



**UNIVERSITÀ
DEGLI STUDI
DI TRIESTE**

Rappresentazione dell'informazione

Prof.ssa Giulia Cisotto

giulia.cisotto@units.it

Trieste, 5 marzo 2025

AGENDA DI OGGI

1. *Introduzione all'informatica*
2. *Programma e metodologie del corso*

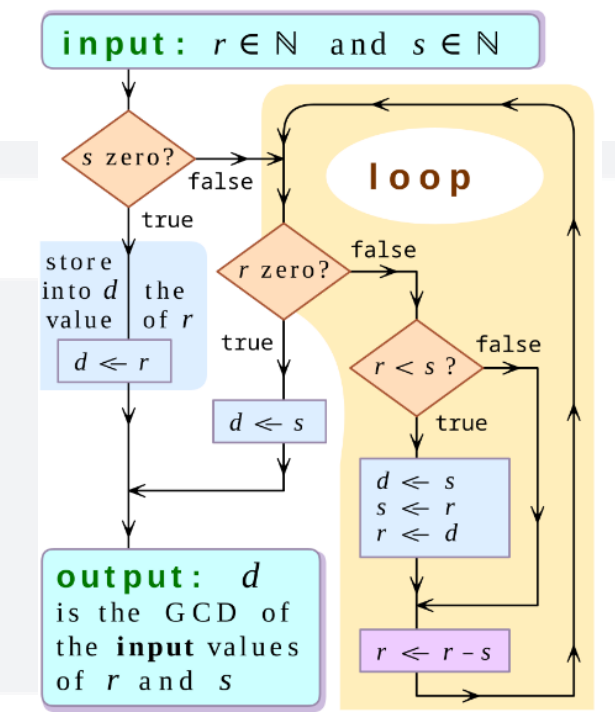
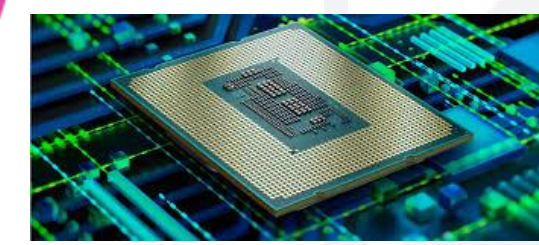
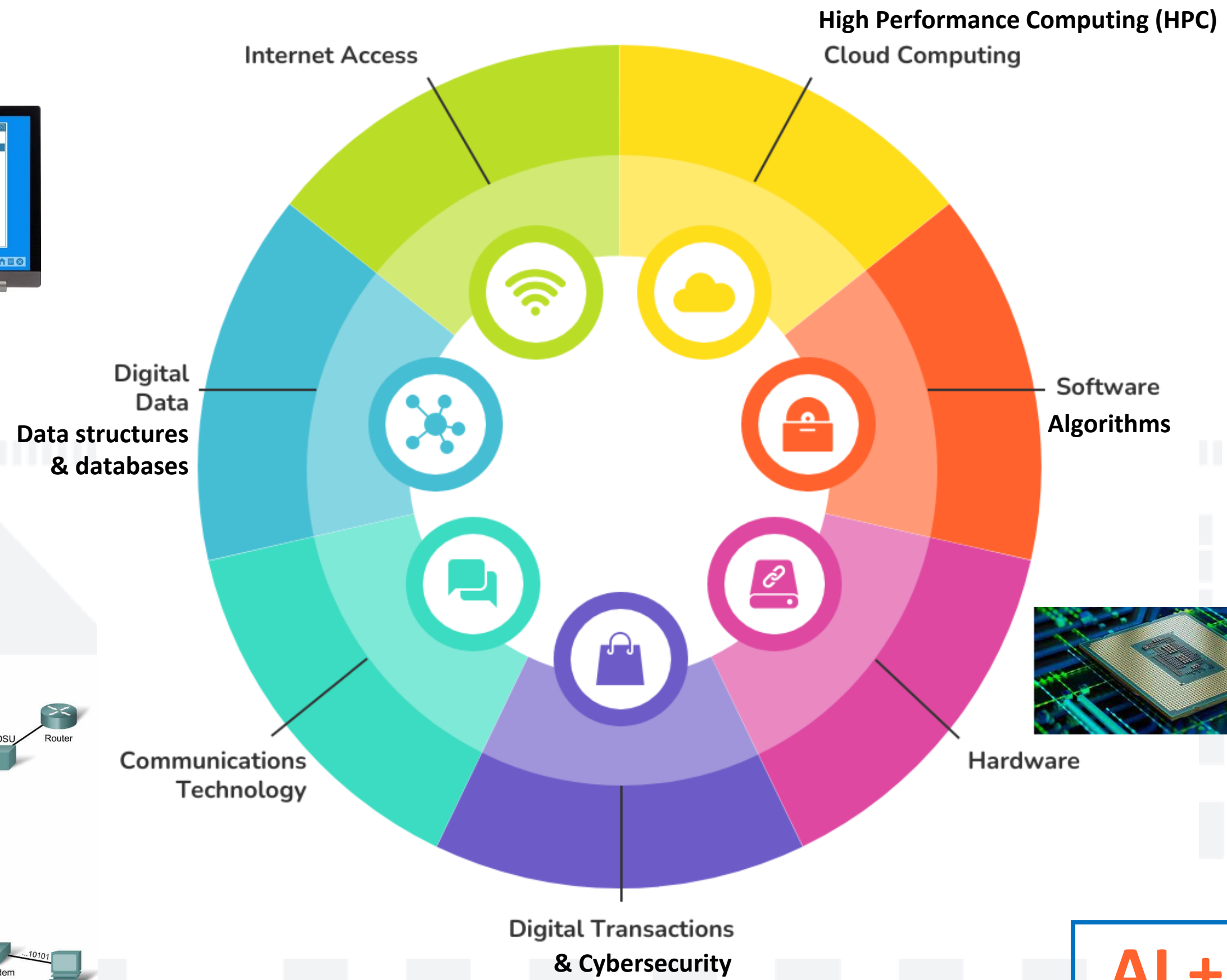
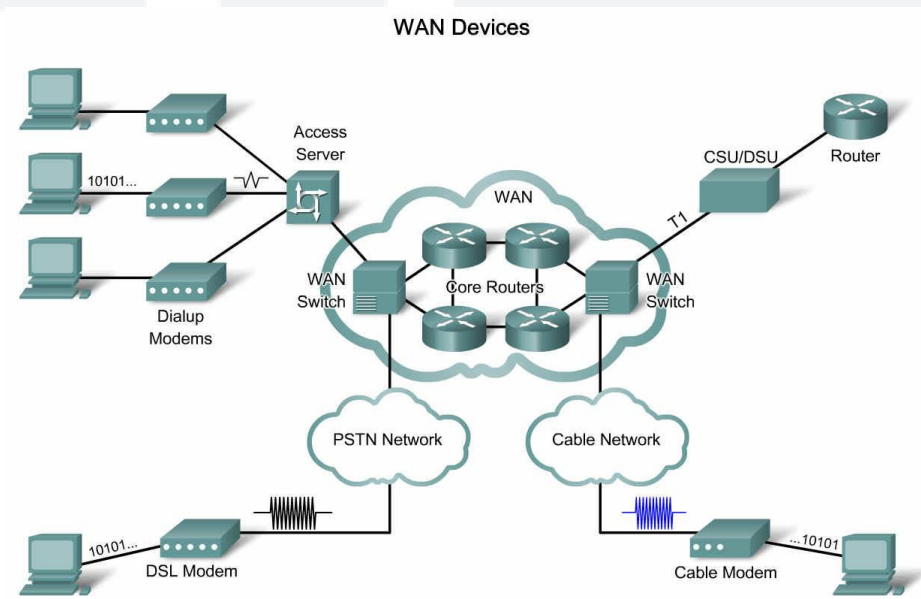
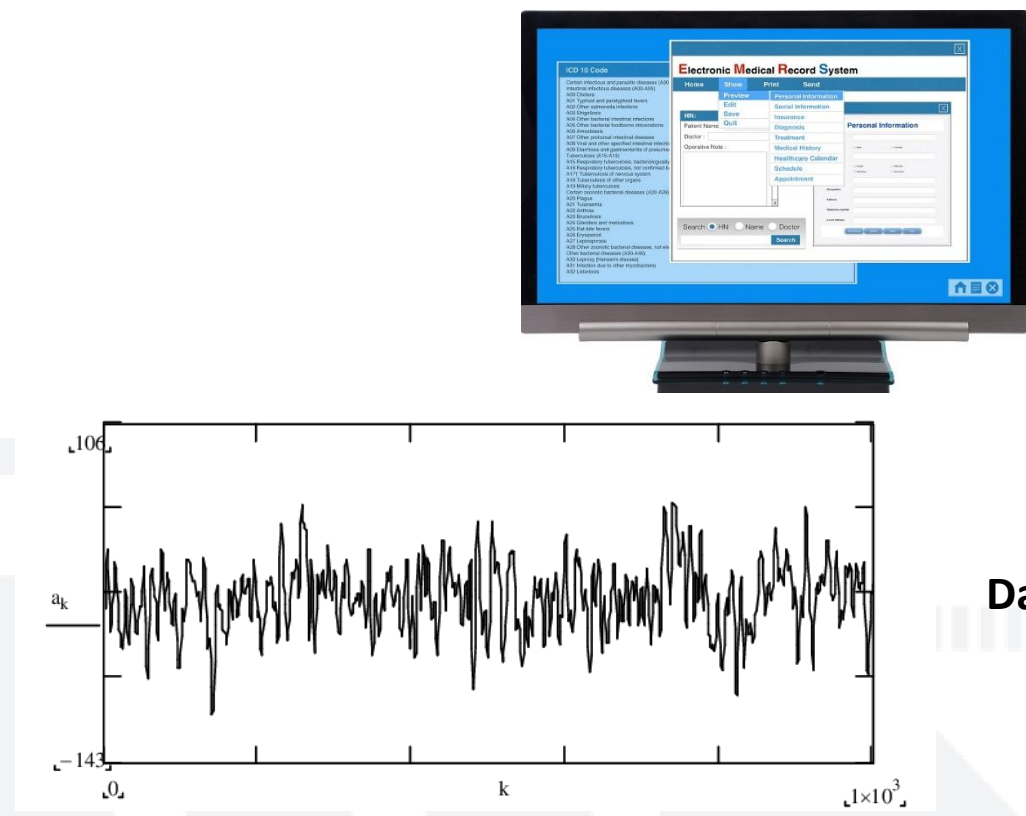
3. *Questionario di inizio corso*



