ECE 6913 INET and INET2 Sections, Quiz 2 Instructions

Computing Systems Architecture

Fall 2021 NYU ECE

Please fill in

Your Name:_____

netID: _____

Quiz 2, November 18th 2021

Maximum time: 150 minutes : **11:00 AM - 1:30 PM** [+ 10 minutes to assemble PDF and upload]

Open Book, Open Notes,

Calculators allowed.

Quiz 2 will be visible as an assignment on NYU Brightspace at 11:00 AM on November 18th 2021

You Must show your work in steps – to get any credit

This is NOT a group project Please DO NOT discuss, share your Quiz solutions with anyone else.

Please stay logged in to Zoom throughout the Quiz – please test your computer/camera before the quiz. If you are not logged in and identifiable, your quiz will not be graded.

Instructor/TAs available online if you have questions on the Quiz, during the Quiz – Please enter question in Zoom chat box at any time during Quiz

This Test has 6 problems. Please attempt all of them. Please show **all work**. Please write **legibly**

1. Please be sure to have 10-15 sheets of white or ruled Paper, a Pen/Pencil & Eraser
2. Please write down your solutions on 8.5 x 11 sheets of white paper, single-sided with your name printed in top right corner of each sheet and with Page Number and Problem number identified clearly on each sheet
3. **Please stop working on your Quiz at 1:30 PM** – you have 10 minutes to scan/take pictures of each sheet and upload them as completed PDF assignment to NYU Classes **by 1:40 PM** – you may use any of several smartphone apps to integrate your scans/pictures of sheets into a PDF file.
4. Please take pictures of each sheet and **upload** the PDF of all sheets after checking you have all sheets in the right order **by 1:40 PM latest.**
5. You may use iPAD to write down your solutions directly rather than on paper
6. Portal **will close at 1:45 PM** not allowing upload of your quiz after 1:45 PM

1. In the conventional fully bypassed 5-stage pipeline discussed in class, we were able to make the assumption that the ALU is able to complete in one cycle because we assumed integer operations. For this problem, we will assume that the ALU takes two cycles to complete. In other words, *the ALU is pipelined itself, making the entire pipeline 6 cycles*. The ALU only *generates a result after 2 cycles* (i.e., there is no way to extract any meaningful result after the ALU's first cycle). Memory (instruction and data) still returns data after only one cycle. You may ignore branch and jump instructions in this problem.

As a first step, we will examine how this changes data hazards. For this question, we will examine only the ALU-ALU read after write (RAW) hazard where the two instructions are consecutive:

```
ADD x1, x2, x3          # x1 <= x2 + x3

ADD x5, x4, x1          # x5 <= x4 + x1
```

1.1 Describe *how would you resolve this hazard with the minimum number of bubbles (if any) using a combination of data forwarding and stalls (if necessary)*.

1.2 Then fill the following timing diagram to illustrate how the pipeline will behave, and show where data forwarding happens, if at all, by drawing an arrow between the two stages that participate in it.

	CC0	CC1	CC2	CC3	CC4	CC5	CC6	CC7	CC8
ADD x1, x2, x3									
ADD x5, x4, x1									

2. Consider the following RISC V Instruction sequence executing in a 5-stage pipeline:

```
or   x13, x12, x11
ld   x10, 0(x13)
ld   x11, 8(x13)
add  x12, x10, x11
subi x13, x12, 16
```

2.1 Identify all of the data hazards and their resolution with NOPs assuming no forwarding or hazard detection hardware is being used

2.2 If there is forwarding, for the first seven cycles during the execution of this code, *specify which signals are asserted in each cycle by hazard detection and forwarding units* in Figure below.

Mux control	Source	Explanation
ForwardA = 00	ID/EX	The first ALU operand comes from the register file.
ForwardA = 10	EX/MEM	The first ALU operand is forwarded from the prior ALU result.
ForwardA = 01	MEM/WB	The first ALU operand is forwarded from data memory or an earlier ALU result.
ForwardB = 00	ID/EX	The second ALU operand comes from the register file.
ForwardB = 10	EX/MEM	The second ALU operand is forwarded from the prior ALU result.
ForwardB = 01	MEM/WB	The second ALU operand is forwarded from data memory or an earlier ALU result.

3. You purchased an old IBM System whose model number is unknown so you do not have access to any manuals or spec sheets of the computer – except those listed below by a previous user:

1. 95% of all memory accesses are found in the cache.
2. Each cache block is two words, and the whole block is read on any miss.
3. The processor sends references to its cache at the rate of 109 words per second.
4. 25% of those references are writes.
5. Assume that the memory system can support 109 words per second, reads or writes.
6. The bus reads or writes a single word at a time (the memory system cannot read or write two words at once).
7. Assume at any one time, 30% of the blocks in the cache have been modified.
8. The cache uses write allocate on a write miss.

You are considering adding an IBM compatible peripheral to the system, and you want to know *how much of the memory system bandwidth is already used*. Calculate the percentage of memory system bandwidth used on the average in the two cases below. Be sure to state your assumptions.

3.1 The cache is Write-Through.

3.2 The cache is Write-Back.

4A. Given a 4-way set associative cache of total size 2MB that has a 26-bit cache address and blocks of size 8KB each, answer the following questions.

4A.1 How many bits in the "index" field of the cache address? Explain your answer

4A.2 How many bits in the "offset" field of the cache address? Explain your answer

4A.3 How many bits in the "Tag" field of the cache address?

4B. Mark whether the following modifications to cache parameters will cause each of the categories to increase, decrease, or whether the modification will have no effect. You can assume the baseline cache is set associative. Explain your reasoning. Assume that in each case the other cache parameters (number of sets, number of ways, number of bytes/line) and the rest of the machine design remain the same.

	compulsory misses	conflict misses	capacity misses
increasing number of sets			
increasing number of ways			
increasing number of bytes per line			

5. Indicate if the following modifications (A,B,C) will cause each of the three metrics (three rightmost columns) to *increase*, *decrease*, or have *no effect*. Explain your reasoning

Assume the initial machine is pipelined. Also assume that any modification is done in a way that preserves correctness and maintains efficiency, but that the rest of the machine remains unchanged.

		Instructions/Program	CPI (Cycles/Instruction)	Circuit complexity
A	Replace the 2 operand ALU with a 3 operand one and add 3 operand register-register instructions to the ISA (for example, `ADD rs1,rs2,rs3,rd`)			
B	Use the same ALU for instructions and for incrementing the PC by 4			
C	Increase the number of user registers from 32 to 64			

6. This problem explores energy efficiency and its relationship with performance. The parts of this problem assume the following energy consumption for activity in Instruction memory, Registers, and Data memory. You can assume that the other components of the datapath consume a negligible amount of energy. ("Register Read" and "Register Write" refer to the register file only.)

I-Mem	1 Register Read	Register Write	D-Mem Read	D-Mem Write
140pJ	70pJ	60pJ	140pJ	120pJ

Assume that components in the datapath have the following latencies. You can assume that the other components of the datapath have negligible latencies.

I-Mem	Control	Register Read or Write	ALU	D-Mem Read or Write
200 ps	150 ps	90 ps	90 ps	250 ps

6.1 How much energy is spent to execute an **addi** instruction in a single-cycle design and in the five-stage pipelined design

6.2 How much energy is spent to execute a **lw** instruction in a single-cycle design

6.3 How much energy is spent to execute a **beq** instruction in a single-cycle design