

Organizzazione degli Elaboratori

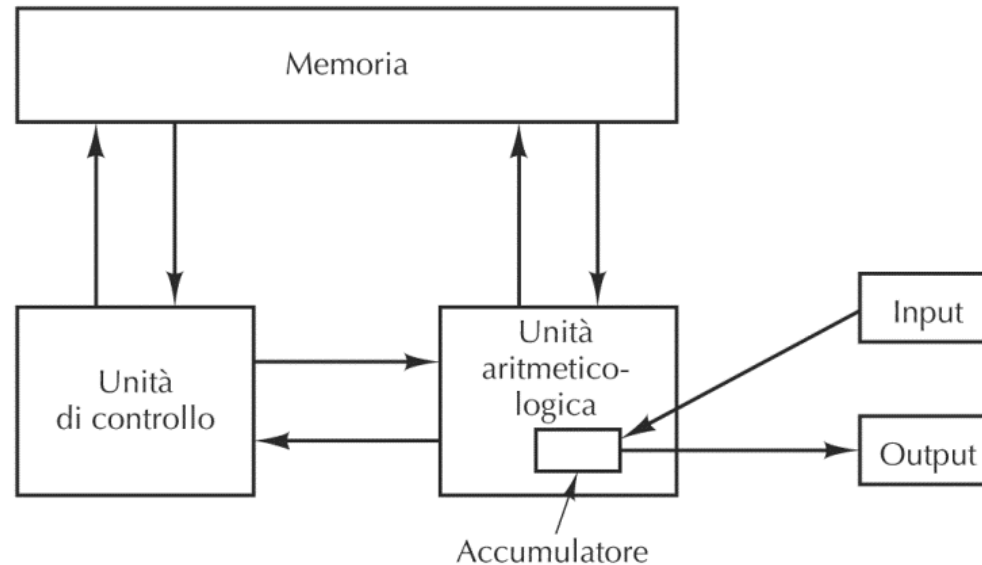


Figura 1.5 La macchina originale di von Neumann.

Prof. Ivan Lanese

Entriamo dentro all'architettura di un tipico calcolatore moderno

- Un calcolatore è un sistema composto da processori, memorie e dispositivi di input/output (I/O)
- Questa organizzazione (con l'unica differenza del bus) è uguale a quella della macchina di Von Neumann
- **Organizzazione "bus oriented":** Un bus è un insieme di connessioni elettriche (fili) parallele utilizzati per trasportare informazioni da un componente all'altro

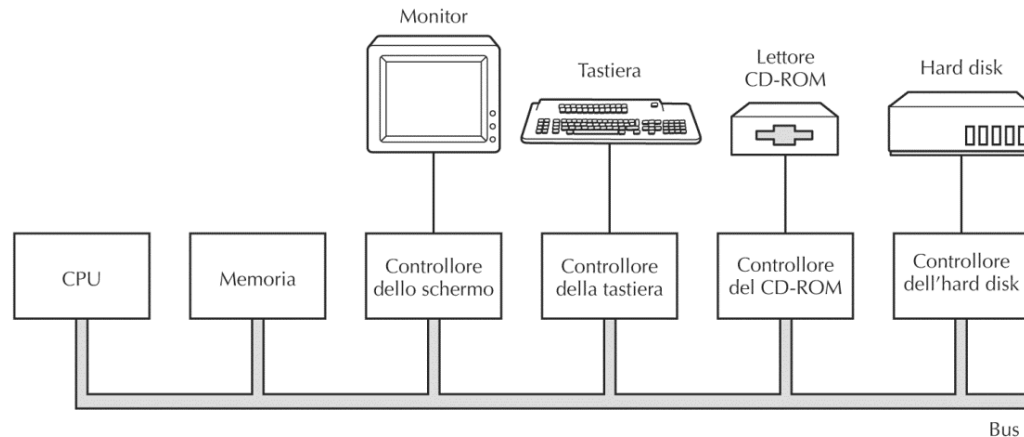
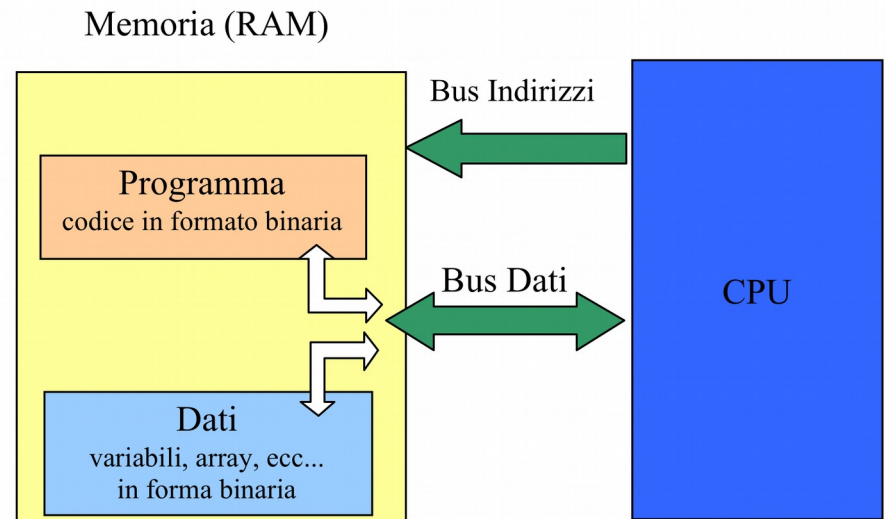


Figura 2.30 Struttura logica di un semplice personal computer.

