i vari componenti interagiscono tra di loro

# Hardware e software

---

- La prima decomposizione di un calcolatore è relativa alle seguenti macro-componenti:

## hardware

- la struttura fisica del calcolatore, costituita da componenti elettronici ed elettromeccanici

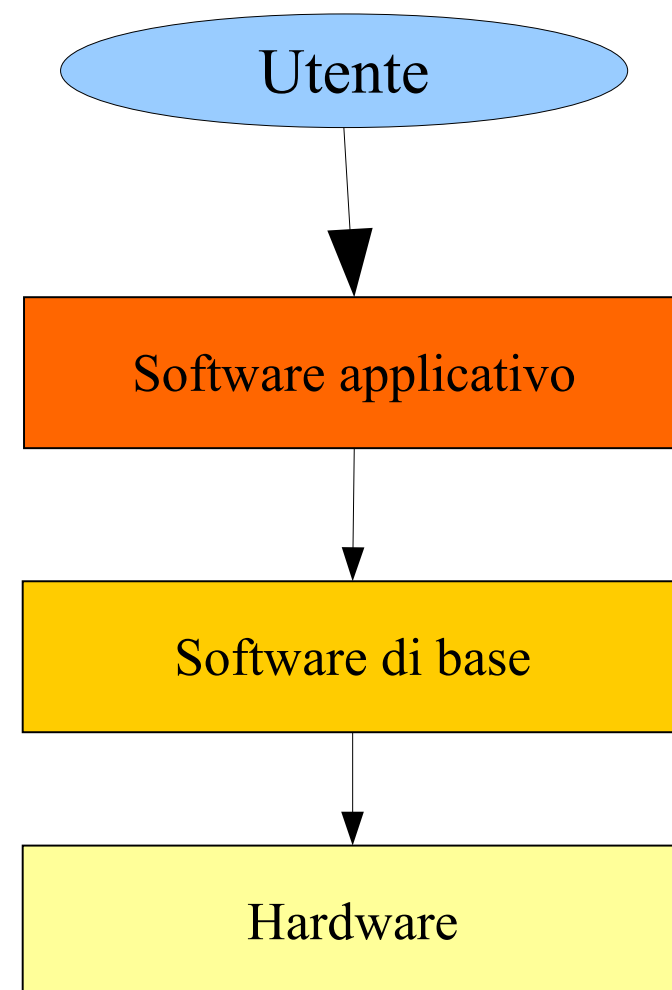
## software

- l'insieme dei programmi che consentono all'hardware di svolgere dei compiti utili
- il software comprende il **software di base** (tra cui il sistema operativo) e il **software applicativo**

# Organizzazione a livelli

---

- Hardware e software sono organizzati a **livelli** (o **strati**).
- Ogni livello usa i servizi offerti dal livello sottostante.
- Un utente agisce usando i servizi del livello software applicativo (web, videogames, programmi di grafica etc)
- Le applicazioni a loro volta usano i servizi del software di base (es. sistema operativo)
- Il livello software di base, utilizza il livello hardware.



# Macchina di Von Neumann

---

E' composta da 4 tipologie di componenti funzionali:

**o** unità centrale di elaborazione (CPU)

**n** esegue istruzioni per l'elaborazione dei dati

**n** svolge anche funzioni di controllo

**o** memoria centrale

**n** memorizza e fornisce l'accesso a dati e programmi

**o** interfacce di ingresso e uscita

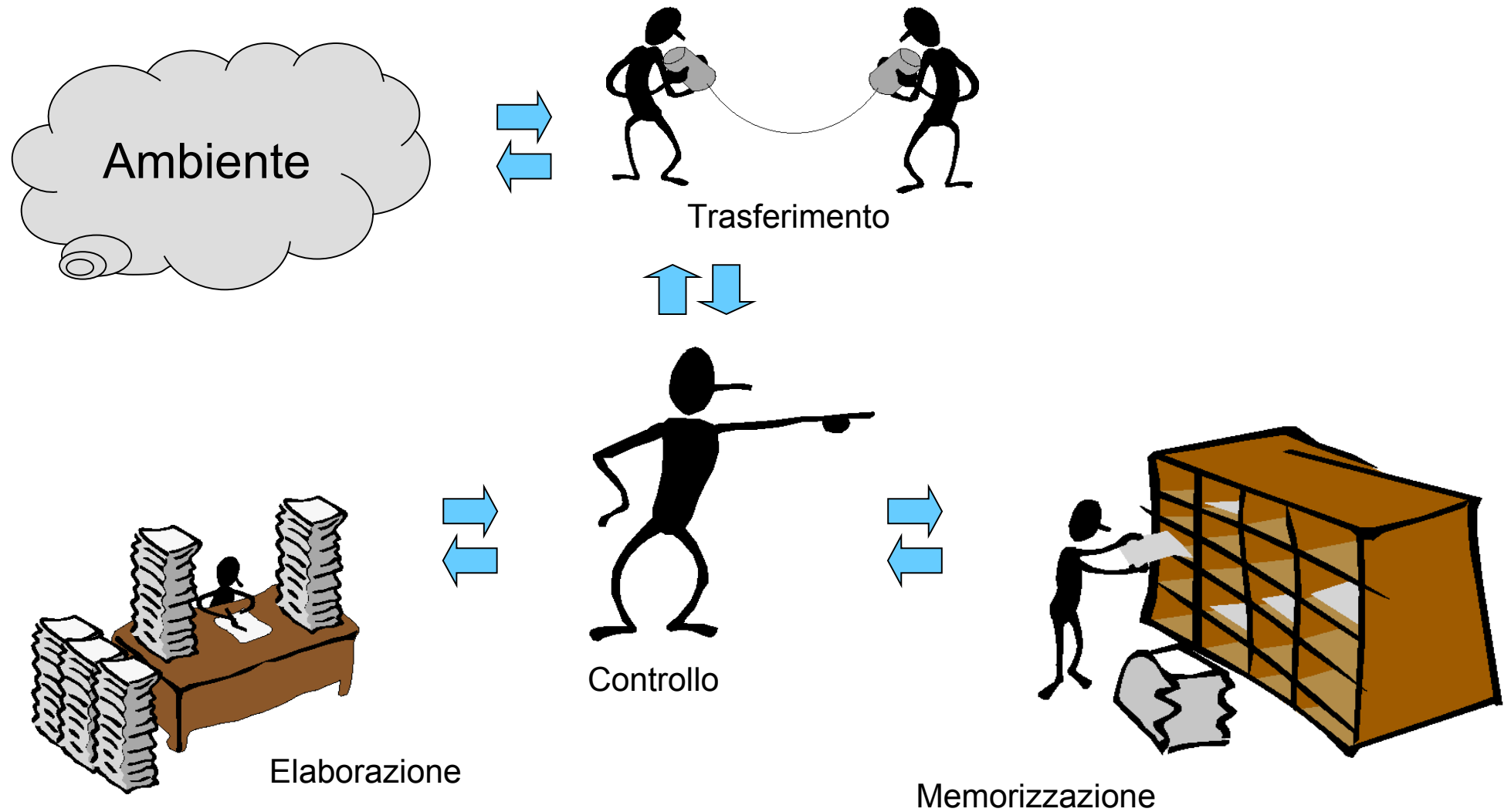
**n** componenti di collegamento con le periferiche del calcolatore

