ENGG2410: Digital Design

Lab 6: Sequential Logic Design "Sequence Recognizer Circuit" via *VHDL*

Shawki Areibi
School of Engineering, University of Guelph
Fall 2023

Start Date Week #9 2023
Duration: 1 Week
Report Due Date Week #10, In the Drop Box

1 Objectives:

At this stage you should be familiar with design entry using both (i) Schematics (ii) VHDL using the Xilinx ISE Foundation tools.

The purpose of this lab is to teach you the basic operation of sequential logic (circuits with memory), by designing them using VHDL. In this lab you will become familiar with state diagrams, state tables and sequential circuit design by creating and demonstrating a "Sequence Recognizer" using **Schematic Capture**. Students should also demonstrate their understanding of how behavioral **VHDL** can be used to implement sequential circuits.

2 Sequence Recognizer

In class we talked about circuits that recognize the occurrence of a particular sequence of bits, regardless of where it occurs in a longer sequence (Sequence Recognizer). This sequence recognizing circuit is to

Input	0	1	1	1	1	0	1	1	0	1	0	1	0	0
Output	0	0	0	0	0	0	0	1	0	0	1	0	0	0

Figure 1: An Example of A Sequence Recognizer.

have one input **X** and one output **Z**. In addition, it has direct resets on its Flip-Flops to initialize the

state of the circuit to all zeros. The circuit is to recognize the occurrence of the sequence **1101** on X by making Z equal to 1 when the previous three inputs to the circuit were 110 and current input is a 1. Otherwise, Z equals 0. A key factor in the formulation of any state diagram is to recognize that states are used to "remember" something about the history of past inputs. One issue that arises in the formulation of any state diagram is whether excess states have been used. If excess states are present, then it is desirable to combine states into the fewest number needed, using state minimization procedures. Figure 1 shows an example of an input/output sequence. Check your lecture notes on how to design a state diagram for both a Mealy and Moore machine.

2.1 Preparation

Design the sequence recognizer circuit **using schematic capture design entry method** based on a **Moore Machine**. You should also demonstrate your understanding to the lab instructor and TA of how a sequence recognizer circuit can be implemented **using the VHDL design entry method**. Notice that the VHDL code provided on the webpage has a slight problem! When the system reaches the state that recognizes the occurrence of the sequence the output Z will not be held at '1'. So you need to modify the code provided to you such that when the three previous inputs to the circuit were 110 and current input is a '1' then and only then the output Z = '1'.

Your preparation should consist of:

1. The state diagram (draw by hand) of a Moore Machine for the sequence recognizer.
2. The state transition table.
3. Circuit Diagram (Flip Flops and associated combinational logic).
4. **Map** your design using **Schematic Capture** on the FPGA board.
5. **Demonstrate** the correctness of your design by using the FPGA board and/or simulation.

Show your partial or complete **Behavioral VHDL code** to the lab instructor and TA that you will include in the report.

2.2 Important: Things to Remember

1. Use LEDs at the output of each Flip-Flop of the sequence recognizer built based on VHDL code (to understand the different state transitions).
2. When mapping your design onto the NEXYS 3 Board use the following Push Buttons: either 'BTN-D' or 'BTN-R' as your manual clock.
3. Since Push Buttons cause bouncing, use the VHDL code available on the student resource section to debounce the Push Button to be used as the manual clock.
4. Read the debouncing VHDL code carefully to understand how to use it in this Lab.

3 Requirements

1. **Read** the tutorial on the web page titled "Tutorial on Using Xilinx ISE Foundation Design Tools: Modeling State Machines".
2. **Implement and test** the sequence recognizer you designed in the preparation **using Schematic Capture**.
3. **Show** your DEMO to either the Lab Instructor or Teaching Assistant.
4. **Prepare** your VHDL code that implements the sequence recognizer. This code should be included in your final report.

4 Deliverables

1. **Lab Preparation File**:
   - Name your file as follows: ENG2410_F23_LAB6_Section(Mon,Tue,Thu,Fri)_Group#_Preparation.pdf
   - The preparation file can be hand written or typed. But has to be neat.
   - The preparation file should consist of the State Diagram, State Table, Input Equations and Circuit Diagram.
   - Only one submission is allowed. Multiple submissions will not be marked.
2. **Lab Final Report File**:
   - Name your file as follows: ENG2410_F23_LAB6_Section(Mon,Tue,Thu,Fri)_Group#_FinalReport.pdf
   - Check the format of the final report below in the next section.
   - Keep the first page only for the title page.
   - Only one submission is allowed. Multiple submissions will not be marked.
3. **Zipped Project File**:
   - Name your file as follows: ENG2410_F23_LAB6_Section(Mon,Tue,Thu,Fri)_Group#_Project.zip
   - The project file should have a README file that explains the switches and LEDs you used in your design.
4. **DEMO**:
   - You will need to demonstrate to the Teaching Assistant that your design fulfills the requirements.
   - The DEMO will take place during the LAB hours and not outside these hours.

5 Report

Below is the general format of the report required:

1. Title Page:
   - Course #, and Date
   - Lab # and name of experiment
   - Your Group #, and Names
2. Start a new page and explain how you implemented your design by providing the following:
   (a) **Problem Statement**,
      i. Briefly describe the problem solved in the lab.
   (b) **Assumptions and Constraints**.
      i. Constraints could be for example using only NAND gates or NOR gates in your design.
   (c) **How you solved the Problem**,
      i. State Diagram.
      ii. State Table.
      iii. Flip Flop Input Equations.
      iv. Circuit Diagram.
   (d) **VHDL Code**,
      i. List the VHDL code that you have written.
      ii. Make sure that the code has comments and documented.
   (e) **System Overview & Justification of Design**
      i. Give an overview of the system to be designed.
      ii. Briefly explain how the system works and reasons behind the design.
3. **System Design**
   (a) State Diagram, State Table.
   (b) K-Maps used for optimization.
   (c) Flip Flop input equations.
   (d) Overall block diagram.
4. **Circuit Diagram**
   (a) Brief explanation of the hardware.
   (b) The schematic you produced using Xilinx ISE tools.
5. **Error Analysis**
   (a) Printed **Simulation** waveform data, showing that the simulation coincides with the expected state diagram of your design.
   (b) Include the VHDL **Test Bench** in an Appendix.
   (c) Describe any problems with the system.
   (d) If no problems in the final system, describe problems/errors encountered during the development and how they were resolved.