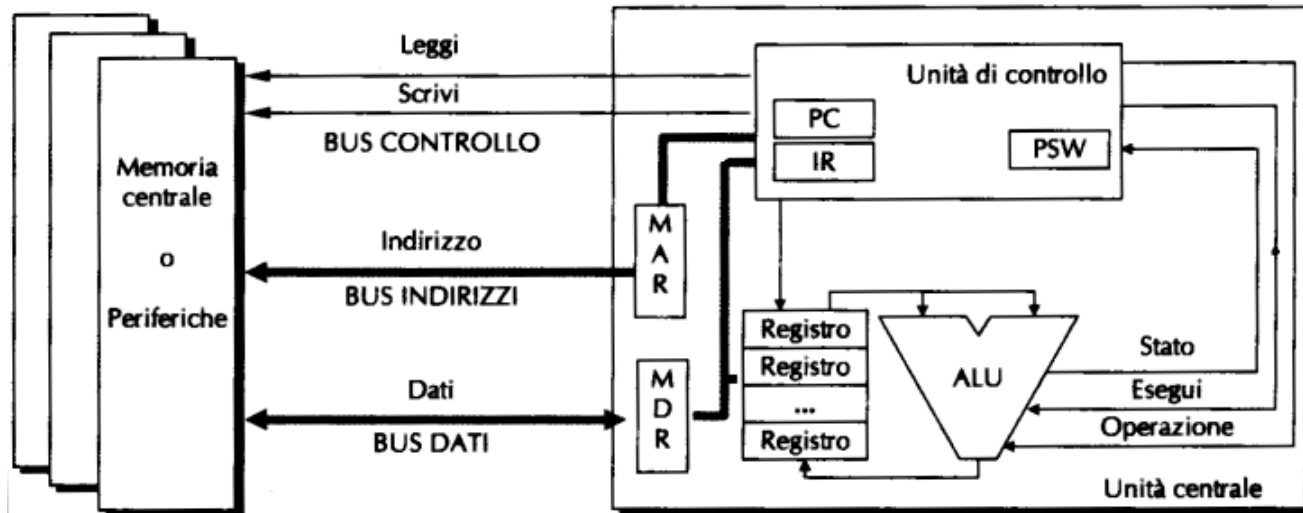
Architettura di Von Neumann

- Usa la memoria non solo per i dati ma anche per i programmi. Quando fu proposta era rivoluzionaria: evitava complesse configurazioni con interruttori e cavi
- Programmi e dati sono trasferiti entrambi attraverso il (sotto)bus dati. Il (sotto)bus indirizzi è utilizzato dalla CPU per indicare alla memoria le locazioni delle informazioni da trasferire



La CPU

- Esegue i programmi nella memoria centrale
- Una CPU è composta da:
 - Unità di controllo: legge e interpreta le istruzioni
 - ALU: esegue le operazioni (AND, OR, addizione, ...)
 - Registri: memorizzano i risultati temporanei e le informazioni necessarie al funzionamento



■ Alcuni registri speciali:

- Program Counter (PC): indica la prossima istruzione in memoria
- Instruction Register (IR): memorizza l'istruzione che si sta per eseguire
- Memory Address Register (MAR): indirizzo della cella di memoria da usare nella prossima lettura/scrittura
- Memory Data Register (MDR): registro che accede al Bus dei Dati (sia in lettura che in scrittura)
- Program Status Word (PSW): indica informazioni sull'ultima operazione eseguita (zero, overflow, ...)

La CPU – esecuzione di una (tipica) istruzione

1. contenuto di PC posto su MAR e attivazione linea Leggi
2. il contenuto in memoria all'indirizzo indicato da MAR viene scritto su MDR attraverso il bus dati
3. contenuto di MDR copiato su IR e relativa decodifica
4. l'istruzione passa in esecuzione sulla ALU
5. se ci sono operandi da prelevare in memoria, si collocano in registri (usando come sopra MAR e MDR)
6. terminata l'esecuzione il risultato va su registro destinazione; se bisogna scrivere in memoria il valore calcolato si usano MAR / MDR attivando linea Scrivi
7. Si ritorna al punto 1 dopo aver aggiornato il valore di PC

Fetch – Decode – Execute (FDE)

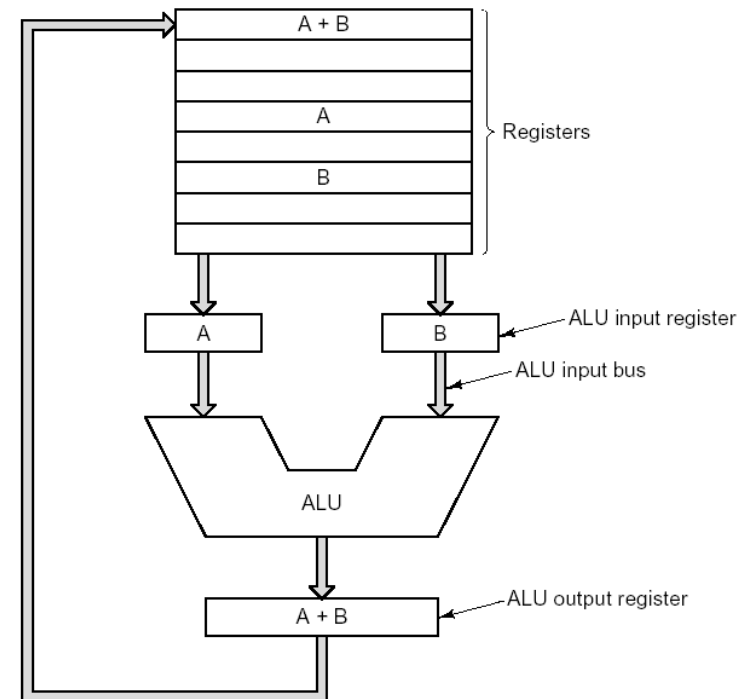
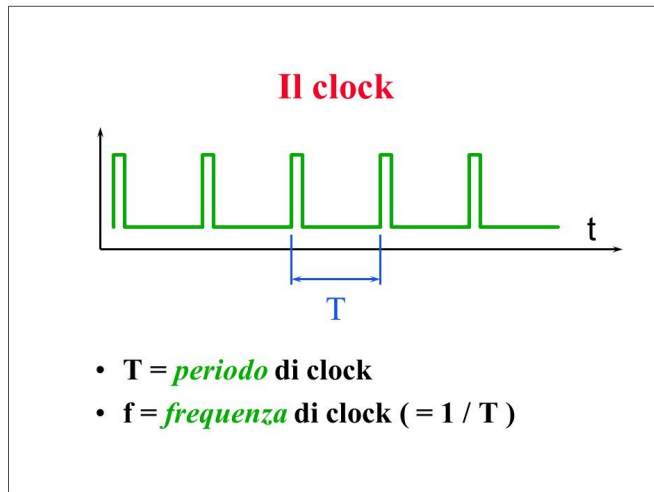
- Il ciclo di esecuzione descritto nella slide precedente è conosciuto come ciclo "Fetch - Decode - Execute":
 - **Caricamento (Fetch)**: acquisizione dalla memoria di un'istruzione del programma
 - **Decodifica (Decode)**: identificazione del tipo di operazione da eseguire
 - **Esecuzione (Execute)**: effettuazione delle operazioni corrispondenti all'istruzione
- In particolare:
 - i passi 1-2 corrispondono a Fetch,
 - il passo 3 a Decode,
 - i passi 4-5-6 ad Execute

Facciamo uno zoom sull'Unità di Controllo

- Il modo più semplice per realizzare l'unità di controllo consiste nel costruirla come un circuito hardware compatibile con un insieme fissato di istruzioni
- Ciò diventa però complicato se l'insieme delle istruzioni è troppo grande; in tal caso, l'unità di controllo risulterebbe troppo costosa
- Una possibile risposta è permettere una (micro)programmazione del comportamento della CPU:
 - Si possono eseguire più istruzioni con i medesimi componenti
 - Cambiando il microprogramma cambia il comportamento della CPU (correzione errori, aggiunta di nuove istruzioni,...)

