

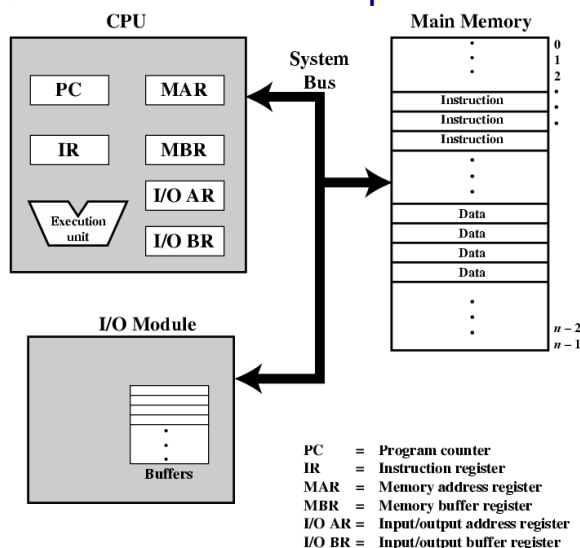
Sommario degli argomenti di lezione

- Introduzione (capitolo 1)
- Rappresentazione binaria/ottale/esadecimale; algebra di Boole
- Struttura del Calcolatore (capitolo 3)
- Memoria Cache (capitoli 4)
- Memoria Interna (capitolo 5)
- Memoria Esterna (capitolo 6)
- Input/Output (capitolo 7)
- Sistemi di Numerazione (capitolo 9)
- (<http://www.box.net/shared/4zcr479igi#/shared/4zcr479igi/1/26116716/270740128/1>)
- Aritmetica del Calcolatore (capitolo 10)
- Linguaggi Macchina (capitoli 12-13)
- Struttura e Funzione della CPU, Pipelining (capitolo 14)
- Processori RISC (capitolo 15)
- Cenni sui Calcolatori Multicore (capitolo 18)

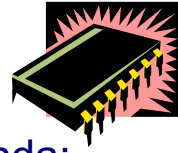
Struttura del Calcolatore

Cercheremo di rispondere alle seguenti domande:

- Qual'è la funzione di un calcolatore ?
 - Eseguire un programma
- Quali sono le sue componenti ?



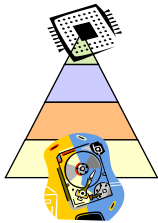
Memoria Cache



Cercheremo di rispondere alla seguente domanda:

- Memorie RAM:
 - quelle veloci sono molto costose e di capacità limitata;
 - quelle di grossa capacità sono lente ma economiche;
- come è possibile combinare l'uso di questi due tipi di memorie in modo da avere una memoria RAM:
- di grossa capacità
 - abbastanza veloce
 - economica ?

Gerarchia di memoria:



- la CPU richiede i dati necessari alla memoria veloce (Cache)
- se la Cache non li contiene, li richiede alla memoria più lenta che contiene tutti i dati
- quindi dati riferiti frequentemente si troveranno in Cache



Memoria Interna



Cercheremo di rispondere alle seguenti domande:

- Che tipo di memoria, e quale tecnologia, viene usata per la memoria Cache ?

SRAM (static RAM)

- Che tipo di memoria, e quale tecnologia, viene usata per la RAM più lenta ma più capiente ?

DRAM (dynamic RAM)



Memoria Esterna



Cercheremo di rispondere alle seguenti domande:

- Come viene memorizzata l'informazione in un disco rigido ?
- Come si accede all'informazione in un disco rigido ?
- Come si possono usare più dischi rigidi in parallelo in modo da recuperare più velocemente le informazioni e preservarla da guasti ?

Redundant Array of Independent Disk (RAID)

- Come funzionano i dischi SSD (*Solid State Disk*) ?
- Come viene memorizzata l'informazione in un CD-ROM o DVD ?
- Con quale tecnologia ?



Input/Output



Cercheremo di rispondere alle seguenti domande:

- Come vengono organizzate e gestite le operazioni di input/output fra la CPU ed i dispositivi esterni (disco rigido, tastiera, video, ...) ?
- Cosa è un modulo di I/O ? Quali funzionalità svolge ? Quale è la sua struttura ?
- Quali sono le alternative possibili di gestione dell' I/O ?
 - I/O da programma
 - I/O driven (guidato da interrupt)
 - accesso diretto alla memoria (DMA)
- Cosa è un canale o processore di I/O ?



Sistemi di Numerazione

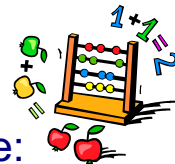


Cercheremo di rispondere alle seguenti domande:

- Come vengono rappresentati i numeri decimali in un calcolatore ?
- Come si converte un numero in rappresentazione decimale in rappresentazione binaria, e viceversa ?
- Cosa è la rappresentazione esadecimale ? Perché è utile ?



Aritmetica del Calcolatore



Cercheremo di rispondere alle seguenti domande:

- Come vengono rappresentati i numeri interi in un calcolatore ?
- ...e come vengono realizzate le relative operazioni aritmetiche ?
- Come vengono rappresentati i numeri reali in un calcolatore ? virgola fissa, virgola mobile
- ...e come vengono realizzate le relative operazioni aritmetiche ?



Linguaggi Macchina

Cercheremo di rispondere alle seguenti domande:



- Che tipo di istruzioni sono eseguite a livello macchina ?
- Come vengono specificati eventuali operandi ?
- Dove risiedono gli operandi delle istruzioni ? Come si indica la loro locazione ?
- Perché esistono vari modi di *indirizzamento* per gli operandi ?
- Come si rappresentano i vari modi di indirizzamento per una istruzione ?
- Cosa è un *set di istruzioni* ?



Struttura e funzione della CPU

Cercheremo di rispondere alle seguenti domande:

- Come si differenziano le CPU dal punto di vista dei registri e del loro uso ?
- Come può una CPU organizzare i calcoli relativi alla esecuzione delle istruzioni in modo da essere più efficiente ?

efficiente:

utilizzare al meglio tutte le componenti della CPU in modo da non avere componenti inattive (inefficienza)

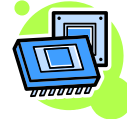
pipelining:

- ogni istruzione, quando è eseguita, “passa” attraverso stadi separati di esecuzione (ad es.: fetch, decodifica, calcolo indirizzi,...);

- **idea:** mantenere tutti gli stadi attivi eseguendo più istruzioni, come in una catena di montaggio (ogni stadio lavora contemporaneamente su una istruzione diversa)



Processori RISC



Cercheremo di rispondere alle seguenti domande:

- Quali sono le istruzioni più frequentemente utilizzate dai programmi scritti nei linguaggi ad alto livello ?
- Si può utilizzare la risposta alla prima domanda per decidere quale deve essere il set delle istruzioni da implementare per una CPU ?

Reduced Instruction Set Computer (**RISC**):

- un ridotto insieme di istruzioni, tutte dello stesso formato
- un elevato numero di registri o l'impiego di un compilatore che ne ottimizza l'uso
- particolare attenzione all'ottimizzazione della pipeline

