

### Exercise 8.5

(a) block size

spatial locality를 이용한 miss rate가 줄어든다.  
또한 block size가 증가하면 초기에

발생하는 compulsory misses가 줄어든다.

하지만 block size가 커지면 가져오는

data도 많기 때문에 memory에서

cache로 가져오는 시간, 즉 miss

penalty가 증가한다. 그리고 block

size가 커지면 conflict misses가

증가한다.

(b) associativity

associativity가 증가하면

conflict misses가 줄어든다.

하지만 associativity가 증가할수록

cpu가 찾는 시간이 증가한다.

(c) cache size

cache size가 증가하면

capacity misses가 감소하고

conflict misses가 감소한다.

하지만 cpu가 접근하는 시간은

증가한다.

### Exercise 8.7

(a) False

Counterexample :

A 2-word cache with block size  
of 1 word and access pattern :

0 4 8 0 4 8

2-way set associative cache miss rate

is 100% , direct mapped

cache miss rate is

$$\frac{5}{6} \times 100 = 83.3\%$$

(b) True

16-kB direct mapped cache가

8-kB direct mapped cache 보다

cache size가 더 크기 때문에

miss rate가 같을 수는 있지만

보통 16-kB direct mapped cache

가 더 낮은 miss rate를 갖는다.

(c) Usually true

block size 가 증가하면  
spatial locality 를 이용하여 compulsory  
misses 가 줄어든다. 따라서 보통은  
block size 가 커지면 miss rate 가  
작아진다. 하지만 같은 associativity 이고  
같은 total capacity 이기 때문에  
conflict misses 는 증가한다.

### Exercise 8.8

$b$ : block size given in numbers of words

$S$ : number of sets

$N$ : number of ways

$A$ : number of address bits

(a) what is the cache capacity,  $C$ ?

$$C = b \times S \times N \times 4 \text{ bytes}$$

(b) what is the total number of  
bits required to store the tags?

$$\left[ A - (\log_2 S + \log_2 b + 2) \right] \times S \times N$$

(c) What are  $S$  and  $N$  for a  
fully associative cache of capacity  $C$   
words with block size  $b$ ?

$$S = 1, N = C/b$$

(d) What is  $S$  for a direct mapped  
cache of size  $C$  words and block  
size  $b$ ?

$$S = C/b$$

### Exercise 8.9

(a) direct mapped cache,  $b = 1$  word

set 0	40 → 80 → 0
set 1	44 → 84 → 4
set 2	48 → 88 → 8
	4C → 8C → C
	90 → 10
	94 → 14
	98 → 18
	9C → 1C
	20
	70
	74
	78
set 15	7C

$$\text{set} = \frac{16}{1} = 16 \text{ 개}$$

70, 74, 78, 7C, 20 이 hit 이고

나머지는 miss 이기 때문에

$$\text{miss rate} = \frac{20}{25} \times 100 = 80\%$$

이다.

(b) fully associative cache, b = 1 word

Set 0	40	44	48	4C	70	74	78	7C	80	84	88	8C	90	94	98	9C
	→ 0	→ 4	→ 8	→ C	→ 10	→ 14	→ 18	→ 1C	→ 20							

hit 가 하나도 없기 때문에 miss rate 는  $\frac{25}{25} \times 100 = 100\%$  이다.

(c) two-way set associative cache, b = 1 word

$$\text{set} = \frac{16}{2 \times 1} = 8 \text{ 개}$$

set 0	40 → 0	80 → 20
set 1	44 → 4	84
set 2	48 → 8	88
set 3	4C → C	8C
set 4	70 → 10	90
set 5	74 → 14	94
set 6	78 → 18	98
set 7	7C → 1C	9C

hit 가 하나도 없기 때문에 miss rate 는  $\frac{25}{25} \times 100 = 100\%$  이다.

(d) direct mapped cache, b = 2 words

$$\text{set} = \frac{16}{1 \times 2} = 8 \text{ 개}$$

set 0	(40, 44) → (80, 84) → (0, 4)
set 1	(48, 4C) → (88, 8C) → (8, C)
set 2	(90, 94) → (10, 14)
set 3	(98, 9C) → (18, 1C)
set 4	(20, 24)
set 5	
set 6	(70, 74)
set 7	(78, 7C)

40, 48, 80, 88, 90, 98, 0, 8,  
10, 18 이 miss 이기 때문에

miss rate 는  $\frac{10}{25} \times 100 = 40\%$  이다.

