CS/ECE 552: Introduction to Computer Architecture
Department of Computer Sciences
University of Wisconsin-Madison

Prof. Karthikeyan Sankaralingam

Mid-term Examination
In-Class
March 14, 2013
Approximate Weight: 25%

1 hour 15 minutes

CLOSED BOOK. You can bring one-cheat sheet (8.5 x 11 page).

Exam is **one-sided, total of 9 numbered pages**. Plan your time carefully.
One blank page included in the end for rough work. Also use left-side blank pages for rough work.

NAME: _____

Email: _____

Problem number	Maximum points	Actual points
1	10	
2	10	
3	10	
4	10	
5	21	
6	20	
7	25	
Total	**106**	

Problem 1 (10 points)

Consider a machine which has the following CPI for three categories of instructions:

Arithmetic: 1 cycles **Loads/stores:10 cycles** **Branch: 3 cycles**

This machine operates at 4 GHz. Consider a program with the following instruction count for each category of instructions:

Arithmetic: 1 billion **Loads/stores:200 million** **Branch: 100 million**

Part a) What is the execution time of the program?

Part b) In a modification to this machine, the CPI of loads/stores is reduced to 2 cycles, and as a result the frequency is also reduced to 2 GHz. What is the new execution time?

Problem 2 (10 points)
Two companies A and B both build machines and their associated software for executing a challenge application for DNA sequencing analysis. Performance is measured in terms of execution time to match one DNA sample to a set of samples in a database. A's final system is 75% faster than B's final system when both machines are provided identical inputs. Explain five possible ways in which A is different allowing it to provide higher performance and for each explain impact on Iron Law of Performance: Execution time = Number of instructions * CPI * 1/clock-frequency

Reason 1

Reason 2

Reason 3

Reason 4

Reason 5

Problem 3 (10 points)

Write MIPS code for the following two lines of C code. Assume variable x must be stored in register $t0, and variable y must be in register $t1

```
x = 0;
y = x + 0x0123ABCD;
```

Problem 4 (10 points)
Assume we make an enhancement to a computer so that some mode of execution is improved by a factor of 5.

Under what circumstances will overall program speedup be 3?

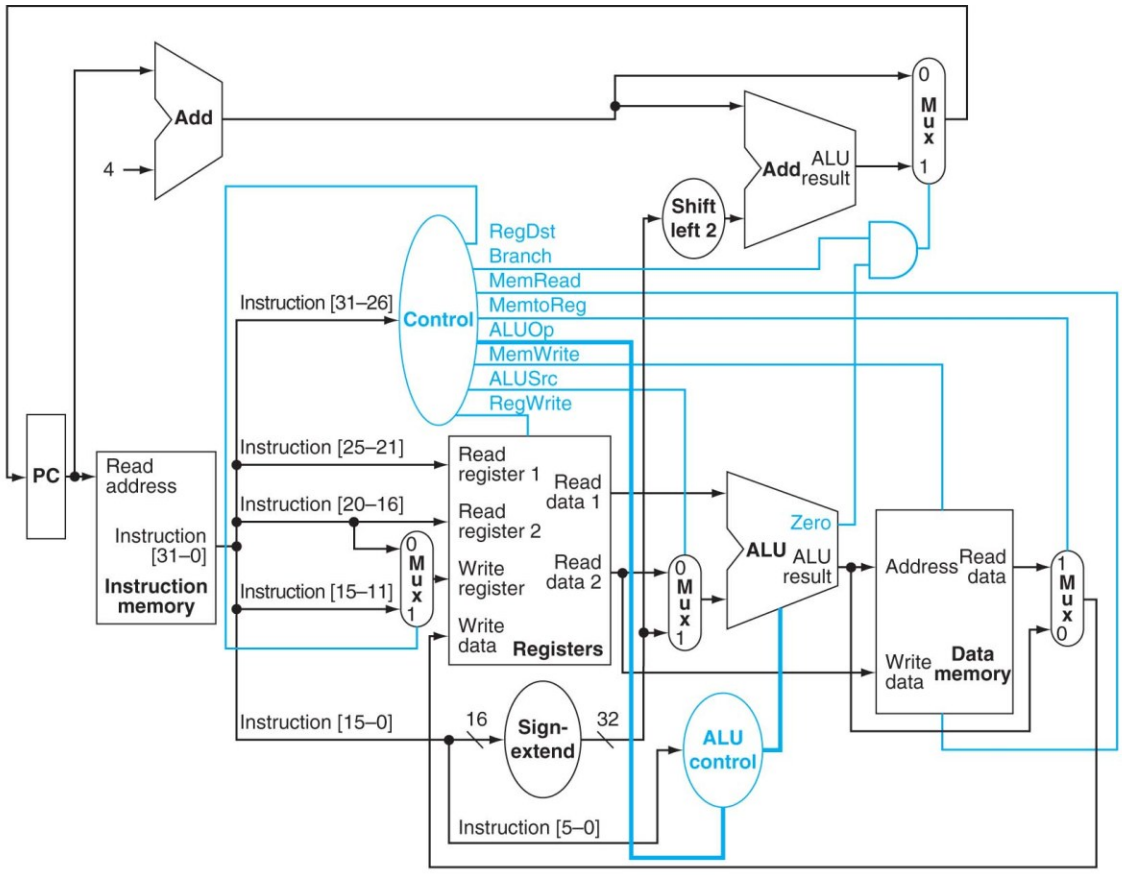
Bonus (5 points)
If the enhanced mode contributes to 50% of the time (and this 50% is percentage of the total execution time when the enhanced mode is used), what is the overall speedup? What percentage of the original execution time is converted to the faster mode?

