

## Assignment 2

**Due: 1:25pm, Wed Sept 14th, 2022**

Note: Make reasonable assumptions where necessary and clearly state them. Feel free to discuss problems with classmates, but the only written material that you may consult while writing your solutions are the textbook and lecture slides/videos. Solutions should be uploaded as a single pdf file on Canvas. **Show your solution steps** so you receive partial credit for incorrect answers and we know you have understood the material. Don't just show us the final answer.

Every homework has an automatic penalty-free 1.5 day extension to accommodate any covid/family-related disruptions. In other words, try to finish your homework by Wednesday 1:25pm to keep up with the lecture content, but if necessary, you may take until Thursday 11:59pm.

### 1. Pipelining (40 points)

An unpipelined processor takes 14 ns to work on one instruction. It then takes 0.2 ns to latch its results into latches. I was able to create a new pipelined processor by converting the circuits into 8 sequential pipeline stages. The stages have the following lengths: 1.6ns; 1.8ns; 1.4ns; 1.9ns; 2.1ns; 0.9ns; 1.7ns; 2.6ns. Answer the following, assuming that there are no stalls in the pipeline.

1. What are the cycle times in both processors?
2. What are the clock speeds in both processors?
3. What are the IPCs in both processors (averaged across millions of instructions and assuming no pipeline hazards)?
4. How long does it take to finish one instruction in both processors (in nano-seconds and cycles)?
5. What is the speedup provided by the 8-stage pipeline?
6. If I was able to build a magical 1000-stage pipeline, where each stage took an equal amount of time, what speedup would I get?

### 2. Instructions in the 5-Stage Pipeline (20 points)

1. What does a load instruction do in the ALU stage of the basic 5-stage pipeline?
2. Provide two example instructions that do not write to registers.
3. Provide an example instruction that writes to memory.
4. Provide an example instruction that does not use the ALU stage of the pipeline.
5. Provide an example instruction that does nothing in the DM stage of the pipeline.
6. Specify the number of input registers used by the following instructions: ADD, LD, ST.

### 3. Data Dependences (40 points)

Consider a basic 5-stage in-order pipeline similar to the one discussed in class. How many stall cycles are introduced between the following pairs of successive instructions (i) for a processor with no register bypassing and (ii) for a processor with full bypassing? Use pipeline diagrams for each case to show if/how the second instruction is stalled.

1. Int-add, followed by a dependent Int-add
2. Load, followed by a dependent Int-add
3. Load, providing the data for a Store
4. Int-add, providing the data for a Store