**o** bus

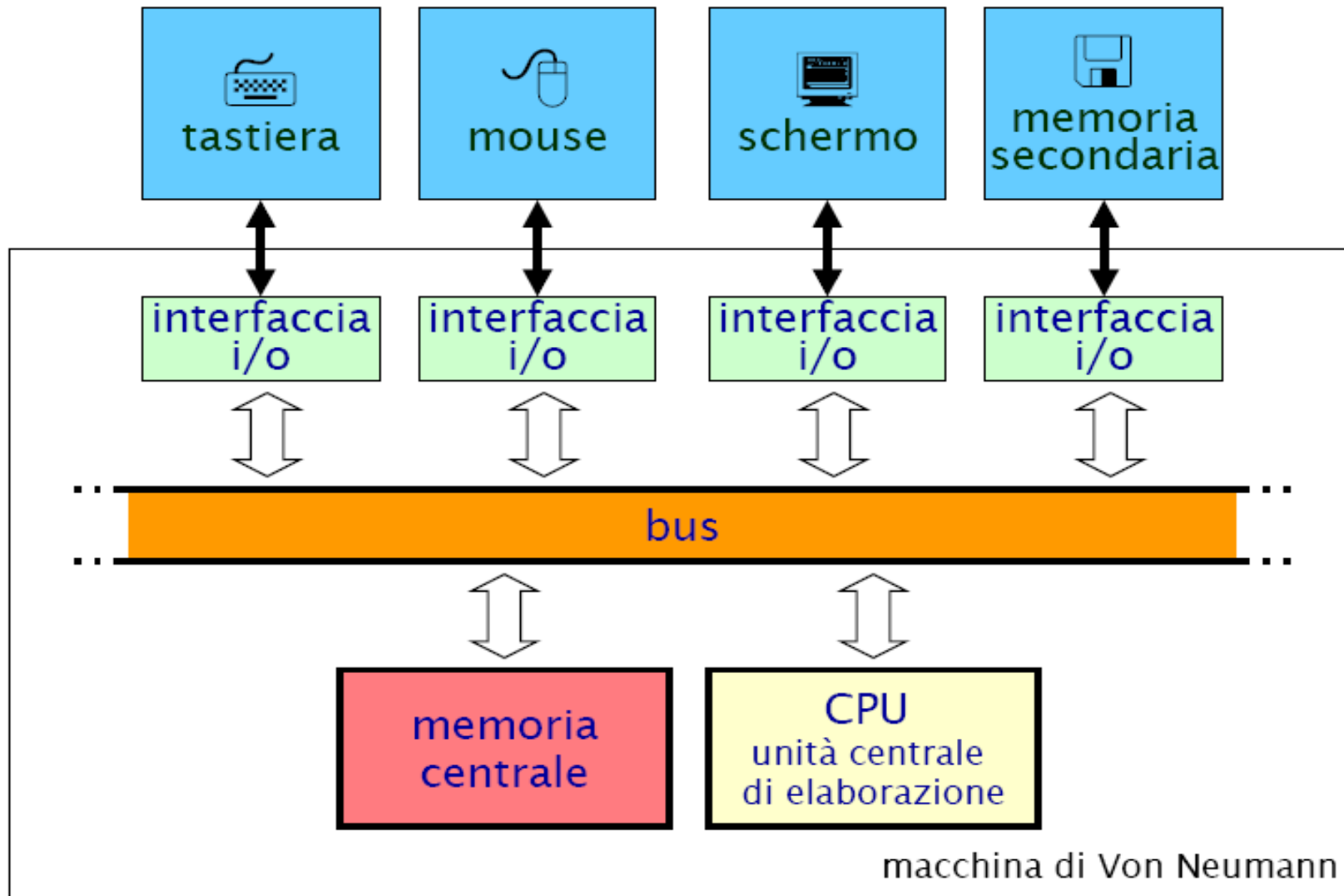
**n** svolge la funzionalità di trasferimento di dati e di informazioni di controllo tra le varie componenti funzionali

# Vista funzionale di un Calcolatore

---



# Componenti di un Calcolatore



# Tipi di Bus

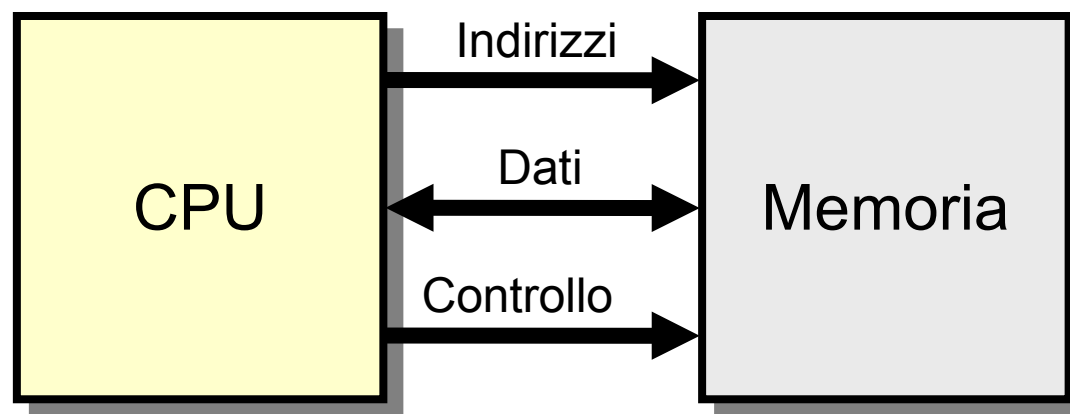
---

- **Bus dati:** utilizzato per trasferire dati (es. fra memoria e CPU, fra CPU e interfacce di I/O)
- **Bus indirizzi:** che identifica la posizione delle celle di memoria un cui la CPU va a scrivere o leggere
- **Bus di controllo:** in cui transitano i segnali di controllo che consentono di selezionare le unità coinvolte in un trasferimento dati (sorgente e destinazione), di definire la direzione dello scambio (scrittura o lettura)