Facciamo uno zoom sul modo di operare della ALU

- Con il termine Data Path (percorso dati) intendiamo la parte della CPU comprendente ALU, i suoi input ed i suoi output (solitamente registri)
- Il flusso da registri a ALU, scrittura dell'output register, e riscrittura in registro del risultato avviene durante il "ciclo di data path" governato da un "clock"



Velocità della CPU

- durata ciclo di data path =
durata ciclo di clock =
 $1 / F$
(dove F è la frequenza di lavoro della CPU; la frequenza viene misurata in Hertz - Hz - che indica il numero di cicli in un secondo)
- durata istruzione ISA (Instruction Set Architecture) =
 $n \times$ durata ciclo di data path
(con n variabile da istruzione a istruzione a seconda della specifica architettura)
- Istruzioni ISA per secondo =
 $1 /$ durata istruzione ISA =
 F / n
(assumendo n cicli per istruzione)

Evoluzione delle frequenza di lavoro delle principali CPU Intel

Chip	Data	MHz	N. transistor	Memoria	Descrizione
4004	4/1971	0,108	2300	640	Primo microprocessore su un solo chip
8008	4/1972	0,108	3500	16 KB	Primo microprocessore a 8 bit
8080	4/1974	2	6000	64 KB	Prima CPU di uso generale su un solo chip
8086	6/1978	5-10	29.000	1 MB	Prima CPU a 16 bit su un solo chip
8088	6/1979	5-8	29.000	1 MB	Usato nel PC IBM
80286	2/1982	8-12	134.000	16 MB	Introduzione della modalità protetta
80386	10/1985	16-33	275.000	4 GB	Prima CPU a 32 bit
80486	4/1989	25-100	1,2 M	4 GB	Memoria cache da 8 KB integrata
Pentium	3/1993	60-233	3,1 M	4 GB	Due pipeline; istruzioni MMX nei modelli successivi
Pentium Pro	3/1995	150-200	5,5 M	4 GB	Cache integrata a due livelli
Pentium II	5/1997	233-450	7,5 M	4 GB	Pentium Pro con istruzioni MMX
Pentium III	2/1999	650-1400	9,5 M	4 GB	Istruzioni SSE per la grafica 3D
Pentium 4	11/2000	1300-3800	42 M	4 GB	Hyperthreading; ulteriori istruzioni SSE
Core Duo	1/2006	1600-3200	152 M	2 GB	Due core in un singolo circuito stampato
Core	7/2006	1200-3200	410 M	64 GB	Architettura quad-core a 64 bit
Core i7	1/2011	1100-3300	1160 M	24 GB	Processore grafico integrato

Figura I.11 I membri principali della famiglia di CPU Intel. Le velocità del clock sono misurate in MHz (megahertz) dove 1 Mhz corrisponde a 1 milione di cicli/s.

- Uno degli aspetti principali dell'evoluzione dei computer è stato l'aumento della velocità di calcolo
 - Molto più rapida dell'evoluzione in altri campi (ad esempio trasporti)
- Il ciclo di clock dipende in gran parte da motivazioni fisiche
 - Ma anche dalla complessità delle operazioni da eseguire
- Fissato il ciclo di clock si può aumentare la velocità aumentando il parallelismo

RISC oppure CISC

- Alla fine degli anni '70, la microprogrammazione permise di realizzare elaboratori con istruzioni molto complesse:
CISC - Complex Instruction Set Computer
- In contrapposizione a questa tendenza nacque l'idea di realizzare architetture con set di istruzioni più semplici:
RISC - Reduced Instruction Set Computer
 - Istruzioni più semplici possono essere eseguite più velocemente (con un ciclo di clock ridotto)
 - Possibilmente evitando la microprogrammazione
- Ecco perché il livello che abbiamo chiamato "microarchitettura" può essere implementato tramite software (microprogrammazione) oppure hardware

Pipelining

- Un modo per migliorare le prestazioni di un processore è eseguire contemporaneamente più cicli FDE, usando per ognuno di essi parti diversi della CPU.
Ad esempio potremmo immaginare 5 sezioni indipendenti:

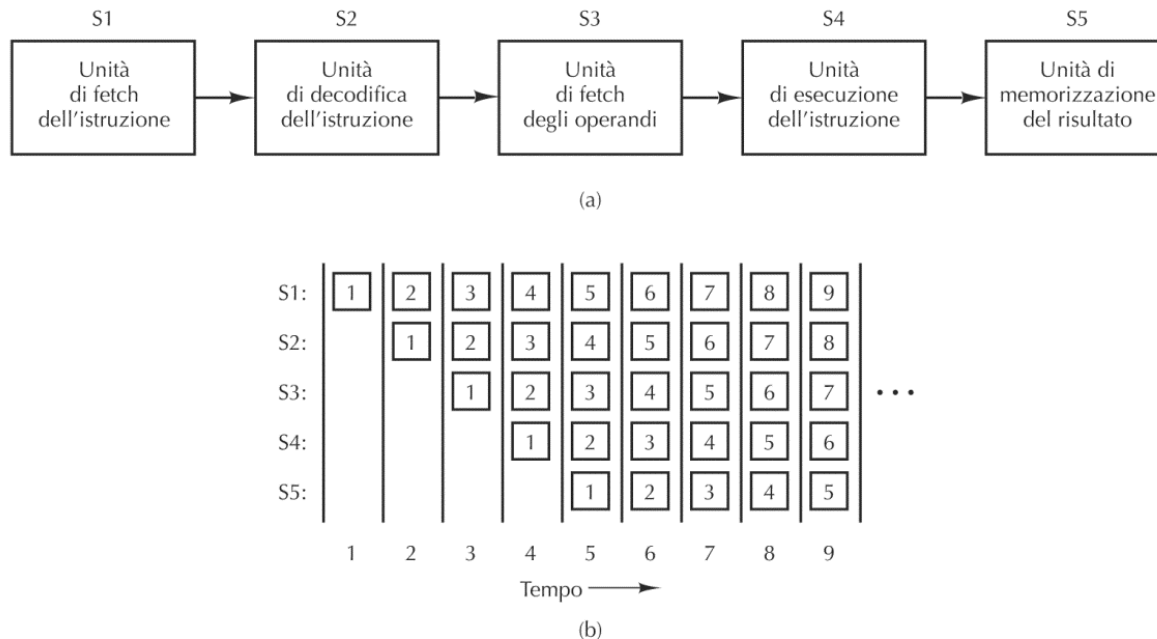


Figura 2.4 (a) Pipeline a cinque stadi. (b) Lo stato degli stadi in funzione del tempo. Sono mostrati nove cicli di clock.