Problem 5 (21 points)

Consider the single cycle datapath we have discussed in class and shown in the textbook. The figure is shown below. For the following 3 instructions complete the values for all the control signals **excluding ALUOp**. Recall that $t0 to $t7 in MIPS assembly are mapped to register numbers 8 to 15. In the code below $0 means directly register 0 to avoid this translation for you.

Complete this table

	RegDst	Branch	MemRead	MemtoReg	MemWrite	ALUSrc	RegWrite
add $3,$1, $2							
lw $1, $2 (0x12)							
beq $0, $2, 28							



Problem 6 (20 points)

Assume a datapath with forwarding and hazard detection and NOT-taken predictor. The corresponding datapath from the textbook is shown below. Draw pipeline diagram for code sequence.

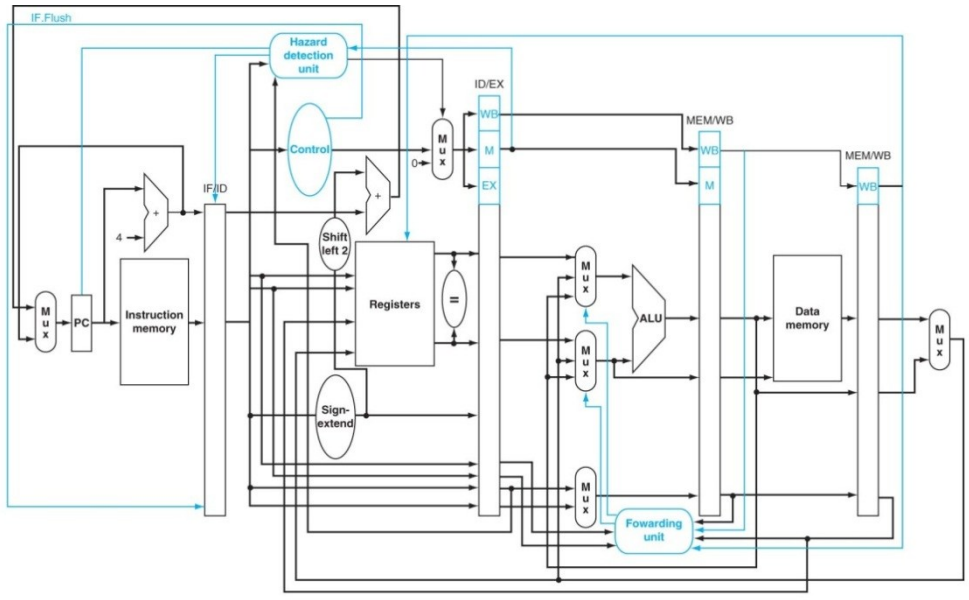
```
beq $t0, $t1, 1
add $t3, $t3, $t3
beq $t0, $t1, 1
add $t3, $t3, $t3
beq $t0, $t1, 1
add $t3, $t3, $t3
```

Part a: Complete this table. **Assume $t0 = 0x12; and $t1 = 0x12**

----------	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14
	F	**D**	**E**	**M**	**W**									

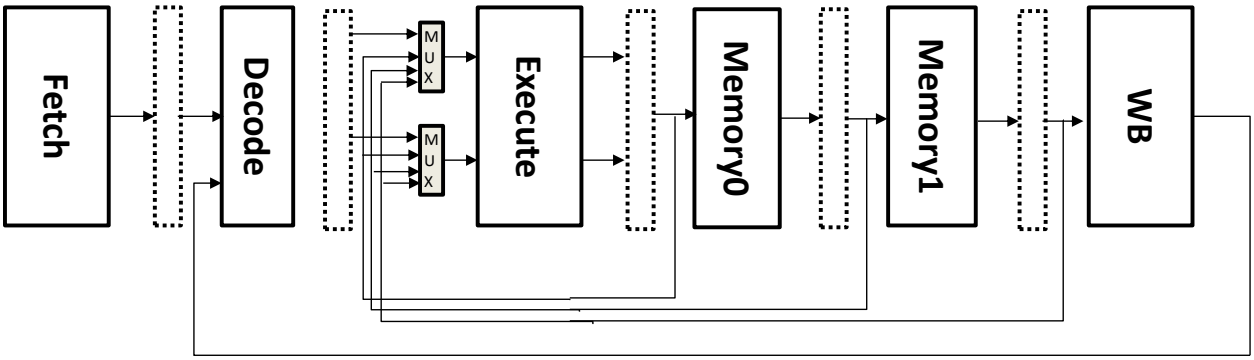
Part b: Complete this table. **Assume $t0 = 0x12; and $t1 = 0x10**

----------	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14
	F	**D**	**E**	**M**	**W**									



Problem 7 (25 points)

Consider a new datapath below. The only difference between this and the datapath we discussed in class is that the memory stage is two cycles. Notice that we have implemented data-forwarding from both of those stages. For load instructions the value is available only at the end of the second memory stage. Complete the pipeline diagram for the code sequence below.



	Cycle																						
Instruction	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**
add $t1, $t2, $t3	F	D	E	M0	M1	W																	
lw $t6, 4($t1)																							
sw $t6, 4($t1)																							
lw $t1, 16($t6)																							
add $t5, $t3, $t1																							

For each cycle where a stall occurs explain why. Answers without explanation will get ZERO.

Blank page