

# Introduzione alle architetture

*Alessandro Pellegrini*  
*a.pellegrini@ing.uniroma2.it*

# Il “pensiero” informatico

- 1941: *Non c'è motivo per cui un individuo debba avere un computer in casa*  
–Ken Olsen, cofondatore di DEC
- 1943: *Penso che ci sia un mercato mondiale per circa cinque computer*  
–Thomas J. Watson, presidente di IBM
- 1949: *I computer del futuro non peseranno più di 1,5 tonnellate*  
–Usa Popular Mechanics
- 1950: *Dovremo inventare problemi più grandi se vogliamo tenerli occupati*  
–Howard Aiken, pioniere del computer all'IBM
- 1957: *Ho viaggiato in lungo e in largo per questo paese e ho parlato con le persone migliori, e posso assicurarvi che l'elaborazione dei dati è una moda che non durerà oltre l'anno*  
–L'editore responsabile dei libri di business per Prentice Hall
- 1968: *Microchip? A cosa serve?*  
–Ingegneri IBM

# Computer! Computer dappertutto!

- Tutte le discipline si affidano ai computer per molti aspetti del loro lavoro
  - non solo elaborazione di testi, fogli di calcolo, CAD, ecc.
  - i metodi computazionali, l'estrazione dei dati, l'analisi e la sintesi sono fondamentali per i progressi in molti campi
  - I computer sono veloci, ma fargli fare ciò che si vuole può essere complesso
- I problemi da risolvere diventano sempre più complicati o decentralizzati, e spesso richiedono l'interconnessione di più computer contemporaneamente

**La buona notizia sui computer è che fanno quello che gli si dice di fare. La cattiva notizia è che fanno quello che gli si dice di fare.**

*— Ted Nelson, Philosopher*

# Che cos'è un calcolatore?

- Dal dizionario Treccani:
  1. In senso generico, chi, o che, fa i calcoli
  2. Con significato più specifico, il tecnico, per lo più ingegnere, che, nel corso di una progettazione di strutture in cemento armato, in acciaio, ecc., esegue i particolari calcoli di stabilità
  3. Persona che pondera attentamente le cose prevedendone con chiarezza gli sviluppi e le conseguenze, e nell'azione mira diritto allo scopo
  4. Macchina elettromeccanica per il calcolo, capace di accettare e immagazzinare informazioni in una forma stabilita, di elaborarle, e di fornire quindi i risultati dell'elaborazione sotto forma o di dati alfanumerici (ma anche grafici o d'altro tipo) o di segnali per il governo automatico di altre macchine o processi

# L'abaco

- Uno strumento di calcolo utilizzato secoli prima dell'adozione del sistema numerico moderno scritto.
- È la prima “macchina” utilizzata per calcolare



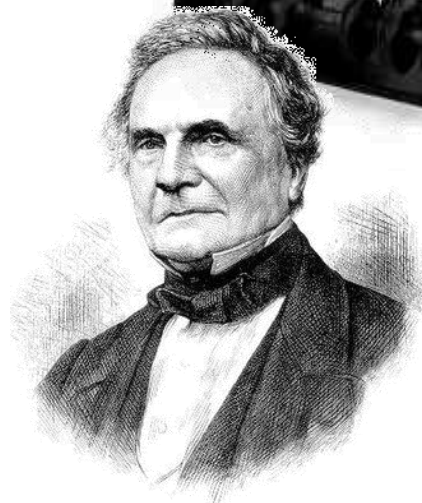
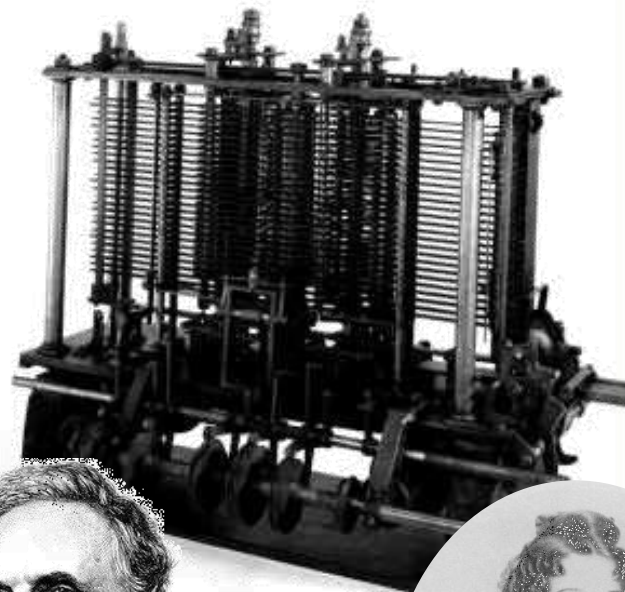
# Il telaio di Jacquard



