Lab Assistant: Jason Fredrick
jasondfredrick@gmail.com
909-996-5548 (SMS text only)

Lab2 Notes

- **The register module is included for this lab (http://cse.csusb.edu/egomez/cs401/source/lab2.zip)**
- **If you add a new module to your project, you must instantiate it for it to show up properly. I have been told that closing the Project Navigator and reopening it will also make the module show up.**
- **The Sign Extension module does not have local names for its Input and Output. Feel free to name it in/out or signExtendIn/signExtendOut.**
- **The control unit module needs to match the table in Figure 2.2. This can be implemented with a case structure. Instruction in the table is wrong and actually is opcode. R-format is synonymous with R-type (opcode = 6'b0). The other opcodes for lw, sw, and beq can be found online under MIPS I-Type opcodes. X's and x's are valid bits in Verilog (x = unknown).**
- **You must instantiate both the IFETCH and the IDECODE modules in the PIPELINE module. Your pipeline module will be your test bench; the memory module will be the stimulus for this lab; refer to the lab manual for the inputs.**
- **Make sure you are changing your Radix on your timing diagram values. Decimal will take up the least amount of space between changes if you can't see the values.**
- **Your timing diagram will have many more signals this time.**

Lab 2 requirements

Submit your Lab write-up (document preferably doc or rtf) along with your Verilog source code as a compressed file to jasondfredrick@gmail.com. Be sure to include your section (Tuesday/Thursday)

You can RAR the package together using WinRAR (http://www.rarlabs.com).

I would prefer that you send the document and source code in one RAR file.

You can work in teams of 2 or individually; however, make sure that both of your names are listed in the compressed file to receive credit.

I expect the file to have the following naming convention: lastname1[-lastname2]-labX-Y.rar where lastname1 [and lastname2] are the surnames of the team members and X is the lab number and Y is the lab section T/R.

Example: fredrick-lab1-R.rar for Fredrick Lab1 on Thursday.

Document the source code by supplying your name(s) and any comments you have regarding your code. Use the extreme programming approach, because it is self-documenting (i.e. meaningful variable names).

The labs will be due one week from the start date. Pace yourself and if you have any questions, feel free to email or text me.

Lab Assistant: Jason Fredrick
jasondfredrick@gmail.com
909-996-5548 (SMS text only)

Write-up Requirements

Introduction: 1 paragraph (5-7 sentences)

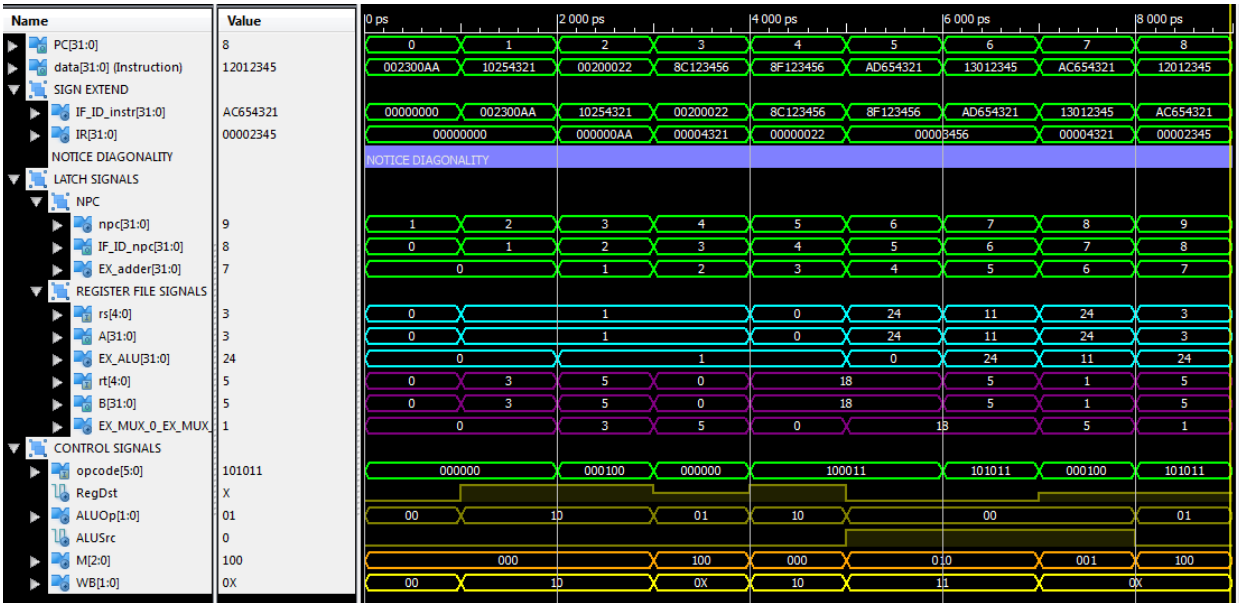
-Provide a description of the Lab and give a brief overview of what the top level module is intended to do. Also include any changes to the previous stages if required.

Summary: 1 paragraph + Timing Diagram and/or Detailed Simulation Log

-Provide an explanation behind the signals on your timing diagram and make sure to remove the unnecessary signals.

-The explanation should be thorough enough so that I can tell that you understand the timing diagram or detailed simulation log.

-Make sure you remove unconnected signals (I.E. high impedance ZZZZZZ…)


Name
Value
0 ps
2 000 ps
4 000 ps
6 000 ps
8 000 ps
PC[31:0] 8 | 0 1 2 3 4 5 6 7 8
data[31:0] (Instruction) 12012345 | 002300AA 10254321 00200022 8C123456 8F123456 AD654321 13012345 AC654321 12012345
SIGN EXTEND
IF_ID_instr[31:0] AC654321 | 00000000 002300AA 10254321 00200022 8C123456 8F123456 AD654321 13012345 AC654321
IR[31:0] 00002345 | 00000000 000000AA 00004321 00000022 00003456 00004321 00002345
NOTICE DIAGONALITY
LATCH SIGNALS
NPC
npc[31:0] 9 | 1 2 3 4 5 6 7 8 9
IF_ID_npc[31:0] 8 | 0 1 2 3 4 5 6 7 8
EX_adder[31:0] 7 | 0 1 2 3 4 5 6 7
REGISTER FILE SIGNALS
rs[4:0] 3 | 0 1 0 24 11 24 3
A[31:0] 3 | 0 1 0 24 11 24 3
EX_ALU[31:0] 24 | 0 1 0 24 11 24
rt[4:0] 5 | 0 3 5 0 18 5 1 5
B[31:0] 5 | 0 3 5 0 18 5 1 5
EX_MUX_0_EX_MUX 1 | 0 3 5 0 18 5 1
CONTROL SIGNALS
opcode[5:0] 101011 | 000000 000100 000000 100011 101011 000100 101011
RegDst X
ALUOp[1:0] 01 | 00 10 01 10 00 01
ALUSrc 0
M[2:0] 100 | 000 100 000 010 001 100
WB[1:0] 0X | 00 10 0X 10 11 0X

Your timing diagram should look something like this: DO NOT USE MY PICTURE PROVIDED. It is merely for your reference.

The required signals for full credit on the timing diagram are RS, RT, RD, A, B, Instruction (data in the diagram), SignExtendOutput (IR in the diagram), opcode and all the 9 bits of the Control Signals. These signals are meant to show that your instruction is being decoded properly.

Extra Credit: Implement initialization of memory and regfile from a txt file using Verilog code.