Computer Science 146/246
Homework #2

Due 11:59 P.M. Thursday, February 16th, 2023

Suggested Readings Computer Architecture: A Quantitative Approach, *Fifth* Edition, Chapter 3.1-3.8

1 Scoreboarding

Question 1: WAR Hazards and Scoreboarding [10 Points]

There is interesting documentation about the classic CDC6600 here.

a. Read the section called **RELEASE** on Page 134 (of the book not the pdf), which explains how the CDC6600 handles WAR hazards. Explain what the problem is with WAR hazards and how the CDC6600 solves it. [5 pts]

b. For architectures that use scoreboarding, is there another way to detect WAR hazards? If so, propose your own solution, explain why you think it will solve the problem, and compare the hardware resource usage (better or worse) between your solution and the CDC6600 approach; if not, explain why. (Do not propose register renaming solutions.) [5 pts]

2 Tomasulo

Consider the following code fragment:

```
LOOP:   LD      F0, 0(R1)
        DIVD    F2, F0, F6
        LD      F6, 8(R1)
        DIVD    F6, F6, F2
        SD      F6, 16(R1)
        DADDI   R1, R1, #-32
        BNEQZ   R1, LOOP
```

For this problem, consider the following architecture specifications:

Functional Unit Type	Cycles in EX	Number of Functional Units
Integer	1	1
FP Divider	15	1

A. Assume you have unlimited reservation stations.

B. Memory accesses use the integer functional unit to perform effective address calculation during the EX stage. For stores, memory is accessed during the EX stage (Tomasulo's algorithm without speculation) or commit stage (Tomasulo's algorithm with speculation). All loads access memory during EX stage.

C. Loads and stores stay in EX for 2 cycles.

D. Functional units are not pipelined.

E. If an instruction moves to its WB stage in cycle x, then an instruction that is waiting on the same functional unit (due to a structural hazard) can start executing in cycle x.

F. An instruction waiting for data on the CDB can move to its EX stage in the cycle after the CDB broadcast.

G. Only one instruction can write to the CDB in one clock cycle. Branches and stores do not need the CDB.

H. Whenever there is a conflict for a functional unit or the CDB, assume that the oldest (by program order) of the conflicting instructions gets access, while others are stalled.

I. Assume that BNEQZ occupies the integer functional unit for its computation and spends one cycle in EX.

J. Assume that the result from the integer functional unit is also broadcast on the CDB and forwarded to dependent instructions through the CDB (just like any floating point instruction).

Question 2: Tomasulo without Speculation [20 Points]

Complete the pipeline summary table using Tomasulo's algorithm but without assuming any hardware speculation on branches. That is, an instruction after a branch cannot issue until the cycle after the branch completes its EX. Assume a single-issue machine. Fill in the cycle numbers in each pipeline stage for each instruction in the first two iterations of the loop.

Show your complete work using a cycle by cycle breakdown PowerPoint presentation, similar to what was shown in lecture using a table for Instruction Status, Reservation Stations, and Register Result Status - **use the template provided on canvas**. Cycles 1 - 4 are provided as an example.

	Iteration 1			Iteration 2		
Instruction	Issue	EX	WB	Issue	EX	WB
`LP:LD F0, 0(R1)`	1	2-3	4			
`DIVD F2, F0, F6`	2					
`LD F6, 8(R1)`	3	4-				
`DIVD F6, F6, F2`	4					
`SD F6, 16(R1)`						
`DADDI R1, R1, #-32`						
`BNEQZ R1, LP`						

Based on your findings to this question, what is the best way to speed up the performance of this loop? Answer why or why not for the following options:

A. Add additional hardware resources such as more FP Dividers or FP Adders.

B. Pipeline the FP Dividers so multiple instructions can execute in the pipeline simultaneously.

C. Improve the performance of the FP Divider so it takes fewer cycles.

3 Submission

Please include solutions to questions 1 and 2 in a single PDF. Only include your completed Instruction Status table for questions 2 in the pdf (so I don't need to go hunting for your answers).

In a separate document, include your cycle-by-cycle PowerPoint showing your complete work. Please write down any assumptions you make and any other notes you'd like me to read justifying your solution / points of confusion.

Please upload the pdf and powerpoint(s) to canvas in a .zip file. Title the submission using the following format `cs146_246_hw2_[first]_[last].zip`

Updated February 10, 2023, Matt Adiletta