COMPUTER ARCHITECTURE & ASSEMBLY LANGUAGE
14:332:331
Rutgers University
Spring 2016

Homework 5
Due: April 29, 2016

Note: Question 6 is for your practice. You do not need to answer question 6.

1. Suppose that your processor has 4MB data cache and its block size is 64B. Physical address to access the memory is 50-bit wide (addr[49:0]). For each of the following cache structures, calculate TAG size and index size.
   (a) A direct-mapped cache implementation
   (b) A 4-Way set associative cache implementation
   (c) A fully associative cache implementation

Answer:

a) # of blocks = 4MB/64B=$2^{22}/2^{6=}2^{16}$→index:16 bits
Tag size= 50- 6 (for block size) – 16 (for index)=28

b) #of set= $2^{16}/4=2^{14}$→index:14 bits
Tag size= 50- 6 (for block size) – 14 (for index)=30

c) Tag size= 50- 6 (for block size) = 44 ; Index Size =0

2. The following **memory addresses** are used consecutively by a running program (from left to right, shown in decimal). Note that the followings are memory address not block number:

   500, 40, 360, 116, 836, 224, 1348, 18, 440, 100, 1024, 44, 168, 40, 104

   In each of the following cache structures, compute the number of hits, misses and the final values stored in each cache location (show finally which block of memory is in each cache block). Each word is 4-bytes and the memory size is 8Kbyte

   (a) Direct-mapped cache with 16-word blocks and a total size of cache is 128 words of data
   (b) 2-way set associative cache with 16-word blocks and a total size of cache is 128 words of data. (LRU replacement)

Answer:

a) # of cache block= 128/16 = 8
Size of each block=16*4=64 byte

Memory Address	Block # in Main Memory	Block in Cache	Word address included	Hit/Miss
500	7(=500/64)	7 (=7%8)	448, 452,……508	Miss
40	0	0	0, 4, 8……60	Miss
360	5	5	320,324……..380	Miss
116	1	1	64,68….124	Miss
836	13	5	832,836,….892	Miss
224	3	3	192,196,…….252	Miss
1348	21	5	1344, 1348,…..1404	Miss
18	0	0	0, 4, 8,…..60	Hit
440	6	6	384,388,……444	Miss
100	1	1	64,6 8,…..124	Hit
1024	16	0	1024, 1028,….1084	Miss
44	0	0	0, 4, 8,…..60	Miss
168	2	2	128, 132,……188	Miss
40	0	0	0, 4, 8,…..60	Hit
104	1	1	64, 68,…..124	Hit

➔ Hit rate=4/15=26.6%, Miss_rate=73.4%. The final values stored in cache are as below:

Block 0 of cache	0, 4, 8,…..60
Block 1 of cache	64, 68,…..124
Block 2 of cache	128, 132,……188
Block 3 of cache	192,196,…….252
Block 4 of cache	Any value from before
Block 5 of cache	1344, 1348,…..1404
Block 6 of cache	384,388,……444
Block 7 of cache	448, 452,……508

b) # of cache block= 128/16=8
# of sets = 8/2 =4
Size of each block=16*4=64 byte

Memory Address	Block # in Main Memory	Set in Cache	Word address included In block 0 of set 3	Word address included In block 1 of set 3	Word address included In block 0 of set 0	Word address included In block 1 of set 0	Word address included In block 0 of set 1	Word address included In block 1 of set 1	Word address included In block 0 of set 2	Word address included In block 0 of set 2	Hit/ Miss
500	7(=500/64)	3	448,452 ….508								Miss
40	0	0	448,452 ….508		0,…60						Miss
360	5	1	448,452 ….508		0,…60		320…380				Miss
116	1	1	448,452 ….508		0,…60		320…380	64,….124			Miss
836	13	1	448,452 ….508		0,…60		832,….892	64,….124			Miss
224	3	3	448,452 ….508	192...252	0,…60		832,….892	64,….124			Miss
1348	21	1	448,452 ….508	192...252	0,…60		832,….892	1344,...1404			Miss
18	0	0	448,452 ….508	192...252	0,…60		832,….892	1344,...1404			Hit
440	6	2	448,452 ….508	192...252	0,…60		832,….892	1344,...1404	384… ….444		Miss
100	1	1	448,452 ….508	192...252	0,…60		64,….124	1344....1404	384… ….444		Miss
1024	16	0	448,452 ….508	192...252	0,…60	1024 …..1084	64,….124	1344,...1404	384… ….444		Miss
44	0	0	448,452 ….508	192...252	0,…60	1024 …..1084	64,….124	1344,...1404	384… ….444		Hit
168	2	2	448,452 ….508	192...252	0,…60	1024 …..1084	64,….124	1344...1404	384… ….444	128… …188	Miss
40	0	0	448,452 ….508	192...252	0,…60	1024 …..1084	64,….124	1344...1404	384… ….444	128… …188	Hit
104	1	1	448,452 ….508	192...252	0,…60	1024 …..1084	64,….124		384… ….444	128… …188	Hit

Hit rate=4/15=26.6%, Miss_rate=73.4%.

