CS7381 Project 6: Verilog Code Development – MIPS ALU Controller Design

Name: Bingying Liang
ID: 48999397
Distance

Apr 2 2023

For this exercise, you will write a Verilog code program to implement the MIPS ALU Control Unit. You will test your Control Unit module using the testbench provided in the assignment.

Recall the following web site links for Verilog help:

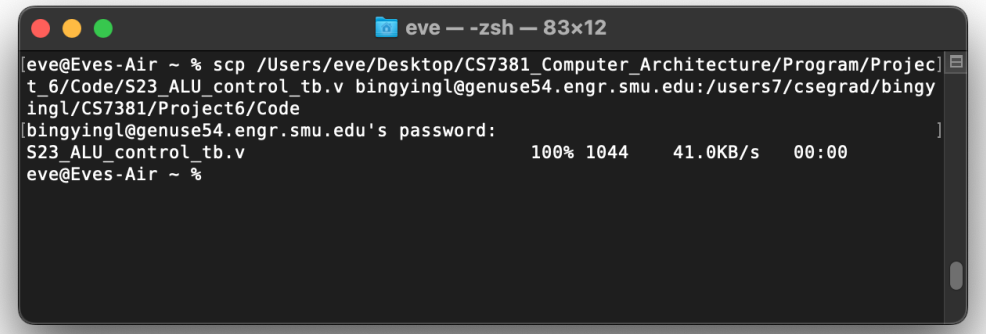
- http://lyle.smu.edu/ manikas/CAD_tool_info.html
- https://s2.smu.edu/ manikas/CAD_Tools/Verilog/Xcelium.html.

In the previous Verilog assignment, you modified the code for a MIPS ALU. One of the inputs to the ALU was the ALU Control signal vector ALUctl. The ALUctl vector is output from the ALU Control Unit, and its value is based on the ALUOp signal vector from the MIPS main Control Unit and funct field of the MIPS instruction. The following table shows how these signals are mapped:

opcode	ALUOp	Operation	funct	ALU function	ALUctl
lw	00	load word	XXXXXX	add	0010
beq	01	branch equal	XXXXXX	subtract	0110
R-type	10	add	100000	add	0010
		subtract	100010	subtract	0110
		AND	100100	AND	0000
		OR	100101	OR	0001
		set-on-less-than	101010	set-on-less-than	0111
		NOR	100111	NOR	1100
		XOR	100110	Exclusive-Or	1110

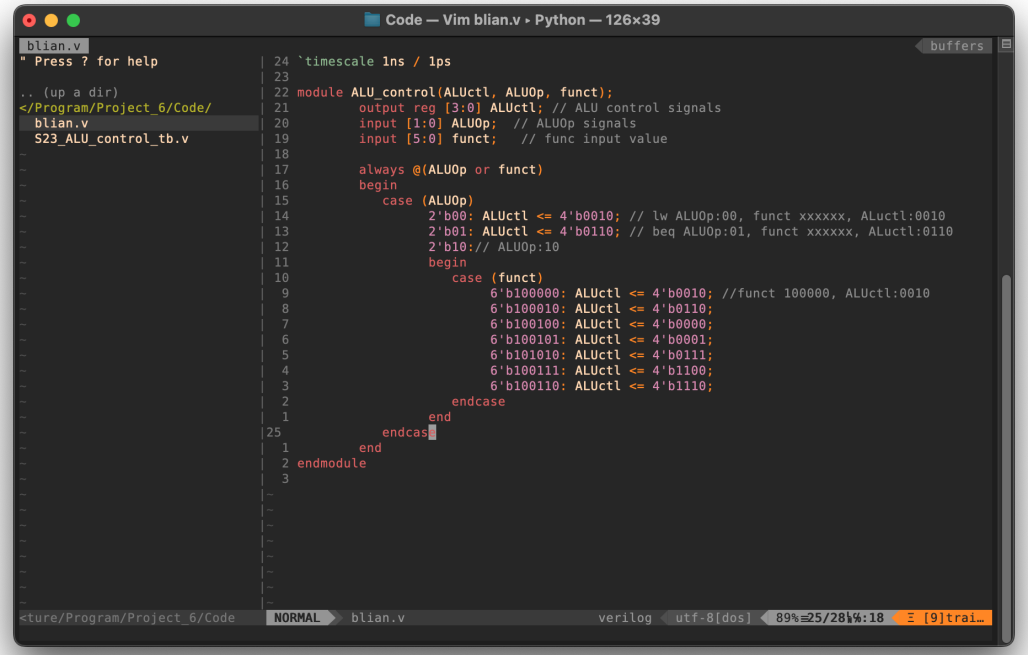
1. Please download the following Verilog file:

