



FONDAMENTI DI INFORMATICA

Lezione n. 10

- CPU-MEMORIA
- CICLO PRINCIPALE CPU
- ARCHITETTURA BASE
- UNITÀ DI ESECUZIONE E UNITÀ DI CONTROLLO

In questa lezione viene analizzata l'architettura di base di un sistema di elaborazione. Essa è formata da un'unità centrale di elaborazione (o CPU - *Central Processing Unit*) e dalla memoria che, nel modello di Von Neuman, immagazzina dati e istruzioni. Sono poi descritte le operazioni elementari svolte dalla CPU e la sua interazione con la memoria. Descriveremo poi le caratteristiche generali di una unità di controllo e le tecniche di realizzazione.

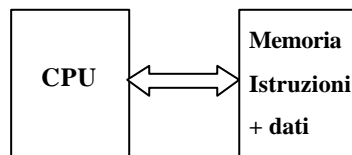


UNITA' DI ELABORAZIONE

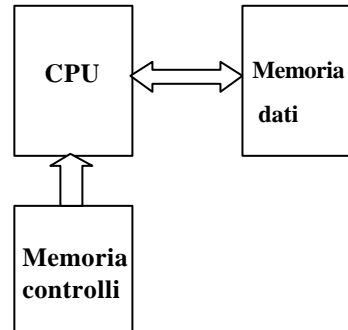
Central Processing Unit + Memoria

Compito: Eseguire i comandi contenuti nella memoria.

Architettura di VonNeuman

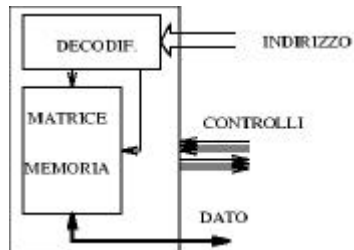


Architettura Harward





LA MEMORIA



La memoria è costituita da celle che memorizzano bit di informazione raggruppate in parole (8 bit) caratterizzate da un indirizzo.

Sulla singola cella di memoria sono possibili le operazioni di lettura e scrittura.

Le linee di dato consentono l'ingresso e l'uscita delle informazioni dalla memoria.

Le linee di dato sono bidirezionali mentre le linee di indirizzo consentono di trasmettere l'informazione solo dalla CPU alla memoria.



IL CICLO DI BASE

Il ciclo base di una qualsiasi istruzione:

- **FETCH:** Lettura dell'istruzione in memoria.
- **ESECUZIONE:** Lettura dei dati necessari.
Svolgimento dell'operazione.

- La fase di **FETCH** consente l'acquisizione della nuova istruzione.
- La fase di **ESECUZIONE** consente lo svolgimento della operazione.
- La fase di **FETCH** è eguale per tutte le istruzioni che ovviamente si differenziano nella fase di **ESECUZIONE**.
- Al termine della fase di esecuzione viene eseguita la fase di **FETCH** della istruzione successiva.



SEQUENZA ISTRUZIONI

Ogni istruzione contiene in modo esplicito o implicito l'indicazione dell'istruzione successiva.

Modo esplicito: il codice dell'istruzione contiene l'indirizzo della cella di memoria dove è contenuta l'istruzione da eseguire.

Modo implicito: l'indirizzo della istruzione successiva è contenuto in un registro interno alla CPU.

La sequenza delle istruzioni è definita dal programma.



INTERRUZIONE

La CPU sorveglia gli altri elementi del sistema:

- Può sospendere la sequenza prevista e svolgere operazioni di maggiore urgenza.
- Al termine ritorna a svolgere la sequenza interrotta.

P INTERRUPT o INTERRUZIONE

Casi nei quali questo evento si può verificare:

- Mancanza della tensione di alimentazione.
- Errori hardware.
- Operazioni di I/O.

Una interruzione sospende l'esecuzione del flusso principale di istruzioni e forza la CPU a svolgere una nuova sequenza di operazioni associata alla particolare richiesta.



