

# Informatica di base 6/ed

Autori: Dennis P. Curtin, Kim Foley, Kunal Sen e Cathleen Morin  
A cura di: Agostino Marengo e Alessandro Pagano



## Capitolo 2 L'architettura del computer e la CPU

Copyright © 2016  
McGraw-Hill Education (Italy) srl



## Un po' di storia



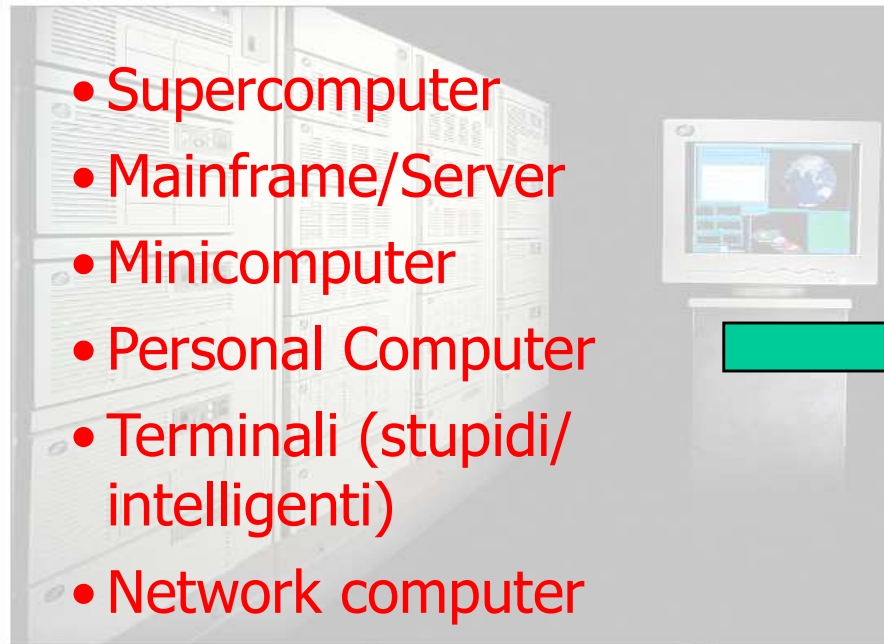
- **Primi elaboratori**

- 50 anni fa circa
- Grandi ambienti con aria condizionata
- Raffreddati ad acqua

- **CPU (Central Processing Unit)**

- 1969: Hoff, ingegnere Intel, progetta il primo microprocessore grande quanto un'unghia

## Tipi di elaboratori



*Un server – per gentile concessione della IBM*



*Un desktop – per gentile concessione della Sony*

Il linguaggio dell'informatica è mutevole.

# Supercomputer



- I più potenti elaboratori disponibili
- Applicazioni in campo tecnico e scientifico
  - Previsioni metereologiche
  - Simulazione fusione nucleare
  - Progettazione automobili
  - Effetti speciali cinematografici
  - ...



## Mainframe e minicomputer (server)



- **Mainframe**

- Utilizzati dalle grandi aziende per funzioni centralizzate
- Occupano una stanza
- Gestiti da personale altamente specializzato



- **Minicomputer**

- Meno potenti dei mainframe
- Diverse dimensioni
- Utilizzati da grandi aziende



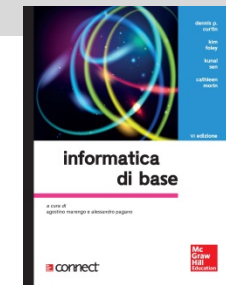
## Personal Computer (PC)



- Elaboratori in grado di lavorare autonomamente.
- Possono elaborare dati provenienti da altri PC.
- Possono essere connessi ad altri PC.
- 1981: IBM realizza il primo personal computer (PC IBM).



## Workstation



- Computer personali ad alto rendimento e piccole dimensioni
- Utilizzo tecnico e scientifico





- **Computer portatili**

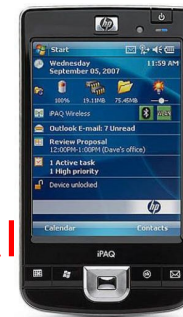
- Leggeri
- Maneggevoli
- Funzionamento a batteria
  - Docking station





## I più piccoli

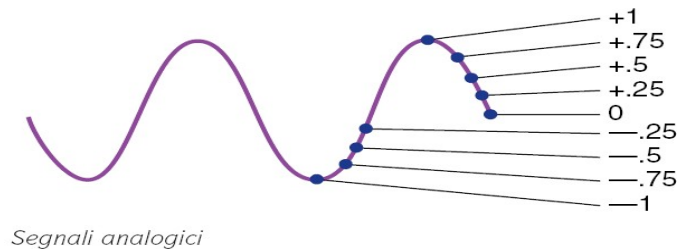
- **Tablet PC**
  - Computer portatili privi di tastiera fissa
  - Touch-screen
- **Computer palmari**
  - I più piccoli
  - Hanno tastiere piccole
- **PDA (Personal Digital Assistant)**
  - Privi di tastiera





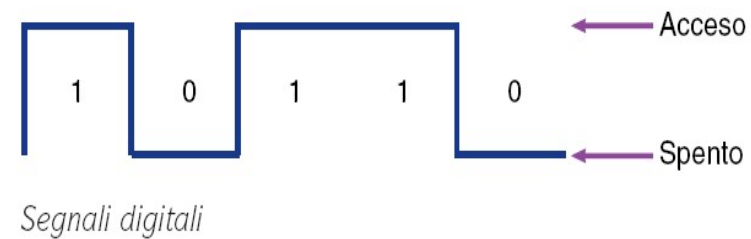
## Segnale analogico

- come un'onda che trasporta informazioni, massimi, minimi e tutti i valori intermedi
- i segnali analogici sono molto sensibili alle interferenze

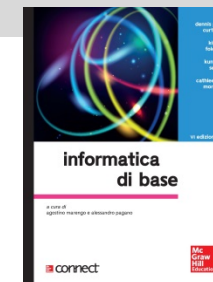


## Segnale digitale

- assume solo due stati: acceso/spento, sì/no, vero/falso
- il segnale digitale è più facile da distinguere, quindi risente meno delle interferenze



## I sistemi di numerazione

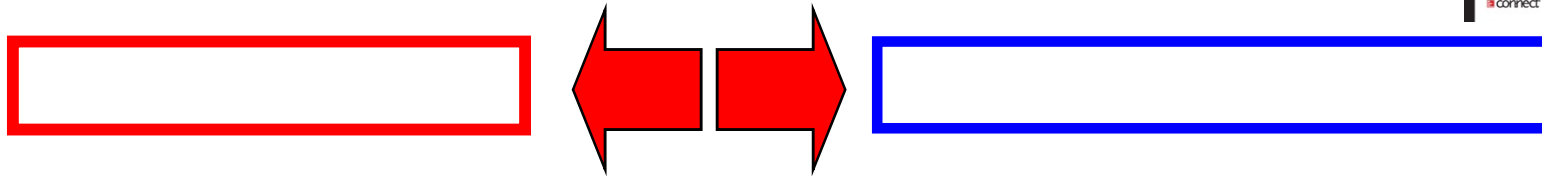


- Molte informazioni sono quantitative, quindi esprimibili in forma numerica.
- Le informazioni numeriche possono essere elaborate attraverso l'applicazione di operazioni.
- Un sistema di numerazione è una struttura matematica che permette di rappresentare i numeri attraverso dei simboli.

Num eri decin ali	Num eri binari			
0.....				0
1.....				1
2.....		1		0
3.....		1		1
4.....	1	0		0
5.....	1	0		1
6.....	1	1		0
7.....	1	1		1
8.....	1	0	0	0
9.....	1	0	0	1
10.....	1	0	1	0



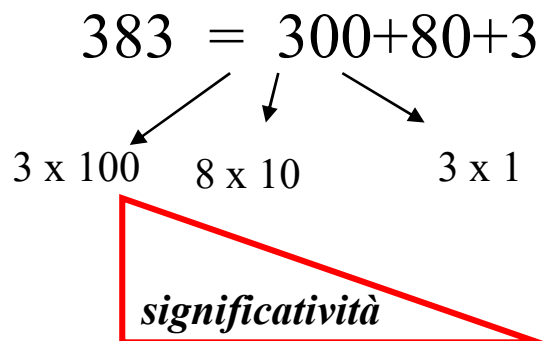
# I sistemi di numerazione



- ai diversi simboli dell'alfabeto (cifre), viene associato un **valore crescente** in modo **lineare** da destra verso sinistra;
- il significato di un simbolo ( il suo valore) dipende ordinatamente dalla sua posizione nella stringa

## ESEMPIO:

- il sistema di numerazione decimale arabo: 10 simboli (0, 1, 2, ...9)



- Il significato dei simboli non dipende dalla loro posizione
- ma è stabilito in base ad una legge additiva dei valori dei singoli simboli (se posti in ordine crescente)

## ESEMPIO:

- il sistema di numerazione romano

$$I = 1$$

$$V = 5$$

$$X = 10$$

$$L = 50$$

...

$$LXIV = 50 + 10 - 1 + 5 = 64$$

## Sistemi di numerazione non posizionali



- **Un simbolo rappresenta un numero.**
  - Esempio (numeri romani)
    - L rappresenta il numero 50
    - X rappresenta il numero 10
    - V rappresenta il numero 5
    - I rappresenta il numero 1
- **Il numero rappresentato da una stringa di simboli si ottiene attraverso regole operazionali applicate ai simboli della stringa.**
  - Esempio (numeri romani): LXXIV rappresenta  $50+10+10-1+5 = 74$
- **Difficile effettuare operazioni.**
- **Rappresentazione non compatta.**

# Sistemi di numerazione posizionali



**Base** = numero di simboli o di cifre numeriche richieste dal sistema per rappresentare la serie infinita dei numeri.

Dato un alfabeto ordinato di  $b$  simboli distinti ( $c_1, c_2, \dots, c_b$  che rappresentano rispettivamente i naturali  $0, 1, 2, \dots, b-1$ ), si rappresenti nel modo più semplice e compatto ogni altro numero  $x \geq b$  mediante una stringa di simboli dell'alfabeto.

