

# Corso di Fondamenti di Informatica



## Lezione 6

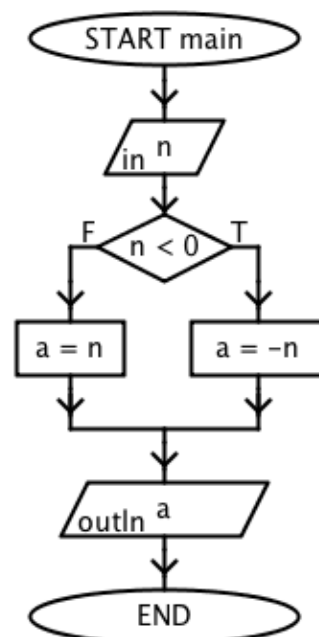
**Nicola Capuano**

Dipartimento di Scienze Aziendali, Management  
& Innovation Systems

[ncapuano@unisa.it](mailto:ncapuano@unisa.it)

### Esercizio per casa (soluzione)

Descrivere, mediante  
diagramma di flusso, un  
algoritmo che calcola e  
visualizza il **valore assoluto** di  
un numero letto in input



# Programma del Corso

## Modulo 1 - Tecnologie dell'informazione e della comunicazione

- Introduzione alle ICT
- Rappresentazione Digitale dell'Informazione
- Rappresentazione Digitale dei Dati Multimediali
- Architettura Hardware di un Computer
- Software e Sistemi Operativi
- Reti di computer

## Architettura Hardware di un Computer

### Parte 1: CPU e Memoria Centrale

#### Bibliografia

- Par. 2.10 e 2.11: l'Unità centrale di elaborazione
- Par. 2.12: Le memorie RAM e ROM
- Par. 2.13: Altri tipi di memoria
- Par. 2.14 e 2.15: I bus
- Par 2.16: Comunicazione con le periferiche
- Par 2.18: Interazione tra CPU e Memoria
- Approfondimenti su queste slide

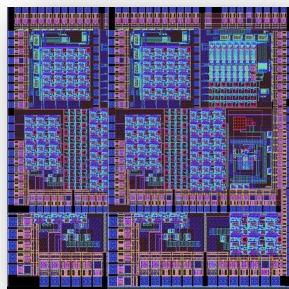


# Unità Centrale di Elaborazione

La **CPU**, o **Unità Centrale di Elaborazione** o **Processore** o **Microprocessore**, è un circuito integrato (**chip**), costituito da una piccola piastra di silicio

Una CPU contiene:

- Miliardi di **componenti elettronici miniaturizzati** detti **transistor**
- **Collegamenti** e altri componenti elettronici



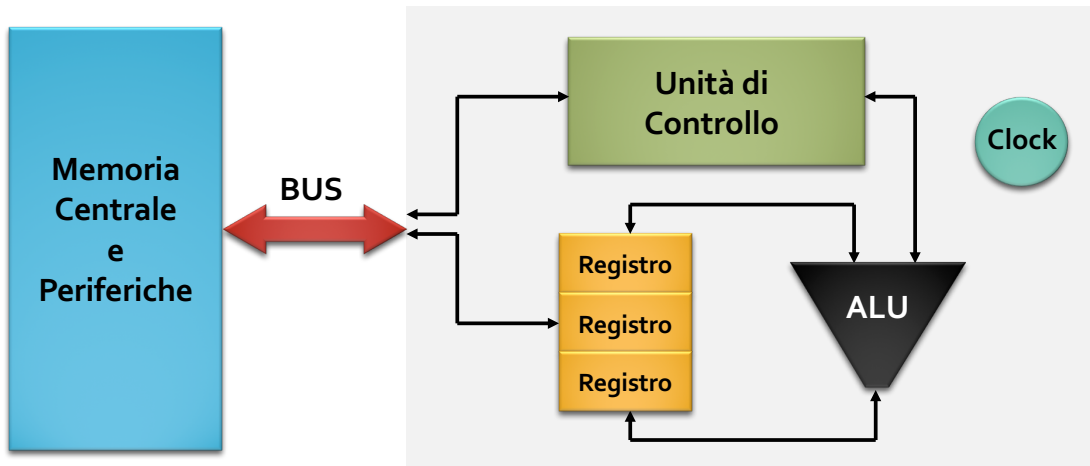
# Transistor

I **transistor** operano in maniera logicamente semplice come degli **interruttori**

