

Esercitazione sulle memorie cache

1. Si progetti una cache di 16K a 8 vie per un sistema con indirizzamento al byte di 32bit, bus dati a 32 bit, bus indirizzi a 30bit, e word di 4 byte. In quale cella e con quale tag viene memorizzato il dato all'indirizzo 0x40404040 in memoria? Quanta memoria è necessaria per implementare la cache?

Una cache a 8 vie è una cache in grado di memorizzare 8 blocchi diversi per ogni indice. 16Kbyte corrispondono a $16/4 \text{ word} = 4\text{K word}$ quindi il numero di *bucket*, di indici diversi, è pari a $4\text{K}/8 = 512 \text{ bucket}$, cioè 9 bit di indice ($2^9 = 512$). Si noti che il trasferimento tra cache e CPU è di 1 word che è anche l'ampiezza dei registri della CPU. Il tag risulta quindi pari a $30-2-9 = 19 \text{ bit}$ (- 2 bit per la lunghezza della word - 9 bit per l'indirizzamento del bucket). L'indirizzo a 30 bit viene smontato in:

	unused	tag	indice	unused
bit	31-30	29-11	10-2	1-0

L'indirizzo **0x40404040 = 01.00 0000 0100 0000 0100 0.000 0100 00.00₂**
 corrisponde all'**indice 000 0100 00 = 16** ed al **tag = 000 0000 1000 0000 1000 = 2056**.

	2 bit	tag (19 bit)	indice (9 bit)	2bit
0x40404040	01	000 0000 1000 0000 1000	000 0100 00	00

Lo spazio fisicamente necessario per implementare la cache è pari a (non consideriamo i bit di *age* necessari per selezionare la cella da rimuovere in caso di conflitto):

$$19 \text{ bit di tag} + 32 \text{ bit di dato} + 1 \text{ bit di validate} = 52 \text{ bit (+age)}$$

moltiplicati per il numero di bucket:

$$52\text{bit} * 512 \text{ bucket} = 26624 \text{ bit} = 26\text{Kbit (+age)}$$

2. Si progetti una cache di 128 byte a corrispondenza diretta organizzata in blocchi di 4 word con word di 2 byte per un sistema con indirizzamento a byte a 16 bit, bus dati a 64 bit, bus indirizzi a 16 bit. Si consideri il tempo di accesso della cache 2ns ed il tempo di accesso alla DRAM di 60ns. Data la sequenza di accessi 0,2,8,10,5,32 dire quanto vale il tempo di accesso totale nell'ipotesi che i bit di validate siano tutti a 0.

128 byte corrispondono a $128/2 = 64$ word. Ogni blocco richiede 4 word. Per selezionare una delle 4 word occorrono 2 bit di offset. Si possono memorizzare $64/4 = 16$ blocchi indirizzabili con 4 bit ($2^4 = 16$). Ne segue che il tag richiede:

16 bit di indirizzo – 1 bit di allineamento – 4 bit dell'indice – 2 bit di offset = 9 bit

Segue che l'indirizzo viene così smontato:

	tag (9 bit)	indice (4 bit)	offset (2 bit)	unused
bit	15-7	6-3	2-1	0

Lo spazio occorrente per implementare questa cache è quindi:

$(9 \text{ bit di tag} + 1 \text{ bit di validate} + 64 \text{ bit di blocco}) * 16 \text{ blocchi} = 1184 \text{ bit}$

L'accesso a **0 (00000000-0000-00-0)** genera un **miss** perchè la cache è totalmente vuota. il blocco corrispondente con **tag=0** e **indice=0** viene caricato dalla memoria in un solo accesso (64 bit di bus dati).

L'accesso a **2 (00000000-0000-01-0)** corrisponde allo stesso blocco, **indice=0** quindi abbiamo un **hit**.

L'accesso a **8 (00000000-0001-00-0)** corrisponde al **tag=0**, **indice=1** che non è ancora stato letto quindi è un **miss**.

L'accesso a **10 (00000000-0001-01-0)** corrisponde a **tag=0**, **indice=1**; in questo caso è un **hit**.

L'accesso a **5 (00000000-0000-10-1)** corrisponde a **tag=0**, **indice=0**; in questo caso è un **hit** (da notare che questo indirizzamento non è allineato alla word).

L'accesso a **32 (00000000-0100-00-0)** corrisponde a **tag=0**, **indice=4**, non ancora presente in cache quindi **miss**.

