P483

5.1.1 4

5.1.2 I, J, B[J][0]

5.1.3 A[I][J]

5.1.4 $\frac{8\times 800}{4}\times 2-\frac{8\times 8}{4}+\frac{8}{4}=3186$

5.1.5 I, J, B(J,0)

5.1.6 A(I,J), B(J,0), A(J,I)

5.2.1 Binary address: 00000011, 10110100, 00101011, 00000010, 10111111, 01011000, 10111110, 00001110, 10110101, 00101100, 10111010, 11111101

Tag: 0000, 1011, 0010, 0000, 1011, 0101, 1011, 0001, 1011, 0010, 1011, 1111

Index: 0011, 0100, 1011, 0010, 1111, 1000, 1110, 1110, 0101, 1100, 1010, 1101

Hit/Miss: M, M, M, M, M, M, M, M, M, M, M, M

5.2.2 Binary address: 00000011, 10110100, 00101011, 00000010, 10111111, 01011000, 10111110, 00001110, 10110101, 00101100, 10111010, 1111101

Tag: 00000, 10110, 00101, 00000, 10111, 01011, 10111, 0001, 1010, 00101, 10111, 11111

Index: 001, 010, 101, 001, 111, 100, 111, 111, 010, 110, 101, 110

Hit/Miss : M, M, M, M, M, M, H, M, H, M, M, M

5.2.3 C_1 : $|BO| = \log_2 |\text{block size}| = \log_2 1 = 0$

$\text{blocks\_in\_cache} = \frac{\text{cache\_size}}{\text{block\_size}} = \frac{8}{1} = 8$

$|\text{Index}| = \log_2 8 = 3$

C_2 : $|BO| = \log_2 |\text{block size}| = \log_2 2 = 1$

$\text{blocks\_in\_cache} = \frac{\text{cache\_size}}{\text{block\_size}} = \frac{8}{2} = 4$

$|\text{Index}| = \log_2 4 = 2$

C_3 : $|BO| = \log_2 |\text{block-size}| = \log_2 4 = 2$

$\text{blocks\_in\_cache} = \frac{\text{cache\_size}}{\text{block\_size}} = \frac{8}{4} = 2$

$|\text{Index}| = \log_2 2 = 1$

word address	binary address	C_1	C_2	C_3
3	00000011	M	M	M
180	10110100	M	M	M
43	00101011	M	M	M
2	00000010	M	M	M
191	10111111	M	M	M
88	01011000	M	M	M
190	10111110	M	H	H
14	00001110	M	M	M
181	10110101	M	H	M
44	00101100	M	M	M
186	10111010	M	M	M
253	11111101	M	M	M

C_1 Miss_rate = $\frac{\text{Misses}}{\text{accesses}} = 1$

C_2 Miss_rate = $\frac{\text{Misses}}{\text{accesses}} = \frac{10}{12} = \frac{5}{6}$ $\therefore$ C_2 is the best

C_3 Miss_rate = $\frac{\text{Misses}}{\text{accesses}} = \frac{11}{12}$

$C_1' = \frac{20}{25}$ $\quad C_2' = \frac{30}{25}$ $\quad C_3' = \frac{55}{25}$

At this time, C_1 is the best

5.2.4 $32\text{ KiB} = 2^{13}$ words $\quad$ block size = 2 words

blocks = 2^{12}

$2^{12} \times (2\times32 + (32-10-2-2)+1) = 2^{12} \times 83 = 83$ kiB

Block size' = 16 words = 2^4 words

blocks = 2^9

$2^9 \times (16\times32 + (32-10-2-2)+1) = 265.5$ Kibibits $= 66.375$ kiB

Because the second one wastes space and too much time on accessing.

5.2.5 Have a lower or equal miss rate than a smaller 2-way set associative cache. At least double the block size.

Advantage: less misses for near by addresses.

Disadvantage: Longer access time

5.2.6 It's possible

The XOR operation lost the data so that more tag bits are needed to identify the address in the cache