4. **Inizio MODULO 1**



INFORMATICA



AI + robotica



INFORMATICA: MODULO 1 «Architettura»

HARDWARE: microprocessore, RAM, scheda video, tastiera, mouse, monitor, ecc.

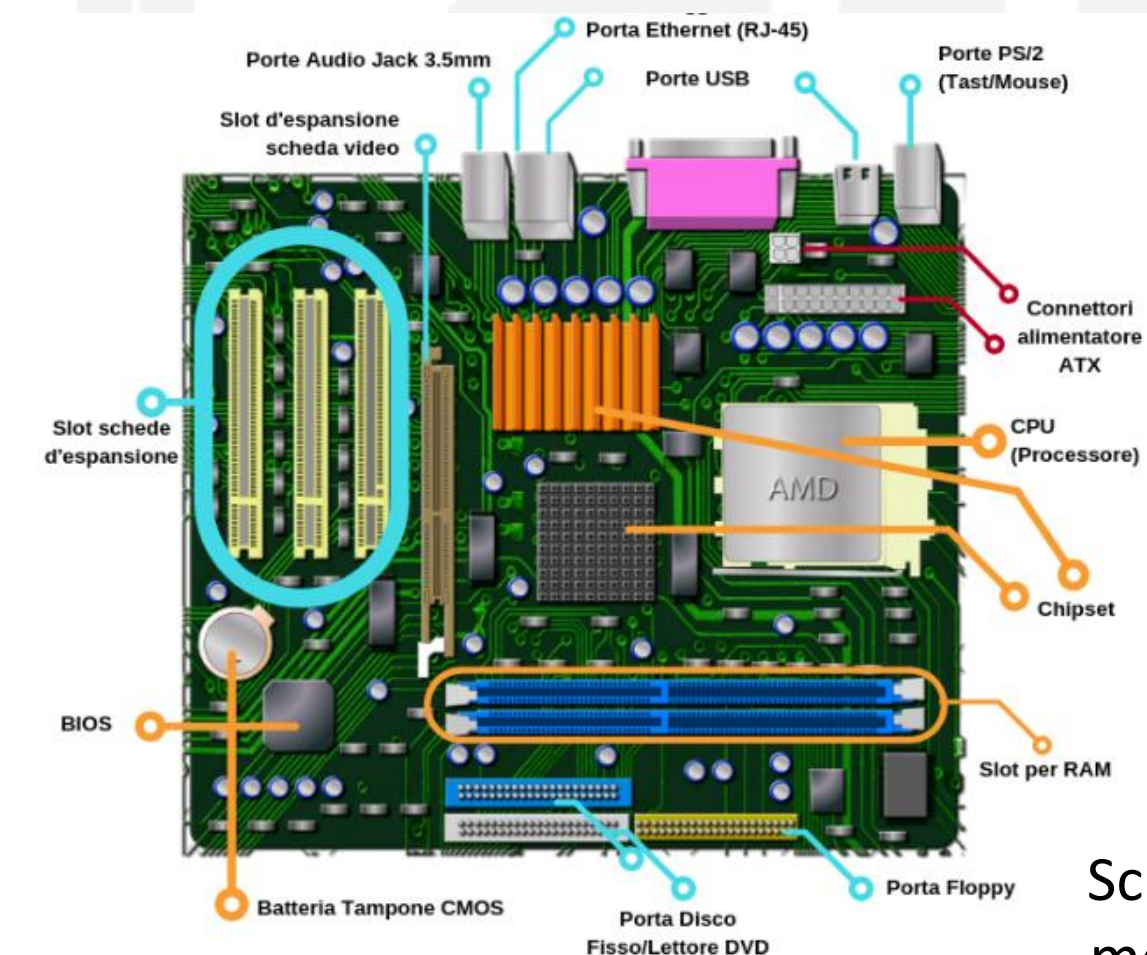
SOFTWARE: sistemi operativi (windows, Linux, Android, IOS), software di videoscrittura o elaborazione testi (Microsoft Office Word, LibreOffice Wiriter, google documenti), fogli elettronici o di calcolo (Excel, Calc, google fogli), browser web (Chrome, Firefox, Safari o Edge).

INFORMATICA: MODULO 1 «Architettura»

HARDWARE: microprocessore, RAM, scheda video, tastiera, mouse, monitor, ecc.

SOFTWARE: sistemi operativi (windows, Linux, Android, IOS), software di videoscrittura o elaborazione testi (Microsoft Office Word, LibreOffice Writer, google documenti), fogli elettronici o di calcolo (Excel, Calc, google fogli), browser web (Chrome, Firefox, Safari o Edge).

ARCHITETTURA: organizzazione «logica» delle sue componenti interne e le modalità secondo le quali queste cooperano tra di loro per compiere azioni più o meno complesse.



Scheda madre di un moderno computer

Legenda dei colori

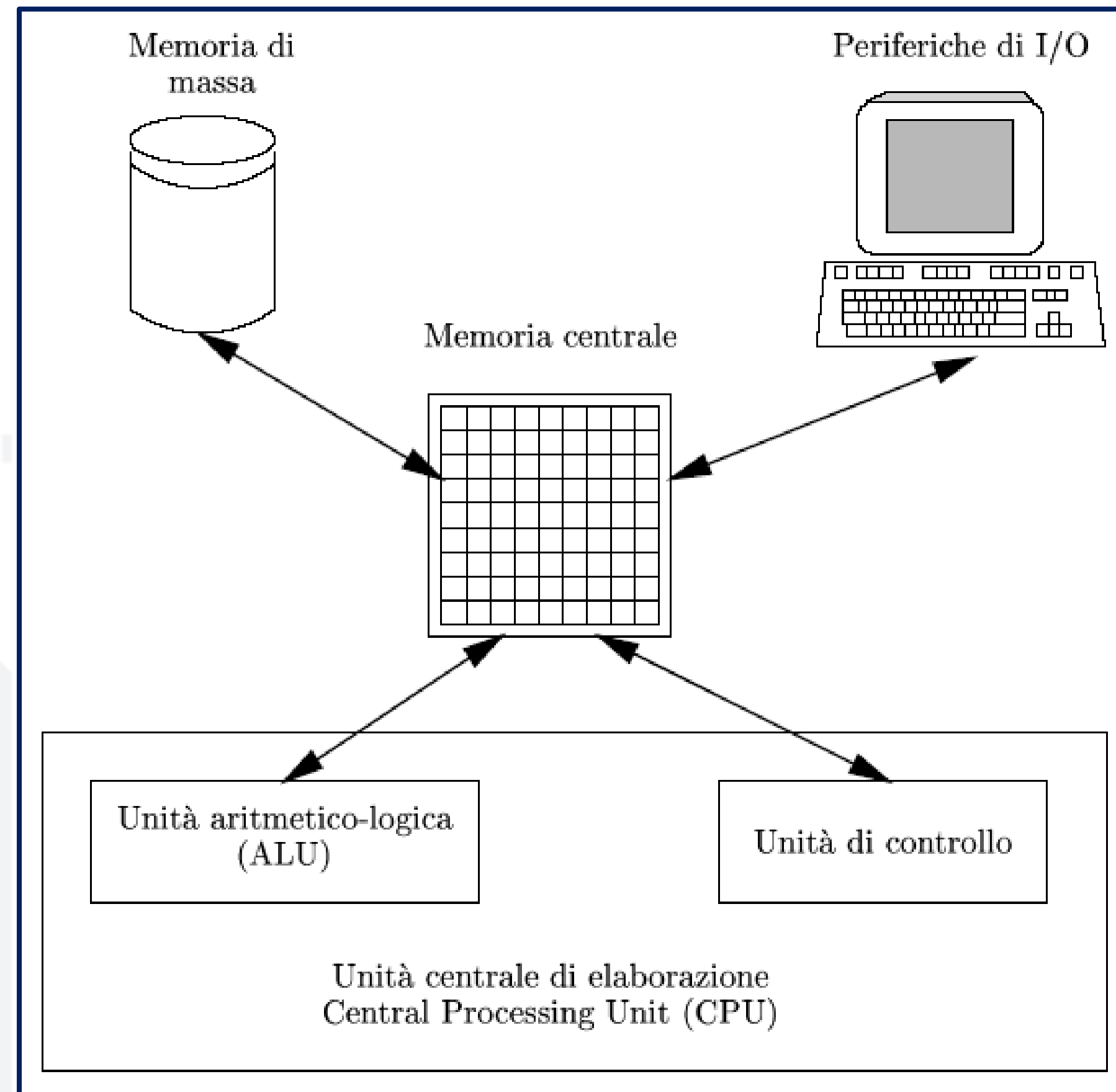
Componenti Slot/porte d'espansione Cavi d'alimentazione

