
Fondamenti di Informatica
per la Sicurezza
a.a. 2003/04

◇ ***Architettura di von Neumann*** ◇

Stefano Ferrari



Università degli Studi di Milano
Dipartimento di Tecnologie dell'Informazione

- Il calcolatore può essere visto come un sistema strutturato di componenti tra loro interconnesse.
- Descrizione gerarchica
- Componenti:
 - interagiscono (tra loro e con l'esterno)
 - realizzano funzionalità
 - scomponibili
- Funzionalità principali:
 - scambio dati con l'esterno
 - memorizzazione
 - elaborazione
 - controllo

- Elaborazione: compromesso tra le caratteristiche
- Memorizzazione:
 - breve termine
 - lungo termine
- Trasferimento dati: Periferiche
 - I/O
 - rete
- Controllo:
 - utente
 - CPU

Il calcolatore è una macchina estremamente flessibile:

- le funzionalità vengono fornite dall'hardware
- la specializzazione viene fornita dal software

Architettura di Von Neumann

- unità elaborazione centrale (*Central Processing Unit* — CPU)
- interazione con l'esterno tramite dispositivi detti *periferiche*
- memoria costituita da un insieme di elementi identificabili tramite il loro *indirizzo*
- unità connesse tramite *bus* su architettura *master-slave*

- Interagiscono con l'utente
- Comunicano con l'interfaccia di I/O

- È il componente principale
- funzioni:
 - controllo
 - coordinamento
 - elaborazione
- tecnologia: microelettronica → microprocessori

- ospita i dati coinvolti nell'elaborazione (talvolta anche quelli delle periferiche)
- insieme di celle adiacenti
- identificate da un indirizzo

- BUS: linea a cui sono collegate *contemporaneamente* diverse unità
- *master*: CPU *slave*: altre unità

pro

- semplicità
- estendibilità
- standardizzabilità

contro

- lentezza
- limite alla capacità del canale
- sovraccarico CPU

Il bus si suddivide in tre componenti:

- bus dati
- bus indirizzi
- bus controllo

L'idea fondamentale (attribuita a Von Neumann) ancora oggi seguita nella realizzazione dei sistemi di calcolo:

- codificare le istruzioni in forma numerica
- inserirle nella memoria centrale

Principi:

- dati e istruzioni memorizzati in un'unica memoria che permette lettura e scrittura
- la memoria è costituita da celle uguali, indirizzate dalla loro posizione
- le istruzioni vengono eseguite in modo sequenziale

- Il linguaggio in cui vengono codificati i programmi è detto linguaggio macchina, o *assembly*.
- Assenza di struttura o tipo di dato.
- Istruzioni semplici e in numero ridotto.
- Le istruzioni sono composte da codice operativo più eventuali operandi.

Esempio: <Somma> <Reg1> <Reg2>

- Codice operativo:
 - CPU della stessa famiglia hanno lo stesso codice operativo.
 - CPU di diversi produttori possono adottare lo stesso codice operativo → compatibilità

Quando il programma e i dati risiedono in memoria, la CPU opera in modo ciclico:

fetch viene dalla memoria recuperata l'istruzione da eseguire

decode viene decodificata

execute vengono eseguite le operazioni che compongono l'istruzione

È composta da:

- *ALU (Arithmetic-Logic Unit)*
- registri:
 - PC** Program Counter
 - IR** Instruction Register
 - MAR** Memory Address Register
 - MDR** Memory Data Register
 - PSW** Processor Status Word (Flag)
 - Registri di lavoro**
- unità di controllo

- Lo stato della CPU è dato dall'insieme dei valori assunti dai registri: è possibile ripristinare un dato stato ponendo nei registri i valori relativi ad esso.

1. l'UC mette PC in MAR
2. l'UC mette il comando di lettura sul bus controllo
3. MDR viene messo in IR — incremento di PC

Esempio di programma in assembly

Effettuare la somma dei valori contenuti nelle celle 10 e 14, e porre il risultato nella cella 8.

- $R1 := M[10]$
- $R2 := M[14]$
- `add R1 R2`
- $M[8] := R1$

- memoria centrale
- memoria di massa

Si classifica per:

- velocità d'accesso
- capacità (byte)
- volatilità
- costo per bit

Diverse tecnologie hanno caratteristiche diverse.

- memoria interna alla CPU: registri, cache (a volte è esterna, ma non passa per il bus)
- memoria interna al calcolatore: memoria centrale
- memoria esterna al calcolatore: memoria di massa (dischi, nastri)

Cache: si basa sulle proprietà di località spaziale e temporale degli accessi.

- Interfaccia di I/O:
 - seriale (usb, bluetooth)
 - parallelo
 - registro dati
 - registro di controllo (di stato)
- periferiche *memory mapped*
- istruzioni dedicate

- Gestione I/O a controllo di programma
la CPU controlla direttamente lo stato della periferica (*polling*)
- Gestione I/O a interrupt
la periferica avvisa la CPU (*polling*)
- I/O tramite DMA
la CPU controlla il controllore DMA(*polling*)

Istruzioni particolari possono alterare il prelievo delle istruzioni da celle consecutive:

- istruzioni di salto
- istruzioni di chiamata a sotto-programmi
- istruzioni di interruzione