

WRITTEN

## HOMEWORK #5

- 1.) Given Fully Associative cache of size  $2^{10}$  bytes. Line size of 16 bytes, show address of 21 bits is partitioned.

- Cache size =  $2^{10} = 1024$  bytes

1024 512 256 128 64 32 16 8 4 2<sup>10</sup>

- Line size =  $2^4 = 16$  bytes

- Offset size =  $\log_2 16 = 4$  bits

- Tag =  $2^9 - \log_2 16 = 25$  bits

partition →

TAG	Offset
28 - 4	3 - 0

- 2.) Given Direct Mapped cache of size  $2^{12}$  bytes. Line size of 32 bytes, show Address of 50 bits is partitioned.

- Cache size =  $2^{12} = 4096$  bytes

- Line size =  $2^5 = 32$  bytes

- Set size =  $\log_2 4096 = 12$  bits

- ↪ Offset =  $\log_2 32 = 5$  bits

- Tag =  $50 - 12 - 5 = 33$  bits remaining

partition →

TAG	SET	OFFSET
49-12	11-5	4-0

- 3.) Given 5-way set Associative cache of size 5242880 bytes with line size of 64 bytes show how an Address of 64 bits would be partitioned

- Cache size = 5242880 bytes

- Line size =  $2^6 = 64$  bytes

- # of sets =  $5 \rightarrow 50$  4-0 bits

- Set size =  $5242880 / 5 = 1,048,576$

- ↪  $1,048,576 = 2^{20} \rightarrow 20$  bits

- Offset =  $2^6 = 64$  bytes 6 bits

partition →

TAG	SET	OFFSET
63-20	19-6	5-0

4.) String of hex address references as byte addresses : 1, 2, 3, 1A.

A, 1B, 16, 14, 3, 12, 9, 23, 3A, 5, 19, 1, 9

1) total size (Direct Mapped Cache) = 16 bytes

line size = 1 byte =  $2^0 \rightarrow 0$  bits offset

	Reference	Hit / Miss	Set	Tag
A B C D E F 10 11 12 13 14 15	1	Miss	1, 5, 9	0 0001, 0101, 1001
T S 10 00 32 44 21 0000 0001 = 1 0000 0010 = 2 0000 0011 = 3 0001 1010 = 1A 0000 1010 = A 0001 1011 = 1B 0001 0110 = 16 0001 0100 = 14 0001 0010 = 12 0000 1001 = 9 0010 0011 = 23 0011 1010 = 58 0000 0101 = 5 0001 1001 = 19	2	Miss	2, 4, 6, 11	1 0010, 0100, 0110, 1011
17 caches accesses	3	Miss	3	2 0011
	1A = 26	Miss	10	3 1010
	A = 10	Miss		
	1B = 27	Miss		
	16	Miss		
	14	Miss		
	3	Hit ✓		
	12	Miss		
	9	Miss		
	23	Miss		
	3A = 58	Miss		
	5	Miss		
	19	Miss		
	1	Hit ✓		
	9	Miss		

Hit Rate = 2 / 17

~~0000 0001 = 1~~      ~~0000 0001~~  
~~0000 0010 = 2~~      ~~0000 0010~~  
~~0000 0011 = 3~~      ~~0000 0011~~  
~~0001 1010 = 1A~~      ~~0001 1010~~  
~~0001 1011 = 1B~~      ~~0001 1011~~  
~~0001 0110 = 16~~      ~~0001 0110~~  
~~0001 0100 = 14~~      ~~0001 0100~~  
~~0001 0010 = 12~~      ~~0001 0010~~  
~~0000 1001 = 9~~      ~~0000 1001~~  
~~0010 0011 = 23~~      ~~0010 0011~~  
~~0011 1010 = 58~~      ~~0011 1010~~  
~~0000 0101 = 5~~      ~~0000 0101~~  
~~0001 1001 = 19~~      ~~0001 1001~~