MODULO 1: Architettura degli elaboratori

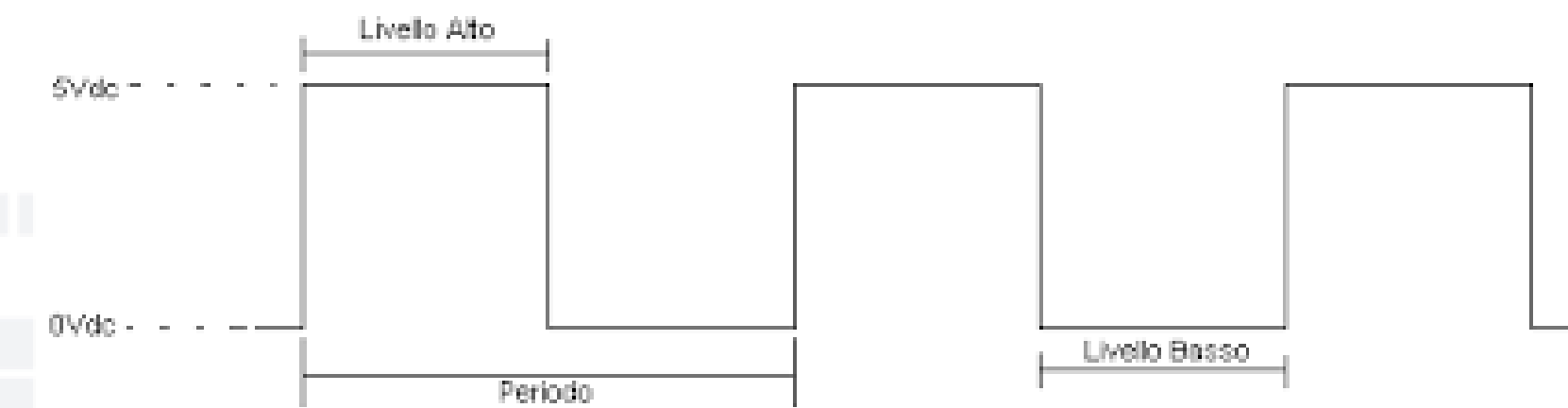
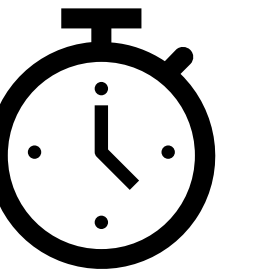


John Von Neumann

Matematico ed informatico di origine ungherese che viveva e lavorava negli Stati Uniti negli anni '40



Le varie parti dell'architettura devono sincronizzare le proprie attività: serve un «**CLOCK**»



E' un segnale periodico.

Più è veloce, più attività si possono fare nell'unità di tempo!

$$1\text{GHz} = 10^9 \text{ Hz}$$

Architettura di un elaboratore di Von Neumann

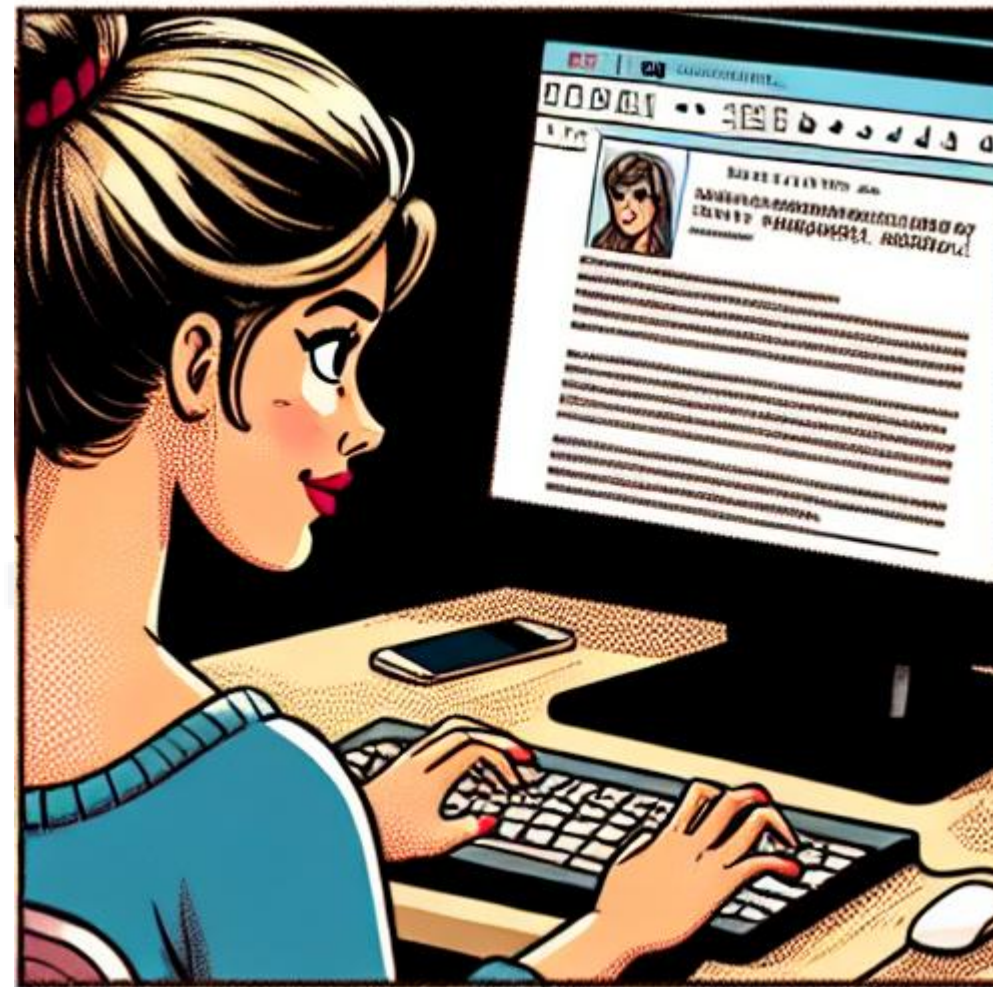
In generale, abbiamo bisogno di COMUNICARE CON IL PC per fargli fare dei task.