3. For each part of Question 2, find the size of the cache required to hold the data (as mentioned in 2.a and 2.b above the cache should hold 128 words of data)?
Answer:

a) # of bits for memory addressing=$\log_2$ 8K=13
Index_size= $\log_2$ 8=3, offset= $\log_2$ 64=6→ tag_size=13-3-6=4

For each block we need: 1 (for valid)+4(for tag)+64*8 (for data per block)=517 bit

➔ Total cache size=517*8(=number of blocks)=4136 bit=517 byte

b) # of bits for memory addressing=$\log_2$ 8K=13

Index_size= $\log_2$ 4=2, offset= $\log_2$ 64=6→ tag_size=13-2-6=5

For each block we need: 1 (for valid)+5(for tag)+64*8 (for data per block)=518 bit

➔ Total cache size=518*8(=number of blocks)=4144 bit=518 byte

4. In each of the following three cases, calculate the CPI for a processor with these specifications:

- Base CPI=2
- Processor speed=2 GHz
- Main memory access time=100ns
- First-level cache miss rate per instruction=7%

(a) We only have a first level (L1) cache

(b) Along with L1 cache, we also have a second level direct-mapped cache in which

. Second-level cache direct-mapped speed=12 cycle

. Global miss rate with second-level cache direct-mapped=3.5%

(c) Along with L1 cache, we also have a second level 8-way set associative cache in which:

.Second-level cache 8-way set-associative speed=28 cycle

.Global miss rate with second-level cache 8-way set-associative=1.5%

Answer:

Main memory access time =100ns, clk_cycle_time=1/(2GHZ)=0.5ns ➔ Main memory access time=100/0.5=200 clock cycle

a) CPI= 2+7%*200=16

b) CPI= 2+7%*12+3.5%*200= 9.84

c) CPI= 2+7%*28+1.5%*200= 6.96

5.

(a) What is the average memory access time (AMAT) if a cache uses write-back strategy and 20% of the data blocks to be swapped out are dirty. Assume that the miss rate is 10%, the hit time of the cache is 1 cycle and the miss penalty is 10 cycles for the data blocks that are not dirty and 30 cycles for those blocks that are dirty.

(b) What is the speedup up if we add a “write-buffer” that eliminates 50% of the stall cycles to write back the dirty blocks?

Answer:

a) AMAT=1+10%(30*20%+10*(1-20%))=1+10%(30*20%+10*(80%))=2.4

b) AMT=1+10%(15*20%+10*80%)=2.1

Speedup=2.1/2.4=1.14

6. As discussed, virtual memory uses a page table to track the mapping of virtual addresses to physical addresses. This exercise shows how this table must be updated as addresses are accessed. The following data constitutes a stream of virtual addresses as seen on a system. Assume 4 KB pages, a 4-entry fully associative TLB, and LRU replacement. If pages must be brought in from disk, increment the next largest page number.

