

Le Memorie

Classe III Telecomunicazioni
Sistemi e Reti

Prof. Tullio Parcesepe

In questa lezione impareremo:

- **La memorizzazione delle informazioni**
- **Tipi differenti di memorie**
- **Gli indirizzi delle celle di memoria**
- **La gestione della memoria**
- **Le memorie flash**

Libro: pagine da 32 a 37

Quanti tipi di memoria conosciamo?



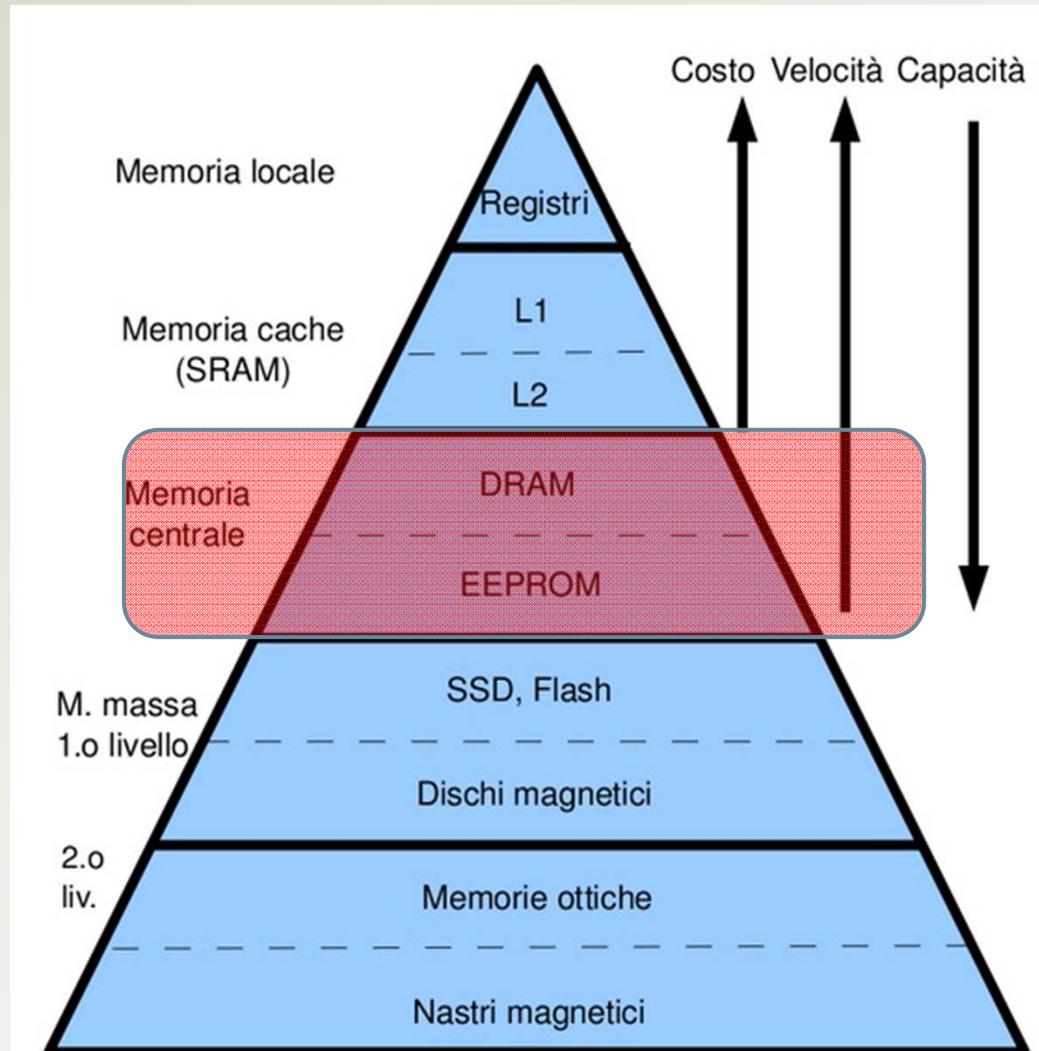
Memoria Principale



Memorie di massa o secondarie

Basta così? E i REGISTRI?..e la CACHE?