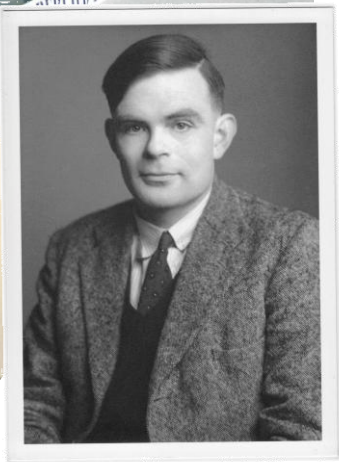
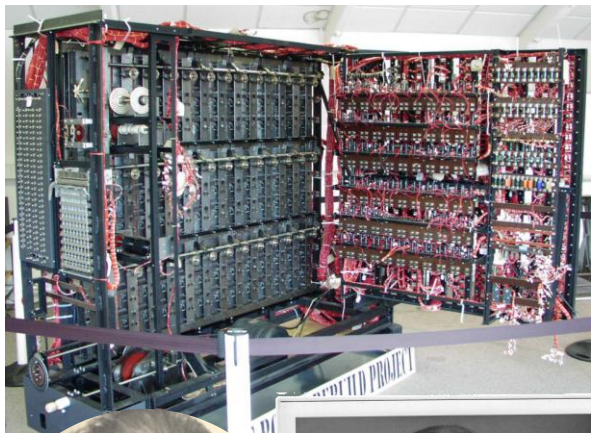
- Un telaio meccanico controllato da una catena di schede perforate
- La prima macchina “programmabile”
- Brevettata da Joseph Marie Jacquard nel 1804

# La macchina analitica

- Una proposta di computer meccanico di uso generale
- Incorporava un'unità logico aritmetica, un flusso di controllo e una memoria integrata.
- L'hardware fu progettato da Charles Babbage nel 1837
- Fu estesa da Ada Lovelace, che scrisse il primo programma informatico



# 1938: Bomba kryptologiczna

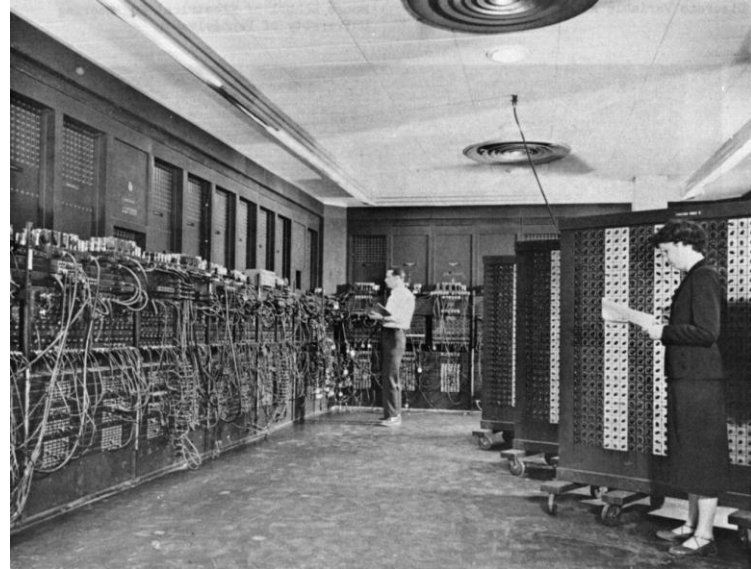


- Una macchina dedicata alla decifrazione a forza bruta dei messaggi segreti tedeschi codificati con Enigma
- Basata sugli studi del 1932 di Marian Rejewski
- Ripresa ed estesa da Alan Turing nel 1939