CPU - MEMORIA

- **La CPU (componente unico) deve essere realizzata con la migliore tecnologia possibile (anche costosa).**
- **La memoria principale (di grande capacità) deve essere realizzata con la tecnologia più conveniente (economicamente).**
- **t_{CPU} : tempo richiesto per la più veloce operazione della CPU (2 - 20 ns).**
- **t_M : intervallo minimo fra due accessi alla memoria (20 ns - 100 ns).**

Questa differenza di prestazioni negli ultimi anni si è accentuata. Le prestazioni delle CPU continuano ad aumentare velocemente mentre il tempo di accesso delle memorie commerciali non è diminuito in modo significativo negli ultimi anni.



REGISTRI INTERNI

Le operazioni fra registri interni avvengono con tempi dell'ordine di t_{CPU} .

Ogni istruzione è trasformata in un certo numero di operazioni RTL interne alla CPU.

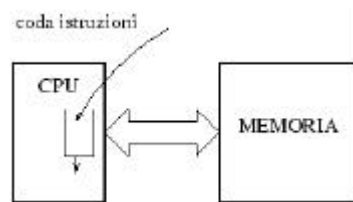
Alcune CPU dispongono di centinaia di registri interni nei quali memorizzare dati temporanei.

Questa tecnica utilizzata nelle architetture RISC.



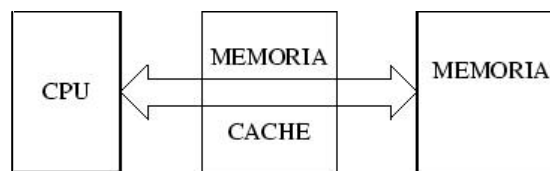
CPU - MEMORIA

Altre tecniche architetturali consentono di accedere alla memoria (ad esempio per l'operazione di FETCH) contemporaneamente a fasi di esecuzione che non richiedono accessi alla memoria.



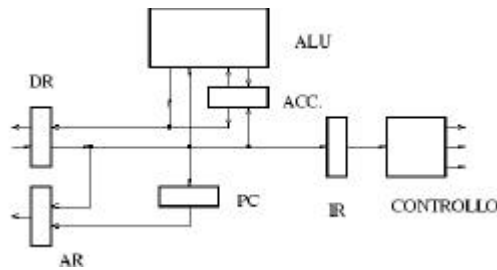
CPU - MEMORIA

Per ridurre i tempi di accesso alla memoria si interpongono tra la memoria principale e la CPU memorie secondarie con prestazioni più elevate. Questa metodologia verrà discussa in dettaglio in seguito.





CPU



La figura rappresenta lo schema più semplice, a livello RTL, di una CPU di tipo Von Neuman.

DR: Data Register - Memorizza i dati provenienti e diretti alla memoria.

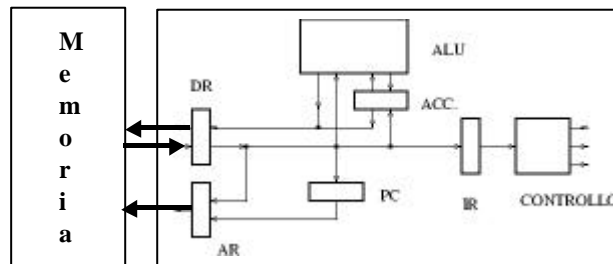
AR: Address Register - Memorizza gli indirizzi da comunicare alla memoria.

PC: Program Counter o Contatore di Programma - Memorizza l'indirizzo della istruzione da eseguire.

IR: Instruction Register - Memorizza il codice dell'istruzione da eseguire.

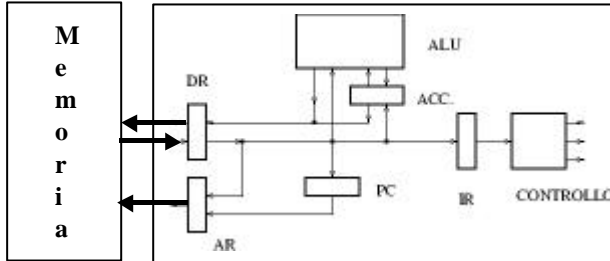


CPU



ALU: Arithmetic Logic Unit - Esegue le operazioni aritmetico-logiche.

ACC: Accumulatore - Immagazzina dati in ingresso e uscita all'ALU.



