浙江大学 20 14–20 15 学年秋冬学期

《数字系统设计 II》课程期末考试试卷

课程号：**111C0130**，开课学院： 信息与电子工程学系

考试试卷：√A 卷、B 卷（请在选定项上打√）

考试形式：√闭、开卷（请在选定项上打√）

允许带 1 张 A4 大小的手写资料和计算器入场

考试日期： 2015 年 1 月 24 日，考试时间： 120 分钟

诚信考试，沉着应考，杜绝违纪。

考生姓名：_____学号：_____ 所属院系（专业）：_____

题序	一	二	三	四	五	六	总分
得分							
评卷人							

1. PROCESSOR (8 points):

This question considers the basic MIPS 5-stage pipeline (IF, ID, EX, MEM, WB).

1) Assume that each of the above steps takes the amount of time specified in the Table Q1.1.

Fetch (IF)	Decode (ID)	Execute (EX)	Memory (MEM)	Write Back (WB)
320 ps	275 ps	310 ps	305 ps	250 ps

Table Q1.1

Given the times for the datapath stages listed above, what would the clock period be for the entire datapath? In a pipelined datapath, assuming no hazards or stalls, how many seconds will it take to execute an instruction?

2) Assume that you have the following sequence of pipelined instructions:

lw $6, 0($7)
add $8, $9, $10
sub $11, $6, $8

Where will the data operands that are processed during the EX stage of the subtract (sub) instruction come from? (Consider two situations: (1) there is no forwarding, and (2) there is full forwarding.)

2. PIPELINE (13 points):

Use the following MIPS code fragment:

```
I1:   ADDI   $3, $0, 100   # $3 = 100
I2:   ADD    $4, $0, $0    # $4 = 0
Loop:
I3:   LW     $5, 0($1)     # $5 = MEM[$1]
I4:   ADD    $4, $4, $5    # $4 = $4 + $5
I5:   LW     $6, 0($2)     # $6 = MEM[$2]
I6:   SUB    $4, $4, $6    # $4 = $4 – $6
I7:   ADDI   $1, $1, 4     # $1 = $1 + 4
I8:   ADDI   $2, $2, 4     # $2 = $2 + 4
I9:   ADDI   $3, $3, -1    # $3 = $3 – 1
I10:  BNE    $3, $0, Loop  #if ($3 != 0) goto Loop
```

1) Show the timing of one loop iteration on the 5-stage MIPS pipeline **without forwarding hardware**. Complete the timing table, shown as **Table Q2.1** in the next page, showing all the stall cycles. Assume that the branch will stall the pipeline for **1 clock cycle only**.

2) According to the timing diagram of **Table Q2.1**, compute the number of clock cycles and the average CPI to execute ALL the iterations of the above loop.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
I1: ADDI	**F**	**D**	**X**	**M**	**W**																				
I2: ADD																									
I3: LW																									
I4: ADD																									
I5: LW																									
I6: SUB																									
I7:ADDI																									
I8:ADDI																									
I9:ADDI																									
I10:BNE																									
I3: LW																									
I4: ADD																									

Table Q2.1 Timing of one loop iteration on the 5-stage MIPS pipeline **without forwarding hardware.** (F: instruction fetch, D: instruction decode, X: execute, M: memory access, W: write back.) We assume that the register write is done in the first half of the clock cycle and that register reads are done in the second half of the cycle.

3. CACHE (10 points):

A 16-byte cache has 8-byte blocks, has 2 sets, and is 2-way set-associative. The cache initially is empty (all valid bits are off: indicated by a blank box in the table below). The cache receives requests in the sequence listed in **Table Q3.1**. For each address in the sequence (a) split it into the tag, index, and offset; (b) categorize the access as a hit, a compulsory miss, a confict miss, or a capacity miss (You can abbreviate hit=H, Compulsory=O, Confict=F, Capacity=P); (c) show the new contents of the cache after the access----write the tags for each way, and note which way is LRU.

The first one is done for you.

Address	Split Address			Result	Set 0			Set 1		
	Tag	Index	Offset		Way 0	Way 1	LRU	Way 0	Way 1	LRU
FF	F	1	7	O			0	F		1
F7										
0B										
24										
FB										
CA										
08										
F8										
35										
F7										
32										

Table Q3.1

4. VIRTUAL MEMORY (9 points):

Describe the number of bits required in each entry of a TLB that has the following characteristics:

· Virtual addresses are 32 bits wide
· Physical addresses are 31 bits wide
· The page size is 2K bytes
· The TLB contains 16 entries of the page table
· The TLB is direct-mapped

Note:

Each entry of the TLB contains the following items:

- a valid bit
- the physical page number of the desired virtual page
- a tag used to see if the desired entry of the page table is stored in the TLB
- ignore the “dirty” bit

1) Physical page number:

2) Tag:

3) The number of bits required in each entry of a TLB:

5. CHOICE (60 points) (note: only one is correct):

(1) Which of the following descriptions is the reason why binary expression is still used in computer technique. ( )

A: It saves components.

