EC605: Computer Engineering Fundamentals Fall 2017

Lab 3: Finite State Machines

Sequential Logic

Goals

- Introduction to sequential logic Verilog.
- Introduction to state machine design in Verilog.
- Review of utilizing clock dividers for Seven-Segment Display.

Overview

In this lab you will design a counter, and a state machine that will light a sequence of LEDs in response to the counter wrap-around (i.e. reaching zero).

Tasks

Task 1: 8-bit Counter

1. Implement an 8-bit bi-directional counter with the following inputs, outputs and specifications:
   - Inputs: clk, rst, direction, 8-bit maximum
   - Output: 8-bit count
   - The rst input resets the counter to zero.
   - The direction input determines if the counter counts up (direction = 1) or down (direction = 0).
   - The maximum input determines the maximum value that the counter may reach.
   - As an up counter, the counter should increment by one on every positive clock edge until it reaches the upper limit, at which point it wraps around to zero.
   - As a down counter, the counter should decrement by one on every positive clock edge until it reaches zero, at which point it wraps around to the upper limit.

   Note: The output of the counter should also be stored in a register.

2. Implement a testbench and simulate your design. Screenshot the waveform and save it to the PDF for submission.

Task 2: Clock Divider

We would like to utilize the clock provided in the Xilinx FPGA board to control our circuit. However, the frequency of this clock is too fast when used as the input clock for the seven-segment display. A clock divider takes an input clock of a given frequency and generates an output clock of a different frequency.

1. Implement a clock divider that receives clk and rst as inputs, and outputs a clock with a frequency of 1Hz. The input clock frequency is 100MHz.

2. Implement a testbench and simulate your design. Screenshot the waveform and save it to the PDF for submission.

Task 3: Wrap-around LEDs

1. Draw a state diagram of your state machine that has the following inputs, outputs and specifications:
   - Inputs: clock and reset, two switches (switch0 and switch1), and the value of an 8-bit counter
   - Outputs: Four 1-bit LED outputs (or a single 4-bit LED output)
   - The state machine operates as follows:
     - On reset, the state machine moves to state 0, the initial state.
     - From the initial state, if switch0 is on, the state machine waits for a counter to wrap-around (reach zero), and turns on LED 0. For the next wrap-around it turns on LED 1 only, then LED 2, and finally LED 3, then returns to the initial state.
     - From the initial state, if switch1 is on, the state machine waits for a counter to wrap-around (reach zero), and turns on LED 3. For the next wrap-around it turns on LED 2 only, then LED 1, and finally LED 0, then returns to the initial state.

2. In a new project, implement the above state machine. Your project should include the counter and clock divider from the previous tasks.
   - The clock input of the state machine is the 1 Hz, which is the output of the clock divider.
   - Remember: the state machine receives two switches and the value of your counter as input, and generates 4 LED outputs (or a single 4-bit LED output).

   Note: Refer to lecture notes for a Verilog state machine implementation example.

3. Implement a testbench and simulate your design. Screenshot the waveform and save it to the PDF for submission.

4. The Top module of the project should contain the following inputs and outputs which will be connected to the board:
   - Inputs:
     - clk – 100 Mhz input clock from board
     - rst – push button which resets state machine elements
     - switches – two switches which when pressed control the direction
     - switches – eight switches to represent the maximum value that the counter can reach (8 bits)

   - Outputs:
     - LEDs - Four LEDs used for State Machine

Deliverables

1. Submit your Verilog code and testbench for each task.
2. Submit a pdf with the following:
    - waveform and the schematics for each task
    - description of the tests done for each task
    - your state diagram from task 3

Sign-up to demo your design on the Xilinx FPGA board to a TA. Come prepared to answer questions on your design.