

King Fahd University of Petroleum and Minerals
 College of Computer Sciences and Engineering
 Department of Computer Engineering

COE 308 – Computer Architecture (T031)

Homework # 02 (due date: Sunday 02/11/2003)

***** Show all your work. No credit will be given if work is not shown! *****

Problem # 1: Consider a computer system that has a cache consisting of 128 blocks. The MM contains 64K blocks, each consisting of 64 words. Answer the following assuming (i) fully associative cache, (ii) direct cache, then (iii) set associative with eight-block sets:

1. How many bits are needed for the MM address?
2. How many bits are needed for each of the TAG, SET, and WORD fields?

Solution:

MM size = $64K \times 64 = 2^{22}$

(i) Fully Associative:

- | | |
|------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------|
| 1. # MM address bits = $\lceil \log_2 \text{MM size} \rceil = \lceil \log_2 2^{22} \rceil = 22$ bits | |
| 2. # WORDs per block = 64 words | \Rightarrow # WORD field bits = $\lceil \log_2 64 \rceil = 6$ bits |
| # SETs = 1 | \Rightarrow # SET field bits = $\lceil \log_2 1 \rceil = 0$ bits |
| # TAGs = # MM blocks / # SETs = $64K / 1 = 64K = 2^{16}$ | \Rightarrow # TAG field bits = $\lceil \log_2 2^{16} \rceil = 16$ bits |

(ii) Direct:

- | | |
|------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------|
| 1. # MM address bits = $\lceil \log_2 \text{MM size} \rceil = \lceil \log_2 2^{22} \rceil = 22$ bits | |
| 2. # WORDs per block = 64 words | \Rightarrow # WORD field bits = $\lceil \log_2 64 \rceil = 6$ bits |
| # SETs = 128 | \Rightarrow # SET field bits = $\lceil \log_2 128 \rceil = 7$ bits |
| # TAGs = # MM blocks / # SETs = $64K / 128 = 512 = 2^9$ | \Rightarrow # TAG field bits = $\lceil \log_2 2^9 \rceil = 9$ bits |

(i) Set Associative: Eight-block sets \Rightarrow 8-way set associative

- | | |
|------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------|
| 1. # MM address bits = $\lceil \log_2 \text{MM size} \rceil = \lceil \log_2 2^{22} \rceil = 22$ bits | |
| 2. # WORDs per block = 64 words | \Rightarrow # WORD field bits = $\lceil \log_2 64 \rceil = 6$ bits |
| # SETs = $128 / 8 = 16$ | \Rightarrow # SET field bits = $\lceil \log_2 16 \rceil = 4$ bits |
| # TAGs = # MM blocks / # SETs = $64K / 16 = 4K = 2^{12}$ | \Rightarrow # TAG field bits = $\lceil \log_2 2^{12} \rceil = 12$ bits |

Problem # 2: Solve problem 3.5 from the textbook.

Solution:

- Cache consists of 256 blocks of 16 bytes each
- Each cache block will have an associated TAG field
- The processor will generate a total of 2^{32} possible addresses
- The number of possible blocks generated by the processor is $2^{32} / 16 = 2^{28}$
- The # bits associated with the TAG, SET, and WORD fields are as follows:

o # WORDs per block = 16 words	\Rightarrow # WORD field bits = $\lceil \log_2 16 \rceil = 4$ bits
o # SETs = 256	\Rightarrow # SET field bits = $\lceil \log_2 256 \rceil = 8$ bits
o # TAGs = $2^{28} / 256 = 2^{20}$	\Rightarrow # TAG field bits = $\lceil \log_2 2^{20} \rceil = 20$ bits

Address	TAG	SET	WORD	Hit/Miss	Comment
000 53272105	53272	10	5	Miss	
000 53502120	53502	12	0	Miss	
000 53271130	53271	13	0	Miss	
000 53272106	53272	10	6	Hit	Same SET & TAG as 1 st reference
000 53502124	53502	12	4	Hit	Same SET & TAG as 2 nd reference
000 53261130	53261	13	0	Miss	Same SET as 3 rd reference but different TAG
000 53272104	53272	10	4	Hit	Same SET & TAG as 1 st reference

Total # of misses generated = 4

Problem # 3: Solve problem 3.14 from the textbook.

Solution:

Address	Tag	Hit/Miss	C ₀	C ₁	C ₂	C ₃	Subsequent actions
		Initialization	0	0	0	0	
2		Miss	0	1	1	1	Block 0 filled
6		Miss	1	0	2	2	Block 1 filled
9		Miss	2	1	0	3	Block 2 filled
7		Miss	3	2	1	0	Block 3 filled
2		Hit	0	3	2	1	Block 0 accessed
3		Miss	1	0	3	2	Block 1 replaced
2		Hit	0	1	3	2	Block 0 accessed
9		Hit	1	2	0	3	Block 2 accessed
6		Miss	2	3	1	0	Block 3 replaced
2		Hit	0	3	2	1	Block 0 accessed
7		Miss	1	0	3	2	Block 1 replaced
4		Miss	2	1	0	3	Block 2 replaced

Problem # 4: Solve problem 3.15 from the textbook.

