



PROGRAMMAZIONE INFORMATICA

ARCHITETTURA DEL CALCOLATORE

ARCHITETTURA DEL CALCOLATORE

I calcolatori elettronici sono **macchine che eseguono ripetutamente operazioni programmate su dati digitali**. Quando parliamo di "architettura del calcolatore", ci riferiamo al modo in cui un calcolatore è progettato, alla struttura che ne definisce il funzionamento.

I primi calcolatori, come l'ENIAC, erano dispositivi programmabili in modo manuale. Ogni volta che si voleva cambiare il programma, bisognava fisicamente cambiare i cavi e gli interruttori.

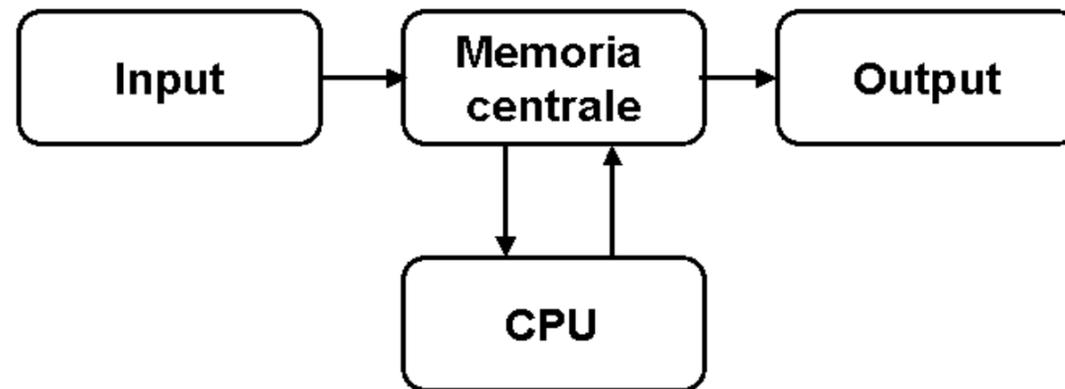
Fu solo negli anni '40 che **John von Neumann** propose un nuovo modello, chiamato appunto modello di von Neumann, che rappresenta una delle più grandi rivoluzioni nella storia dell'informatica. Questo modello prevedeva una memoria comune per istruzioni e dati, rendendo possibile memorizzare programmi in modo digitale e permettendo al calcolatore di caricare ed eseguire diversi programmi senza dover riconfigurare fisicamente la macchina.



ARCHITETTURA DI VON NEUMANN

Questo modello descrive la struttura base di un calcolatore, che è costituita da tre componenti principali:

- 1 - La **CPU**, o Unità di Elaborazione Centrale, che esegue le istruzioni.
- 2 - La **memoria**, che conserva sia i dati che le istruzioni.
- 3 - Le **unità di Input e Output**, che permettono al calcolatore di comunicare con il mondo esterno tramite un'interfaccia.



