

COEN 122L - Spring 2019

By: Patterson Jaffurs (pjaffurs@scu.edu)

Professor: Dr. Moe Amouzgar

Assignment 3 - Register File, Data Memory, and Instruction Memory

Description

There are three memory modules that we need to construct for the CPU proper: the registers, the data memory, and the instruction memory. The register file holds the values of the registers in the system. The data memory loads and stores the data like a cache. The instruction memory contains the binary instructions and provides them to the pipeline for use.

Goal

In this lab, you will be creating the three memory modules you need to use in the final project and learning how to sync different processor components up to a clock. A clock alternates a signal from high to low to correctly sequence hardware operations. The high part of the signal is referred to as a *posedge* and the low a *negedge*. You will be using the clock's posedge for this lab to maintain simplicity.

- Register File: the register file has 64 32-bit registers and can read and write on the same clock cycle.
 - Inputs: clock, write signal, three 6-bit register addresses, 32-bit data in
 - Outputs: two 32-bit data out (\$rs & \$rt)
- Data Memory: our data memory will be using 16-bit addressing so as to not crash Vivado due to simulating a large amount of 32-bit addresses. So we limit the address range to 0-65535. Thus, to access 16 bits of the 32-bit input, you use 'address[15:0]' in Verilog. Read and write are exclusive operations here.
 - Inputs: clock, write signal, read signal, 32-bit address, 32-bit data in
 - Outputs: 32-bit data out
- Instruction Memory: the instruction cache holds a number of 32-bit instructions and while there is no required number, 256 instructions are recommended. Instructions are fetched from here and passed forward to be decoded and executed.
 - Inputs: clock, 8-bit address
 - Outputs: 32-bit instruction

Only a single test-bench file is necessary for this lab, as all three modules can be simulated at one time. All modules should be connected to the same clock and execute in time with it.

Deliverable

To receive credit for the lab, you must demo your lab to me and also submit a .zip file on Camino containing: commented source code, commented test-bench code, and a screenshot of your waveform.

Please paste your code into .txt files and submit your .zip titled *firstname_lastname.zip*.

Grading is as follows:

- Demo 60%
- Submission 40%
- Due two weeks after lab, -10% late one week, no credit after two weeks