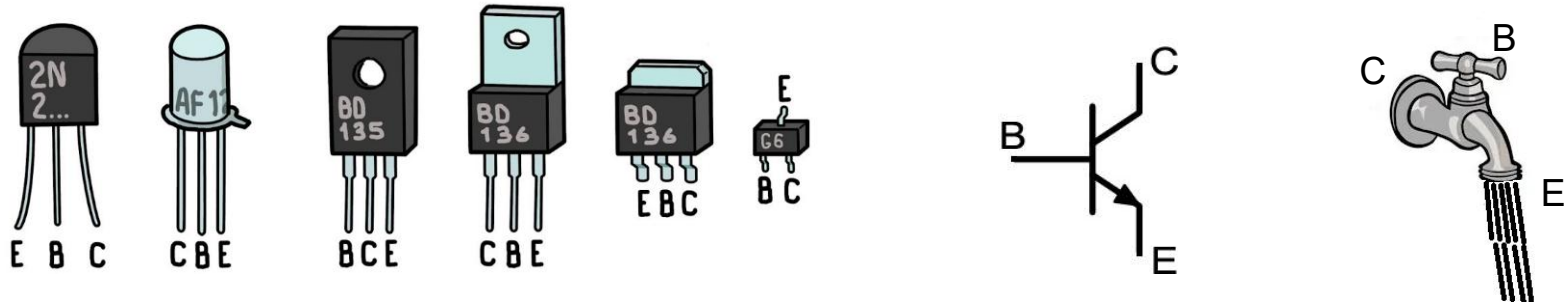
# 1946: ENIAC

- Il primo computer *general purpose* della storia
- La sua memoria poteva contenere “ben” 20 numeri interi di 10 cifre
- Occupava una superficie di 180 metri quadrati e aveva un peso di 30 tonnellate
- Il primo problema che risolse riguardava il progetto Manhattan
- Richiese 1 milione di schede perforate



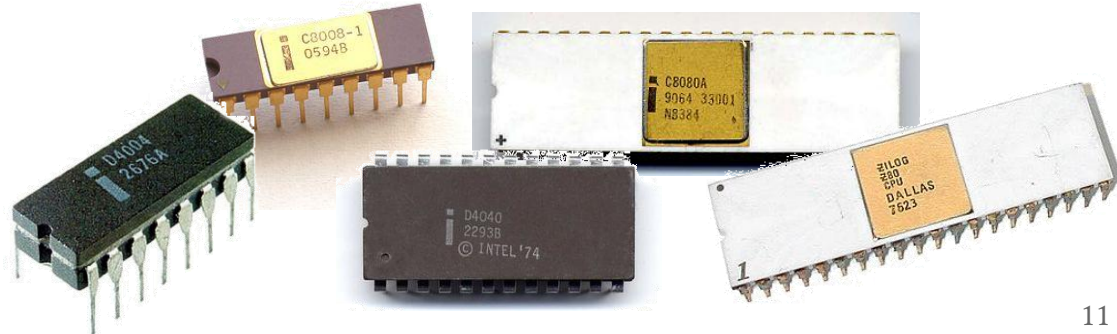
# L'epoca del transistor

- Il transistor fu inventato nel 1925 da Julius Lilienfeld nel 1925
- I primi dispositivi furono costruiti solo nel 1952
- Il tipo più comune, il MOSFET, risale al 1960



# I microprocessori

- Sono circuiti elettronici di dimensioni molto ridotte per l'elaborazione di istruzioni
- La tecnologia a microprocessore è alla base della costruzione di CPU e GPU
- La loro realizzazione è stata possibile grazie alla tecnologia LSI sviluppata da Federico Faggin nel 1968



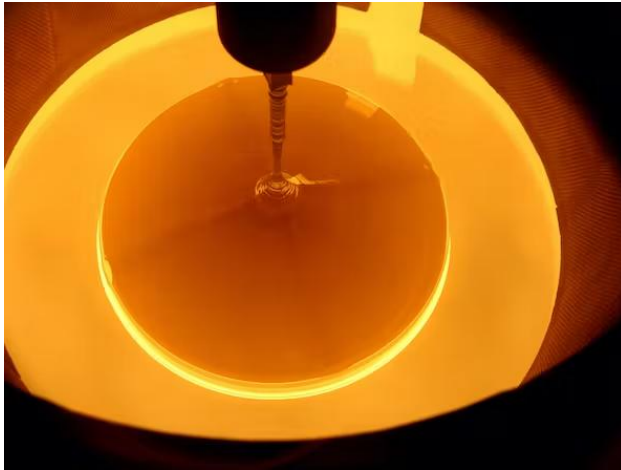
# Com'è costruito un microprocessore?

- I microprocessori sono fatti di sabbia!
- Si trova abbondantemente nella crosta terrestre ed è costituita da circa il 25-50% di biossido di silicio.
- Viene lavorata per separare il silicio da tutti gli altri materiali presenti nella sabbia.
- Il silicio è un materiale ottimo, perché non è un buon conduttore



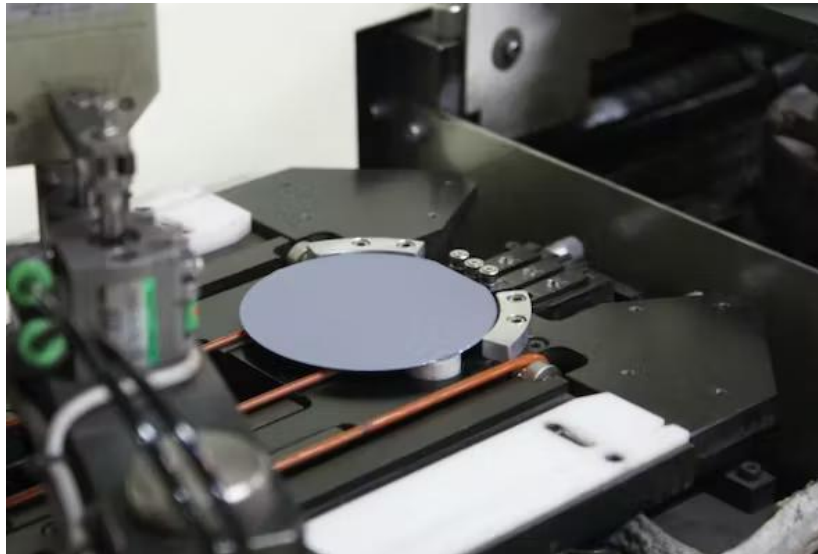
# Com'è costruito un microprocessore?