**$b$  = base del sistema di numerazione**

$b = 60$  babilonesi

$b = 20 \vee 18$  maya

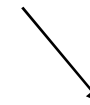
$b = 10$  arabi (0,1,2,3,4,5,6,7,8,9)

$b = 2 \vee 8 \vee 16$  informatici (0,1)

(0,1,2,3,4,5,6,7)

(0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F)

101 = ?



***In quale base?***

## Valore della posizione



*La posizione di un simbolo all'interno di un numero indica il valore che esso esprime, o più precisamente **l'esponente che bisogna dare alla base per ottenere il valore corretto.***

Il valore (o la quantità) di **0, 1, 2, 3, 4, 5, 6, 7, 8 e 9** dipende dalla **posizione** che ciascuno di essi assume all'interno del numero:

**la prima cifra a destra rappresenta le unità (il coefficiente di  $10^0$ ), la seconda le decine ( $10^1$ ), la terza le centinaia ( $10^2$ ), e così via.**

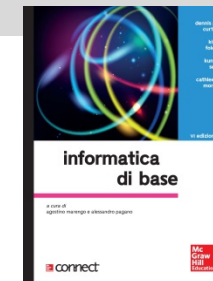
Il numero **3.098.323** è una rappresentazione abbreviata di



$$(3 \times 10^6) + (0 \times 10^5) + (9 \times 10^4) + (8 \times 10^3) + (3 \times 10^2) + (2 \times 10^1) + (3 \times 10^0 = 3 \times 1).$$

Il primo 3 (leggendo da destra a sinistra) rappresenta 3 unità; il secondo 3, sta per 300 unità, o 3 centinaia; infine il terzo 3, per 3 milioni di unità.

## Notazione posizionale



Richiede **b** simboli diversi x rappresentare i numeri da **0** a **b-1**

**decimale** → 0,1,2,3,4,5,6,7,8,9 (**b=10**)

**binario** → 0,1 (**b=2, base due**)

**ottale** → 0, 1, 2, 3, 4, 5, 6, 7 (**b=8**)

**esadecimale** → 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A (dieci),  
B (undici), C (dodici), ..., F (quindici). (**b=16**)

**Esempio:**

  $30.155_6$  nel sistema **in base sei** è il numero

$$(3 \times 6^4) + (0 \times 6^3) + (1 \times 6^2) + (5 \times 6^1) + (5 \times 6^0) = 3959 \text{ nel s.d.};$$

  $2EF_{16}$  del **sistema esadecimale** è il numero

$$(2 \times 16^2) + (14 \times 16^1) + (15 \times 16^0) = 751 \text{ del s.d.}$$



## Il sistema di numerazione decimale



- È il sistema più conosciuto dall'uomo.
- La base  $\beta$  è pari a  $10$ .
- I simboli utilizzati sono  $0, 1, 2, \dots, 9$  dal significato ovvio.
  - Esempio: la stringa  $2349$  rappresenta il numero  $2 \cdot 10^3 + 3 \cdot 10^2 + 4 \cdot 10^1 + 9 \cdot 10^0$ .
- I *numeri decimali* sono facilmente intelligibili.

## Il sistema di numerazione binario



- È il sistema maggiormente utilizzato dai sistemi di elaborazione.
- La base  $\beta$  è pari a 2.
- I simboli utilizzati sono 0 e 1, rappresentanti lo zero e l'unità.
  - Esempio: la stringa binaria 10010 rappresenta il numero  $1*2^4+0*2^3+0*2^2+1*2^1+0*2^0$  (=18 in decimale).
- Per rappresentare un numero  $x$ , sono necessarie  $\lceil \log_2(x+1) \rceil$  cifre binarie.
- È scarsamente leggibile, specie quando le stringhe sono molto lunghe.

## Un esempio di messaggio digitale



- **La cavalcata di Paul Revere**
  - “Una lanterna se vengono da terra, e due se vengono dal mare”
- **Segnale digitale?**
- **Quanti stati?**
  - Lanterne entrambe spente (00)
  - Una lanterna accesa (01 e 10)
  - Entrambe le lanterne accese (11)



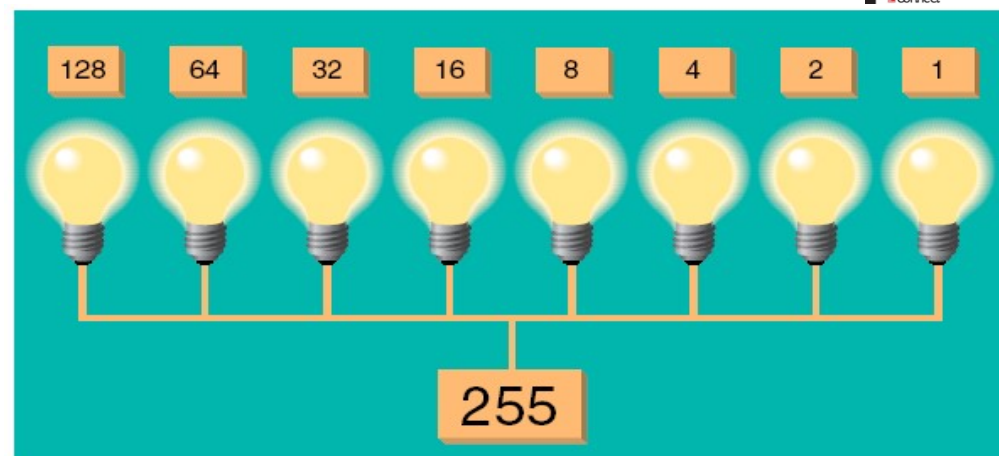
1 bit = 1 cifra binaria: 0 o 1

1 byte = 8 bit, es. 00000110

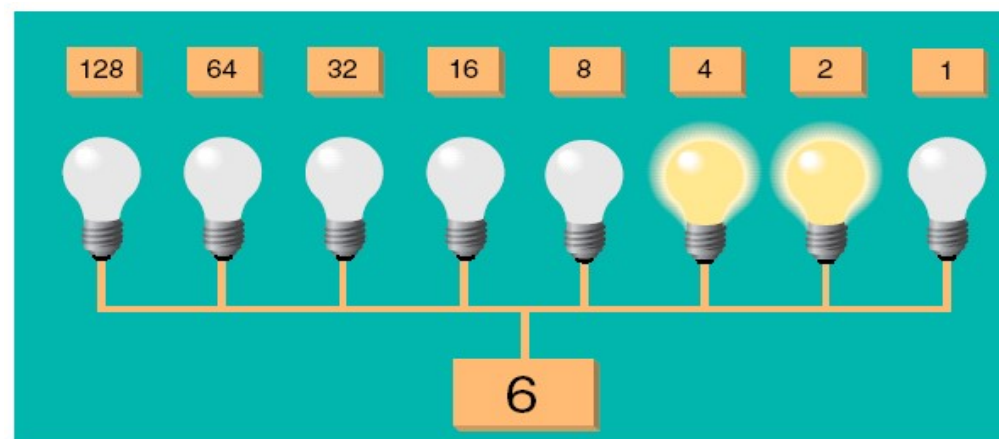
Nel sistema decimale:  
312 = 3 centinaia, 1 decina e  
due unità, cioè:  
 $3 \times 10^2 + 1 \times 10^1 + 2 \times 10^0$

Nel sistema binario:  
 $0 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 +$   
 $+ 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 =$   
 $= 0 + 0 + 0 + 0 + 0 + 4 + 2 + 0 = 6$

Quindi il numero binario  
00000110 in decimale è il  
numero 6.



*I bit che danno come somma 255*



*I bit che danno come somma 6*

## Le unità di misura nell'informatica



*(bit e byte)*

**1 bit (cifra che può assumere solo due valori, 0/1)**

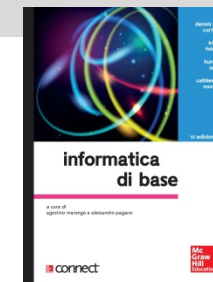
**8 bit = 1 Byte = 1 carattere**

- L'aggregazione degli 8 bit necessari per definire un carattere alfanumerico viene definita byte e rappresenta l'unità pratica principale in informatica, in quanto permette di esprimere un singolo carattere alfanumerico.
- I multipli del byte, espressi con 2 (il numero delle cifre nel sistema di numerazione binaria) elevato alle potenze di 10, sono:
  - **$2^{10}$  byte = 1024 byte = 1 Kilobyte = 1 KB**
  - **$2^{20}$  byte = 1024 Kbyte = 1 Megabyte = 1 MB**
  - **$2^{30}$  byte = 1024 Mbyte = 1 Gigabyte = 1 GB**
  - **$2^{40}$  byte = 1024 Gbyte = 1 Terabyte = 1 TB**

## I codici



- **Codice: associazione di un significato ad una sequenza di simboli.**
- **Codice : {entità} → {sequenze di simboli}**
  - Un codice è una rappresentazione
- **Un codice:**
  - permette l'elaborazione e la memorizzazione di entità non gestibili nella loro forma originale
    - i sistemi di numerazione sono codici
  - permettere l'interpretazione dei simboli
    - un dizionario di lingua italiana è un codice
  - Aggiungere proprietà ad un sistema di simboli
    - comprimere la lunghezza delle stringhe
    - aumentare l'affidabilità di trasmissione



	0	1	2	3	4	5	6	7	8	9
0										
1										
2										
3				!	"	#	\$	%	&	'
4	(	)	*	+	,	-	.	/	0	1
5	2	3	4	5	6	7	8	9	:	;
6	<	=	>	?	@	A	B	C	D	E
7	F	G	H	I	J	K	L	M	N	O
8	P	Q	R	S	T	U	V	W	X	Y
9	Z	[	\	]	^	_	`	a	b	c
10	d	e	f	g	h	i	j	k	l	m
11	n	o	p	q	r	s	t	u	v	w
12	x	y	z	{		}	~	•	•	•
13	,	f	»	...	†	‡	~	%	\$	<
14	œ	•	•	•	•	•	•	•	•	•
15	-	—	-	™	§	>	œ	•	•	ÿ
16		ı	ç	£	¤	¥	ı	§	¨	©
17	ª	«	¬	-	®	-	°	±	²	³
18	´	µ	¶	·	,	1	°	»	¼	½
19	¾	¿	À	Á	Â	Ã	Ä	Å	Æ	Ç
20	È	É	Ê	Ë	Ì	Í	Î	Ï	Ð	Ñ
21	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û
22	Ü	Ý	ß	à	á	â	ã	ä	å	
23	æ	ç	è	é	ê	ë	ì	í	î	ï
24	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù
25	ú	û	ü	ý	þ	ÿ				

Il codice ASCII

La tabella ASCII è stata progettata per convertire un numero da 0 a 255 in un carattere o simbolo del linguaggio naturale. Il codice ASCII è rappresentabile con soli otto bit (un byte), dal numero binario 00000000 (decimale 0) a 11111111 (decimale 255).