- In alcuni casi, all'interno della medesima CPU si replicano unità di controllo e ALU per esecuzioni di attività in parallelo
 - Tali architetture sono dette multicore

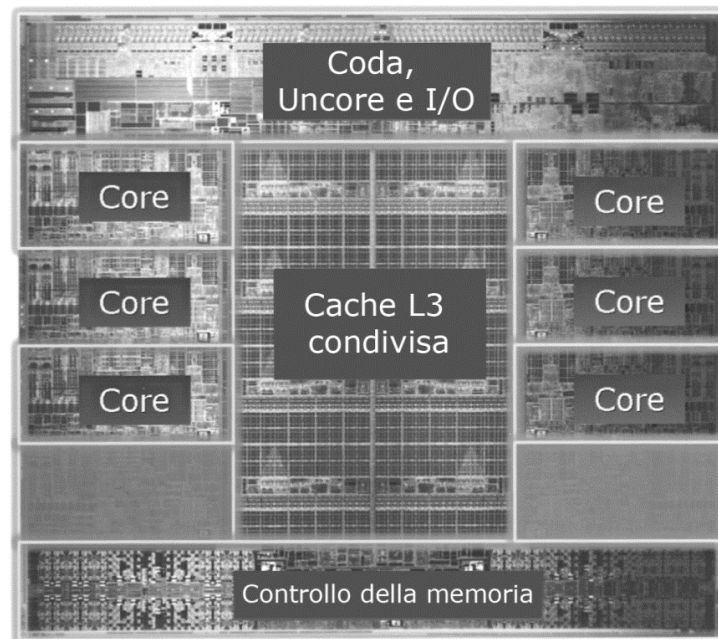
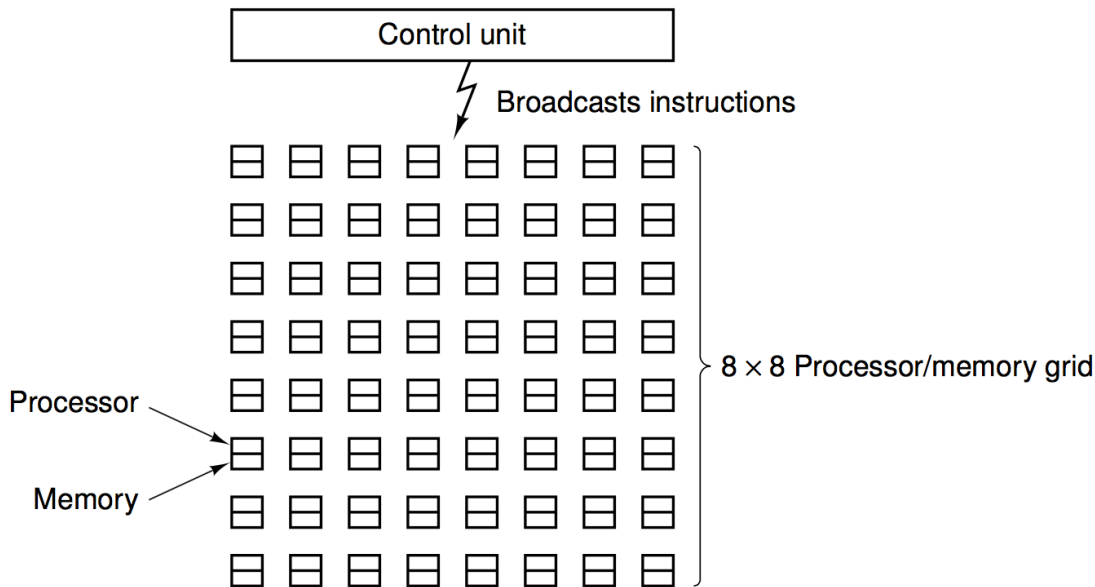


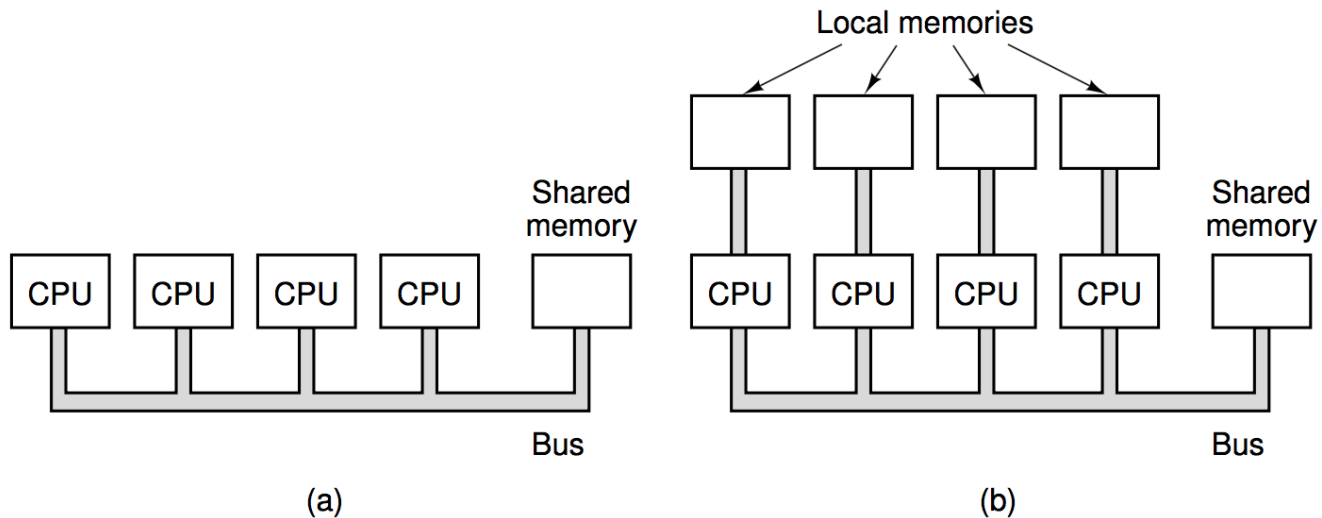
Figura 1.12 Il microprocessore Intel Core i7-3960X, © 2011 Intel Corporation. Il chip misura 21×21 mm² e contiene 2,27 miliardi di transistor (con “Uncore” si intendono le funzionalità esterne al core).

