DEPARTMENT OF COMPUTER ENGINEERING,

FACULTY OF ENGINEERING, UNIVERSITY OF JAFFNA

EC5110 – Computer Architecture and Organization

LABORATORY SESSION 3

Design and Simulation of a single cycle MIPS processor

(Time duration 3 hours only)

AIM: Design of a single cycle MIPS processor

Objectives:

- Understanding HDL code of components of the single MIPS processor
- Designing modules using Verilog
- Testing HDL design of a single MIPS processor

Software:

- ModelSim - Intel FPGA Starter Edition Model Technology ModelSim - Intel FPGA
- Quartus Prime 20.1 Lite

Exercises: You have been given Verilog files for all components of the single cycle MIPS processor. However, the Verilog file for ALU control is blank.

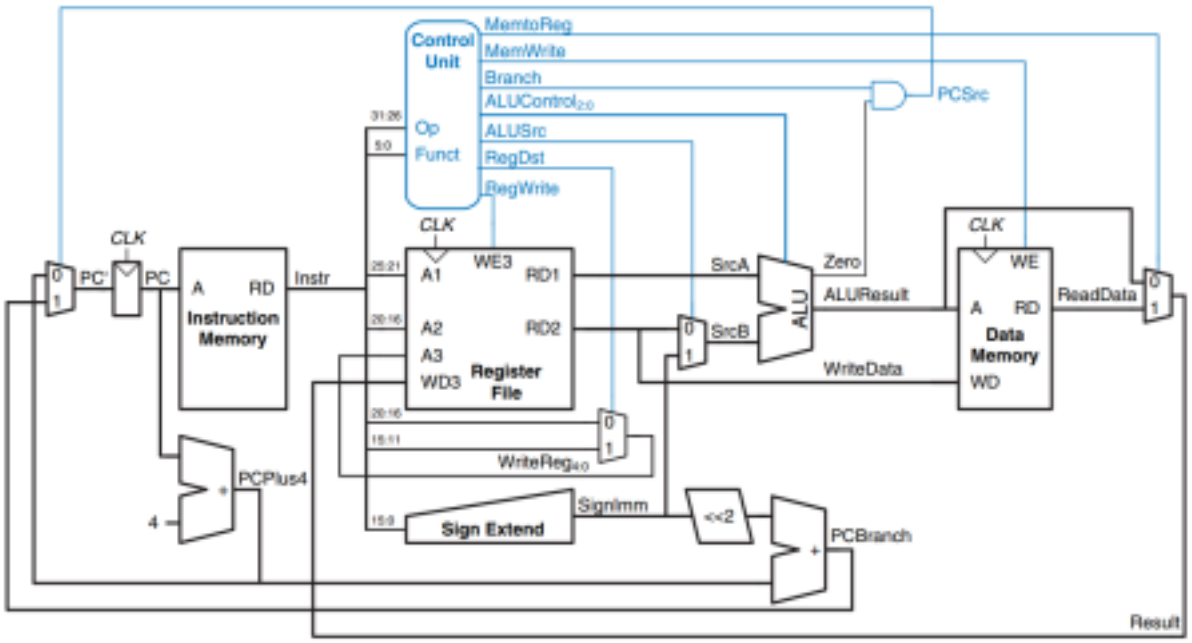

Figure 1: Single cycle datapath without jump

- Refer the other files and write the Verilog code for the “alu” module.
- Test the module using the given test bench

The data file given is based on the following code:

```
# mipstest.asm
# David_Harris@hmc.edu, Sarah_Harris@hmc.edu 31 March 2012
#
# Test the MIPS processor.
# add, sub, and, or, slt, addi, lw, sw, beq, j
# If successful, it should write the value 7 to address 84

#          Assembly              Description            Address    Machine
main:      addi $2, $0, 5        # initialize $2 = 5      0          20020005
           addi $3, $0, 12       # initialize $3 = 12     4          2003000c
           addi $7, $3, -9       # initialize $7 = 3      8          2067fff7
           or   $4, $7, $2       # $4 = (3 OR 5) = 7      c          00e22025
           and  $5, $3, $4       # $5 = (12 AND 7) = 4    10         00642824
           add  $5, $5, $4       # $5 = 4 + 7 = 11        14         00a42820
           beq  $5, $7, end      # shouldn't be taken     18         10a7000a
           slt  $4, $3, $4       # $4 = 12 < 7 = 0        1c         0064202a
           beq  $4, $0, around   # should be taken        20         10800001
           addi $5, $0, 0        # shouldn't happen       24         20050000
around:    slt  $4, $7, $2       # $4 = 3 < 5 = 1         28         00e2202a
           add  $7, $4, $5       # $7 = 1 + 11 = 12       2c         00853820
           sub  $7, $7, $2       # $7 = 12 - 5 = 7        30         00e23822
           sw   $7, 68($3)       # [80] = 7               34         ac670044
           lw   $2, 80($0)       # $2 = [80] = 7          38         8c020050
           j    end              # should be taken        3c         08000011
           addi $2, $0, 1        # shouldn't happen       40         20020001
end:       sw   $2, 84($0)       # write mem[84] = 7      44         ac020054
```

Figure 2: Assembly and machine code for MIPS test program

Submission:

- You **only change in the code** for the **alu** module
- Write a report which consists of the output results recorded as screenshots.
- The report should be a word file with name as L3_20YY_E_XXX. Here 20YY_E_XXX is your registration number, you must use the given word file.
- Create a zip file which contains the verilog program and the report file as in the given structure, you should use the given file names except when 20YY_E_XXX appears in replace YY and XXX according to your registration number.
- The zip file should only contain your word file report and code files in the folder with the name L3_20YY_E_XXX, no additional files should be attached.
- Upload the zip file on/before given deadline via team.