- Un transistor ha **tre contatti**: **gate**, **source** e **drain**
- Se è **applicata un tensione al gate** la corrente scorre tra **source** e **drain**, altrimenti no
- **Combinando in maniera lineare** questo tipo di funzionamento è possibile costruire **circuiti complessi** e **interi processori**
- Ne sono necessari **miliardi** per raggiungere una logica complessa in grado di fornire **le prestazioni** e **le funzionalità dei processori moderni**



# Architettura della CPU



Basata sul modello astratto definito da **Von Neumann** nel 1945

# Componenti di una CPU

## Unità di controllo (Control Unit)

- Sovrintende alla **corretta esecuzione dei programmi**
- **coordina il lavoro delle altre componenti hardware** e le attività dell'intero computer



## Unità Aritmetico/Logica (ALU)

- **Esegue le istruzioni** in linguaggio macchina
- Accede unicamente ai dati ed alle istruzioni **presenti nei registri**



# Componenti di una CPU

## Registri

- memorie **molto veloci** a contatto con la CPU
- Sono, in tutto, **qualche decina**

La CPU **non può lavorare direttamente con e informazioni della memoria centrale** ma solo **su dati e istruzioni presenti nei registri**

- ogni volta che la CPU deve eseguire un'istruzione su uno o più dati, deve copiare nei registri
  - **l'istruzione da eseguire**
  - **i dati su cui l'istruzione opera**

Registro

Registro

Registro

# Componenti di una CPU

## Clock

- Emette con regolarità un numero determinato di **impulsi al secondo**
- È utilizzato per **sincronizzare il funzionamento** degli altri componenti

La **velocità del clock** è il numero di impulsi emessi al secondo (**cicli**)

- **Si misura in Hertz:**
  - **1 hertz (Hz)** = 1 ciclo al secondo
  - **1 megahertz (MHz)** = 1 milione di cicli al secondo
  - **1 gigahertz (GHz)** = 1 miliardo di cicli al secondo

Clock

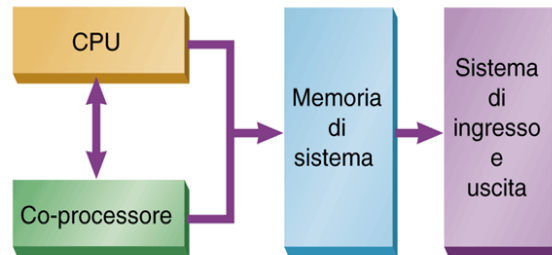
# Coprocessori

Svolgono **elaborazioni specialistiche** che richiederebbero alla CPU un notevole dispendio di tempo

- Operano sotto il diretto controllo della CPU

## Esempi

- **Coprocessore Matematico**, specializzato nell'esecuzione di **calcoli matematici**
- **Graphics Processing Unit (GPU)**, specializzato nella generazione e visualizzazione di **immagini grafiche**
- **Controller di periferica**, è preposto alla gestione di una o più **periferiche di input/output**



# Processori Paralleli

Processori che **collaborano paritariamente** all'esecuzione delle elaborazioni

- **Multi-CPU** sono computer dotati di scheda madre in grado di contenere fisicamente più processori
- **Multi-Core** sono CPU composte da più nuclei di processori fisici detti **core** montati sullo stesso chip

## I processori paralleli:

- possono eseguire lo **stesso programma** contemporaneamente su **parti diverse dei dati in input**
- possono eseguire contemporaneamente **programmi differenti**
- hanno lo scopo di **migliorare le prestazioni del computer**

# Processori incorporati

I microprocessori **non sono utilizzati solo nei computer** ma si trovano anche in:

- automobili, telefoni, termostati, schede telefoniche, carte di credito, ecc.