Parallelismo

- Ulteriori incrementi delle prestazioni si possono ottenere con molte CPU che lavorano in modo coordinato
- Un primo esempio è dato dagli "Array computer":
 - Processori eseguono la stessa istruzione su dati diversi
 - **SIMD**: single instruction-stream, multiple data-stream

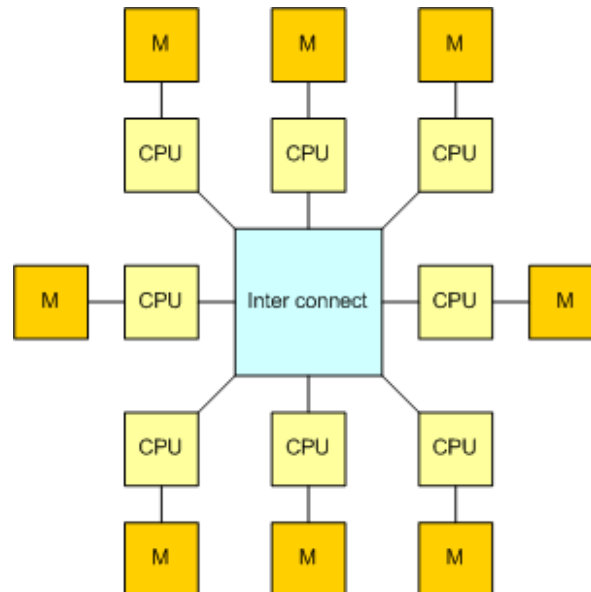


- Un altro esempio importante è quello dei multiprocessori
 - Molti processori anche diversi che condividono una memoria, senza eseguire necessariamente la stessa istruzione
 - **MIMD**: multiple instruction-stream, multiple data-stream.



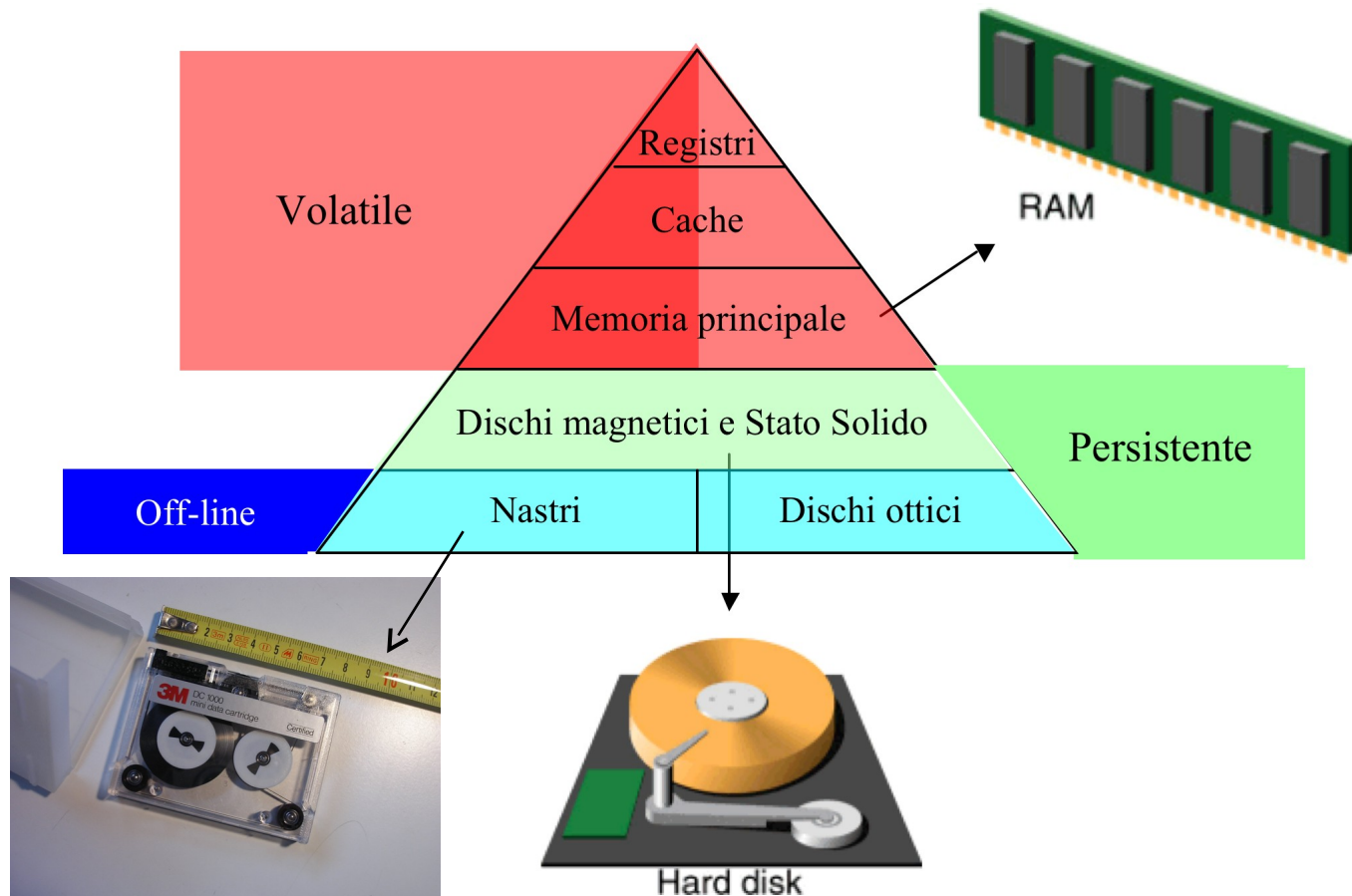
Parallelismo - continua

- Infine, si possono avere i multicomputer, la forma più estrema di parallelismo..
 - Molti processori che non condividono una memoria e che comunicano scambiandosi dei messaggi
 - Si può fare in modo che moltissime CPU cooperino



Le memorie

- Le memorie sono le componenti del calcolatore in grado di memorizzare le informazioni: dati, programmi e risultati indispensabili per il suo funzionamento



- Si usano vari tipi di memoria diversi per scopi diversi:
 - **Volatile:** l'informazione rimane memorizzata fino a che il calcolatore è alimentato
 - **Persistente:** l'informazione rimane memorizzata anche quando il calcolatore non è alimentato (spento)
 - **On-line:** i dati sono sempre accessibili
 - **Off-line:** il supporto deve essere montato per poter accedere ai dati
- Il costo di memorizzazione per byte cresce salendo la piramide
- La capacità (q.tà di byte) cresce scendendo la piramide

Organizzazione delle memorie

- Le memorie si organizzano in celle
 - Una cella è una sequenza di bit con un suo specifico indirizzo

