

QUIZ 4

CS 305

Autumn 2016

Quiz 4

Nov 2, 2016

Time Limit: 60 Minutes

IIT Roll No.:

Umesh Bellur

This exam contains 7 pages (including this cover page) and 7 problems. Check to see if any pages are missing. Enter all requested information on the top of this page, and put your IIT Roll Number clearly on the top of every page, in case the pages become separated.

You may *not* use any books or a calculator on this exam. The last page of this paper has the MIPS instructions for your reference.

You are required to show your work on each problem on this exam. The following rules apply:

- If you use any well known results you must indicate which one.
- Organize your work, in a reasonably neat and coherent way, in the space provided. Work scattered all over the page without a clear ordering will receive very little credit.
- Mysterious or unsupported answers will not receive ANY credit. A correct answer, unsupported by calculations, explanation, or algebraic work will receive no credit; an incorrect answer supported by substantially correct calculations and explanations will still receive partial credit.

Do not write in the table to the right.

Ques	Points	Score
1	6	
2	8	
3	6	
4	3	
5	2	
6	3	
7	5	
Total:	33	

1. Consider a 2-way set associative cache with 8 one-word blocks. The cache is initially empty. The computer system uses word addresses instead of byte addresses (i.e. a word is the minimum unit of memory access).

Assume that each word stored in the main memory has the same value as its address. For example, the word at address 2015 has a value of 2015. Consider a sequence of memory references to the following word addresses:

4, 16, 6, 14, 16, 20, 14, 18, 6, 4, 4, 6

- (a) (5 points) Suppose that the cache block replacement policy is LRU. In the table below, show the content of the cache after each memory reference. The first three rows have been filled in already.

Word Address Referenced	Hit or Miss	Contents of cache blocks AFTER reference			
		Set 0		Set 1	
		block 1	block 2	block 1	block 2
4	Miss	4	-	-	-
16	Miss	4	16	-	-
6	Miss	4	16	-	-
14	MISS	4	16	-	-
16	HIT	4	16	-	-
20	MISS	20	16	-	-
14	HIT	20	16	-	-
18	MISS	20	16	-	-
6	MISS	20	16	-	-
4	MISS	20	4	-	-
4	HIT	20	4	-	-
6	MISS	20	4	-	-

10 "— means empty

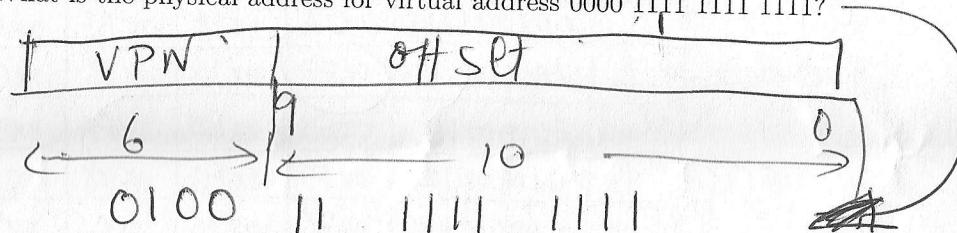
- (b) (1 point) Compute the miss rate for this trace for the LRU replacement policy.

$$\frac{8}{12} = 0.67$$

2. **Virtual memory:** The virtual memory system of a 16-bit computer architecture has a page size of 1024 (2^{10}) bytes. There are a total of 16 (2^4) physical pages (frames) in the physical memory. The first few entries of a page table are shown below (all fields in the page table are binary numbers):

Valid bit	Physical page number
1	0101
1	1100
0	1110
1	0100
1	0001
1	1010
0	0110
1	1001
	⋮

- (a) (2 points) Show the partition of the virtual address (indicate the number of bits for each field). What is the physical address for virtual address 0000 1111 1111 1111?



- (b) (2 points) Suppose we add a four-entry, fully associative TLB to the virtual memory system. The content of the TLB is shown below (all are binary numbers). Which of the following virtual address(es) will result in TLB hit?

Address 1: 0000 1111 1111 1111 — H
 Address 2: 0000 0100 0000 0000 — M
 Address 3: 0101 0110 1100 1100 — H
 Address 4: 1000 0000 1111 0000 — M

Valid Bit	Virtual Page Number	Physical Page no.
1	000011	0100
1	100011	0011
1	010101	1000
0	000001	1111

TLB

- (c) (4 points) Suppose we also add a direct-mapped cache to the virtual memory system. The cache has 8 blocks with a block size of 4 bytes. The content of the cache is shown below (all are binary numbers except the bytes in the cache blocks). This cache takes a physical address for cache lookup.

Given virtual (byte) address 0001 1100 1111 0000, describe the key steps in order to obtain its content. What is its content? Please use the cache given in this question, the page table in (a), and the TLB in (b).