- Il silicio puro al 99,9999% viene fuso per formare un *lingotto*
- Tipicamente, ha un diametro di 300 mm e un peso di 100 kg



# Com'è costruito un microprocessore?

- Il lingotto viene poi “affettato” in *wafer* di silicio
- Lo spessore si aggira intorno ai 900  $\mu\text{m}$
- La superficie viene lucidata a specchio, per rimuovere le impurità



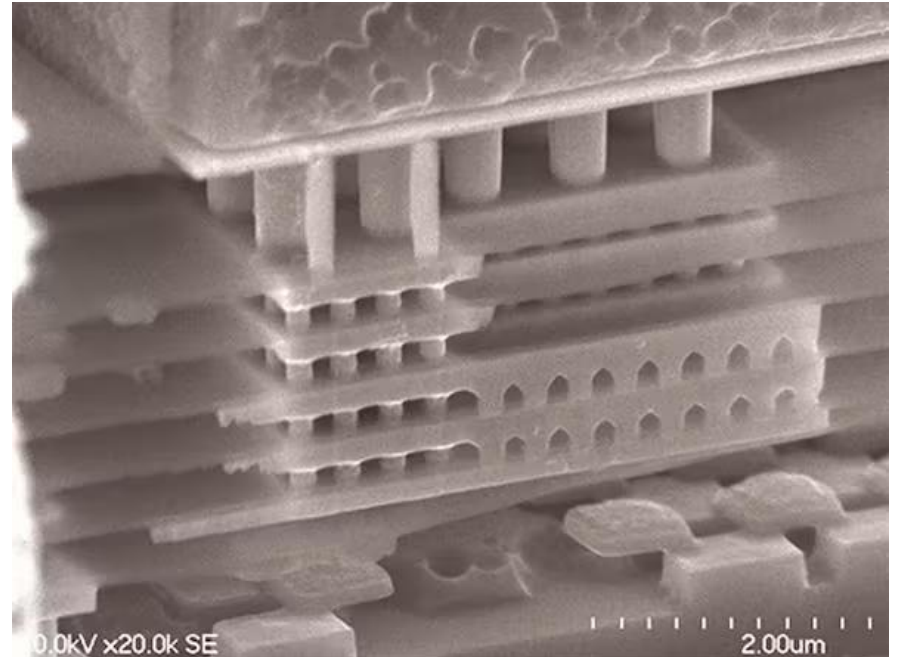
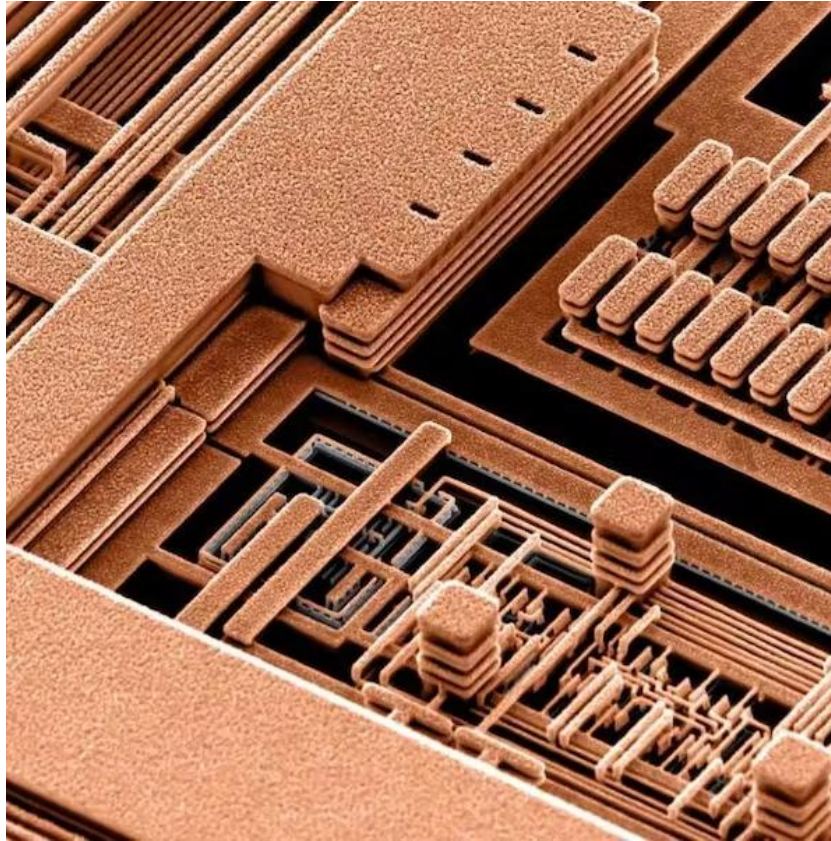
# Com'è costruito un microprocessore?