## I processori incorporati (**embedded**)

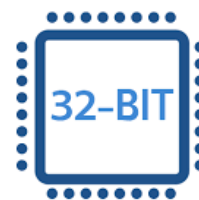
- Sono progettati appositamente per una **determinata applicazione** (special purpose)
- Non sono riprogrammabili dall'utente per altri scopi
- Sono **integrati** nel sistema che controllano



# Capacità di una CPU

Maggiore è la **quantità di dati** che un computer elabora **simultaneamente** e maggiori sono la sua **velocità** e la sua **potenza**

- La **dimensione della parola (word)** indica il numero di bit che un processore tratta **simultaneamente**
- Essa coincide con:
  - La **dimensione dei registri** interni della CPU
  - La **dimensione dei bus di sistema**
- Nei primi computer era di **1 byte**
- Attualmente varia da **32 bit** a **128 bit**



# Insieme di Istruzioni

Quando un computer esegue un programma la CPU esegue sequenze di istruzioni in **linguaggio macchina**

L'**insieme delle istruzioni** di un processore è **piuttosto limitato** ed include tipicamente:

- trasferimento di dati (MOV, ecc.)
- operazioni aritmetiche (ADD, SUB, MUL, DIV, ecc.)
- operazioni logiche (AND, OR, NOT, ecc.)
- confronto tra numeri (CMP, ecc.)
- manipolazione dei bit (SHR, SHL, ecc.)
- salti (JPM, ecc.)

**Esempio: somma di due interi**

```
MOV AH, 05  
MOV AL, 04  
ADD AH, AL
```

# Insieme di Istruzioni

Ogni **famiglia di processori** ha il suo insieme di istruzioni

- Programmi scritti per **famiglie diverse non sono compatibili**
- I programmi scritti per **modelli più recenti** di una data famiglia possono non funzionare in modelli più datati

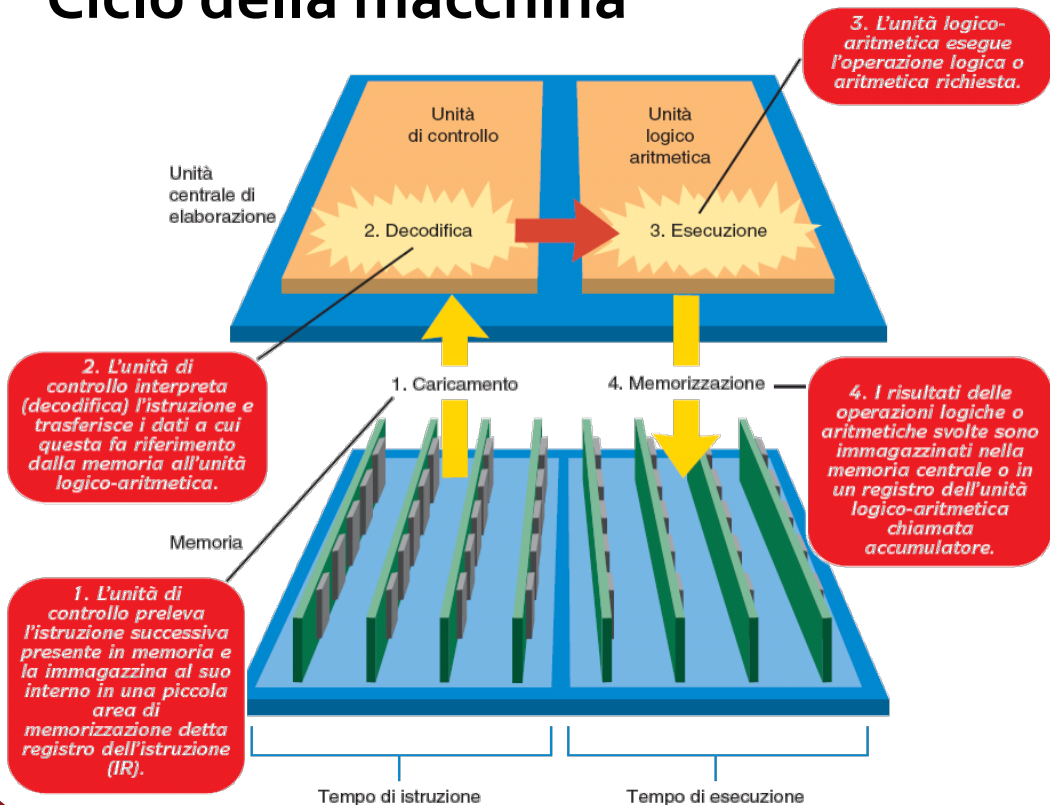
In base all'**insieme di istruzioni** i processori si dividono in:

- **RISC (Reduced Instruction Set Computer)** hanno istruzioni semplici e veloci e molti registri per i risultati intermedi
- **CISC (Complex Instruction Set Computer)** forniscono istruzioni più complesse, che mimano quelle dei linguaggi di alto livello

