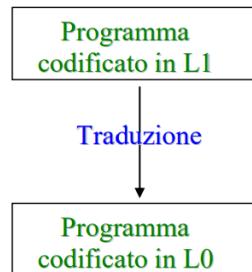


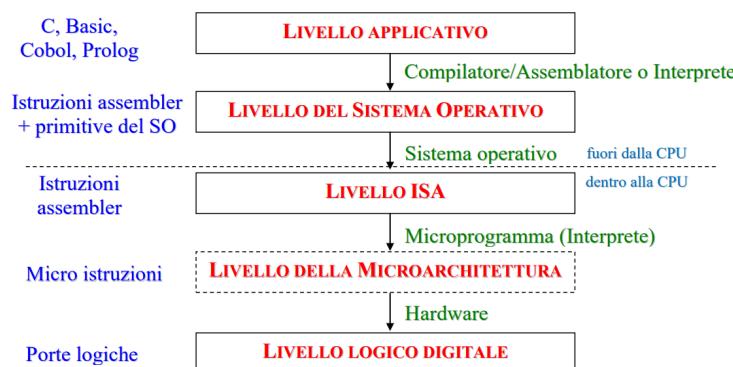
# INTRODUZIONE

## Macchine a più livelli

- Linguaggio macchina (**assembly**): insieme di istruzioni eseguite direttamente dall' hardware di un calcolatore, L0
- Limitato a causa di:
  - **Costo**: realizzare hardware che faccia operazioni complesse è costoso
  - **Complessità**
- Per semplificare programmazione sono nate le **macchine a più livelli**:
  - Viene creato un nuovo linguaggio L1, più facile da utilizzare, eseguibile da un computer virtuale
  - La macchina esegue solo istruzioni L0, tradotte da L1 a L0

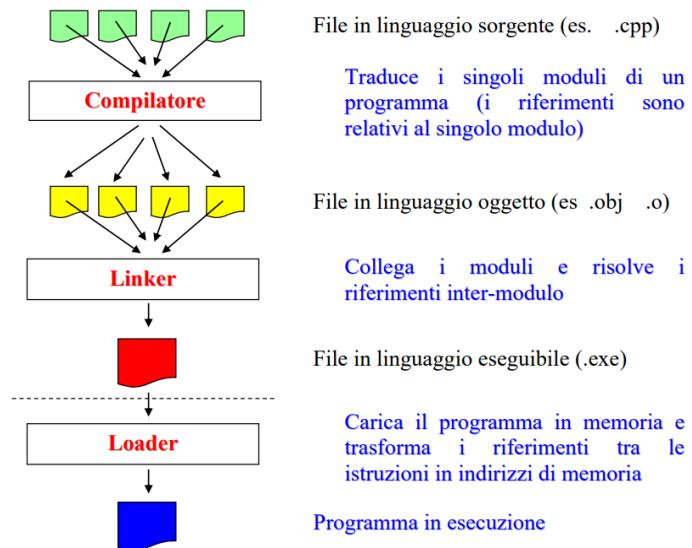


- Possono esistere + **linguaggi Ln**, ad ogni livello *i* è necessario un traduttore, che traduca le istruzioni in linguaggio macchina *i*-1
- Esistono 2 tecniche di traduzione:
  - **Compilazione**: il traduttore (**compilatore/assemblatore**) legge il programma in linguaggio *i* e produce un nuovo programma in linguaggio *i*-1.
  - **Interpretazione**: il traduttore (**interprete**) legge il programma in linguaggio macchina *i* ed esegue direttamente la traduzione sulla macchina virtuale a livello *i*-1.



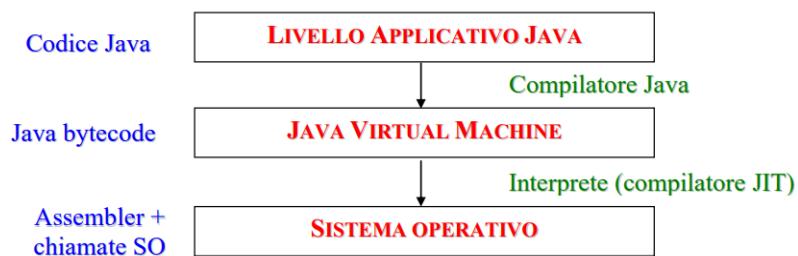
- **Livello Logico Digitale**: l'hardware del calcolatore
- **Livello della microarchitettura**: effettua microistruzioni come prelevare due numeri da registri e sommarli. Implementata tramite l'unità di controllo della CPU.
- **Livello ISA (Istruzioni Assembler)**: esegue il linguaggio assembly del calcolatore. La complessità e il numero delle istruzioni presenti a questo livello (instruction set) variano da calcolatore a calcolatore. La traduzione verso il livello della microarchitettura viene effettuata da microprogramma.
- **Livello del sistema operativo**: istruzioni eseguibili a questo livello sono le stesse del livello ISA, con funzionalità più sofisticate quali, ad esempio, una diversa organizzazione della memoria e la possibilità di far funzionare più programmi contemporaneamente.
- **Livello applicativo**: linguaggi utilizzati dai programmati (C, C++, Java).

## Traduzione di programmi C, C++

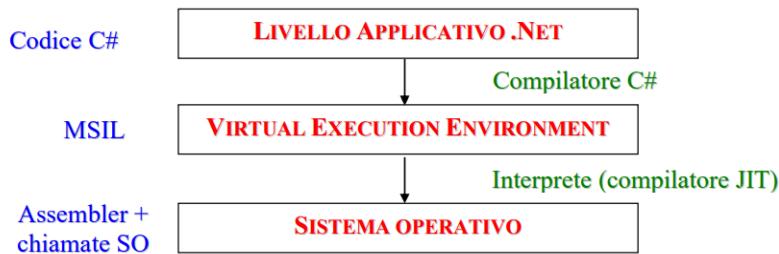


- Il traduttore è detto **compilatore** quando il linguaggio sorgente è di alto livello.
  - Il rapporto tra istruzione sorgente e istruzione macchina è **1 a molti**.
- È detto **assemblatore** quando il linguaggio sorgente è il linguaggio assembly.
  - Il rapporto tra istruzione sorgente e istruzione macchina è **1 a 1**.

### Java:

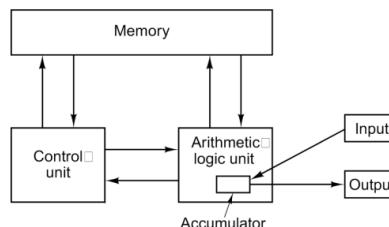


### .Net (C#):



# STORIA DEI CALCOLATORI

Anno	Nome	Costruttore	Commenti
1642	Macchina calcolatrice	Blaise Pascal	Dispositivo meccanico in grado di eseguire addizioni e sottrazioni
1672	Macchina calcolatrice	Von Leibniz	Dispositivo meccanico in grado di eseguire anche moltiplicazioni e divisioni
1834	Analytical Engine	Babbage	1° tentativo di costruire un calcolatore programmabile
1936	Z1	Zuse	1° calcolatore funzionante a relè
1943	COLOSSUS	Gov. Britannico	Primo calcolatore elettronico
1944	Mark I	Aiken	1° calcolatore americano general purpose
1946	ENIAC	Eckert/Mauchley	La storia dei calcolatori moderni comincia da questa macchina
1949	EDSAC	Wilkes	1° calcolatore con programma in memoria
1951	Whirlwind I	MIT	1° calcolatore in tempo reale
1952	IAS	Von Neumann	I calcolatori attuali usano la stessa struttura
1953	IBM 701	IBM	1° calcolatore IBM
1960	PDP-1	DEC	1° minicalcolatore (50 esemplari)
1961	1401	IBM	Calcolatore di grande successo per applicazioni commerciali in piccole aziende
1962	7094	IBM	La macchina più usata per il calcolo scientifico all'inizio degli anni '60
1963	B5000	Burroughs	1° calcolatore progettato per supportare linguaggi ad alto livello
1964	360	IBM	1° famiglia di calcolatori
1964	6600	CDC	1° calcolatore scientifico
1965	PDP-8	DEC	1° calcolatore largamente diffuso (50.000 esemplari)
1970	PDP-11	DEC	Domina il mercato dei minicalcolatori negli anni '70
1974	8080	Intel	1° calcolatore general purpose a 8 bit su un solo chip
1974	CRAY-1	Cray	1° supercalcolatore vettoriale
1978	VAX	DEC	1° super-mini a 32 bit
1981	IBM PC	IBM	Inizia l'era dei PC
1985	MIPS	MIPS	Prima macchina commerciale RISC
1987	SPARC	Sun	1° workstation RISC basata sull'architettura SPARC
1990	RS6000	IBM	1° calcolatore superscalare



## Generazione 0: CALCOLATORI MECCANICI 1642 - 1834

## Generazione 1: VALVOLE TERMOIONICHE E RELE' 1936 - 1953

- Primi calcolatori elettrici e digitali, permettono di memorizzare info. binaria
- **COLOSSUS (UK) 1943**
  - Nato per decodificare messaggi dei nazisti
  - Evoluzione di "Bomba" di Alan Turing
- **ENIAC (USA) 1946**
  - Nasce a scopo bellico
  - Basato su aritmetica decimale
- **IAS (1952)**
  - Denominata **macchina di Von Neumann**
  - Architettura tuttora alla base della maggior parte dei calcolatori digitali.
  - Memorizza anche i programmi (e non solo i dati) in forma numerica nella memoria

## Generazione 2: I Transistor (1948) 1960 - 1963

- **PDP-1**
  - Primo minicalcolatore, economico
- **Elea-9003 (Olivetti)**
  - Uno dei primi mainframe computer a transistor
  - Azienda italiana

## Generazione 3: I Circuiti Integrati (1958) 1964 - 1965

- Permisero di inserire dozzine di transistor su una singola piastrina di silicio
- Favorirono la costruzione di calcolatori più piccoli, più veloci e meno costosi.
- **IBM-360**
  - Prima famiglia di calcolatori
  - Possibilità di avere più programmi in memoria

## Generazione 4: Very Large Scale Integration '70 - '80 - Oggi

- **Federico Faggin** viene assunto da Intel e progetta il primo **microprocessore**, progenitore della famiglia x86.
- Anni '80, presenti sempre più transistor su una scheda, riducendo costo dei computer, inizia l'era dei PC
- **Osborne-I**: primo portatile

## Generazione 5: Computer invisibili e computazione omnipresente

- Forti riduzioni di **dimensioni** e integrazione in altri oggetti (**sistemi embedded**).
- **Legge di Moore (1965)**
  - Regola che descrive l'aumento del numero di transistor su un singolo chip che raddoppiano ogni 18 mesi (circa il 60% in più ogni anno).

# RAPPR. DIGITALE DELL'INFO.

## Sistemi digitali

- 2 segnali:
  - **Analogico**: valori sono infiniti in un intervallo e non numerabili.
  - **Digitale**: valori che lo rappresentano sono discreti e finiti.

