Name: KEY **Id#**

COE 301/ICS 233, Term 151

Computer Architecture & Assembly Language

Quiz# 5

Date: Thursday, Dec. 10, 2015

Q1.

(i) Given the following instruction mix of a program on a RISC processor:

Class	CPI	Frequency
ALU	2	40%
Branch	2	25%
Jump	1	15%
Load	4	10%
Store	3	10%

What is the average CPI?

Average CPI=2*0.4 + 2*0.25+1*0.15+4*0.10+3*0.10=2.15

(i) Suppose that a program runs in 150 seconds on a machine, with ALU operations responsible for 40 seconds of this time, multiply operations responsible for 50 seconds of this time and divide operations responsible for 40 seconds of this time. The remaining time is taken by the remaining operations. Suppose that a new implementation of the machine has improved the execution time of the ALU by a factor of 2, the multiplier by a factor of 1.5 and the divider by a factor of 1.6. Determine the new execution time and the speedup of the program based on the new implementation.

Execution time of new implementation = 40/2 + 50/1.5 + 40/1.6 + 20
=20+33.33+25+20=98.33 seconds

Speedup = 150/98.33 = 1.525

Q2. Consider the code given below:

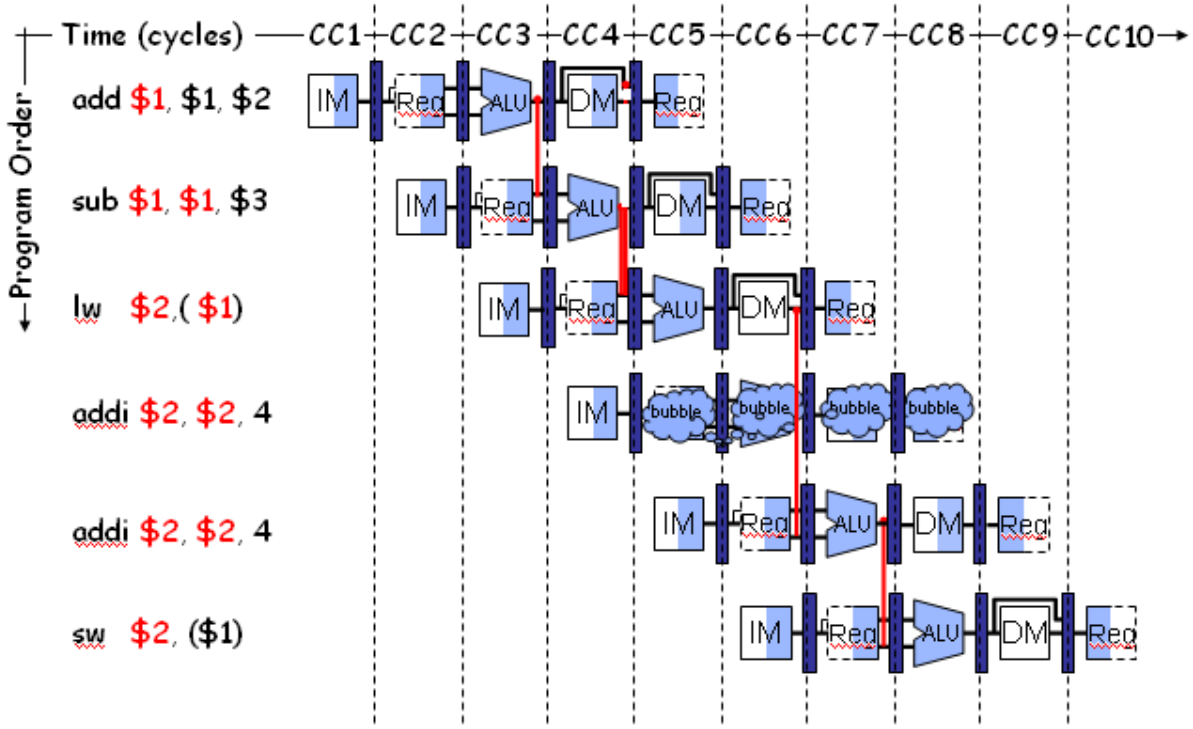
```
add  $1, $1, $2
sub  $1, $1, $3
lw   $2, ($1)
addi $2, $2, 4
sw   $2, ($1)
```

(i) Identify all the **RAW** data dependencies in the above code. Which dependencies are data hazards that will be resolved by forwarding? Which dependencies are data hazards that will cause a stall?