Valid bit	Tag	Cache block Content			
		byte 0	byte 1	byte 2	byte 3
0 0 0 →	1	0/011100	0xAB	0xFA	0xDE
0 0 1 →	1	110000010	0x12	0x10	0x22
0 1 0 →	1	0/0010101	0x11	0x67	0xA0
0 1 1 →	1	001101111	0x1A	0x88	0xC8
1 0 0 →	1	100100111	0xFF	0x28	0x76
	1	101010111	0xEB	0xBC	0xFA
	1	011000101	0x45	0xEA	0x08
	1	100011000	0x86	0x16	0xC6

000111 001110000

TLB MISS

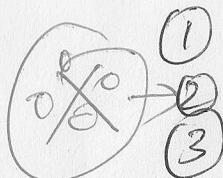
Page table 000111 → 1001

PA = 10010011110000
 Tag index b0

Valid 1 & Tag matches ⇒ Cache hit

value = 0xff

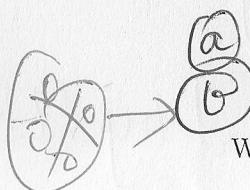
3. (6 points) You are designing the memory hierarchy for a new processor. The access latency of main memory is 200 cycles. You have the following choices for the L2 cache design:



- 2MB, 16-way set-associative. $T_{hit} = 30$ cycles. $\%miss = 1\%$.
- 1MB, 8-way set-associative. $T_{hit} = 20$ cycles. $\%miss = 5\%$.
- 512KB, 4-way set-associative. $T_{hit} = 15$ cycles. $\%miss = 10\%$.

}

and the following choices for L1 designs:



- 64KB, 8-way set-associative. $T_{hit} = 2$ cycles. $\%miss = 4\%$
- 32KB, 4-way set-associative. $T_{hit} = 1$ cycle. $\%miss = 5\%$.

Which cache designs would you choose and why?

L2 design is independent of L1 design

for L2

$$\textcircled{1} \quad T_{Avg} = 0.99 \times 30 + 0.01 \times 200 = 31.7$$

$$\textcircled{2} \quad T_{Avg} = 0.95 \times 20 + 0.05 \times 200 = 29$$

$$\textcircled{3} \quad T_{Avg} = 0.90 \times 15 + 0.1 \times 200 = 33.5$$

\rightarrow we pick (2)

NOW for L1 design

$$\textcircled{A} \quad T_{Avg} = 0.90 \times 2 + 0.04 \times 29 = 3.08$$

$$\textcircled{B} \quad T_{Avg} = 0.95 \times 1 + 0.05 \times 29 = 2.4$$

(B) is better

4. (3 points) Assume that for a given system, virtual addresses are 40 bits long and physical addresses are 30 bits long. There are 8 Kbytes of addressable entries per page. The TLB in the address translation path has 128 entries. How many virtual addresses can be quickly translated by the TLB? Would increasing the page size make your answer better or worse?

2^7 entries in TLB 8 each entry
 covers 2^{13} addressable entries
 $\Rightarrow 2^{20}$ addresses are covered by TLB.
 ↑ page size is better.

5. (2 points) A 2-way set associative write back cache with perfect LRU replacement requires 15×2^9 bits of storage to implement its tag store (including bits for valid, dirty and LRU). The cache block is 8 bytes large. What is the size of the data store of the cache in bytes?

2^9 sets 8 each set has 2 blocks of 8 bytes each
 $\Rightarrow 2^9 \times 2 \times 2^3 = 2^{13}$ bytes or 8 KB.

6. (3 points) Assume:

- A processor has a direct mapped cache
- Data words are 8 bits long (i.e. 1 byte)
- Data addresses are to the word
- A physical address is 20 bits long
- The tag is 11 bits
- Each block holds 16 bytes of data

How many blocks are in this cache?

20 bit address
 tag = 11 bits
~~11~~ Block offset = 4 bits
 \Rightarrow Index = 5 bits
 2⁵ = 32 blocks

7. (5 points) A program repeatedly performs a three-step process: It reads in a 4 KB block of data from disk, does some processing on that data, and then writes out the result as another 4 KB block elsewhere on the disk. Each block is contiguous and randomly located on a single track on the disk. The disk drive rotates at 7200RPM, has an average seek time of 8ms, and has a transfer rate of 20 MB/sec. The controller overhead is 2ms. No other program is using the disk or processor, and there is no overlapping of disk operation with processing. The processing step takes 20 million clock cycles, and the clock rate is 400MHz. What is the overall speed of the system in blocks processed per second assuming no other overhead?

$$\begin{aligned}
 \text{Disk Head time} &= \text{Seek time} \\
 &\quad + \text{rotational delay} \\
 &\quad + \text{transfer time} \\
 &\quad + \text{controller overhead} \\
 &= 8 + \left(0.5 \times 60 \times \frac{1000}{7200} \right) + 4 \times \frac{1024 \times 1000}{20 \times 2^{20}} \\
 &\quad + 2 \\
 &= 14.17 \text{ ms}
 \end{aligned}$$

$$\text{Processing time} = \frac{20 \times 10^6}{400 \times 10^6} = 0.05 \text{ s} = 50 \text{ ms}$$

$$\text{Disk Write time} = 14.17 \text{ ms}$$

$$\text{Total/Block} = 2 \times 14.17 + 50 = 78.34 \text{ ms}$$

$$\Rightarrow \text{Blocks/sec} = \frac{1000}{78.34} = \underline{\underline{12.76}}$$