



# Le memorie Cache associative

Prof. Alberto Borghese  
Dipartimento di Scienze dell'Informazione  
[alberto.borghese@unimi.it](mailto:alberto.borghese@unimi.it)

Università degli Studi di Milano

Riferimento Patterson: 5.2, 5.3



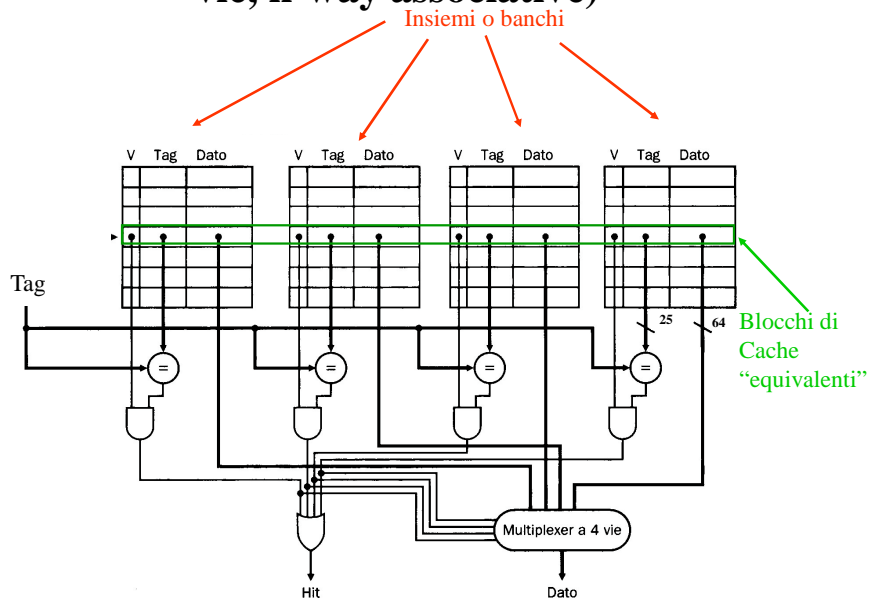
## Sommario

**Memorie n-associative**

Esempi



## Memorie n-associative (o associative a n-vie, n-way associative)



A.A. 2015-2016

3/19

<http://horghese.di.unimi.it/>



## Memorie n-associative



n-associative o set associative o a n vie.

La memoria è suddivisa in n insiemi, o banchi, ciascuno di k linee, posti in parallelo.

**Blocco (linea di cache):** #parole (byte) lette/scritte contemporaneamente in cache, "parola" della cache.

**Insieme (banco):** cache elementare.

**Cache:** è l'insieme dei banchi più i circuiti che li gestiscono.

**Capacità della cache:** #parole = #Insiemi \* (#blocchi / insieme) \* (#parole / blocco).

La corrispondenza tra Memoria Principale e linea di un banco è a mappatura diretta.  
La corrispondenza tra Memoria Principale e banco è associativa.

Per cercare un dato non devo più analizzare tutte le linee di una cache, ma un'unica linea per ogni banco.

