0907432 Computer Design (Summer 2015)

Midterm Exam

الاسم: الرقم التسلسلي: رقم الشعبة: 1

==========

Instructions: Time **50** minutes. Open book and notes exam. No electronics. Please answer all problems in the space provided and limit your answer to the space provided. **No questions are allowed**.

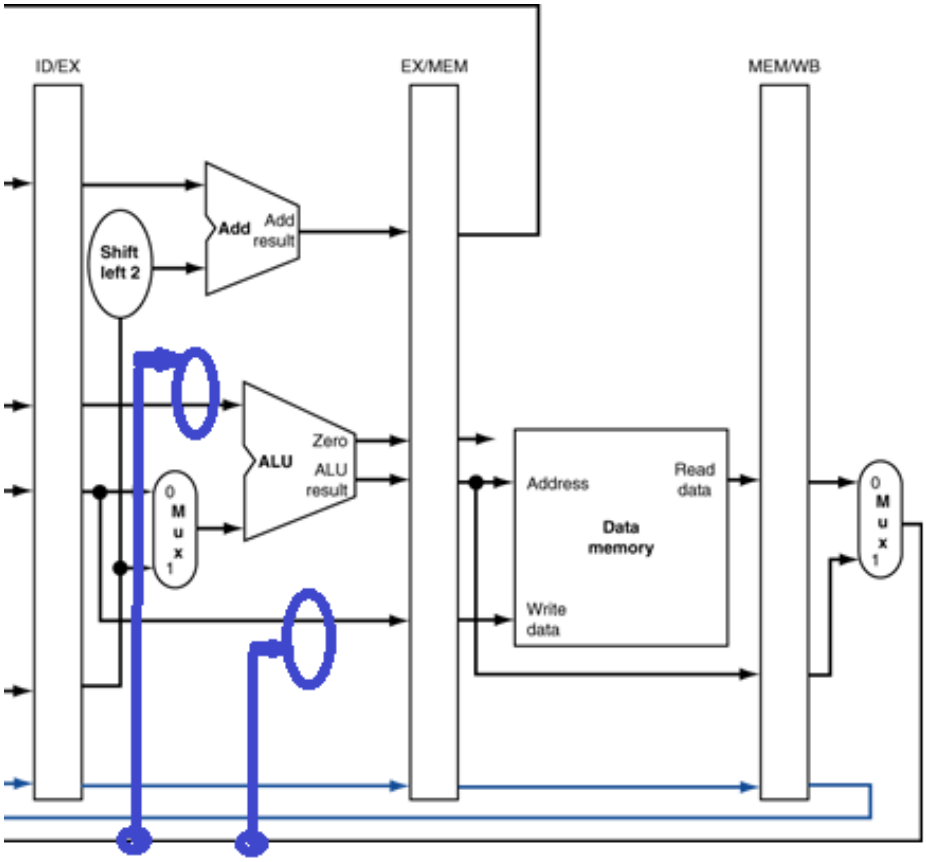
<Good Luck>

Q1. Assume that the classical 5-stage MIPS pipeline processor studied in the class uses forwarding and stalls to solve data hazards. Given the following code sequence, answer the following three questions.

A) Using the multicycle diagram below, these four instructions will be executed in ___**9**____ cycles. *[5 marks]*

	1	2	3	4	5	6	7	8	9	10	11	12
`lw r2,0(r1)`	**F**	**D**	**E**	**M**	**W**							
`sw r2,0(r2)`		**F**	**D**	**D**	**E**	**M**	**W**					
`add r4,r3,r2`			**F**	**F**	**D**	**E**	**M**	**W**				
`addi r1,r1,4`					**F**	**D**	**E**	**M**	**W**			

B) On the following diagram, draw the forwarding paths needed to solve the data hazard between the `lw` and `sw` instructions of the above code sequence. Show needed multiplexers and data buses. *[5 marks]*



C) Schedule these four instructions in order to speed up their execution on this processor.

[3 marks]

```
lw   r2,0(r1)
addi r1,r1,4
sw   r2,0(r2)
add  r4,r3,r2
```

Q2. Assume that the following code sequence is executed by a dual-issue speculative pipelined processor. This processor uses reservation stations, common data buses, and reorder buffer. The fetch stage takes one cycle and the issue stage takes one cycle. The integer latency is 1 cycle, the branch latency is 1 cycle, and the load latency is 2 cycles (1 cycle for address calculation and 1 cycle for data memory access). The processor has one address calculation unit, one memory access unit, one integer ALU unit, and one branch unit. Assume that the `beq` instruction is predicted not taken and is actually not taken.

A) Using the multi-cycle pipeline diagram below, specify the execution of these instructions in this processor pipeline.

[6 marks]

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
`lw r4,0(r1)`	**F**	**I**	**A**	**M**	**W**	**C**									
`lw r5,0(r2)`	**F**	**I**		**A**	**M**	**W**	**C**								
`add r6,r4,r5`		**F**	**I**				**E**	**W**	**C**						
`beq r4,r7,L1`		**F**	**I**			**E**	**W**		**C**						
`sw r6,0(r1)`			**F**	**I**	**A**					**C**					
`sub r6,r4,r5`			**F**	**I**				**E**	**W**	**C**					
`sw r5,8(r1)`				**F**	**I**	**A**					**C**				
`sw r6,12(r1)`				**F**	**I**		**A**				**C**				

B) Register `r4` is updated in Cycle __**6**__

[2 mark]

C) The data operand of the reservation station that has the last `sw` is updated in Cycle __**9**__

[2 mark]

Q3. Assume that you have a dual inline memory module (DIMM) that uses DDR3-SDRAM chips operating at 1,000 MHz clock.

A) What is the peak bandwidth if the DIMM's data bus width is 64 bits?

[3 marks]

Peak BW = 64 bits/transfer × (1 byte/8 bits) × (2 transfers/cycle) × 1000 Mega Cycles/sec
= (64/8) × 2 × 1000 Mega Bytes/sec
= 16,000 MB/sec

B) Assume that it takes 25 nsec for the DIMM to start sending the data from the start of the access request. What is the total time needed to access and transfer a 64-Byte memory block?

[4 marks]

Total time = Access time + Transfer time
= 25 + 64 bytes/block × (1 transfer/8bytes) × (1 cycle/ 2 transfers) × (1 sec/1000 Mcycles)
= 25 + (64/8) /2 nsec
= 29 nsec