

Corso di Calcolatori Elettronici I

Calcolatori Elettronici:
primi cenni introduttivi

Roberto Canonico

Università degli Studi di Napoli Federico II

A.A. 2019-2020





Riprendiamo alcuni concetti fondamentali, già trattati nel corso di *Fondamenti di Informatica*.

- *Informazione*
- *Dato*
- *Algoritmo*
- *Sistema di Elaborazione*
- *Calcolatore Elettronico*



- L'informazione è legata al concetto di **scelta** (Claude Shannon, 1948)
- Un'**informazione** è la scelta di un singolo elemento fra tutti gli elementi di un insieme
- L'insieme su cui si opera la scelta è il **tipo** dell'informazione
- L'elemento scelto è il **valore** dell'informazione
- Il significato della scelta è quello che chiamiamo **attributo** dell'informazione
- Si dice che un'informazione è una tripla **{tipo, valore, attributo}**
- La scelta più semplice che si possa operare è quella tra due soli valori (*scelta binaria*)
- Si usa il termine **dato** per indicare una rappresentazione del valore di un'informazione secondo un certo *codice*



- *Elaborare* delle informazioni significa:
 - forniti i valori di un insieme di informazioni di partenza, opportunamente rappresentati (*dati di input*, X)
 - determinare i valori di un insieme di informazioni risultato, opportunamente rappresentati (*dati di output*, Y)
 - secondo una certa regola F
- Chiamiamo *elaborazione* il processo che consente di produrre i dati di output a partire dai dati di input
- $Y = F(X)$



- Il concetto di algoritmo è legato a quello di esecutore
- L'**esecutore** è un'entità (persona o macchina) in grado di eseguire un certo insieme di azioni di elaborazione elementari $Y=F(X)$
- Eseguendo una sequenza finita di azioni di elaborazione elementari su dei dati di input, un esecutore può risolvere problemi complessi
- Un **algoritmo** è una **sequenza** (1) **finita** (2) **di azioni elaborative** (3) che **risolve** un problema (4)
- Durante l'esecuzione dell'algoritmo, l'esecutore può produrre dei dati intermedi
- L'algoritmo deve essere **deterministico** (produce sempre stessi risultati se rieseguito) e **automatico** (non occorrono interventi dall'esterno)
- Un *sistema di elaborazione* è una macchina in grado di eseguire automaticamente un algoritmo



- Con il termine *Calcolatore Elettronico* intendiamo un sistema di elaborazione realizzato in tecnologia elettronica, che opera eseguendo sequenze di istruzioni (programma) prelevate da un organo di memoria, su dati rappresentati mediante codici binari



Un *sistema* è una entità complessa costituita da una pluralità di elementi collegati tra di loro e che svolgono una assegnata funzione.

- Sistemi naturali
- Sistemi ingegneristici
- Sistemi sociali
- ...

Per descrivere un sistema si distingue tra:

- il comportamento che esibisce (*cosa fa*)
- la sua struttura (*come è fatto*)

Proprietà di un sistema:

- Proprietà Funzionali (attengono al comportamento del sistema)
- Proprietà Non-Funzionali (performance, affidabilità, sicurezza, ...)



- *Progetto di un sistema*: determinazione di una struttura del sistema che sia
 - compatibile con un insieme di specifiche comportamentali e funzionali
 - realizzabile
 - economicamente conveniente
- I sistemi ingegneristici sono tipicamente realizzati attraverso la composizione di sotto-sistemi
- Metodologie di progetto top-down: scompongono la struttura del sistema in sotto-sistemi più semplici opportunamente interconnessi
- Nel progetto di sistemi hardware, si riconduce il progetto a sotto-sistemi elementari noti, assunti come "mattoni elementari" della realizzazione (porte logiche, macchine fondamentali come multiplexer, contatori, ecc. ...)

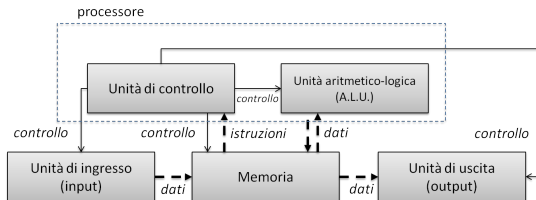


- La caratteristica principale di un *calcolatore elettronico* consiste nella possibilità di svolgere compiti diversi utilizzando programmi (*software*) che governano in modi differenti lo stesso *hardware*
 - Useremo da qui in avanti il termine *Sistema di Elaborazione* come sinonimo di *calcolatore elettronico*
- In questo corso presenteremo la struttura interna di un *Sistema di Elaborazione* descrivendone i suoi principali sotto-sistemi e la loro organizzazione, secondo modelli "architetturali" di alto livello
- Abbiamo visto che il passo decisivo verso la realizzazione dei moderni sistemi di elaborazione è consistito nell'aver concepito le macchine a *programma registrato*



Il modello di von Neumann identifica, in un calcolatore, i seguenti sottosistemi:

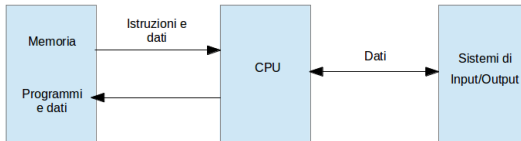
- Processore (CPU)
- Memoria centrale
- Input/Output
- Sottosistema di interconnessione



