ECEC 355 – Computer Architecture

Project Two – Pipelining RISC-V Simulation with Data Forwarding

Instructor: Dr. Anup Das

Distributed, Intelligent, and Scalable COmputing (DISCO) Lab

ECE Department

Drexel University

June 24, 2019

1. Objective

This project is intended to be a comprehensive introduction to pipelining RISC-V simulation. Please submit your work by August 30th, 2019, at 11:59 pm, via Bblearn. You may work on this project in teams of up to two people.

2. Required Reading

Chapter 4, The Processor, Sections 4.5 – 4.7

3. What to do

3.1 Pipelining without hazard detection

Your first task is to divide the single-cycle core into five stages, represented by five new structures. Complete all the five stages and simulate cpu_traces/task_0 with initialization:

- x1 = 0; x2 = 10; x3 = -15; x4 = 20; x5 = 30; x6 = -35
- 40(x1) = -63, 48(x1) = 63

3.2 Pipelining with hazard detection

Integrate your pipeline simulator with a hazard detection unit to detect control hazard as well as data hazard. Conventionally, the zero signal is generated in the EX stage leading to two potential flushes for instructions following a conditional jump. Please add a comparator and modify corresponding stages to make decisions in the ID stage, with this modification, only one flush is needed.

- Simulate cpu_traces/task_1 with initialization:
  - x1 = 8; x3 = -4; x5 = 255; x6 = 1023

- Simulate cpu_traces/task_3 with initialization:
  - x1 = 0; x2 = -5; x5 = -10; x6 = 25
  - 100(x7) = -100
- Simulate cpu_traces/task_3 with initialization:
  - x1 = 8; x2 = -5; x5 = -10; x6 = 25
  - 100(x7) = -100

3.3 Pipelining with data forwarding

Integrate your pipeline simulator with a forwarding unit.

- Simulate cpu_traces/task_1 with initialization:
  - x1 = 8; x3 = -15; x5 = 255; x6 = 1023
- Simulate cpu_traces/task_2 with initialization:
  - x5 = 26; x6 = -27;
  - 20(x1) = 100

4. Submissions

Zip the followings and submit through Bblearn:

- Report on how you complete the five stages
- Report on how you implement the hazard detection unit
- Report on how you implement the forwarding unit
- Your source codes