

Architettura dei Calcolatori 1

Lezione 1 | 25 Febbraio

Università degli Studi di Modena e Reggio Emilia — Unimore

Nota

Il file 24-02.md era vuoto — questa è effettivamente la prima lezione con appunti.

1. Transistor e circuiti integrati

Ogni componente di un computer è formato da **transistor**: interruttori che fanno passare (o no) la corrente, trasformando un segnale elettrico in un segnale logico. Più transistor hai, più la rete logica è potente.

Legge di Moore

Il numero di transistor in un circuito integrato raddoppia circa ogni 2 anni.

2. Tipi di computer

2.1. Personal Computer (PC)

General purpose. Il trade-off classico tra costi e prestazioni.

2.2. Server

Da un piccolo server privato fino a interi edifici. Le prestazioni contano, ma conta di più l'**affidabilità**: deve essere sempre attivo. Se si rompe una parte, non deve impattare il servizio. Quindi: tanta **ridondanza**.

2.3. Supercomputer

Macchine con decine di migliaia di nodi di calcolo per il **High Performance Computing (HPC)**: simulazioni atmosferiche, chimica computazionale, medicina, fisica.

Nota

Esempio vicino a noi: **CINECA** a Bologna (Emilia-Romagna).

2.4. Embedded Computer

Sistemi **integrati** in dispositivi che di per sé non sono computer: auto, smartwatch, bracci robotici, droni.

Caratteristiche dominanti:

- Operano spesso a batteria → **consumo energetico** è il vincolo principale
- Memory footprint e prestazioni limitate
- Il software deve essere ottimizzato per il consumo

3. L'era Post-PC: PMD, Cloud e IoT

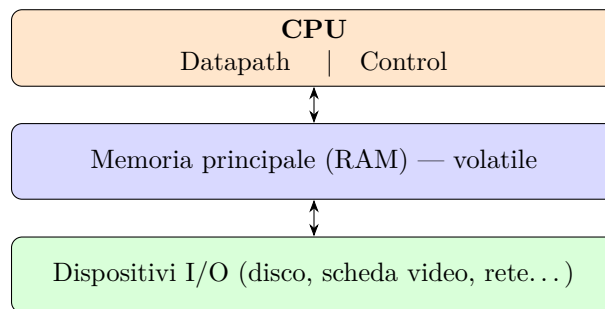
I computer che predominano oggi non sono più i PC, ma i **Personal Mobile Devices (PMD)**.

Paradigma	Caratteristiche
PMD	Funzionano a batteria, alta connettività (Wi-Fi, BT, rete mobile), costo contenuto
Cloud Computing	Il computing come servizio: non sai dove esegue il tuo programma
IoT	Internet di <i>oggetti</i> , non solo pagine web. Reti distribuite di sensori e attuatori

Compute Continuum

Una rete distribuita che parte da un sensore piccolo (embedded) e arriva fino al cloud. I computer sono **pervasivi e onnipresenti**: capire come funzionano è una conoscenza spendibile in qualsiasi contesto.

4. Cosa c'è dentro un computer?



4.1. CPU: Datapath + Control

La CPU **non lavora direttamente con la RAM**: prima carica i dati internamente, ci lavora, poi li riporta in RAM.

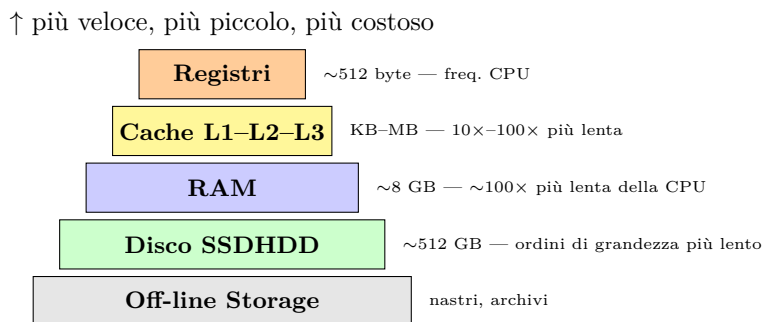
Definizione

- **Datapath**: le reti logiche che eseguono le operazioni (ALU, registri)
- **Control**: il “direttore di lavori” — dice al datapath cosa fare in ogni istante

4.2. Memoria volatile vs non volatile

Tipo	Volatile?	Uso
DRAM (RAM)	Sì — si azzerà allo spegnimento	Memoria di lavoro
Hard Disk / SSD	No — persistente	Memoria secondaria
Flash / Nastri	No	Storage off-line

5. La gerarchia di memoria



Perché la cache?