   (a) S23_ALU_control_tb.v - the testbench for testing your ALU Control Unit

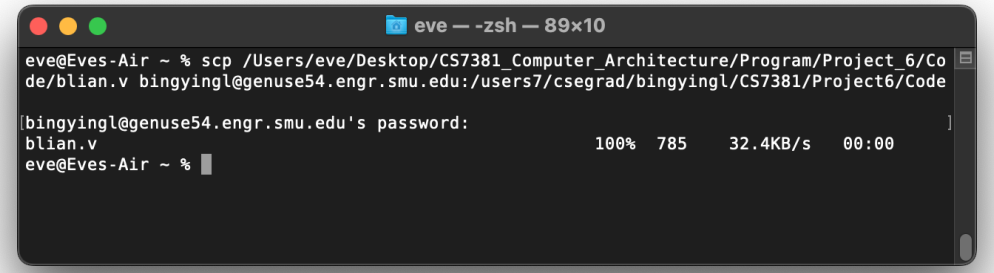


2. Using the function mapping table above, design your ALU Control Unit so that it produces the correct ALU Control signal vector given the inputs ALUOp and funct.

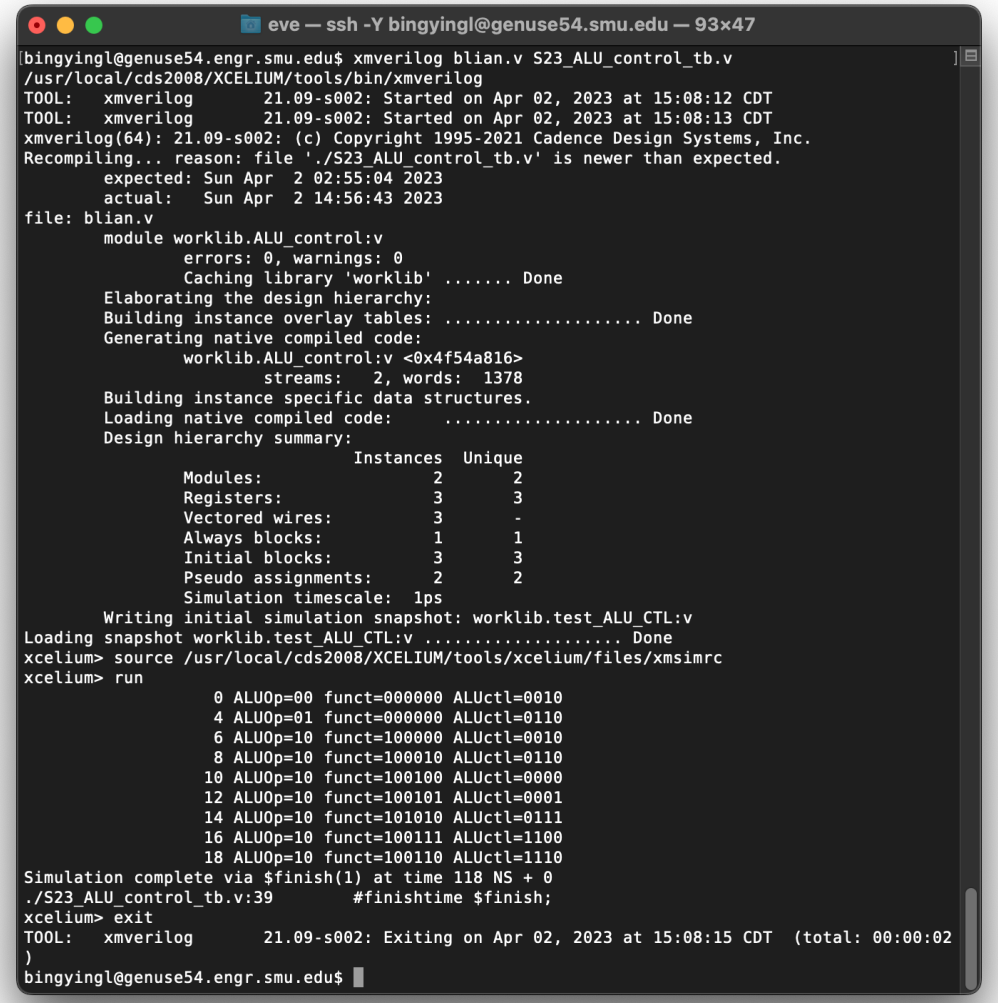
   (a) Save the program as a *.v file – use the first initial of your first name and the first 4 letters of your last name, then the number 2 (to distinguish from your code for Project 5). For example, my file submission name would be tmani2.v.



