ECE140: Design of Logic Systems I
Laboratory #13: Introduction to FPGA

The purpose of this lab is to get you acquainted with the Digilent Basys FPGA board. You will learn how to make pin assignments on the FPGA, program the board, and run projects on the board using the Verilog code for the half adder from a previous lab session.

1. Sign out a Digilent Basys FPGA board from the Tompkins 3rd floor stock room. There are enough units for each person in the class to have their own.
    a. Verify that your box contains the board, the USB connector, a CD, and an information sheet with a picture of the board on it.
    b. Place the CD into your computer and install the Digilent Adept programming software.

2. On the board, verify that the jumper JP3 is connected over JTAG and not ROM and that switch SW8 is on VUSB and not VEXT. Connect the board via USB to your computer and open Digilent Adept.
    a. Click on the Test tab and then click on Start Test. (Do it if you can see a test tab, some version may not have this option. Therefore, answer Q1 if you can see the test tab.)
    b. *Q1: What happens on the board when you move a switch or press a button? What happens on the computer? (Very short answers, plz! This is just to verify that the board is connected properly and functional.)*
    c. Disconnect the board and set it aside. Close Digilent Adept.

3. Open the half_adder project that was completed during lab 2 weeks ago. Under Tools, click on PlanAhead, then I/O Pin Planning (PlanAhead) – Pre-Synthesis. (If you are prompted to add a UCF file to the project, click Yes.)
    a. In the top right-hand corner of your screen, you will see the package layout. Zoom in on the bottom left-hand corner of the package so that you can read the text in the circles on the package.
    b. In the I/O Ports section of the screen, expand Scalar Ports. You should see your two inputs and two outputs now (they are ports of your half_adder module).
    c. You can assign the appropriate site (the site of switches, LEDs and pushbuttons) for each port of the design module. Just by individually right-clicking on each port and clicking on I/O Port Properties, you will see a input bar for “Site”. Do not forget to save your settings.
    d. Make sure that you have done the following pin assignments:

A	P36
B	P38
Sum	P14
C_out	P15

    e. *While making these assignments, Q2: what differences did you notice in the Package window in the top right-hand corner of your screen?*

    f. *Using the diagram on the sheet that came in the FPGA's box, Q3: what have the inputs been assigned to on the board itself? The outputs? (I'm looking for the names of the switches and LEDs.)*
    g. Click on File -> Save Project. Then close PlanAhead.

4. In Project Navigator, click on Process -> Implement Top Module. Wait for the implementation to complete (observe the Console at the bottom of the screen). Then click on Generate Programming File (Make sure that half_adder.v is highlighted in the small upper left-hand window in Project Navigator or else you will not have that option).

5. Reconnect the FPGA board and open the Digilent Adept programmer.
   On the Config tab of Adept, in line with the FPGA icon, click on Browse.
    a. Navigate to the directory where you saved your half_adder project. Open the half_adder.bit file
    b. Click on Program.

6. The program is now on the board. Make sure you understand exactly what is happening… *then flag me down and provide me with a demonstration and explanation! (This is easy…)*

7. Close out Adept and close the half_adder project in Project Navigator. Disconnect the board from the computer.

8. Create a new Verilog project entitled io_test in Project Navigator. Write a Verilog module that will do the following all in one module:

    a. *When either switch SW5 or switch SW6, but not both of them, is high, LED LD6 turns on.*

    b. *When either or both switches SW3 and SW4 are high, LED LD4 is on.*

    c. *When switch SW1 is high and button BTN1 is pressed, LED LD1 is on. When switch SW1 is low and button BTN1 is pressed, LED LD1 does not turn on.*

   Configure the I/O assignments accordingly (HINT: use the diagram that came in the FPGA's case to find the appropriate pin numbers) and program the FPGA. Demonstrate the above to me for full lab credit. Include the entire module of Verilog code you wrote in your lab report. In one or two sentences for each, explain why you implemented each function the way you did.