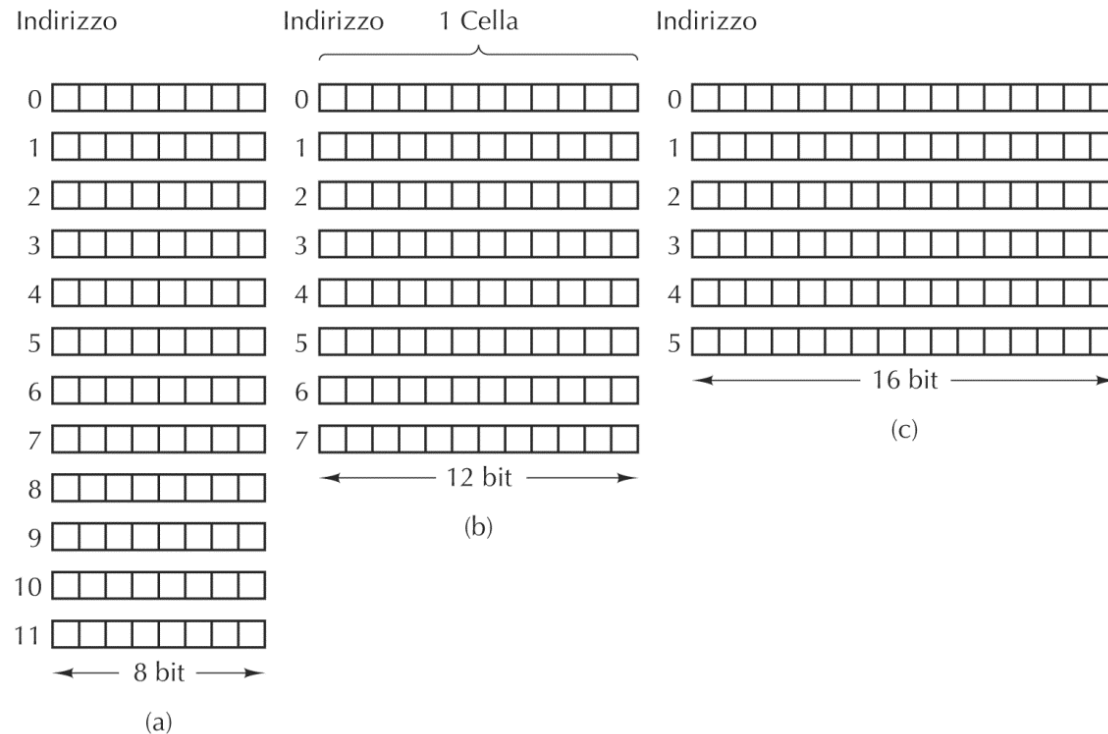


Figura 2.9 Tre modi di organizzare una memoria a 96 bit.

Perché byte... e quanto valgono i Kbyte, Mbyte,...

- Un bit è una informazione binaria (0 oppure 1)
- Negli anni '60, con l'avvento dell'IBM 360, emerse l'abitudine di organizzare le memorie in celle da 8 bit, i cosiddetti "byte"
- Il Kbyte coincide con la capacità di memorizzazione di una memoria con celle da 8 bit, indirizzabile con indirizzi a 10 bit

Nome	Significato
Bit	cifra binaria
Byte	8 bit
KByte (KB)	2^{10} ($\sim 10^3$) byte
MByte (MB)	2^{20} ($\sim 10^6$) byte
GByte (GB)	2^{30} ($\sim 10^9$) byte
TByte (TB)	2^{40} ($\sim 10^{12}$) byte

Ma a volte blocchi di 8 bit sono troppo piccoli...

- Molti calcolatori lavorano su blocchi di dimensione superiore ad 8 (ad esempio registri a 32 o 64 bit)
 - Tali blocchi vengono dette parole (in inglese "word")
- Esistono 2 modi per memorizzare le "word" in celle di memoria di dimensione standard (1 byte): big endian (indirizzi assegnati da sx a dx) e little endian (opposto)

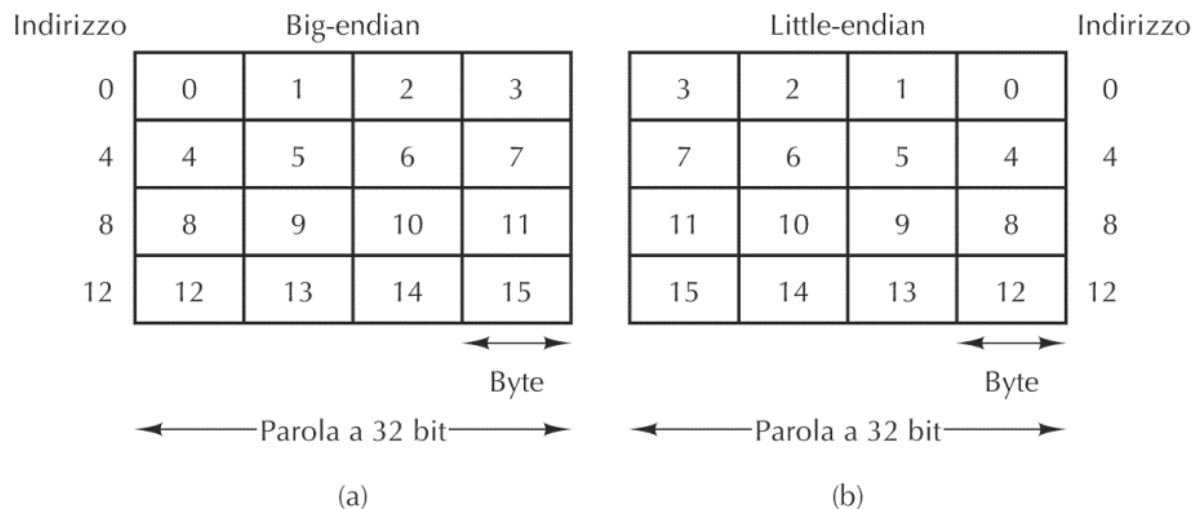


Figura 2.11 (a) Memoria big endian. (b) Memoria little endian.

Memoria cache

- La cache è una memoria poco capiente ma molto veloce
 - le parole di memoria usate più di frequente sono mantenute all'interno della cache
- Quando la CPU necessita di una parola, la cerca nella cache e se non è presente, la preleva dalla memoria
 - In tal caso il contenuto della memoria centrale viene trasferito nella cache

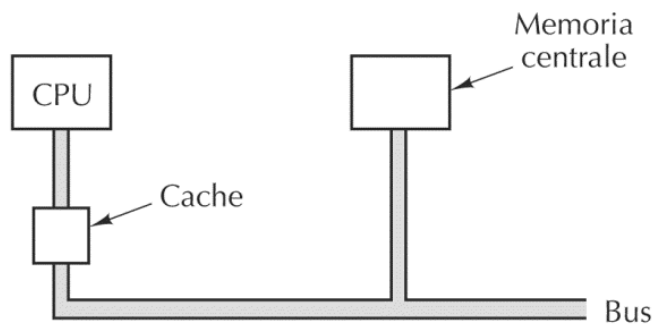


Figura 2.16 Da un punto di vista logico la cache si trova tra la CPU e la memoria centrale. Fisicamente può essere collocata in varie posizioni.

