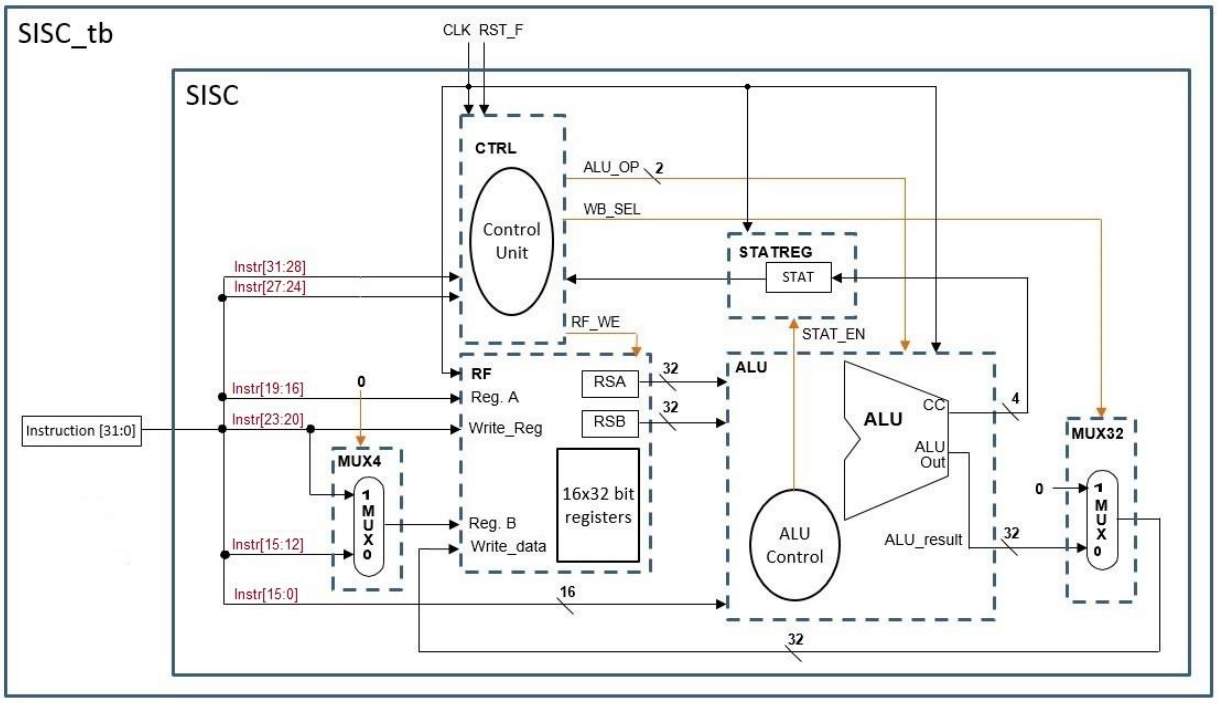
ECE:3350 Spring 2018 -
Computer Architecture and Organization
Simple Instruction Set Computer (SISC) Project

Part 1: Due Monday, February 26, at 8:30 am

In this first part of the project you will implement the modules and datapath depicted below:



You are given:

- All of alu.v, rf.v, statreg.v, mux4.v, and mux32.v, which you may not modify.
- Detailed descriptions of each module's function and control lines, in block comment form, at the top of every Verilog file provided to you.
- A mostly empty ctrl.v, which contains descriptions for the inputs and outputs, as well as some parameters, reset processes, and other miscellaneous data.

You are required to:

- Finish the implementation of ctrl.v as a finite state machine, so that the control lines shown above in orange operate correctly as described by the given files.
- Write a Verilog file named 'sisc.v' which contains a module 'sisc' that instantiates each of the six modules shown above, and connects their control and data signals.

- You are given a Verilog file named 'sisc_tb_p1.v' which contains a module 'sisc_tb' that instantiates the sisc module, drives the CLK and RST_F signals as described below, and generates instructions that test each of the supported instructions for Part 1.
- Scan or digitally draw a state diagram for the FSM in ctrl.v and save it in your project folder.
- Compress your project directory, including the 'work' directory and the state diagram, into a .zip file named 'project_p1.zip' and submit it to your "Verilog Project – Part 1" dropbox on ICON.

Part 1 Notes:

- Read the documentation included with each file carefully before beginning to connect them.
- In the RF module, the outputs of the register file R[A] and R[B] are latched in the RSA and RSB registers on the positive edge of the clock.
- In the ALU module, the output of the ALU is latched in the ALUOut register on the positive edge of the clock.
- The sisc module should take in three inputs (IR, CLK, and RST_F) and have no outputs.
- The sisc_tb module should have no inputs or outputs.
- In the testbench file, the clock process runs at the rate of 1 cycle/10 time units. Since the default time unit in ModelSim is 1 ns, this gives us a 10 ns clock period. The RST_F reset line is active low and is held at 0 for 20 ns or 2 clock cycles before being set to logic 1. Generation of the first instruction is delayed until time = 35 ns so that it appears at IR on the second clock cycle after the reset line has been set to logic 1. All further instructions are delayed by 50 ns per instruction to allow for all five cycles to execute in the control unit. At the time the instruction changes, the cntrl.v module should enter the FETCH state.
- The instructions your design should support by the end of Part 1 are NOP, ADD, ADD IMM, SUB, NOT, OR, AND, XOR, ROTR, ROTL, SHFR, SHFL, and HLT. Always include a HLT command at the end of your instructions to end execution.
- You may use any $monitor(…) statements you wish to test your design, but in the sisc.v file that you submit, please monitor the following signals: IR, R1 through R6, RD_SEL, ALU_OP, WB_SEL, RF_WE, and the 32-bit line connected to the write_data input of the rf module.
    - To monitor a signal within an instantiated module, such as R1 and R2, use the dot operator. If you have instantiated module rf with the name my_rf, you can access R1 with the following syntax: `$monitor("R1=%h", my_rf.ram_array[1]);`.

Submission Overview:

- Your .zip file should be named "Project_p1.zip" and contain:
    - A short description of your project implementation and the names of your project team
    - Alu.v, mux4.v, mux32.v, rf.v, and statreg.v, exactly as they were provided.
    - Ctrl.v, sisc.v, and sisc_tb.v, completed by your group.
        - Note that sisc.v should contain the $monitor statement described above!
    - The 'work' directory.

- Your FSM state diagram.
- Your simulation transcript.
- ModelSim screen captures.

Part 1 Grading Rubric:

FSM Diagram	10 pts
Ctrl.v Implementation	15 pts
Sisc_tb.v Implementation	15 pts
Correct Execution of All Instructions	30 pts
Total	**70 pts**