Computer Organization 2023 Lab 4
110550126 曾家祐

Finished part:

I used the template from TA so I only modify the four file.

- Decoder.v:
- the input is instr[31:26], and distinguishes it is R type, jump, lw, sw,etc.and output the regwrite, ALUOP, ALUsrc, etc.
- ALU_Ctrl.v:

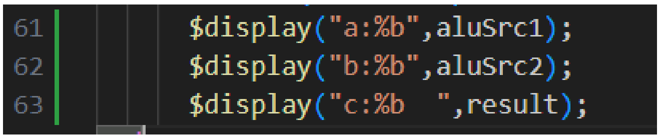
  From the ALUop we got from decoder,set the ALU_operation and Furslt, if ALUop is R type, from function code we can know it is add, sub, or, etc.
- ALU.v:

  From the input ALU_operation to know what operation to do and from ALUsrc1, ALUsrc2 to get the result.
- Simple_Single_CPU.v:

  Combine the module with the wires and MUX to finish the simple single cpu.

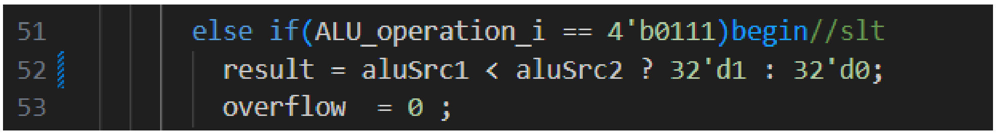
Problems you met and solutions:

When I finish my code, I meet an error is ERROR: JUMP(r15) , so I go to check my code about jump. However, I didn’ t find any mistake. Therefore, in my ALU.v I print out the alusrc1, alusrc2, and result to check every testcase where is wrong.



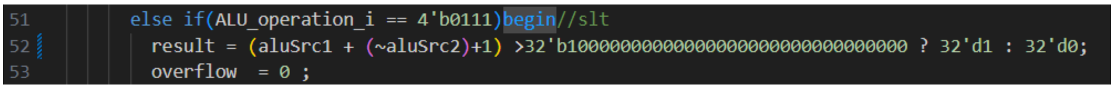
```
            $display("a:%b",aluSrc1);
            $display("b:%b",aluSrc2);
            $display("c:%b  ",result);
```

Then I found that after jump the instruction is “slt r15,r5,r3” R5 is -2 R3 is 2, and expect that r15 = 1.but my output is r15 = 0. After display it I found it my original code will seen the number as unsigned. So the code seen R5 = 2^32-2 not -2. So the result is 0 not 1.



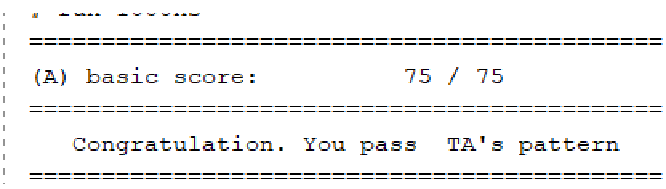
```
        else if(ALU_operation_i == 4'b0111)begin//slt
          result = aluSrc1 < aluSrc2 ? 32'd1 : 32'd0;
          overflow  = 0 ;
```

So I change my code, use the form we learn before and check the leftmost bit is 1 or not.



```
        else if(ALU_operation_i == 4'b0111)begin//slt
          result = (aluSrc1 + (~aluSrc2)+1) >32'b10000000000000000000000000000000 ? 32'd1 : 32'd0;
          overflow  = 0 ;
```

And I got the correct answer.



```
=============================================
 (A) basic score:           75 / 75
=============================================
   Congratulation. You pass  TA's pattern
=============================================
```