# Exercise 8.10

(c) 2-way set associative cache,

(a) direct mapped cache,  $b = 1$  word

$b = 2$  words

$$\text{set} = \frac{16}{1 \times 1} = 16 \text{ H}$$

$$\text{Set} = \frac{16}{2 \times 2} = 4 \text{ H}$$

set 0	
set 1	84
	88 → 388
	38C → 8C → 18C
	A0
	AC
	74 → 34
	78 → 38
set 15	7C → 13C

set 0	(A0, A4)	(80, 84)
set 1	(388, 38C) → (88, 8C) (188, 18C) → (A8, AC)	(A8, AC) → (388, 38C) → (88, 8C)
set 2	(70, 74)	(30, 34)
set 3	(78, 7C) → (138, 13C) → (38, 3C)	(38, 3C) → (78, 7C)

74, A0, 38C, 84, 8C, 7C, 34 만

hit 이기 때문에 hit rate =  $\frac{7}{14}$

miss rate 는  $\frac{7}{14} \times 100 = \boxed{50\%}$

이다.

84, A0, AC 를 제외하면 모두

miss 이기 때문에

miss rate 는  $\frac{11}{14} \times 100 = \boxed{79\%}$  이다.

(b) fully associative cache,  $b = 2$  words

set 0	(70, 74) → (38, 3C) → (80, 84)	(A0, A4) → (138, 13C) → (88, 8C)	(78, 7C)	(388, 38C)	(A8, AC) → (188, 18C) → (30, 34)	(80, 84) → (70, 74)	(88, 8C) → (A0, A4)	(30, 34) → (A8, AC)
-------	--------------------------------------	--	----------	------------	--	------------------------	------------------------	------------------------

78, 38C, 8C, 7C, 388 만 hit 이기 때문에 hit rate =  $\frac{5}{14}$ , miss rate 는

$$\frac{14-5}{14} \times 100 = \frac{9}{14} \times 100 = \boxed{64\%} \text{ 이다.}$$

(d) direct mapped cache,  $b = 4$  words

$$\text{set} = \frac{16}{1 \times 4} = 4 \text{개}$$

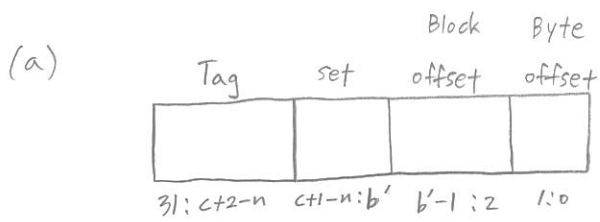
set0	$(380, 384, 388, 392) \rightarrow (80, 84, 88, 92) \rightarrow (380, 384, 388, 392) \rightarrow (180, 184, 188, 192)$
set1	
set2	$(A0, A4, A8, AC)$
set3	$(70, 74, 78, 7C) \rightarrow (30, 34, 38, 3C) \rightarrow (130, 134, 138, 13C)$

A0, 78, AC, 88, 8C, 7C, 38 이 hit 이기 때문에

$$\text{hit rate} = \frac{7}{14}, \text{ miss rate} = \frac{7}{14} \times 100 = \boxed{50\%} \text{ 이다.}$$



# Exercise 8.12



$$b'-1 : 2$$

(b)

$$c+1-n : b'$$

(c)

$$\begin{aligned}
 & 31 - (c+2-n) + 1 \\
 &= 32 - (c+2-n) \\
 &= \boxed{30 - (c-n) \text{ bits}}
 \end{aligned}$$

(d)

$$\begin{aligned}
 & \# \text{ tag bits} \times \# \text{ blocks} \\
 &= (30 - (c-n)) \times 2^{c - (b'-2)} \\
 &= \boxed{(30 - (c-n)) \times 2^{c+2-b'}}
 \end{aligned}$$

### Exercise 8.14

(a) 메모리에 있는 word가 하나는 on-chip cache 에 또 하나는 off-chip cache 에 있다.

(b) on-chip cache가 4-way set associative 이기 때문에  $set = \frac{512}{4} = 128 = 2^7$  개이고,

Set bits 는 7 bits 이다. 1 word = 4 bytes 이기 때문에

block size 는  $\frac{16 \text{ bytes}}{4 \text{ bytes}} = 4 \text{ word}$  이고, 따라서

block offset bit 는 2 bits 이다. 그리고 byte offset bits 는 2 bit 이다.

tag bits 는  $32 - (7 + 2 + 2) = 21 \text{ bits}$  이다.

