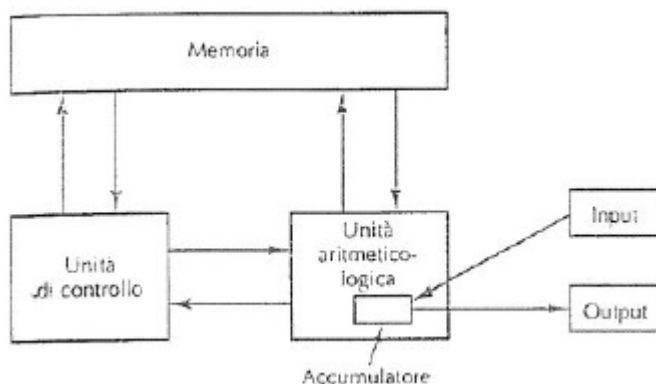


# 1 - Introduzione

## Von Neumann Machine

Von Neumann nel 1952 idealizza il modello concettuale di un calcolatore, la sua macchina, costruita da 5 componenti principali:

- Memoria
- Unità aritmetico-logica
- Unità di controllo
- Dispositivi di input
- Dispositivi di output



L'unità di controllo e aritmetico-logica formavano insieme il "cervello dei computer", nei computer moderni sono stati combinati in una **CPU** (Central Processing Unit, Unità Centrale di Calcolo).

All'interno dell'unità aritmetico logica era presente uno speciale registro da 40 bit, chiamato **accumulatore**, la tipica istruzione somitava una parola di memoria nell'accumulatore oppure ne copiava il contenuto in memoria, la macchina poi non presentava un'aritmetica in virgola mobile, dato che Von Neumann riteneva che ogni matematico che si rispetti dovesse essere in grado di tener traccia a mente la posizione della virgola.

## Livelli macchina

Una macchina lavora a 6 livelli:

- Una macchina originariamente lavora in livello L0, nel suo linguaggio nativo, quello che l'hardware può comprendere (accesso e spento) e formato da porte logiche, dispositivi digitali che prendono in input un valore e ne calcolano in output una semplice funzione dei valori d'ingresso, come le porte AND o OR.
- Subito dopo troviamo il suo livello di microarchitettura, livello L1, con una memoria locale formata da un gruppo di registri (da 8 a 32) e un circuito chiamato ALU (Arithmetic Logic Unit), capace di effettuare semplici operazioni aritmetiche. Tutti i registri sono connessi

all'ALU in modo da formare un percorso nel quale i dati si spostano, permettendo poi all'ALU di operare su di loro

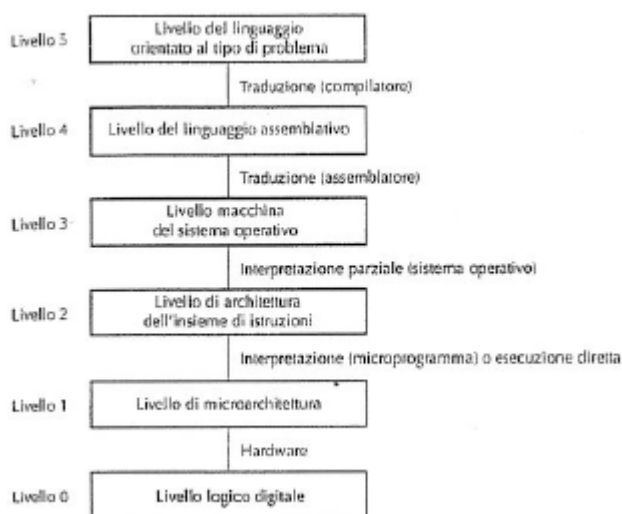
- Il livello L2 è costituito dall'ISA (Instruction Set Architecture Level), cioè le istruzioni che la macchina a livello di architettura
- Nel livello L3 troviamo il sistema operativo

Tra il livello L3 e il livello L4 c'è una spaccatura fondamentale, i 3 livelli inferiori non sono progettati per essere usati dal programmatore medio ma solo come interpreti e traduttori per i livelli alti, mentre i livelli dal 4 in su sono pensati per chi deve risolvere problemi applicativi;

Oltre questo, i livelli superiori sono supportati dalla traduzione, al contrario di quelli inferiori (L2 e L3) che sono interpreti

- Nel livello L4 troviamo il linguaggio assemblativo, una forma simbolica di linguaggi sottostanti, fornisce ai programmatori un modo per scrivere codice.
- Nel livello 5 troviamo i veri linguaggi, quelli **ad alto livello**, che si usano per la programmazione vera e propria (Python, C, Java)

Da livello L1 in poi tutte le interpretazioni e traduzioni sono presenti in una macchina virtuale



## Assembler

- In principio il linguaggio macchina è composta dai 0 ai 5V, trasformato in 0 e 1.
- Per facilitare la programmazione fu introdotto il linguaggio assembly
- L'assembly definisce stretta relazione (1:1) con i codici di linguaggio macchina, quindi tra il linguaggio istruzione e macchina. Il programma scritto in assembly (il suo traduttore) è convertito automaticamente in linguaggio macchina.

Più si è vicini alla macchina e più il linguaggio non diventa **immediato**;

Il rapporto diventa molto più complicato quando si sale di più livelli, **un** istruzione di linguaggio ad alto livello corrisponde a **più** istruzioni a basso livello

Il linguaggio assembly viene utilizzato principalmente quando l'esecuzione ha vincoli stringenti sul tempo e sulla sua architettura, viceversa si usano linguaggi più naturali,

linguaggi di **alto livello**.

I linguaggi di alto livello sono elementi intermedi di una varietà di linguaggi ai cui estremi si trovano il linguaggio macchina ed i linguaggi naturali

I linguaggi che non dipendono dall'architettura della macchina offrono tre vantaggi fondamentali:

- **Portabilità:** I programmatori possono astrarre dei dettagli architetturali di ogni calcolatore, si può spostare il codice su qualunque macchina e deve poter girare
- **Leggibilità:** I programmi devono essere ordinati, semplici da leggere e da modificare
- **Manutenibilità:** Il programma deve essere mantenibile nel tempo, poterlo modificare senza distruggerlo