

***Introduction***

Tasks which were completed in lab:

* Review MIPS instructions.
* Create block diagrams for:
  + Datapath
  + Control Unit
  + Instruction Memory and Data Memory
  + Processor Core
  + Complete Processor
* Produce a DUT and an eyeballing test bench to validate the design through waveform simulation.

***Design Methodology***

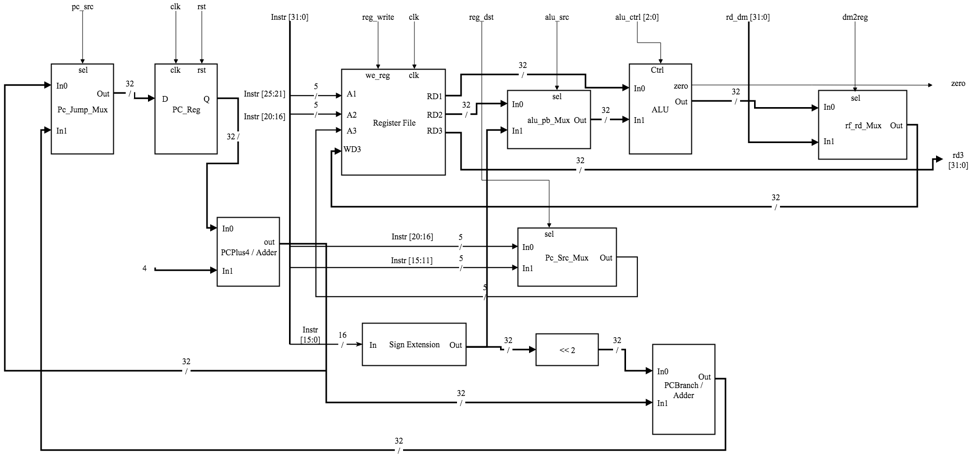


Figure 1. Data Path Diagram

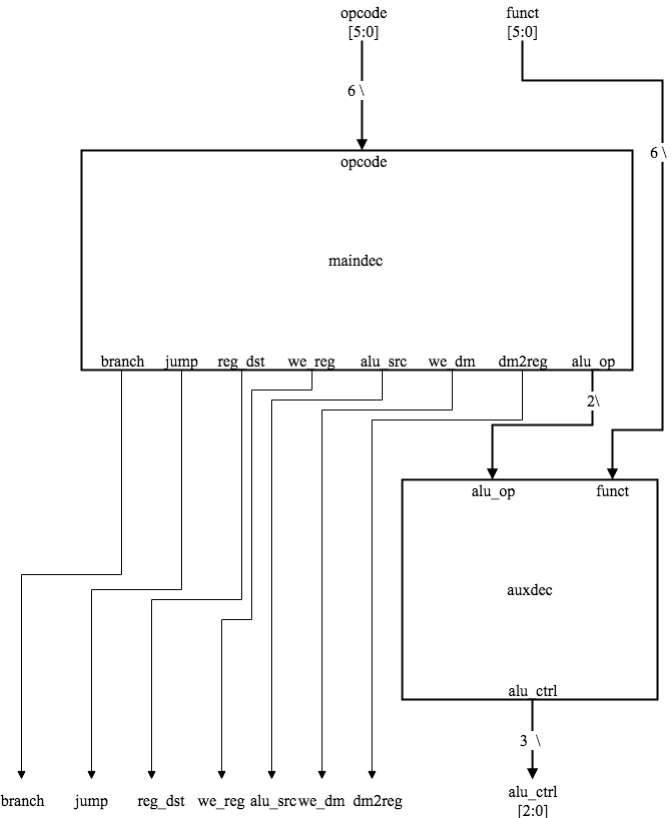


Figure 2. Control Unit Diagram

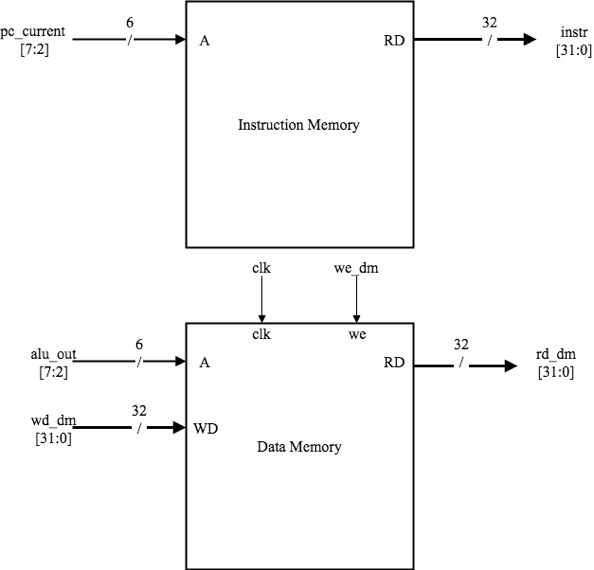


Figure 3. Instruction Memory and Data Memory Diagram

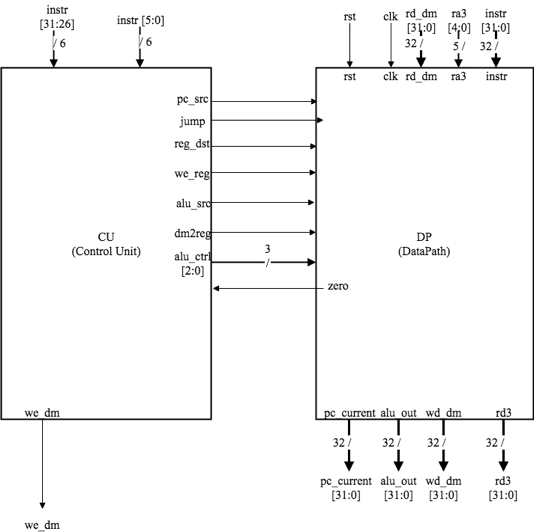


Figure 4. Control Unit and Data Path Diagram

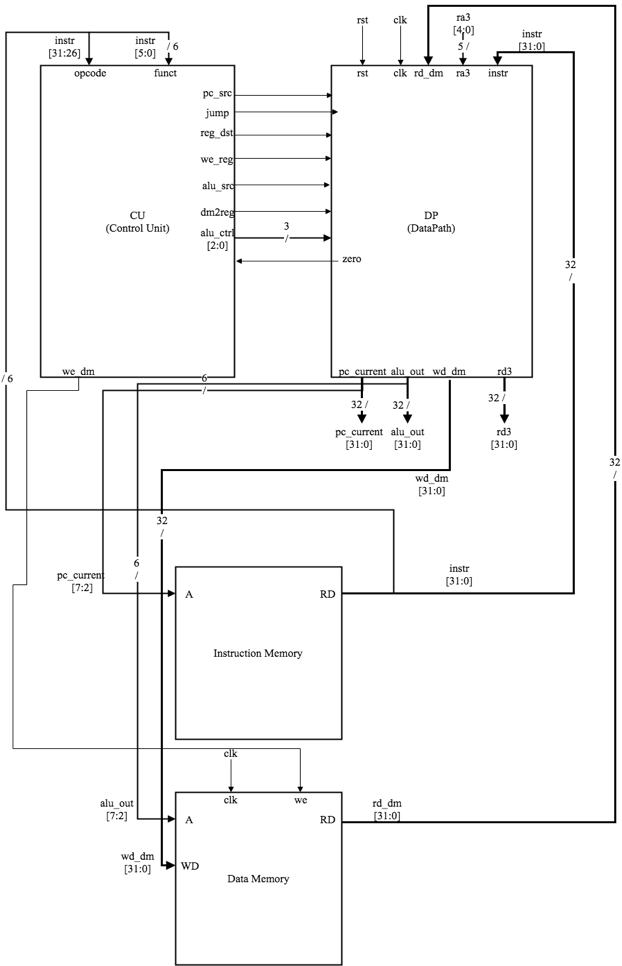


Figure 5. Complete Processor Diagram

***Simulation Test Plan***

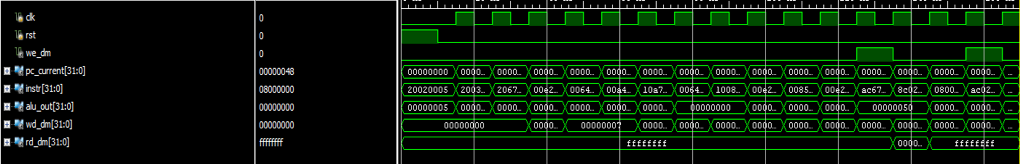


Figure 6. Test Bench Waveform

***Conclusion***

This lab assignment was completed with ease. The low involvement on the student’s part made this a fairly simple task to complete. The main benefit from this lab to the student is the ability to see the inner workings of a MIPS processor in a format that is familiar to them. This also deepen our understanding of how the three types of MIPS instructions work.

***Source Code***

| **Lab5.asm** |
| --- |
| main:  addi $2, $0, 5  addi $3, $0, 12  addi $7, $3, -9  or $4, $7, $2  and $5, $3, $4  add $5, $5, $4  beq $5, $7, end  slt $4, $3, $4  beq $4, $0, around  addi $5, $0, 0  around:  slt $4, $7, $2  add $7, $4, $5  sub $7, $7, $2  sw $7, 68($3)  lw $2, 80($0)  j end  addi $2, $0, 1  end:  sw $2, 84($0)  j main |

| **memfile.dat** |
| --- |
| 20020005  2003000C  2067FFF7  00E22025  00642824  00A42820  10A7000A  0064202A  10080001  20050000  00E2202A  00853820  00E23822  AC670044  8C020050  08000011  20020001  AC020054  08000000 |