## Numeri a precisione finita

Causano errori:

- **Underflow**: il risultato dell'operazione è minore del + piccolo valore rappresentabile.
- **Overflow**: il risultato dell'operazione è maggiore del + grande valore rappresentabile.
- **Non appartenenza all'insieme**

## Sistemi di numerazione

- Un sistema posizionale in base b richiede b simboli per rappresentare i diversi valori tra 0 e (b-1)
- Sistema **decimale** (b=10) 0 1 2 3 4 5 6 7 8 9
- Sistema **binario** (b=2) 0 1:
  - Ogni cifra, detta **bit** (Binary digit), può essere rappresentata direttamente tramite un livello elettrico di tensione
- Sistema **ottale** (b=8) 0 1 2 3 4 5 6 7
- Sistema **esadecimale** (b=16) 0 1 2 3 4 5 6 7 8 9 A B C D E F:
  - Utilizzato nel linguaggio dell'assemblatore poiché è molto **compatto**.
  - Ogni cifra corrisponde esattamente a 4 cifre binarie.
- **Byte**:
  - Più importante unità di misura dell'informazione del calcolatore
  - Composto da 8 bit
- **Word**: sequenze di bit più lunghe di un byte
- Con n bit sono possibili  $2^n$  configurazioni.

## Numeri binari negativi

- 4 rappresentazioni:
  - Grandezza e segno
  - Complemento a uno
  - Complemento a due
  - Eccesso  $2^{m-1}$
- **Grandezza e segno**:
  - Il bit più a sinistra viene utilizzato per il segno: 0 valori positivi; 1 quelli negativi.
  - I bit rimanenti contengono il valore assoluto del numero.

76  $\Leftrightarrow$  01001100

-76  $\Leftrightarrow$  11001100

- **Complemento a due**:
  - Il bit più a sinistra viene utilizzato per il segno: 0 valori positivi; 1 quelli negativi.

- Si sostituiscono tutti gli uno con degli zero e viceversa.
- Si aggiunge 1 al risultato.
- **Eccesso  $2^{m-1}$ :**
  - Rappresenta i numeri come somma di se stessi con  $2m-1$  dove m è il numero di bit utilizzati per rappresentare il valore.
- Complemento a due ed eccesso sono le + **efficienti**, consentono di trattare la sottrazione tra numeri come una somma tra numeri di segno opposto.
- Il numero di configurazioni rappresentabili non cambia ma l'intervallo positivo è **dimezzato** a favore dei valori negativi.

<b>n</b>	<b>Configurazioni</b>	<b>Positivi</b>	<b>Positivi e Negativi</b>
8	256	0 – 255	-128 – 127
16	65.536	0 – 65.535	-32768 – 32767
32	4.294.967.296	0 – 4.294.967.295	-2.147.483.648 – 2.147.483.647

## Mascherature dei bit

- **Maschera di bit:** configurazione di un insieme di bit di lunghezza predefinita.
- **Impostare a 1** uno o più bit:
  - Uso **OR**, metto 1 dove voglio impostare, 0 nei restanti
- **Impostare a 0** uno o più bit:
  - Uso **AND**, metto 0 dove voglio impostare, 1 nei restanti
- **Verificare** se uno o più bit è a 0 oppure 1:
  - Uso **AND**, metto 1 dove mi interessa, 0 nei restanti

## Numeri floating-point

- Notazione in cui la gamma dei valori esprimibili è indipendente dal numero di cifre significative

$$n = f \times 10^e$$

↓  
 frazione o mantissa

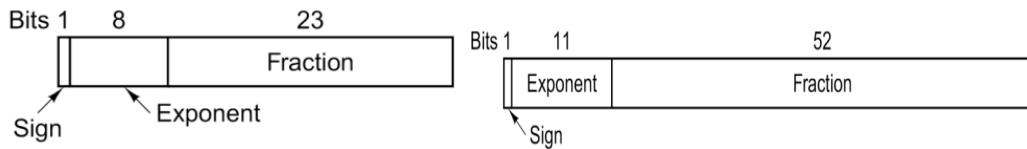
esponente

- E' stata definita una **codifica standard** in cui i valori della mantissa f sono compresi tra 0.1 e 1 ( $0.1 \leq f \leq 1$ ).

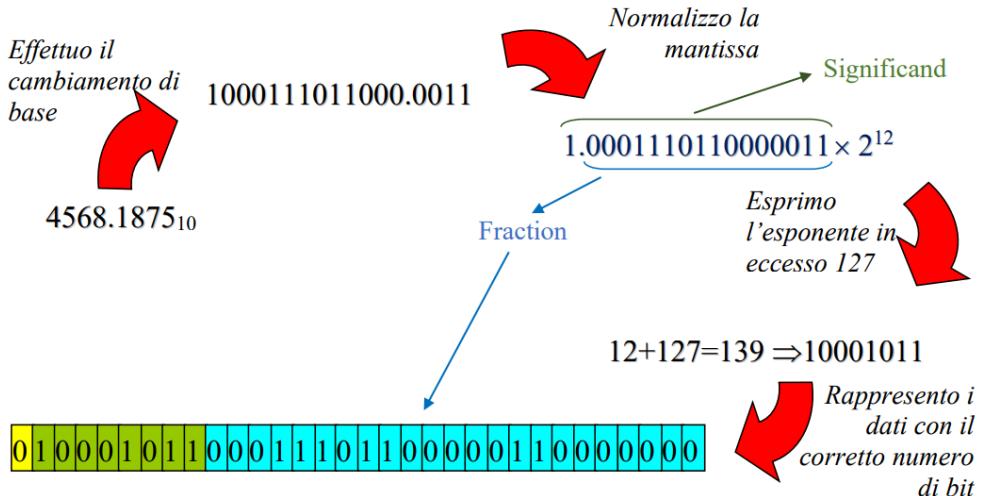
## IEEE 754: Standard floating-point

- Prevede vari formati, i più utilizzati sono:
  - **Singola precisione**
  - **Doppia precisione**

Elemento	Singola Precisione	Doppia Precisione
Bit di segno	1	1
Bit per l'esponente	8	11
Bit nella parte frazionaria (fraction)	23	52
Numero totale di bit	32	64
Rappresentazione dell'esponente	Eccesso 127	Eccesso 1023
Campo dell'esponente	Da -126 a +127	Da -1022 a +1023
Numero più piccolo	$2^{-126} \approx 10^{-38}$	$2^{-1022} \approx 10^{-308}$
Numero più grande	$\approx 2^{128} \approx 10^{38}$	$\approx 2^{1024} \approx 10^{308}$



- **Forma normalizzata:**



## Rappresentazione di caratteri

- All'interno del calcolatore i caratteri sono codificati sotto forma di numeri.
- La mappatura dei caratteri in numeri è detta **codice di caratteri**
- Per poter comunicare i calcolatori devono utilizzare la stessa mappatura.
- **ASCII** (American Standard Code for Information Interchange):
  - Ogni carattere ASCII utilizza **7 bit**, sono quindi possibili **128 configurazioni**.
  - I caratteri da 0 a 31 sono caratteri di **controllo**.
  - Dato che i dati vengono trasmessi in byte l'ottavo bit viene utilizzato per verificare la correttezza del dato inviato
  - **Latin-1**: prima estensione dell'ASCII
- **UNICODE**:
  - Sistema per la codifica dei caratteri di quasi tutte le lingue vive
  - **v1.0**: codifica a 16 bit, quindi con 65.536 simboli (chiamati **code point**)
  - **v2.0**: code point codificati con un numero di bit maggiore di 16.
  - **v10.0** (2017): 136.000 code point, con codifica fino a 21 bit.
  - E' uno standard internazionale (**ISO/IEC 10646**)
  - Varie codifiche possibili:
    - **UTF-8** (unità da 8 bit, 1/2/3/4 byte per code point): utilizzato per compatibilità; tipicamente adottato nei file txt/html/xml.
    - **UTF-16** (unità da 16 bit, 2/4 byte per code point): utilizzato dai moderni sistemi operativi e linguaggi di programmazione per codifica di stringhe
    - **UTF-32** (unità da 32 bit): tutti i code point sono codificati in 4 byte.

## Rappresentazione di suoni

- Un suono è rappresentato come un'onda che descrive la variazione della pressione dell'aria nel tempo. Si tratta di un **segnale analogico**.
- Parametri importanti della codifica digitale:
  - **Campionamento:** scompongo la funzione iniziale
  - Numero di bit per la **quantizzazione** di ciascun campione: quantizzo il segnale a un livello discreto (assegno un valore a ciascun segnale: 1, 2, 7...)
- Alta frequenza ed elevato numero di bit per quantizzazione = alta qualità del suono

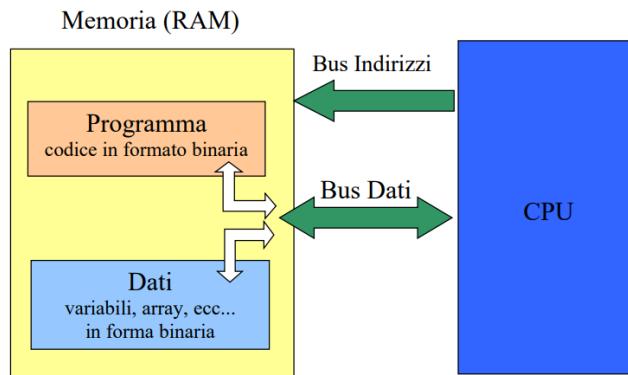
## Rappresentazione di immagini

- L'immagine del video è rappresentata tramite una matrice di **pixel** (PIcture Element) di ognuno dei quali è memorizzata l'intensità luminosa (e/o il colore).
- Parametri importanti sono:
  - **Dimensione** (risoluzione)
  - **Profondità**
  - **Formato** di rappresentazione (Grayscale, RGB, Palette)
- **Grayscale:**
  - 1 pixel codificato con **1 byte** → 256 livelli di grigio
  - Livello 0 → Nero
  - Livello 255 → Bianco
  - Un'immagine di dimensioni  $W \times H$  occupa  **$W \times H$  bytes**.
- **RGB(24):**
  - 1 pixel codificato con **3 byte** (R,G,B) ciascuno dei quali specifica l'intensità di uno dei tre colori fondamentali (sintesi additiva)
  - Il numero di colori disponibili è  $2^{24} = 16.777.216$
  - Livello 0,0,0 → Nero
  - Livello 255,255,255 → Bianco
  - X,X,X → Grigio
  - Un'immagine di dimensioni  $W \times H$  occupa  **$W \times H \times 3$  byte**.
- **N numeri di colori:**
  - N bit per codificare ciascun pixel → il numero di colori disponibili è  $2^N$ .
  - Un'immagine di dimensioni  $W \times H$  occupa  $(W \times H \times N) / 8$  byte.
- **Palette:**
  - Tabella che definisce  $2^c$  colori (ognuno caratterizzato da una tripletta RGB, con profondità p bit) e associa a ciascun colore un indice (posizione nella paletta).
  - Il colore di ogni pixel è definito dall'**indice** del colore nella paletta. In questo modo in un'immagine, pur potendo utilizzare  $2^c$  colori diversi, questi potranno essere  $2^p$ .