B: It has fast computing ability.

C: It is decided by the physical property of components.

D: It has nice convenience in coping with information.

(2) Which one is not one of the five classic components of a computer? ( )

A: Input

B: Bus

C: Memory

D: Output

(3) What is the range of exponent of IEEE 745 single precision? ( )

A: 1~254

B: -128~126

C: -126 ~127

D:-127~128

(4) Assume a test program A is running on computer A. It consumes 100 seconds totally, 90 for CPU and 10 for I/O. Now, the CPU speed improved by 50% and I/O speed hasn't changed. How much time it takes to run program A now? ( )

A: 55 sec

B: 60 sec

C: 65 sec

D: 70 sec

(5) A simple program is running on a 32-bit computer. x(int), y(short), z are variables in this program. If x = 127 and y = -9, what are the contents in computer memory after executing the assigning statement z = x + y? ( )

A: x = 0000007FH, y = FFF9H, z = 0076H

B: x = 0000007FH, y = FFF9H, z = 00000076H

C: x = 0000007FH, y = FFF7H, z = 0076H

D: x = 0000007FH, y = FFF7H, z = 00000076H

(6) Which of the following instructions could this single-cycle datapath description be referring to? ( )

Description :Two source registers (ReadReg) and one destination register (WriteReg) are selected, and the values are read from the source registers and sent as input to the ALU. The ALU operation is performed, the result is written to the destination register, and the PC is updated.

A: add B: ori C: li D: sll

(7) The following commands were used to store the contents of registers $s0 and $s1 onto the stack:

```
addi $sp, $sp, -8
sw $s0, 0($sp)
sw $s1, 4($sp)
# insert various unrelated instructions here
```

Assuming that neither the stack pointer nor the stack has been changed during the "various unrelated instructions" part, which of the following would allow you to recover the contents of $s0 and $s1 while returning $sp to its original (pre-decremented) value? ( )

A: addi $sp, $sp, 8; lw $s0, 4($sp); lw $s1, 0($sp)
B: addi $sp, $sp, 8; lw $s0, 0($sp); lw $s1, 4($sp)
C: lw $s0, 4($sp); lw $s1, 0($sp); addi $sp, $sp, 8
D: lw $s0, 0($sp); lw $s1, 4($sp); addi $sp, $sp, 8

(8) How the cache conflict misses will be affected by the following modifications? ( )
Assume the baseline cache is set associative.
(a). Double the associativity while keep the capacity and line size constant
(b). Double the number of sets while keep the capacity and line size constant

A: Decrease; Increase
B: Increase; Decrease
C: Increase; Increase
D: Decrease; Decrease

(9) Which of the following statements about multiplication and division is incorrect? ( )
A: Integer multiplications takes an input two 32-bit values and returns a 64-bit value
B: The result of a multiplication is stored in a read-only (for the programmer at least) "product" register
C: The product of two numbers is accesssed using two separate instructions - mfhi to get bits 0-31 (the rightmost bits), and mflo to get bits 32-63
D: The div command stores the quotient and the remainder in the product register, and the two can be accessed using mfhi and mflo

(10) Which of the following is generally true about a design with two levels of caches? ( )
A: First-level caches are more concerned about hit time, and second-level caches are more concerned about miss rate.
B: First-level caches are more concerned about miss rate, and second-level caches are more concerned about hit time.
C: Second-level caches often use lower associativity than first-level caches given the focus of reducing miss rates.
D: Second-level caches are as fast as first-level caches.

(11) The communication between central system and the outside environment is done by ( )

A: Input-output subsystem

B: Control system

C: Memory system

D: Logic system

(12) Which of the following statements about flash memory is wrong? ( )

A: The information can be either read or written, and the read speed is the same as write speed.

B: The storage unit is consist of MOSFET, so it is a semiconductor storage.

C: Information will not lose after power down.

D: It can be a replacement for the external memory.