(b) Test your ALU Control Unit using the testbench provided in Step 1.



```
eve — -zsh — 89×10
eve@Eves-Air ~ % scp /Users/eve/Desktop/CS7381_Computer_Architecture/Program/Project_6/Co
de/blian.v bingyingl@genuse54.engr.smu.edu:/users7/csegrad/bingyingl/CS7381/Project6/Code

bingyingl@genuse54.engr.smu.edu's password:
blian.v                                           100%  785    32.4KB/s   00:00
eve@Eves-Air ~ %
```



```
eve — ssh -Y bingyingl@genuse54.smu.edu — 93×47
bingyingl@genuse54.engr.smu.edu$ xmverilog blian.v S23_ALU_control_tb.v
/usr/local/cds2008/XCELIUM/tools/bin/xmverilog
TOOL:   xmverilog       21.09-s002: Started on Apr 02, 2023 at 15:08:12 CDT
TOOL:   xmverilog       21.09-s002: Started on Apr 02, 2023 at 15:08:13 CDT
xmverilog(64): 21.09-s002: (c) Copyright 1995-2021 Cadence Design Systems, Inc.
Recompiling... reason: file './S23_ALU_control_tb.v' is newer than expected.
        expected: Sun Apr  2 02:55:04 2023
        actual:   Sun Apr  2 14:56:43 2023
file: blian.v
        module worklib.ALU_control:v
                errors: 0, warnings: 0
                Caching library 'worklib' ....... Done
        Elaborating the design hierarchy:
        Building instance overlay tables: ..................... Done
        Generating native compiled code:
                worklib.ALU_control:v <0x4f54a816>
                        streams:   2, words:  1378
        Building instance specific data structures.
        Loading native compiled code:     ..................... Done
        Design hierarchy summary:
                                Instances  Unique
                Modules:                2       2
                Registers:              3       3
                Vectored wires:         3       -
                Always blocks:          1       1
                Initial blocks:         3       3
                Pseudo assignments:     2       2
                Simulation timescale:  1ps
        Writing initial simulation snapshot: worklib.test_ALU_CTL:v
Loading snapshot worklib.test_ALU_CTL:v ..................... Done
xcelium> source /usr/local/cds2008/XCELIUM/tools/xcelium/files/xmsimrc
xcelium> run
                  0 ALUOp=00 funct=000000 ALUctl=0010
                  4 ALUOp=01 funct=000000 ALUctl=0110
                  6 ALUOp=10 funct=100000 ALUctl=0010
                  8 ALUOp=10 funct=100010 ALUctl=0110
                 10 ALUOp=10 funct=100100 ALUctl=0000
                 12 ALUOp=10 funct=100101 ALUctl=0001
                 14 ALUOp=10 funct=101010 ALUctl=0111
                 16 ALUOp=10 funct=100111 ALUctl=1100
                 18 ALUOp=10 funct=100110 ALUctl=1110
Simulation complete via $finish(1) at time 118 NS + 0
./S23_ALU_control_tb.v:39        #finishtime $finish;
xcelium> exit
TOOL:   xmverilog       21.09-s002: Exiting on Apr 02, 2023 at 15:08:15 CDT  (total: 00:00:02
)
bingyingl@genuse54.engr.smu.edu$
```

Compare the result with testbench annotation and table they are the same and mapped

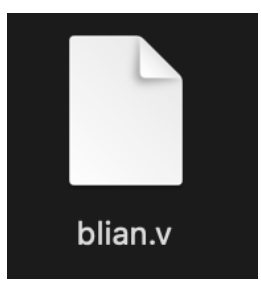
correctly.

eve — ssh -Y bingyingl@genuse54.smu.edu — 93×47

```
// ============================================================
// S23_ALU_control_tb.v
// T. Manikas   2022 Dec 29
//
// testbench for MIPS ALU Control Unit
// ============================================================

`timescale 1ns / 1ps

module test_ALU_CTL;

   parameter finishtime = 100;

   reg [1:0] ALUOp;
   reg [5:0] funct;
   wire [3:0] ALUctl;

   ALU_control DUT(ALUctl, ALUOp, funct);

   initial
     begin
         ALUOp = 0;  funct = 0;
     end


   initial
     begin
         #2 ALUOp = 2'b00;       // lw
         #2 ALUOp = 2'b01;      // beq
         #2 ALUOp = 2'b10;      // R-type
         funct = 6'b100000;       // add
         #2 funct = 6'b100010;       // sub
         #2 funct = 6'b100100;       // and
         #2 funct = 6'b100101;       // or
         #2 funct = 6'b101010;       // slt
         #2 funct = 6'b100111;          // nor
         #2 funct = 6'b100110;          // xor

       #finishtime $finish;

     end

   initial $monitor($time, " ALUOp=%b funct=%b ALUctl=%b",
                    ALUOp, funct, ALUctl);

endmodule // test_ALU_CTL
"S23_ALU_control_tb.v" [dos] 49L, 1044C                    30,1-8        Top
```