La maggior parte dei processori consumer sono oggi dei **CRISP** cioè un misto fra i due



# Ciclo della macchina

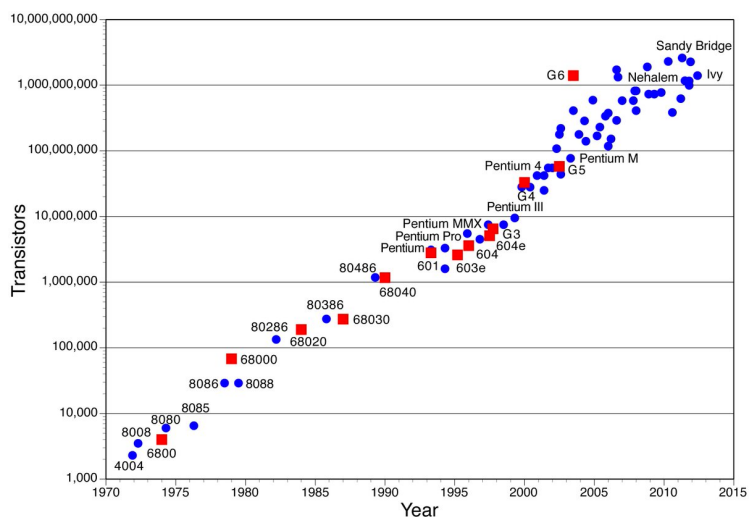


# La legge di Moore

**Gordon Moore**, cofondatore di **Intel**, formulò nel 1965 il seguente enunciato basato su un'osservazione empirica:

*La complessità di un microcircuito, misurata ad esempio tramite il numero di transistor per chip, raddoppia ogni 18 mesi*

La previsione si è rivelata corretta fino ad oggi



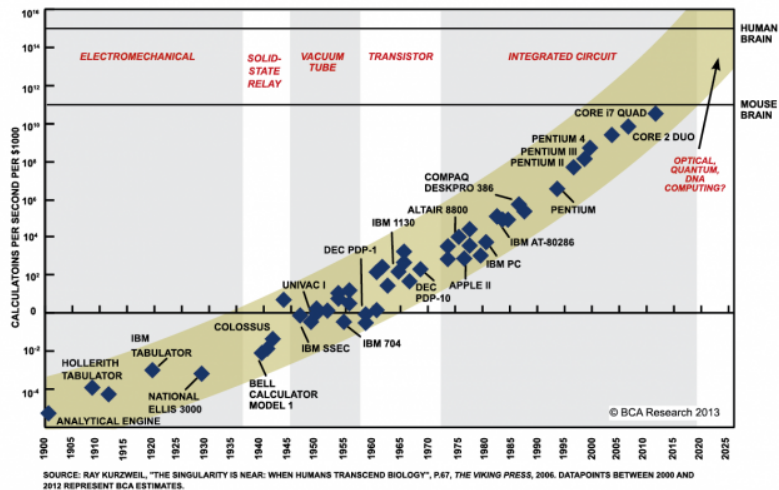
# La legge di Joy

**William Joy**, cofondatore di **Sun Microsystem**, formulò nel 1984 l'ipotesi che le prestazioni degli elaboratori sarebbero aumentate secondo un indice prevedibile:

$$\text{MIPS} = 2^{(\text{anno}-1984)}$$

Anche questa previsione si è rivelata corretta

**MIPS** = milioni di istruzioni per secondo

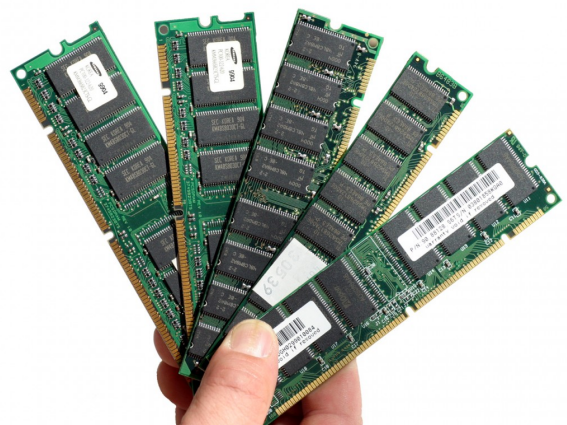


# Memoria Centrale

Un computer ha bisogno di spazio per contenere i **programmi** che sta utilizzando e i **dati** che sta elaborando