Second-level cache 는 direct mapped 이기 때문에

$Set = \frac{256K}{1} = 256K$  개 이다. 따라서 set bits 는 18 bits 이다.

block size 는  $\frac{16 \text{ bytes}}{4 \text{ bytes}} = 4 \text{ word}$  이고, 따라서 block offset

bit 는 2 bits 이다. 그리고 byte offset bits 는 2 bit 이다.

tag bits 는  $32 - (18 + 2 + 2) = 10 \text{ bits}$  이다.

(c)  $AMAT = t_{cache} + MR_{cache} (t_{MM} + MR_{MM} t_{VM})$  인데

virtual memory 가 없고 L2 cache 가 있으므로

$$AMAT = t_{cache} + MR_{cache} (t_{L2\ cache} + MR_{L2\ cache} t_{MM}) \text{ 이다.}$$

$t_{cache}$  는 on-chip cache 에 access 하는 time 이고,

$MR_{cache}$  는 on-chip cache 에서의 miss rate 이다.

$t_{L2\ cache}$  는 second-level cache 에 access 하는 time 이고,

$MR_{L2\ cache}$  는 second-level cache 에서의 miss rate 이다.

$t_{MM}$  은 main memory 에 access 하는 time 이다.

$$\therefore t_a + (1-A)(t_b + (1-B)t_m)$$

(d) on-chip cache 를 이용하면 second level cache 에서

hit 임에도 불구하고 on-chip cache 에서 hit 가 된다.

하지만 on-chip cache 가 disabled 되면 second level

cache 에 access 하기 때문에 second level cache 의

hit rate 가 올라간다.



### Exercise 8.16

(a)  $AMAT = t_{cache} + MR_{cache} t_{MM}$

$$\text{Cycle time} = t_{cache} = \frac{1}{1\text{GHz}} = 1\text{ns}$$

$$MR_{cache} = 5\% = 0.05$$

$$\therefore AMAT = 1\text{ns} + 0.05 \times 60\text{ns} = \boxed{4\text{ns}}$$

(b) Average memory access time 이 4ns 이기 때문에

$$\frac{4\text{ns}}{1\text{ns}} = 4 \text{ clocks 이다.} \quad \text{main memory 에 access 하는}$$

$$\text{time 이 } 60\text{ns 이기 때문에 } \frac{60\text{ns}}{1\text{ns}} = 60 \text{ clocks 이다.}$$

$$\therefore CPI = 4 + 4 = 8 \text{ cycles (load)}$$

$$CPI = 4 + 3 = 7 \text{ cycles (store)}$$

(c)

Average CPI

$$= \left( (0.11 + 0.02) \times 3 \right) + (0.52 \times 4) + (0.1 \times 7) + (0.25 \times 8)$$
$$= \boxed{5.17 \text{ cycles}}$$

(d)

$$\text{Average CPI} = 5.17 + (0.07 \times 60) = \boxed{9.37 \text{ cycles}}$$

Exercise 8.20

(a) 8MB 이므로  $2^3 \times 2^{20} = 2^{23}$  이니까  $\boxed{23 \text{ bits}}$  이다.

(b)  $\frac{2^{32}}{2^{12}} = \boxed{2^{20} \text{ virtual pages}}$

(c)  $\frac{8\text{MB}}{4\text{kB}} = \frac{2^{23}}{2^{12}} = \boxed{2^{11} \text{ physical pages}}$

(d) Virtual page number :  $\log_2 2^{20} = 20 \text{ bits}$   
physical page number :  $\log_2 2^{11} = 11 \text{ bits}$

(e)  $\frac{\# \text{Virtual pages}}{\# \text{physical pages}} = \frac{2^{20}}{2^{11}} = 2^9$  개의 virtual pages 가

physical pages 에 mapped 된다.

address 0x00000000 주위의 data를 operating 하는 memory address 가 0x01000000 인 프로그램을 생각해보면, physical page에 0이 두개의 virtual page 사이에서 계속 바뀌므로 심각한 스래싱은 유발한다.

(f)

Virtual page가  $2^{20}$  개 이므로

$2^{20}$  page table entries 이다.