## I codici



- **ASCII (American Standard Code for Information Interchange)**
  - 128 caratteri
- **ASCII esteso**
  - 256 caratteri
- **EBCDIC (Extended Binary Coded Decimal Interchange Code)**
- **UNICODE**
  - 96.000 caratteri



## Le origini dell'Informatica

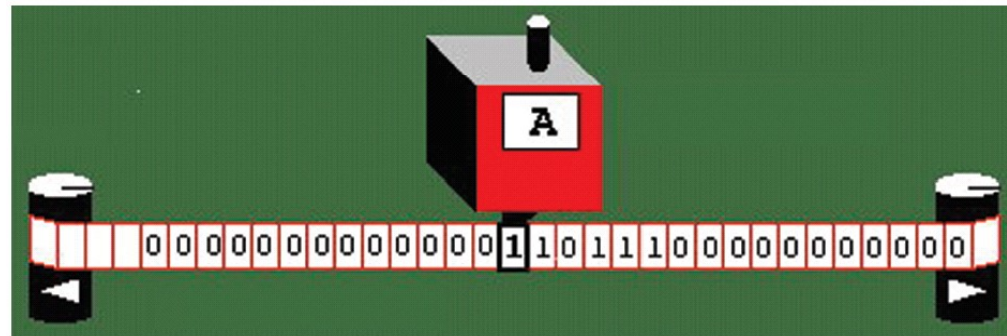


- **Alan Mathison Turing (1912-1954)**
  - Matematico e logico
  - A 25 anni elaborò la teoria della “macchina di Turing”
- **John Von Neumann (1903-1957)**
  - Matematico e informatico
  - Fondatore della teoria dei giochi
- **I calcolatori attuali sono macchine o automi di calcolo generale**
  - Turing e von Neumann sono i padri della moderna informatica

# La macchina di Turing



- Macchina teorica in grado di eseguire procedure logiche e matematiche
- Procedure  $\Leftrightarrow$  algoritmo



La rappresentazione classica di una macchina di Turing

Componenti:

- un *nastro* diviso in celle (memoria esterna);
- una *unità di lettura e scrittura*;
- un *insieme finito di simboli*;
- una *memoria interna*.

La macchina permette di formalizzare la nozione di procedura effettiva di calcolo, o *algoritmo*.

## Teoria della macchina universale



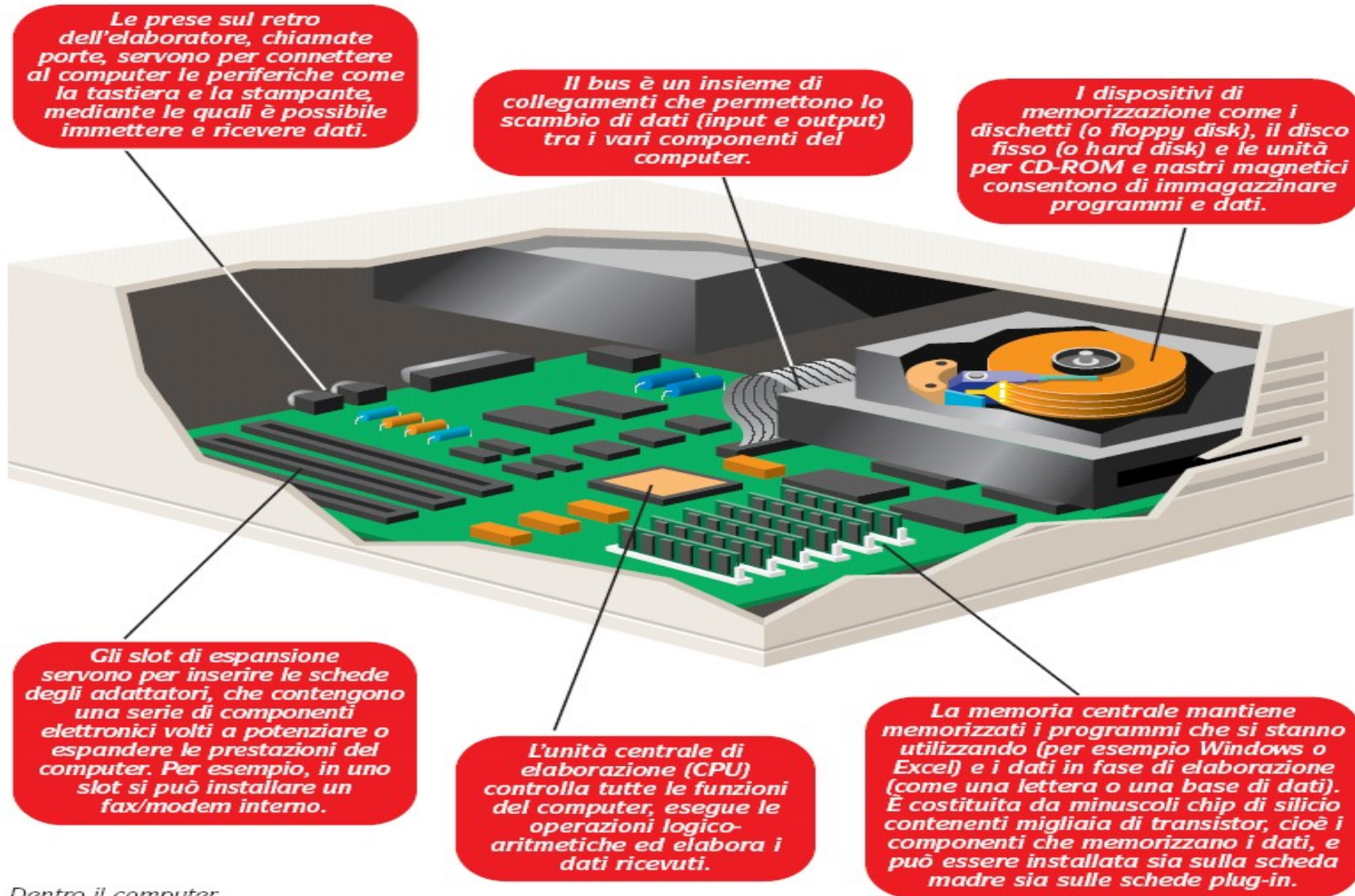
- Turing dimostrò che è possibile progettare una **macchina universale** in grado di imitare qualsiasi altra macchina di Turing.
- **Tesi di Church**
  - L'insieme dei problemi effettivamente risolvibili con qualsivoglia metodo meccanico coincide con quello dei problemi risolvibili dalla macchina di Turing.

## La macchina a registri a programma memorizzato



- Ideata da von Neumann.
- Ispirata dalla macchina (concettuale) di Turing.
- Costituita da:
  - unità di elaborazione centrale (CPU);
  - unità aritmetico-logica (ALU).
- Anche la macchina di von Neumann è una macchina universale.

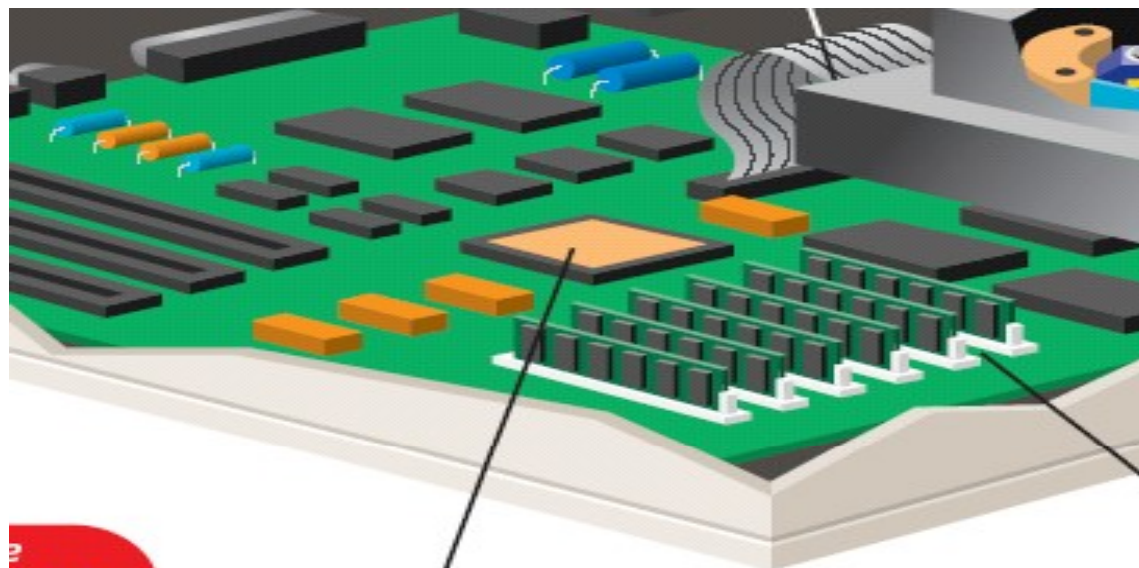
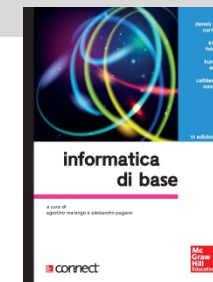
## Dentro la scatola



