5.3, 5.6 郑鑫颖 11912039

5.3 For a direct-mapped cache design with a 32-bit address, the following bits of the address are used to access the cache.

Tag	Index	Offset
31–10	9–5	4–0

allocate

5.3.1 [5] <§5.3> What is the cache block size (in words)?

8 words in the block

5.3.2 [5] <§5.3> How many entries does the cache have?

2^5 entries in the cache

5.3.3 [5] <§5.3> What is the ratio between total bits required for such a cache implementation over the data storage bits?

total: $(2^3 \times 32 + 22 + 1) = 279$

$\frac{\text{total}}{\text{Data}} = \frac{279}{256} \approx 1.09$

Data: $2^3 \times 32 = 256$

Starting from power on, the following byte-addressed cache references are recorded.

Address											
0	4	16	132	232	160	1024	30	140	3100	180	2180

5.3.4 [10] <§5.3> How many blocks are replaced?

0 0
1 0 0 0
1 0 0 0 0 0
1 0 0 0 0 1 0 0 4
1 1 1 0 1 0 0 0 7
1 0 1 0 0 0 0 0 5
0 1 0 0 0 0 0 0 0 0 0 0 0 ←
0 0 0 1 1 1 1 0 0 ←
1 0 0 0 1 1 0 0 4
1 1 0 0 0 0 0 1 1 1 0 0 0 ←
1 0 1 1 0 1 0 0 5
1 0 0 0 1 0 0 0 0 1 0 0 4 ←

So totally 4 blocks are replaced.

5.3.5 [10] <§5.3> What is the hit ratio?

$\frac{4}{12} = \frac{1}{3} \approx 33\%$

5.3.6 [20] <§5.3> List the final state of the cache, with each valid entry represented as a record of <index, tag, data>.

<Index, tag, data>

<00000, 0000…11, Mem[3072-3103]> (22 bits)

<00100, 0000…10, Mem[2176-2207]>

<00101, 0000...00, Mem[160-191]

<00111, 000...00, Mem[224-255]

5.6 In this exercise, we will look at the different ways capacity affects overall performance. In general, cache access time is proportional to capacity. Assume that main memory accesses take 70 ns and that memory accesses are 36% of all instructions. The following table shows data for L1 caches attached to each of two processors, P1 and P2.

70ns

	L1 Size	L1 Miss Rate	L1 Hit Time
P1	2 KiB	8.0%	0.66 ns
P2	4 KiB	6.0%	0.90 ns

70/0.66 = 107

70/0.90 = 78

5.6.1 [5] <§5.4> Assuming that the L1 hit time determines the cycle times for P1 and P2, what are their respective clock rates?

clock rate = $\frac{1}{\text{cycle time}}$ $f_1 = \frac{1}{0.66\text{ns}} \approx 1.52\text{GHz}$

$f_2 = \frac{1}{0.90\text{ns}} \approx 1.11\text{GHz}$.

5.6.2 [5] <§5.4> What is the Average Memory Access Time for P1 and P2?

P1: 0.66 + 8% × 70 = 6.26ns

P2: 0.90 + 6% × 70 = 5.1ns.

5.6.3 [5] <§5.4> Assuming a base CPI of 1.0 without any memory stalls, what is the total CPI for P1 and P2? Which processor is faster?

P1: 1 + 8% × 107 + 36% × 8% × 107 = 12.6416 cycles/inst

P2: 1 + 6% × 78 + 36% × 6% × 78 = 7.3648 cycles/inst

P1 t = 12.6416 × 0.66 ≈ 8.34 ns

P2 t = 7.3648 × 0.90 ≈ 6.63 ns

So P2 is faster

For the next three problems, we will consider the addition of an L2 cache to P1 to presumably make up for its limited L1 cache capacity. Use the L1 cache capacities and hit times from the previous table when solving these problems. The L2 miss rate indicated is its local miss rate.

L2 Size	L2 Miss Rate	L2 Hit Time
1 MiB	95%	5.62 ns

9 cycles

5.6.4 [10] <§5.4> What is the AMAT for P1 with the addition of an L2 cache? Is the AMAT better or worse with the L2 cache?

L1

$0.66 + 8\% \times (5.62 + 95\% \times 70) = 6.43\text{ns}$

worse.

5.6.5 [5] <§5.4> Assuming a base CPI of 1.0 without any memory stalls, what is the total CPI for P1 with the addition of an L2 cache?

$CPI = 1 + 8\% \times (9 + 95\% \times 107) + 36\% \times 8\% \times (9 + 95\% \times 107)$

$= 13.04$ cycles/inst

5.6.6 [10] <§5.4> Which processor is faster, now that P1 has an L2 cache? If P1 is faster, what miss rate would P2 need in its L1 cache to match P1's performance? If P2 is faster, what miss rate would P1 need in its L1 cache to match P2's performance?

P2 is faster

$\underbrace{(1 + a\% \times (9 + 95\% \times 107) \times (1 + 36\%))}_{P1} \times 0.66 = \underbrace{7.3648 \times 0.90}_{P2}$.

$a\% \approx 6\%$

So the miss rate should be 6% to match the performance.