(13) Which of the following situation will not happen? ( )

A: TLB miss, Cache miss, Page miss

B: TLB miss, Cache hit, Page hit

C: TLB hit, Cache hit, Page hit

D: TLB hit, Cache hit, Page miss

(14) The techniques which move the program blocks to or from the physical memory is called as _____. ( )

A: Paging B: Virtual memory organisation C: Overlays D: Framing

(15) The method of synchronising the processor with the I/O device in which the device sends a signal when it is ready is_____. ( )

A: Exceptions B: Signal handling C: Interrupts D: DMA

(16) Let's say we have an array with 4 integer elements. The address of the first element in the array is stored in $t0. Which of the following gives us the result of the last element of the array (stored in $t1)? ( )

A: addi $t1, $t0, 3; sll $t1, 2;

B: addi $t1, $t0, 16

C: sll $t0, 2; addi $t1, $t0, 3

D: addi $t1, $t0, 12

(17) How many total bits are required for a direct-mapped cache with 16KB of data and 4-word blocks, assuming a 32-bit address? Take valid bit into consideration. ( )

A: 146Kbits B: 147Kbits C: 148Kbits D: 149Kbits

(18) Consider a virtual memory system with 32-bit virtual byte address, 4KB/page, 32 bits each entry. The physical memory is 512MB. Then, the total size of page table needs. ( )

A: 1MB B: about 3MB C: 4MB D: 8MB

(19) What is the average time to read or write a 512-byte sector for a typical disk rotating at 10,000 RPM? The advertised average seek time is 6 ms, the transfer rate is 50 MB/sec, and the controller overhead is 0.2 ms. Assume that the disk is idle so that there is no waiting time. ( )

A: 6.0ms B: 9.0ms C: 9.01ms D: 9.21ms

(20) In cache, the replacement strategy includes RAND, FIFO and LRU. Which of these strategies is relative to the locality principle? ( )

A: RAND B: FIFO C: LRU D: NONE

6. CACHE DESIGN (10 points) (Additional Questions)

To improve the hit rate for our data cache, we made it 2-way set associative (it was formerly direct mapped). Sadly as a consequence the hit time has gone up, and we are going to use way-prediction to improve it. Each cache set will have a way prediction indicating which way is likely to be accessed.

When doing a cache access, the prediction is used to route the data. If it is incorrect, there will be a delay as the correct way is used. If the desired data is not resident in the cache, it is like a normal cache miss. After a cache miss, the prediction is not used since the correct block is already known. Figure Q6.1-A summarizes this process.

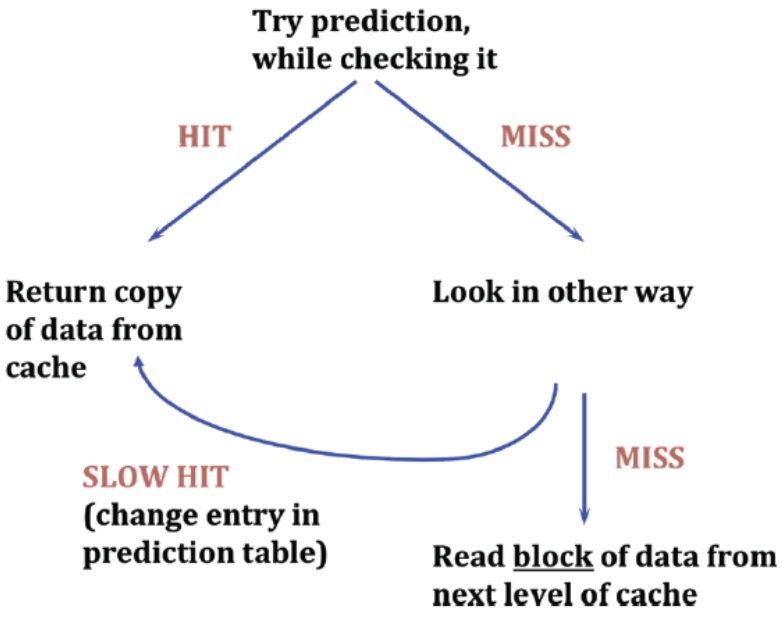

Figure Q6.1-A: Way-prediction FSM

Since there are two ways, only one bit will be used per prediction, and its value will directly correspond to the way. How the predictions are generated or maintained are beyond the scope of this problem. You can assume that at the beginning of a cycle, the selected prediction is available, and determining the prediction is not on the critical path. The diagram of the data portion of our cache is shown in Figure Q.6.1-B.