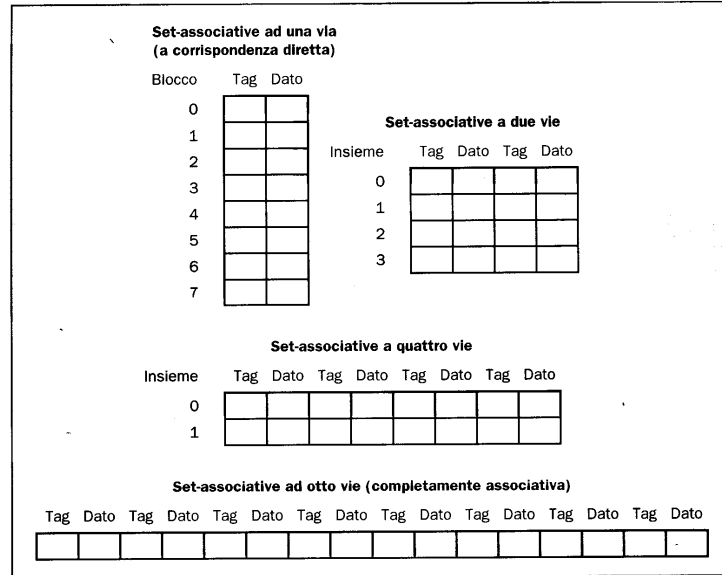
A.A. 2015-2016

4/19

<http://horghese.di.unimi.it/>



## Dalle cache a mappatura diretta alle cache associative



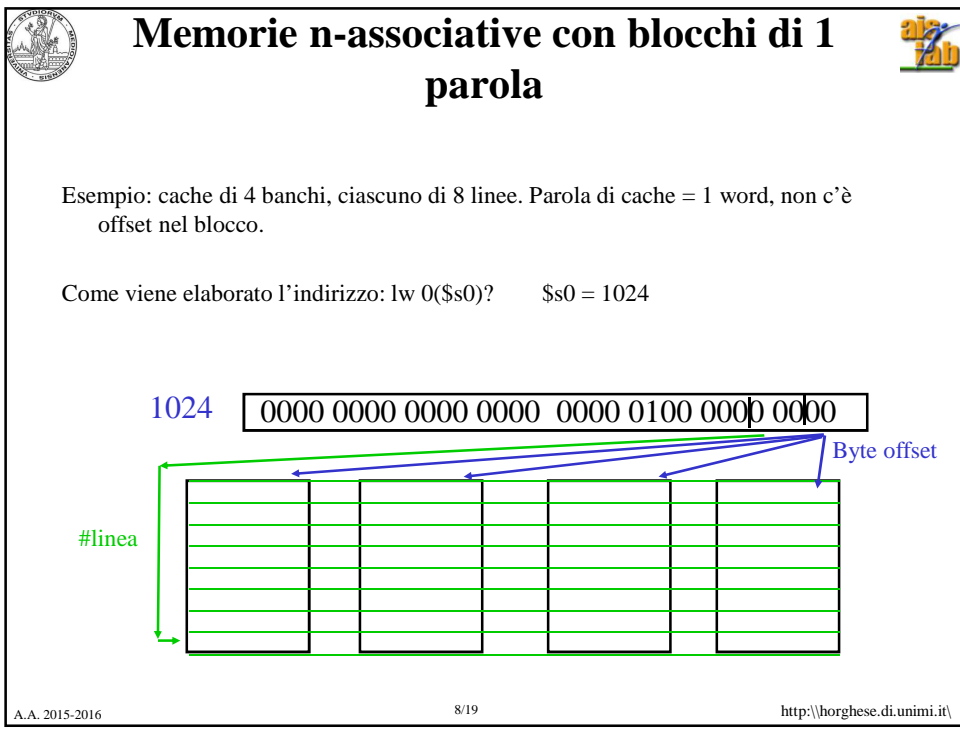
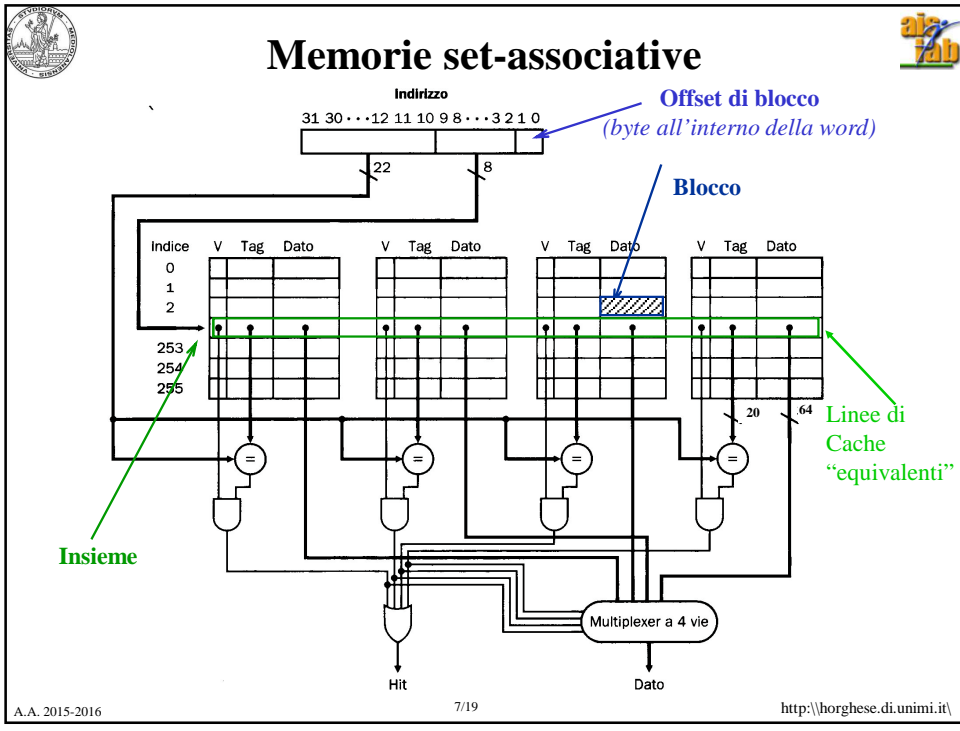
## Accesso a cache ad n-vie