- La fase successiva è la “stampa”, o *fotolitografia*
- Un fascio di elettroni viene concentrato sulla superficie del wafer
- Apposite maschere schermano aree specifiche della superficie
- La regione non esposta viene rimossa, mediante incisione chimica
- Nelle aree incise, viene poi depositato altro materiale per formare le componenti elettroniche
- È un processo a bassa resa: solo circa il 30% dei *die* viene stampato correttamente





# Il risultato



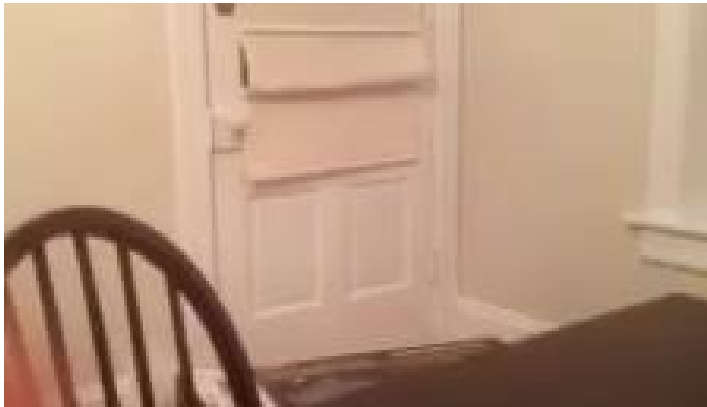


# Livelli di astrazione

- L'ingegneria informatica è tutta una questione di *astrazione*
- I problemi devono essere astratti in modo “computazionalmente fattibile”.
- Si astraggono anche i vari livelli che compongono un “sistema”.
- Il numero di livelli di astrazione che lavorano insieme in un dato momento nei computer di oggi è enorme



# Livelli di astrazione



AN x64 PROCESSOR IS SCREAMING ALONG AT BILLIONS OF CYCLES PER SECOND TO RUN THE XNU KERNEL, WHICH IS FRANTICALLY WORKING THROUGH ALL THE POSIX-SPECIFIED ABSTRACTION TO CREATE THE DARWIN SYSTEM UNDERLYING OS X, WHICH IN TURN IS STRAINING ITSELF TO RUN FIREFOX AND ITS GECKO RENDERER, WHICH CREATES A FLASH OBJECT WHICH RENDERS DOZENS OF VIDEO FRAMES EVERY SECOND

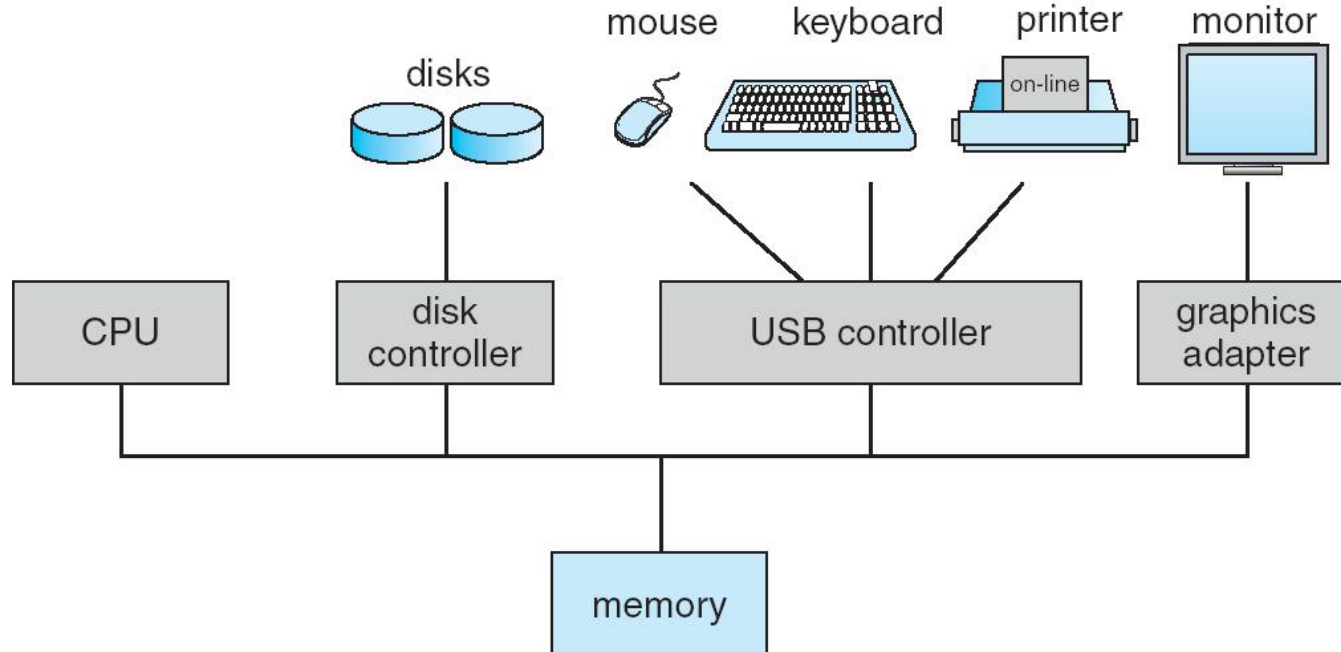
BECAUSE I WANTED TO SEE A CAT JUMP INTO A BOX AND FALL OVER.