Tale spazio è la **Memoria Centrale**

- È costituita da numerosissime **celle** capaci di memorizzare **1 bit**
- Le celle sono raggruppate in **registri di memoria** (tipicamente ogni registro è composto da 8 bit = **1 byte**)
- Le **operazioni** che possono essere effettuate su un registro sono due: **lettura (load)** e **scrittura (store)**
- Ogni registro è individuato da un **indirizzo numerico**

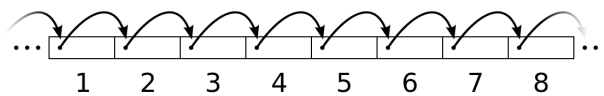


# Memoria Centrale

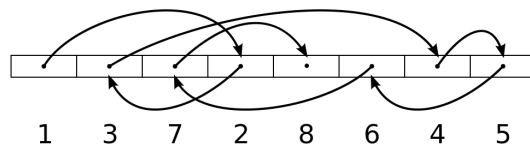
La memoria centrale è ad **accesso casuale** (o diretto)

- È possibile accedere ad un registro in un **tempo costante** e **indipendente dalla sua posizione**
- Si contrappone all'**accesso sequenziale** in cui l'accesso ad un elemento richiede più o meno tempo a seconda della sua posizione

## Accesso sequenziale



## Accesso casuale



Il tempo di accesso a un indirizzo di memoria è dell'ordine dei 0.5 nano-secondi (miliardesimi di secondo)

# Tipologie di Memoria Centrale

## RAM (Random Access Memory)

- Accessibile in **lettura e scrittura**
- Contiene **dati e programmi in esecuzione**
- **Perde il contenuto** senza alimentazione elettrica (**volatile**)

## ROM (Read Only Memory)

- Accessibile **solo in lettura** (non ci si può scrivere)
- Contiene il **firmware**: istruzioni necessarie all'avvio del computer, programma di configurazione, ecc.
- **Non perde il contenuto** allo spegnimento (**non volatile**)

# Tipologie di RAM

## DRAM (Dynamic RAM)

È il genere più diffuso, le celle di memoria **devono essere aggiornate centinaia di volte al secondo**

### Sottotipi:

- **FPM**: la prima tipologia diffusa per i sistemi desktop
- **EDO**: evoluzione della FPM, quando viene richiesto un indirizzo **invia anche il successivo** per risparmiare tempo
- **SDRAM**: **sincronizzata con il clock della CPU** per eliminare discrepanze di tempo
- **DDR**: trasporta il **doppio dei dati** nello stesso ciclo di accesso
- **DDR2**: trasporta il **quadruplo dei dati** nello stesso ciclo di accesso
- **SRAM**: **RAM statica molto veloce e costosa**, non ha bisogno di essere aggiornata continuamente, viene **utilizzata per la cache memory**

# Cache Memory

**La CPU non può lavorare direttamente sulla RAM**, ma solo su dati e istruzioni presenti nei **registri interni**

- Per eseguire un'istruzione, il processore deve **copiarla**, assieme ai dati, nei registri interni
- A volte si impiega più **tempo per la copia** che per l'esecuzione dell'istruzione sui dati di lavoro
- A che serve avere una CPU veloce se **passa il tempo in attesa** che l'istruzione e i dati vengano copiati dalla RAM?

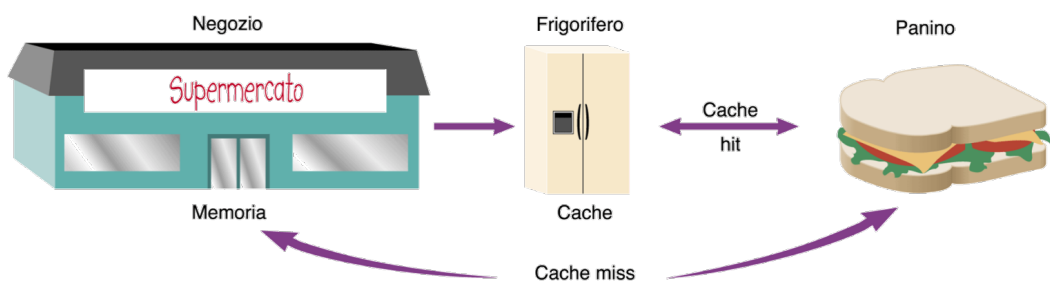
## Cache Memory

- è una memoria **posta a diretto contatto con il processore**
- è molto **veloce**, molto più **grande** dei registri interni, molto più **piccola** della RAM