In totale abbiamo tre **hit** e tre **miss**. Ne segue che il tempo totale di accesso è:

$$3 * 2ns + 3 * (2ns + 60ns) = 192ns \quad (\text{in media } 192ns/6 = 32ns)$$

Se il bus dati fosse stato a 16 bit sarebbero occorsi 4 accessi per ogni miss quindi:

$$3 * 2ns + 3 * (2ns + 4 * 60ns) = 252ns \quad (\text{in media } 252ns/6 = 42ns)$$

3. 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?

Supponiamo che le parole siano di 4 bytes come per il MIPS e che l'indirizzamento sia al byte. Segue che il numero di parole della cache è:

$$64\text{kbytes}/4\text{bytes} = 16\text{K words}$$

Ogni blocco e' costituito da una singola parola quindi il numero di blocchi è:

$$16\text{Kword}/1\text{word} = 16\text{K blocchi}$$

per indirizzare 16K blocchi occorrono:

$$\log_2(16\text{K}) = 14 \text{ bit}$$

per indirizzare le parole all'interno del blocco non occorrono bit (una sola parola).

Ne segue che il tag avrà lunghezza:

$$32-14-2=16 \text{ bit}$$

L'indirizzo sarà così scomposto:

	tag (16 bit)	indice (14 bit)	offset (0 bit)	unused (2 bit)
bit	31-15	15-2	nessuno	1-0

Lo spazio occorrente per un blocco è:

$$32 \text{ bit di dato} + 16 \text{ bit di tag} + 1 \text{ bit validate} = 49 \text{ bit}$$

Lo spazio totale quindi necessario è:

$$49 \text{ bit} * 16\text{K} = 802816 \text{ bits} = 98\text{Kbytes}$$

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

<i>lb \$t1, 0x0000(\$t0)</i>	<i>\$t0 = 1kbyte = 1,024 byte</i>
<i>lb \$t1, 0x0000(\$t0)</i>	<i>\$t0 = 0</i>
<i>lb \$t1, 0x0202(\$t0)</i>	<i>\$t0 = 1kbyte = 1,024 byte</i>
<i>lb \$t1, 0x0001(\$t0)</i>	<i>\$t0 = 0</i>
<i>lb \$t1, 0x0201(\$t0)</i>	<i>\$t0 = 1kbyte = 1,024 byte</i>

Supponiamo di utilizzare una cache ad indirizzamento diretto organizzata in word di 4 bytes. Segue che il numero di parole della cache è:

$$512\text{bytes}/4\text{bytes} = 128 \text{ words}$$

Ogni blocco e' costituito da una singola parola quindi il numero di blocchi è:

$$128\text{word}/1\text{word} = 128 \text{ blocchi}$$

per indirizzare 128 blocchi occorrono:

$$\log_2(128) = 7 \text{ bit}$$

per indirizzare le parole all'interno del blocco non occorrono bit (una sola parola).

Ne segue che il tag avrà lunghezza:

$$32-7-2=23 \text{ bit}$$

L'indirizzo sarà così scomposto:

	tag (23 bit)	indice (7 bit)	offset (0 bit)	unused (2 bit)
bit	31-9	8-2	nessuno	1-0

Lo spazio occorrente per un blocco è:

$$32 \text{ bit di dato} + 23 \text{ bit di tag} + 1 \text{ bit validate} = 56 \text{ bit}$$

Lo spazio totale quindi necessario è:

$$56 \text{ bit} * 128 = 7168 \text{ bits} = 896 \text{ bytes}$$

Gli indirizzi richiesti da ogni istruzione sono i seguenti:

<i>lb \$t1, 0x0000(\$t0)</i>	<i>\$t0 = 1024</i>	<i>0x0000(\$t0) = 0x0400</i>
<i>lb \$t1, 0x0000(\$t0)</i>	<i>\$t0 = 0</i>	<i>0x0000(\$t0) = 0x0000</i>
<i>lb \$t1, 0x0202(\$t0)</i>	<i>\$t0 = 1024</i>	<i>0x0202(\$t0) = 0x0602</i>
<i>lb \$t1, 0x0001(\$t0)</i>	<i>\$t0 = 0</i>	<i>0x0001(\$t0) = 0x0001</i>
<i>lb \$t1, 0x0201(\$t0)</i>	<i>\$t0 = 1024</i>	<i>0x0201(\$t0) = 0x0601</i>

Il tag si può ottenere anche facendo la divisione intera per $2^{(7+2)}=512$, equivalente allo shift di 9 posizioni a destra.

L'indice si può ottenere anche facendo il modulo di $2^{(7+2)}=512$ e quindi la divisione intera per $2^2=4$, equivalente a prendere 9 bit e su questi scartare i primi 2.

