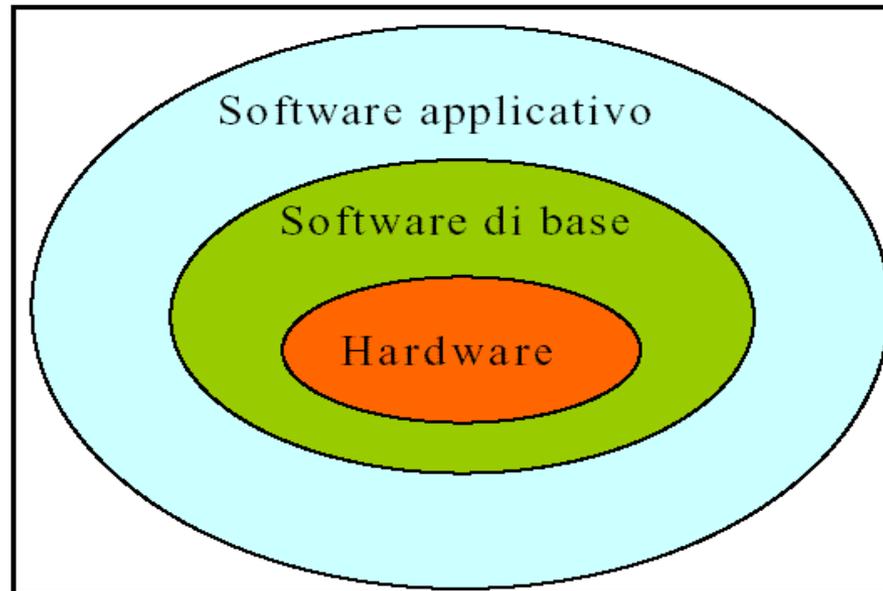


# Architettura del calcolatore

---

- La prima decomposizione di un calcolatore è relativa a due macro-componenti:
  - Hardware
  - Software



# Architettura del calcolatore

---

- L'architettura dell'**hardware** di un calcolatore reale è molto complessa
  - La **macchina di von Neumann** è un modello semplificato dei calcolatori moderni
    - **John von Neumann**, matematico ungherese, progettò, verso il 1945, il primo calcolatore con programmi memorizzabili anziché codificati mediante cavi e interruttori
-

# Macchina di Von Neumann

---

E' composta da 4 tipologie di componenti funzionali:

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

- esegue istruzioni per l'elaborazione dei dati
- svolge anche funzioni di controllo