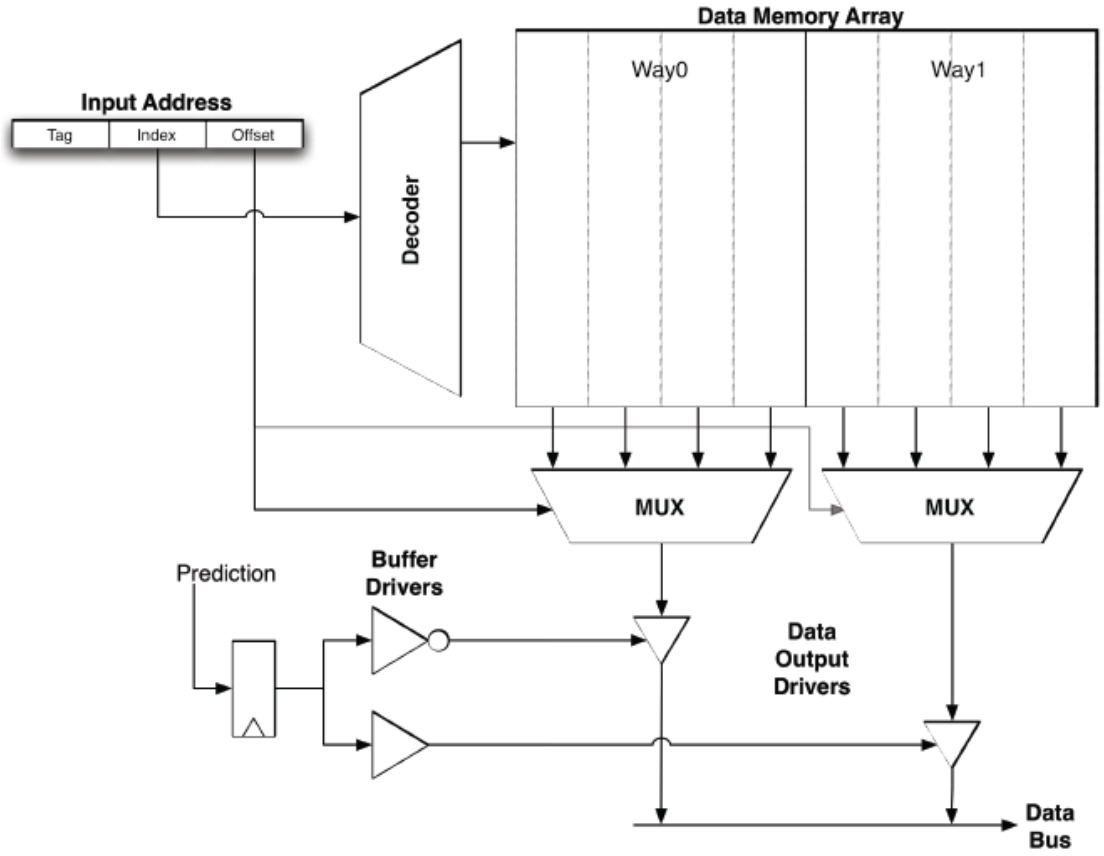

Figure Q6.1-B Data portion of the cache

Our cache has **16 byte** lines, is **2-way** set associative, and has a total capacity of **4kB**.

A. Please complete Table Q6.1 with delays across each element of the cache. Using the data you compute in Table Q6.1, calculate the critical path delay through this cache (from when the Input Address is set to when the correct data is on the Data Bus).

Component	**Dealy equation (ns)**	**Total (ns)**
Decoder	0.2x(# of index bits) + 1	
Memory Array	$0.2x\log_2$(#of rows) + $0.2x\log_2$(@bits in a row) + 1	
N-to-1 MUX	$0.5x\log_2 N + 1$	
Buffer driver	2	
Data output driver	0.5 x (associativtiy) + 1	
Critical Path Dealy		

Table Q6.1

You may assume that the prediction register is correctly loaded at the start of the cycle, and the clk-to-q delay is 100ps. The inverting and non-inverting buffer drivers both have the same delay. You only need to worry about the case of a fast hit (cache hit with correct prediction).

B. Now we will study the impact of way prediction on cache hit rate. For this problem, the cache is a **128 byte**, **2-way** set associative cache with **16 bytes** per cache line. The cache is byte addressable and uses a least recently used (LRU) replacement policy.

Please complete Table Q6.2 showing a trace of memory accesses. In the table, each entry contains the {tag, index} contents of that line, or "-", if no data is present. You should only fill in elements in the table when a value changes. For simplicity, the addresses are only 8 bits.

7	6	5	4	3	2	1	0
TAG		**INDEX**		**WORD SELECT**		**BYTE SELECT**	

The first 3 lines of the table have been filled in for you. The initial values marked with a '*' are the least recently used ways in that set. For your convenience, the address breakdown for access to the main cache is depicted below.

Read Addresss	Set0		Set1		Set2		Set3		Cache Hit?	Way-Prediction	
	Way0	Way1	Way0	Way1	Way0	Way1	Way0	Way1		Prediction	Correct?
	0	**-***	**9**	**-***	**A***	**6**	**7**	**B***			
04									**Y**	**0**	Y
68									**Y**	**1**	Y
2C					**2**				**N**	**1**	-
B4										**0**	
54										**0**	
3C										**0**	
94										**1**	
28										**0**	
64										**0**	
80										**0**	
64										**1**	
B4										**1**	

Table Q6.2