Computer Architecture

Fall, 2019

Week 15

2019.12.16

組別：_____　簽名：_____

[group2] (對抗賽)

1. Which of the following statements are generally true?

(1) Caches take advantage of temporal locality.

~~(2) On a read, the value returned depends on which blocks are in the cache.~~

~~(3) Most of the cost of the memory hierarchy is at the highest level.~~

(4) Most of the capacity of the memory hierarchy is at the lowest level.

A1:

1: True

2: ~~False, The value returned by a read remains the same.~~

~~3: False, Most of the cost of the memory hierarchy is at the lowest level.~~

4: True

[group4]

2. Describe the advantage and disadvantage of increasing the block sizes.

Ans:

advantage : reduce miss rate due to spatial locality

disadvantage : if block sizes too big ,then the number of block will decrease

⇒ miss rate increase

[group3] (對抗賽)

3. The following bits of the address are used to access a direct-mapped cache designed with a 16-bit address:

Tag	Index	Offset
15-10	9-5	4-0

And there is a list of memory address references: (start from power on)

0, 4, 16, 132, 232, 106, 1024, 30, 140, 3100, 180, 2180

How many blocks are replaced? What's the hit ratio?

A:

Demical Address	Binary Address	Tag	Index	Hit/Miss	Replacement
0	0000 0000 0000 0000	0	0	Miss	No
4	0000 0000 0000 0100	0	0	Hit	No
16	0000 0000 0001 0000	0	0	Hit	No
132	0000 0000 1000 0100	0	4	Miss	No
232	0000 0000 1110 1000	0	7	Miss	No
106	0000 0000 1010 0000	0	5	Miss	No
1024	0000 0100 0000 0000	1	0	Miss	Yes
30	0000 0000 0001 1110	0	0	Miss	Yes
140	0000 0000 1000 1100	0	4	Hit	No
3100	0000 1100 0001 1100	3	0	Miss	Yes
180	0000 0000 1011 0100	0	5	Hit	No
2180	0000 1000 1000 0100	2	4	Miss	Yes

(1)4

(2)4/12*100% = 33.34%

[group4] (對抗賽)

4. For a Address sub-division design with 32-bit, the following bits of the address are used to access the cache a, b and c

cache	Tag	Index	Offset
a	31-10	9-4	3-0
b	31-12	11-5	4-0
c	31-13	12-4	3-0

According to the table above , please answer the following answer.

(1)Find the block size of a ,b and c.

(2)How many entries does cache a, b and c have?

(3)Give a byte address 2400, find the block number of cache a, b and c.

Ans:

(1) a : 16, b : 32, c : 16

(2) a : 2^(9-4+1) = 2^6 = 64
b : 2^(11-5+1) = 2^7 = 128
c : 2^(12-4+1) = 2^9 = 512

(3) 2400_{10} = 0000 1001 0110 0000_2
a : 0000 1001 0110 0000_2 ⇒ 2^4+2^2+2^1 = 21
b : 0000 1001 0110 0000_2 ⇒ 2^6+2^3+2^1+2^0 = 75
c : 0000 1001 0110 0000_2 ⇒ 2^7+2^4+2^2+2^1 = 149

[group12] (對抗賽)

5. 關於 write-through 和 write-back 的敘述，下列何者正確？

A. 當 write hit 時，write-through 會將 cache 和 memory 的資料同時更新
B. 當 write-back 的 dirty bit 為 1 時，代表 cache 裡沒有資料
C. 在 cache 和 memory，Write-back 的資料是非一致性的
D. 當 write miss 時，對於 write-through 通常都是 fetch the block

Ans：

a)先存到 write buffer 再慢慢存回 memory
b) cache 有被寫過資料
d) write-back

[group11] (對抗賽)

6. Which of the following statement is true?

(A) Write-through : The information is written to both the block in the cache and the block in the lower level of the memory hierarchy (main memory for a cache).
(B) Write-back : The information is written only to the lower level of the memory hierarchy. The modified block is written to the lower level of the hierarchy only when it is replaced.
(C)In a fixed-sized cache, larger blocks will always reduce miss rate.
(D) Static Random Access Memory need to be refreshed regularly.

ANS:

(A)True.
(B) Write-back : The information is written only to the block in the cache. The modified block is written to the lower level of the hierarchy only when it is replaced.
(C) In a fixed-sized cache, larger blocks will increased miss rate.
(D) Dynamic Random Access Memory need to be refreshed regularly

[group9] (對抗賽)

7. Please give out 2 reasons why we use memory hierarchy approach.

Ans.

DRAM's speed can't follow up CPU's speed, in fact processor grows 50% faster per yaer, and the cost of SRAM is too high.
Since 10% of code executed 90% of time, most of the time we the memory data we access is not as wide as the whole memory, so we can transfer those data to SRAM to increase performance.

[Final Project]

Consider a cache memory system with 4 cache sets, 4-byte block size, and 8 address bits, namely $a_7, a_6, a_5, a_4, a_3, a_2, a_1$, and a_0, shown as follows. (Assume the system is byte addressable.)

```
Address_bits: 8
Sets: 4
Associativity: 1 (direct-map)
Block_size: 4
```

Also, the cache accessing sequence is given in ordering ($a_7a_6a_5a_4a_3a_2a_1a_0$).

```
00000000 (a7a6a5a4a3a2a1a0)
00010000
00100000
00000000
00101100
00000000
00101100
```

What is the number of indexing bits? What is the number of offset bits? What is the indexing bits? (Use the least significant bits) If all cache entries are invalid at the beginning, what is cache miss rate? Please answer the questions in the following form.

```
Indexing bit count:
Indexing bits: a7a6a5a4a3a2a1a0 (choose indexing bits)
Offset bit count:

00000000 hit/miss (choose one)
00010000 hit/miss (choose one)
00100000 hit/miss (choose one)
00000000 hit/miss (choose one)
00101100 hit/miss (choose one)
00000000 hit/miss (choose one)
00101100 hit/miss (choose one)

Cache miss rate:
```

Answer:

Indexing bit count: 2

Indexing bits: $a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0$ (choose indexing bits)

Offset bit count: 2

00000000 hit/miss (choose one)

00010000 hit/miss (choose one)

00100000 hit/miss (choose one)

00000000 hit/miss (choose one)

00101100 hit/miss (choose one)

00000000 hit/miss (choose one)

00101100 hit/miss (choose one)

Cache miss rate: 5/7