La RAM è circa 100× più lenta della CPU. La cache sfrutta il principio di **località**: i programmi tendono a riutilizzare gli stessi dati (o dati vicini) più volte. La cache porta in anticipo i dati vicini a quelli richiesti, dando l'**illusione** che la RAM sia veloce quasi come la CPU, a costo molto minore.

Attenzione

Accedere al disco è **LENTISSIMO**. Se il sistema è costretto a usare il disco come memoria virtuale (swap), i ritardi diventano mostruosi e l'esperienza utente ne risente.

Nota

Perché non fare tutta la memoria come registri? I registri sono carissimi e minuscoli: un processore tipico ha ~32 registri (qualche centinaio di byte). Aumentarli sarebbe proibitivo in costo e spazio sul die.

6. Dentro un sistema reale: scheda madre e SoC

6.1. Scheda madre (motherboard)

Componente	Ruolo
CPU / SoC	Il processore principale
Slot RAM	Banchi di memoria inseribili
Connettore SATA	Hard disk / SSD
Slot PCIe	Espansioni (uso tipico: GPU discreta)

6.2. Coffee Lake System-on-Chip (2018)

Un SoC moderno integra su un unico chip:

Componente	Note
2–8 core CPU	Multi-core, processo a 14 nm
Cache L3	Condivisa tra i core
GPU integrata	Per output grafico senza GPU discreta
Acceleratori media	Gestione video / codec

Le cache L1 e L2 sono *dentro* ogni singolo core — non visibili nel die shot di alto livello.

7. Caso studio: iPad Pro 11 (2018)

Smontato, un iPad mostra:

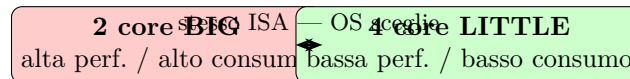
- **Schermo** → occupa la maggior parte dello spazio
- **Batterie** → grandi, per nutrire un dispositivo sempre connesso
- **PCB** (equivalente della motherboard) → sorprendentemente piccola

Sulla PCB tutto è **saldato**: SoC, RAM, storage. Non sostituibile.

7.1. A12 Bionic SoC

Componente	Note
4+2 core CPU	Paradigma big.LITTLE (ARM)
4 core GPU	Integrata nel die
8 core NPU	Neural Processing Unit

7.2. Paradigma big.LITTLE (ARM)

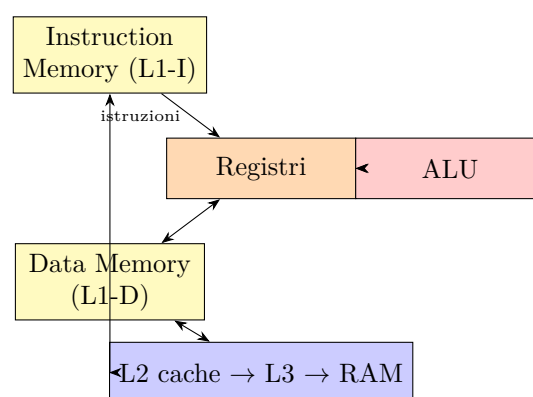


Task semplici → core LITTLE (risparmio batteria). Task pesanti → core BIG (massima potenza).

7.3. NPU — Neural Processing Unit

Unità dedicata all'esecuzione di **reti neurali** — il paradigma di calcolo principale per l'AI. Esempio: riconoscimento facciale (Face ID). I processori moderni integrano sempre più unità dedicate per il calcolo AI.

8. Dentro il CPU core: struttura interna



Due percorsi distinti: uno per le **istruzioni** (I-cache) e uno per i **dati** (D-cache), per accedervi in parallelo senza conflitti.

Panoramica del corso Unimore

Corso	Contenuti
Architettura dei Calcolatori 1	Questo corso — fundamenta
Architettura dei Calcolatori 2	Parallelismo intra-CPU, eterogeneità (GPU, AI/ML), sottosistema di memoria, progetto cache
High Performance Computing (magistrale)	Architetture parallele e omogenee — progettazione e programmazione