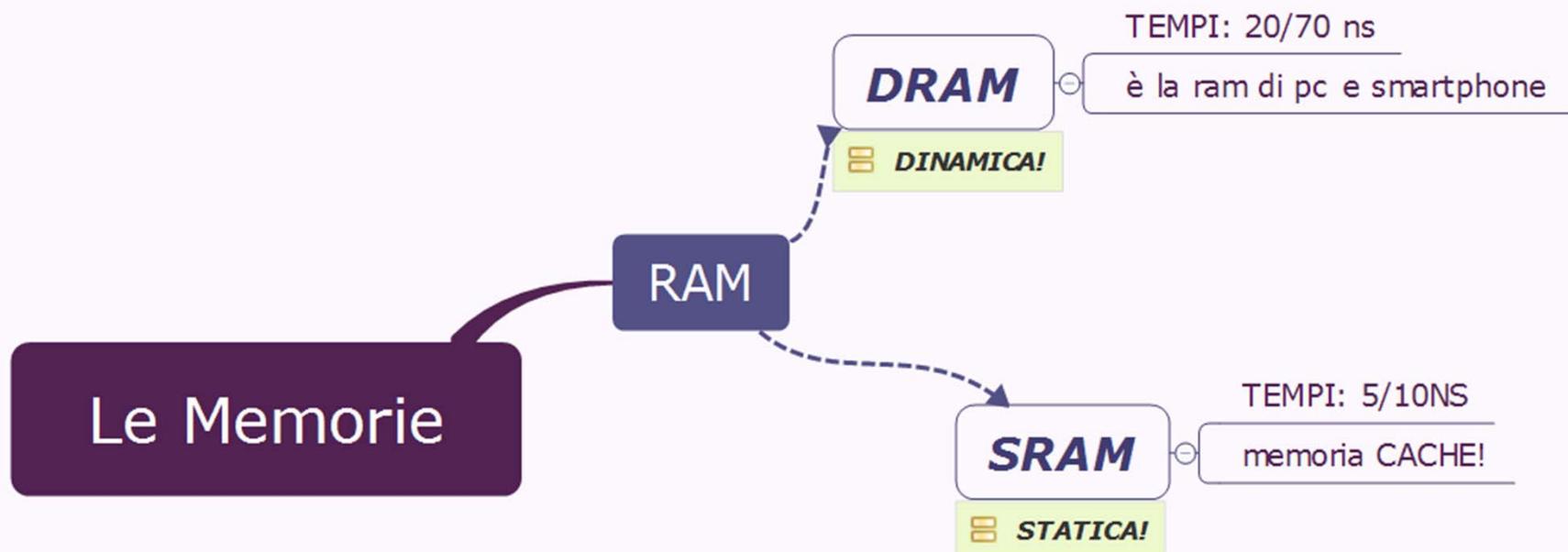
Esiste una GERARCHIA!



I tipi di memoria: RAM statica e dinamica

- Le RAM si suddividono in due sottocategorie chiamate RAM **dinamica (DRAM)** e RAM **statica (SRAM)**
- **RAM dinamica** mantiene i dati solo se alimentata (tempo di accesso tra 20 ns e 70 ns) con rinfresco (**refresh**)
- **RAM statica** = tempo di accesso da 5 a 10 ns, **no refresh** (di solito usata nella **cache**)