I AM A GOD.

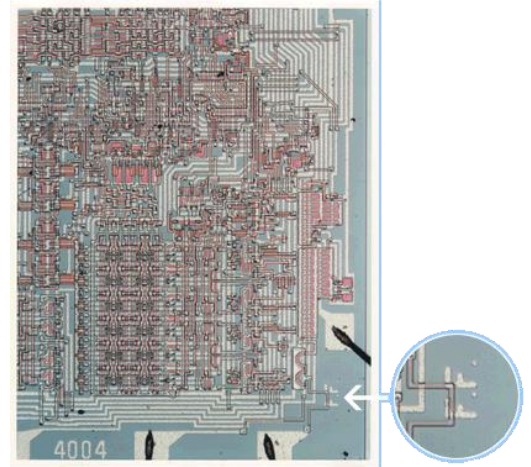
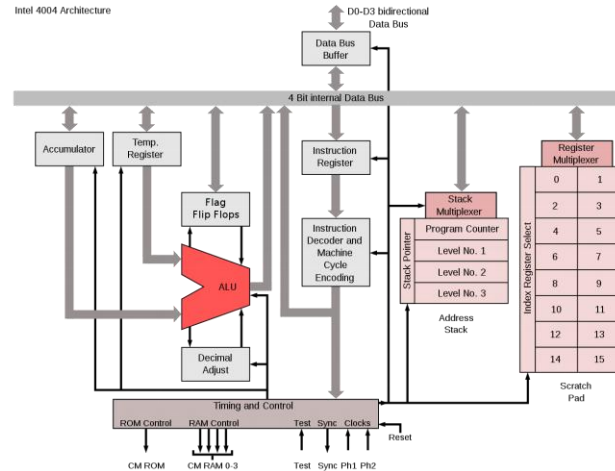
# Calcolatori e architettura dei calcolatori

- L'architettura dei calcolatori descrive la funzionalità, l'organizzazione e l'implementazione di un sistema informatico



# L'hardware

- L'hardware comprende le parti fisiche di un computer
- La “rivoluzione informatica” è iniziata quando l'hardware è diventato programmabile
- Anche la progettazione dell'hardware è tipicamente astratta



# Il software

- Il software è un *insieme* di *dati* e *istruzioni* che indicano al computer come lavorare.
- Le unità di elaborazione (CPU) parlano il “codice macchina”
  - in genere è difficile da scrivere direttamente
- Esistono *astrazioni* più elevate che semplificano lo sviluppo di applicazioni: i *linguaggi di programmazione*
- Esistono diversi *paradigmi di programmazione*, a seconda delle caratteristiche del linguaggio
  - *Imperativi*: il programmatore istruisce la macchina su come cambiare il suo stato
    - procedurale, orientato agli oggetti
  - *dichiarativi*: il programmatore si limita a dichiarare le proprietà del risultato desiderato
    - funzionale, logico, matematico