# ARCHITETTURA DEL CALCOLATORE

- Un calcolatore digitale è un sistema composto da processori, memorie e dispositivi di input/output (I/O) collegati tra loro tramite un Bus.
- Organizzazione **bus-oriented**: un bus è un insieme di connessioni elettriche parallele utilizzate per trasportare le informazioni da un componente all'altro.

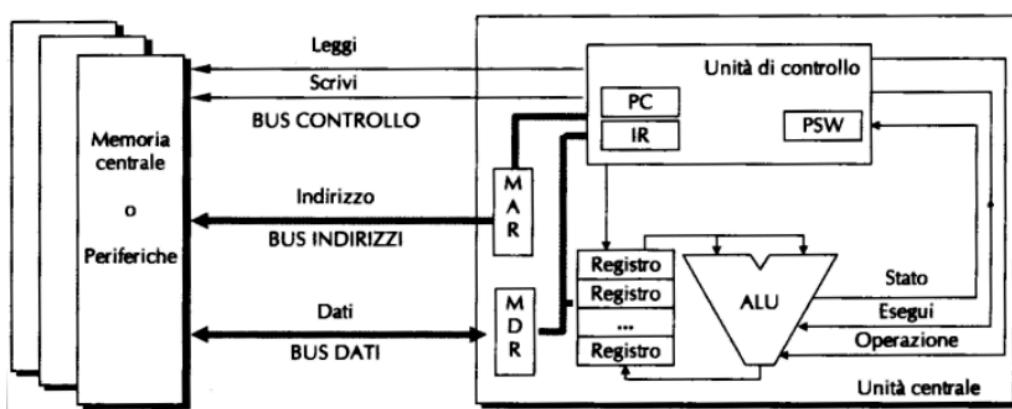
## Architettura di Von-Neumann



- Novità: utilizzare la memoria non solo per i dati ma anche per i programmi
- Programmi e dati al tempo di esecuzione sono caricati in memoria
- Trasferiti entrambi sul bus dati, il bus indirizzi serve alla cpu per indicare alla memoria le locazioni dove risiedono le informazioni da trasferire.

## CPU

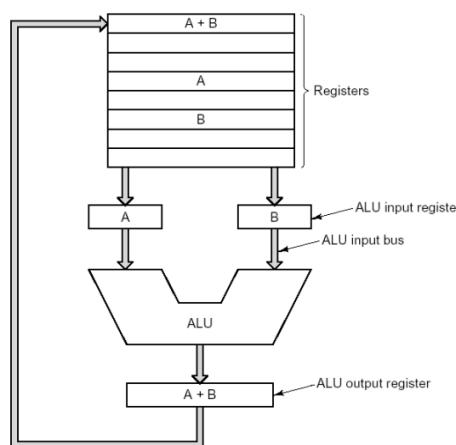
- Il compito della CPU è quello di eseguire i programmi immagazzinati nella memoria centrale leggendo le loro istruzioni ed eseguendole in sequenza.
- Composta da:
  - **Unità di controllo:**
    - legge le istruzioni dalla memoria centrale e ne determina il tipo.
  - **ALU:**
    - esegue le operazioni necessarie all'esecuzione delle istruzioni (AND, OR, addizione binaria).
  - **Registri:**
    - sono una piccola memoria ad alta velocità utilizzata per memorizzare i risultati temporanei e le informazioni di controllo necessarie al funzionamento dell'ALU.
    - i registri risiedono all'interno della CPU.
    - **Program Counter (PC):** registro più importante, indica la prossima istruzione da eseguire.
    - **Instruction Register (IR):** memorizza l'istruzione che si sta per eseguire.



Esecuzione di un'istruzione:

1. la CPU mette il valore di **PC** su **MAR** e attiva la linea **Leggi**;
2. la memoria con il **bus indirizzi** accede a **MAR** e lo scrive su **MDR** con il **bus dati**;
3. la CPU copia su **IR** il valore di **MDR** e **decodifica** l'istruzione;
4. l'istruzione passa in esecuzione sulla **ALU**;
5. se l'istruzione prevede la lettura di operandi dalla memoria, devono essere caricati sui registri; per ciascun operando da reperire:
  - 5.1. la CPU mette l'indirizzo dell'operando su **MAR** e attiva la linea **Leggi**;
  - 5.2. la memoria con il **bus indirizzi** accede a **MAR** e lo scrive su **MDR** con il **bus dati**;
  - 5.3. la CPU copia sul registro destinazione il valore dell'operando che è in **MDR**;
6. terminata l'esecuzione la CPU copia sul registro destinazione il valore prodotto dalla **ALU**; se è prevista scrittura in memoria del valore calcolato:
  - 6.1. la CPU mette l'indirizzo della cella di destinazione su **MAR** e il risultato su **MDR** e attiva la linea **Scrivi**;
  - 6.2. la memoria attraverso il **bus indirizzi** accede a **MAR**, attraverso il **bus dati** a **MDR** e, una volta reperito il valore in **MDR**, lo scrive sulla propria cella interna indicata da **MAR**;
7. Si ritorna al punto 1 dopo aver aggiornato il valore di **PC** (prossima istruzione da eseguire).

## Data Path

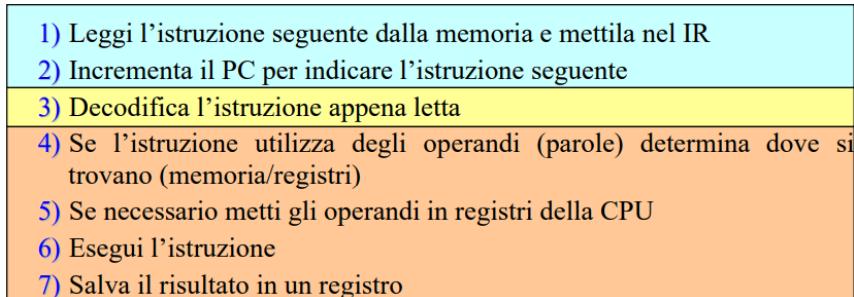


- Data Path di una CPU di Von-Neumann, comprende l'**ALU** e i **registri**.
- **Ciclo di Data Path**: passaggio di due operandi attraverso la **ALU** e la memorizzazione del risultato in un nuovo registro
- Ogni istruzione ISA richiede 1 o più cicli di data path, es. la divisione
- Ciclo di data path = ciclo di clock (tempo sincronizzazione operazioni della cpu in ns)

- durata ciclo di data path = durata ciclo di clock =  $1/F$ ; 1 Hz = 1 operazione al secondo
- durata istruzione ISA =  $n \times$  durata ciclo di data path ( $n$  variabile)
- Istruzioni ISA per sec. =  $1/\text{durata istruzione ISA} = F / n$

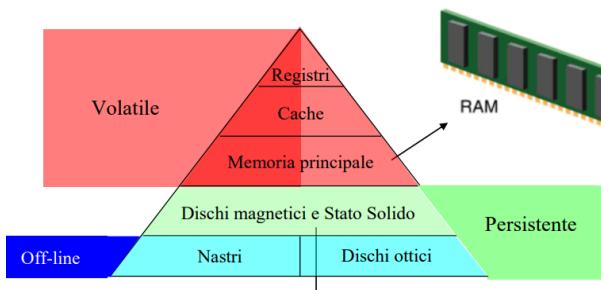
## Esecuzione delle istruzioni

- CPU opera in modo ciclico, ripete le seguenti operazioni fino al termine
- **Fetch:** acquisizione dalla memoria di un'istruzione del programma.
- **Decode:** decodifica istruzione appena letta
- **Execute:** effettuazione delle operazioni corrispondenti all'istruzione.



## Memorie

- Componenti del calcolatore per memorizzare le informazioni



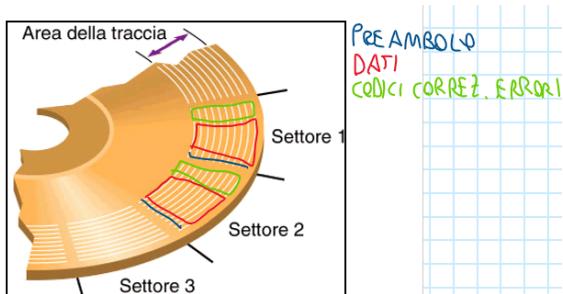
- **Volatile:** l'informazione rimane memorizzata fino a che il calcolatore è alimentato
  - + veloci, - capacità, andando in giù - veloci + capacità
- **Persistente:** l'informazione rimane memorizzata anche se il calcolatore è spento
- **On-line:** i dati sono sempre accessibili
- **Off-line:** il supporto deve essere montato per poter accedere ai dati

PER CORREZIONE ERRORI MEMORIZZAZIONE GUARDARE DISPENSE

- Memoria Principale:
  - Immagazzina i programmi in esecuzione e i relativi dati.
  - Volatile
- Memorie composte da celle in grado di memorizzare le info. e hanno indirizzo
- Parola:
  - Numero di bit associato a un indirizzo
  - Parola minima: byte(8 bit) ma ci sono anche da 32 o 64 bit
- Gli indirizzi di memoria sono in binario: se un indirizzo ha  $m$  bit il numero massimo di celle indirizzabili sarà  $2^m$ .
- SIMM (obsolete):
  - Contatti solo su un lato, trasferiscono 32 bit per ciclo di clock, bus dati a 32 bit
- DIMM:
  - Contatti su entrambi lati, bus dati a 64 bit
- SO-DIMM:
  - dimm di dimensione ridotta per portatili
- ECC-DIMM: hanno meccanismo di correzione errori

## Dischi magnetici

- **Hard-disk:**
  - Dispositivo per la conservazione di informazioni sotto forma magnetica
  - Supporto rotante a forma di piatto su cui agiscono delle testine di lettura/scrittura.
  - La testina di un disco è sospesa sopra la superficie e viene sostenuta da un cuscino d'aria.
- Scrittura:
  - Quando passa la corrente nella testina, viene magnetizzata la superficie appena sotto la testina e le particelle magnetiche si allineano verso sinistra o destra a seconda della polarizzazione della corrente
- Lettura:
  - Quando una testina passa sopra un'area magnetizzata viene indotta una corrente nella testina e ciò permette di rileggere i bit memorizzati.
- **Traccia:** sequenza circolare di bit
- **Settore:** porzione di traccia di dimensione fissa



- **Preambolo:** necessario alla testina per sincronizzarsi prima della lettura/scrittura.
- **Dati:** l'insieme dei byte memorizzati nel settore
- **Codice correzione errori**
- **Cilindro:** insieme delle tracce in una data posizione radiale
- La capacità di un hard disk è in gran parte definita dalla densità di registrazione
- Le prestazioni di un hard disk dipendono da diversi fattori:
  - Seek time: tempo necessario per spostare le testine sul cilindro desiderato
  - Latency time: tempo necessario affinché il settore interessato passi sotto la testina. Dipende dalla velocità di rotazione del disco