Dentro il computer



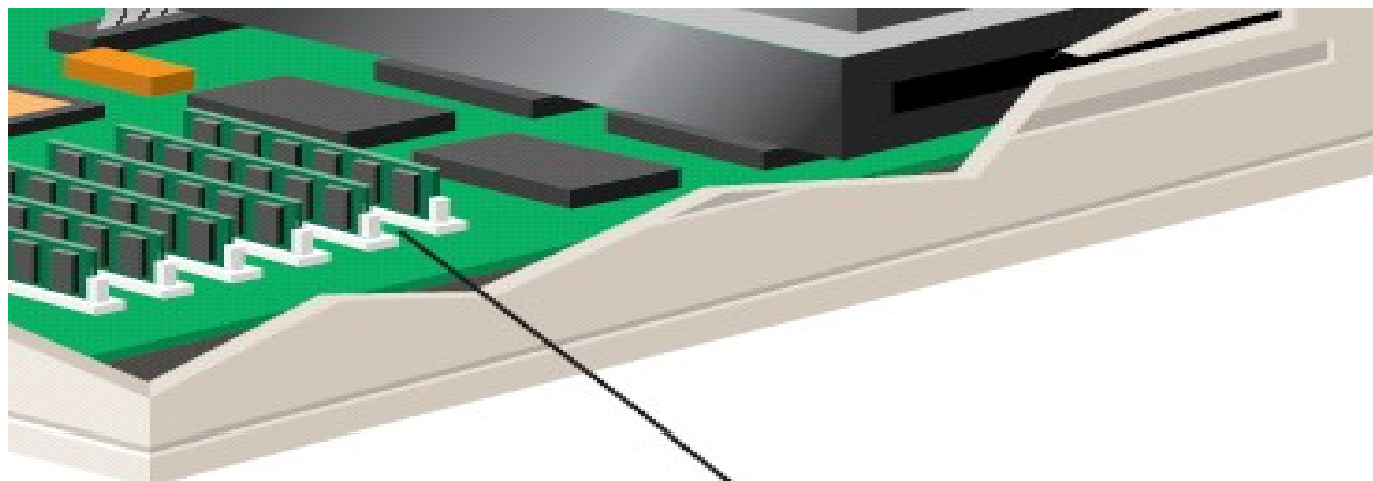
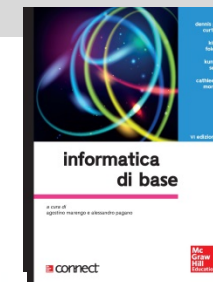
## Dentro la scatola: la CPU



e  
hede  
ngono  
ati  
are o  
del  
n uno  
in

*L'unità centrale di elaborazione (CPU) controlla tutte le funzioni del computer, esegue le operazioni logico-aritmetiche ed elabora i dati ricevuti.*

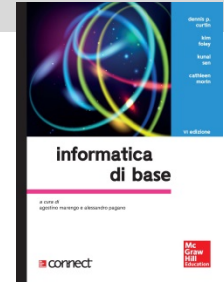
*La m  
memoriz  
utilizza  
Excel) e  
(come u  
È costitu  
contener  
compon  
può esse  
mad*



*le di  
CPU)  
funzioni  
segue le  
nico-  
abora i  
i.*

*La memoria centrale mantiene memorizzati i programmi che si stanno utilizzando (per esempio Windows o Excel) e i dati in fase di elaborazione (come una lettera o una base di dati). È costituita da minuscoli chip di silicio contenenti migliaia di transistor, cioè i componenti che memorizzano i dati, e può essere installata sia sulla scheda madre sia sulle schede plug-in.*

## Dentro la scatola: l'hard disk



*...sieme di  
...permettono lo  
...put e output)  
...onenti del  
...er.*

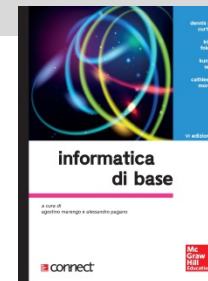
*I dispositivi di  
memorizzazione come i  
dischetti (o floppy disk), il disco  
fisso (o hard disk) e le unità  
per CD-ROM e nastri magnetici  
consentono di immagazzinare  
programmi e dati.*



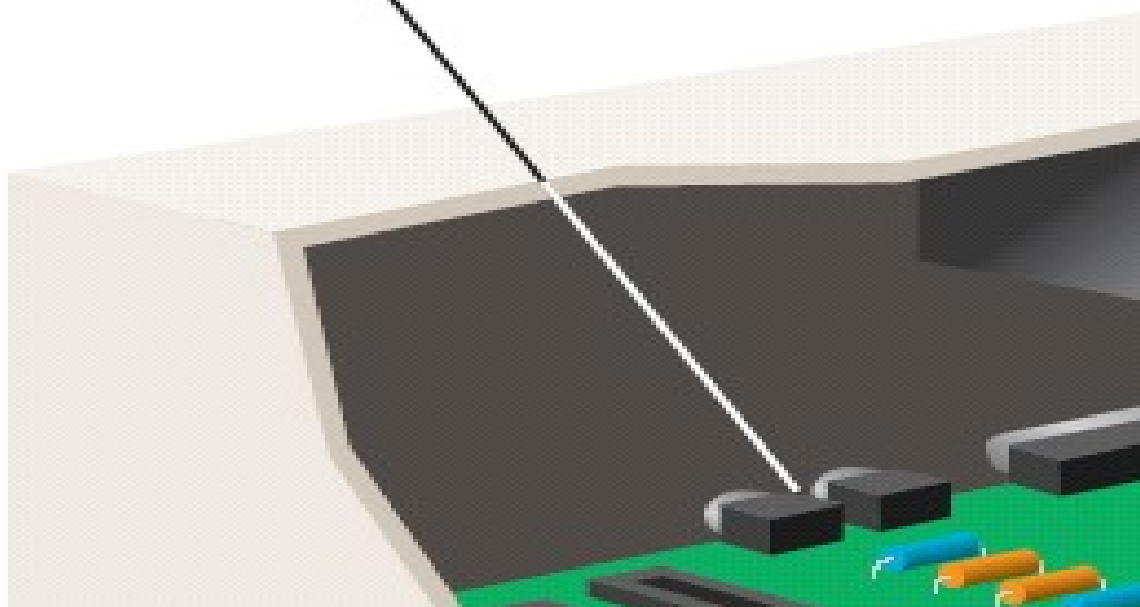


## Dentro la scatola: il bus

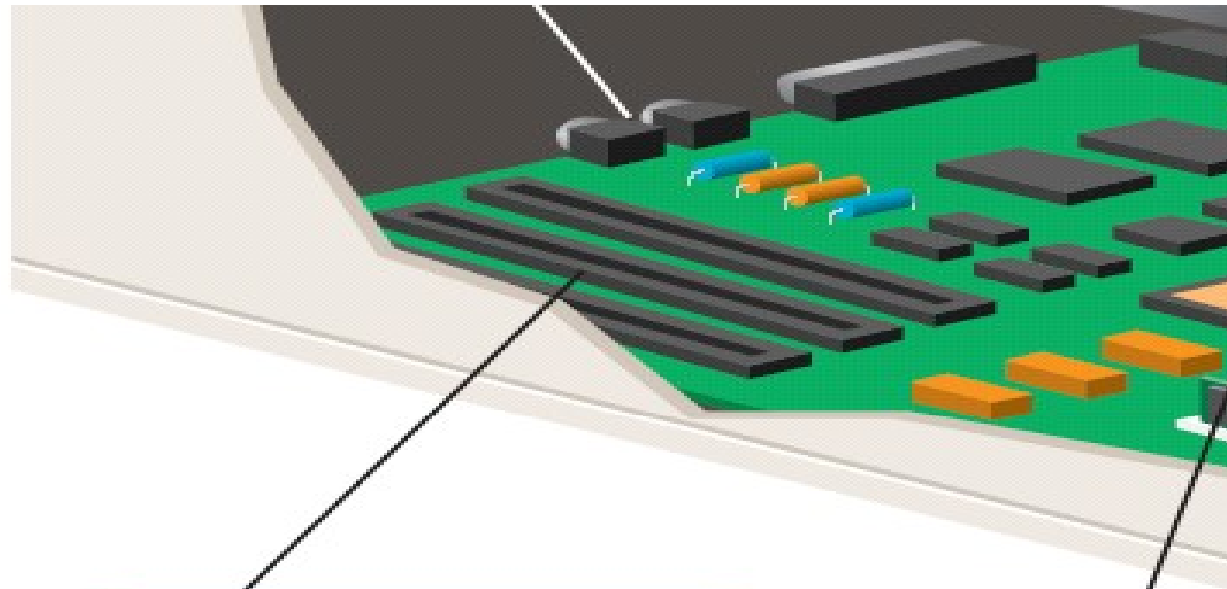
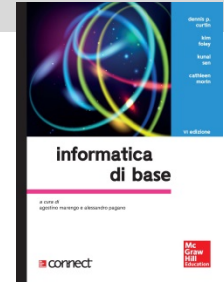




*Le prese sul retro dell'elaboratore, chiamate porte, servono per connettere al computer le periferiche come la tastiera e la stampante, mediante le quali è possibile immettere e ricevere dati.*