(g)

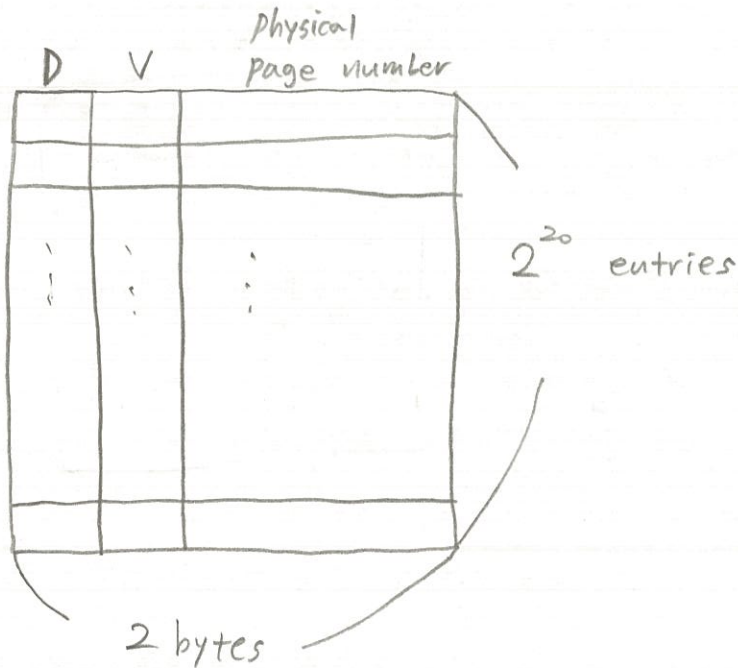
각각의 entry는 physical page number 11 bits 와

valid bit 1 bit, dirty bit 1 bit 해서

총  $11 + 1 + 1 = 13$  bits 가 필요하다. 1 byte 가 8 bits 이기

때문에 2 bytes 가 필요하다.

(h) page table 의 전체 사이즈는  $2 \text{ bytes} \times 2^{20} = 2^{21} \text{ bytes}$  이다.



Exercise 8.22

(a)

TLB가 없을 때 :

$$\begin{aligned} \text{AMAT} &= t_{MM} + \left[ t_{\text{cache}} + MR_{\text{cache}} (t_{MM} + MR_{MM} t_{VM}) \right] \\ &= 100 + \left[ 1 + 0.02 (100 + 0.000003 (1000000)) \right] \\ &= \boxed{103.06 \text{ cycles}} \end{aligned}$$

TLB가 있을 때 :

$$\begin{aligned} \text{AMAT} &= \left[ t_{TLB} + MR_{TLB} (t_{MM}) \right] + \left[ t_{\text{cache}} + MR_{\text{cache}} (t_{MM} + MR_{MM} t_{VM}) \right] \\ &= \left[ 1 + 0.0005 (100) \right] + \left[ 1 + 0.02 (100 + 0.000003 \times 1000000) \right] \\ &= \boxed{4.11 \text{ cycles}} \end{aligned}$$

(b) valid bit = 1 bit

virtual page number = 20 bits

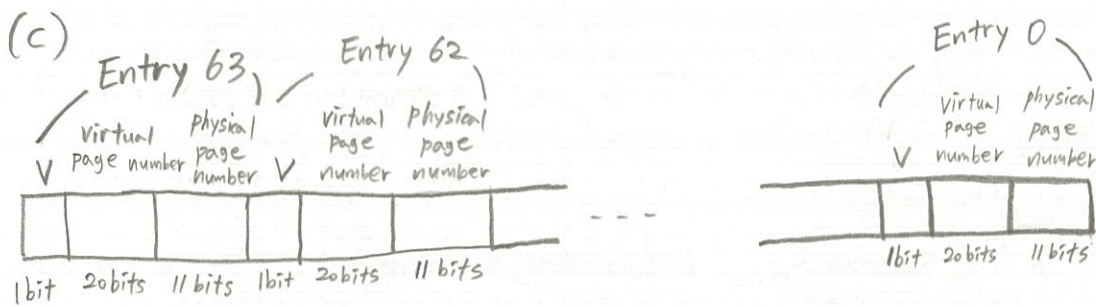
physical page number = 11 bits

각 entry 당 비트수는 valid bit 와 virtual page number,

physical page number을 더한  $1 + 20 + 11 = 32 \text{ bits}$ 이다.

64 entries 이기 때문에 TLB의 전체 사이즈는

$$64 \times 32 \text{ bits} = \boxed{2048 \text{ bits}} \text{ 이다.}$$



(d) TLB는 Fully associative 이므로 depth 는 1012,  
width 는 2048 bit 이다.

∴ 1 × 2048 bit SRAM