CPU

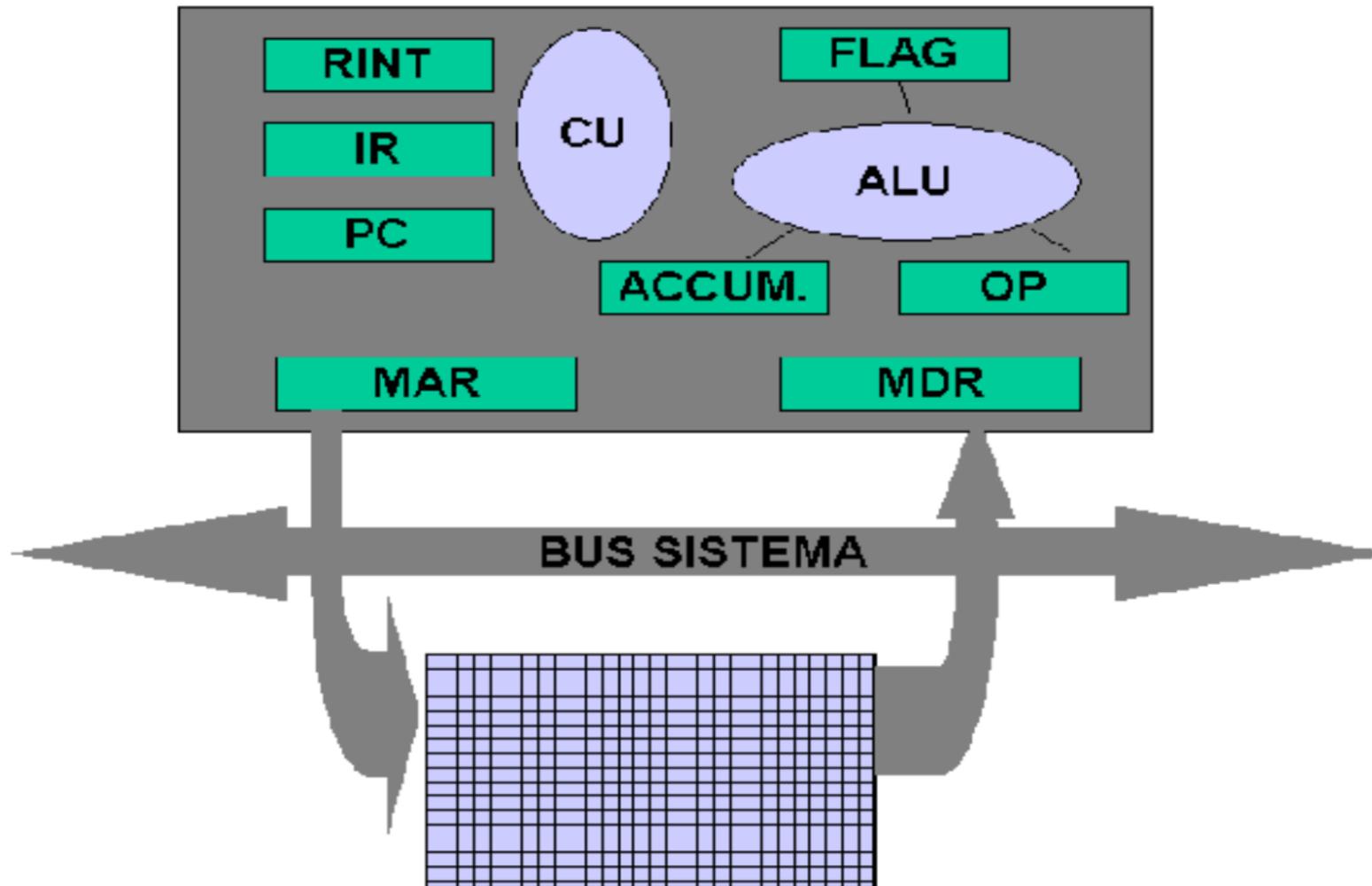
La **CPU (Central Processing Unit)** è appunto il cuore del nostro sistema. Essa contiene gli elementi circuitali necessari al funzionamento dell'elaboratore, quei circuiti integrati che operano effettivamente sulle sequenze di valori binari.

Esegue i programmi che risiedono nella memoria centrale, e coordina il trasferimento dei dati tra le varie unità. È composta da:

- una **unità di controllo (CU)**
- una **unità aritmetico-logica (ALU)**
- alcuni dispositivi di memoria detti **registri**



CPU



REGISTRI DELLA CPU

I registri principali della CPU sono:

- **Program counter (PC)** che tiene traccia dell'indirizzo della prossima istruzione da eseguire. Dopo ogni ciclo di istruzione, viene aggiornato per puntare all'istruzione successiva.
- **Registro delle Istruzioni (Instruction Register - IR)** che memorizza l'istruzione corrente che la CPU sta eseguendo. Fornisce l'input al decodificatore di istruzioni per la fase di decodifica.
- **Registro indirizzi di memoria (AR, Address Register):** in esso viene memorizzato l'indirizzo di memoria dove estrarre o scrivere il dato ed è collegato direttamente al bus di controllo.