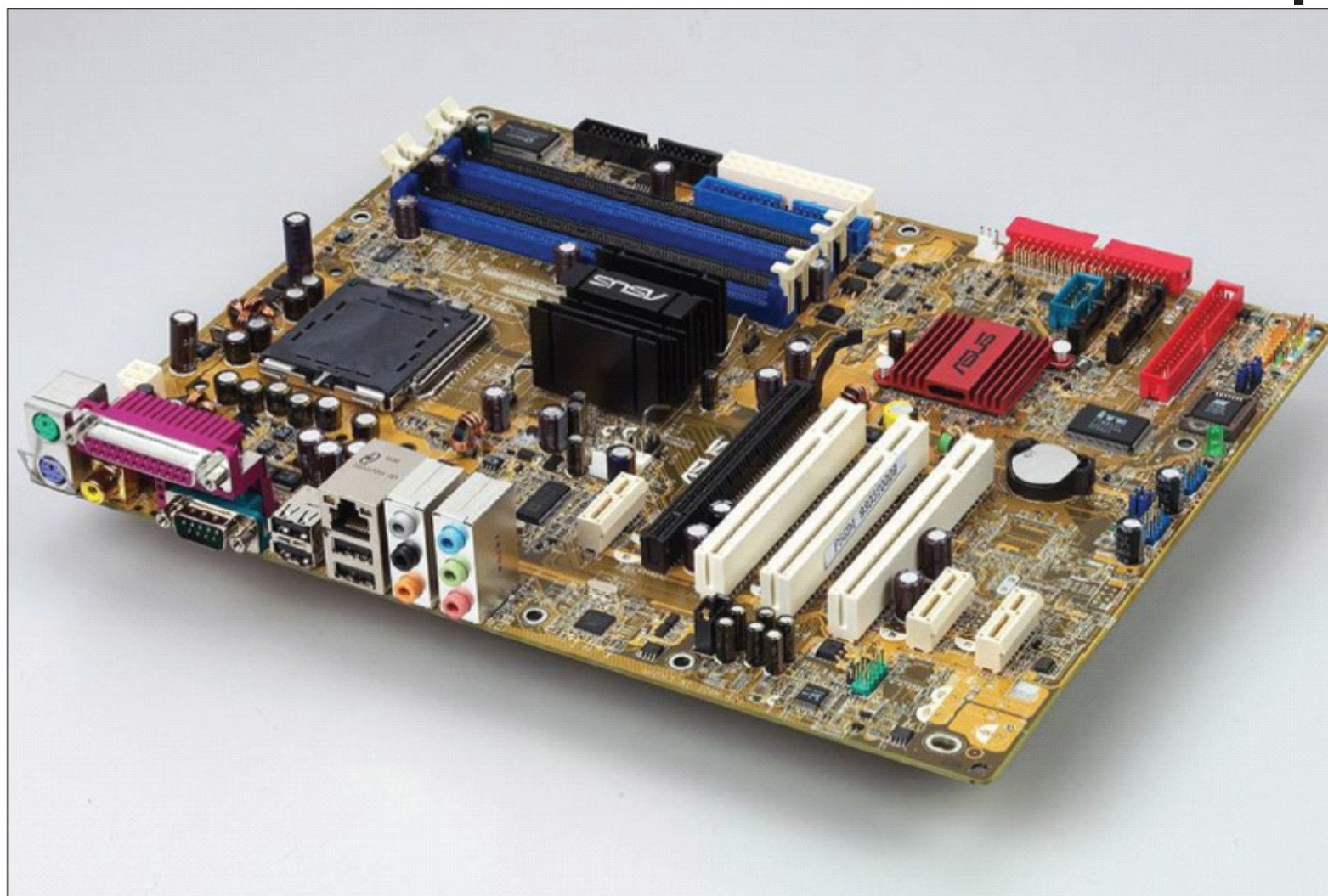


## Dentro la scatola: gli slot di espansione



*Gli slot di espansione servono per inserire le schede degli adattatori, che contengono una serie di componenti elettronici volti a potenziare o espandere le prestazioni del computer. Per esempio, in uno slot si può installare un fax/modem interno.*

*L'unità centrale elaborazione controlla tutte le operazioni aritmetiche e i dati.*



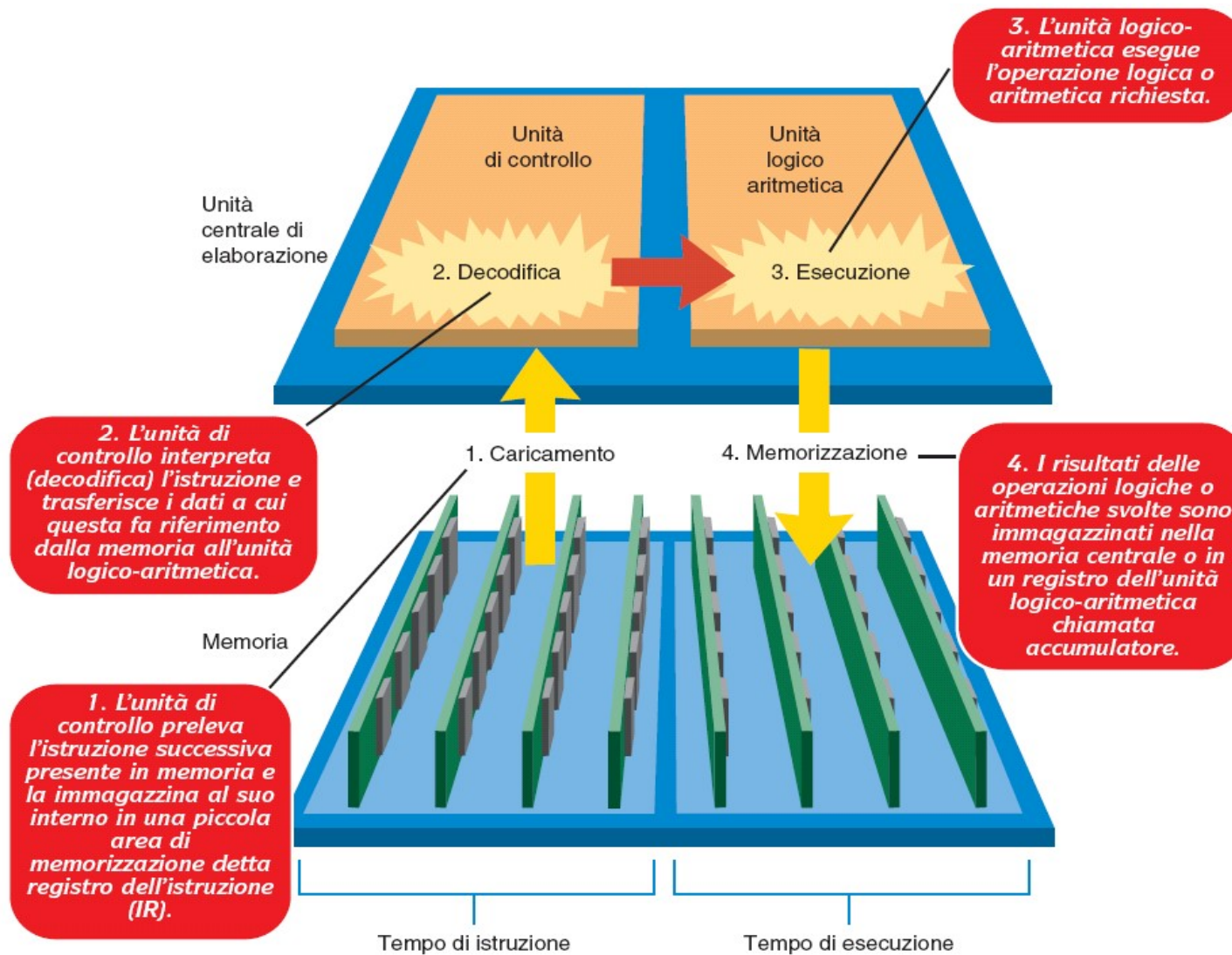
*Una scheda madre – per gentile concessione della Asus*



## L'interazione tra CPU e memoria

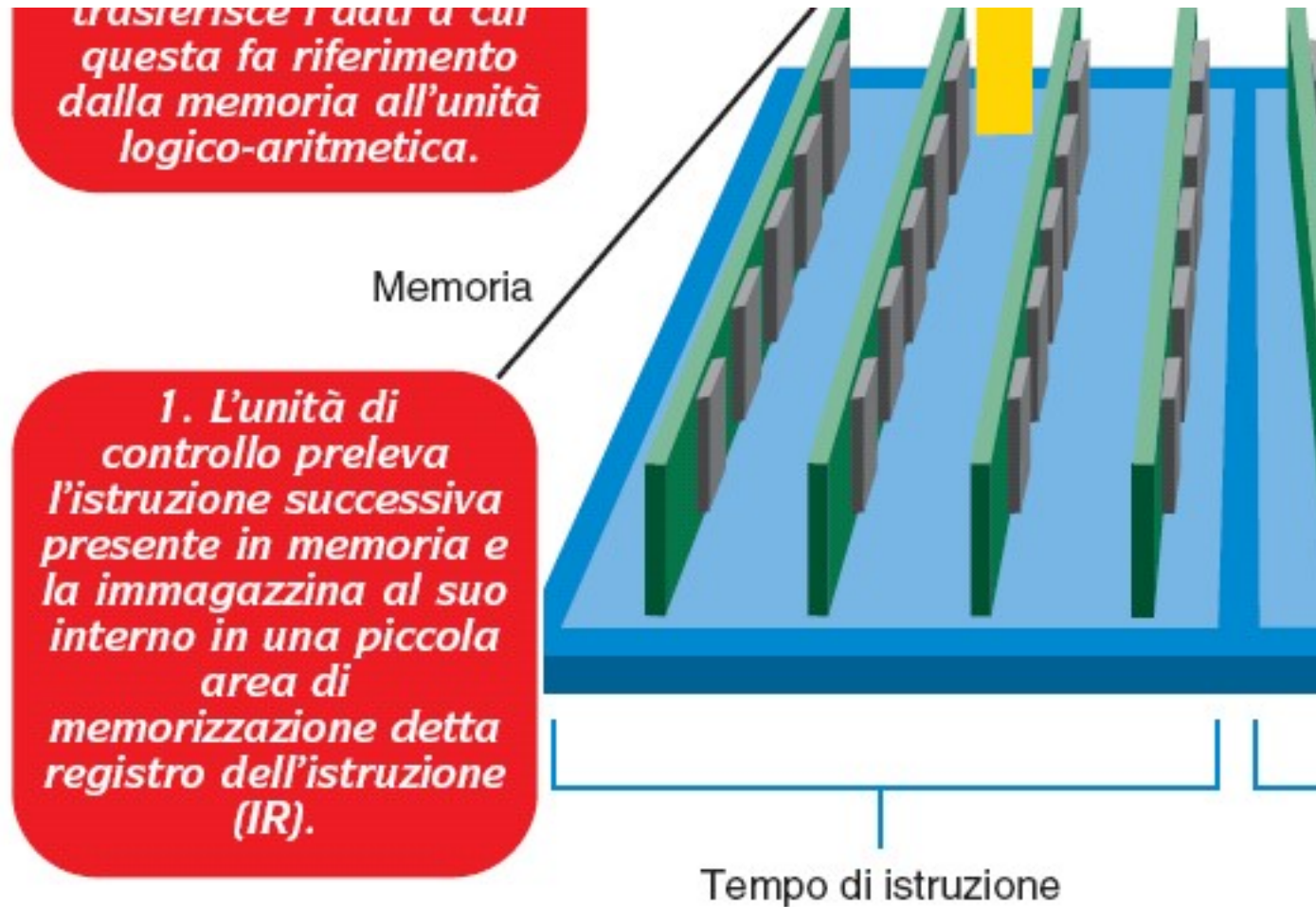
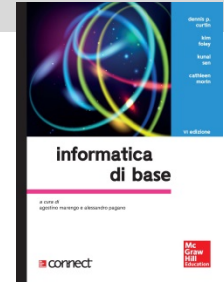