Virtual addresses: 4669, 2227, 13916, 34587, 48870, 12608, 49225

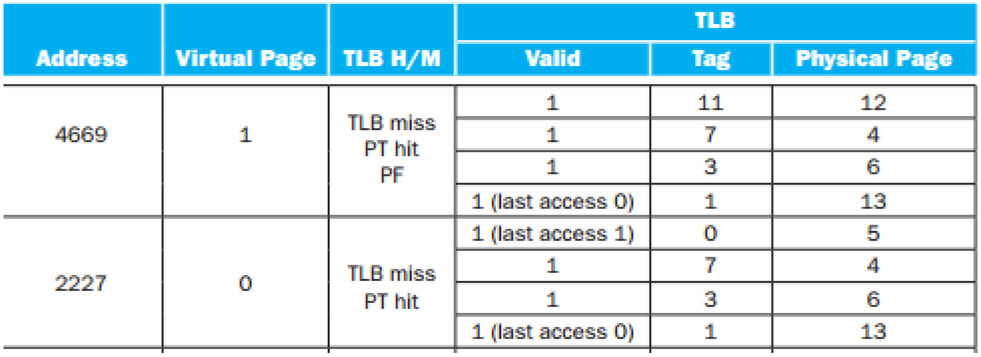
TLB

Valid	Tag	Physical Page Number
1	11	12
1	7	4
1	3	6
0	4	9

Page Table

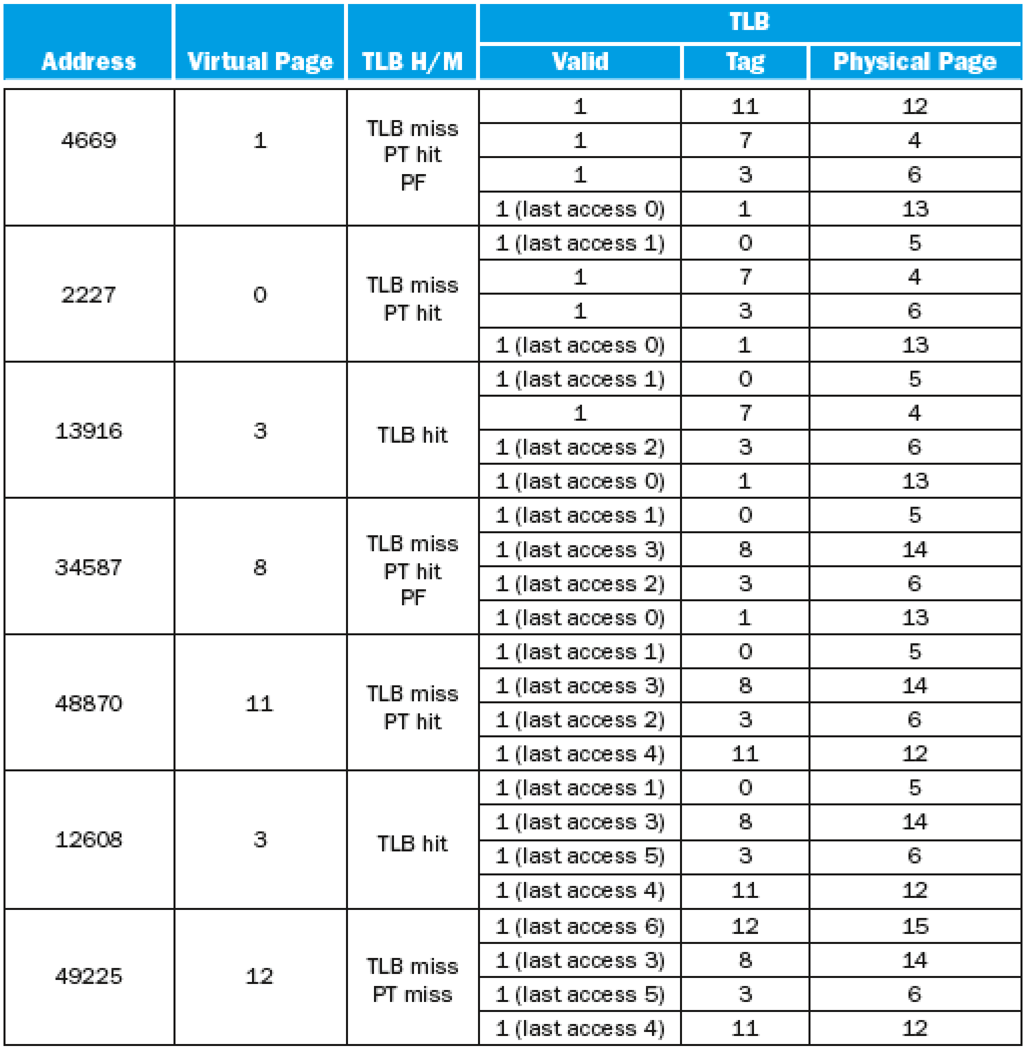
Valid	Physical Page or in Disk
1	5
0	Disk
0	Disk
1	68
1	9
1	11
0	Disk
1	4
0	Disk
0	Disk
1	3
1	12

Given the address stream shown, and initial TLB and Page Table States provided above, show if each reference is hit in the TLB, a hit in the page table, or a page fault. (To make it clear, the first two accesses have been shown below. Please continue in the same way)



Address	Virtual Page	TLB H/M	TLB		
			Valid	Tag	Physical Page
4669	1	TLB miss PT hit PF	1	11	12
			1	7	4
			1	3	6
			1 (last access 0)	1	13
2227	0	TLB miss PT hit	1 (last access 1)	0	5
			1	7	4
			1	3	6
			1 (last access 0)	1	13

Answer: The following table shows the status of TLB after each access.



Address	Virtual Page	TLB H/M	TLB		
			Valid	Tag	Physical Page
4669	1	TLB miss PT hit PF	1	11	12
			1	7	4
			1	3	6
			1 (last access 0)	1	13
2227	0	TLB miss PT hit	1 (last access 1)	0	5
			1	7	4
			1	3	6
			1 (last access 0)	1	13
13916	3	TLB hit	1 (last access 1)	0	5
			1	7	4
			1 (last access 2)	3	6
			1 (last access 0)	1	13
34587	8	TLB miss PT hit PF	1 (last access 1)	0	5
			1 (last access 3)	8	14
			1 (last access 2)	3	6
			1 (last access 0)	1	13
48870	11	TLB miss PT hit	1 (last access 1)	0	5
			1 (last access 3)	8	14
			1 (last access 2)	3	6
			1 (last access 4)	11	12
12608	3	TLB hit	1 (last access 1)	0	5
			1 (last access 3)	8	14
			1 (last access 5)	3	6
			1 (last access 4)	11	12
49225	12	TLB miss PT miss	1 (last access 6)	12	15
			1 (last access 3)	8	14
			1 (last access 5)	3	6
			1 (last access 4)	11	12