□ **memoria centrale**

- memorizza e fornisce l'accesso a dati e programmi

□ **interfacce di ingresso e uscita**

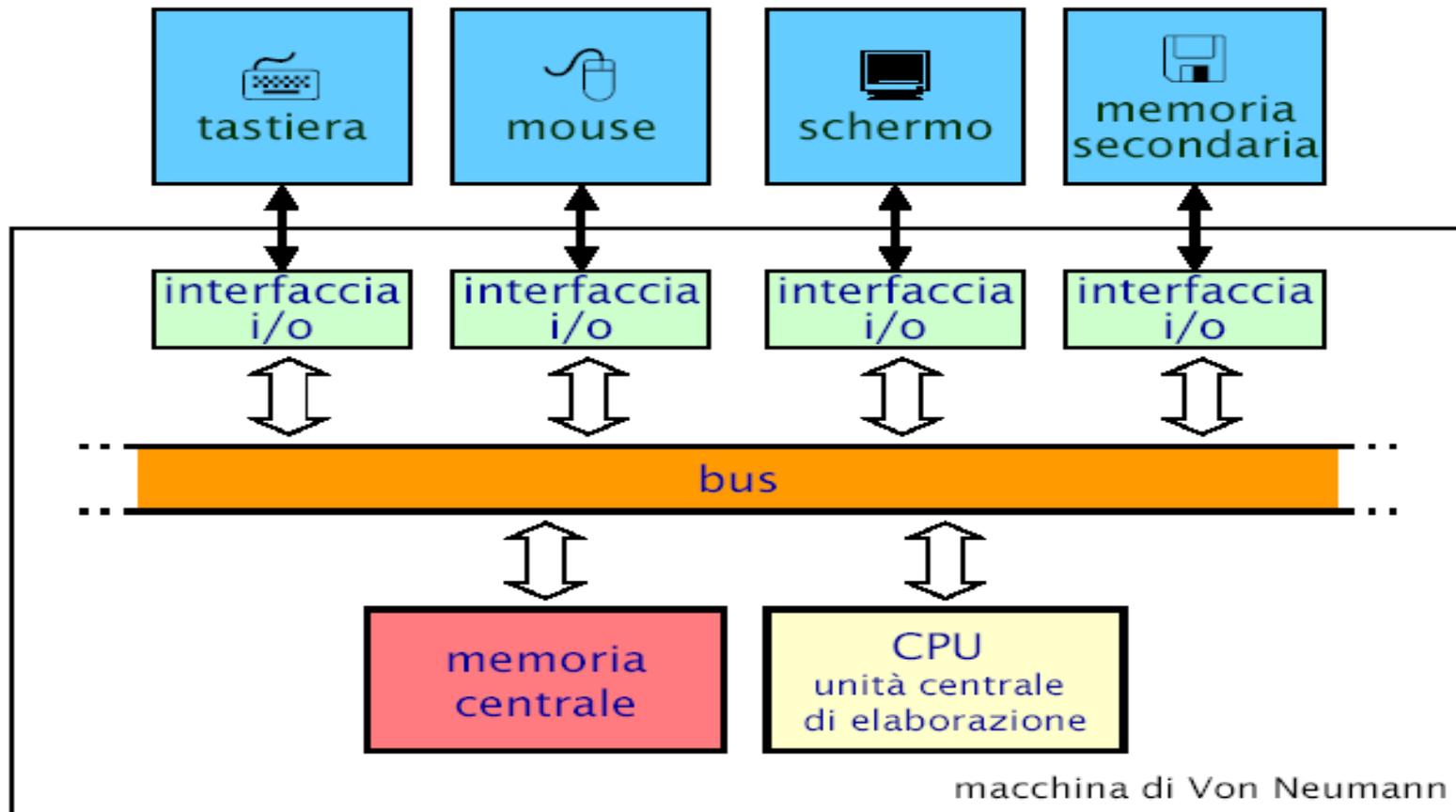
- componenti di collegamento con le periferiche del calcolatore

□ **bus**

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

# Macchina di Von Neumann

---



# Macchina di Von Neumann

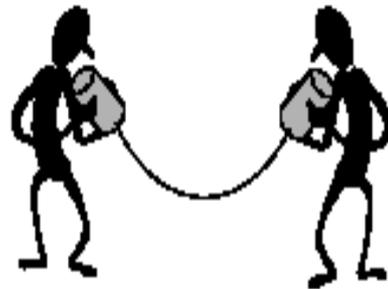
---

Il funzionamento di un calcolatore è descrivibile in termini di poche componenti (macro-unità) funzionali

- ogni macro-unità è specializzata nello svolgimento di una tipologia omogenea di funzionalità
  - *Eccezione*: l'unità centrale di elaborazione, che svolge sia funzionalità di elaborazione che di controllo
-

# Macchina di Von Neumann

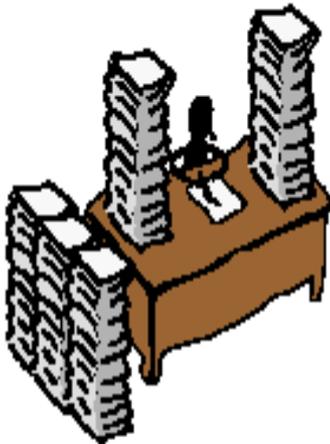
---



trasferimento



scambio di dati con l'utente



elaborazione



controllo



memorizzazione

---

# Memorizzazione

---

- Un calcolatore memorizza
    - i dati, che rappresentano informazioni di interesse
    - i programmi per l'elaborazione dei dati
  - La **memoria** è l'unità responsabile della memorizzazione dei dati
  - Una unità di memoria fornisce due sole operazioni
    - memorizzazione di un valore (**scrittura**)
    - accesso al valore memorizzato (**lettura**)
-

# Elaborazione

---

- Le istruzioni di un programma corrispondono ad operazioni elementari di elaborazione
    - operazioni aritmetiche
    - operazioni relazionali (confronto tra dati)
    - operazioni su caratteri e valori di verità
    - altre operazioni numeriche
-