- **CPU (Central Processing Unit)**
  - CU (Control Unit o Unità di controllo)
    - Esegue le istruzioni
    - Coordina le attività del processore
    - Controlla il flusso di informazioni con la memoria
  - ALU (Arithmetic Logic Unit o Unità Aritmetico-Logica)
    - Effettua le operazioni aritmetiche
    - Effettua le operazioni logiche
- **Le due unità collaborano con la memoria in quattro fasi (ciclo della macchina).**



Il ciclo della macchina

## Ciclo della macchina: fase di caricamento

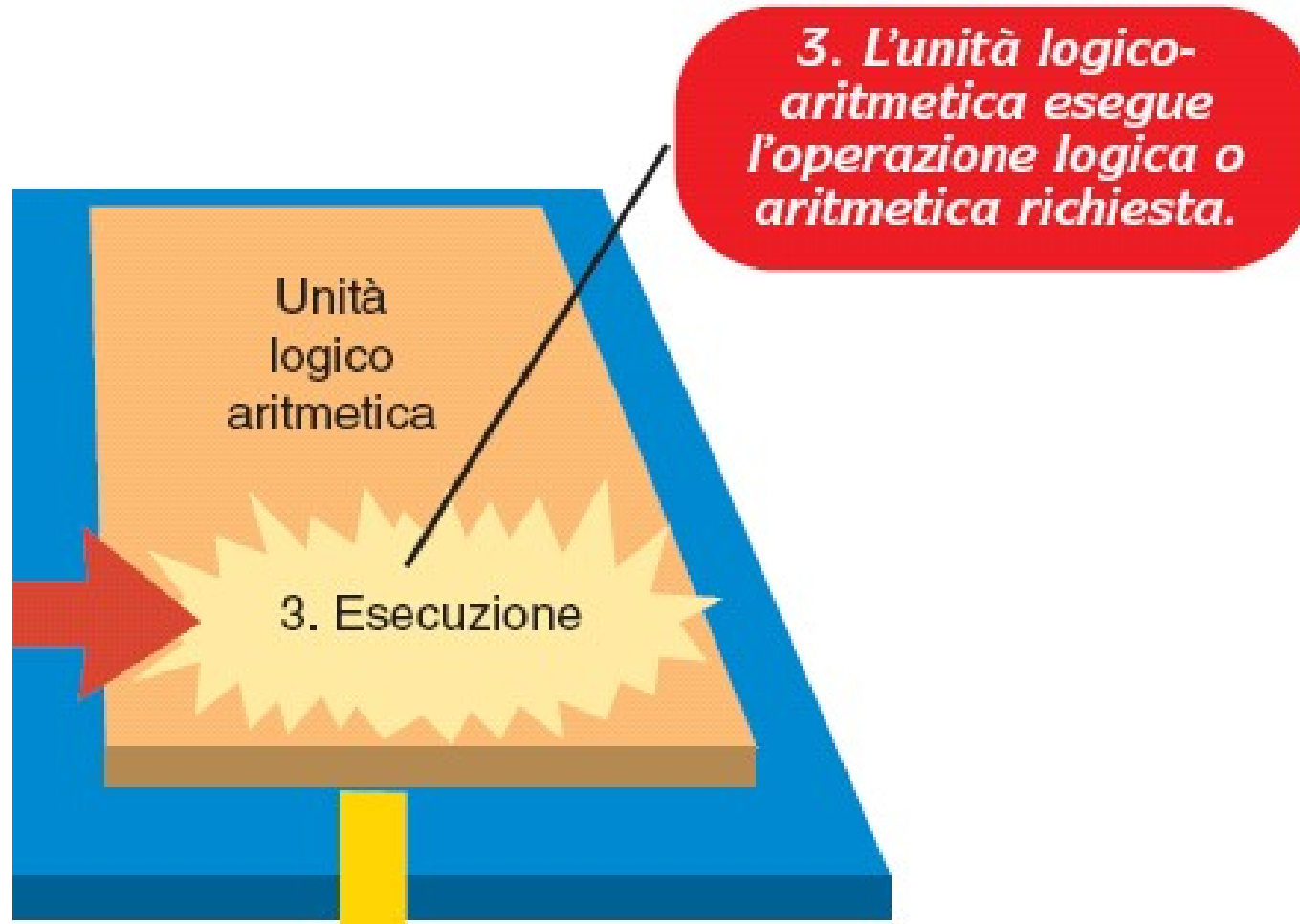


## Ciclo della macchina: fase di decodifica

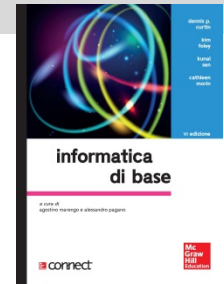
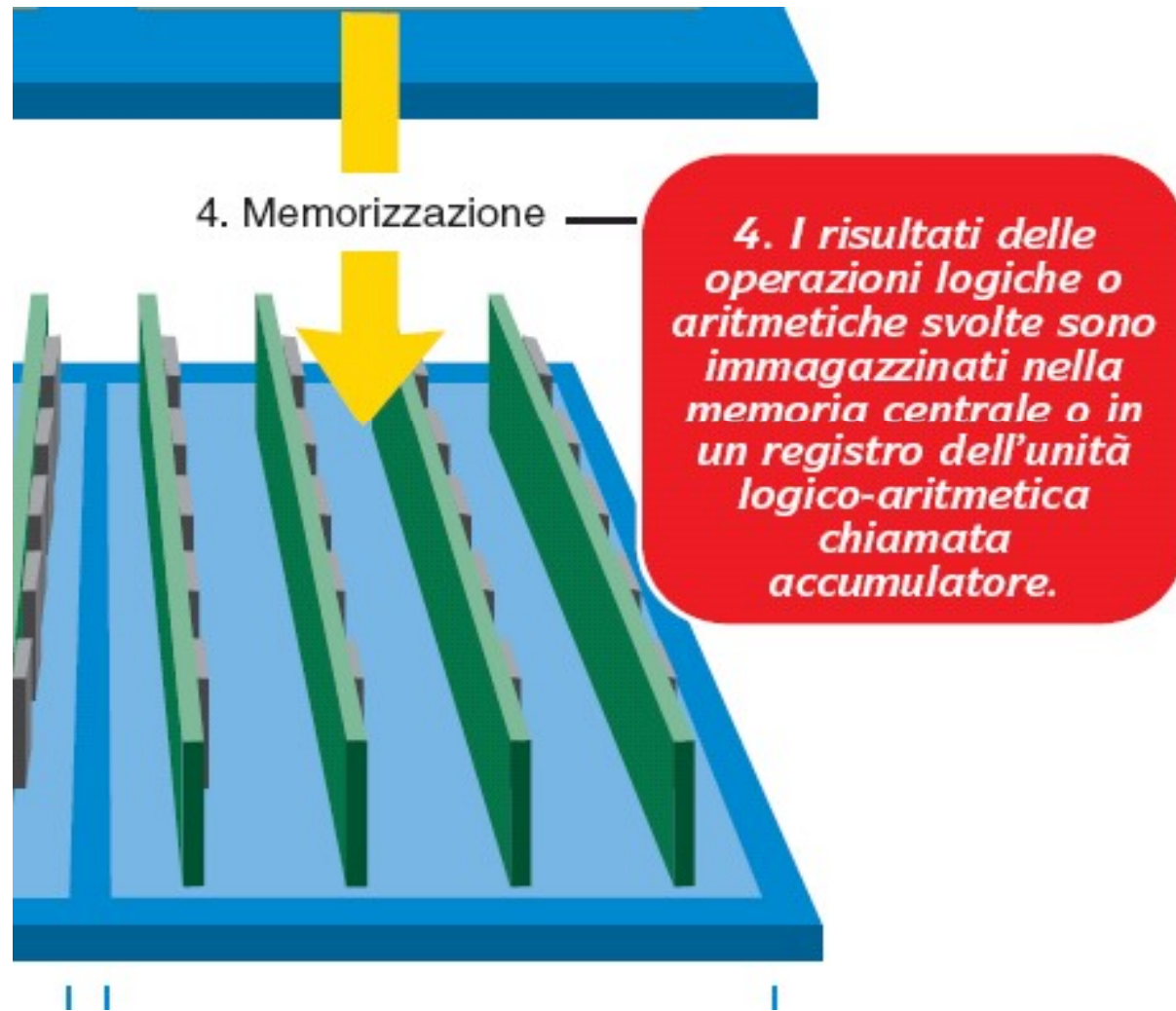