## SSD

- Dispositivi elettronici, basati su memorie flash, nessuna parte in movimento
- Pro:
  - Tempo di accesso ridotto: < 100 microsecondi
  - Maggiore velocità trasferimento dati
  - Minore possibilità di rottura e maggiore durata
  - Rumorosità assente
  - Minori consumi
  - Maggiore resistenza agli urti
  - Maggiore capacità: 100TB vs 20TB degli HD
- Contro:
  - Maggiore prezzo per bit
  - Possibile minor durata se usata per frequenti scritture

## Dischi magnetici: Interfacce

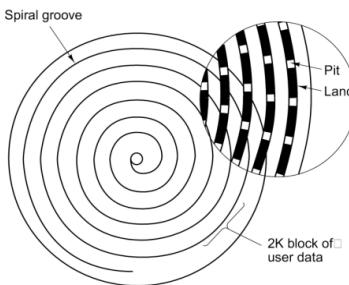
- diversi tipi di interfacce di trasferimento dati tra hard-disk e bus
- **Serial ATA:**
  - Velocità elevata (6 Gbit/s)
  - Costo minore per i cavi
  - Migliore ventilazione del PC
  - Consumi ridotti
  - **Hot Swap** (collegamento senza dover spegnere sistema)
  - 128 device collegabili contro i 2 di PATA
- **NVMe:** interfaccia specifica per collegare SSD al bus PCI express

## Dischi magnetici: RAID

- Dischi **RAID**: soluzione parallela per colmare divario prestazioni CPU e disco
  - Gestito da SO
  - Appare al sistema come unico disco
  - Aumenta prestazioni distribuendo dati su più dischi
  - Fornisce modalità di backup, permette di funzionare anche se una non va
- Diverse configurazioni:
  - Non-redundant data striping
  - Redundant data striping
  - Block-interleaved parity

## Dischi ottici

- **CD (Compact Disc)** usano principio ottico, non magnetico, per memorizzazione
- Nati nell' '80 per memorizzazione musica
- Info codificate per mezzo di fori (**Pit**) di 0,8 micron di diametro
- Pit alternati con zone piane (**Land**) lungo un'unica spirale
- Passaggio Pit-Land o Land-Pit codifica un 1, l'assenza di variazioni lo 0



- Info lette tramite raggio laser
- CD prodotti utilizzando stampo dove viene iniettata resina liquida che preserva Pit
- Nella spirale i dati sono memorizzati con la stessa densità, quindi il CD ruota con velocità angolare non costante
- **CD-ROM (CD-Read Only Memory):** stessa tecnologia dei CD
- Ogni byte codificato da un **simbolo** di 14 bit
- **Frame:** gruppo di 42 simboli, 192 bit di dati e 396 di correzione errore
- **Settore di CD-ROM:** gruppo di 98 frame
  - Ogni settore inizia con un preambolo di 16 byte per identificare settore
- 2 modalità di registrazione:
  - Modo I: 16 byte preambolo+2048 byte dati+288 byte corr. errori
  - Modo II: 16 byte preambolo+2336 byte dati, usato per musica

- **Single Speed:** velocità base dei CD-ROM, 75 settori/sec
- Hanno proprio file-system chiamato **High Sierra**
- Si danneggiano facilmente
  
- **CD-R (CD-Recordables):** '90 stesse funzioni dei ROM ma registrabili da utenti
- Riflettività di pit e land viene ottenuta bruciando con raggio laser materiale colorato
- Lo standard prevede che si possa scrivere su disco in modo incrementale
- **CD-ROM Track:** gruppo di settori scritti nello stesso momento
- Tracce si possono raggruppare in sessioni, **CD-ROM Multisessione**
  
- **DVD (Digital Video Disk):** fine '90
- Stesso progetto CD-ROM con innovazioni:
  - Pit più piccoli (0,4 micron)
  - Spirale più stretta
  - Raggio laser rosso
- Single Speed: 1,4 MB/sec
- 4 formati:
  - Lato unico - Strato unico
  - Lato unico - Strato doppio
  - Lato doppio- Strato unico
  - Lato doppio - Strato doppio
  
- **Blu-Ray:** tecnologia per sostituire DVD
- Usa laser blu che, con una minore lunghezza d'onda, ha pit e land più piccoli e migliore messa a fuoco.

## Schede grafiche

- Traduce la rappresentazione dell'immagine prodotta dal processore in un formato visualizzabile dal monitor.
- Segnale **Analogico:** monitor con tubo catodico
  - Traduzione del segnale effettuato da RAMDAC
- Segnale **Digitale:** monitor LCD
  - **Adatta** l'immagine a colori e risoluzione
  - **Memoria** da utilizzare come buffer
  - Supporta **accelerazione grafica** (3D)
- Principali formati di output:
  - VGA:
    - Standard analogico per monitor tubo catodico
    - Problemi: rumore elettrico, distorsione immagine
  - DVI:
    - Monitor LCD
    - Risolve problemi VGA
    - Ogni pixel dell'output corrisponde pixel dello schermo
  - HDMI:
    - Sostituisce standard precedenti, trasporta anche dati audio
  - DP:
    - Standard che collega sorgente video digitale a un monitor
- Dimensione della griglia di pixel:
  - determina la definizione dell'immagine
  - Rapporto tipico: 4:3 e 16:9

- Profondità del colore dei pixel:
  - numero di bit utilizzati per codificare il colore di ogni singolo pixel
- **PCI Express:**
  - Evoluzione del bus PCI
  - Interfaccia di connessione delle schede grafiche
- **GPU (Graphical Processing Unit):**
  - Processore dedicato alle schede grafiche per evitare spreco di CPU
  - Utilizzato per grafica 3D e grafica raster
- Nelle librerie standard (OpenGL), che permettono programmati di prescindere dalla scheda grafica utilizzata

## Monitor LCD (Liquid Crystal Display)

- Qualità grafica migliore, minore ingombro e peso rispetto ai CRT
- Cristalli liquidi:
  - Molecole organiche vischiose
  - Scorrano in un liquido ma sono anche dotate di una struttura spaziale.
- Con campo elettrico si fanno variare le proprietà ottiche dei cristalli e quindi l'angolo di polarizzazione della luce che li attraversa
- Composto da **Celle**:
  - All'interno delle celle ci sono i cristalli liquidi
  - Ogni cella è provvista di **contatti elettrici** per applicare campo elettrico
  - Celle contenute in **2 schermi polarizzatori** lungo assi perpendicolari
  - Una **luce** dietro la lastra posteriore
- Cristalli polarizzati: pixel scuri
- Cristalli non polarizzati: pixel chiari
- Ogni cella divisa in 3 sezioni, una con filtro rosso, verde, blu
- LCD a Matrice Passiva (vecchi):
  - 1 gruppo di contatti per ogni riga e colonna dello schermo
  - Problemi di:
    - Persistenza
    - Diafonia
- LCD a Matrice Attiva:
  - A ogni pixel associato un TFT
  - Campo elettrico applicato a ogni pixel
- AMOLED (+ recente):
  - Costituita da led organici che non bloccano la luce ma la emettono

## Mouse

- Periferica di puntamento
- 3 tipi:
  - Meccanici:
    - Sfera fa girare due rotelle forate
    - Velocità di rotazione misurata da sensori a infrarossi
  - Ottici:
    - Utilizzavano un led e un array di fotodiodi
    - Rilevava movimenti sulla superficie d'appoggio
  - Laser:
    - Sono come ottici ma usano un laser al posto di un led
    - Miglior precisione e sensibilità al movimento

## Stampanti

- Uno dei principali dispositivi di output
- Livello di definizione di stampa si misura in **dpi** (dots per inch)
- Diversi tipi:
  - **Stampante a matrice:**
    - Basate su una testina mobile contenente aghi
    - Qualità di stampa dipende da numero di aghi
  - **Stampante a getto d'inchiostro:**
    - Ampiamente utilizzate per stampe a colori
    - Testina mobile contiene 4 cartucce di colore (ciano, giallo, magenta, nero) che viene spruzzato sulla carta per mezzo di minuscoli ugelli.
    - Modelli **piezoelettrici:**
      - Cristallo opera come tappo
      - Si deforma quando si applica una tensione e fa uscire goccia
    - Modelli **termici o bubble-jet:**
      - Dentro ogni ugello una goccia viene scaldata fino a ebollizione che comporta scoppio della goccia e la sua fuoriuscita
      - Qualità di stampa: 300dpi fino a 4000dpi
  - **Stampante Laser:**
    - Tamburo caricato elettricamente e coperto di materiale fotosensibile
    - Tramite specchio una luce laser passa sul tamburo, punti colpiti dal laser perdono la carica elettrica
    - Ruotando il rullo raggiunge il toner e le parti cariche attirano il materiale colorante
    - Rulli riscaldati fissano il materiale colorante sulla carta
    - Rullo viene scaricato e pulito
    - Per diverse intensità di grigio si usa **halftoning:**
      - Immagine divisa in celle 6x6, in base al livello di grigio medio della cella vengono lasciati bianchi un tot di pixel

# LIV. ISA(Instruction Set Architecture)

- **Assembly language:**
  - Linguaggio costituito da codici mnemonici corrispondenti alle istruzioni ISA.
  - Fornisce facilitazioni come etichette simboliche, macro ecc...
- **Assembler:**
  - Traduce i codici mnemonici in numerici
  - L'insieme di questi codici costituisce i programmi eseguibili .exe
- **IA-32:** è l'ISA dei processori x86 a 32 bit
- Una CPU IA-32 ha 3 modalità operative:
  - **Real:**
    - Opera fisicamente come un 8086(capostipite degli x86 a 16 bit)
  - **Virtual-8086:**
    - Opera in emulazione 8086 grazie al SO
  - **Protected:**
    - Opera come una CPU IA-32 e non come un costoso 8086
- Nelle CPU x86 informazioni memorizzate in modalità **Little Endian**:
  - Byte meno significativo del numero memorizzato nel byte con offset minore
- **Big Endian:**
  - Byte più significativo del numero memorizzato nel byte con offset minore

## Registri di base

- EIP (Instruction Pointer):
  - Contiene l'indirizzo della prossima istruzione da eseguire
- EFLAGS (Bit di stato):
  - Contiene diversi bit utili
  - Condition code:
    - Riflettono il risultato dell'operazione più recente
    - Utilizzati nelle istruzioni di salto condizionale

## Unità Floating Point

- Unità incorporata all'interno del chip della CPU
- Utilizza registri aggiuntivi
- Fornisce molte istruzioni:
  - Aritmetiche: moltiplicazione, radice quadrata, somma ecc...
  - Confronto
  - Funzioni trascendenti: seno, coseno, log ecc...
  - Caricamento di costanti note:  $\pi$ ,  $e$ , ecc...

