ECEN2350 Digital Logic – Lab 1

Fall, 2019

Due October 7th, 2019, 4:00pm – 10% of Course Grade

Successful completion of this lab will require use of iVerilog, GTKWave, Quartus, and the DE10-Lite board. You should confirm proper operation of all tools early in the process.

You will work on this lab with one partner. You or your partner must email the instructor at the beginning of the assignment stating you will work together.

This lab will require you to code and compile the design using iVerilog, simulate the design using both GTKWave and textual output, demonstrate the working design to the instructor and/or ELAs, and submit a lab report. While you can work as a team on all aspects of the lab, you must submit your own lab report, written by you alone.

The purpose of Lab 1 is to introduce you to Verilog coding, Verilog testbench creation, and use of Quartus and the DE10-Lite board. Skills acquired by the completion of the lab will include skill and confidence in Verilog coding, knowledge of testbench creation, use of Quartus software, and the ability to map your design into the DE10-Lite hardware.

Lab Description:

Lab1 consists of 3 separate design blocks. These blocks can be created and tested independently of each other. However, the ultimate goal of this lab is to create a final design that integrates all three design blocks. For this reason, how you approach the design process may be influenced by the goal of successful integration. You should read carefully the remainder of this document before beginning to design.

1. Design block #1 is a simple design that tests knowledge of porting a simple Verilog based project to the DE10-Lite board.
2. Design block #2 is the most complex block, and will require understanding of 2's complement arithmetic, and low level implementation of binary adders and subtractors.
3. In design block #3 you will design comparison operations between both unsigned values and 2's complement signed values.
4. The integrated design will combine all three design blocks, and the behavior of the board will depend upon the settings of slide switches [9:8]. This portion of the lab will test your knowledge of hierarchical design. The ease or difficulty of

successful integration will depend on design choices you make for each of the 3 design blocks.

Grading (total 100%)

1. Correct function / demonstration of design block #1. 15%
2. Correct function / demonstration of design block #2. 25%
3. Correct function / demonstration of design block #3. 20%
4. Successful integration of all 3 blocks into a single, demonstratable design. 20%
5. Lab report. 20%

If you are unable to complete item #4, you must demonstrate each of the three design blocks separately to get full credit for items 1 - 3. If you successfully complete item #4, demonstration of the integrated design is the only demonstration necessary. If you are only partially finished with a section when the lab is due, provide what you have for partial credit.

Note: The lab report is mandatory. No credit will be given if the lab report is not submitted. Document as much of the Lab as you were able to complete.

Guidelines for lab report:

This project consists of four sections – Design Block #1, Design Block #2, Design Block #3, and the Integrated Design. For each section, provide the following:

a) If possible limit to one page per design block with the following sections:
   a. Name of design block
   b. Concise theory of operation
   c. Hierarchy of source files
   d. Description of testbench operation (what you tested)
   e. Did your demonstration work properly?

You will submit one PDF Lab Report document containing all four sections, along with one .zip file containing all your Verilog source files and testbench files, testbench output capture text files, and GTKWave screen captures. Please either name your files in a manner that explains what they are, or include a readme.txt file that explains what they are.

Definitions:

Per the DE10-Lite User Manual, the slide switches are defined as DOWN or Logic 0 when the switches are nearest the edge of the DE10-Lite board, and UP or Logic 1 when the switches are pushed toward the center of the DE10-Lite board.

With the DE10-Lite board positioned with the 7 segment displays and slide switches at the bottom of the board:

HEX5 is the leftmost 7 segment display, HEX0 is the rightmost 7 segment display.
SW[9] is the leftmost slide switch, SW[0] is the rightmost slide switch.
LEDR[9] is the leftmost LED, LEDR[0] is the rightmost LED.
KEY[0] is the pushbutton switch closest to the VGA video connector.

1. **Design block #1**

Connect the DE10-Lite slide switches [7:0] to the corresponding LEDs. When a switch is DOWN, the corresponding LED should be on. Moving the switch to the UP position will turn the LED off. When KEY0 is pressed and held, the state of the eight LEDs will toggle.

On the 7 segment displays, display the birthday of one lab partner in MMDDYY format. For example, April 1st, 1999 would display as 040199. When KEY1 is pressed, display the birthday of the other lab partner.

Construct a simple testbench that demonstrates the effect pressing KEY0 and KEY1. Capture the output as either a text file or a GTKWave screen capture.

Hint: Create a separate module to drive the 7 segment displays.

2. **Design block #2**

Build a 4 bit add / subtract circuit will uses slide switches [7:4] as input1 and slide switches [3:0] as the input2. Both values input1 and input2 **represent 2’s complement numbers**. Therefore, the range of each input is -8 to +7, which implies the range of the sum of the inputs will be from -16 to +14, and the range of the difference of the inputs will be from -15 to +15. However, the 2’s complement result must fit in 4 bits, so you will deal with arithmetic overflow.

Create add / subtract logic using multiple full adder circuits to create ripple carry logic. Your design can instantiate logic gates, or use Boolean equations, but **you cannot use the Verilog arithmetic (+ or -) operators**.

The value of input1 is set using SW [7:4], and HEX4 will display the absolute value of input1 as a Base10 number. HEX5 will display the sign of the input1 (blank for positive, - for negative). The value of input2 is set using SW [3:0], and HEX2 will display the absolute value of input2 as a Base10 number. HEX3 will display the sign of input 2 (blank for positive, - for negative). For example, an input of 4'b1000 will display as -8, and input of 4'b0110 will display as <blank display> 6.

HEX0 will display the absolute value of (input1 + input2) in Base10. When pushbutton KEY0 is pressed, the absolute value of (input1 – input2) will be displayed on HEX0. In both cases, use HEX0 to display the result, and use HEX1 to display the sign of the result. For the input combinations that results in arithmetic overflow, display "0F" on HEX1 and HEX0.

Since LEDR[7:0] are not used, turn them off.

Construct a test bench with at least one set of input values and capture outputs in both a text file and a GTKWave screen capture.

Hint: You may need to extend the functionality of your 7 segment decode module created for design block #1. If you do, go back and make the necessary changes to design block #1. When you integrate the multiple blocks, having a single module that works for all blocks will be invaluable.

3. Design block #3

Compare the values entered via SW[7:4] (input1) and SW[3:0] (input2). When the inputs are equal, light LEDR[2]. When input1 > input 2, light LEDR[1]; When input 1 < input, light LEDR[0].

When SW[9:8] = 2'b10, interpret the two input values are unsigned values. When SW[9:8] = 2'b11, interpret the two input values as 2's complement values.

Display the value of input1 using HEX5 (sign which is blank for positive or – for negative), and on HEX4 display the absolute value of the input. Display the value of input 0 using HEX1 (sign which is blank for positive or – for negative), and on HEX0 display the absolute value of the input.

You may use any Verilog constructs you like for this block.

Construct a test bench with at least one set of input values and capture outputs in both a text file and a GTKWave screen capture.

Hint: Verilog defaults to interpreting numbers as unsigned, but does have a signed data type. Google is your friend.

4. Integration of Design blocks

When slide switches [9:8] = 2’b00, the board will function as design block #1. When slide switches [9:8] = 2’b01, the board will function as design block #2. When slide switches [9:8] = 2’b1x, the board will function as design block #3.

You will generate a single programming file that allows the board to demonstrate all three blocks, one at a time, by switching SW[9:8]. You should also indicate the position of SW[9:8] in LEDR[9:8], with the LEDRs off when the switches are down.

It is not necessary to create a testbench for the integrated design.