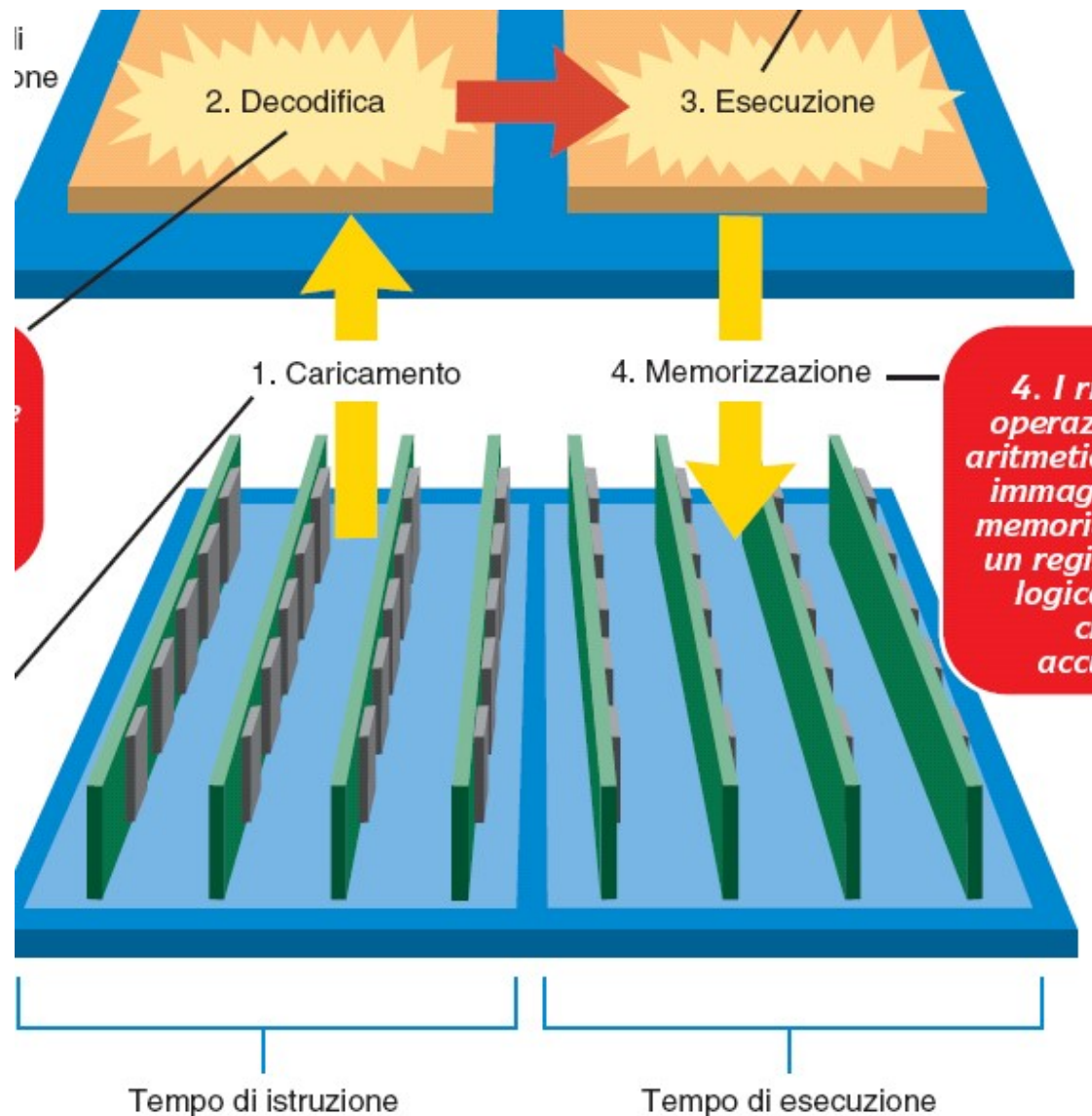
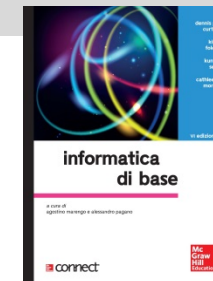
## Ciclo della macchina: fase di esecuzione



## Ciclo della macchina: fase di memorizzazione



# Ciclo della macchina

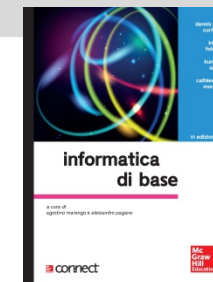


## Il processore



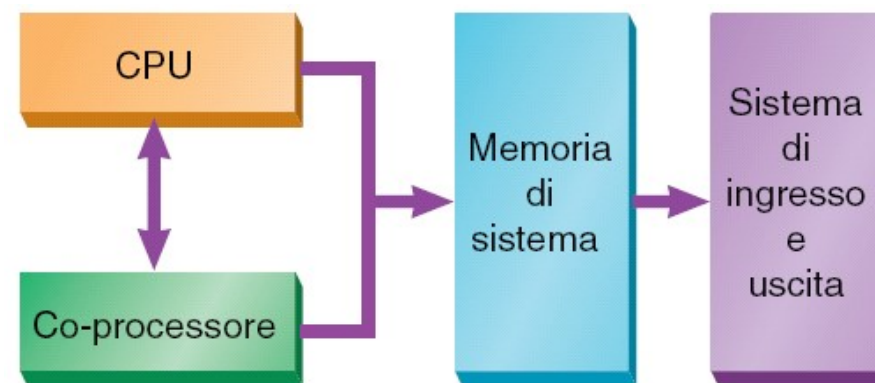
- **C.P.U. Central Processing Unit**
  - Esecutore dei programmi
  - Interagisce con la RAM
- **Composto da:**
  - ALU
  - Unità di Controllo
  - Unità di Memoria



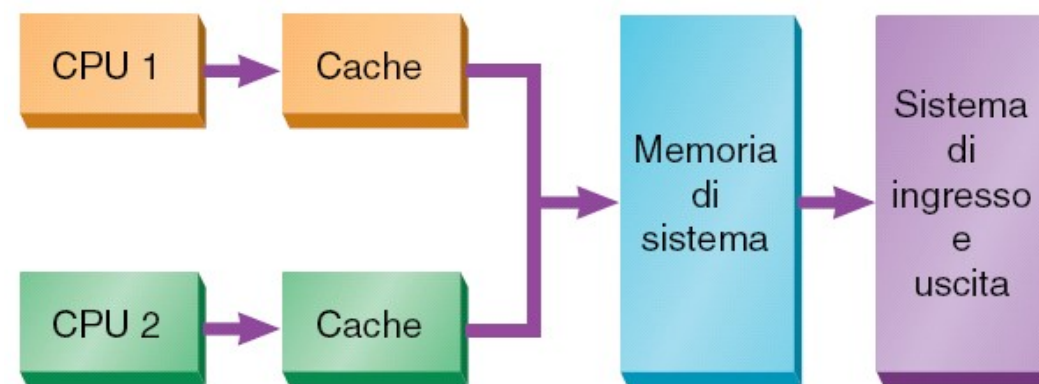


CPU = Central Processing Unit, centro nevralgico del computer.  
 Ogni CPU ha un set di istruzioni diverso, e incompatibile tra diversi produttori. Il software per Windows (CPU Intel) non può funzionare su Apple (CPU Motorola).

In molti computer, e anche all'interno di CPU particolarmente evolute, ci sono coprocessori e multiprocessori.



*Coprocessori*



*Multiprocessori*

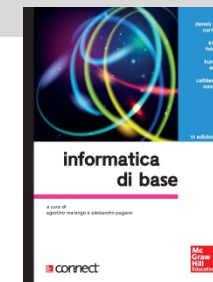
## I coprocessori e i processori paralleli



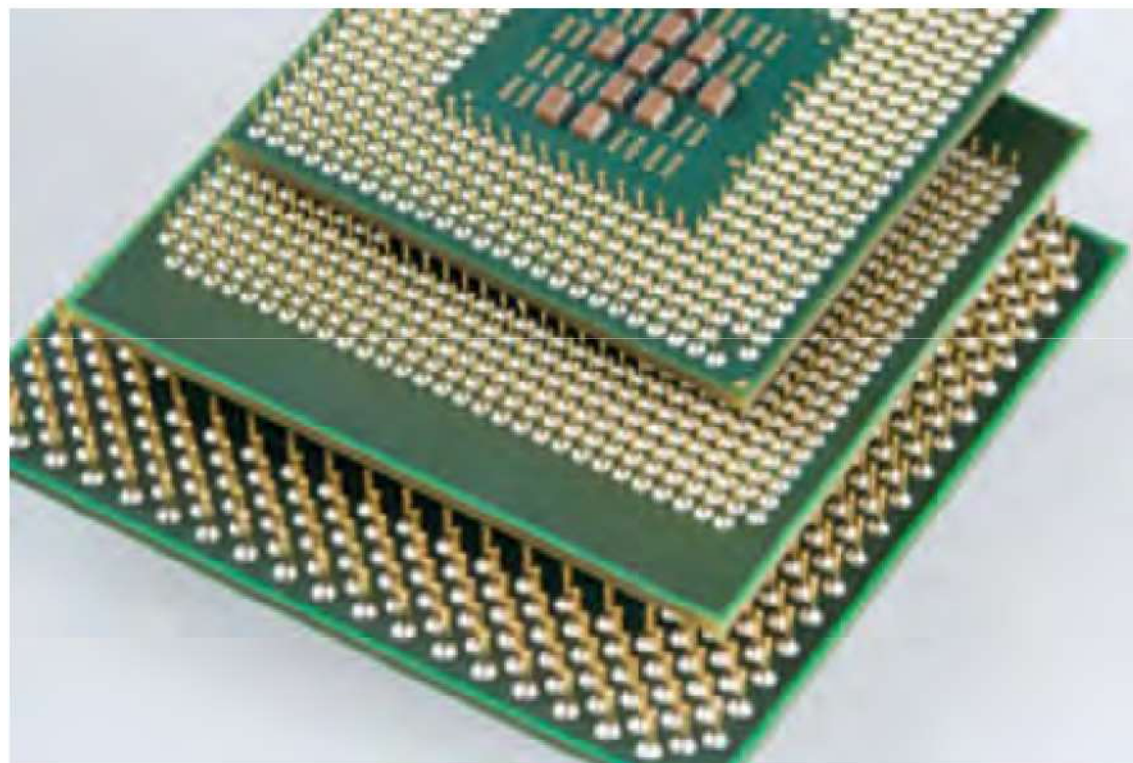
- I Coprocessori sono processori dedicati quali:
  - coprocessore grafico
  - coprocessore audio
  - coprocessore video
  - coprocessore matematico
  - piattaforme mono-processore
- I processori paralleli condividono alla pari le risorse del sistema.