**INDICE.** Se la parola richiesta è memorizzata in cache, si trova in una particolare linea di uno dei banchi. Questa linea è individuata dall'indice. L'indice è costituito da  $k$  bit, dove  $k = \log_2(\#linee)$ . E' analogo al numero di linea nelle cache a mappatura diretta.

**TAG** – contiene il blocco della RAM a cui appartiene il dato. Cerca il tag di Memoria Principale all'interno dei TAG associati alla linea individuata in ciascun banco.

L'insieme dei segnali di HIT pilotano anche il MUX che trasferiscono in uscita il contenuto del banco opportuno della cache.



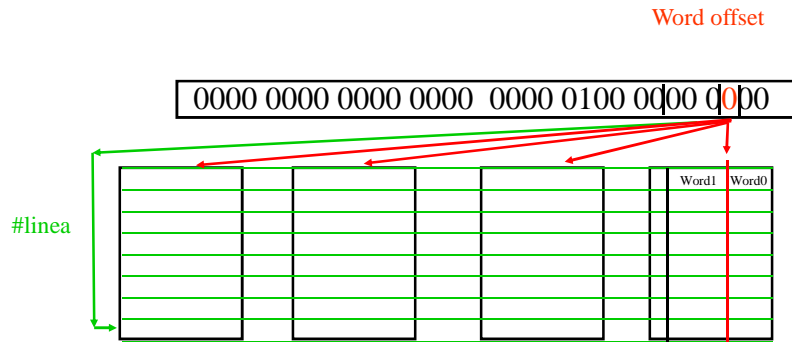


## Memorie n-associative con blocchi di 2 parole

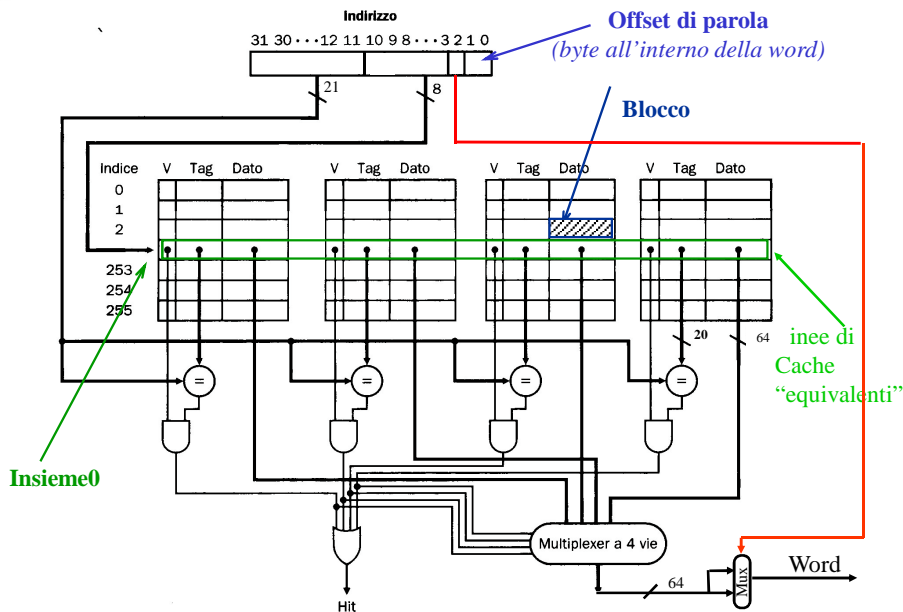


Esempio: cache di 4 banche, ciascuno di 8 linee. Parola di cache = 2 word.

Come viene elaborato l'indirizzo: lw 0(\$s0)? \$s0 = 1024



## Memorie set-associative





## Criteri di sostituzione di un blocco



### Dove inserisco il blocco letto dalla RAM?

Soluzione hardware, algoritmo semplice.

LRU – Least recently Used. Viene associato ad ogni blocco un bit di USE.  
Efficiente per memorie a 2 vie.

FIFO – Implementazione tramite buffer circolare.