Schematizziamo subito

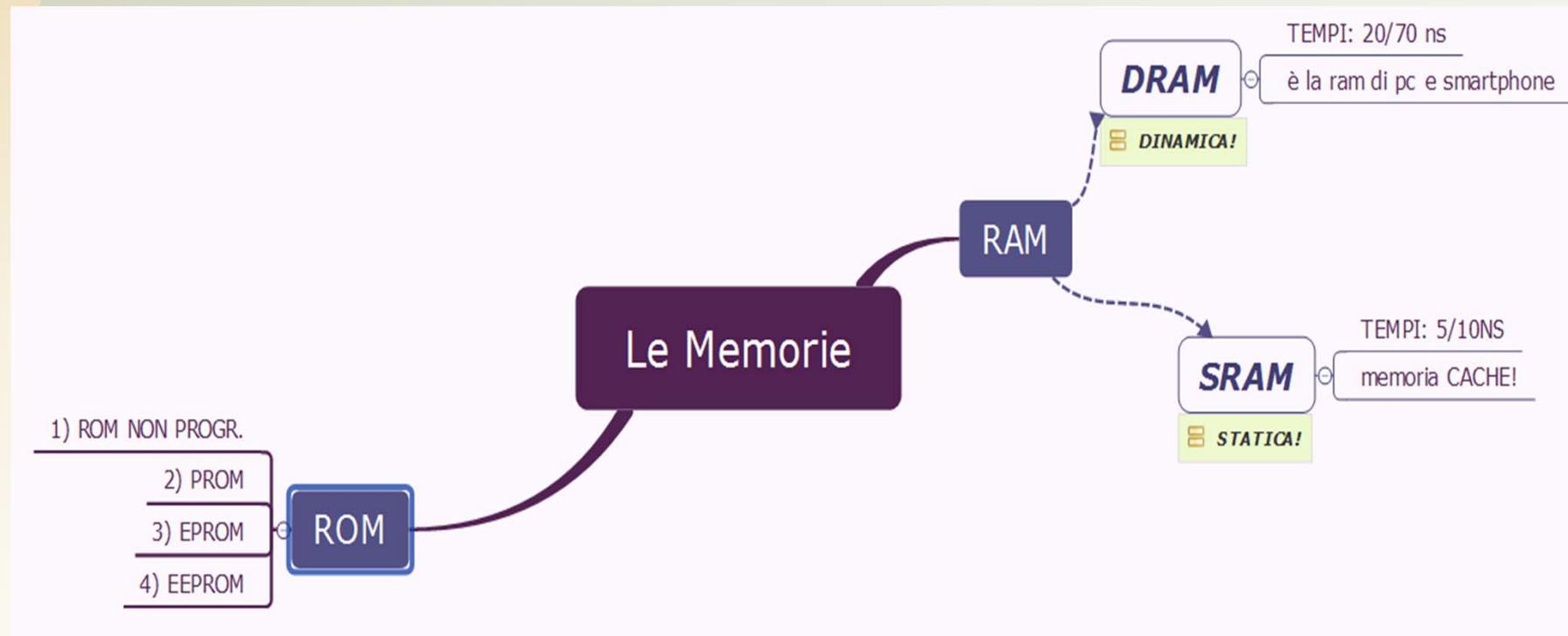


Memoria ROM (Read Only Memory)

Sono memorie accessibili solo in lettura e conservano l'informazione anche senza corrente (memorizzano configurazioni importanti: ripristino android/firmware)

