Organization of Digital Computers Lab
EECS 112L

LAB 2

EECS Department
Henry Samueli School of Engineering
University of California, Irvine

Winter 2019

Due Date : Feb 8th at 4:00 pm

1 Introduction

The goal of this lab is to write a testbench to test your simplified processor. Last week, you tested RISC-V processor with given instructions. In this lab, you should write your own instruction set for all instructions. You are not supposed to add any new instructions to your processor in this session.

You have 2 weeks to finish this lab. Submit a zip file containing all your files to Canvas.

2 Instructions Binary File

1. make sure to pull new repository from GitHub.Then, cd to "Lab2" directory.

2. In design folder, open "instructionmemory.sv" file. As you can see, there is not any instruction in this file anymore.

3. In verif folder, there is another directory named "program". In program directory, there is file named "inst.bin". In this file, we should have the binary codes for instructions. We already copy-pasted the instruction set from last lab to this file.

4. You should modify your testbench to be able to copy instructions from "inst.bin" to "instructionmemory.sv" automatically. So, from now on, you just need to add/delete any instruction in "inst.bin". Add the following command to your testbench file:

$readmemb ("$verif/program/inst.bin" , riscv.Datapath.instr_mem.Inst_mem);

- **$readmemb** will read a binary file, and initialize memory array. If you want to write instructions in hexadecimal format, use $readmemh.
- **"$verif/program/inst.bin"** is the address to the binary file (.bin).
- **riscv.Datapath.instr_mem.Inst_mem** is the path to instruction memory register. In riscv.sv, Datapath is instantiated. In Datapath, instruction memory is instantiated, and Inst_mem is the name of the 512*32bits register.
- Now, you should simulate riscv module. It should give you the same result as what you got in lab1.
- Finally, write your own instruction set in "inst.bin", and make sure that your riscv, works correctly for all possible intructions.

3 Material to be submitted

Report (20% of your total grade):

- Reports written in LaTeX are preferred. Submit pdf file of your report.
- Block diagram of the design with the detailed explanation of how your processor works. Your Block diagram should contain all Your design modules.
- Put a screenshot from the simulated waveform. your screenshot is from the result of testbench simulation so it should contain Clk, reset, and ALU-Result.
- After synthesis, put the riscv.elab.rpt and riscv.qor.rpt files in the doc folder of your project.
- Put the synthesis results in your report. Critical path length, critical path slack, and area and power report.

Important notes:

- Submit the project in zip format.
- The name of uploaded zip file should be your name.
- The compressed file should include design, sim, syn, verif, and doc, directories.

- The report should contain your name and student ID (on the cover page).

- No points if you submit a wrong file by mistake. We **do not** accept any changes in code after the deadline.

- **NO Late submission is permitted.**