# Elaborazione

---

- Un calcolatore sa svolgere poche tipologie di operazioni elementari ma in modo **molto efficiente**
    - un calcolatore può eseguire decine o centinaia di milioni di istruzioni al secondo
  - L'elaborazione dei dati viene svolta dall'**unità aritmetico-logica (ALU)**, che è un componente dell'unità centrale di elaborazione
-

# Trasferimento

---

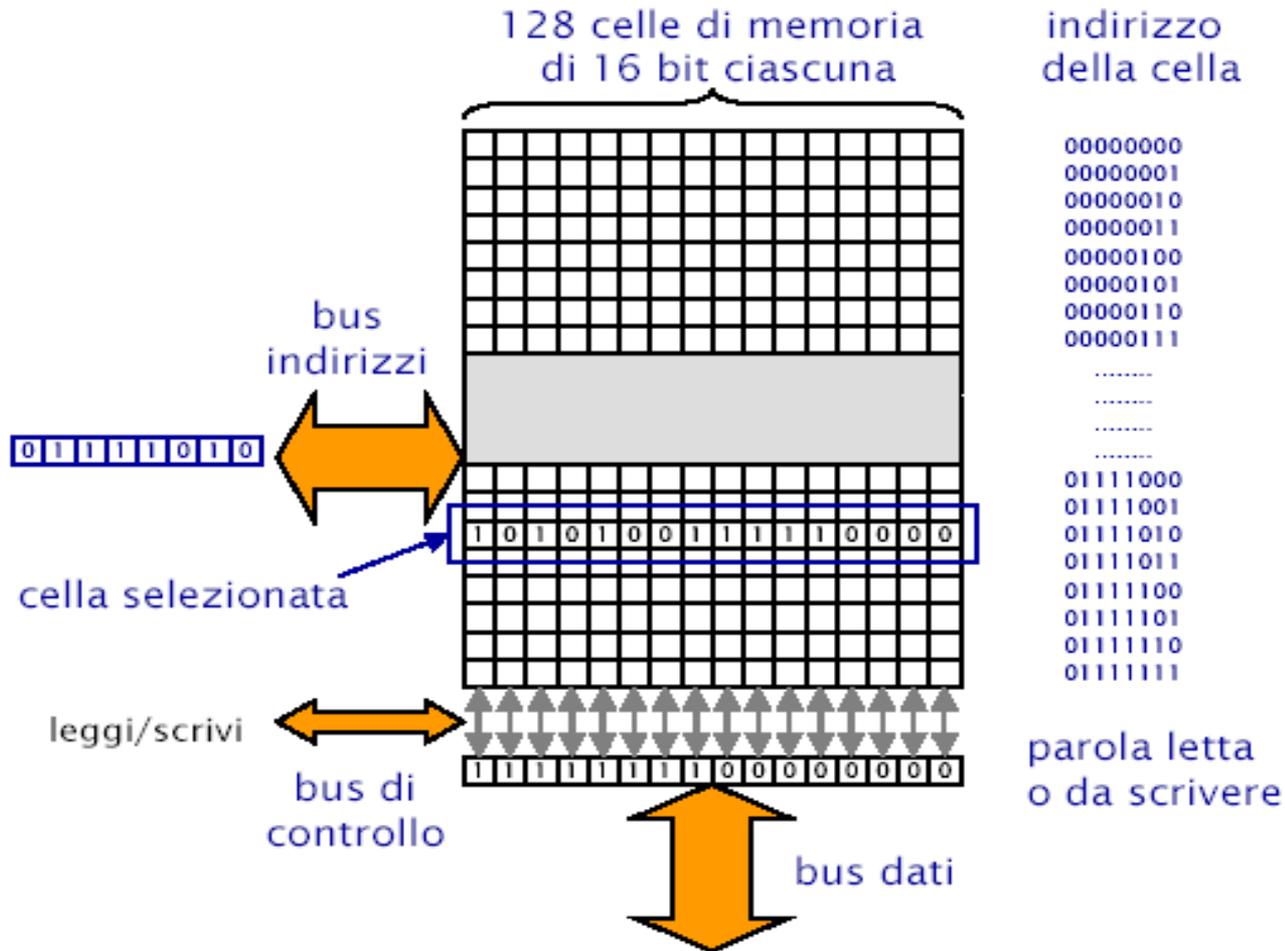
- Obiettivo: permettere lo scambio di informazioni tra le varie componenti funzionali del calcolatore
    - trasferimento dei dati e delle informazioni di controllo
  - Due possibili soluzioni
    - collegare ciascun componente con ogni altro componente
    - collegare tutti i componenti a un unico canale (**bus**)
  - L'utilizzo di un bus favorisce la modularità e l'espandibilità del calcolatore
-