# Architettura di Von Neumann

---

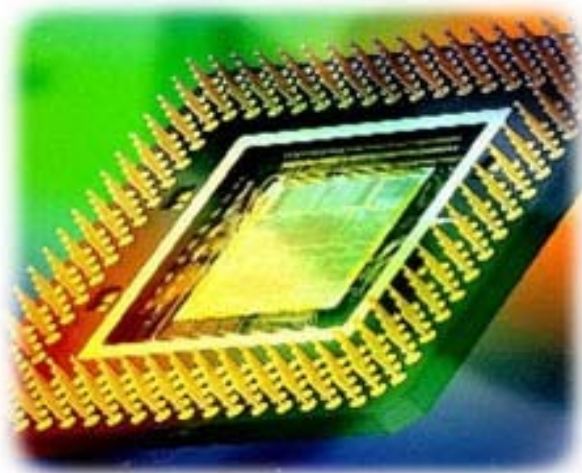
- Burks, Goldstein e Von Neumann sono stati i primi a proporre che il codice del programma potesse essere memorizzato nella stessa memoria dei dati



- Memoria indifferenziata per dati o istruzioni
- Solo l'interpretazione da parte di CPU stabilisce se una data configurazione di bit è da riguardarsi come un dato o come un'istruzione

---

# L'Unità Centrale di Elaborazione (CPU)



# Elementi di una CPU

---

## ■ Unità di controllo

→ Legge le istruzioni dalla memoria e ne determina il tipo

## ■ Unità aritmetico-logica

→ Esegue le operazioni necessarie per eseguire le istruzioni

## ■ Registri

→ Memoria ad alta velocità usata per risultati temporanei

→ Determina il parallelismo della CPU

→ Esistono registri generici e registri specifici

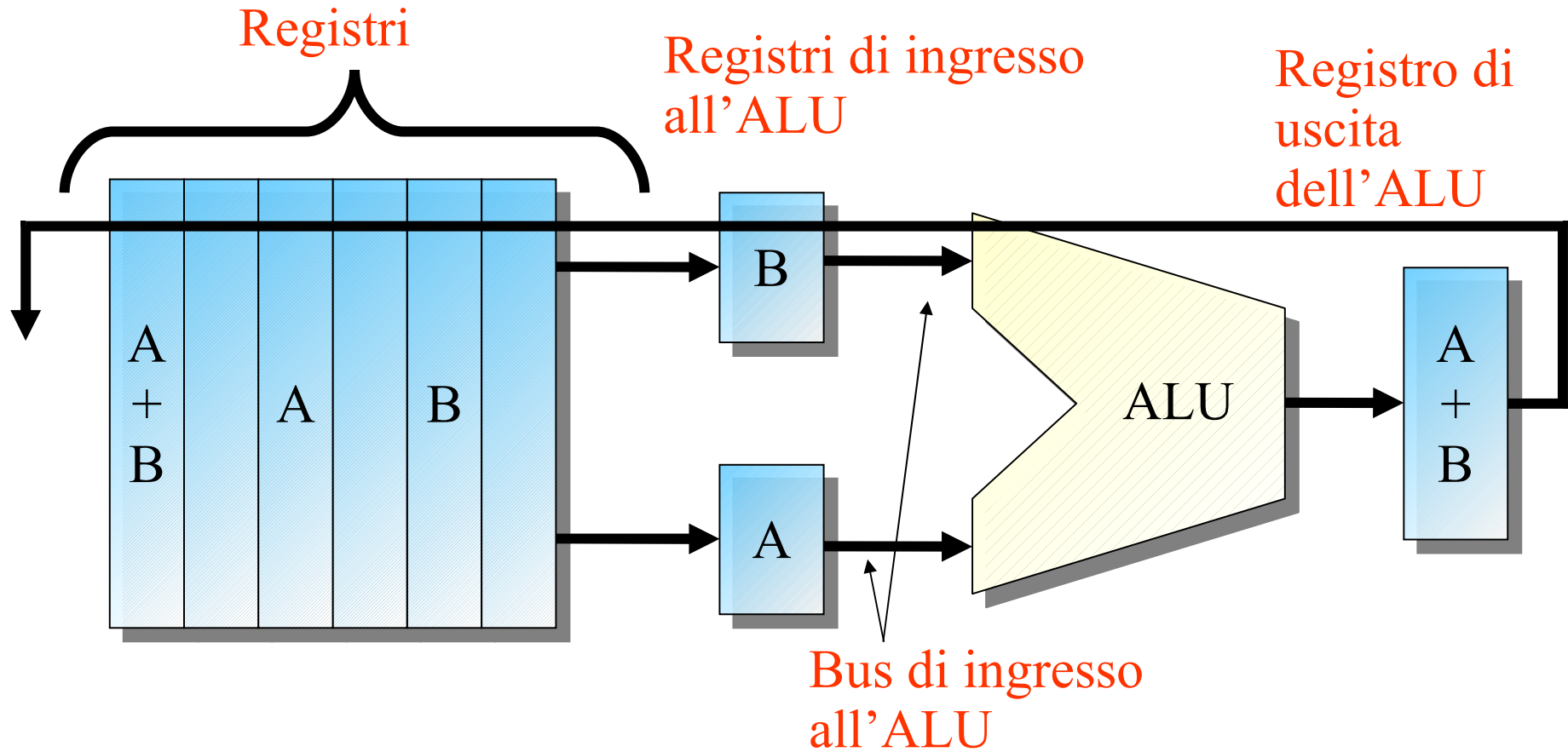
- ✓ Program Counter (PC)
- ✓ Instruction Register (IR)
- ✓ ...