Perché usare una cache

- La maggior parte dei programmi esibiscono il cosiddetto principio di località
 - Riferimenti alla memoria eseguiti a breve distanza di tempo sono spesso anche in locazioni vicine
- Si sfrutta il principio di località mettendo in cache blocchi di locazioni contigue, aumentando così la probabilità che successivi accessi alla memoria possano essere all'interno del blocco già trasferito
 - il guadagno dato dal più veloce accesso a dati già in cache è solitamente superiore al tempo perso per trasferire i blocchi da memoria a cache, e viceversa

Quantificare l'impatto di una cache sulle prestazioni

■ Supponiamo che

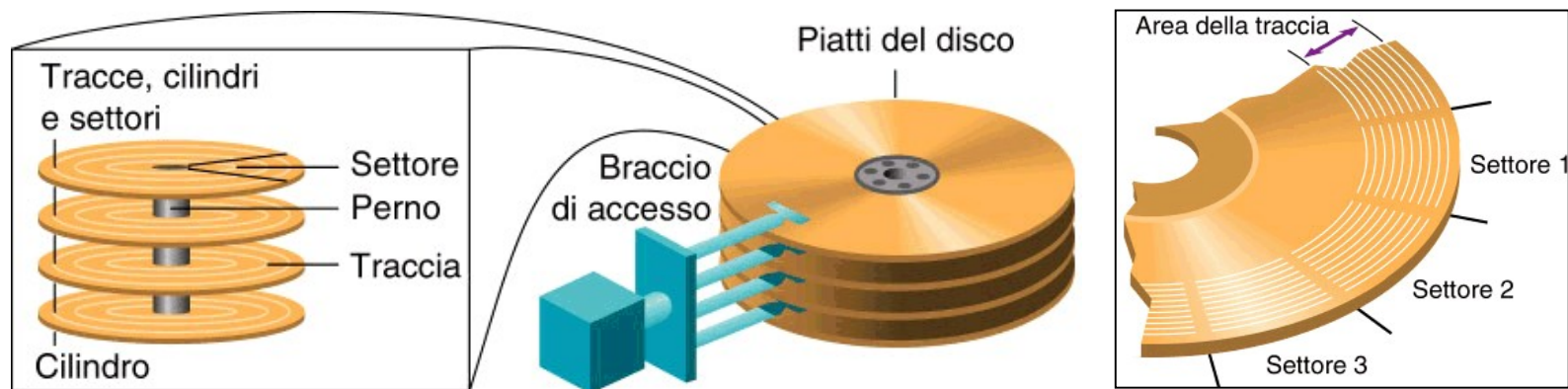
- c sia il tempo di accesso alla memoria cache
- m sia il tempo di accesso alla memoria centrale
- h sia l'hit-ratio, cioè la frazione di riferimenti che può essere soddisfatta dalla cache

allora il tempo medio di accesso t è $c+(1-h)m$

- ## ■
- Quando h si avvicina a 1, il tempo medio t si avvicina a c , mentre quando h si avvicina a 0, t si avvicina a $c+m$

Dischi magnetici

- Un hard disk (HD) è un dispositivo elettro-meccanico per la conservazione di informazioni sotto forma magnetica
 - **Testina:** magnetizza e legge lo stato di magnetizzazione della superficie del disco
 - **Traccia:** sequenza circolare di bit
 - **Settore:** porzione di traccia che contiene una quantità prefissata di bit (uguale per tutti i settori)



Memorie allo stato solido

- Si tratta di dispositivi completamente elettronici, senza nessuna parte in movimento.
 - La comune denominazione "disco allo stato solido" è pertanto inadeguata
 - Si diffusero velocemente per i portatili principalmente per i minori consumi e per la maggiore resistenza ad urti (non avendo parti meccaniche)
 - Più veloci ma meno capienti dei dischi rigidi (ordine dei 200 GB contro ordine dei TB)
- Tecnologia simile alle chiavette USB, ma più elaborata



RAID (Redundant Array of Inexpensive Disks)

- La differenza nelle prestazioni delle CPU e dei dischi magnetici aumenta continuamente
- Per ridurre tale gap, e per migliorare la affidabilità dei dischi, la tecnica RAID utilizza in parallelo più dischi
 - Più veloce perché si leggono più dati contemporaneamente, più affidabile se si replicano i dati e si aggiungono bit di controllo per verificare/correggere errori
- Esistono diversi modi per usare i vari dischi in parallelo

RAID (Redundant Array of Inexpensive Disks) - continua

- Non-redundant data striping
- Redundant data striping
- Data striping at bit level
- Bit-interleaved parity
- Block-interleaved parity
- Block-interleaved distributed parity