# Controllo

---

- Il coordinamento tra le varie parti del calcolatore è svolto dall'**unità di controllo**
  - è un componente dell'unità centrale di elaborazione
  - ogni componente del calcolatore esegue solo le azioni che gli vengono richieste dall'unità di controllo
- il controllo consiste nel coordinamento dell'esecuzione temporale delle operazioni
  - sia internamente all'unità di elaborazione sia negli altri elementi funzionali

# 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
-

# La RAM

---

0	00101111
1	11001101
2	01010100
3	11111101
...	.....
65.536	10000110

---

---

# **Il processore - CPU** **(CENTRAL PROCESSING UNIT)**

---

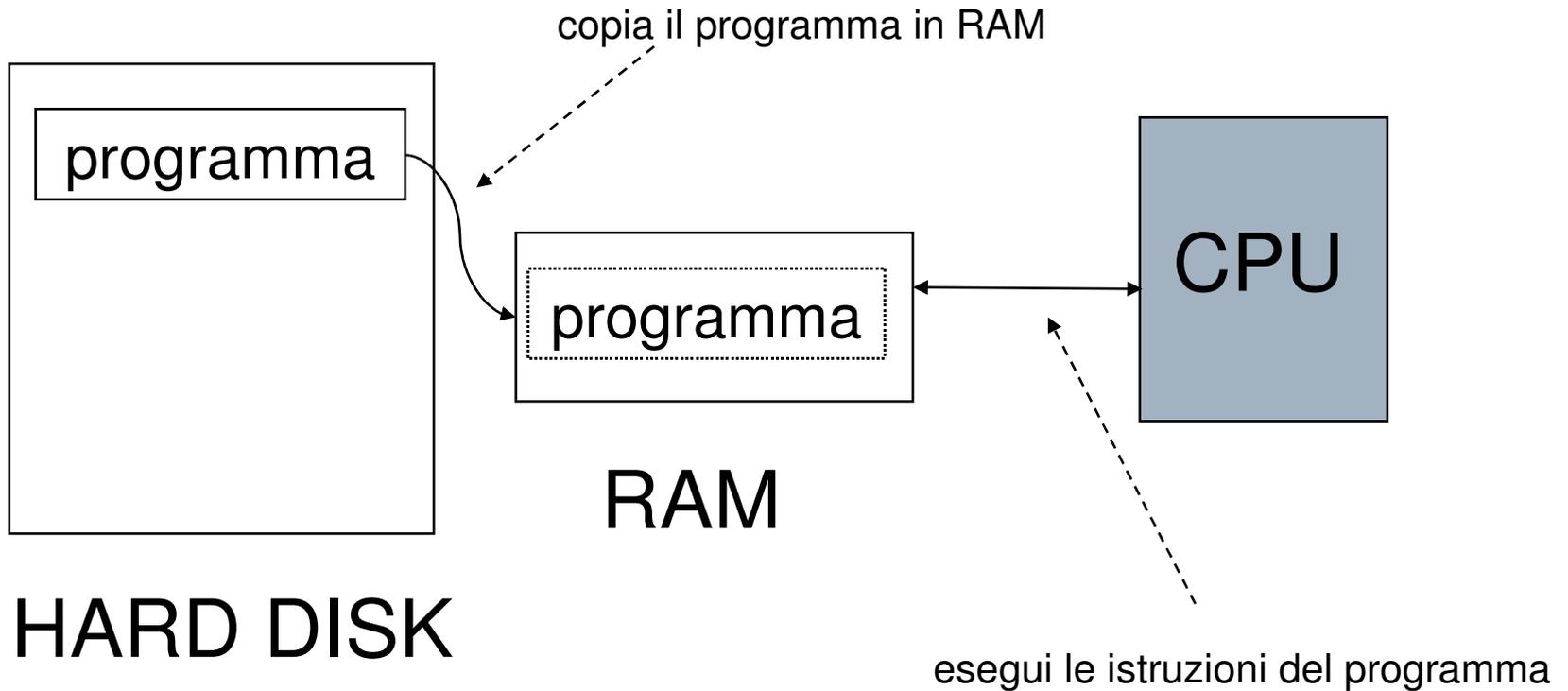
# Funzionamento

---

- Programmi e dati risiedono in file memorizzati in memoria secondaria.
  - Per essere eseguiti (i programmi) e usati (i dati) vengono copiati nella memoria primaria.