# Cache Memory

Nella **Cache Memory** vengono memorizzati:

- dati e istruzioni usati più **frequentemente** o vicini a quelli appena utilizzati
- In questo modo il **reperimento è molto veloce**
  - se il dato/istruzione è **presente** nella cache (**cache hit**) lo si usa
  - se invece non c'è (**cache miss**) lo si prende dalla RAM e lo si copia nella cache: la prossima volta verrà ritrovato nella cache



# Memoria Virtuale

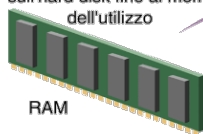
La RAM può **saturarsi** se:

- sono in esecuzione **molti programmi** o sono presenti **molti dati**

La **Memoria Virtuale** è una tecnica usata per **liberare spazio** in memoria centrale:

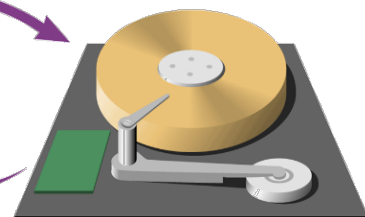
- Dati e programmi (o porzioni di essi) **poco utilizzati** vengono **spostati provvisoriamente sull'hard disk** in un'area di swap
- Tale operazione è tuttavia piuttosto **costosa in termini di tempo**

I dati in eccesso rispetto allo spazio disponibile nella RAM vengono immagazzinati sull'hard disk fino al momento dell'utilizzo



RAM

All'occorrenza i dati vengono trasferiti nella RAM, da dove altri dati passano nell'hard disk per fare posto



Hard disk

# Buffer

È una **zona di memoria** usata per compensare **differenze di velocità** nel trasferimento di dati

## Esempio

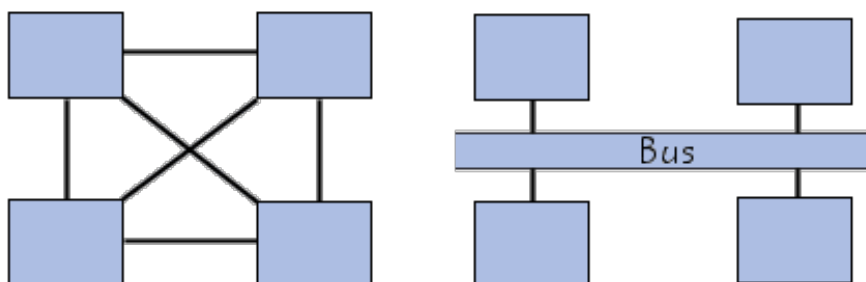
- La CPU deve inviare alcuni dati alla stampante ma:
  - la CPU lavora ad **alta velocità**
  - La **stampante** supporta una **velocità molto minore**
- La CPU **scrive i dati in un buffer** e continua a lavorare ad altro
- La stampante può stampare i dati **leggendoli dal buffer senza interrompere la CPU**



# Bus

Sono le **linee di comunicazione** che consentono la **trasmissione delle informazioni** tra le parti del computer

- Invece di collegare ogni elemento con tutti gli altri, **si collegano tutti i dispositivi ad un'unica linea**
- Per **aggiungere o eliminare** un dispositivo basta **collegarlo o scollegarlo** al bus



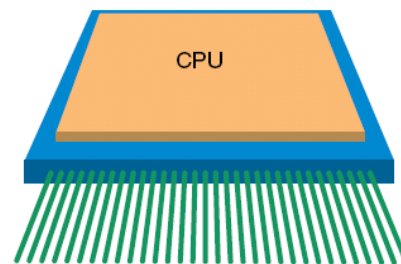
# Il Bus di Sistema

Collega la CPU agli altri dispositivi e comprende:

- Le **linee di indirizzi** che trasmettono gli indirizzi dei dati da scrivere/leggere in memoria centrale
- Le **linee di dati** che trasmettono i bit dei dati da scrivere o letti in memoria centrale

Più linee ci sono e più sono i bit che possono essere **inviati con una sola operazione**

- I primi bus erano a **8 bit**, poi si è passati a **16, 32, 64 e 128 bit**
- In genere il **numero di linee corrisponde con la capacità dei registri della CPU**