Come facciamo a dare INPUT al computer?

Cos'è un «INPUT»?



Linguaggio naturale,
input nel PC



Rappresentazione
dell'informazione in linguaggio
macchina, flusso dati nel PC



Output dal PC, linguaggio
naturale

«INPUT» possono essere testi scritti, suoni, immagini, ...

3 GRANDI DOMANDE

1. Come posso **rappresentare l'informazione** nel pc?

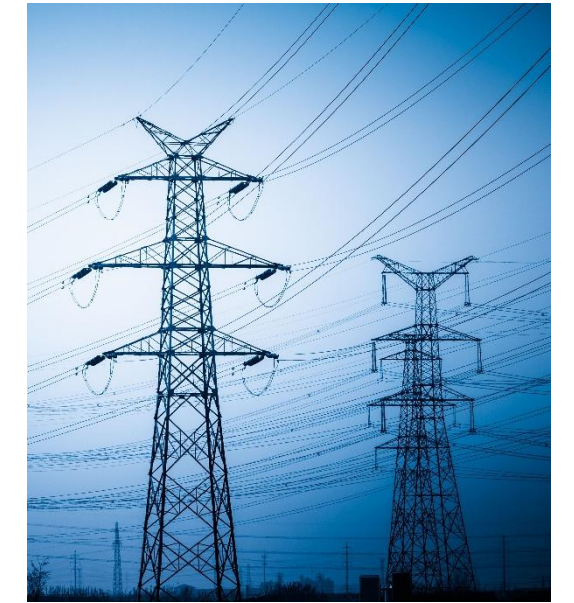
2. Come posso realizzare fisicamente tale rappresentazione?

3. Come posso costruire il pc a partire dagli elementi di base?

Computer e transistor: dai 240 V a sequenze di 0 e 1

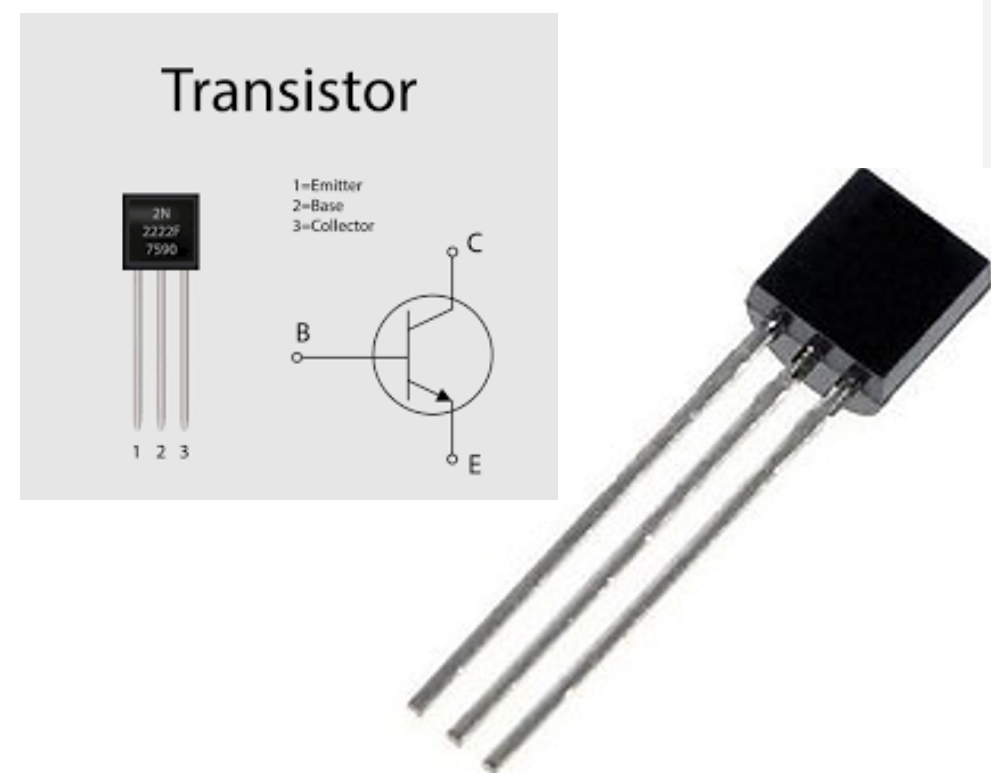


Linfa vitale: corrente di alimentazione che scorre nei circuiti.



Il trasformatore del PC converte la corrente alternata (AC) a **240V** in tensioni più basse e stabili, come **12V**, **5V** e **3.3V** necessarie per il funzionamento del computer.

