EECE 277
FPGA Design

Fall 2014, Dr. William H. Robinson

Laboratory Assignment Two

Due: *Friday, October 10, 2014 at the beginning of class*

OBJECTIVE

The purpose of the assignment is to gain more experience with the EDA tools used in the course (Quartus II and ModelSim-Altera) and the DE2-115 Development and Education Board. In addition, you will also use VHDL to model some designs.

SAMPLE FILES

The software files related to the Altera DE2-115 Development and Education Board are available on OAK under "Labs – DE2-115 Board Information." In addition, your Roth & John textbook has a CD that contains all the VHDL code from the textbook.

COLLABORATION POLICY

Each team is responsible for completing the laboratory. Collaboration among teams is encouraged on the projects in this class, but only under the following conditions. Teams may discuss the laboratory assignment, the general approach taken to solve the problems, and compare results **without** comparing the entire solution. However, copying of code, results, or analyses of results is **NOT** allowed.

LABORATORY DESCRIPTION

You will need to read the following material from your textbook:

- CHAPTER 2 – Introduction to VHDL
- SECTION 4.1 – BCD to Seven-Segment Display Decoder
- CHAPTER 8 – Additional Topics in VHDL

In addition, please refer to the documentation posted in the "DE2-115 Board Information" folder under the "Lab" section on OAK.

For each problem below, you should turn in the schematic or VHDL corresponding to the design as well as the simulation code and/or results that verify the design. Your VHDL code should include appropriate comments including the header file described in your lecture notes. In addition, you should answer any problem-specific questions (e.g., find the delay time). You will also need to demonstrate your solutions to your instructor. Demonstrations should be as efficient as possible. Simulation results can be as simple as loading the wave log file (WLF) from ModelSim-Altera. DE2-115 demonstrations can be as simple as opening the Quartus II programmer and configuring the FPGA. **You will also need to upload all project-related files (including VHDL code) electronically to OAK for full credit.** Place each problem/project in a separate folder. Make a ZIP file of all the folders and upload it into the "file exchange" of your Team Page.

- **Problem 1:** Using Section 4.1 as a guide, implement a Hexadecimal to Seven-Segment Display. Rewrite and compile the synthesizable VHDL model given in the text, translating the PROCESS and CASE statements into a WITH…SELECT statement, and use STD_LOGIC_VECTOR instead of BIT_VECTOR. Use this model as a component in a structural VHDL model with the DE2_115_Top file as the top level. Verify in hardware that your design is correct using 4 switches as the input.

- **Problem 2:** The state machine in Figure 1 is for a serial adder. Create a synthesizable VHDL model for this finite state machine (FSM). All inputs and outputs should be synchronous. **Your FSM model should use two processes, one for combinational logic and one for the registers. Please refer to your lecture notes for an example. This is standard coding practice for state machines and should be followed anytime you implement a state machine.** Create a test bench to verify correct operation with a simulation using the ModelSim-Altera tool. The simulation should exercise all arcs in the state diagram. The two inputs are **a** and **b**, while **s** is the output. Use switches for the inputs and a seven-segment display for the output **s** (i.e., "0" for zero and "1" for one). Use a switch for the **Reset** signal. Use a push button for the **Clock** signal. **G** and **H** are the two states. Use enumerated types for the states in your VHDL model. For state **G**, illuminate a green LED and for state **H**, illuminate a red LED. Use the Classic Timing Analyzer to determine the maximum clock frequency. Download your design to the DE2-115 Board for testing.

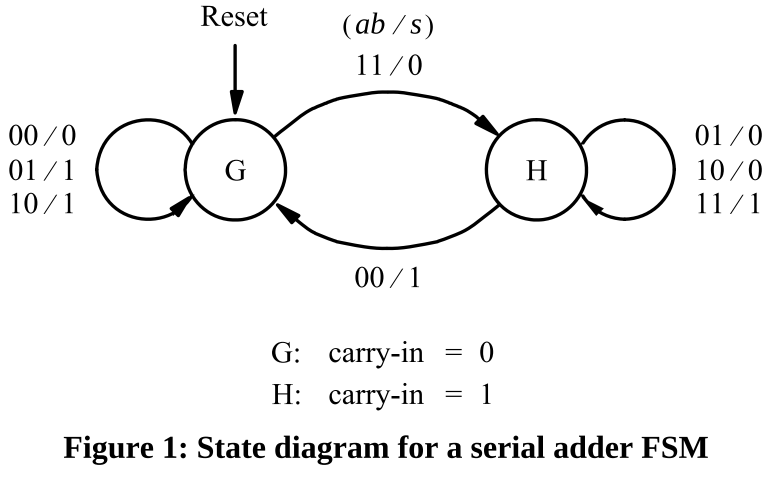

Figure 1: State diagram for a serial adder FSM

- **Problem 3:** Design a circuit for the DE2-115 board that scrolls the word "HELLO" in ticker-tape fashion on the eight 7-segment displays HEX7 − HEX0. The letters should move from right to left each time you apply a manual clock pulse to the circuit. After the word "HELLO" scrolls off the left side of the displays, it then starts again on the right side. Design your circuit by using eight 7-bit registers connected in a queue-like fashion, such that the outputs of the first register feed the inputs of the second, the second feeds the third, and so on. This type of connection between registers is often called a pipeline. Each register's outputs should directly drive the seven segments of one display. You are to design a finite state machine that controls the pipeline in two ways:

  1. For the first eight clock pulses after the system is reset, the FSM inserts the correct characters (H,E,L,L,0, , ,) into the first of the 7-bit registers in the pipeline.
  2. After step 1 is complete, the FSM configures the pipeline into a loop that connects the last register back to the first one, so that the letters continue to scroll indefinitely.

  Write VHDL code for the ticker-tape circuit and create a Quartus II project for your design. Use KEY[0] on the DE2-115 board to clock the FSM and pipeline registers and use SW[0] as a synchronous active-low reset input. Your FSM model should use two processes, one for combinational logic and one for the registers. Compile your VHDL code, download it onto the DE2-115 board and test the circuit.

- **Problem 4:** Create a synthesizable VHDL model to implement a 3-digit BCD counter. Display the contents of the counter on the seven-segment displays, HEX2 − HEX0. Derive a control signal from the 50-MHz clock signal provided on the Altera DE2-115 board to increment BCD counter at one-second intervals. Use one switch to enable the count, and another switch to reset the count to zero. Download your design to the DE2-115 Board for testing.

- **Problem 5:** Complete Part I, Part II, Part III, and Part IV from Laboratory Exercise 8 Memory Blocks. This lab is available on OAK (lab8_VHDL.pdf).

TEAM RESPONSIBILITY

This is a team assignment. Divide the assignment up such that each team member contributes equally to the problems described above. It is the team's responsibility to design and run adequate test cases. You may utilize the file exchange in OAK under "Groups." There should be only one (1) submission of the results/solutions for each team.

TEAM MEMBER EVALUATION

You will evaluate your teammate(s) on the quality and quantity of their contributions and cooperation. This team evaluation will comprise 10% of an individual's grade for the programming assignment. Each team member **MUST** complete the teammate evaluation form and submit it electronically before class. Use the link provided in **Session 23 - Friday, October 10, 2014** (under the "Lectures" menu item). **DO NOT USE THE FILE EXCHANGE FOR YOUR EVALUATION FORM!!!** Failure to submit your evaluation will result in a **10% penalty** for your individual grade.