2) Direct Mapped Cache = 16 bytes

Line Size = 4 bytes

	Reference	Hit / Miss	Set	Tag
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1	Miss	0	0 $\log_2 16 = 4$ set
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1	2	Hit ✓	1	0 $\log_2 4 = 2$ offset
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0	3	Hit ✓	2	0 $\frac{4}{2} = 2$ offset
0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1	1A	Miss	3	
0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0	A	Miss		
0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 1	1B	Miss		
0 0 0 0 0 0 0 0 0 0 1 0 1 0 1 1	16	Miss		
0 0 0 0 0 0 0 0 0 1 0 1 0 1 1 0	14	Hit ✓		

3	Hit
12	MISS
9	MISS
23	MISS
3A	MISS
5	MISS
19	MISS
1	MISS
9	MISS

HIT RATE = 4/17

3.) 2-Way Associative Cache = 16 bytes big

Line Size = 1 byte

LRU replacement strategy

Reference	Hit / Miss	Set		
		1	2	3
1	MISS	0		
2	MISS	1	1	0
3	MISS	2	7	2
1A	MISS	3	0	4
A	MISS	4	2	
1B	MISS	5	0	
16	MISS	6	2	
14	MISS	7		
3	Hit ✓			
12	MISS			
9	MISS			
23	MISS			
3A	MISS			
5	MISS			
19	MISS			
1	MISS			
9	MISS			

HIT Rate = 1/17

4.) Fully Associative Cache = 16 bytes

Line Size = 1 byte

LRU Replacement Strategy

Reference	Hit / Miss	Set	Tag
1 1 1 1 A A P P	1 MISS	1	1
2 2 2 2 B B B B	2 MISS		2
3 3 3 3 16 16 16 16	3 MISS		3
IA IA IA IA IA IA IA IA			
A 3 1B 1B 16 16 14 14	1A MISS		1A
	A MISS		A
	1B MISS		1B
	16 MISS		16
	14 MISS		14
	3 HIT		12
	12 MISS		9
	9 MISS		23
	23 MISS		3A
	3A MISS		5
	5 MISS		19
	19 MISS		
	1 HIT		
	9 HIT		

Hit Rate = 3/17

5.) Fully Associative Cache = 16 bytes

Line Size = 4 bytes

LRU Replacement Strategy

Reference	Hit / Miss
1	MISS
2	HIT ✓
3	HIT ✓
IA	MISS
A	MISS
1B	HIT ✓
16	MISS
14	HIT ✓

3	Hit ✓
12	MISS
9	MISS
23	MISS
3 A	MISS
5	MISS
19	MISS
1	MISS
9	MISS

Hit Rate = 5/17

5.) CPU has  $2^{16}$  bytes of RAM, virtual memory system whose virtual address is 32 bits long & is using page size  $2^{10}$  bytes.

1.) physical pages =

$$2^{16} / 2^{10} = 2^6 = 64$$

2.) virtual pages =

$$2^{32} / 2^{10} = 2^{22} = 4194304$$

3.) each entry size in page table (bits) =

$$\log_2(\text{physical pages}) = \log_2 64 = 2^6 \rightarrow 6 \text{ bits}$$

4.) page table size for single process (bytes) =

$$6 \times 2^{22} \text{ bits} = 1524288 = 2^{19} \text{ bytes}$$

5.) Problems with a page table that size?

Yes, it is larger than the memory.

6.) CPU that has  $2^{18}$  bytes of RAM, virtual memory system w/ secondary page tables. virtual address is 32 bits long & page size is  $2^{10}$  bytes. Size of secondary pages = page size.

1.) physical pages =

$$2^{18} / 2^{10} = 2^8 = 256$$

2.) virtual pages =

$$2^{32} / 2^{10} = 2^{22}$$

3.) secondary page tables =

$$2^{22} / 2^{10} = 2^{12}$$

4.) each entry size primary page table (bytes) =

$$\log_2 256 = 2^8 \rightarrow 8 \text{ bits} = 1 \text{ byte}$$

5.) each entry size secondary page table (bytes) =

Secondary pages size = page size  $\rightarrow 1 \text{ byte}$

6.) primary page table size (bytes) =

$$2^{12} \times 1 = 2^{12} \text{ bytes}$$