Solution:

F = Fill
A = Access
R = Replace

TAG	2	6	9	7	2
Block	0	1	2	3	0
Comment	F	F	F	F	A
LRU			3	0	1
Block					

TAG	3	2	9	6	2
Block	1	0	2	3	0
Comment	R	A	A	R	A
LRU	2	2	3	1	1
Block					

TAG	7	4
Block	1	2
Comment	R	R
LRU	2	3
Block		

Problem # 5: Consider a 4×8 array of numbers, A. Assume that each number in the array occupies one word in MM and that the array elements are stored **column-major** in MM starting from location 1000 to location 1031. The cache consists of 8 blocks each consisting of just two words. Assume also that whenever needed, LRU replacement policy is used. Analyze the cache for the following segment of a program:

```

sum = 0;
For i = 0 to 1
    For j = 0 to 7
        sum = sum + A(i, j)
    EndFor
EndFor

```

Fill out the following assuming (i) direct mapping, then (ii) fully associative mapping. Also, calculate the hit ratio, the number of block replacements, and the percentage of cache utilization in each case.

Solution:

(i) Direct Mapping

Request	Cache Hit/Miss	MM Address	Cache Block number	Cache Content									
				B0	B1	B2	B3	B4	B5	B6	B7		
A(0, 0)	Miss	1000	0	00	10								
A(0, 1)	Miss	1004	2	00	10		01	11					
A(0, 2)	Miss	1008	4	00	10		01	11		02	12		
A(0, 3)	Miss	1012	6	00	10		01	11		02	12		03 13
A(0, 4)	Miss	1016	0	04	14		01	11		02	12		03 13
A(0, 5)	Miss	1020	2	04	14		05	15		02	12		03 13
A(0, 6)	Miss	1024	4	04	14		05	15		06	16		03 13
A(0, 7)	Miss	1028	6	04	14		05	15		06	16		07 17
A(1, 0)	Miss	1001	0	00	10		05	15		06	16		07 17
A(1, 1)	Miss	1005	2	00	10		01	11		06	16		07 17
A(1, 2)	Miss	1009	4	00	10		01	11		02	12		07 17
A(1, 3)	Miss	1013	6	00	10		01	11		02	12		03 13
A(1, 4)	Miss	1017	0	04	14		01	11		02	12		03 13
A(1, 5)	Miss	1021	2	04	14		05	15		02	12		03 13
A(1, 6)	Miss	1025	4	04	14		05	15		06	16		03 13
A(1, 7)	Miss	1029	6	04	14		05	15		06	16		07 17

Hit ratio = 0 / 16 = 0, **# block replacements** = 12 (shaded areas in the table), **Cache utilization%** = 4 / 8 = 50%

(ii) Fully Associative Mapping

Request	Cache Hit/Miss	MM Addr.	Cache Block number	Cache Content												
				B0	B1	B2	B3	B4	B5	B6	B7					
A(0, 0)	Miss	1000	0	00	10											
A(0, 1)	Miss	1004	1	00	10	01	11									
A(0, 2)	Miss	1008	2	00	10	01	11	02	12							
A(0, 3)	Miss	1012	3	00	10	01	11	02	12	03	13					
A(0, 4)	Miss	1016	4	00	10	01	11	02	12	03	13	04	14			
A(0, 5)	Miss	1020	5	00	10	01	11	02	12	03	13	04	14	05 15		
A(0, 6)	Miss	1024	6	00	10	01	11	02	12	03	13	04	14	05 15	06 16	
A(0, 7)	Miss	1028	7	00	10	01	11	02	12	03	13	04	14	05 15	06 16	07 17
A(1, 0)	Hit	1001	0	00	10	01	11	02	12	03	13	04	14	05 15	06 16	07 17
A(1, 1)	Hit	1005	1	00	10	01	11	02	12	03	13	04	14	05 15	06 16	07 17
A(1, 2)	Hit	1009	2	00	10	01	11	02	12	03	13	04	14	05 15	06 16	07 17
A(1, 3)	Hit	1013	3	00	10	01	11	02	12	03	13	04	14	05 15	06 16	07 17
A(1, 4)	Hit	1017	4	00	10	01	11	02	12	03	13	04	14	05 15	06 16	07 17
A(1, 5)	Hit	1021	5	00	10	01	11	02	12	03	13	04	14	05 15	06 16	07 17
A(1, 6)	Hit	1025	6	00	10	01	11	02	12	03	13	04	14	05 15	06 16	07 17
A(1, 7)	Hit	1029	7	00	10	01	11	02	12	03	13	04	14	05 15	06 16	07 17

Hit ratio = 8 / 16 = 0.5, **# block replacements** = 0, **Cache utilization%** = 8 / 8 = 100%