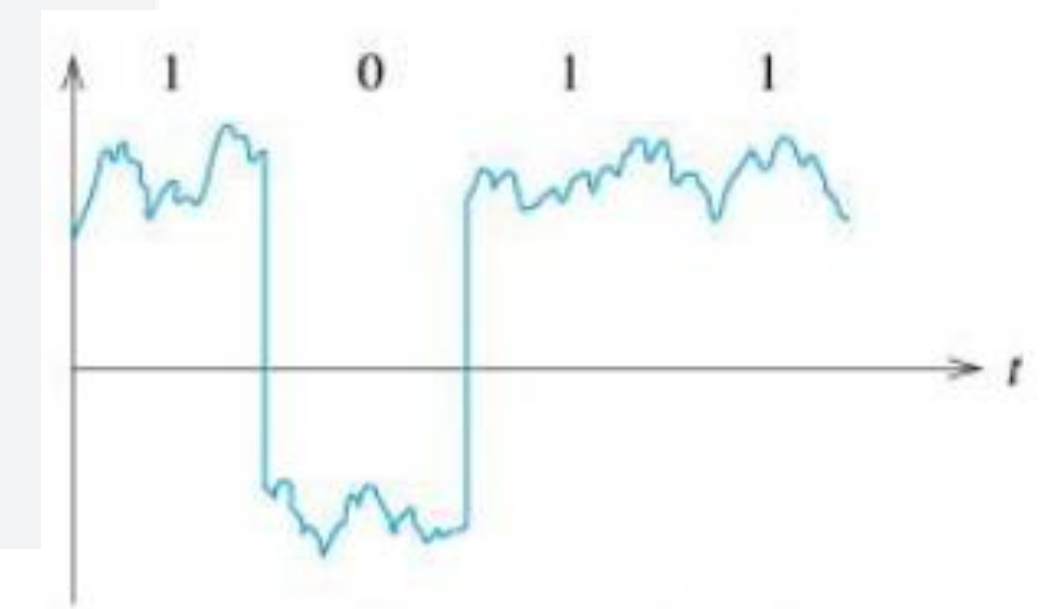
L'alimentatore fornisce i **12V** per la CPU e la GPU, i **5V** per le porte USB e i dischi, e i **3.3V** per la scheda madre e altri componenti elettronici.



Transistor e altra «logica» (circuiteria) permette di trasformare il segnale continuo che oscilla tra livello basso e alto in un'onda quadra:

Se la tensione è **alta** (es. **5V** o **12V**) → viene interpretata come **1**.

Se la tensione è **bassa** (es. **0V**) → viene interpretata come **0**.

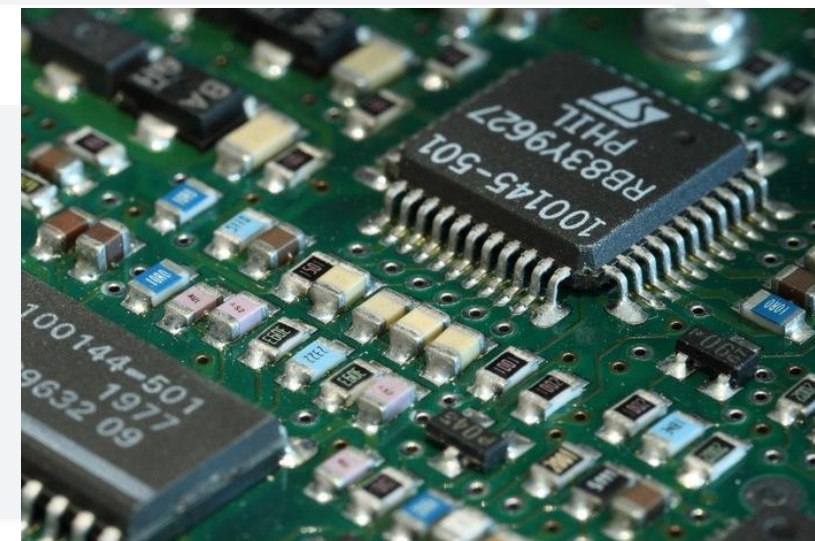


Computer e transistor: dalla sua invenzione al computer quantistico, oltre (?) la legge di Moore

Il primo prototipo funzionante fu realizzato nel mese di dicembre del 1947 da due ricercatori dei **laboratori Bell Labs**: Walter Brattain e John Bardeen del gruppo di ricerca guidato da William Shockley.



Il transistor è alla base dei microprocessori, della RAM e dei circuiti integrati in generale: ogni microprocessore contiene tantissimi transistor che ne permettono il funzionamento e **maggiore è il numero di transistor maggiore sarà la velocità del microprocessore stesso e quindi del computer.**



