Laboratory Exercise 6

Simulation of Verilog Code

This is an introductory exercise that is intended to prepare you for Laboratory Exercises 7 and 8.

1. For Labs 7 and 8 you will be required to design, with Verilog code, your own (simple) processor. You will need to simulate your Verilog code with the ModelSim simulator, using testbenches. To ensure that you know how to use ModelSim and testbenches, perform the steps in the tutorial called "Using the ModelSim-Intel FPGA Simulator with Verilog Testbenches." This tutorial is included as part of the *Design Files* for this exercise. The example ModelSim setup files, and Verilog source code files, that are used in the ModelSim tutorial are included with this exercise. Copy these files onto the computer where you are running ModelSim (your home computer or ECF/DESL).

2. You will implement your simple processor in the FPGA device on the DE1-SoC board, using the Quartus Prime software. To re-familiarize yourself with Quartus Prime, which you have used previously in ECE241, read through the tutorial called "Quartus Prime Introduction Using Verilog Designs". This tutorial is included as part of the *Design Files* for this exercise.

3. If you have purchased a DE1-SoC board for home use, then you will be able to use that board to implement the circuits produced from the Verilog code for your processor. For Labs 7 and 8 you will write some programs that execute on your processor, and some of these programs will make use of I/O devices on the board, such as LED lights, switches, and seven-segment displays. As an alternative to running your processor on an actual DE1-SoC board, you can "implement" it on a *simulated* representation of the board, by using a software application called *DESim*. The *DESim* software is similar to the *fake_fpga* tool that you were given for ECE241. An advantage of DESim is that it is easier to install in comparison to *fake_fpga*, and it comes with some simple-to-understand example (*demos*) Verilog projects. The DESim tool is available from a GitHub repository, which is referenced at the bottom of this page. A brief description of DESim, from its repository, is given below:

   The DESim application provides a graphical user interface (GUI) that represents some of the features of a DE1-SoC board. This GUI serves as a "front end" for the ModelSim simulator. Using the DESim GUI you can invoke both the ModelSim Verilog compiler and simulator. Inputs to the ModelSim simulator can be provided by clicking on features in the DESim GUI, which also shows results produced by the simulator on displays that look like the ones on a DE1-SoC board.

   The DESim application is available on the DESL and ECF computers at the University. Instructions for using this software on these machines can be found in the tutorial "Using DESim on DESL and ECF Computers," which is provided as part of the *Design Files* for this exercise.

   In addition to using DESim on the University computers, you can also install this software for use on your home computer, following the instructions in the document "Installing the DESim Application." To learn how to use DESim, perform the steps in the tutorial entitled "Using the DESim Application with Verilog Code." These documents are included in the *Design Files* for this exercise, and are also available in the DESim GitHub repository: https://github.com/fpgacademy/DESim.