

# Lezione 1 Prof.Arcieri

October 2025

## 1 Fondamenti di Informatica e Astrazioni Hardware

### 1.1 Scopo del Computer

Un computer ha come principale funzione la gestione e la manipolazione delle informazioni. In questo contesto, il termine “informazione” si riferisce a una quantità quantizzabile e ridotta, il cui elemento minimale è il **bit**. I bit possono essere manipolati attraverso operazioni logiche fondamentali, come AND, OR, NOT, XOR, NAND e NOR.

### 1.2 Evoluzione dei Microprocessori e Rappresentazione Digitale

Il progresso dei microprocessori ha reso più concreto il passaggio dall’analogico al digitale: il primo passo di astrazione consiste nell’interpretare il potenziale elettrico come informazione codificata in forma minimale, ad esempio associando il livello alto (**True**, tipicamente 5V) e il livello basso (**False**, tipicamente 0V) a uno stato binario.

#### 1.2.1 Overclocking e Sicurezza

L’overclocking consiste nell’aumentare deliberatamente la frequenza di clock del processore, rendendo dunque più rapida l’elaborazione delle istruzioni. Tuttavia, modifiche forzate delle tensioni possono anche essere usate per tentativi di manipolazione o attacco hardware, con possibili effetti non desiderati sul comportamento logico della CPU, che basa la sua azione sulla distinzione tra vero e falso.

### 1.3 Logica dei Transistor e Operatori

Un computer moderno è composto da milioni di transistor, che permettono di associare lo stato true” o false” al segnale elettrico e quindi rappresentare e processare informazioni. Gli operatori logici sono nativi, mentre l’operatore di somma (ADD) non lo è – infatti, mentre  $T \oplus T = T$ , la somma numerica non

segue una regola logica diretta ( $5 + 4 = 9$  non è una regola booleana). Per implementare la somma, ad esempio, il circuito dell'“adder” utilizza le porte XOR e AND, costituendo uno dei primi algoritmi hardware (non primitivo).

## **1.4 Gestione delle Informazioni e Interfacce**

La CPU gestisce informazioni non solo per l'elaborazione interna, ma anche per interagire con il mondo esterno. Storicamente, questa interazione avveniva principalmente tramite dispositivi come la stampante. In generale, la CPU governa l'ingresso (input) e l'uscita (output) dati.

## **1.5 Astrazione e Codifica**

Un ulteriore step di astrazione consiste nella codifica: ad esempio, un testo viene tradotto in una sequenza di bit prima di essere stampato su un foglio.

Esempio: Testo  $\rightarrow$  codice bit  $\rightarrow$  stampa

## **1.6 Cenno Storico su Sistemi Operativi**

IBM ha introdotto innovazioni come il file system dinamico e la virtual machine. I sistemi IBM360, IBM370 e IBM390 sono stati storicamente fondamentali. L'IBM390 era celebre perché includeva una macchina virtuale IBM360, rendendolo compatibile con il mercato sviluppatosi attorno al 360.

## **1.7 Archiviazione e Codifica Standard in RAM**

Le informazioni vengono estratte e archiviate nella RAM, che le codifica secondo lo standard ASCII (American Standard Code for Information Interchange). Gli ASCII da 0 a 31 rappresentano codici di controllo usati storicamente per la gestione delle stampanti e la comunicazione (ad esempio segnali RTS/OTR), e costituiscono la base dei protocolli di comunicazione: un altro importante passaggio di astrazione.

## **1.8 Posizionamento dei Bit e Sistemi di Numerazione**

Mentre normalmente leggiamo da sinistra verso destra, i sistemi binari convenzionali leggono i bit da destra verso sinistra. Questo ragionamento si connette allo studio delle diverse basi numeriche, tra cui la base 24: popoli antichi come Babilonesi, Sumeri e Pitagora contribuirono all'evoluzione delle basi numeriche nella storia matematica.

## **1.9 Astrazione nella Codifica: Caratteri e Numeri**

La codifica dei caratteri e dei numeri si basa su specifici pattern di bit. Quando si elaborano numeri, occorre trattare la rappresentazione dei valori negativi e dei numeri con virgola. Le principali tecniche sono:

- Bit di segno (sign bit)
- Complemento a 2 (two's complement)
- Virgola fissa e mobile (fixed/floating point)

### 1.10 Nibble, Segno, Complemento

Un **nibble** è un gruppo di 4 bit, metà di un byte. Nella rappresentazione numerica, la distinzione tra valore, segno e complemento è fondamentale per gestire correttamente interi e numeri con virgola.

### 1.11 Riepilogo e Collegamento

Tutti questi passaggi costituiscono un percorso di astrazione progressiva: dagli impulsi elettrici elementari agli operatori logici, fino alle complesse codifiche di caratteri e numeri, passando per standard di codifica e sistemi di numerazione che riflettono sia l'evoluzione tecnologica che matematica della storia dei calcolatori.