| **tb\_mips\_top.v** |
| --- |
| `timescale 1ns / 1ps  //////////////////////////////////////////////////////////////////////////////////  // Company:  // Engineer:  //  // Create Date:  // Design Name:  // Module Name: tb\_mips\_top  // Project Name:  // Target Devices:  // Tool Versions:  // Description:  //  // Dependencies:  //  // Revision:  // Revision 0.01 - File Created  // Additional Comments:  //  //////////////////////////////////////////////////////////////////////////////////  module tb\_mips\_top;  reg clk, rst;  wire we\_dm;  wire [31:0] pc\_current, instr, alu\_out, wd\_dm, rd\_dm;      mips\_top DUT (clk, rst, we\_dm, pc\_current, instr, alu\_out, wd\_dm, rd\_dm);  task tick; begin #5 clk = 1; #5 clk = 0; end endtask  task rest; begin #5 rst = 1; #5 rst = 0; end endtask    initial begin  clk = 0;  rst = 1;  rest;  while(pc\_current != 32'h48)  begin  tick;  case(pc\_current)  32'h0: begin  if(alu\_out != 5) $display("Error with alu at instruction addr: 0");  if(instr != 32'h20020005) $display("Error with instr at instruction addr: 0");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 0");  if(we\_dm != 1) $display("Error with we\_dm at instruction addr: 0");  end  32'h4: begin  if(alu\_out != 12) $display("Error with alu at instruction addr: 4");  if(instr != 32'h2003000c) $display("Error with instr at instruction addr: 4");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 4");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: 4");  end  32'h8: begin  if(alu\_out != 3) $display("Error with alu at instruction addr: 8");  if(instr != 32'h2067fff7) $display("Error with instr at instruction addr: 8");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: 8");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 8");  end  32'hc: begin  if(alu\_out != 7) $display("Error with alu at instruction addr: c");  if(instr != 32'h00e22025) $display("Error with instr at instruction addr: c");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: c");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: c");  end  32'h10: begin  if(alu\_out != 4) $display("Error with alu at instruction addr: 10");  if(instr != 32'h00642824) $display("Error with instr at instruction addr: 10");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: 10");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 10");  end  32'h14: begin  if(alu\_out != 11) $display("Error with alu at instruction addr: 14");  if(instr != 32'h00a42820) $display("Error with instr at instruction addr: 14");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: 14");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 14");  end  32'h18: begin  if(instr != 32'h10a7000a) $display("Error with instr at instruction addr: 18");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: 18");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 18");  end  32'h1c: begin  if(alu\_out != 0) $display("Error with alu at instruction addr: 1c");  if(instr != 32'h0064202a) $display("Error with instr at instruction addr: 1c");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: 1c");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 1c");  end  32'h20: begin  if(instr != 32'h10080001) $display("Error with instr at instruction addr: 20");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: 20");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 20");  end  32'h24: begin $display("Error with beq instr at instruction addr: 24"); end  32'h28: begin  if(instr != 32'h00e2202a) $display("Error with instr at instruction addr: 28");  if(alu\_out != 1) $display("Error with alu instr at instruction addr: 28");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: 28");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 28");  end  32'h2c: begin  if(instr != 32'h00853820) $display("Error with instr at instruction addr: 2c");  if(alu\_out != 12) $display("Error with alu instr at instruction addr: 2c");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: 2c");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 2c");  end  32'h30: begin  if(instr != 32'h00e23822) $display("Error with instr at instruction addr: 30");  if(alu\_out != 7) $display("Error with alu instr at instruction addr: 30");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: 30");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 30");  end  32'h34: begin  if(instr != 32'hac670044) $display("Error with instr at instruction addr: 34");  if(wd\_dm != 32'h7) $display("Error with wd\_dm at instruction addr: 34");  if(we\_dm != 1) $display("Error with we\_dm at instruction addr: 34");  if(alu\_out != 32'h50) $display("Error with alu at instruction addr: 34");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 34");  end  32'h38: begin  if(instr != 32'h8c020050) $display("Error with instr at instruction addr: 38");  if(rd\_dm != 32'h7) $display("Error with wd\_dm at instruction addr: 38");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: 38");  if(alu\_out != 32'h50) $display("Error with alu at instruction addr: 38");  end  32'h3c: begin  if(instr != 32'h08000011) $display("Error with instr at instruction addr: 3c");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: 3c");  if(alu\_out != 32'h0) $display("Error with alu at instruction addr: 3c");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 3c");  end  32'h40: begin  $display("Error with jump instr at instruction addr: 3c");  end  32'h44: begin  if(we\_dm != 1) $display("Error with we\_dm at instruction addr: 44");  if(alu\_out != 32'h54) $display("Error with alu\_out at instruction addr: 44");  if(wd\_dm != 32'h7) $display("Error with wd\_dm at instruction addr: 44");  if(instr != 32'hac020054) $display("Error with instr at instruction addr: 44");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 44");  end  32'h48: begin  if(instr != 32'h08000000) $display("Error with instr at instruction addr: 48");  if(we\_dm != 0) $display("Error with we\_dm at instruction addr: 48");  if(alu\_out != 32'h0) $display("Error with alu\_out at instruction addr: 48");  if(rd\_dm != 32'hffffffff) $display("Error with rd\_dm at instruction addr: 48");  end  endcase  end  $display("End of Simulation");  $finish;  end  endmodule |