# Come nasce un programma

Codice sorgente

`a = b + c`

compilatore

Assembly

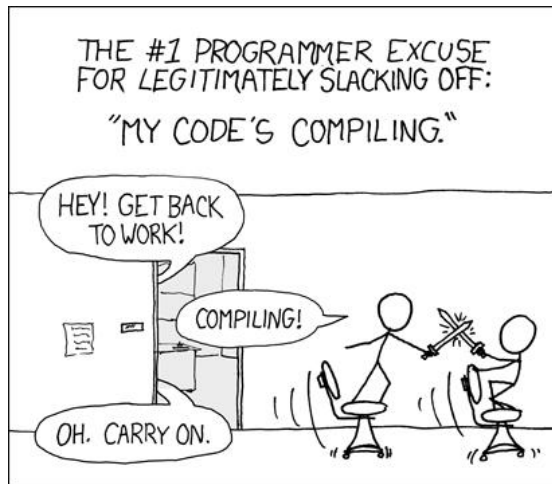
```
movw b, %ax
movw c, %bx
addw %ax, %bx
movw %bx, a
```

assemblatore

Codice macchina

```
000101..0101001
1011101..010100
01011..11101010
010..1110101010
```

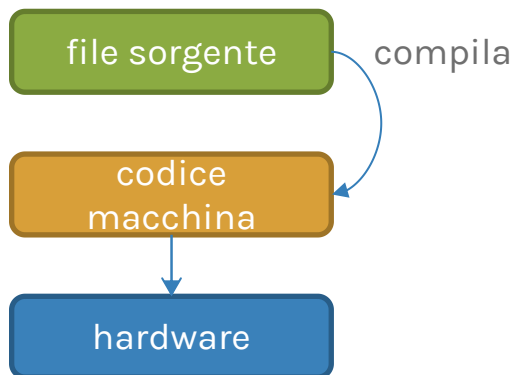
Set istruzioni



# Modelli di esecuzione e linguaggi di programmazione

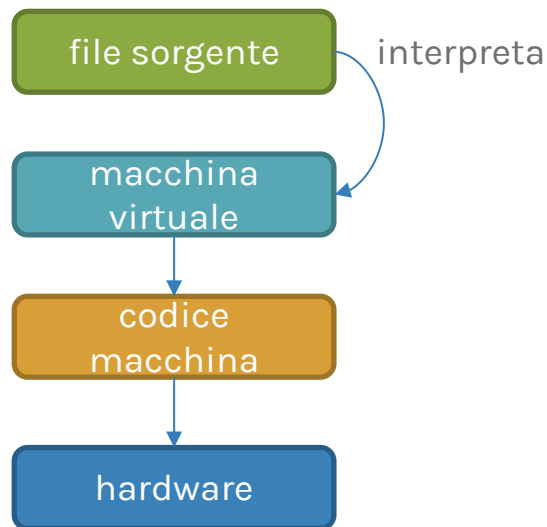
## compilati

C, C++, Go, Fortran, Pascal, Rust



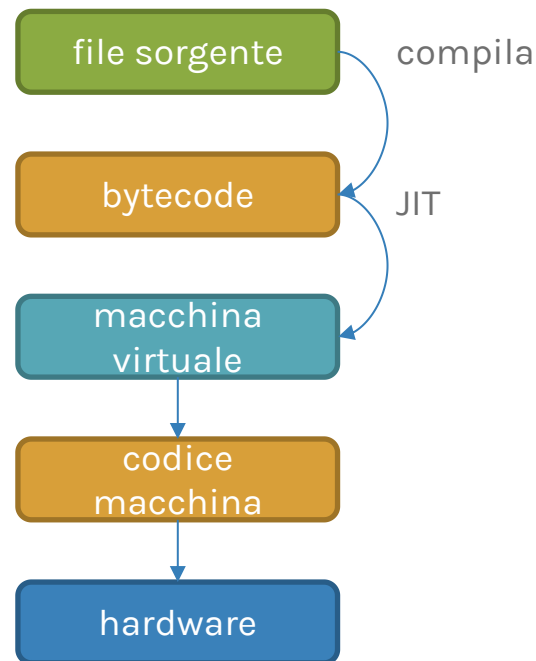
## interpretati

Python, PHP, Ruby, Javascript



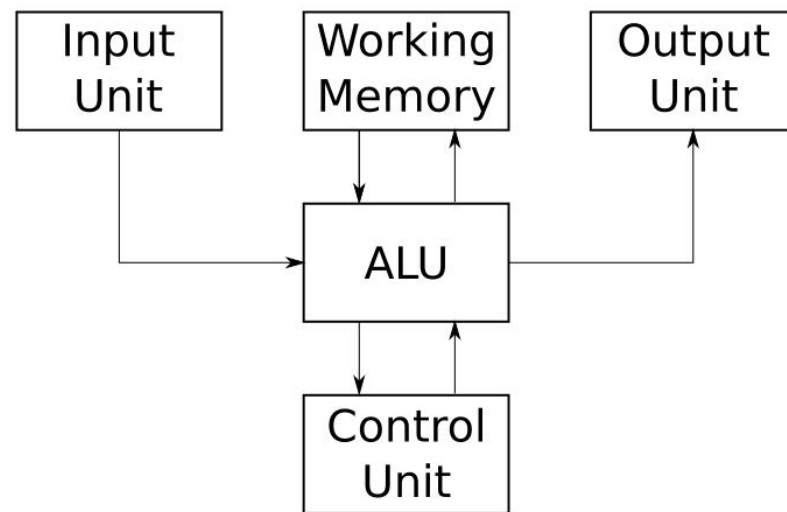
## basati su bytecode

Java, Kotlin



# Architettura di von Neumann

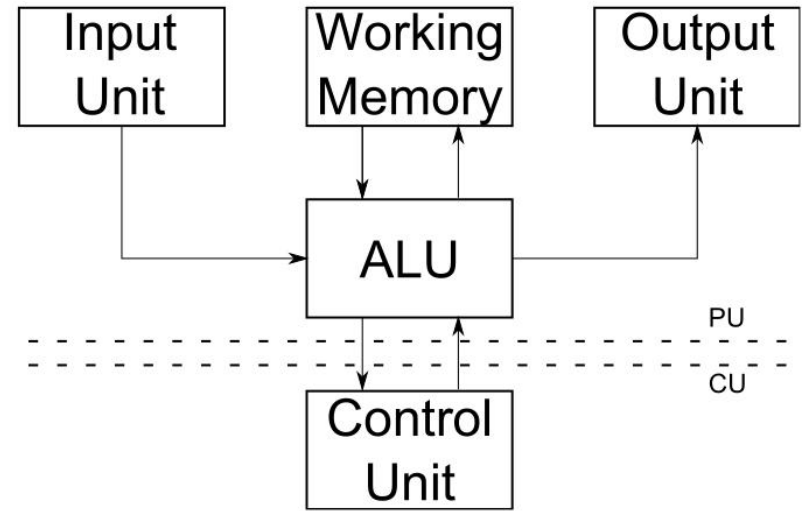
- La componente centrale è l'unità logico aritmetica (ALU)
  - oggi il termine ALU viene usato per identificare un componente del processore
- La ALU è pilotata da una unità di controllo
- La memoria di lavoro permette di memorizzare risultati intermedi
- La ALU comunica con dispositivi di ingresso e di uscita





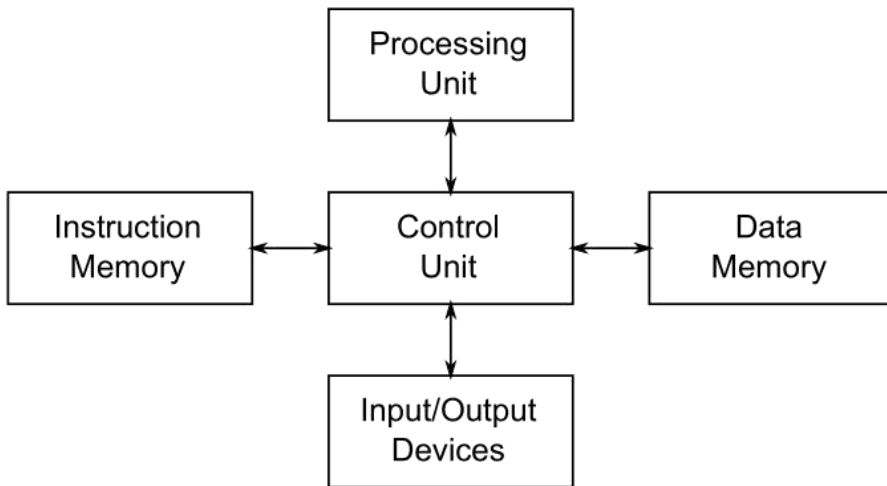
# Organizzazione CU/PU

- La ALU e l'unità di controllo sono realizzate in maniera differente
- Hanno requisiti e obiettivi diversi
- Possiamo operare una separazione logica tra:
  - Unità di processamento: tutto ciò che “esegue il lavoro”
  - Unità di controllo: la parte “intelligente”
- Problema: una sola memoria!
  - Latenza di accesso maggiore
  - Collo di bottiglia di von Neumann



# Architettura Harvard

- Datapath alternativi per le istruzioni e i dati
- Maggiore reattività del sistema
- L'unità di controllo deve governare l'accesso alle risorse in maniera più complessa
- Problema:
  - Programma piccolo, tanti dati
  - Programma grande, pochi dati
- Di fatto, è utilizzata solo in scenari molto specifici
- Presente comunque anche nei nostri computer



# Un'architettura moderna

- Il processore è troppo veloce per la memoria e gli altri dispositivi
  - perderebbe troppo tempo ad aspettarli
- Si fa aiutare da alcuni *coprocessori* che fanno da tramite per esso
- Nei processori più moderni, i coprocessori sono all'interno della CPU

