

## Esercizi Cache

**Es1:** Si consideri una cache di 4KB con associazione a gruppi a 8 vie ( 8-way set associative) in congiunzione con una memoria centrale di 1MB.

Supponendo che un blocco sia di dimensione 64B, si dica come un indirizzo di memoria è suddiviso in campi e a quanto ammonta la dimensione di ogni campo.

## Esercizi Cache

### Soluz.:

- trattandosi di una cache con associazione a gruppi, l'indirizzo di memoria centrale deve essere suddiviso nei campi tag, set, e parola.
- la memoria centrale è di 1MB, cioè  $2^{20}$  byte; pertanto un indirizzo di memoria centrale è espresso in 20 bit.
- la dimensione del campo parola è individuato univocamente dalla dimensione del blocco, che è di 64B, cioè  $2^6$  byte; pertanto il campo parola è di 6 bit.
- una cache di 4KB possiede  $2^{12}$  byte; ogni linea deve contenere un blocco e quindi impegna  $2^6$  byte; quindi la cache contiene  $2^{12} / 2^6 = 2^6$  linee. Poiché un insieme deve contenere 8 linee, il numero di insiemi della cache è pari a  $2^6 / 2^3 = 2^3$ . Pertanto il campo set è di 3 bit.
- la dimensione del campo tag sarà dunque:  $20 - 3 - 6 = 11$  bit

## Esercizi Cache

**Es2:** Si consideri una cache di 16KB con associazione a gruppi a 4 vie ( 4-way set associative) e dimensione di linea di 32B.

Supponendo che il campo tag sia di 12 bit, si dica quale è la dimensione massima (in byte) di memoria principale che la cache è in grado di gestire, assumendo il singolo byte come unità di indirizzamento della memoria.

## Esercizi Cache

**Soluz.:** per calcolare la quantità massima di memoria principale gestibile, bisogna calcolare il numero di bit totali che esprimono una generica locazione di memoria.

- trattandosi di una cache con associazione a gruppi, l'indirizzo di memoria centrale deve essere suddiviso nei campi tag, set, e parola.
- sappiamo che il campo tag è di 12 bit; quindi occorre calcolare la dimensione dei campi set e parola.
- la dimensione del campo parola è individuato univocamente dalla dimensione del blocco, che è di 32B, cioè  $2^5$  byte; pertanto il campo parola è di 5 bit.
- una cache di 16KB possiede  $2^{14}$  byte; ogni linea deve contenere un blocco e quindi impegna  $2^5$  byte; quindi la cache contiene  $2^{14}/2^5 = 2^9$  linee. Poiché un insieme deve contenere 4 linee, il numero di insiemi della cache è pari a  $2^9/2^2 = 2^7$ . Pertanto il campo set è di 7 bit.
- quindi la dimensione massima di memoria gestibile è:  $2^{12+7+5}$ , cioè 16MB

## Esercizi Cache

**Es3:** Si consideri una cache di 4KB con associazione a gruppi a 4 vie ( 4-way set associative) in congiunzione con una memoria centrale di 256KB.

Supponendo che un blocco sia di dimensione 64B, si dica:

- a) se le locazioni di memoria con indirizzi (in esadecimale) **30E5C** e **17A87** hanno la possibilità di essere caricate all'interno dello stesso set di linee;
- b) se in cache è presente la locazione con indirizzo **05ABC**, quali altre locazioni sono sicuramente presenti nella cache.

## Esercizi Cache

### Soluz.:

a) procedendo come visto negli esercizi precedenti, abbiamo che un indirizzo di memoria è scomponibile in un campo parola di 6 bit, un campo set di 4 bit, ed un campo tag di 8 bit.

• le due locazioni di indirizzo **30E5C** e **17A87** possono trovarsi nello stesso insieme se il loro campo set è identico. Quindi basta controllare se i bit da 10 a 7 (a partire da destra) sono identici:

$$30E5C = (\text{su 18 bit}) 110000111001011100$$

$$17A87 = (\text{su 18 bit}) 010111101010000111$$

• non essendo identici, la risposta è no.

b) le altre locazioni che necessariamente saranno presenti con la locazione di indirizzo **05ABC** sono quelle all'interno del medesimo blocco.

• poiché **05ABC** = (su 18 bit) 000101101010111100, tutte le locazioni con indirizzo da 000101101010000000 (hex **05A80**) a 000101101010111111 (hex **05ABF**) si troveranno simultaneamente in cache.

**Es4:** Sia data la seguente sequenza di istruzioni assembler, dove i dati immediati sono espressi in esadecimale ed il registro R0 contiene il valore 0:

```

LB  R3, 100(R0)  ! load byte da mem[100+[R0]]
ADD R2, R0, R0  ! R2 = R0 + R0
LB  R1, 108(R2)  ! load byte da mem[108+[R2]]
ADDI R2, R2, 5   ! R2 = R2 + 5
SUB  R4, R3, R2  ! R4 = R3 - R2
ADDI R1, R1, 3   ! R1 = R1 + 3
SB  R1, 108(R2)  ! store byte in mem[108+[R2]]
BGTZ R4, -6      ! PC = PC - 6 se [R4] > 0
                        ! cioè salta alla istruzione LB R1, 108(R2)

```

Si assuma la presenza di due cache, una dati ed una istruzioni.  
La cache dati, in particolare, è di ampiezza 8B, con dimensione di blocco 4B, inizialmente vuota, ed associazione diretta (con politica di scrittura write-through).

Si assuma che la memoria abbia il contenuto esadecimale mostrato di seguito (si esprimano gli indirizzi su 12 bit):

Indirizzo	byte	byte	byte	byte
100	0F	00	07	02
104	00	00	00	00
108	AE	13	A1	23
10C	A1	42	90	75
110	B9	16	00	00
114	0A	07	03	71

Si mostri come sia il contenuto della cache dati che il contenuto della memoria cambia a causa della esecuzione del codice assembler.

Soluzione: vedi pagina web del corso