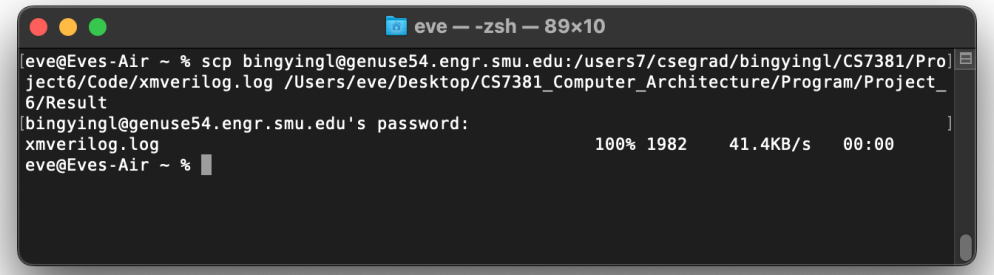
3. Please include the following for your homework submission:

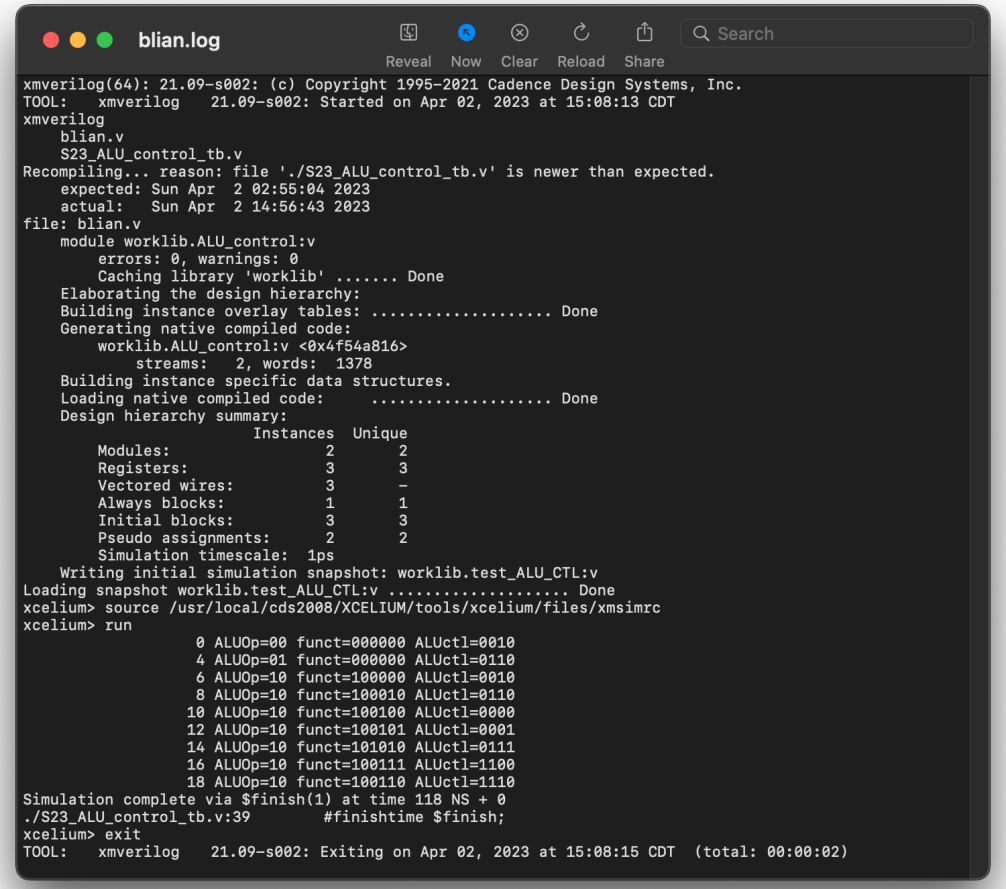
   (a) Your ALU Control Unit Verilog file – submit the actual *.v file so that the grader can run them.



(b) Your testbench results – this can be a copy of the results on a Word document.

I download the result file log from the server and modify the name as blian.log





(c) PLEASE MAKE SURE THAT YOUR NAME APPEARS ON ALL SUBMITTED ITEMS FOR PROPER CREDIT