EEE 333, ASU

Lab #2 (V1)

Team Lab (3-4 students)

The objective of this lab is to practice your Verilog coding and the modeling of combinational logic.

Lab Goal:

For this lab, you will write the Verilog code for a 4-bit arithmetic/logic unit (ALU) according to the specifications outlined under the specifications section of this document. You will write a TestBench to thoroughly test your design and you will program the BASYS3 board in order to implement your design.

Design Specifications for the ALU

The VERILOG model you implement should be for a 4-bit arithmetic/logic unit (ALU) which has as inputs two 4-bit vectors *aluin_a* and *aluin_b* as well as a carry in, Cin. The output is a 4-bit vector *alu_out*. The ALU should operate on the inputs depending on the control inputs *C* in the following table:

C(3 down to 0)	Operation
0 0 0 0	*alu_out* = *aluin_a* + *aluin_b*
0 0 0 1	*alu_out* = *aluin_a* + *aluin_b* + *Cin*
0 0 1 0	*alu_out* = *aluin_a* – *aluin_b*
0 0 1 1	*alu_out* = *aluin_a* – *aluin_b* – *Cin*
0 1 0 0	*alu_out* = *aluin_a* logical shifted right by *aluin_b*
0 1 0 1	*alu_out* = *aluin_a* arithmetic shifted right by *aluin_b*
0 1 1 0	*alu_out* = *aluin_a* rotated right by *aluin_b*
0 1 1 1	*alu_out* = all 0's
1 0 0 0	*alu_out* = *aluin_a* OR *aluin_b* (bitwise or)
1 0 0 1	*alu_out* = *aluin_a* AND *aluin_b* (bitwise and)
1 0 1 0	*alu_out* = *aluin_a* XOR *aluin_b* (bitwise xor)
1 0 1 1	*alu_out* = NOT *aluin_a* (bitwise inversion)
1 1 0 0	*alu_out = undefined. But in this mode, all 16 LEDs should be used to display the 1st and 2nd character of your last name (again)*
1 1 0 1	*alu_out = You decide and tell me what you chose to implement*
1 1 1 0	*alu_out = aluin_a logical shifted left by aluin_b*
1 1 1 1	*alu_out = aluin_a rotated left by aluin_b*

Other Design Requirements

1. If a carry out is generated, then a signal *Cout* should be set.
2. If the result is all 0's, then a signal *zero* should be set.
3. The test bench should test every operation as well as the *Cout* and *Cin* for correct operation. Meaning that the test bench should give the inputs such that you can verify all the test cases.

Signal to BASYS3 Hardware Mapping

The signals you use should map to the following BASYS3 I/Os as per the following table:

aluin_a[3:0]	SW3, SW2, SW1, SW0
aluin_b[3:0]	SW7, SW6, SW5, SW4
C[3:0] (Conditions)	SW15, SW14, SW13, SW12
Cin	BTNC
alu_out[3:0]	LD3, LD2, LD1, LD0
Cout	LD14
zero	LD15
Special Case when C=1100, First two characters of your last name in ASCII (You choose one of team member last name).	All 16 LEDs like in Lab 1

Submission of Completed Lab: ONE PER TEAM

Upon completing this lab, you must upload to blackboard the following files: (do not forget to comment your code).

ALU.v ← The file containing the ALU module and any additional modules you have instantiated into it.

ALU_tb.v ← The file containing your testbench file

ALU.bit ← The bitstream file for programming the BASYS3 board.

A report document uploaded as a .pdf file showing:

1. Cover page with only the following information: course title, experiment number and your team names (First, Last).
2. Example waveforms (a screen capture) containing aluin_a, aluin_b, C, Cin, Cout, alu_out and zero when tested by your testbench for any one of the C conditions in Table 1 (except C=1100). Make sure the signals are clear and the signal values are legible. Put in figure captions explaining what the figure shows.
3. An explanation of what you chose to implement when input C is 1101
4. Your argument and code examples as to why you believe your testbench exhaustively tests your design.
5. Write a short paragraph summarizing the work you did for this experiment, and describe any problems you may have encountered while obtaining your solutions. You may include any helpful hints and improvements you may think of for this experiment.

GRADING:

80% of the grade will be based on whether your ALU.bit file works when downloaded to a BASYS3 board and your in lab demo to the TAs.

20% of the grade will be based on your report. (do not forget to comment your code).

TA Grading specifics (in addition to above stated specs from the professor):

Group Demo grading (100 pts – will be converted to /80):

Your demo will be done by one of the two TAs during your signed up time through doodle. Please **ensure your entire group members** are here:

Verification of the following for grading purposes only:

1.	Simulation -	20 pts
2.	Synthesis –	20 pts
3.	Implementation –	20 pts
4.	Generation of bit file and writing to BASYS3 board –	20 pts
5.	Q&A session of 2 questions –	10 pts each

Two questions will be asked during the demo from the following list to each member, each questions carries a weightage of 10 pts:

- What is combinational logic?
- What is behavioral modeling?
- What is structural modeling?
- State its differences.
- What does the module statement do?
- What is an always block in the code indicate? What is its function?
- What is a MUX?
- Are you using a MUX in this project? If yes, what MUX are you using here?
- How is the MUX represented in the code?
- What is testbench? Why do you need it?
- What is a simulation? What does it simulate in this project?
- What does it mean when you say “programming to the BASYS3 board”? Does this have anything to do with the simulation part of code?

Group Report grading (20 pts total):

- Proper presentation(flow and organization) of the report (5 pts)
  - This would mean, it has a proper introduction which states the objective of the project, followed by an explanation of the fundamentals you learnt in Verilog used for the project, then explanation of design of the code, followed by waveforms and the design errors you encountered while implementing the code on the board.
- Waveforms (as stated by professor) presented in a neat and readable form (5 pts)
- Your argument and code examples as to why you believe your testbench exhaustively tests your design (5pts).
- An explanation of what you chose to implement when input C is 1101 (3 pts)
- A short paragraph summarizing the errors you encountered while simulating OR implementing the design to the board (2 pts)