REGISTRI DELLA CPU

- **Registro dati di memoria (DR, Data Register):** contiene i dati da leggere o da scrivere nella memoria, indirizzati dall'AR.
- **Registro di controllo (CR, Control Register):** specifica se i dati indirizzati in memoria dall'AR sono in lettura o scrittura.
- **Registri accumulatori o di lavoro (R1, R2, . . .):** contengono operandi e risultati intermedi delle operazioni eseguite.
- **Registro interruzioni (INTR, Interrupt Register):** contiene dati relativi alla richiesta di operazioni dalle periferiche di I/O.
- **Registro di stato (SR, Status Register):** contiene informazioni sul risultato dell'ultima operazione eseguita dalla ALU (overflow, riporto, segno, ecc.).



UNITA' DI CONTROLLO CU

L'Unità di Controllo (o CU, dall'inglese Control Unit) è una parte fondamentale della CPU (Central Processing Unit), responsabile del **controllo e del coordinamento di tutte le attività all'interno del processore**. La CU non esegue direttamente operazioni aritmetiche o logiche (compito dell'ALU), ma è incaricata di gestire il flusso di dati e le istruzioni, decidendo come e quando eseguire le operazioni richieste.

È responsabile dell'esecuzione del ciclo fondamentale della CPU, noto come ciclo di istruzione o **Fetch-Decode-Execute Cycle**:

- **Fetch**: la CU legge l'istruzione dalla memoria principale utilizzando l'indirizzo fornito dal Program Counter. Una volta prelevata, la memorizza nel Registro delle Istruzioni.
- **Decode**: l'istruzione viene decodificata per capire quale operazione deve essere eseguita e quali risorse (registri, memoria, ALU) sono necessarie.
- **Execute**: La CU invia i segnali di controllo appropriati per eseguire l'operazione. Se l'operazione richiede un calcolo, viene attivata l'ALU. Se l'operazione prevede la lettura o scrittura della memoria, la CU gestisce il trasferimento dei dati.



UNITA' DI CONTROLLO CU

La CU, oltre ai registri di memoria, utilizza altre componenti per gestire le operazioni della CPU:

- **Decodificatore di Istruzioni (Instruction Decoder)** che Riceve l'istruzione prelevata dalla memoria e la decodifica per determinare quale operazione deve essere eseguita. Questa operazione traduce l'istruzione da un codice macchina (binario) in segnali di controllo comprensibili dalla CPU.
- **Generatore di Segnali di Controllo** che la CU invia ai vari componenti della CPU (come l'ALU, i registri e la memoria) per coordinare le operazioni. Questi segnali specificano il tipo di operazione da eseguire, quali dati usare, dove memorizzare i risultati, ecc.



UNITA' DI CONTROLLO CU

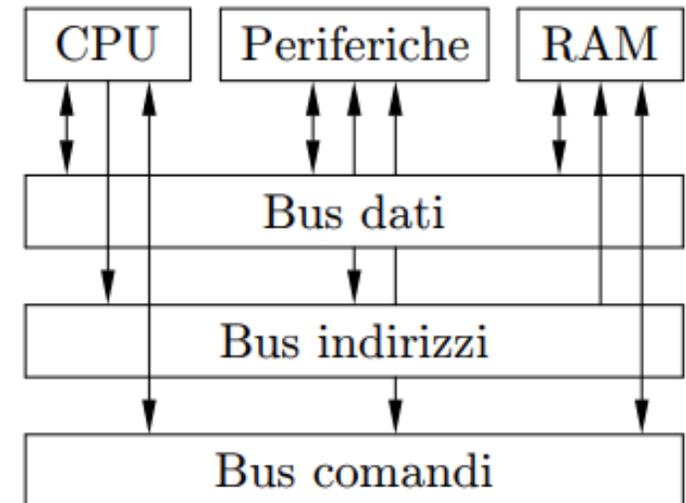
- **Clock** che scandisce il ritmo delle operazioni. Ogni ciclo di clock determina l'inizio e la fine delle varie fasi di elaborazione.
- **Bus di Sistema** che comunica con altre parti del sistema inviando segnali per attivare/disattivare altre unità funzionali e coordinare il flusso di dati all'interno della CPU.