  - La CPU e' in grado di eseguire le istruzioni di cui sono composti i programmi
-

# Funzionamento

---



# Il processore - CPU (CENTRAL PROCESSING UNIT)

---

- Si occupa di **eseguire** i programmi
- I programmi che la CPU è in grado di eseguire sono scritti in **linguaggio macchina**
- **Istruzioni macchina:**

<b>Codice istruzione</b>   <b>argom. 1</b>   <b>argom. 2</b>
--

- 16 o 32 bit di lunghezza
  - gli argomenti possono mancare
-

# Un programma in linguaggio macchina (**ASSEMBLER**)

---

1000	LOAD 3568 R1
1004	LOAD 3574 R2
1008	ADD R1 R2
1012	STORE R1 3568
1016	JUMP 1000
	.....

---

# Il set di istruzioni macchina

---

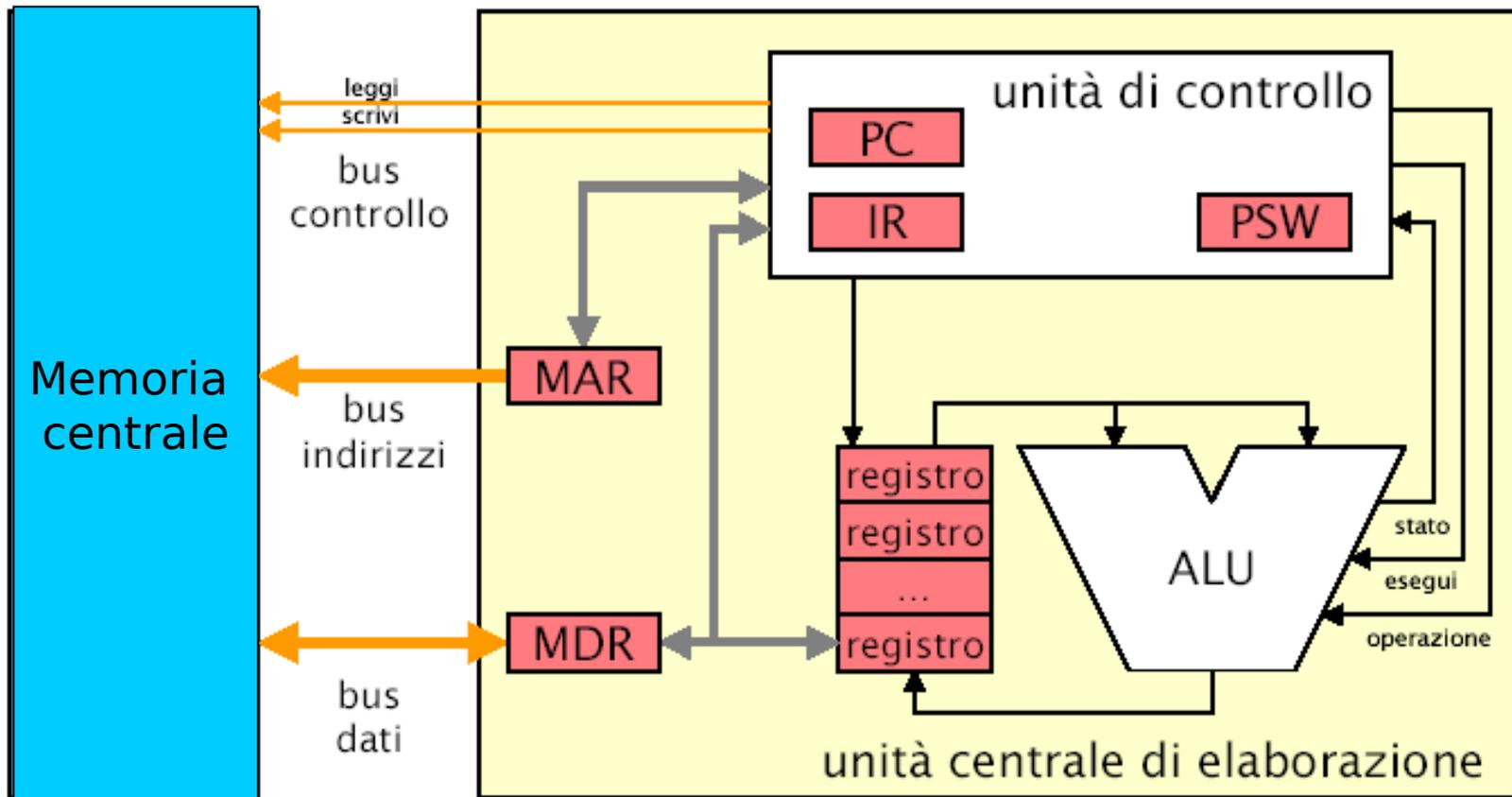
- Ogni tipo di processore e' in grado di eseguire un numero limitato (40/100) di istruzioni
  - Combinando in modo diverso sequenze anche molto lunghe di istruzioni (i programmi) si possono far fare al computer tantissime cose completamente diverse
-