# Tre Tipologie di Istruzioni

---

- Istruzioni Aritmetico Logiche (Elaborazione dati)
  - Somma, sottrazione, divisione, ...
  - And, Or, Xor, ...
  - Maggiore, minore, uguale, maggiore uguale, ...
- Controllo del flusso delle istruzioni
  - Sequenza
  - Selezione
  - Ciclo a condizione iniziale, a condizione finale, ...
- Trasferimento di informazione
  - Trasferimento dati e istruzioni tra CPU e memoria
  - Trasferimento dati e istruzioni tra CPU e dispositivi di I/O

# Struttura del "data path"



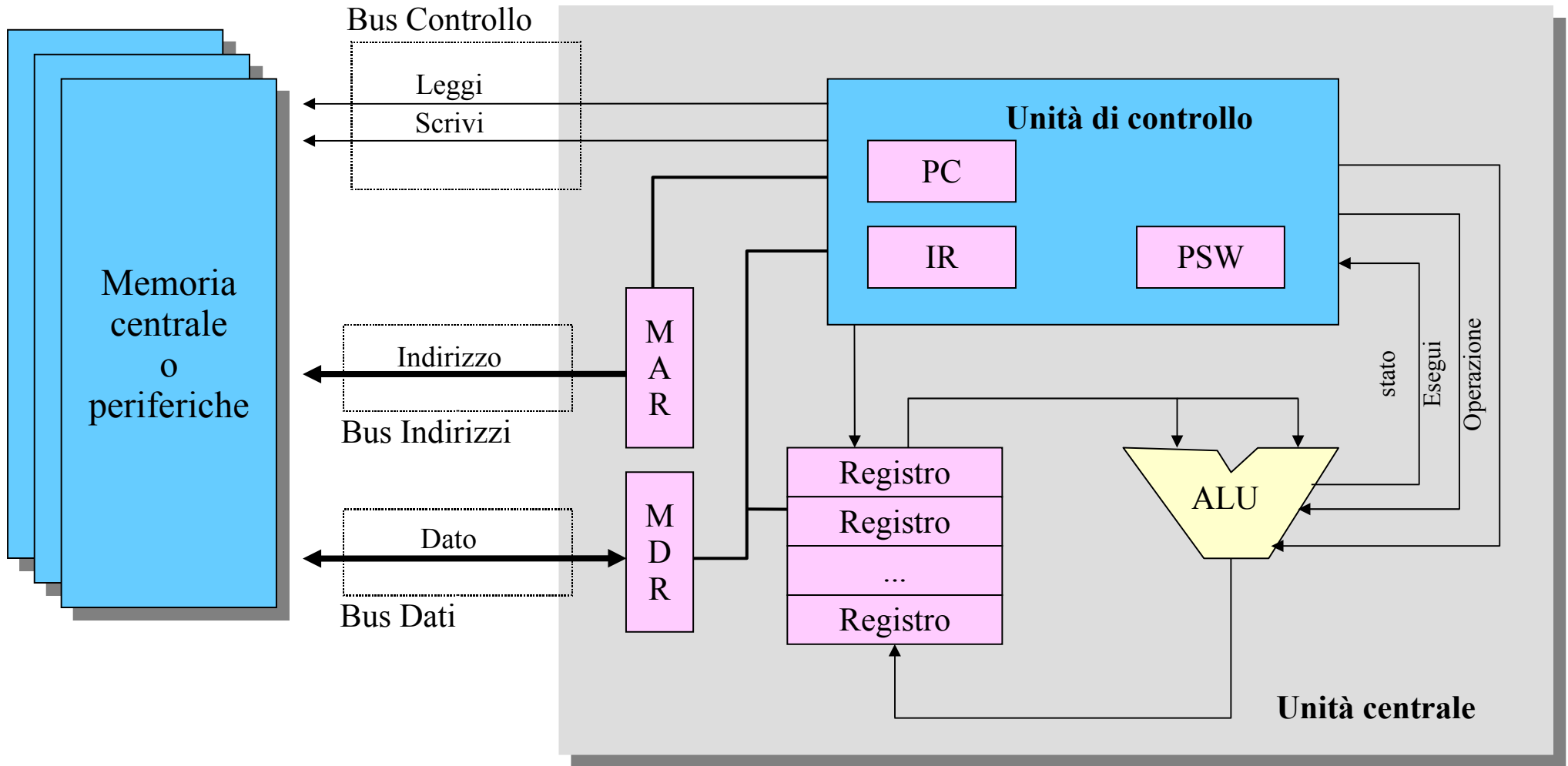
# Esecuzione delle Istruzioni

---

## ■ Ciclo **Fetch-Decode-Execute**

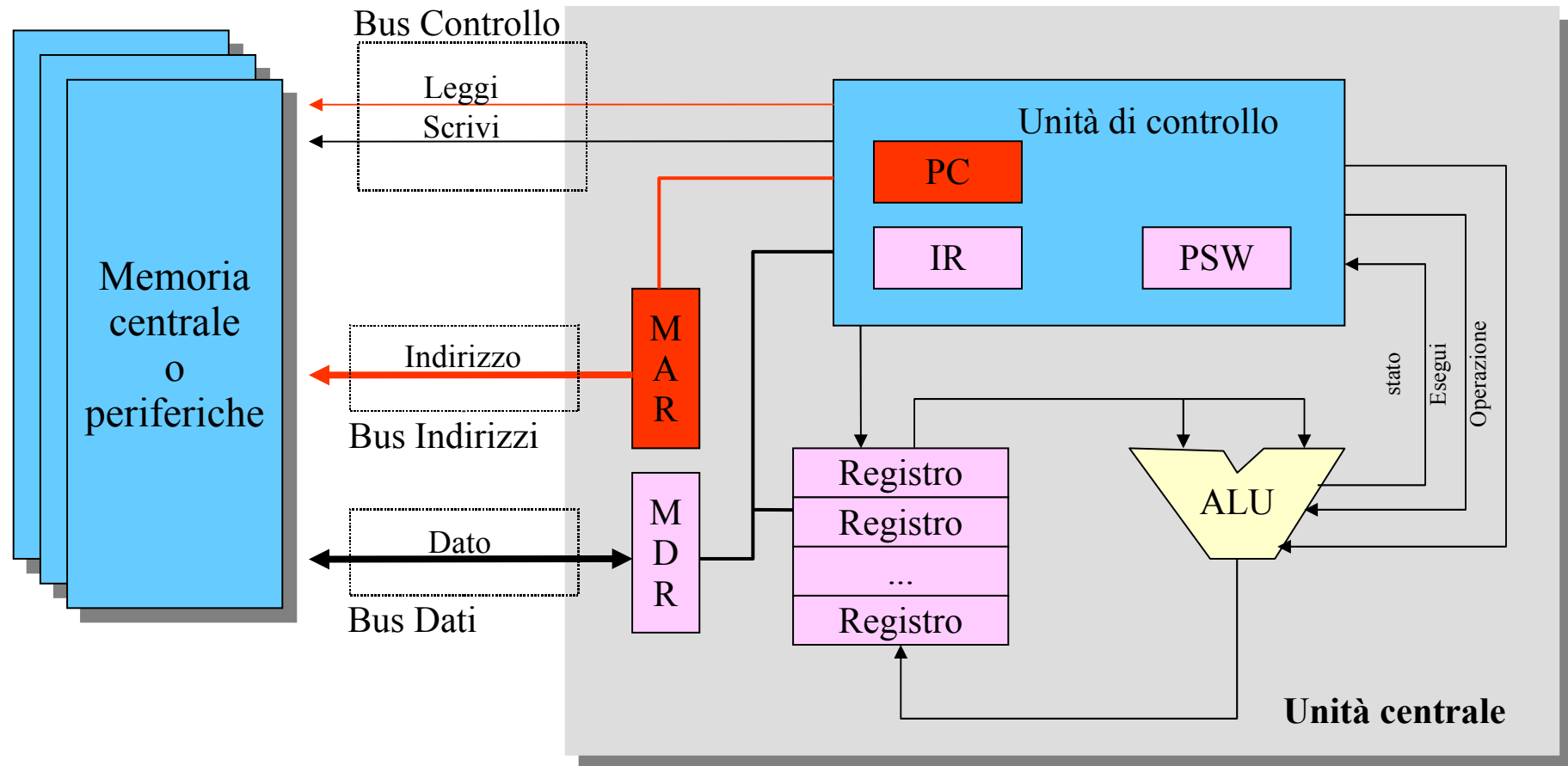
- Prendi l'istruzione corrente dalla memoria e mettila nel registro istruzioni (**IR**) [**Fetch**]
- Incrementa il program counter (**PC**) in modo che contenga l'indirizzo dell'istruzione successiva
- Determina il tipo dell'istruzione corrente [**Decodifica**]
- Se l'istruzione usa una parola in memoria determina dove si trova
- Carica la parola, se necessario, in un registro della CPU
- Esegui l'istruzione [**Execute**]
- Torna al punto **1.**