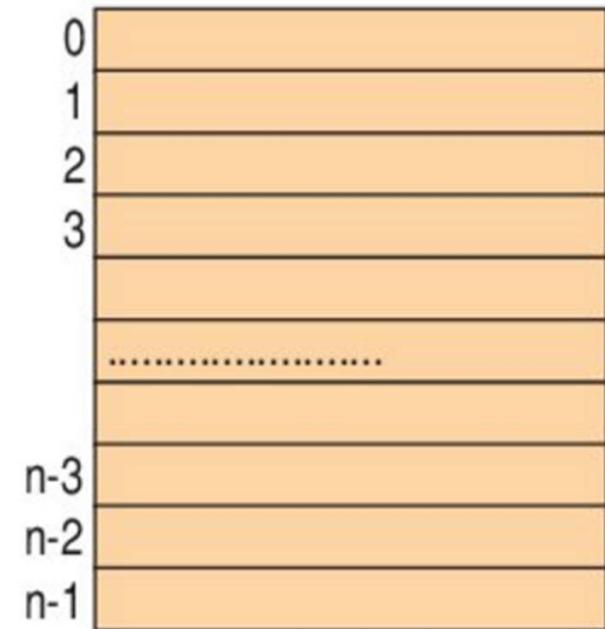
- **ROM** non programmabili
- **PROM** (**Programmable ROM**)
- **EPROM** (**Erasable Programmable ROM**)
- **EEPROM** (**Electrical Erasable Programmable ROM**)

Aggiungiamo alla mappa



Gli indirizzi delle celle di memoria

- L'**indirizzo** di ciascuna cella è definito dalla posizione relativa della cella rispetto alla prima cella
- **Spiazzamento** o **displacement** = spostamento necessario per raggiungere la cella desiderata iniziando dalla prima, che ha indirizzo 0
- Celle di memoria identificate univocamente da un indirizzo (**memory address**)



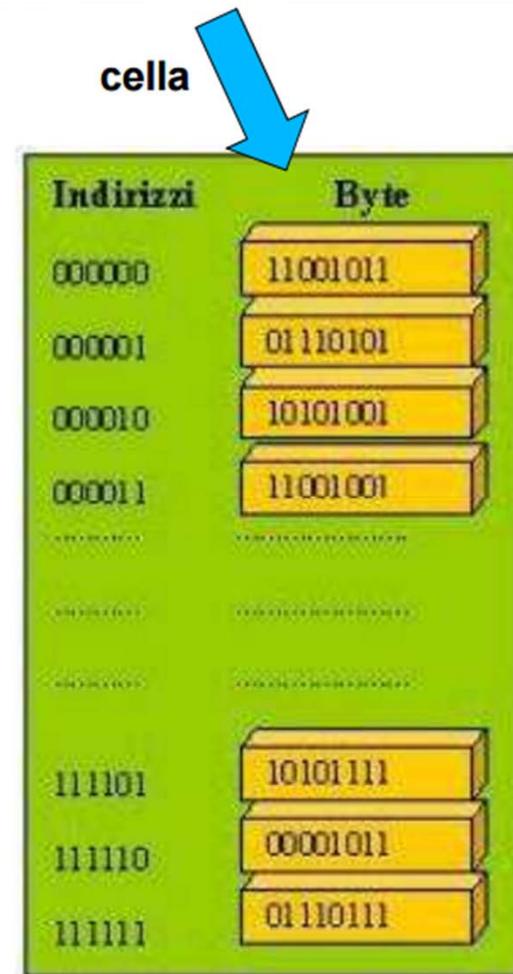
..ragioniamo in bit

COSA CONTIENE?

- **NUMERI** : Dati numerici cioè numeri binari
- **CARATTERI** : Dati alfanumerici cioè in codice ASCII
- **COMANDI** : Un comando in linguaggio della macchina

ADDRESS o INDIRIZZO.

Le celle sono numerate : 0,1,2,3,4,5,6 ed il numero della cella è detto indirizzo.... (address). L'indirizzo è però espresso in **binario** od in **esadecimale**.



Spazio di indirizzamento

- È il numero di bit necessario a indirizzare tutte le celle di memoria.