RAW Dependencies:

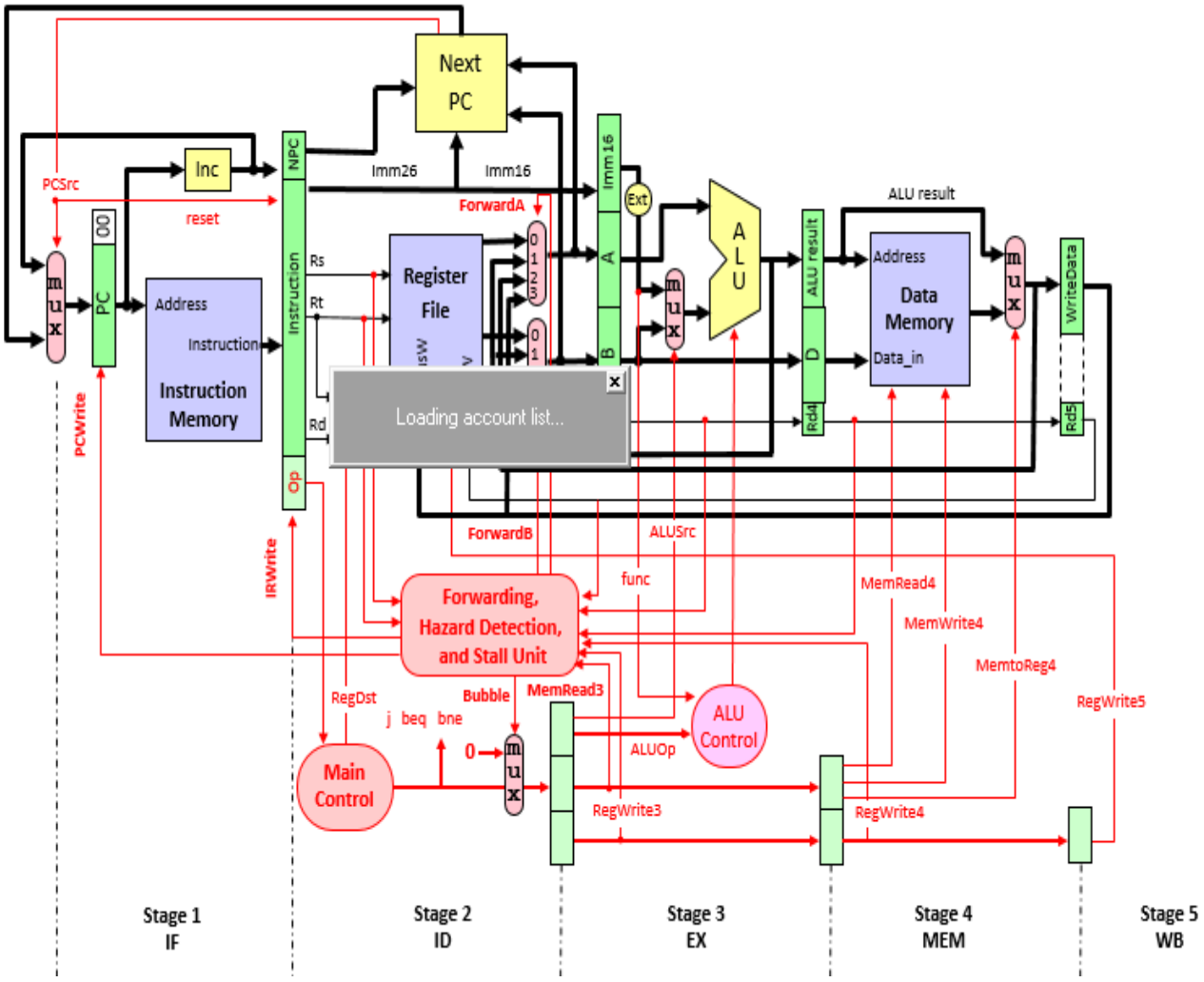
add $1, $1, $2	**and**	**sub $1, $1, $3**	**(forwarding)**
sub $1, $1, $3	**and**	**lw $2, ($1)**	**(forwarding)**
lw $2, ($1)	**and**	**addi $2, $2, 4**	**(stall 1 cycle & forwarding)**
addi $2, $2, 4	**and**	**sw $2, ($1)**	**(forwarding)**

(ii) Using a multiple-clock-cycle graphical representation, show the instruction execution across the pipeline including forwarding paths and stalled cycles if any. How many clock cycles will be needed to execute the instructions?



Q3.

(i) Consider the 5-stage pipelined CPU design given below.



a) Show the control signals that will be used for stalling the pipeline due to data and control hazards along with their conditions.

b) Add the necessary changes to the design, on the given diagram, to allow it to handle data hazards due to load instructions and control hazards.

1. Stalling the Pipeline due to Load Instruction:

if ((MemRead3 == 1) // Detect Load in EX stage
and (ForwardA==1 or ForwardB==1)) Stall // RAW Hazard

OR:

if ((MemRead3 == 1)
and (Rd3 ≠ 0) and ((Rs == Rd3) or (Rt == Rd3))) Stall

Stall means that the signals PCWrite=0 and IRWrite=0, which will freeze the content of PC and IR registers and bubble=1 which will introduce a bubble in stage 2 control register by setting the control signals to 0.

2. Stalling the Pipeline due to taken branch Instruction:

Also, when PCSrc=1, reset=1 and the content of IR register will be reset to 0 to make the fetched instruction a NOP.