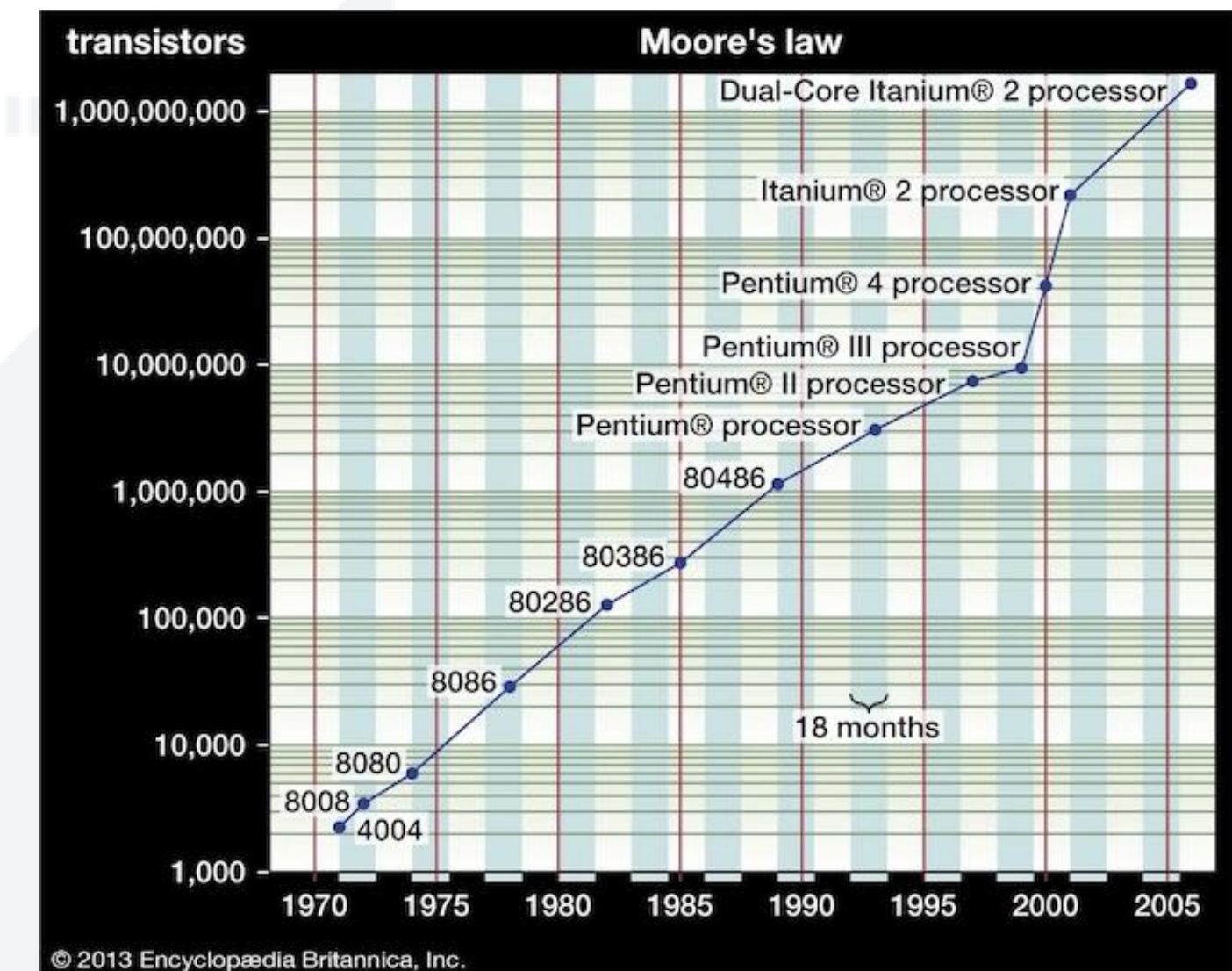
La Legge di Moore (1965):

«La complessità di un microcircuito, misurata ad esempio tramite il numero di transistor per chip, raddoppia ogni 18 mesi*.» [Wiki](#)

**24 mesi (1975)*



1929-2023,
[Wiki](#)



C'è un limite alla legge di Moore? Sì, ed è un limite fisico.

La **miniaturizzazione** dei transistor comporta la riduzione della loro dimensione e della tensione di alimentazione, ma entrambi questi fattori possono causare **problemi di calore e interferenza elettromagnetica**. In secondo luogo, la miniaturizzazione dei transistor richiede una **precisione estremamente elevata** nella produzione dei chip.

Molte innovazioni tecnologiche che possono migliorare le prestazioni dei chip: **materiali alternativi** ai semiconduttori tradizionali (al posto del silicio si potrebbe usare il **grafene**, ad esempio), **uso di architetture di chip innovative, computer quantistici.**



[Link articolo](#)

IL SOLO **MOORE**
TECNOLOGIA

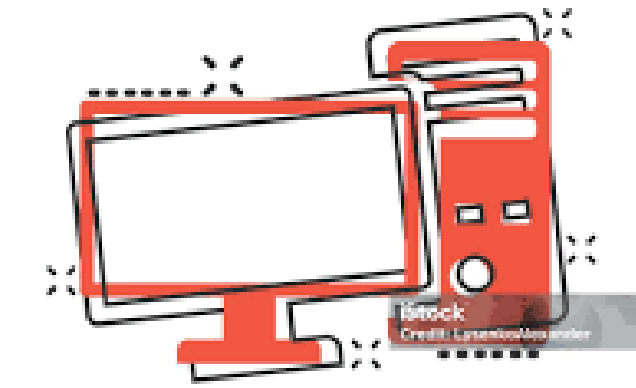


01110011011110110010101000100011110011001100000010010001001110
11101011100100000110111101010001100110010010000001100011000011
11101000101110110111111111011000110000001111001110010010110000
01111110111001101010111010000010011010011100011000100111011101
10101011100100011111111101100001111000101010001111111001101000
00111011010010111011000011100100000110111010111001110000001000
00111110001011100001111011011111111000111100001101010011111100
110110

SISTEMI NUMERICI



SISTEMA NUMERICO DECIMALE	SISTEMA NUMERICO BINARIO	SISTEMA NUMERICO ESADECIMALE
0	0	0
1	1	1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
		A
		B
		C
		D
		E
		F



BIT E CONFIGURAZIONI

Con il termine **bit** definiamo l'**unità di misura dell'informazione**. Un bit può assumere solo il valore di 0 o 1.

Combinando tra loro più bit si ottengono strutture più complesse (sequenze).

In particolare:

- **byte, 8 bit**
- nybble, 4 bit
- **word, 32 bit**
- **halfword, 16 bit**
- doubleword, 64 bit

BIT E CONFIGURAZIONI

Quante *configurazioni* diverse può assumere una sequenza di due bit?

Chiamiamo **a** il primo bit e **b** il secondo bit e scriviamo tutte le loro possibili combinazioni:

a	b
0	0
0	1
1	0
1	1

In totale ci sono 4 configurazioni.

In generale, **dati k bit, il numero di configurazioni ottenibili è 2^k .**

BIT E CONFIGURAZIONI

Quante *configurazioni* diverse può assumere una sequenza di due bit?

Chiamiamo **a** il primo bit e **b** il secondo bit e scriviamo tutte le loro possibili combinazioni:

a	b
0	0
0	1
1	0
1	1

In totale ci sono 4 configurazioni.

In generale, **dati k bit, il numero di configurazioni ottenibili è 2^k .**

ENTITÀ E RAPPRESENTAZIONI

Entità o *valore* è lo stesso modo per intendere il valore numerico

Una *rappresentazione* è un modo per descrivere un'entità.

Esempio di rappresentazione dell'entità «sedici»

- nel sistema decimale, la rappresentazione è 16_{10}
- nel sistema binario, la rappresentazione è 10000_2
- nel sistema esadecimale (in base sedici), la rappresentazione è 10_{16}

Tutte queste rappresentazioni sono alternative ma **equivalenti per rappresentare in modo inequivocabile lo stesso valore o entità.**

RAPPRESENTAZIONI POSIZIONALI

Tutte queste rappresentazioni sono di tipo posizionale, ovvero **ogni cifra assume un valore diverso a seconda della posizione** in cui si trova rispetto alle altre cifre.