BUS DI SISTEMA

È l'elemento che **collega CPU, memoria e periferiche** e permette lo scambio di dati tra questi. Collega due unità funzionali alla volta: una trasmette e l'altra riceve (**modalità master-slave**). Il trasferimento di dati viene sotto il controllo della CPU, e può essere **seriale** se i bit vengono trasmessi in maniera sequenziale su un'unica linea o **parallelo**, se ci sono più linee che trasmettono i bit contemporaneamente aumentando la velocità di trasmissione.

- **Bus dati:** utilizzato dalla CPU per trasmettere dati dalla memoria al registro dati o alle periferiche, e viceversa.
- **Bus indirizzi:** utilizzato dalla CPU per trasmettere il contenuto del registro indirizzi alla memoria o alle periferiche di per selezionare il dispositivo da usare.
- **Bus comandi:** utilizzato dalla CPU per inviare alla memoria o alle periferiche i segnali di comando/sincronizzazione tra i vari elementi.



UNITA' LOGICO-ARITMETICA ALU

L'Unita' aritmetico-logica è costituita da dispositivi circuitali che consentono di eseguire le **operazioni aritmetiche** somma, sottrazione, prodotto, divisione (ADD, SUB, MUL, DIV) o **logiche** (AND, OR, NOT).

Riporta nel **registro di stato (SR, Status Register)** l'esito, codificato, delle operazioni eseguite. Dal contenuto dello SR si può quindi capire se l'ultima operazione eseguita dalla ALU ha dato luogo ad overflow, riporto, risultato di segno positivo o negativo, ecc.



MEMORIA

La **memoria** è una componente essenziale di un sistema informatico e serve a **memorizzare dati e istruzioni** che la CPU può usare durante l'elaborazione. Senza memoria, un computer non potrebbe conservare temporaneamente o permanentemente dati né eseguire i programmi.

La memoria di un computer può essere classificata in due grandi categorie principali: **memoria principale** (o primaria) e **memoria secondaria**.



MEMORIA PRINCIPALE

La memoria principale è la memoria direttamente accessibile dalla CPU e comprende principalmente la **RAM** e la **ROM**.

La **RAM** è una **memoria temporanea e volatile**, ovvero perde tutti i dati quando il computer viene spento. È utilizzata per memorizzare i dati e le istruzioni che la CPU deve elaborare durante l'esecuzione dei programmi. È il luogo in cui vengono caricati i programmi e i dati attivi che la CPU deve processare in tempo reale.

La **ROM** è una memoria permanente e non volatile, ovvero conserva i dati anche quando il computer è spento. È utilizzata per memorizzare il software di base del sistema, che è essenziale per l'avvio del computer. Memorizza istruzioni fisse e programmi che non devono essere modificati dall'utente.



MEMORIA PRINCIPALE

La **memoria cache** è un particolare tipo di memoria caratterizzata da un'**elevata velocità** di accesso, dalle 10 alle 100 volte maggiore di quella della RAM, ma dimensione minore. Contiene una **copia di dati ed istruzioni di frequente utilizzo**, presenti nella RAM, ed alle quali la CPU può accedere velocemente senza dover accedere alla RAM. La memoria cache può essere di vari livelli a seconda delle dimensioni e del tempo di accesso:

- **cache di primo livello o L1**: è interna alla CPU, è la più veloce ed è di dimensione ridotta (8-64 kB)
- **cache di secondo livello o L2**: è solitamente esterna alla CPU, è più lenta di L1 ma ha dimensione maggiore (256 kB-8MB).

Quando un dato o un'istruzione vengono richiesti dalla CPU, questi vengono ricercati prima nella memoria cache (nell'ordine prima L1, poi L2): se sono presenti vengono copiati nei registri (**cache hit**), altrimenti vengono copiati dalla RAM alla cache (**cache miss**).