# Struttura Semplificata di una CPU



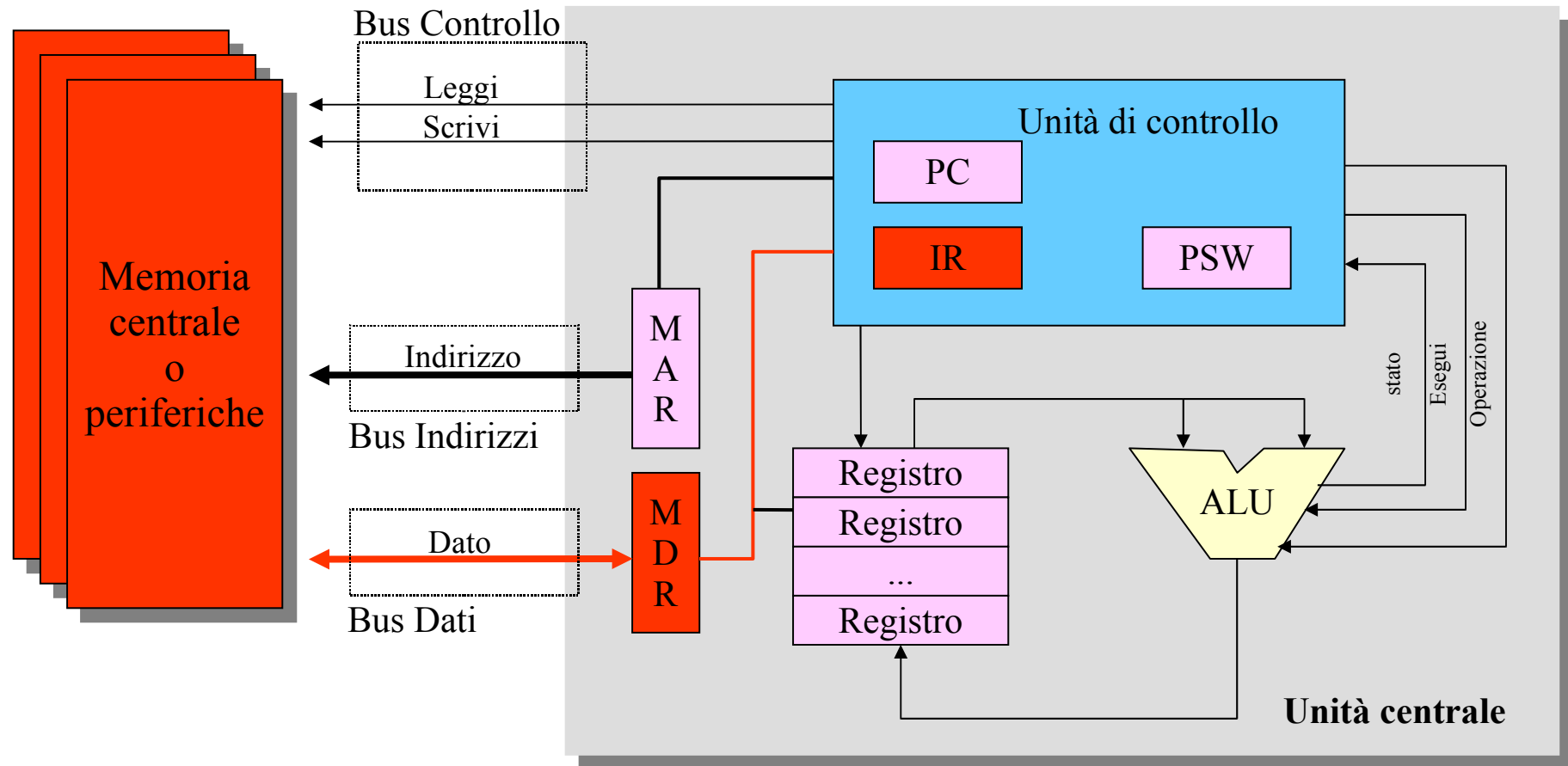
# Esempio: Somma di due registri

## ■ Fase di Fetch (1 di 2)



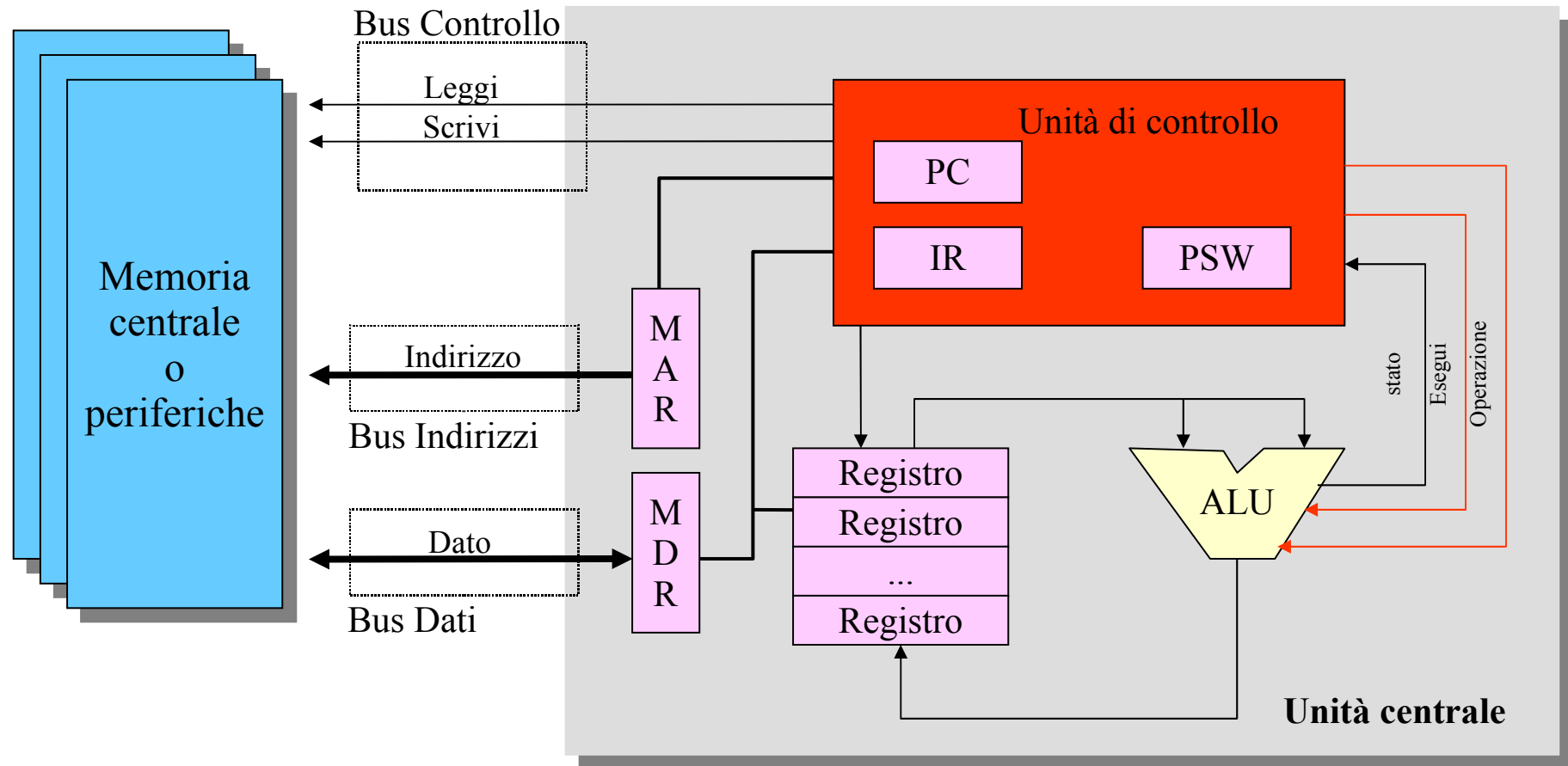
# Esempio: Somma di due registri

## ■ Fase di Fetch (2 di 2)



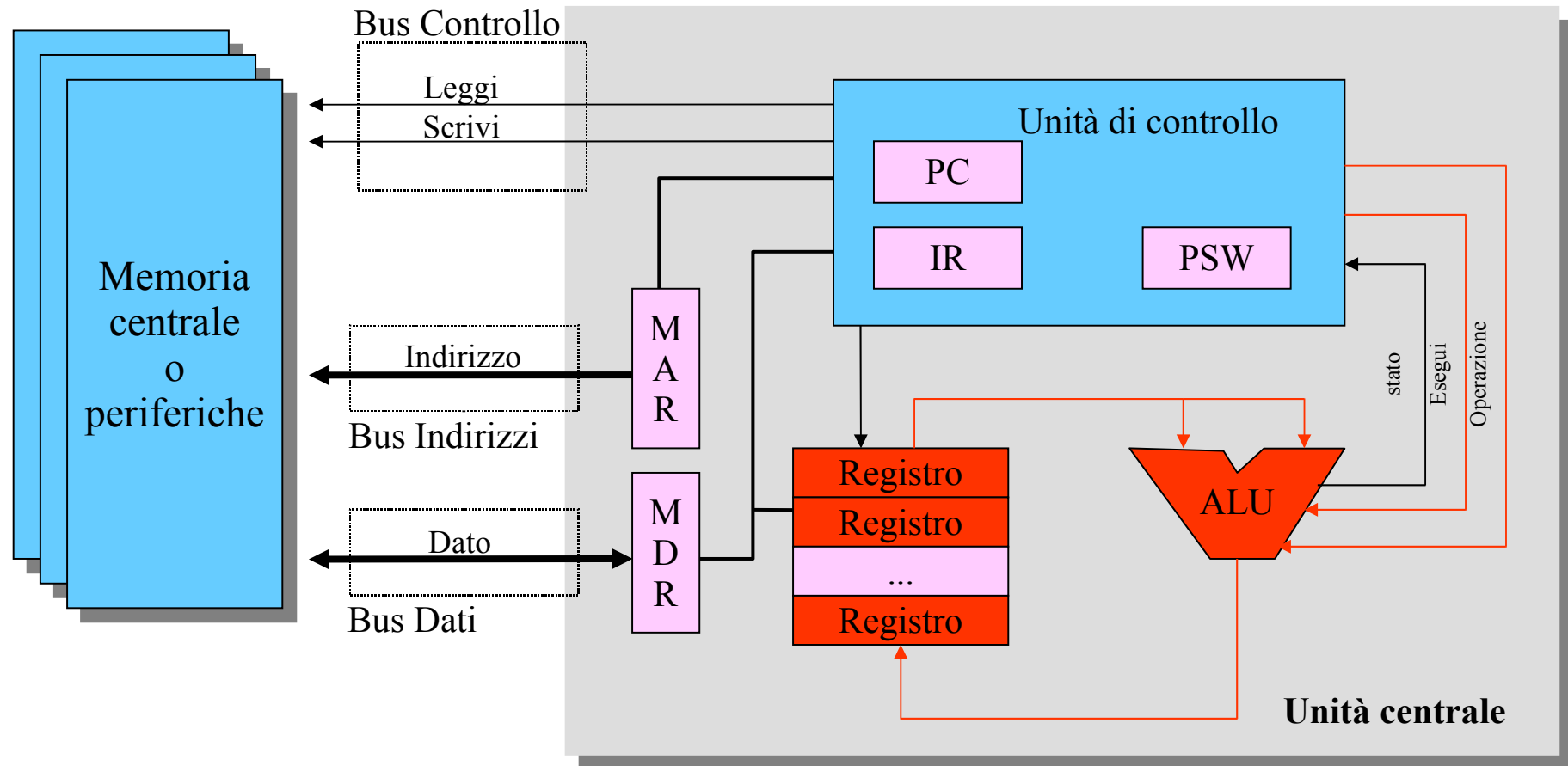
# Esempio: Somma tra due registri

## ■ Decodifica



# Esempio: Somma tra due registri

## ■ Esecuzione



# Frequenza di clock

---

La frequenza con cui si eseguono i cicli di esecuzione è scandita dal clock (orologio interno)