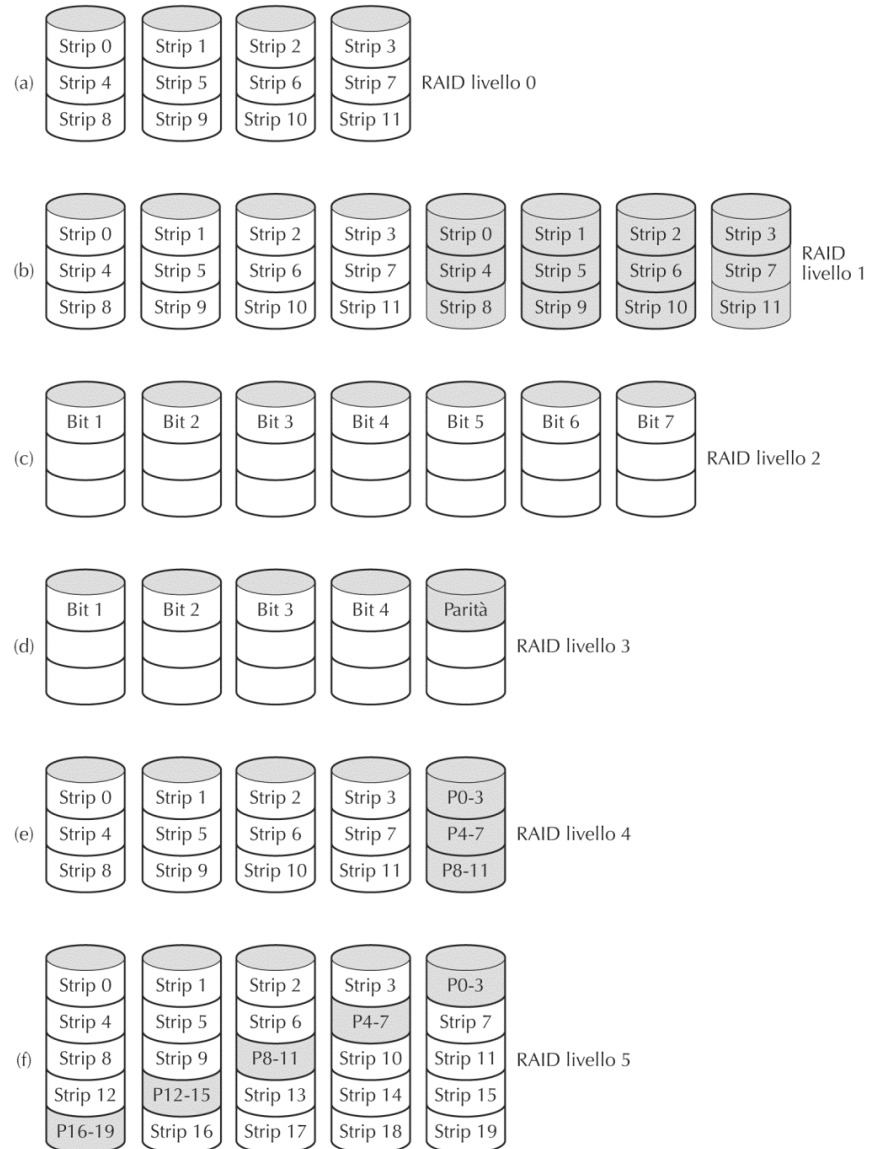


Figura 2.23 RAID dal livello 0 al livello 5. I dischi di backup e di parità sono in grigio.

- I CD (Compact Disc) utilizzano un principio ottico
 - Informazioni codificate tramite fori (Pit) alternati con zone piane (Land)
 - Un passaggio Pit-Land o Land-Pit codifica un 1
 - L'assenza di variazioni codifica lo 0
 - Le informazioni sono lette tramite un raggio laser che viene riflesso diversamente al passaggio su pit e land

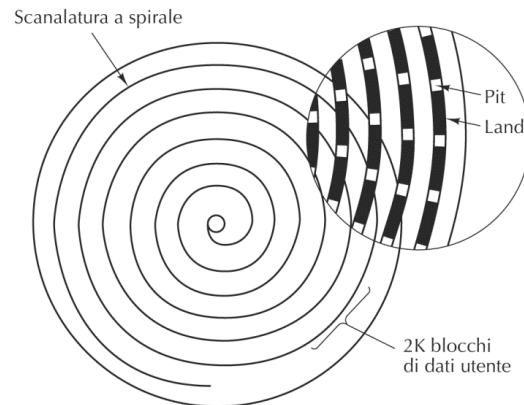


Figura 2.25 Struttura di registrazione di Compact Disc e CD-ROM.

Tipi di dischi ottici

- **CD-ROM:** scritti dal costruttore (Read Only Memory)
- **CD-R o CD Registrabili:** scrivibili una sola volta
- **CD-RW:** ri-scrivibili più volte
- **DVD:** Digital Video Disk o Digital Versatile Disk
- **Blu Ray:** laser blu e non rosso (diversa frequenza)

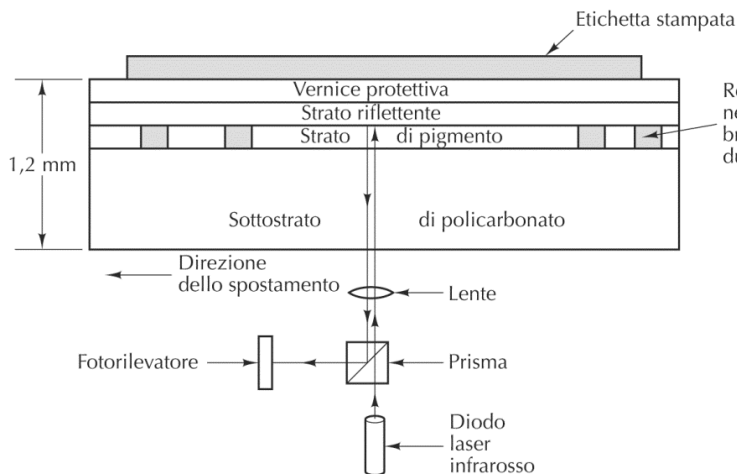


Figura 2.27 Spaccato di un disco CD-R e del laser (non in scala). La struttura di un CD-ROM è simile, fatta eccezione per la mancanza dello strato di pigmento e per la presenza di uno strato di alluminio con pit al posto dello strato riflettente.

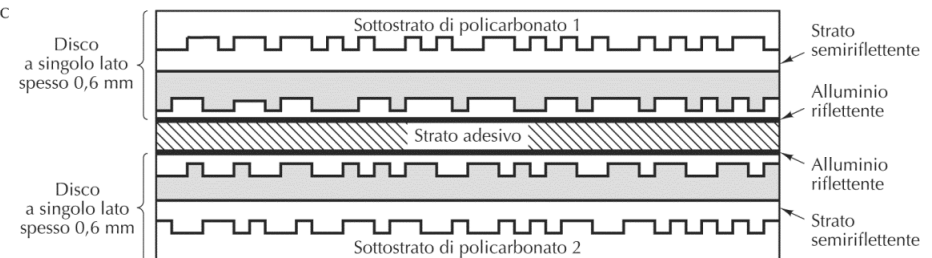


Figura 2.28 Disco DVD a doppio lato e doppio strato.

- Oltre le memorie, i calcolatori includono altri componenti quali, ad esempio:
 - Tastiera, mouse
 - Monitor: richiede una specifica scheda video che contiene un proprio processore detto GPU (Graphical Processing Unit)
 - Solitamente basate su architetture multicore estremamente avanzate
 - Programmabili (tramite specifici linguaggi, quali Cuda-C) per eseguire programmi non grafici
 - Stampanti
 - Schede di rete (wifi, bluetooth, ethernet, modem, ...)

Il calcolatore è quindi un sistema che combina varie componenti

- Come abbiamo già visto le componenti del calcolatore sono fra loro collegate tramite bus
 - I dispositivi si collegano al bus tramite un "controller"
 - Alcuni controller accedono direttamente alla memoria
 - Tecnica detta DMA (direct memory access)
 - Al termine della scrittura/lettura si invia un segnale (interrupt) alla CPU
 - L'accesso al bus condiviso viene gestito da un "arbitro" del bus
 - PCI-X, PCI, SCSI, ISA, ... sono tipi di bus

