Lezione 1 – Principio di funzionamento della memoria *cache*

Architettura degli elaboratori

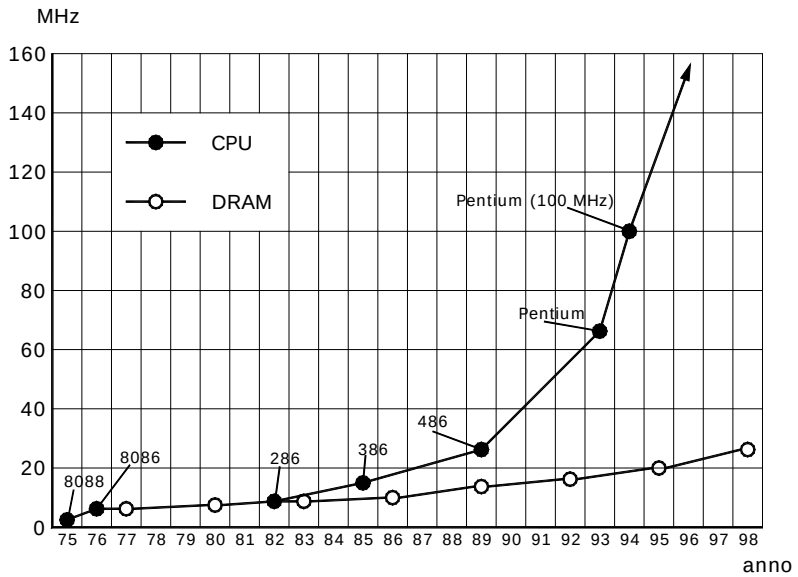
Modulo 5 - Principali linee di evoluzione architetturale

Unità didattica 1 - Memoria *cache* e gerarchia di memoria

Nello Scarabottolo

Università degli Studi di Milano - Ssri - CDL ONLINE

Un grafico di qualche anno fa...

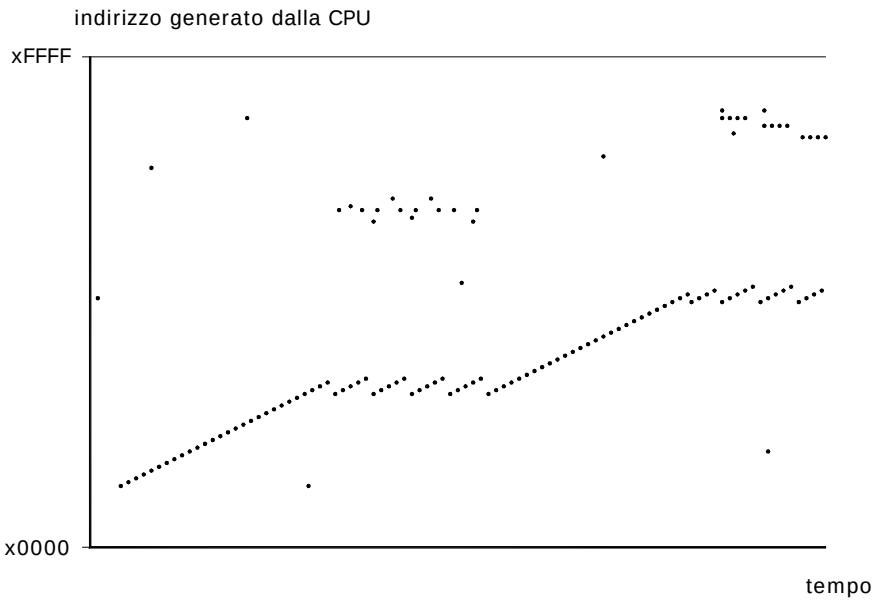


C'è qualcosa che non torna...

- A partire dal 1982, la frequenza di lavoro delle CPU della famiglia *i*86 cresce decisamente più della frequenza di lavoro dei chip di DRAM (cioè dei costituenti la memoria di lavoro).
- Verso la fine degli anni '90, il *gap* è di un ordine di grandezza (un fattore 10).

Come è possibile che i PC funzionassero con una memoria di lavoro 10 volte più lenta della CPU ???

Cosa serve davvero alla CPU?



Principio di località

Se all'istante *t* la CPU genera l'indirizzo di memoria xNNNN, è molto probabile che nell'immediato futuro generi di nuovo lo stesso indirizzo xNNNN o indirizzi vicini ("locali") all'indirizzo xNNNN.

- Località **spaziale**:
  - il fetch delle istruzioni procede in celle consecutive;
  - i programmi sono organizzati in moduli, con le variabili del singolo modulo memorizzate vicine.
- Località **temporale**:
  - l'essenza della programmazione sono i **cicli**: istruzioni e variabili usate nei cicli vengono "ripassate".

Sfruttiamo il principio di località

Lavoriamo su base statistica.

Quando la CPU genera un indirizzo di memoria, portiamo il contenuto della cella richiesta e un certo numero di celle vicine (blocco) in una memoria:

- più veloce della DRAM;
- ovviamente più piccola, perché più costosa da realizzare.

Chiamiamo questa memoria *cache*:

- deriva dal francese *caché* (nascosto) perché la sua esistenza non è nota né al programmatore, né alla CPU;
- serve solo a velocizzare gli accessi a memoria.

Perché dal 1982?

Processore	Anno	Costo	MIPS iniziali	MIPS massimi	n° transistor
8086	1978	-	0.33	0.75	29 K
286	1982	$ 8	1.20	2.66	134 K
386	1985	$ 91	5.00	11.40	275 K
486	1989	$ 317	20.00	54.00	1.2 M
Pentium	1993	$ 900		112.00	3.1 M

Ogni 4 anni una nuova generazione.

Lo stesso processore raddoppia le proprie prestazioni:

⇒ miglioramento **tecnologico** (in orizzontale)

Tra una generazione e l'altra, triplica la complessità:

⇒ miglioramento **architetturale** (in verticale).

Come usiamo i transistori in più?

Per esempio per realizzare una memoria *cache* a bordo del processore, che lavora alla sua stessa frequenza di clock:

- **cache L1** (di primo livello) - qualche KB.

La frequenza del processore, però, è cresciuta ancora, e la sua differenza rispetto alla DRAM si è enfatizzata:

- **cache L2** (di secondo livello) esterna al processore - qualche centinaio di KB.

E se le cose degenerano...:

- **cache L3** (di terzo livello) esterna al processore - qualche decina di MB!

In sintesi...

Grazie alla località degli accessi a memoria da parte della CPU:

- possiamo copiare in una memoria ad alte prestazioni (***cache***) le celle richieste, che hanno maggiore probabilità di essere usate di nuovo;
- possiamo creare una **gerarchia** di *cache* via via più grandi e più lente man mano che ci allontaniamo dalla CPU e ci avviciniamo alla memoria di lavoro;
- le celle con più alta probabilità di riutilizzo sono ricopiate nella *cache* a bordo della CPU;
- tutte le celle disponibili sono presenti in memoria di lavoro.

Chiusura

Fine della lezione