MEMORIA SECONDARIA

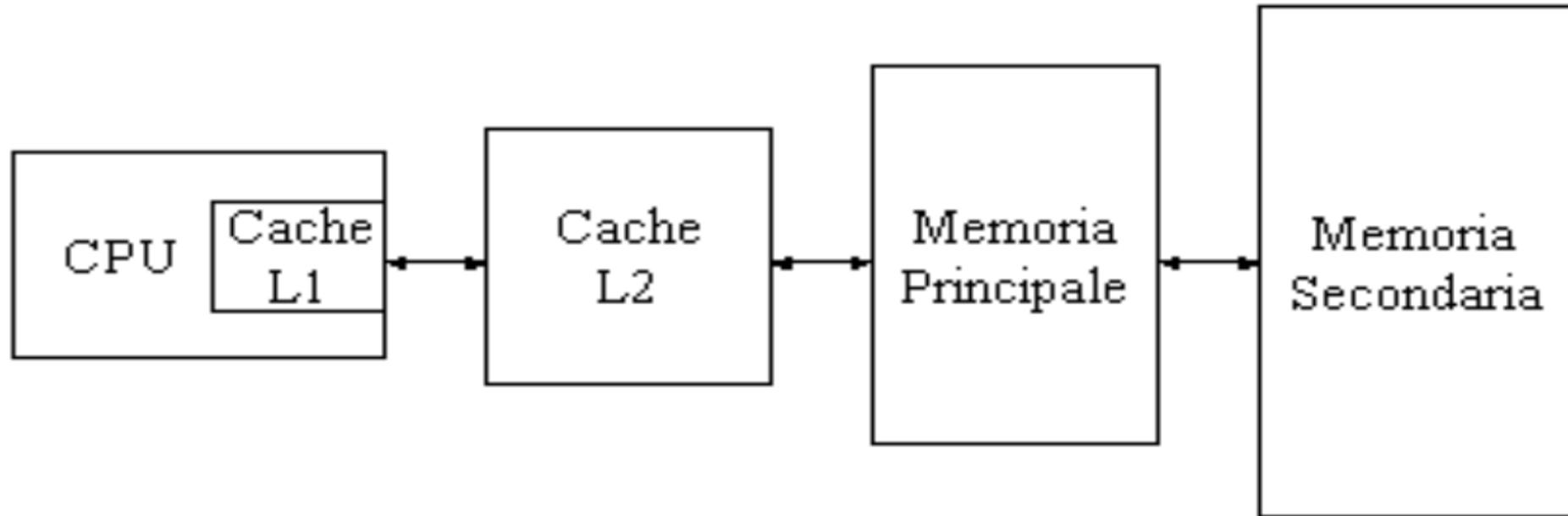
La **memoria secondaria o memoria di massa** è usata per **l'archiviazione permanente di dati e programmi**, ed è più lenta rispetto alla memoria principale. Tuttavia, ha una capacità molto maggiore. È in genere considerata esterna al modello di Von Neumann, e quindi considerabile in tale schematizzazione come **dispositivi periferici**.

Le memorie di massa più comuni sono:

- **Hard Disk Drive (HDD)**, una memoria magnetica a piatti rotanti che memorizza dati in modo permanente.
- **Solid State Drive (SSD)**, memoria a stato solido basata su chip di memoria flash, che è molto più veloce rispetto agli HDD.
- **Memoria flash**, una forma di memoria non volatile utilizzata in dispositivi come chiavette USB, schede SD e SSD.
- **CD, DVD, Blu-ray**, memoria non volatile e più economica, ma con capacità di archiviazione relativamente bassa e tempi di accesso più lunghi rispetto agli SSD e HDD.