Esempio : nel sistema numerico decimale, scrivere **12** non è equivalente a scrivere **21**.

Per convenzione, le cifre che rappresentano valori più piccoli sono a destra, mentre quelle che rappresentano valori più grandi sono a sinistra.

Esempio: nel sistema numerico decimale, la cifra più a destra rappresenta le **unità**, la cifra subito alla sua sinistra le **decine**, quella alla sua sinistra le **centinaia** e così via..

Quindi il numero **21** è la somma di **1 unità e 2 decine**, ovvero: $1 + 2 * 10 = 1 + 20 = 21$.

RAPPRESENTAZIONI POSIZIONALI

Esempi nel sistema numerico binario ed esadecimale:

- $21_{16} = 1 + 2 \cdot 16 = 33_{10}$
- $10_{16} = 0 + 1 \cdot 16 = 16_{10}$
- $10_2 = 0 + 1 \cdot 2 = 2_{10}$
- $C2A5_{16} = 5 + A \cdot 16 + 2 \cdot 16 \cdot 16 + C \cdot 16 \cdot 16 \cdot 16 = 5 + 10 \cdot 16 + 2 \cdot 64 + 12 \cdot 4096 = 49829_{10}$

Fate un po' di prove voi, poi il calcolo manuale può essere confermato o corretto usando [questo strumento online](#).

SISTEMA NUMERICO DECIMALE	SISTEMA NUMERICO BINARIO	SISTEMA NUMERICO ESADECIMALE
0	0	0
1	1	1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
		A
		B
		C
		D
		E
		F

RAPPRESENTAZIONI *NON* POSIZIONALI

Vi vengono in mente sistemi numerici non posizionali??

il sistema numerico romano

Non esistono posizioni per le unità, decine, centinaia. Per ognuno di questi valori c'è una «cifra» di questo sistema.

Esempi:

- **X** si legge sempre 10, indipendentemente dalla sua posizione
- il numero **IV** è 4. **I** non è una decina, né **V** sono le unità, altrimenti si leggerebbe 15.

RAPPRESENTAZIONE: REGOLA GENERALE

In generale, nei sistemi posizionali il valore numerico \mathbf{N} è caratterizzato dalla seguente rappresentazione:

$$\mathbf{N} = [d_{n-1}, d_{n-2}, \dots, d_1, d_0, d_{-1}, \dots, d_{-m}]$$

$$\mathbf{N} = d_{n-1} \cdot r^{n-1} + d_{n-2} \cdot r^{n-2} + \dots + d_1 \cdot r^1 + d_0 \cdot r^0 + d_{-1} \cdot r^{-1} + \dots + d_{-m} \cdot r^{-m} = \sum_{i=-m}^{n-1} d_i \cdot r^i$$

dove:

- \mathbf{d} rappresenta la singola cifra (digit)
- \mathbf{r} è la radice o base del sistema
- \mathbf{n} è il numero di cifre della parte intera (sinistra della virgola)
- \mathbf{m} è il numero di cifre della parte frazionaria (destra della virgola)

ESEMPIO (SIST. DECIMALE)

Esempio nel **sistema numerico decimale**:

- Base $r = 10$
- Cifre $\mathbf{d} = 0, 1, \dots, 9$

$$N = 243,75$$

$$N = \sum_{i=-m}^{n-1} d_i \cdot r^i$$

$$N = d_{n-1} \cdot 10^{n-1} + d_{n-2} \cdot 10^{n-2} + \dots + d_1 \cdot 10 + d_0 \cdot 1 + d_{-1} \cdot 10^{-1} + \dots + d_{-m} \cdot 10^{-m} =$$

$$= 2 \cdot 10^2 + 4 \cdot 10 + 3 \cdot 1 + 7 \cdot 10^{-1} + 5 \cdot 10^{-2}$$

CONVERSIONE DA BINARIO A DECIMALE

Esempio nel **sistema numerico binario**:

- Base $r = 2$
- Cifre $\mathbf{d} = 0, 1$

$$N = 1001,11$$

$$N = \sum_{i=-m}^{n-1} d_i \cdot r^i$$

$$N = d_{n-1} \cdot 2^{n-1} + d_{n-2} \cdot 2^{n-2} + \dots + d_1 \cdot 2 + d_0 \cdot 1 + d_{-1} \cdot 2^{-1} + \dots + d_{-m} \cdot 2^{-m} =$$

$$= 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2 + 1 \cdot 1 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2} = 8 + 1 + 0,5 + 0,25 = (9,75)_{10}$$

Fate qualche prova
voi e controllate il
vostro risultato [qui](#)

CONVERSIONE DA ESADECIMALE A DECIMALE

Esempio nel **sistema numerico esadecimale**:

- Base $r = 16$
- Cifre $\mathbf{d} = 0, 1, \dots, 9, A, B, C, D, E, F$

Provate a convertire questo numero in decimale:

$$\mathbf{N} = (4C, 1)_{16}$$

$$\mathbf{N} = (C2A5)_{16} = 0xC2A5 \qquad \mathbf{N} = \sum_{i=-m}^{n-1} d_i \cdot r^i$$

$$\mathbf{N} = d_{n-1} \cdot 16^{n-1} + d_{n-2} \cdot 16^{n-2} + \dots + d_1 \cdot 16 + d_0 \cdot 1 + d_{-1} \cdot 16^{-1} + \dots + d_{-m} \cdot 16^{-m} =$$

$$= C \cdot 16^3 + 2 \cdot 16^2 + A \cdot 16 + 5 \cdot 1 = 12 \cdot 16^3 + 2 \cdot 16^2 + 10 \cdot 16 + 5 = 49152 + 512 + 160 + 5 = (49829)_{10}$$

Materiale per la lezione

- Appendice A di Tanenbaum, "Structured computer Organization", 5th ed.
- Link nelle slide

Prossima lezione: 6 marzo, h.9:00, aula 4C