## La CPU: dai numeri ai nomi



- 8086/8088
- 286
- 386
- 486
- 586 → Pentium®
- Multi-core



## Velocità del microprocessore



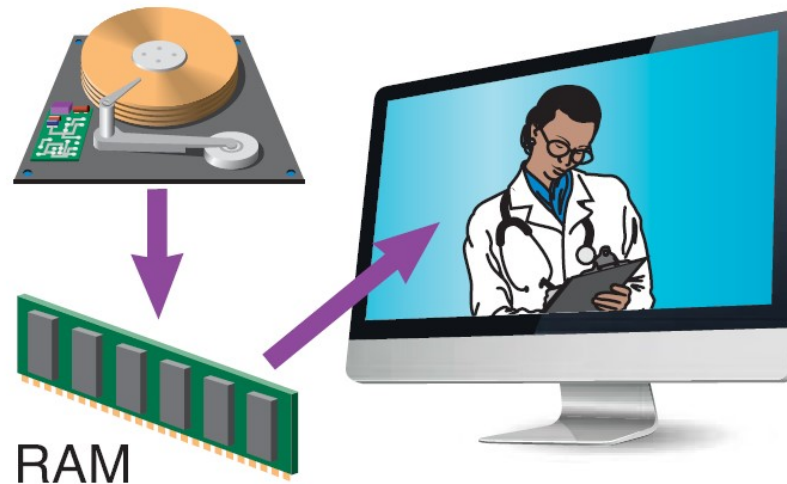
- Velocità del clock
- MegaHertz (MHz o milioni di cicli al secondo)
- GigaHertz (GHz o miliardi di operazioni al secondo)



ROM = memoria non volatile, in cui è memorizzato il firmware.



RAM = memoria volatile, random (accesso libero a qualunque sua parte), in cui si memorizza il software.

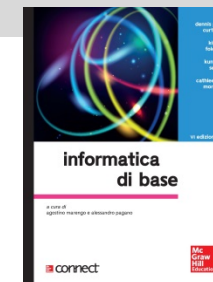


# Memoria RAM

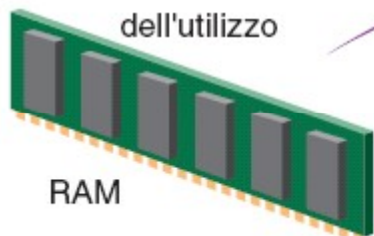


- **Memoria centrale o memoria dell'utente**
- **Memoria volatile**
- **Ad accesso random**
  
- **Primi PC: 128 KB**
- **Oggi: almeno 2 GB**
  - Circa 16.000 volte più grande!

# Funzionamento della memoria virtuale



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



RAM

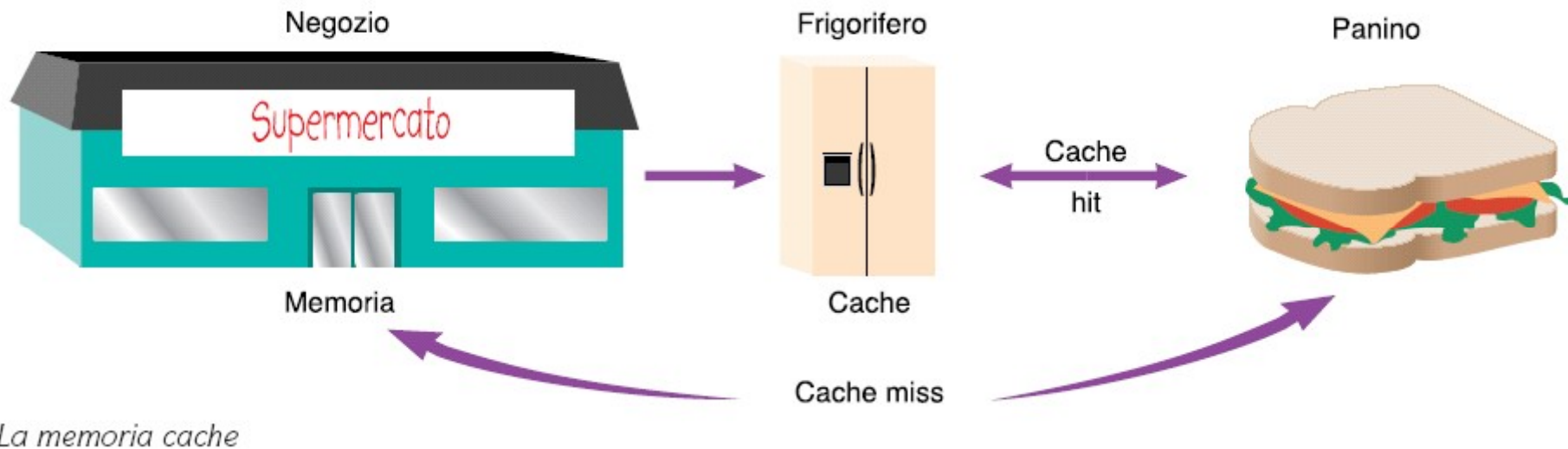
All'occorrenza i dati vengono trasferiti nella RAM, da dove altri dati passano nell'hard disk per fare posto



Hard disk

*La memoria virtuale*

La memoria virtuale rallenta l'esecuzione dei programmi, ma permette di far funzionare applicazioni più grandi della ram stessa.



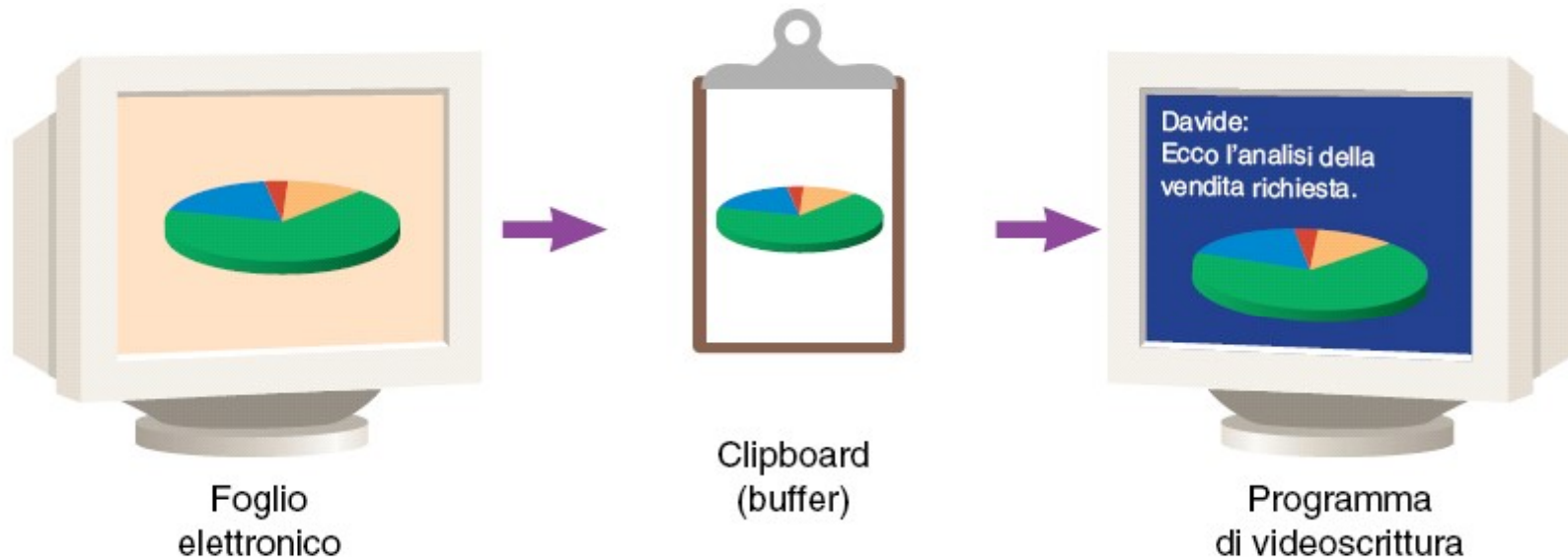
La memoria cache è velocissima ma il suo contenuto può essere superfluo per le operazioni in corso, quindi risultare inutile.

## Memoria cache: livelli



- **Livello 1**
  - Cache interna o primaria o L1
  - interna al processore
- **Livello 2**
  - Cache esterna o secondaria o L2
  - Esterna al processore
  - Chip di memoria

# Funzionamento del buffer (clipboard)



*La clipboard*

La clipboard (o buffer, o appunti) serve a trasferire informazioni tra applicazioni contemporaneamente aperte, oppure in punti diversi della stessa applicazione (es., Copia e Incolla).



Bus, *di sistema* oppure *locale* = connessione fisica tra CPU e le altre componenti del computer; trasporta i dati da un dispositivo all'altro.

Porta *seriale* o *parallela* = connessioni “storiche” dei computer, usate la prima per il modem e la seconda per le stampanti.

Porte USB e Firewire = più recenti, hanno molti vantaggi rispetto alle precedenti connessioni.

- 1) Velocità.
- 2) Si possono connettere in serie, un dispositivo dopo l'altro.
- 3) Riconoscimento automatico dei dispositivi connessi (*Plug & Play*).

