CECS 341 – LAB 4

MIPS Write Back

09 March 2020

Rifa Safeer Shah

017138353

I certify that this submission is my original work

Lab Report: Lab Assignment 4 – MIPS Write Back

1. Goal: The goal of the MIPS Write Back Lab Assignment is to use Structural Verilog to complete the writeback stage in the single cycle MIPS processor.

2. Steps:
    a. Go to edaplayground.com and log in to your account.
    b. Rewrite the outline provided with this lab (design.sv and testbench.sv)
    c. Create the mux2.v file and fill it with the contents provided.
    d. Identify what is to be done to produce specified output.
    e. We can use the Figure 4.1 to understand writeback stage well.
    f. Check for errors.
    g. Run the code for test cases.

3. Results: In this lab assignment we tested for different ALU operations using Structural Verilog. The output displayed 5 test cases which was the same amount that was expected. Since the index starts from 0 the output includes test cases 0-4. The output included control signals from the MemtoReg and the Writereg registers.

4. Conclusion: In conclusion of this lab, I learned about the different stages of execution in the single cycle MIPS processor. The stage of execution we focused on was the MIPS Write Back stage. It is the simplest stage that involves a single

mux and a bunch of buses. One of the challenging parts of this lab was following what was happening in the Figure 4.1. Other challenging part was to keep track of the variables because they were pretty similar. I still don't understand the warning given by eda playground in the testbench.

5. Notes: ALUOut is a result of an ALU computation that is 32-bit. Read Data is 32-bit data read from DMEM. WriteReg bus carries a value which selects a particular register to be overwritten. MemtoReg control signal is passed into WB stage and it determines if the output of the ALU is passed back to the register file or if a value from DMEM is passed to register file. Register Write Control signal is passed from the MEM stage to the WB stage, it is used to determine if data should be written to a register.

6. Lab Screenshots:

```
testbench.sv:18: warning: task definition for "testcase" has an empty port declaration list!
Test Case           0
aluoutw = 12153524      readdata = c0895e81
memtoregm = 0   resultw = 12153524
regwritem = 1   regwritew = 1
writeregm = 04  writeregw = 04

Test Case           1
aluoutw = b1f05663      readdata = 06b97b0d
memtoregm = 1   resultw = 06b97b0d
regwritem = 1   regwritew = 1
writeregm = 06  writeregw = 06

Test Case           2
aluoutw = b2c28465      readdata = 89375212
memtoregm = 0   resultw = b2c28465
regwritem = 1   regwritew = 1
writeregm = 00  writeregw = 00

Test Case           3
aluoutw = 06d7cd0d      readdata = 3b23f176
memtoregm = 1   resultw = 3b23f176
regwritem = 1   regwritew = 1
writeregm = 1e  writeregw = 1e

Test Case           4
aluoutw = 76d457ed      readdata = 462df78c
memtoregm = 0   resultw = 76d457ed
regwritem = 1   regwritew = 1
writeregm = 1c  writeregw = 1c

Done
```