- n** ad ogni impulso di clock l'unità di controllo esegue un ciclo di esecuzione
  - n** la velocità di elaborazione di un microprocessore dipende dalla frequenza del suo clock (300, 400, ... MHz) (es.: 300 milioni di cicli al secondo). Attualmente si parla di GHz.
-

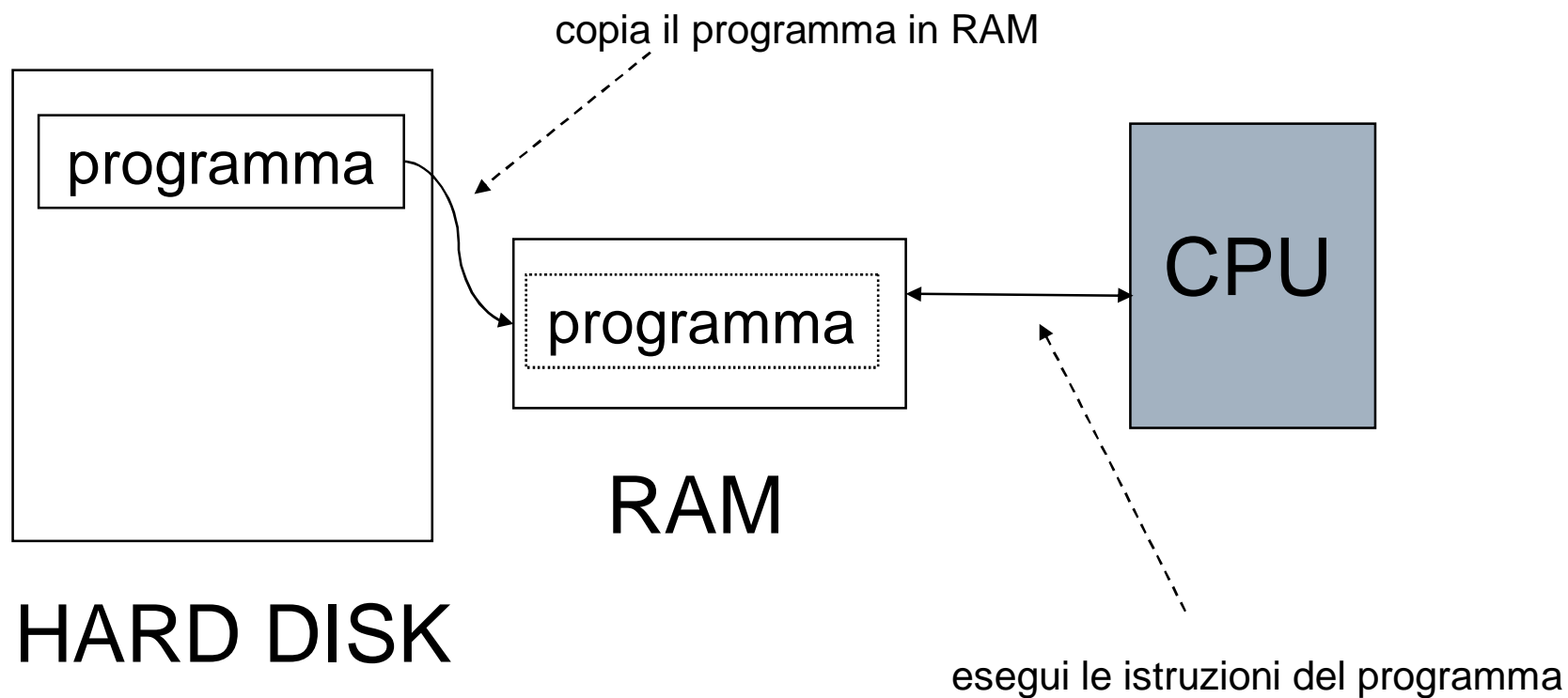
# Memoria centrale

---



# Funzionamento

---

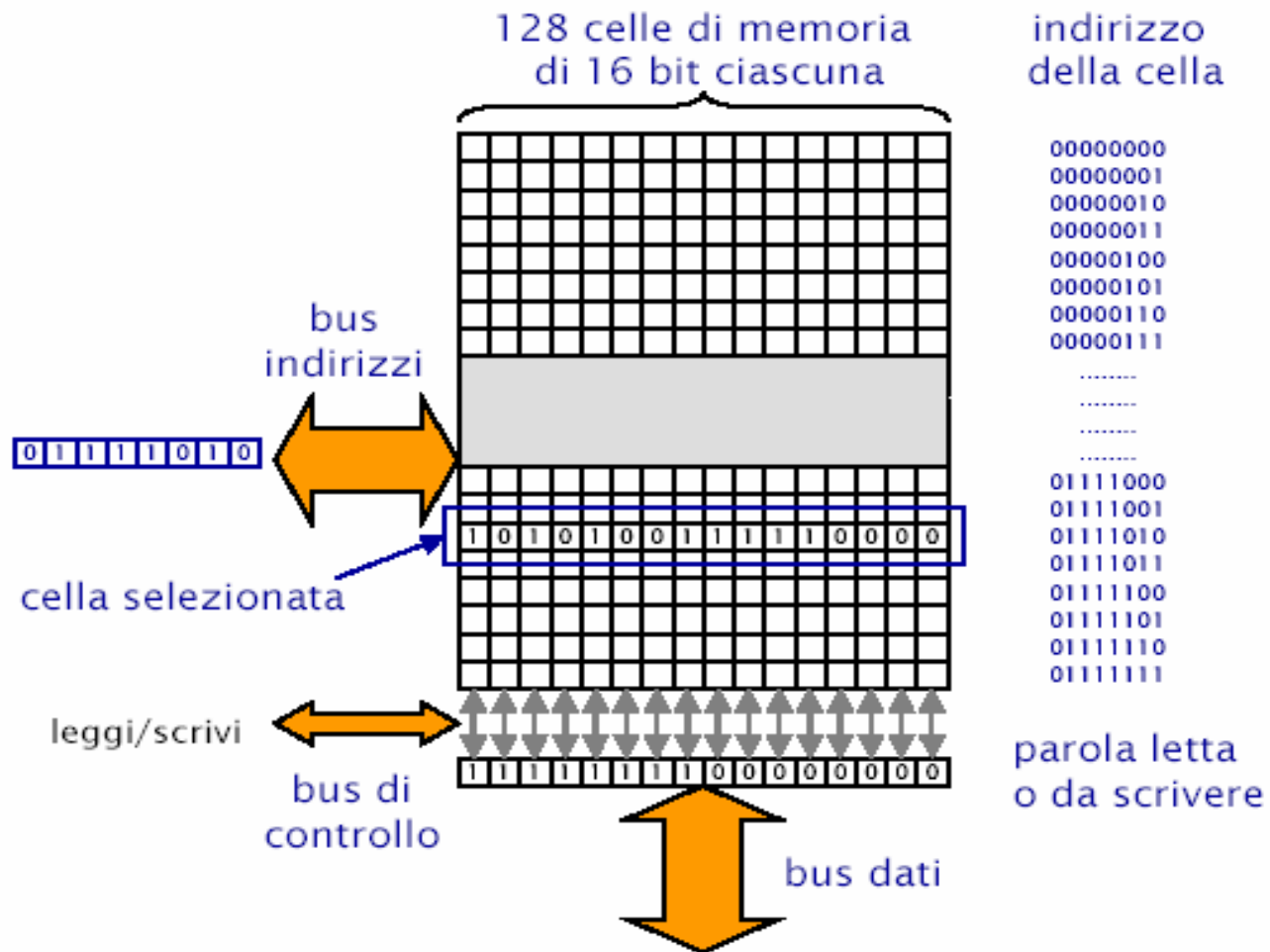


# Memorie RAM e memorie ROM

---

- Le memorie **RAM** (random access memory)
    - n possono essere accedute sia in lettura che in scrittura
    - n sono volatili (i dati memorizzati vengono persi allo spegnimento del calcolatore)
    - n sono usate per memorizzare dati e programmi
-

# Struttura della RAM



# La RAM

---

- Le celle sono numerate in sequenza: il numero di ogni cella costituisce il suo **indirizzo**