Se ho 256 celle di memoria ... bastano 8 bit

Se ho 65536 celle di memoria ... servono 16 bit

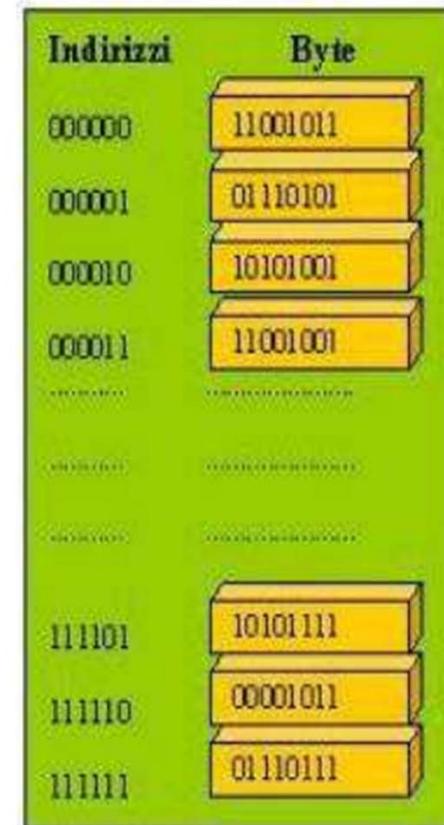
Spazio di indirizzamento

Esempio : la memoria in figura.

In questo caso gli **indirizzi** sono a **6 bit** e pure lo **spazio di indirizzamento** è a 6 bit.
e quindi vuol dire che ci sono **$2^6 = 64$** celle di memoria.

Inoltre ogni **cella** (word) contiene poi **1 byte**
quindi vuol dire che la cella può immagazzinare

- un **numero** di 8 bit
- oppure **1** solo carattere **ASCII** ,