# Il set di istruzioni macchina

---

- Istruzioni per l'elaborazione dei dati
    - aritmetiche
    - logiche (AND, OR, NOT)
    - relazionali (maggiore, minore, uguale, ...)
  - Controllo del flusso delle istruzioni
    - sequenza
    - selezione
    - ciclo
  - Trasferimento di informazioni
    - dati ed istruzioni fra CPU e memoria
    - dati fra CPU e dispositivi di I/O (tramite interfacce)
-

# Struttura del processore



# Le componenti della CPU:

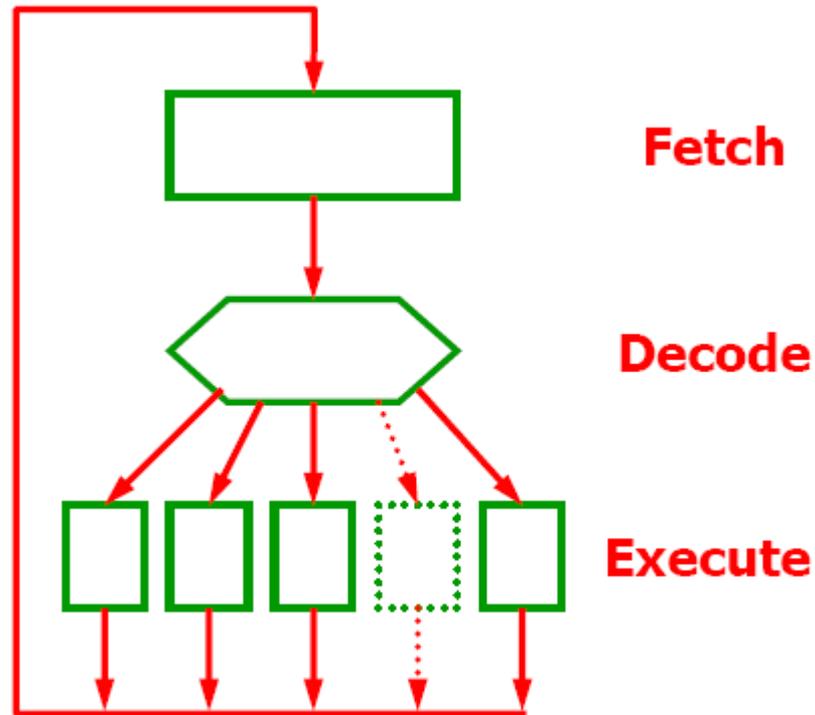
## LA CONTROL UNIT (CU)

---

- L'unità di controllo del processore esegue una istruzione svolgendo le seguenti tre operazioni di base
    - **Fetch** (lettura)
    - **Decode** (decodifica)
    - **Execute** (esecuzione)
  - Un programma è eseguito reiterando il ciclo *fetch-decode-execute* (**ciclo macchina**) per eseguire ordinatamente le sue istruzioni
-

