ECEN3763 – Lab Week 1

Spring 2022

Due Friday January 28th, 11:59pm - 35 Points

Purpose and Summary

The purpose of Lab Week 1 is to verify correct operation of Quartus, Modelsim, and your Terasic DE10-Standard board. A secondary goal is to get those (System)Verilog neurons firing again.

Why is this lab due in 2 weeks?

Notice that this lab is not due until January 28. The lab assignment for Week 2 will make some modifications to this lab, and both Week 1 and Week 2 labs will be due on January 28. This will allow time to deal with tool installation issues, and a little extra time to remember how Quartus works, and to remember some (System)Verilog from years past.

IMPORTANT

This lab was designed to allow you to test your design on your DE10-Standard board. However, since you don't yet have your board, do everything in this lab except test your design on your board. When you submit your lab, I will test your design, and if your design does not work correctly, you will be given one more week to fix your design (you will have a board by that time).

Submission

Your submission will be a single .zip (or .7z or .tar) file uploaded to Canvas for Lab Week 1. The submitted file should contain the following:

a) A Quartus archive of your project (.qar file). You can create this archive file in the Quartus GUI using *Project > Archive Project*. Using the default settings when creating the archive will capture all required files.

b) Three screen captures of your Modelsim simulation waveforms, in focus and at a proper zoom level to convey that demonsrate counter reset, counter direction change, and counter load function. These captures can be JPEG, PNG, PDF, or some common image format.

c) You may choose to include a brief text file describing issues with your final design (if any), an explanation of your simulation (if needed), and anything else you want me to know about your submission.

Grading

This lab will be graded as follows:

Quartus project that successfully compiles and downloads to board: **10 points**
Counter design operating on your board: **15 points**
Modelsim simulation (testbench and screen capture): **10 points**

Lab Instructions

Using Terasic SystemBuilder

Generate a set of Quartus project files for this lab. This lab requires the signals *CLOCK_50, LEDR[9:0], KEY[2:0], and SW[9:0]*. If you cannot run Terasic SystemBuilder for any reason, ask the instructor or SA to create the Quartus project files for you.

Create a simple counter circuit using SystemVerilog with this behavior:

Ten of the counter outputs connect to the 10 LEDs in order to display the current counter value in an easy to view binary format. The counting rate displayed should be in the range of 2 – 10 Hertz. The exact frequency value is not important, but the display should change such that your instructor and SA can see what is happening.

The rightmost pushbutton KEY[0] will hold the counter in reset when pressed.

KEY[1] will force the displayed counter output to the values of the 10 switches SW[9:0] – a load (or preload) function. When KEY[1] is pressed, the counter value will match SW[9:0], and when KEY[1] is released, the counter will resume counting from the preloaded value.

KEY[2] will control the direction of counting. Count up (increment) by default, but when KEY[2] is pressed and held, counting will reverse direction (decrement).

KEY[0] is the highest priority input (reset), KEY[1] is next (load), and KEY[2] is the lowest priority input (direction).

Do not latch the KEY[2:0] pushbuttons.

Write a simple testbench to demonstrate correct counter operation

Run the simulation using Modelsim. Confirm your simulation works correctly when the counter is reset, when the load function is used, and when direction is changed. You should organize your Modelsim waveform display such that a screen capture demonstrates reset operation, load operation, and changes in direction. You may submit more than one screen capture if needed.

Verify correct operation of your counter on your Terasic DE10-Standard board

Additional Guidance for this Lab

a) Test your .qar file prior to submission to Canvas. Quartus creates a new directory when a .qar is opened, eliminating any concern about overwriting anything in your Quartus project. You can test your project archive by double clicking on the .qar file in File Explorer, or in Quartus, *File > Open Project* and select your .qar file.

b) Do not use a clock divider for your counter. I know we did this in ECEN2350, but now that we have a proper simulator, a clock divider is not required.

c) Do not have any spaces in your directory path names for your project. Modelsim is especially sensitive to spaces in path names. In general, avoiding spaces in Windows path names is always a good idea.

c) You will be creating a number of different Quartus projects during the semester. Now would be an excellent time to think about a class directory structure that allows you to manage your projects.

d) Remember that SystemVerilog is case sensitive.

e) Use some of the SystemVerilog syntax discussed in the course lectures. Avoid using reg and wire data types, use logic instead.