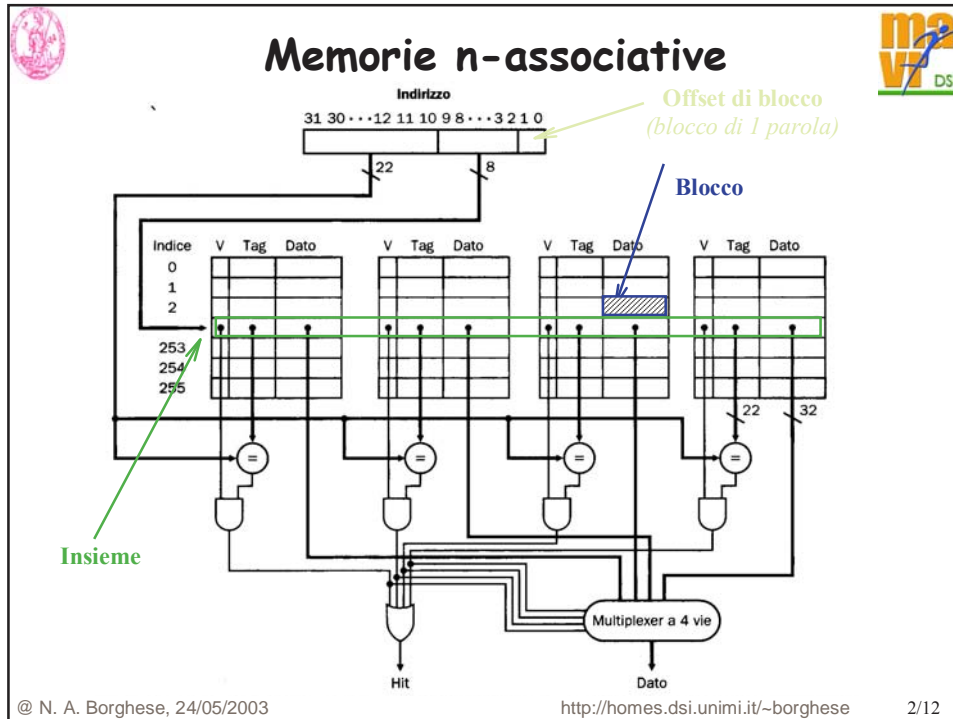


Memoria

Architettura degli Elaboratori e delle Reti, Turno I



Alberto Borghese
Università degli Studi di Milano
Dipartimento di Scienze dell'Informazione
email: borghese@dsi.unimi.it





Capacità della cache n-associativa



Blocco: #parole(byte) lette/scritte contemporaneamente in cache, “parola” della cache.

Insieme: #blocchi di cui è costituito ciascun elemento della cache.

Cache: #Insiemi.

Capacità della cache:

$\#parole = \#Insiemi * (\#blocchi / insieme) * (\#parole / blocco)$



Accesso a cache ad n-way



INDICE - seleziona l'*insieme* che potrebbe contenere il blocco desiderato. L'insieme è costituito da diversi blocchi. Se la parola di RAM è memorizzata in cache, si trova in uno dei blocchi che costituiscono questo insieme. L'indice è costituito da k bit, dove $k = \log_2(\#Insiemi)$.

TAG – contiene il blocco della RAM da cercare. Cerca il tag di RAM all'interno dei TAG dei blocchi che costituiscono l'insieme.

Il comparatore pilota anche il MUX che trasferisce in uscita il contenuto dei banchi della cache.



Criteri di sostituzione di un blocco



Dove inserisco il blocco letto dalla memoria?

Al posto del blocco letto meno recentemente (least recently used, LRU).



Criteri di progettazione



Cache primaria: massimizzo Hit time.

Cache secondaria: minimizzo Miss penalty.



Dove si può posizionare un blocco di RAM in cache?



Corrispondenza diretta: in un'unica posizione.
Ciascun insieme è costituito da un unico blocco
 $\#insiemi = \#blocchi$.

Completamente associative: in n posizioni (n blocchi).
Ciascun insieme è costituito da 1 blocco.
n insiemi.

N-associative: in m posizioni (m grado di associatività).
Ciascun insieme è costituito da m blocchi.
 $\#insiemi = n / m$.



Come si trova un blocco di RAM in cache?



Corrispondenza diretta: indicizzazione.
Controllo del tag del blocco (1 comparazione).

Associativa: ricerca in tutti gli elementi della cache.
n comparazioni: controllo di tutti i tag.
La memoria virtuale è di questo tipo (tramite la *Page Table*).

N-associativa: ricerca negli m blocchi dell'insieme,
indicizzazione dell'insieme.
m comparazioni.



Quale blocco viene sostituito in caso di miss?



Corrispondenza diretta: non c'è scelta.

Casuale.

LRU (Least Recently Used).

Facile con 1 bit. Non è detto che sia la soluzione migliore!



Come funziona la scrittura?



Write-through. Scrittura in cache e contemporaneamente in RAM.

Write_buffer per liberare la CPU.

Write-back. Scrittura ritardata. Scrivo quando devo scaricare il blocco di cache.

Vantaggiosa con cache n-associative.

Alla RAM trasferisco l'insieme dei blocchi.



Principi sulla Memoria Virtuale



Due motivazioni:

- Estensione della memoria fisica. Maggiore quantità di memoria.
- Gestione del multi-tasking. Negli anni '90 *overlay* definito nel linker, ora trasparente tramite il gestore della memoria virtuale.

Ogni programma ha il suo spazio di indirizzamento.
Mappatura dello spazio di indirizzamento nella memoria fisica
(*memory mapping* tramite la *page table*).



Il meccanismo della memoria virtuale



Memoria virtuale (estensione su disco) è concettualmente analoga alla cache.

Blocco di memoria → Pagina.
Miss → Page Fault.