- oppure **1** istruzione semplice .



Calcolare rapidamente lo spazio di indirizzamento

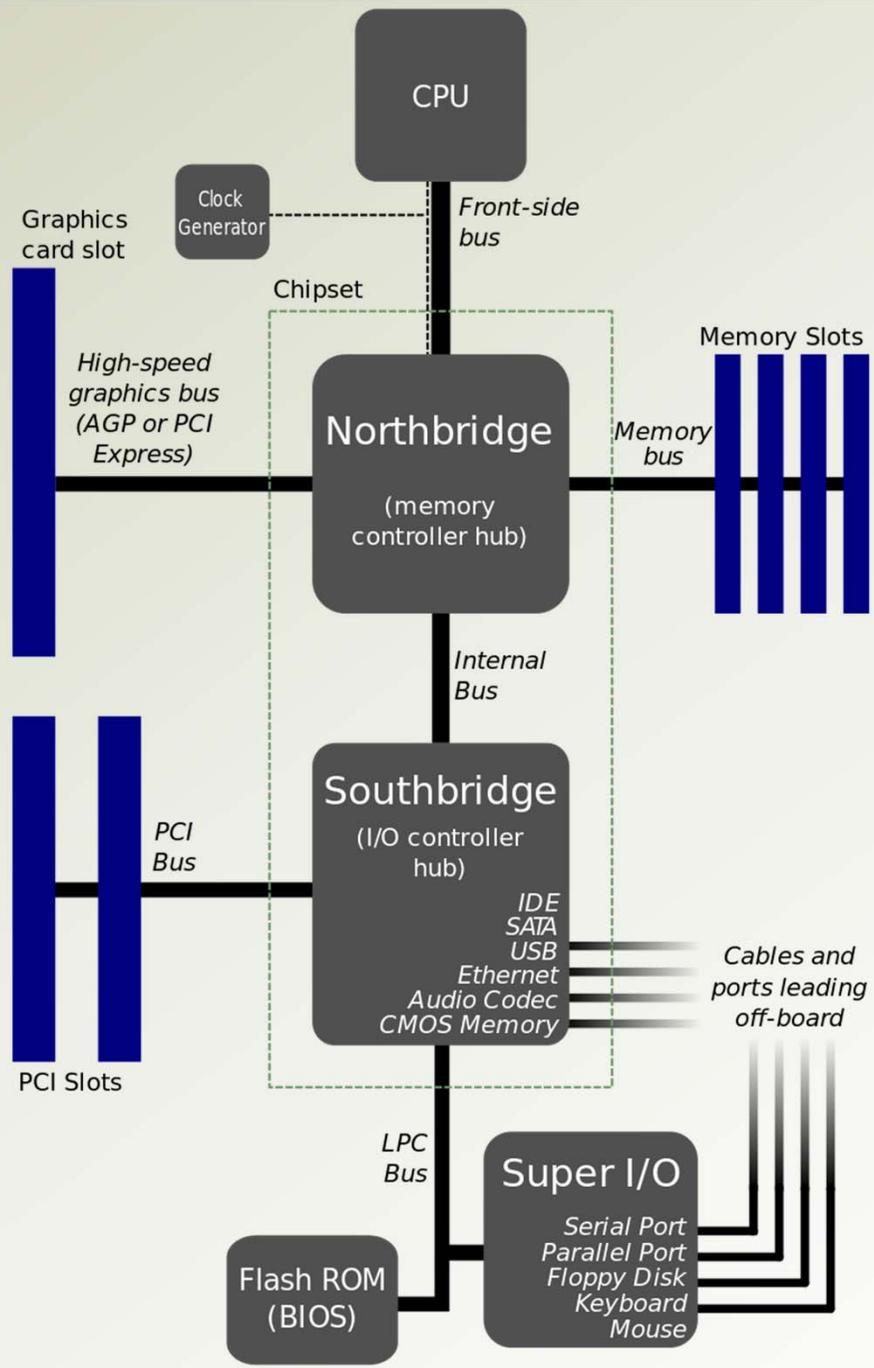
- ▶ 10 bit = 1 k celle (1024 celle)
- ▶ 16 bit = 64 k celle (65536 celle)
- ▶ 20 bit = 1 M celle (RAM dei processori fino all'80286)
- ▶ 32 bit = 4 G celle (RAM dei processori fino al Pentium)
- ▶ 36 bit = 64 G celle (RAM dei processori fino al Pentium IV)
- ▶ 40 bit = 1 T celle (RAM dei processori fino all'Athlon 64)
- ▶ 64 bit = 16 E celle (RAM dei processori fino al Core i7)