# Algebra di Boole

- Tipo di algebra in cui le variabili e funzioni hanno solo valori 0 e 1.
- **Isomorfa** ai circuiti digitali:
  - Circuito digitale può essere espresso tramite un'espressione booleana e viceversa.
- Una funzione se ha  $n$  variabili di input,  $2^n$  combinazioni possibili.
- Una funzione descritta tramite tabella di verità con  $2^n$  righe
- Algebra di Boole si basa su 3 operazioni: AND, OR, NOT
- Due funzioni booleane sono equivalenti se hanno stesso output per qualsiasi input

Name	AND form	OR form
Identity law	$1A = A$	$0 + A = A$
Null law	$0A = 0$	$1 + A = 1$
Idempotent law	$AA = A$	$A + A = A$
Inverse law	$A\bar{A} = 0$	$A + \bar{A} = 1$
Commutative law	$AB = BA$	$A + B = B + A$
Associative law	$(AB)C = A(BC)$	$(A + B) + C = A + (B + C)$
Distributive law	$A + BC = (A + B)(A + C)$	$A(B + C) = AB + AC$
Absorption law	$A(A + B) = A$	$A + AB = A$
De Morgan's law	$\bar{AB} = \bar{A} + \bar{B}$	$\bar{A} + \bar{B} = \bar{A}\bar{B}$

## Livello Logico Digitale

- Porte logiche vendute in unità chiamate Circuiti Integrati (IC) o Chip:
  - **SSI**(Small Scale Integrated): da 1 a 10 porte.
  - **MSI**(Medium Scale Integrated): da 10 a 100 porte
  - **LSI**(Large Scale Integrated): da 100 a 10.000 porte
- E' un pezzo rettangolare di silicio su cui sono realizzate la porte

## Fabbricazione circuiti integrati

1. Fusione di **sabbie silicee** e realizzazione dei **wafer** (dischi)
2. **Deposito** di strati di materiali (semiconduttori, metalli) sul wafer
3. **Fotolitografia**:
  - a. Ricopertura del wafer con un **fotoresist**
  - b. Esposizione a **luce UV** che passa attraverso zone chiare di una **maschera**
  - c. Zone illuminate solubili e si asporta il fotoresist, aree rimangono scoperte
  - d. Con **lavaggio chimico** si rimuovono fotoresist sottostanti
4. Fotolitografia e deposito viene ripetuto svariate volte
5. **Separazione** dei chip dello stesso wafer

## Printed Circuit Board (PCB)

- Scheda utilizzata per il supporto meccanico e collegamento di componenti elettronici
- Costituito da **substrato isolante** e da strati conduttori in **rame** che consentono collegamento tra componenti tramite **piste**

# Tipologie di circuiti digitali

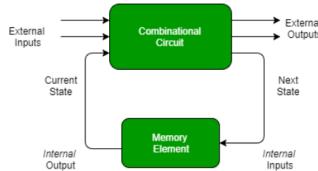
- **Circuito Combinatorio:**

- Il valore delle uscite è determinato solo dai valori degli ingressi.
- Assenza di connessioni di feedback (all'indietro).
- Analisi circuitale **semplice**.



- **Circuito Sequenziale:**

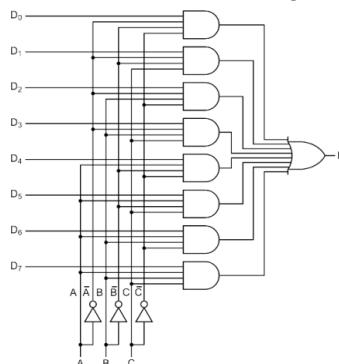
- Il valore delle uscite dipende anche dalla storia passata del circuito.
- Connessioni di feedback.
- Sincroni e asincroni
- Analisi circuitale complessa



## Alcuni circuiti MSI (combinatori)

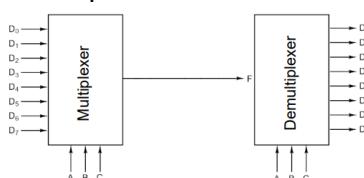
- **Multiplexer:**

- Circuito con  $2^n$  Input, 1 Output e  $n$  input di **Controllo**.
- Quelli di controllo selezionano quale tra gli input deve essere portato in uscita



- **Demultiplexer:**

- Circuito inverso del multiplexer
- 1 Input smistato da  $n$  input di **controllo** su una delle  $2^n$  linee **output**.



- **Decoder:**

- $n$  segnali in **input**,  $2^n$  linee **output**, solo una impostata a 1 dal num. in input
- **Traduttore** dal sistema binario posizionale a dove ogni bit è un carattere

- **Comparatore:**

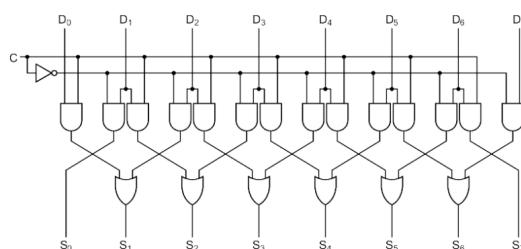
- **Confronta** 2 parole in input e produce un output che indica uguaglianza o no
- Può essere realizzato con porte **XOR**

## Array di porte logiche programmabili (PLA)

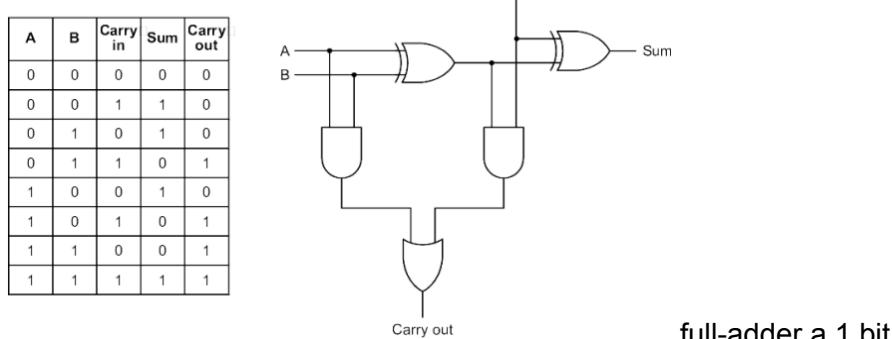
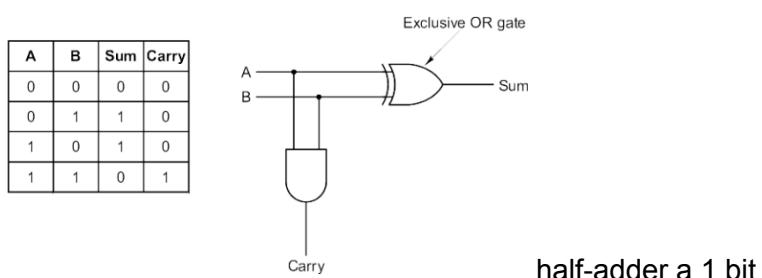
- Chip costituito da molte porte collegate internamente
- In un circuito con  $n$  Input,  $m$  Output e  $k$  unità interne:
  - Di ciascuno degli  $n$  input viene internamente generato il complemento.
  - Il chip contiene un array di  $k$  porte AND a  $2 \times n$  input.
  - Il chip contiene  $m$  porte OR a  $k$  input.
- **FPGA:**
  - PLA complesso, milioni di porte logiche.

## Circuiti aritmetici

- Circuiti specializzati nel calcolo di semplici operazioni aritmetiche.
- **Shifter:**
  - Circuito con  $n$  Input ( $D_0 \dots$ ) e  $n$  Output ( $S_0 \dots$ ).
  - I bit in output sono la copia di quelli in input traslati di una posizione a dx o sx
  - Direzione impostata da bit di controllo  $C$



- **Adder:**
  - Per realizzare un sommatore a  $n$  bit, utilizzati  $n$  full-adder a 1 bit
  - Full-adder realizzati a loro volta da half-adder a 1 bit



- Per full-adder a  $n$  bit si utilizzano  $n$  full-adder a 1 bit in parallelo

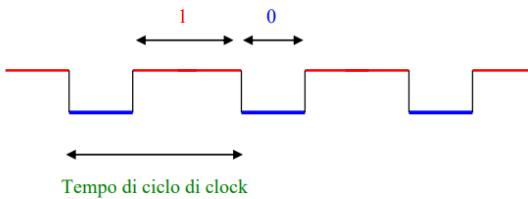
### • **ALU:**

- Circuito capace di eseguire operazioni aritmetiche e logiche
- AND, OR, NOT, A + B (somma aritmetica)
- Composta da:
  - **Decoder a 2 bit**

- Unità logica per le 3 operazioni logiche
- Full-adder a 1 bit

## Circuiti Sequenziali e Memorie

- **Clock:** circuito che emette impulsi digitali di lunghezza e intervallo di tempo specifici

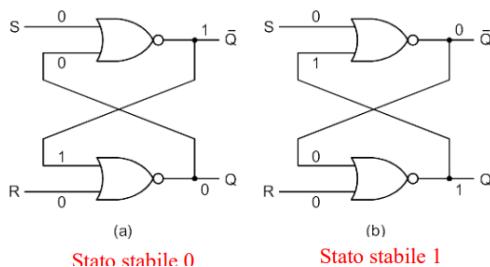


- **Tempo di ciclo di clock:** specifica l'intervallo di tempo tra i fronti corrispondenti in due impulsi successivi
- **Frequenza di clock:**
  - Specifica il numero di periodi di clock per unità di tempo
  - Inverso del tempo di ciclo di clock
  - **Hertz:** unità di misura

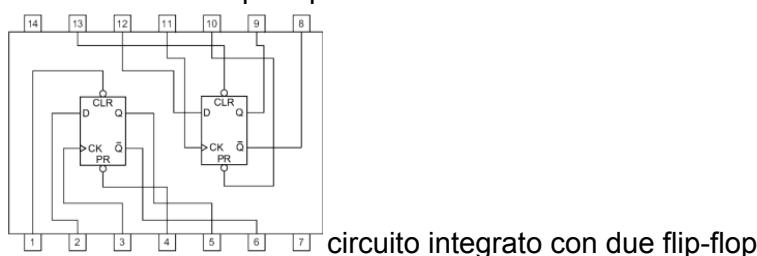
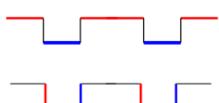
$$\text{frequenza di clock} = \frac{1}{\text{tempo di ciclo di clock}}$$

## Latch e Flip-Flop

- **Circuito Bistabile:**
  - Circuito in grado di mantenersi in posizione di stabilità in due differenti stati.
  - Il cambio di stato è provocato da un particolare segnale o impulso
- **Latch SR:**
  - Circuito bistabile realizzato con due porte NOR
  - Input **S(setting)**: usato per impostare lo stato 1
  - Input **R(reset)**: usato per impostare lo stato 0
  - Output **Q**
  - Se in stato 0, S diventa 1, passa in stato 1
  - Se in stato 1, R diventa 1, passa in stato 0
  - Ricorda qual è stato l'ultimo valore impostato, base per realizzazione RAM



- **Flip-Flop:**
  - Circuito molto simile al Latch
  - Unica differenza: istante in cui il segnale di clock determina cambio di stato
    - Latch: il cambio determinato dal **livello** del clock
    - Nel Flip-Flop dal **fronte** del clock



## RAM (Random Access Memory)

- Memoria il cui accesso può avvenire in modo casuale e non sequenziale
- Può essere:
  - Statica (**SRAM**)
  - Dinamica (**DRAM**)
- Memorie **volatili** (conservano valore solo se alimentate)

