Lab 1: Sequential Circuits and FSM

1 Overview

1.1 Introduction

In this lab we will implement a sequential logic circuit, i.e. a circuit which includes different states. You will design a Finite State Machine (FSM) to track the number of cars entering and leaving a carpark. Recall the two types of FSMs: Mealy and Moore FSM. You will revise the characteristics of both types of machine and decide the appropriate one for your design.

You will also revise the top-down approach to designing your solution in Vivado. The FSM solution involves multiple modules which implement different functions and you will have to decide the appropriate hierarchy for your final design.

1.2 Learning Outcomes

The purpose of this lab is to revise the concepts of combinational and sequential logic. On completing this practical you will be able to:

- Recognise combinational and sequential logic
- Apply a top-down approach to FSM design to produce clear, structured code
- Use the Peripherals of the PYNQ-Z2 board to help demo your design

1.3 Definitions

- Hardware Description Language (HDL): HDLs are used to describe the behaviour or structure of digital circuits. They provide a formal description of a circuit which allows for automated analysis and simulated testing of the circuit. Verilog and VHDL are the most widely used HDLs and in this course, we will be using Verilog HDL.

1.4 Resources

- Refresher guide for Vivado: https://reference.digilentinc.com/vivado/getting_started/start

- Vivado Design Suite User Guide: https://china.xilinx.com/content/dam/xilinx/support/documents/sw_manuals/xilinx2019_1/ug908-vivado-programming-debugging.pdf

2 Problem Description

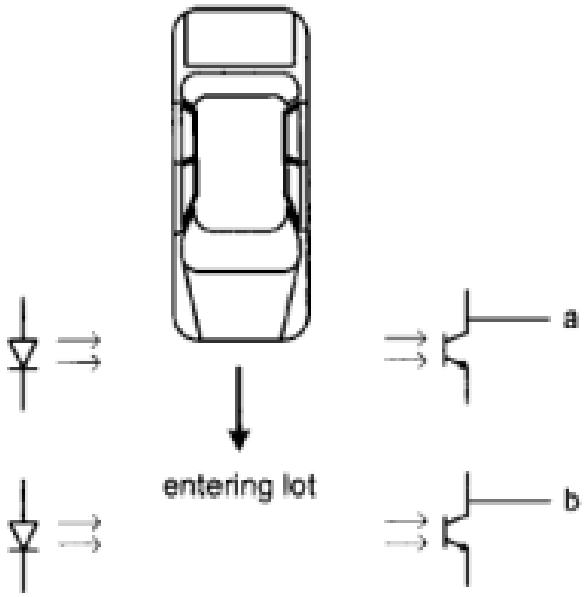

Figure 5.11 Conceptual diagram of gate sensors.

Figure 1. Example Taken from Chu p. 136

Consider a carpark with a single entry and exit gate. Two pairs of photo sensors are used to monitor the activity of cars, as shown in Figure 5.11 of the Chu book, reproduced in Figure 1 above. When an object is between the photo transmitter and the photo receiver, the light is blocked and the corresponding output is asserted to 1. By monitoring the inputs from two sensors, we can determine whether a car is entering or exiting.

For example, the following sequence indicates that a car **enters** the carpark:

• Initially, both sensors are unblocked (input from sensors a and b is "00").

• Sensor a is blocked (input "10").

• Both sensors are blocked (input "11").

• Sensor a is unblocked (input “01”).

• Both sensors become unblocked (input “00”).

You need to design a carpark occupancy counter for a carpark of 15 spaces.

---

3 Instructions

3.1 FSM Design

I. Design an FSM with four input signals; clk, reset, a and b, and two output signals enter and exit. The enter and exit signals should be asserted for one clock cycle when a car enters and one clock cycle when a car exits the carpark, respectively.

II. Draw a state diagram to describe the desired FSM behaviour, www.draw.io is a useful tool for this. (Hand-drawn diagrams are also acceptable).

III. Write the RTL code to describe the FSM. Register all outputs and apply any naming conventions as required.

IV. Construct a testbench to test this FSM.

3.2 Counter

I. Design a counter with two control signals, inc and dec, which increment and decrement the counter when asserted.

II. Draw a block diagram of the desired module. (Use www.draw.io or similar, or hand-draw).

III. Write the RTL code.

IV. Construct a testbench to test this counter.

3.3 Top Level

I. Create a top module and instantiate your FSM and counter.

II. Connect their signals such that an enter signal from the FSM will cause the count to increment, and exit will cause a decrement.

III. Use the provided debouncer.sv module to implement sensor inputs a and b as push buttons, use another push button to provide a reset.

IV. Use the board's LEDs to display the current car occupancy count. You will need to modify the PYNQ constraints file (.xdc) for the PYNQ-Z2 board to connect the necessary board interfaces to the top level module's ports.

4 Submission

Please submit a **brief** lab report containing:

- A block diagram of your FSM design.
- Briefly describe the difference between a Mealy and Moore FSM.
- Did you use a Mealy or Moore FSM in your Design? Why?

Please submit your zipped project folder using your name and lab1 as the file name (e.g. AWalsh_lab1).