Unità di misura per la memoria

Unità di misura	simbolo	Equivale a	cioè
byte	B	8 bit	Un carattere alfanumerico
kilobyte	KB	2^{10} byte	Un terzo di pagina di testo
megabyte	MB	2^{20} byte	Circa 300 pagine di testo
gigabyte	GB	2^{30} byte	Circa 300000 pagine di testo
terabyte	TB	2^{40} byte	Circa 300 milioni di pagine di testo

La gestione della memoria nel PC

- Nell'architettura Von Neumann, il canale di comunicazione tra CPU e memoria è il punto critico del sistema in quanto la tecnologia consente di realizzare processori sempre più veloci e memorie sempre più capienti, tuttavia la velocità di accesso delle memorie non è adeguato alla crescita repentina delle CPU (**collo di bottiglia**)



Soluzione con memoria a velocità differenti



Cache memory

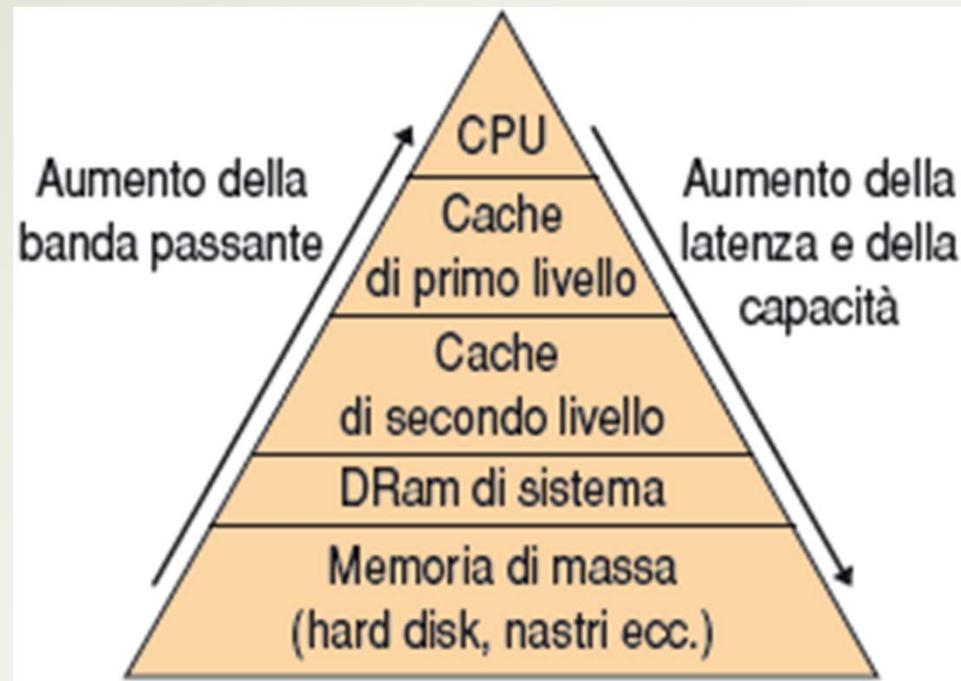
- La memoria del PC è organizzata a **livelli gerarchici**: ogni livello è caratterizzato da una dimensione crescente e da un tempo di accesso decrescente
- La RAM è molto più **lenta** della CPU, per migliorare le prestazioni vengono combinati tipi di memoria veloce con tipi di memoria più capienti ma lente (**Cache**)

..un problema vecchio di 50 anni

- Sin dagli anni '70 la ram è sempre stata più lenta della CPU e la differenza è aumentata col tempo
- È inutile progettare CPU sempre più veloci se poi il reperimento dei dati/istruzioni è lento!
- Le **CACHE** sono memorie piccole ma veloci (64 KB – 256KB – 2/6MB), vicino alla CPU e **contenenti parte delle informazioni della RAM**

Livelli di Cache

- La **CPU** legge e scrive i dati direttamente sulla cache di primo livello, quindi via via sulle memorie inferiori



Cache Hit e Cache miss

Quando il processore deve svolgere un'operazione, cerca prima in **CACHE** e poi in RAM i dati di cui ha bisogno:

- se li trova in CACHE apre il programma (o esegue un'operazione) molto più velocemente!
Il processore riesce nel suo intento, ha successo e, in linguaggio tecnico, si verifica un **CACHE HIT!**
- Se, al contrario, non trova i dati in cache, deve accedere alla memoria ram (più lenta della cache) e impiegherà un po' di tempo in più per svolgere l'operazione. In questo caso fallisce, ovvero genera un **CACHE MISS!**

(..quando fallisce, il processore paga anche un pegno in termini di tempo... perché???)