## ROM (Read Only Memory)

- Significa memoria di sola lettura
- Non può essere cancellata o riscritta
- Memoria non volatile (conserva valore anche se non alimentata)
- Usata per memorizzare permanentemente programmi stabili
- **PROM** (Programmable ROM):
  - Convenienti in fase di prototipazione
  - Possono essere scritti una sola volta con strumenti poco costosi
- **EPROM** (Erasable PROM):
  - Particolare tipo di PROM
  - Scritta come PROM
  - Contenuto può essere cancellato con **luce ultravioletta**
- **EEPROM** (Electrically EPROM):
  - Particolare tipo di EPROM
  - Può essere cancellata **elettricamente** (+ veloce)
  - Ri-scrittura può essere fatta **in-circuit**, senza smontare chip dal circuito
- **FLASH**:
  - Particolare tipo di EEPROM
  - Utilizzata da dispositivi mobili (chiavette)
  - Tempi di accesso buoni ma tempo di vita “limitato”
  - Vendute sotto forma di chip o memory card
  - Tipi di flash memory card: **SSD**, USB memory flash

## RAM Statica (SRAM)

- Realizzate con circuiti simili a Flip-Flop tipo D
- Molto veloci ma costose
- Utilizzate nei calcolatori odierni per realizzare registri e cache
- Per realizzare una SRAM 32K x 8 quanti flip-flop servono?
  - 32K locazioni (32.768) e ciascuna locazione 8 bit
  - $32.768 \times 8 = 262.144$  bit
  - Un flip-flop memorizza 1 bit, quindi servono 262.144 flip-flop

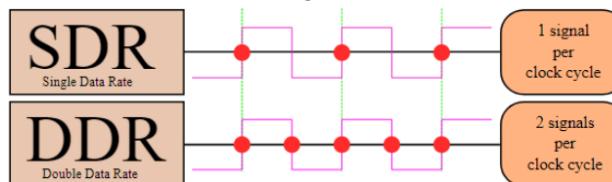
## RAM Dinamica (DRAM)

- Non realizzate con flip-flop ma con **array di celle** ognuna costituita da un transistor e un condensatore
- Più economiche e dense delle SRAM ma molto più lente
- DRAM **Asincrone**:
  - Comunicazione con CPU non sincronizzata da un segnale di clock
  - Necessarie linee di sincronizzazione addizionali

- DRAM Sincrone (**SDRAM**):
  - Un segnale di clock stabilisce le tempistiche nello scambio di dati
  - Per accesso a sequenze di byte ad indirizzi contigui si usa modalità burst
  - Svantaggio: deve avvenire tutto in multipli di clock

## DDR (Double Data Rate SDRAM)

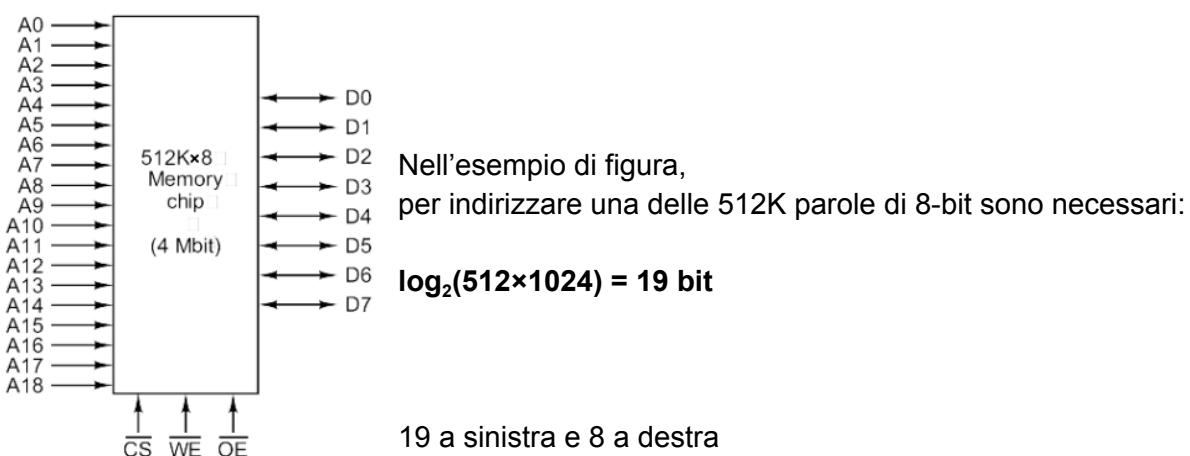
- Memorie più utilizzate nei PC e server
- Variante delle SDRAM
- Può inviare dati alla CPU **due volte** ogni ciclo di clock



- Previsti 2 canali di accesso paralleli 64 bit (128 bit)
- Elevata **larghezza di banda**:
  - Opera con frequenza di bus pari a N MHz
  - Larghezza di banda =  $N \times 2$  (trasf/ciclo)  $\times 8$  (ampiezza bus)
  - Ulteriormente raddoppiato se ha 2 canali di accesso paralleli
- **Latenza**: tempo necessario per reperire la prima parola
- 5 generazioni: DDR1, DDR2, ..., DDR5

## Organizzazione Chip di Memoria

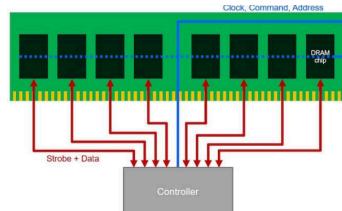
- Memoria organizzata in Chip, ognuno caratterizzato da:
  - **Lunghezza di parola**
    - Specifica dimensione in bit di ogni unità di informazione
    - Tipiche lunghezze: 1, 4, 8 e 16 bit
  - **Numero di parole**
    - Indica quante parole sono contenute nel Chip
- La dimensione in bit di un Chip di memoria si ottiene con:
  - **numero parole x lunghezza parola**
- Per legge di Moore, Chip di memoria aumenta di capacità di **due volte ogni 18 mesi**
- Tipica capacità di un Chip di SDRAM è 8-16 Gigabit
- Dotato di piedini di I/O per collegamento a CPU e BUS
  - Se lunghezza della parola è N, ci sono **N PIN** che leggono e scrivono dati
  - Per selezionare la parola, necessari  **$\log_2(N)$  PIN** che specificano l'indirizzo
- Organizzazione di un Chip di RAM da 4 Mbit:



- La RAM è costituita da diversi chip, è necessario abilitare o meno un certo Chip:
  - Input **CS**(Chip Select) serve a questo
  - Input **WE**(Write Enable): indica alla RAM se vogliamo leggere o scrivere
  - Input **OE**(Output Enable): controlla segnali di output
- Come realizzare una memoria  $512K \times 32$  utilizzando Chip  $512K \times 8$  ?
  - Si utilizzano 4 Chip  $512K \times 8$
  - Ciascuna parola di memoria è suddivisa in 4 parti e ogni parte è memorizzata da un diverso chip (allo stesso indirizzo).

## DIMM-DDR

- Organizzazione delle DIMM con Chip di memoria DDR ha bus dati 64 bit
- Su ciascuna DIMM sono montati 8 Chip che operano in parallelo
- Ogni Chip collegato con controller della memoria



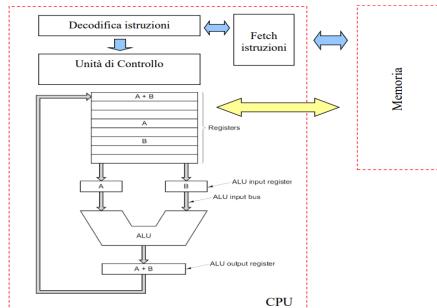
## Memoria Cache

- DRAM è più lenta della CPU, quindi leggere dalla memoria causa rallentamento
- Anche se banda di trasferimento con DDR è elevata, il problema rimane la **latenza**
- Poichè le SRAM sono molto più veloci ma anche costose, viene introdotto il meccanismo della **cache**, una SRAM veloce e dimensione limitata tra CPU e RAM
- Funzionamento:
  - CPU reperisce sempre dati dalla cache
  - Se la parola desiderata è presente nella cache (**cache hit**), grande vantaggio di tempo
  - Se non è presente (**cache miss**), bisogna trasferirla da DRAM a cache, tempo totale maggiore rispetto a lettura da DRAM. Se cache piena, si applica **politica di rimpiazzamento** per caricare nuovo dato al posto di un altro
- La copia di dati tra memoria principale e cache avviene per blocchi denominati **linee di cache**
- Tecniche di gestione della cache basate sul principio di:
  - **Località spaziale:**
    - Alta probabilità di accedere in un breve tempo a celle di memoria con **indirizzo vicino**
    - Politiche di allocazione leggono più dati di quelli necessari
  - **Località temporale:**
    - Alta probabilità di accedere in un breve tempo ad una cella di memoria alla quale si è **appena avuto accesso**
    - Politiche di rimpiazzamento decidono quale linea di cache rimpiazzare
    - Politica più utilizzata: **LRU**, rimpiazzato il blocco utilizzato meno recentemente
- Calcolatori moderni utilizzano più cache:
  - Per ottimizzare il rapporto costi/prestazioni si usano più livelli di cache
  - Cache diverse utilizzate per istruzioni e dati

Come fa una cache a capire se un dato è presente?

- Nelle **cache a mappatura diretta**:
    - Una linea quando presente può essere caricata in una sola posizione
    - Svantaggio:
      - Non consente l'implementazione di politiche di rimpiazzamento efficaci
  - Nelle **cache associative a n-vie**:
    - Ciascuna linea può essere ospitata in n posizioni diverse
    - Garantisce migliori prestazioni potendo applicare LRU

# Chip della CPU



- CPU legge istruzioni assembly da eseguire dalla memoria (Fetch)
  - Istruzioni decodificate e passate all'unità di controllo che può essere:
    - Microprogrammata (CPU CISC)
    - Cablata (CPU RISC)
  - CPU contiene ALU e registri che le permettono di eseguire compiti elementari
  - CPU moderne realizzate all'interno di un Chip ULSI dotato di PIN per collegamento con mondo esterno

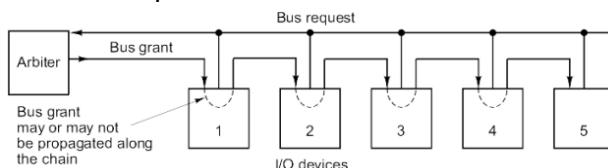
# BUS dei Calcolatori

- Collegamento elettrico comune tra più dispositivi
  - Possono essere:
    - **Interni alla CPU** per trasportare dati da registri alla ALU
    - **Esterini alla CPU** per collegare CPU alla memoria e periferiche
  - I BUS esterni devono attenersi a delle specifiche chiamate **protocollo del BUS**
  - BUS più diffusi: PCI-Express(PC), Bus SCSI(per periferiche), USB ...
  - Dispositivi possono essere:
    - Attivi (**Master**): possono iniziare trasferimento
    - Passivi (**Slave**): possono solo rimanere in attesa di richieste
    - Esempio: Fetch di istruzioni, CPU attiva, Memoria passiva
  - **Larghezza del Bus:**
    - Uno dei parametri più importanti per la progettazione del Bus
    - Bus molto ampio richiede molti collegamenti, spazio sulla scheda madre e connettori più grandi
  - Possono essere:
    - **Sincroni:**
      - Una delle linee pilotata da un segnale di clock
      - Difetto: tutte le operazioni hanno durata multipla del periodo di clock
      - Se Bus opera con dispositivi lenti e veloci, si adatta a quelli lenti
      - Vantaggio: semplicità di progettazione