<i>0x0400 = <u>0000.0000</u> 0000.0000 0000.0100 0000.0000₂</i>	<i>tag=2</i>	<i>indice=0</i>
<i>0x0000 = <u>0000.0000</u> 0000.0000 0000.0000 0000.0000₂</i>	<i>tag=0</i>	<i>indice=0</i>
<i>0x0602 = <u>0000.0000</u> 0000.0000 0000.0110 0000.0010₂</i>	<i>tag=3</i>	<i>indice=0</i>
<i>0x0001 = <u>0000.0000</u> 0000.0000 0000.0000 0000.0001₂</i>	<i>tag=0</i>	<i>indice=0</i>
<i>0x0601 = <u>0000.0000</u> 0000.0000 0000.0110 0000.0000₂</i>	<i>tag=3</i>	<i>indice=0</i>

Supponiamo come condizione iniziale la cache inizializzata a 0:

indice	tag	validate	dato
0	0	0	0
...
64	0	0	0
...
127	0	0	0

Al primo accesso, 0x0400, viene richiesta la cella di indice 0. Il bit di validate è a 0 quindi ho un **miss**. La cache diventerà dopo l'aggiornamento:

indice	tag	validate	dato
0	2	1	0x1111
...
64	0	0	0
...
127	0	0	0

Al secondo accesso, 0x0000, viene richiesta la cella di indice 0. Il bit di validate è a 1 ma il tag, 2, è diverso da quello richiesto, 0, quindi ho un **miss**. La cache diventerà dopo l'aggiornamento:

indice	tag	validate	dato
0	0	1	0x2222
...
64	0	0	0
...
127	0	0	0

Al terzo accesso, *0x0602*, viene richiesta la cella di indice 0. Il bit di validate è a 1 ma il tag differisce quindi ho un **miss**. La cache diventerà dopo l'aggiornamento:

indice	tag	validate	dato
0	3	1	0x3333
...
64	0	0	0
...
127	0	0	0

Al quarto accesso, *0x0001*, viene richiesta la cella di indice 0. Il bit di validate è a 1 e il tag differisce quindi ho un **miss**. La cache diventerà dopo l'aggiornamento:

indice	tag	validate	dato
0	0	1	0x4444
...
64	0	0	0
...
127	0	0	0

Al quinto accesso, *0x0601*, viene richiesta la cella di indice 64. Il bit di validate è a 1 ed il tag differisce quindi ho un **miss**. La cache diventerà dopo l'aggiornamento:

indice	tag	validate	dato
0	3	1	0x5555
...
64	0	0	0
...
127	0	0	0

5. Si progetti una cache a corrispondenza diretta di 64Kbytes che usa word di 2 bytes e memorizza 4 word per ogni indice, bus dati a 16 bit, bus indirizzi a 29 bit., indirizzamento al byte. Dimensionare tutti i parametri della cache. Quanti accessi richiede un **miss**?

Poiché usa word di 2 byte segue che la cache conterrà:

$$\mathbf{64Kbytes/2bytes = 32K words}$$

Un bit dell'indirizzo viene ignorato poiché si usa l'indirizzamento al byte.

Poiché memorizza 4 word per ogni indice, il numero di linee della cache sarà:

$$\mathbf{32Kwords / 4words = 8K linee}$$

Per indicizzare 8K linee servono:

$$\mathbf{\log_2(8K) = 13bit}$$

Per indicizzare le 4 word nel blocco servono:

$$\mathbf{\log_2(4) = 2bit}$$

Poiché si usa un bus indirizzi a 29 bit, ne segue che il tag avrà lunghezza:

$$\mathbf{29-13-2-1=13 bit}$$

L'indirizzo sarà così scomposto:

	tag (13 bit)	indice (13 bit)	offset (2 bit)	unused (1 bit)
bit	28-16	15-3	2-1	0

Lo spazio occorrente per un blocco è:

$$\mathbf{(4word*2bytes=64 bit) di dato + 13 bit di tag + 1 bit validate = 78 bit}$$

Lo spazio totale quindi necessario è:

$$\mathbf{78 bit * 8K = 638976 bits = 78 Kbytes}$$

Poiché il bus dati è a 16 bit si può trasferire solo una word alla volta quindi per eseguire un **miss** occorrono 4 accessi alla memoria.

6. *Si progetti una cache di 1Kbyte, 2-associativa, contenente 16 parole per linee con parole di 8 byte. Dimensionare correttamente la cache e definire cosa succede per la seguente sequenza di istruzioni:*

lw \$t1, 32(\$0)

lw \$t2, 1024(\$0)

lw \$t3, 0(\$0)

lw \$t4, 512(\$0)

lw \$t5, 1032(\$0)

giustificare le risposte.