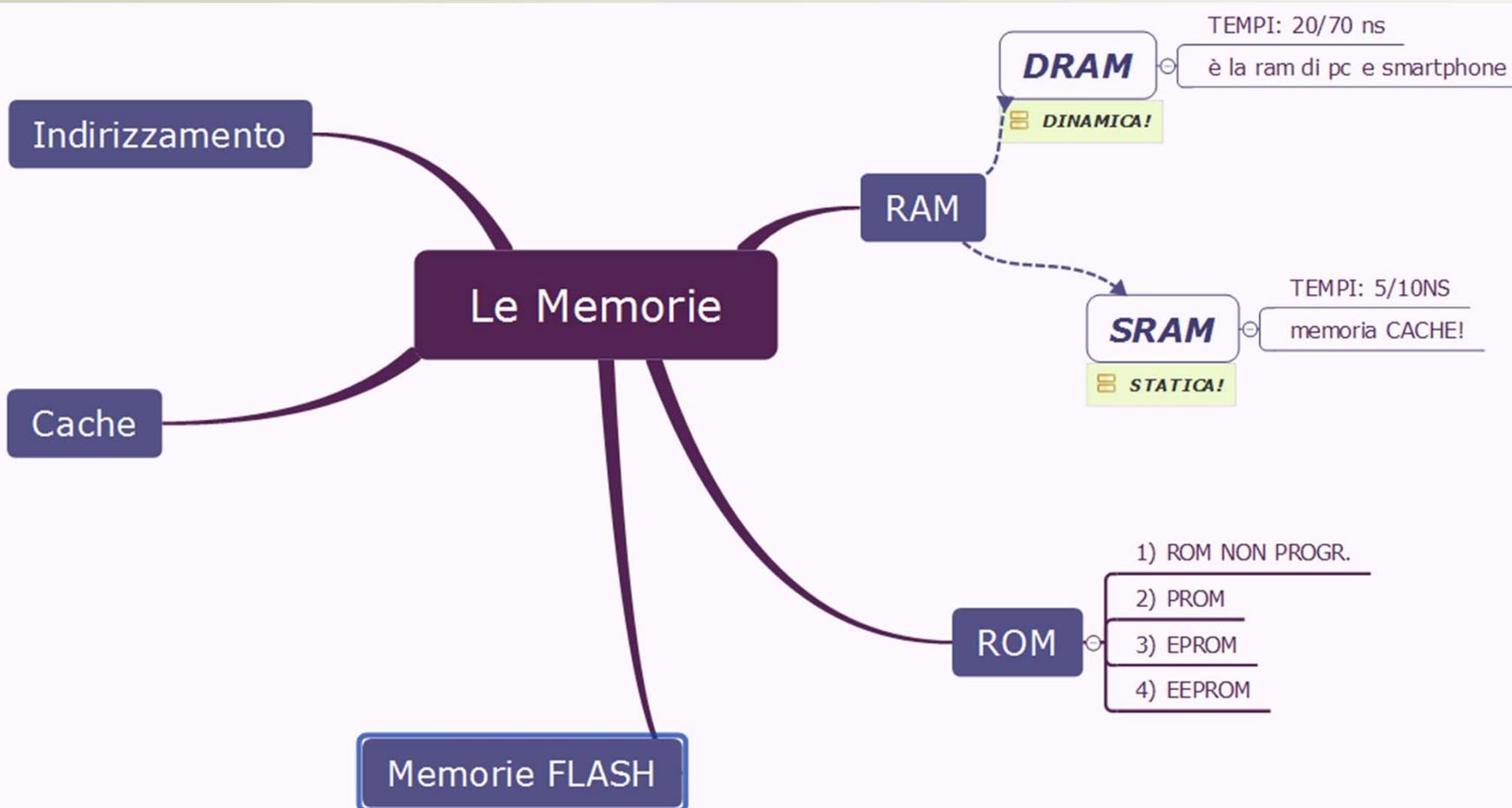
Prestazioni della memoria

- **Latenza**: è il tempo necessario, in cicli macchina dell'**FSB**, affinché un dato venga letto dalla memoria (**RITARDO!**)
- **Banda di trasferimento dati**: misura la quantità di informazioni al secondo che vengono trasmesse con la memoria
- **Frequenza di funzionamento**: blocchi di dati trasferiti al secondo (Es. 4GB DDR3 1600Mhz)

Memorie Flash

- La **memoria flash** è un tipo di memoria non volatile che può essere programmata elettronicamente
- Due tipi:
 - **NOR** (**Negated OR**)
 - **NAND** (**Negated AND**)
- Usate principalmente per dischi **SSD**, **pen drive**, memorie **Secure Digital**

Mappa finale INCOMPLETA!



Mappa creata col software XMIND.. a buon intenditor... ;)

Gerarchia della memoria