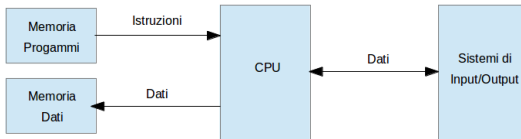
Architetture di riferimento

Ad un primo livello di dettaglio, rispetto all'interazione con la memoria, l'architettura di un sistema di elaborazione si può ricondurre a due modelli fondamentali:

- architettura di von Neumann o di Princeton
 - una sola memoria per dati ed istruzioni



- architettura di Harvard
 - due memorie distinte per dati ed istruzioni
 - possibili trasferimenti contemporanei dalle due memorie





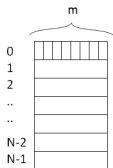
- È in grado di eseguire un set di azioni elaborative elementari (*istruzioni*) più o meno complesse
 - Ogni processore è caratterizzato dal suo proprio *repertorio di istruzioni*
- Le istruzioni sono comandi espliciti che
 - trasferiscono dati sia all'interno del processore sia tra processore e memoria o dispositivi di I/O
 - effettuano operazioni aritmetiche e logiche su dati
 - governano il flusso di esecuzione delle istruzioni
- La sequenza di istruzioni da eseguire (*programma*) è immagazzinata nella memoria centrale
- Il processore *esegue il programma* prelevando ed eseguendo le istruzioni "una ad una" dalla memoria
 - Facciamo per il momento riferimento a processori secondo l'architettura di Von Neumann
 - Faremo cenno più avanti nel corso ad architetture avanzate nelle quali alcune unità del sistema di elaborazione operano con un certo grado di *parallelismo*
 - I processori che operano secondo l'architettura di Harvard, ad esempio, possono accedere in parallelo a dati ed istruzioni



- Il funzionamento del processore è governato da un segnale periodico che ha funzione di tempificazione: il *clock*
- Detto T il periodo del segnale di clock, ed $f = 1/T$ la sua frequenza, osserviamo che maggiore è la frequenza del clock, maggiore sarà il numero di *istruzioni per secondo* che il processore è in grado di eseguire
- Il *numero (medio) di istruzioni per secondo* non dipende solo dalla frequenza del clock ma anche dalla durata media, in numero di cicli di clock, della singola istruzione
 - Si misura in MIPS (*Million Instructions Per Second*)
 - Prestazioni dei supercalcolatori espresse in multipli di FLOPS (*FLOating-point Operations Per Second*)
- Per aumentare il numero di istruzioni per secondo eseguite, il primo approccio consiste nell'aumentare la frequenza di clock
 - Limiti tecnologici impediscono, per una data circuiteria, di spingere la frequenza di clock oltre certi limiti
- Primi microprocessori: clock a frequenze di qualche *MHz*
- Processori attuali: clock a frequenze di qualche *GHz*

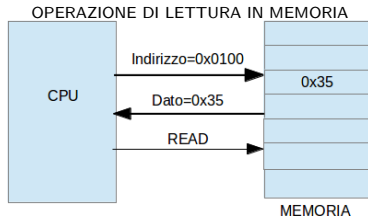
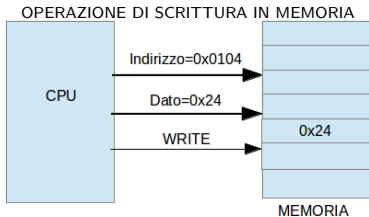


- La memoria centrale di un computer è organizzata come un array di stringhe di bit di lunghezza m , dette *celle* o *locazioni*
- La locazione è l'unità minima di memoria accessibile dal processore
- Ogni locazione è individuata da un *indirizzo*, cioè un intero compreso tra 0 ed $N-1$, con $N = 2^c$
 - $[0, N - 1] = \text{SPAZIO DI INDIRIZZAMENTO}$
- La memoria centrale è ad accesso casuale cioè il tempo di accesso *non dipende* dalla posizione del dato
- La memoria centrale è prevalentemente *volatile* ma contiene anche porzioni *non-volatili* (ROM) per contenere programmi che devono essere disponibili al calcolatore alla sua accensione (BIOS, bootloader)





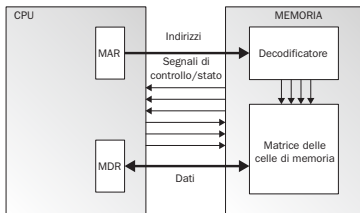
- Le unità funzionali fondamentali del calcolatore si scambiano informazioni (dati, indirizzi, segnali di controllo/stato) mediante appositi sistemi di interconnessione
- La forma più comune di sistema di interconnessione è il *bus*: un fascio di fili che trasportano bit
- Un calcolatore contiene uno o più bus
 - *bus interni* al processore
 - *bus esterni* per il collegamento del processore con le altre unità



Nella interazione CPU-memoria l'iniziativa del trasferimento è presa dalla CPU, che ha un ruolo di *master* rispetto alla memoria (*slave*)

La CPU si interfaccia con la memoria attraverso due registri:

- MAR - *Memory Address Register*
 - Attraverso MAR la CPU fornisce alla memoria l'indirizzo della locazione interessata alla operazione di lettura o scrittura
- MDR - *Memory Data Register*
 - In MDR transita il dato trasferito da o verso la memoria



- La CPU scambia con la memoria anche altre informazioni di controllo/stato attraverso apposite linee