# I Bus Locali

Collegano **periferiche che trattano grandi quantità di dati** direttamente alla CPU ed alla memoria centrale

- In questo modo i dati **non passano per il bus di sistema**
- Sono utilizzati per fornire **canali privilegiati** a periferiche critiche

## Esempio

- Una schermata su un **monitor 1600 x 1200** con **3 byte** per pixel (RGB) va composta in memoria dalla CPU (o dalla GPU) e inviata al monitor **60 volte al secondo (frequenza di refresh)**
- $1600 \times 1200 \times 3 \times 60 \sim$  **330 MB al secondo** che dovrebbero viaggiare tutti sul bus di sistema
- questo traffico **intaserebbe il bus di sistema e farebbe decadere le prestazioni**



# Tipologie di Bus

## ISA (Industry Standard Architecture)

- Creato nel **1981** da **IBM** per il suo **primo personal computer**
- Ampiezza di 8 o 16 bit (32 con EISA), velocità di **5 MB/s**

## PCI (Peripheral Component Interconnect)

- Sviluppato da **Intel** all'inizio degli anni '90
- Ampiezza di 32 o 64 bit, velocità di **132 MB/s**

## AGP (Accelerated Graphics Port)

- Sviluppato da **Intel** per aumentare le prestazioni delle **schede grafiche**
- Ampiezza di 32 bit, velocità di **266 MB/s fino a 2 GB/s**

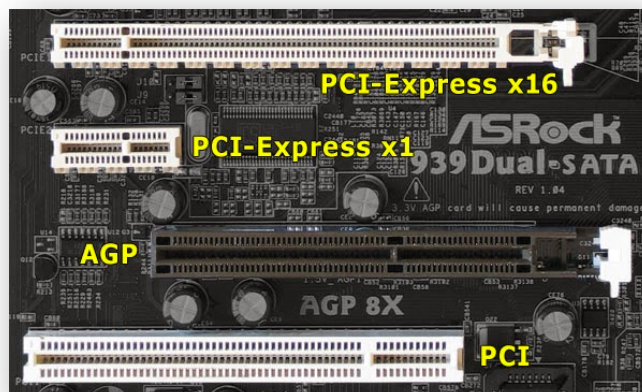
## PCI-X, PCI Express

- Evoluzioni del bus PCI, con velocità fino a **8 GB/s**

# Slot di Espansione

Sono **alloggiamenti presenti** sulla **scheda madre** collegati direttamente al **bus di sistema**

- Negli slot si collegano le **schede degli adattatori** che estendono le capacità/prestazioni del computer
- Esempi: scheda video, audio, di rete, fax, ecc.

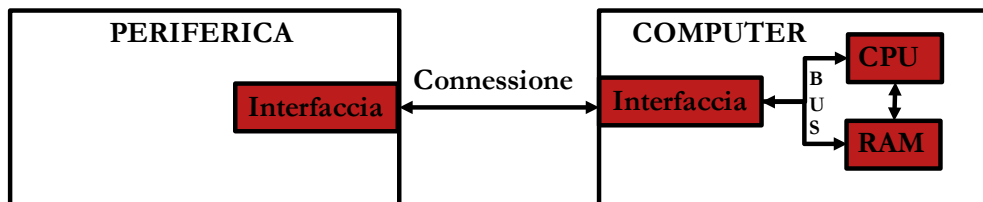




# Interfacce e Porte

Per poter essere utilizzate dalla CPU, le periferiche di I/O devono essere **collegate al bus**