GERARCHIA MEMORIE



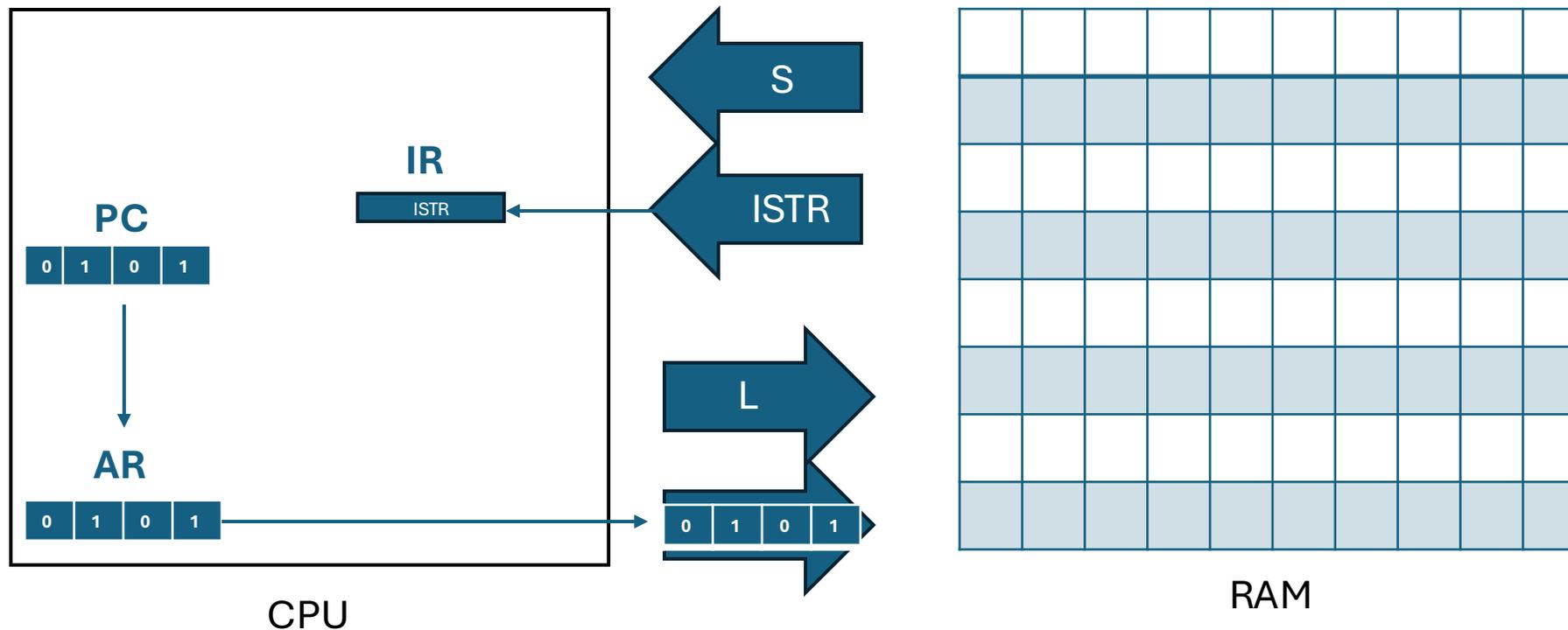
MEMORIE PICCOLE
E VELOCI



MEMORIE CAPACI E
LENTE

ESECUZIONE MODELLO VONN NEUMANN

- Nella **fase di fetch**, La CU utilizza il Program Counter (**PC**) per ottenere l'indirizzo della prossima istruzione RAM. La richiesta di ricerca dell'istruzione viene realizzata mediante la scrittura dell'indirizzo nel registro **AR** e l'attivazione di un segnale di controllo sul **bus** che determina l'azione di lettura. Poi l'istruzione viene caricata nel Registro delle Istruzioni (**IR**).



ESECUZIONE MODELLO VONN NEUMANN

- Nella **fase di decode**, l'istruzione è stata caricata nel registro delle istruzioni, viene inviata al **Decodificatore di Istruzioni**, che la traduce in un insieme di segnali di controllo comprensibili dalla CPU. Questo passaggio permette alla CU di capire che tipo di operazione deve essere eseguita. Se l'istruzione richiede operandi (come un'operazione aritmetica tra due numeri), la CU identifica quali registri o aree di memoria contengono i dati necessari.
- La **fase di execute** può comprendere diversi passaggi a seconda dell'istruzione, tra cui l'**attivazione dell'ALU** se l'istruzione prevede un'operazione aritmetica o logica (la CU invia segnali di controllo all'ALU per eseguire l'operazione sugli operandi), e la **gestione della memoria** se l'istruzione richiede il trasferimento di dati (lettura o scrittura) con la memoria principale. Inoltre, in questa fase il **Program Counter** viene aggiornato per puntare alla successiva istruzione da eseguire.