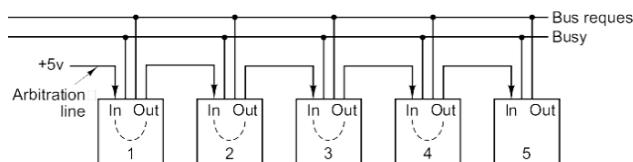
- **Asincroni:**
  - Non dotato di un clock principale, ma le parti si devono sincronizzare
  - Usati per Bus esterni (tipo USB)

## Arbitraggio del BUS

- Politica di gestione del Bus che dopo richieste multiple del Bus lo assegna a un dispositivo per volta
- **Arbitraggio Centralizzato:**
  - Il più noto è il **daisy chaining**
  - Una linea di Grant è collegata in serie a tutti i dispositivi che possono richiedere di diventare Master
  - Una linea di richiesta, comune a tutti i dispositivi, è collegata al chip arbitro
  - Quando un dispositivo vuole diventare Master, manda un segnale sulla linea di richiesta. L'arbitro attiva la linea di Grant
  - Il dispositivo più vicino all'arbitro controlla per sapere se è stato lui il richiedente. In caso affermativo, non propaga al successivo il segnale di Grant. In caso negativo invece il segnale viene propagato e il secondo dispositivo, come il primo controlla ...



- **Arbitraggio Distribuito:**
  - Variante distribuita del daisy chaining
  - La linea di Grant, invece che all'arbitro è collegata allo stato logico 1.
  - Se un dispositivo necessita del BUS, controlla se il BUS è in quel momento occupato (linea BUSY) e se il proprio Input di Grant (IN) è attivo
  - Quando il dispositivo controlla il BUS, nega il proprio Output di Grant (OUT) e attiva la linea BUSY.



In entrambi il dispositivo più a sinistra ha priorità maggiore

## Il Bus PCI

- Introdotto da Intel nel '90
- Bus sincrono con 32 linee dati e opera a 33 MHz e larghezza di banda di 133 MB/s
- Successivamente 64 linee dati a 66 MHz
- Per limitare numero dei contatti, le linee dati sono comuni (64 bit)
- Protocollo del Bus prevede diversi tipi di transazioni (durano al minimo tre cicli)
- Arbitraggio del Bus è centralizzato; arbitro decide quale dispositivo tra richiedenti ha priorità maggiore

## USB (Universal Serial Bus)

- Nato negli anni '90 per risolvere problema del collegamento di periferiche lente
- Principali obiettivi dei progettisti:
  - Un solo tipo di cavo poco costoso e lungo
  - Alimentare i dispositivi
  - Collegare molti dispositivi allo stesso PC
  - Poder collegare/scollegare periferiche a PC acceso
- Protocollo di comunicazione:
  - Supporta 4 modalità di comunicazione:
    - Control
    - Isochronous
    - Bulk
    - Interrupt
  - Esistono 4 tipi di pacchetti:
    - Token
    - Data
    - Handshake
    - Special
- Standard USB-C:
  - 24 pin, simmetrico
  - Utilizza protocolli Thunderbolt 4 e USB 4

## FireWire, Thunderbolt e USB 4

- FireWire era un Bus per dispositivi ad alte prestazioni, analogo a USB ma più performante e costoso
- Abbandonato a favore di USB 3 e Thunderbolt
- Intel poi sviluppò protocollo Thunderbolt 3
- In seguito USB 4

## PCI Express

- Bus seriale con connessioni punto a punto ad alta velocità
- Sono possibili collegamenti simultanei tra coppie master-slave (**multi-master**)
- Prevede più canali indipendenti, anche combinati per aumentare banda
- Device che chiedono alta banda eseguono operazioni in parallelo
- Scambio di dati avviene attraverso protocollo basato su pacchetti

## Motherboard

- Scheda elettronica che contiene la circuiteria e collegamenti di interfaccia tra vari componenti interni principali del PC:
  - Socket CPU
  - Bus interno
  - Slot di espansione
  - Slot per inserimento dei moduli di RAM
  - Controllori dei dispositivi di I/O fondamentali: SATA, CD

- Chipset: insieme dei chip che permette collegamento della CPU alle periferiche

## Interfacciamento di Periferiche

- Molteplici obiettivi:
  - Si vuole evitare che una periferica, ad esempio un disco che deve trasferire in memoria 1 MB di dati tenga impegnata la CPU durante tutto il trasferimento, ovvero che richieda che ogni parola venga prima trasferita dal disco alla CPU e in seguito da questa alla memoria. La tecnica del DMA (Direct Memory Access) consente di mettere in diretto contatto dispositivi e memoria in modo che questi possano operare sul bus indipendentemente dalla CPU.
  - Si vuole evitare di mantenere la CPU costantemente impegnata nel monitorare periferiche che devono compiere un certo lavoro. Tramite la tecnica dell'Interrupt la CPU può dedicarsi ad altro ed essere interrotta solo al momento opportuno.
- DMA:
  - Tecnica che consente trasferimento di dati tra la memoria e periferiche
  - Non necessario intervento della CPU
  - Necessario circuito chiamato controllore DMA che ne controlla funzionamento
  - Ha al suo interno 4 registri:
    - 1° contiene l'indirizzo iniziale di memoria dal quale leggere/scrivere
    - 2° indica quanti byte devono essere trasferiti
    - 3° indica il numero di identificazione del dispositivo
    - 4° indica se dati vanno letti o scritti dal/sul dispositivo
- Interrupt:
  - Tecnica che permette a una periferica di inviare un segnale elettrico su una linea della CPU (INT) per segnalare un determinato evento
  - CPU può decidere di accettare o meno l'interruzione
  - Se ci sono più periferiche potrebbero esserci più interrupt, se ne occupavano chip appositi

# ARCHITETTURE A CONFRONTO

- IPS(Instruction Per Second): misura che serve a valutare le prestazioni di un sistema
- MIPS (Milion Instruction Per Second):

$$MIPS = \frac{\text{Frequenza del clock}}{10^6 \cdot CPI}$$

- Misura fortemente approssimata, non tiene conto di:
  - Ampiezza dei bus, della presenza di cache
  - Percentuali delle diverse istruzioni all'interno di programmi
- FLOPS (Floating point Per Second):
  - Misura significativa se si utilizza programma dove molte operazioni sono floating point
  - Si usano MFLOPS, GFLOPS, TFLOPS

## Classificazione dei Calcolatori

- Computer usa e getta: alcuni MIPS
- Sistemi embedded: 1000 MIPS
- Smartphone: 2000 GFLOPS
- Console: 10.000 GFLOPS (GPU)
- PC: 5000 GFLOPS (CPU)
- Server: 5000 TFLOPS (CPU + GPU)
- Cluster (raggruppamento di più server): 10.000 TFLOPS
- Supercalcolatore: 1 EFLOPS
- **Legge di Amdhal** afferma che il miglioramento delle prestazioni che si ottiene in un calcolatore accelerando un qualsiasi sottoinsieme del calcolatore è proporzionale alla percentuale di tempo per cui quel sottoinsieme è utilizzato.

## RISC vs CISC

- **CISC** (Complex Instruction Set): CPU x86
  - L'instruction set del calcolatore deve contenere quante più istruzioni possibili
  - Ognuna di queste richiede **più cicli di data path**
- **RISC** (Reduced Instruction Set): ARM
  - Ogni istruzione dell'instruction set deve essere eseguita in uno o **pochi cicli di data path**
  - Necessarie più istruzioni RISC per avere stesso ris. di un'istruzione CISC
  - Più veloce poiché non necessario interpretare le istruzioni

## Principi di Progettazione RISC

- Istruzioni ISA eseguite direttamente dall'hardware:
  - Unità di controllo della CPU è cablata e non microprogrammata
  - Eliminato livello di interpretazione, aumenta velocità delle istruzioni
- Ottimizzare velocità in cui istruzioni vengono mandate al primo stadio di esecuzione:
  - Ampio uso di pipelining e parallelismo
- Istruzioni **facilmente decodificabili**
- Solo istruzioni Load e Store dovrebbero contenere indirizzi di memoria
- Disporre di **molti registri** per ridurre tempo di accesso alla memoria

# Come Aumentare le Prestazioni

- Ridurre il numero di **cicli di data path** per le istruzioni ISA
- Aumentare la **frequenza di clock**
- **Parallelismo:** sovrapporre esecuzione delle istruzioni(pipelining)

## Parallelismo

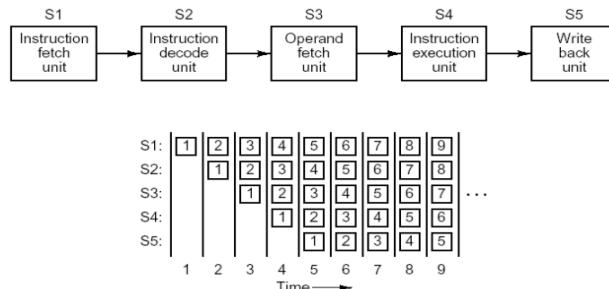
- **A livello d'istruzione:**
  - Più istruzioni eseguite contemporaneamente tramite pipelining
- **A livello di core:**
  - Più core/CPU cooperano per la soluzione dello stesso problema

## Aumento Frequenza di Clock

- Siamo arrivati a 4 GHz di frequenza e nei prossimi anni non aumenterà molto
- Limiti fisici:
  - Alte frequenze creano **disturbi** e aumentano **calore**
  - Ritardi nella propagazione del segnale
  - **Bus Skew** (segnali su linee diverse viaggiano a velocità diverse)

## Pipelining

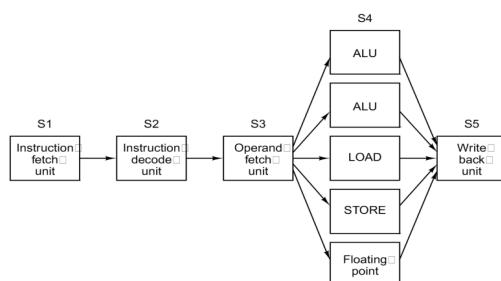
- Tecnica che suddivide l'esecuzione di ogni istruzione in **stadi**, gestiti da hardware dedicato



- Se il ciclo di clock della macchina è di 2 nsec sono necessari 10 nsec per completare l'esecuzione della prima istruzione (nessun risparmio rispetto all'assenza di pipelining), ma una volta riempita la pipeline, si completerà una istruzione ogni 2 nsec
- Possibile avere anche più di una pipeline

## Architetture Superscalari

- Una sola pipeline dotata di unità funzionali multiple



## Predizione di Salto

- Tecnica che a fronte di istruzioni di salto condizionale cercano di prevedere se programma salterà oppure no
- Ottimizza operazioni di pre-fetching
- **Predizione statica:**
  - Utilizzati criteri di buon senso derivati dalle abitudini dei programmati
- **Predizione dinamica:**
  - Mantenute statistiche sulla frequenza con cui i recenti salti condizionali sono stati eseguiti

## Esecuzione Fuori Ordine ed Esecuzione Speculativa

- **Esecuzione fuori ordine:**
  - Tecnica che consente di **saltare** temporaneamente istruzioni che hanno dipendenze (lasciandole in attesa) e di passare ad eseguire istruzioni successive non dipendenti.
- **Esecuzione speculativa:**
  - Tecnica che consiste nell'**anticipare** il più possibile l'esecuzione di alcune parti del codice prima ancora di sapere se queste serviranno.