# Il ciclo Fetch-Decode-Execute

---



# Le componenti della CPU:

## I REGISTRI

---

- piccole unita' di memoria (2, 4, 8 byte) con tempi di accesso molto piu' bassi delle celle della memoria primaria
  - Ospitano le informazioni necessarie per eseguire l'istruzione corrente
  - In numero molto limitato (10, 20, 64) si dividono in registri **speciali** e **generali**
-

# Le componenti della CPU:

## I REGISTRI SPECIALI

---

### □ Il Program Counter (PC)

- contiene l'indirizzo in memoria centrale della prossima istruzione da eseguire.
  - All'inizio dell'esecuzione di un programma viene caricato con l'indirizzo della prima istruzione di quel programma.
  - Ad ogni istruzione eseguita il PC viene modificato per contenere l'indirizzo della istruzione successiva
-

# Le componenti della CPU:

## I REGISTRI SPECIALI

---

### □ L'Instruction Register (IR)

- contiene l'istruzione correntemente in esecuzione
- la CPU legge l'istruzione contenuta nell'Instruction Register e la esegue

### □ IL Registro di stato (PSW)

- descrive lo stato corrente della esecuzione
  - segnala eventuali errori (ad es.: overflow)
-

# Le componenti della CPU:

## I REGISTRI SPECIALI

---

- **Registro Indirizzi Memoria (MAR)**
    - contiene l'indirizzo della cella da cui leggere o in cui scrivere un dato
  
  - **Registro dati Memoria (MDR)**
    - contiene il dato letto dalla memoria o da scrivere in memoria
-

# Le componenti della CPU:

## I REGISTRI GENERALI

---

- **I registri generali**
    - in numero di 8, 16, 64
    - sono usati come memorie temporanee per contenere gli operandi delle istruzioni e i risultati parziali durante l'esecuzione delle istruzioni.
-