ESECUZIONE ISTRUZIONI

Fase di Fetch comune a tutte le istruzioni:

- AR \leftarrow PC ;
- DR \leftarrow M(AR) ; Lettura in memoria dell'istruzione
- IR \leftarrow DR ; Codice di istruzione all'unità di controllo
- PC \leftarrow PC + 1 ; Si prepara il PC per la lettura della prossima istruzione

Segue la esecuzione dell'istruzione.



STRUTTURA DI BASE

ESECUZIONE-CONTROLLO.

Una CPU è formata da:

- Unità di esecuzione o percorso dati (data path).
- Unità di controllo (a più livelli):
 - della sequenza delle istruzioni,
 - della esecuzione delle istruzioni.





TECNICHE DI IMPLEMENTAZIONE

- **Controllo cablato.**
 - Circuito sequenziale speciale che realizza la funzione desiderata.
 - Meno costoso e più conveniente per sistemi semplici.
 - Di difficile progettazione per sistemi complessi.
- **Controllo a microprogramma.**
 - Microistruzioni che contengono direttamente le indicazioni per l'attivazione dei segnali.
 - Progettazione concorrente di parte operativa e controllo.
 - Progettazione strutturata.
 - Semplicità per l'aggiornamento dell'HW.



CONTROLLO CABLATO

Metodi di progetto:

- Generatori di sequenze basati su elementi di ritardo (asincroni) o contatori (sincroni).
- Macchine a stati finiti (progetto tradizionale di circuiti sequenziali).

GENERAZIONE DI SEQUENZE

- Da un diagramma di flusso (o un programma, ...) che definisce il comportamento della parte operativa.
- Si deve generare una sequenza temporale di vettori:

$$C_1, C_2, \dots, C_n \quad \text{con} \quad C_i = \{c_1, c_2, \dots, c_m\}$$

ai tempi: t_1, t_2, \dots, t_n .



UNITA' DI CONTROLLO CPU

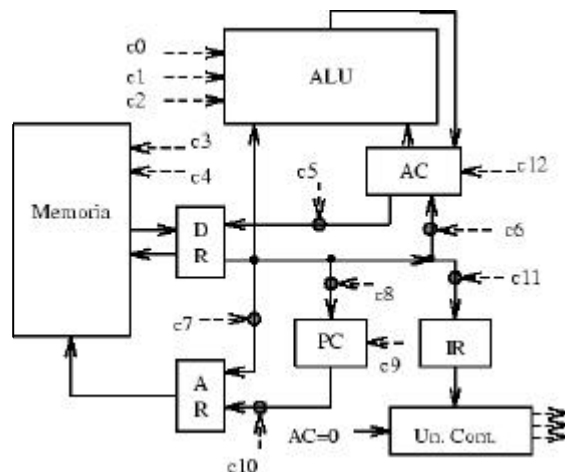
Si consideri una semplice CPU con le seguenti caratteristiche:

- 8 istruzioni a un indirizzo.
- Unità aritmetica che svolge le seguenti funzioni: ADD, COMP, AND.

LOAD X	AC	← M(X)
STORE X	M(X)	← AC
ADD X	AC	← AC + M(X)
AND X	AC	← AC & M(X)
JUMP X	PC	← X
JUMPZ X	se AC=0 allora PC ← X	
COMP	AC	← \overline{AC}
RSHIFT	Scorr. destra di AC	



CPU





SEGNALI DI CONTROLLO

C ₀	AC	\bar{U} AC+DR
C ₁	AC	\bar{U} AC \bar{U} DR
C ₂	AC	\bar{U} AC
C ₃	DR	\bar{U} M(AR) (read)
C ₄	M(AR)	\bar{U} DR (write)
C ₅	DR	\bar{U} AC
C ₆	AC	\bar{U} DR
C ₇	AR	\bar{U} DR(ADR)
C ₈	PC	\bar{U} DR(ADR)
C ₉	PC	\bar{U} PC + 1
C ₁₀	AR	\bar{U} PC
C ₁₁	IR	\bar{U} DR(OP)
C ₁₂	Scorr. destra di AC	