  - Specificando l'indirizzo di una cella, la CPU e' in grado di leggere e/o modificare il valore del byte memorizzato in quella cella
  - Random Access Memory (**RAM**), perche' ogni cella e' indirizzabile direttamente
-

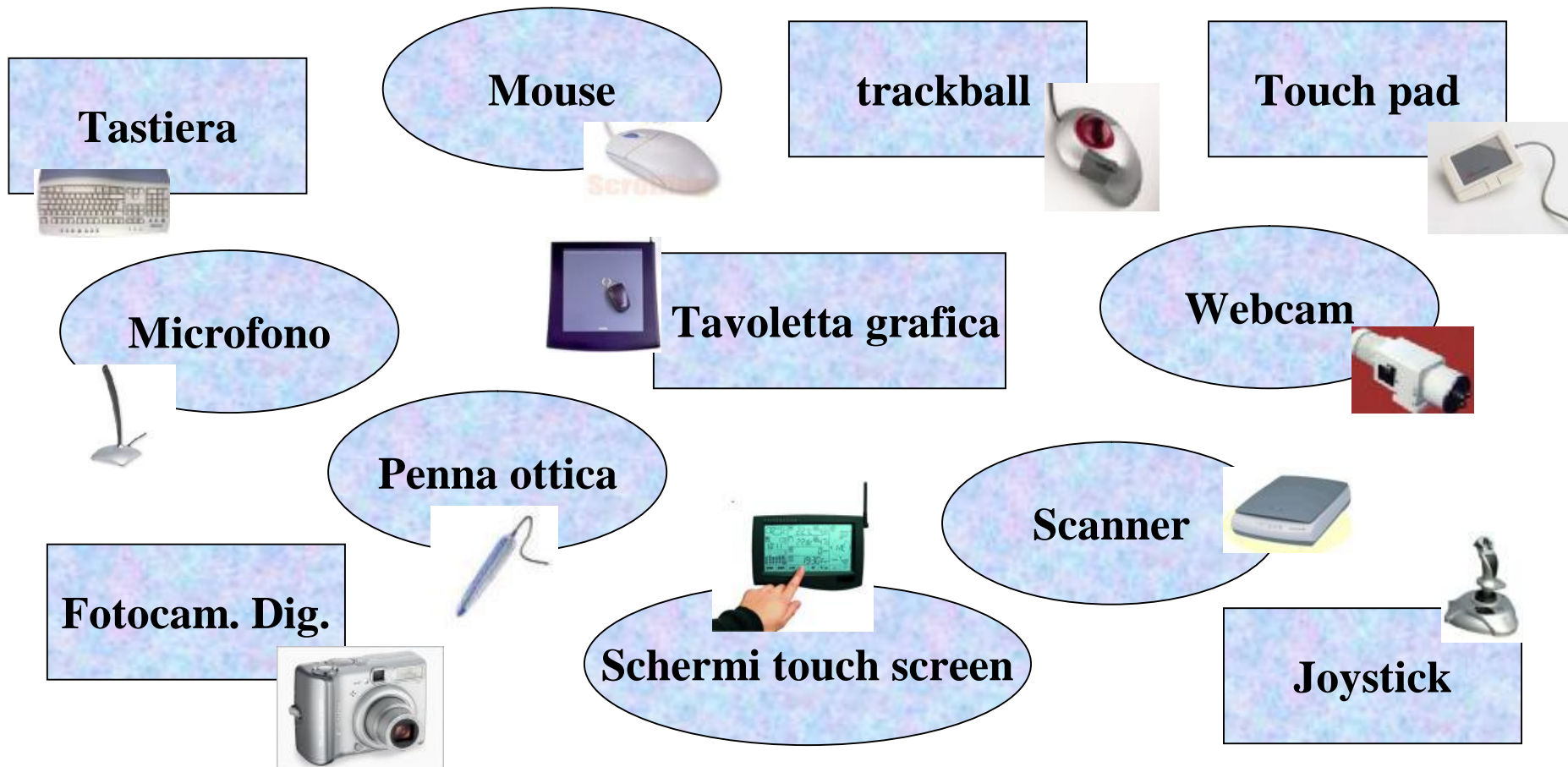
# Dispositivi di input/output (I/O, PERIFERICHE)

---

- **Terminali.** Tastiera + Video:  
n risoluzione, dimensione in pollici,...
  - **Stampanti:**  
n ad aghi, a getto, d'inchiostro, laser,...
  - **Modem:** per collegarsi in rete
  - **Scanner:** per digitalizzare le immagini
-

# Dispositivi di Input

---



# Dispositivi di Output

---

**Monitor**



**Plotter**



**Stampante**



**Casse acustiche**



**Cuffie**



**Videoproiettore**



# Piastra madre