# Le componenti della CPU:

## ARITHMETIC-LOGIC UNIT (ALU)

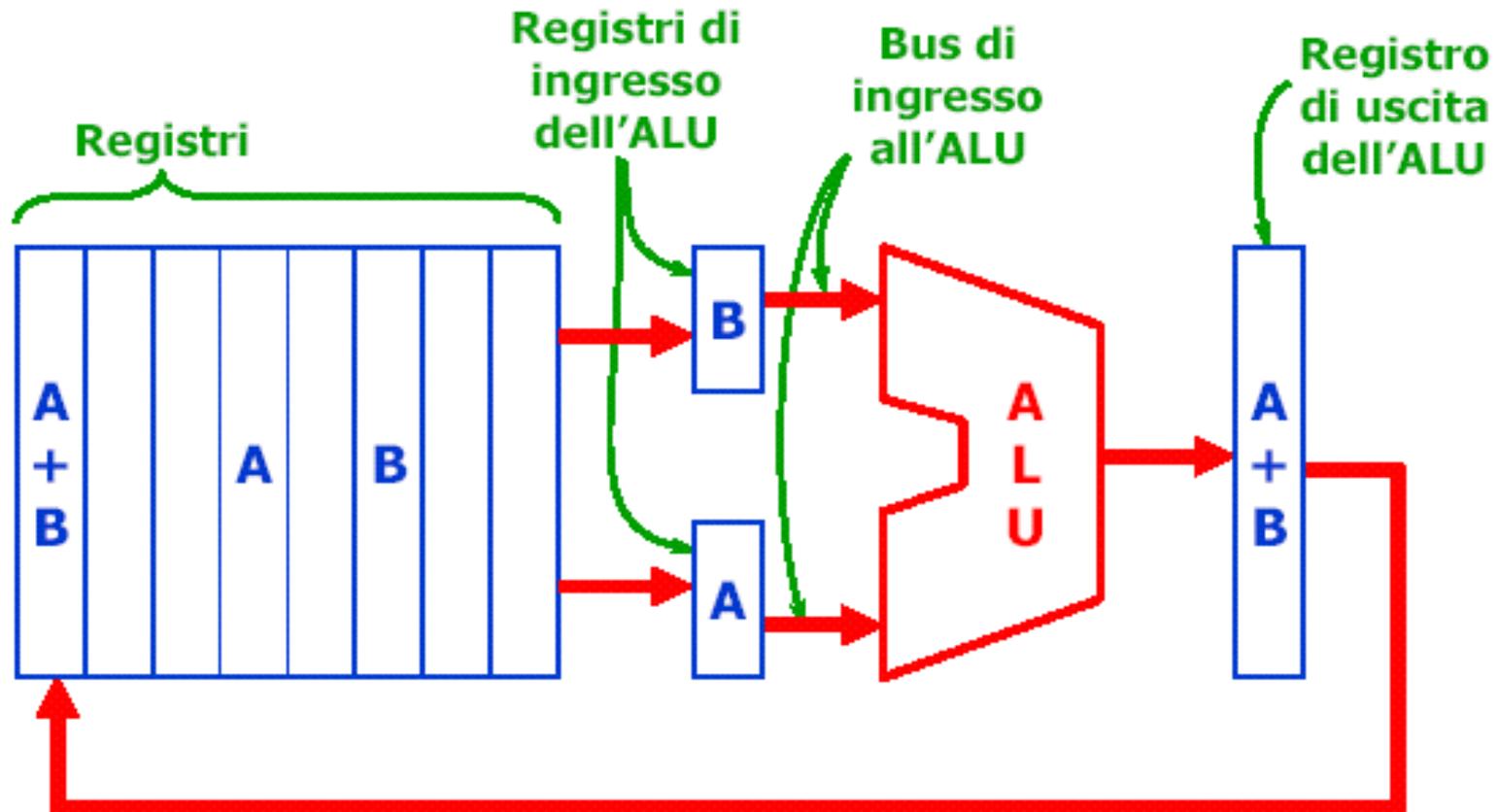
---

- Si occupa di eseguire le operazioni di tipo aritmetico/logico: somme, confronti...
  - preleva gli operandi dai / deposita il risultato delle operazioni nei: registri generali
  - A volte e' affiancata da un **co-processore matematico**
-

# Le componenti della CPU:

## ARITHMETIC-LOGIC UNIT (ALU)

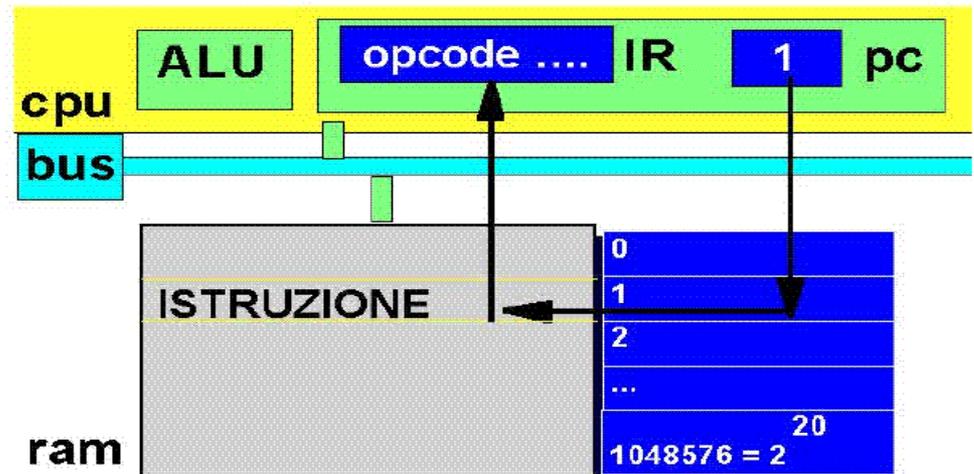
---



# Ciclo *fetch-decode-execute*

## 1) **FETCH:**

- si accede alla prossima istruzione, riferita dal registro contatore dell'istruzione (PC)
- si porta tale istruzione dalla memoria centrale al Registro Istruzioni (IR)



# Ciclo *fetch-decode-execute*

---

- **2) DECODE:** decodifica dell'istruzione
    - si individua il tipo dell'operazione e gli operandi (dati) usati
    - si trasferiscono i dati nei registri opportuni
-

# Ciclo *fetch-decode-execute*

- **3) EXECUTE:** esecuzione dell'istruzione
  - si incrementa il registro contatore dell'istruzione (PC)
  - ciascuna azione viene richiesta al componente opportuno