## Meltdown

- Bug di sicurezza, coinvolse CPU moderne facenti uso di Cache ed esec. fuori ordine
- Permette di leggere il contenuto di tutta la memoria fisica eludendo i vincoli che non consentono a un processo di vedere la memoria al di fuori della porzione a lui dedicata.

## Architettura Core i7

- Esteriormente si presenta come una tradizionale CPU CISC a 32/64 bit
  - Molte istruzioni e complesse
  - Registri generali sono solo 8
- All'interno però contiene un nucleo RISC che fa uso di pipelining e architettura superscalare
- Sandy Bridge:
  - Architettura che contiene dai 2 ai 6 core, analizziamo questa
- Ogni core costituito da 4 parti principali:
  - **Sottosistema di memoria:**
    - Include cache privata L2 e una condivisa L3
  - **Front end:**
    - Preleva istruzioni da cache L1 e le decodifica in micro-operazioni
    - Micro-operazioni salvate nella **Micro-op cache**
  - **Controllo dell'esecuzione fuori sequenza:**
    - Trasferisce micro-operazioni allo schedulatore
    - Schedulatore può mandarle in esecuzione fuori ordine per ottimizzare prestazioni
  - **Unità esecutive:**
    - Eseguono operazioni tra interi, in virgola mobile...

## CPU per Sistemi Embedded

- ARM (Advanced RISC Machine)
  - Processori RISC a 32 bit sviluppata dell'azienda inglese ARM Holdings

# ARCHITETTURE PARALLELE

- Consiste nell'utilizzo in parallelo di più unità di calcolo
- Diversi tipi di parallelismo:
  - Nel chip, coprocessori, multi-processori, multi-computer...
- Fattori che caratterizzano un sistema parallelo:
  - Natura e numero degli **elementi di calcolo**
  - Natura e numero degli **elementi di memoria**
  - **Modalità di interconnessione:**
    - Statica: legami tra le CPU sono determinati a priori e sono fissi
    - Dinamica: legami tra le CPU sono definiti in base alle necessità da switch in grado di instradare i messaggi
    - Sistemi debolmente accoppiati: poche CPU, grandi e connessioni a basse velocità
    - Sistemi fortemente accoppiati: parallelismo realizzato con componenti più piccole e che interagiscono fortemente tra loro
  - **Parallelismo course-grained:** elemento software che viene parallelizzato è grande (es. programma)
  - **Parallelismo fine-grained:** elemento software che viene parallelizzato è piccolo (singola operazione)

## Parallelismo nel Chip

- **Parallelismo a livello di istruzioni:** Pipelining e Architetture SuperScalari.
- **Multi-threading:**
  - CPU esegue contemporaneamente due thread (parti di programma) come se esistessero due CPU virtuali.
  - Se uno dei due deve attendere per un cache-miss l'altro può continuare l'esecuzione senza lasciare la CPU in attesa. È il caso dell'Hyper Threading
- **Multi-core**
- **Più core eterogenei nel chip:** ingloba nello stesso chip più core ma con funzionalità specializzate.

## Coprocessori

- Processore indipendente che esegue compiti specializzati sotto il controllo del processore principale
- **Processori di rete:**
  - Gestiscono ad alta velocità lo smistamento di pacchetti che viaggiano in rete
  - Scheda di rete per esempio
- **Crittoprocessori:**
  - cifrano/decifrano molto velocemente flussi di dati
- **Processori grafici (GPU):**
  - Nelle schede grafiche processano grandi quantità di dati video e grafica 3D

## Multiprocessori

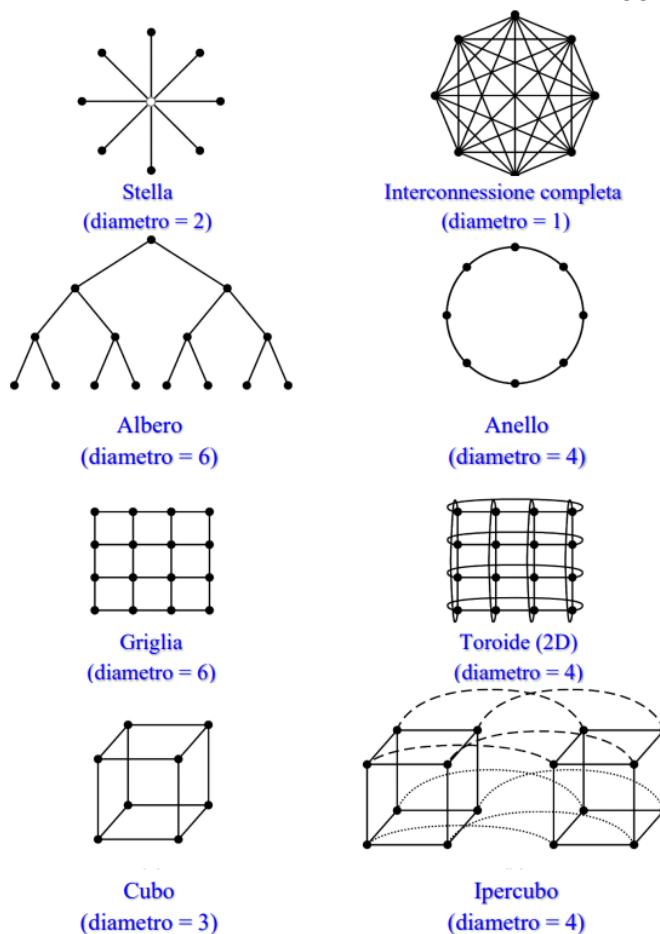
- Sistemi a memoria condivisa ossia le CPU condividono una memoria fisica
- Qualsiasi processo può leggere o scrivere tutta la memoria
- Sono fortemente accoppiati
- Una sola copia del sistema operativo in esecuzione

## Multicomputer

- Hanno più CPU ognuna dotata di una propria memoria (memoria distribuita)
- Sono debolmente accoppiati
- Complesso sistema di routing
- Su ogni CPU è in esecuzione una copia del sistema operativo
- Programmare un multicomputer è più complesso che programmare un multiprocessore, ma costruire un multiprocessore è più complesso e più costoso che costruire un multicomputer

## Multicomputer: Reti di Connessione

- Fanout:
  - Numero di archi collegati a un nodo
  - Determina la fault tollerance della rete
- Diametro:
  - Distanza tra i due nodi più distanti del grafo
  - Dà informazioni sul tempo di comunicazione (caso peggiore)



- **Toroide 3D:**
  - Una delle tipologie di rete più utilizzata nei supercomputer
  - Buon compromesso tra diametro e numero di connessioni
- **Fat tree:**
  - I rami più vicini alla radice offrono una maggiore larghezza di banda
- Un'architettura è detta **scalabile** se le sue prestazioni aumentano aumentando il numero di processori

## Tassonomia di Sistemi Paralleli

- Tassonomia più utilizzata è quella di **Flynn** che si basa su due concetti principali:
  - **Sequenza di istruzioni:** insieme di istruzioni associate a un program-counter
  - **Sequenza di dati:** insieme di operandi

Sequenze di istruzioni	Sequenze di dati	Nome	Esempi
1	1	SISD	Macchina di Von Neumann
1	Molte	SIMD	
Molte	1	MISD	
Molte	Molte	MIMD	Multiprocessore, Multicomputer

## Computer Più Potenti

- Fugaku (Giappone):
  - 442 Petaflops
  - Topologia di rete: Tofu, variante del toroide 3D
- Leonardo (Bologna, CINECA):
  - 175 PetaFlops
  - Topologia di rete: Dragonfly, che utilizza Fat Tree
- Frontier (USA):
  - 1° al mondo
  - 1.1 ExaFlops
  - Topologia di rete: Dragonfly

## Cluster di Computer

- Costituito da PC collegati tra loro tramite schede di rete
- Cluster centralizzati: PC vicini tra loro, montati in armadi
- Cluster decentralizzati: PC lontani collegati da LAN
- Tre tipi di cluster:
  - Fail-over Cluster:
    - Funzionamento delle macchine continuamente monitorato
    - Se uno smette di funzionare un'altra macchina subentra
    - Scopo: garantire un servizio continuativo
  - Cluster con load-balancing:
    - Sistema dove richieste di lavoro sono inviate alla macchina meno impegnata
  - HPC Cluster: prestazioni estremamente alte

# FORMULE ESAME

## Esercizi Ultima Parte

- Miglioramento prestazioni:

$$T_F = \rho \cdot \frac{T_i}{\alpha} + (1 - \rho) \cdot T_i$$

- Prestazioni di due CPU con frequenze:

$$T_{ES} = T_{ci} \sum_{i=1}^n N_i \cdot CPI_i \quad T_{ES} \approx T_{CL} \cdot N_{TOT} \cdot \widetilde{CPI}$$

$$T_{CL} = \frac{1}{F}$$

- Memorie Cache:

$$T_{TOT} = N_{TOT} \cdot T_1 + (N_{TOT} - N_1) \cdot T_2 + (N_T - N_1 - N_2) + T_R(N_T - N_1 - N_2)$$

## Esercizi Bit

- Se ho  $n$  bit rappresento  $2^n$  combinazioni, (con 4 bit sono codificabili 16 valori)
- Per codificare  $x$  numeri serve la parte intera superiore di  $\log_2(x)$  bit
- In complemento a due sempre  $2^n$  combinazioni.  $[-2^{n-1}; 2^{n-1}-1]$

## Algebra di Boole

$$(\bar{A}C + A) = (A + C)$$

Name	AND form	OR form
Identity law	$1A = A$	$0 + A = A$
Null law	$0A = 0$	$1 + A = 1$
Idempotent law	$AA = A$	$A + A = A$
Inverse law	$A\bar{A} = 0$	$A + \bar{A} = 1$
Commutative law	$AB = BA$	$A + B = B + A$
Associative law	$(AB)C = A(BC)$	$(A + B) + C = A + (B + C)$
Distributive law	$A + BC = (A + B)(A + C)$	$A(B + C) = AB + AC$
Absorption law	$A(A + B) = A$	$A + AB = A$
De Morgan's law	$\bar{AB} = \bar{A} + \bar{B}$	$\bar{A + B} = \bar{A}\bar{B}$