LFU – Least frequently Used. Associa un contatore ad ogni blocco di cache.

RANDOM – Non funziona molto peggio!!



## Dove si può posizionare un blocco di RAM in cache?



Corrispondenza diretta: in un'unica posizione.

Memoria ad 1 via.  
 $\#posizioni = \#linee$ .

Completamente associative: in n posizioni (n banchi).

Ciascun banco è costituito da 1 linea.  
n insiemi o banchi.

N-associative: in m posizioni (m grado di associatività).

Ho m insiemi (banchi)  
Ciascun insieme è costituito da n linee.



## 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 insiemi,  
m comparazioni.



## Sommario



Memorie n-associative

**Esempi**



## Esempio di operazioni su cache



All'inizio il bit di data\_valid = False. Suppongo cache di 8 linee di 1 byte.  
Non importa quello che è contenuto. Cambio il contenuto di \$t1.

			TAG	#L	Valid
• lb \$t0, 0(\$t1)	\$t1 = 22 (10 110)	MISS	10	6	0->1
• lb \$t0, 0(\$t1)	\$t1 = 26 (11 010)	MISS	10	4	0->1
• lb \$t0, 0(\$t1)	\$t1 = 22 (10 110)	HIT	10	6	1->1
• lb \$t0, 0(\$t1)	\$t1 = 26 (11 010)	MISS	11	4	1->1
• lb \$t0, 0(\$t1)	\$t1 = 16 (10 000)	MISS	10	0	0->1
• lb \$t0, 0(\$t1)	\$t1 = 10 (00 010)	MISS	00	2	0->1
• lb \$t0, 0(\$t1)	\$t1 = 7 (00 111)	MISS	00	7	0->1
• lb \$t0, 0(\$t1)	\$t1 = 16 (10 000)	HIT	10	0	1->1
• lb \$t0, 0(\$t1)	\$t1 = 18 (10 010)	MISS	10	2	1->1



## Esempio di operazioni su cache



All'inizio il bit di data\_valid = False. Suppongo cache di 8 linee di 2 byte.  
Non importa quello che è contenuto. Cambio il contenuto di \$t1.

			TAG	#L	#W
• lb \$t0, 0(\$t1)	\$t1 = 22 (1 011 0)	MISS	1	3	0
• lb \$t0, 0(\$t1)	\$t1 = 26 (1 101 0)	MISS	1	5	0
• lb \$t0, 0(\$t1)	\$t1 = 22 (1 011 0)	HIT	1	3	0
• lb \$t0, 0(\$t1)	\$t1 = 27 (1 101 0)	HIT	1	5	1
• lb \$t0, 0(\$t1)	\$t1 = 16 (1 000 0)	MISS	1	0	0
• lb \$t0, 0(\$t1)	\$t1 = 2 (0 001 0)	MISS	0	1	0
• lb \$t0, 0(\$t1)	\$t1 = 7 (0 011 1)	MISS	0	3	1
• sb \$t0, 0(\$t1)	\$t1 = 8 (0 100 0)	n/a	0	4	0
• lb \$t0, 0(\$t1)	\$t1 = 16 (1 000 0)	HIT	1	0	0
• lb \$t0, 0(\$t1)	\$t1 = 18 (1 001 0)	MISS	1	1	0





## Esercizi



Sia data una cache a corrispondenza diretta contenente 64Kbyte di dati e avente blocchi di 1 parola. Assumendo che gli indirizzi siano di 32 bit quale è il numero totale di bit richiesto per l'implementazione della cache?

Sia data una cache a 4-vie contenente 64Kbyte di dati complessivamente (16Kbyte per via) e avente blocchi di 1 parola. Assumendo che gli indirizzi siano di 32 bit quale è il numero totale di bit richiesto per l'implementazione della cache?

Quale richiede più bit? Perché?



## Esercizi



Supponendo che il MIPS abbia una cache di 512 Byte, indicare cosa succede nei campi della cache quando vengono eseguite le seguenti istruzioni:

`lw $t1, 0x0000($t0) $t0 = 1kbyte = 1,024 byte`

`lw $t1, 0x0000($t0) $t0 = 0`

`lw $t1, 0x0202($t0) $t0 = 1kbyte = 1,024 byte`

`lw $t1, 0x0001($t0) $t0 = 0`

`lw $t1, 0x0201($t0) $t0 = 1kbyte = 1,024 byte`

Costruire la porta di scrittura della Cache di primo livello.



## Sommario



Memorie n-associative

Esempi