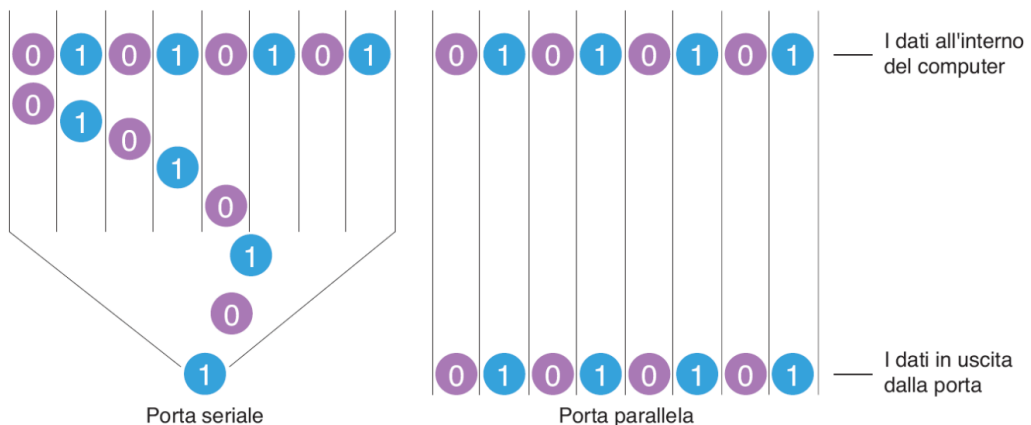
- Il collegamento tra periferica e CPU è gestito da un insieme di **hardware e software** detto **interfaccia**
- le interfacce sono **internamente connesse al bus** e terminano con **una presa** detta **porta**
- È possibile **connettere una periferica alla porta** tramite un **cavo compatibile**



# Interfacce e Porte

**Interfacce Seriali:** trasferiscono i bit uno di seguito all'altro

**Interfacce Parallele:** trasferiscono più bit (in genere 8 o 16) contemporaneamente lungo **canali separati**

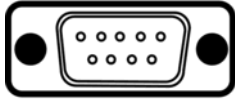


# Interfacce Seriali

Sono **più complesse** dal punto di vista architettonico e gestionale ma **più diffuse** rispetto alle parallele

- **Meno costose** (richiedono un minor numero di fili)
- **Più tolleranti rispetto a interferenze ed errori** di trasmissione

## Esempi



**RS-232**: usata nei primi PC, ormai **obsoleta**, raggiungeva velocità di circa **100 Kb/s**



**USB (Universal Serial Bus)**: permette di collegare fino a **127 dispositivi** in serie, velocità fino a **5 Gb/s**, distribuisce anche **corrente elettrica**



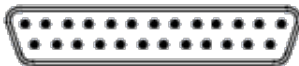
**SATA**: viene utilizzata per connettere **hard disk** o **drive ottici** interni, velocità fino a **7 Gb/s**

# Interfacce Parallele

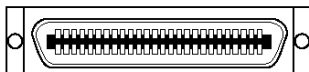
Sono **meno complesse** dal punto di vista architettonico e gestionale ma ormai anche **meno diffuse** rispetto alle seriali

- **Più costose** (richiedono un maggior numero di fili)
- **Meno tolleranti rispetto a interferenze ed errori** di trasmissione

## Esempi



**Centronics**: usata in passato soprattutto per connettere **stampanti**, velocità fino a **2 MB/s**



**SCSI** usata soprattutto per **hard disk** o **drive ottici** interni ed esterni, meno diffusa ora che in passato, velocità fino a **640 MB/s**



**PATA**: utilizzata per connettere **hard disk** o **drive ottici** interni, velocità fino a **133 MB/s**

# Interfacce Wireless

Consentono di collegare periferiche **senza utilizzare cavi di connessione**

## IrDa (Infrared Data Association)

- La comunicazione avviene tramite **emissione di raggi infrarossi**
- Velocità massima di **4 Mbit/s**
- Distanza massima di circa **1 metro**
- **non devono esserci ostacoli tra periferica e computer**

## Bluetooth

- La comunicazione avviene tramite **emissione di onde radio**
- Velocità massima di **3 Mbit/s**
- Distanza massima fino a **100 metri** (ridotta in presenza di ostacoli)

# Bluetooth

Che significa **Bluetooth**?

- Il nome è ispirato a re **Harald Blåtand** (Harold Bluetooth) abile diplomatico che riuscì a **unificare il regno di Danimarca**
- Pare che il sovrano, in battaglia, fosse solito **colorarsi i denti d'azzurro**
- Gli inventori della tecnologia hanno pensato che fosse un nome adatto per un **protocollo capace di mettere in comunicazione dispositivi diversi**
- Il logo della tecnologia unisce infatti le rune nordiche analoghe alle moderne **H e B**



# Bibliografia

- Par. 2.10 e 2.11: l'Unità centrale di elaborazione
- Par. 2.12 e 2.13: Le memorie
- Par. 2.14 e 2.15: I bus
- Par 2.16: Comunicazione con le periferiche
- Par 2.18: Interazione tra CPU e Memoria
- Approfondimenti su queste slide

