ELEC 326 Lab 1

Calculator

Objectives

Learn and work in Vivado and design a simple calculator in Verilog. Learn how FPGA programming works and how to interact with Input and Output pins through the development board.

Inputs/Outputs

Inputs to the Calculator

- CLK – the onboard 50Mhz clock
  - This is needed for the 7---segment display, but will already be hooked up correctly in the provided Verilog
- SW[7:0] – the eight board switches
- BTN[3:0] –four board push buttons

Outputs of the Calculator

- LED[7:0] – the eight board LEDs
- SEG[6:0], DP, AN[3:0] – the seven segment display
  - These will already be connected in the provided Verilog

Step 0: Nexys 4 DDR

Download and read the Nexys 4 DDRReference Manual from FPGA Labs section on the course Canvas page. Read closely chapter 10 on I/O and how Seven Segment Displays work. Describe your understanding as part of your lab report.

Step 1: Provided files

Download lab1.v and Nexys4DDR_Master.xdc from the course webpage.

Step 2: Seven Segment Display

In lab1.v, implement the segmentFormatter module.

The seven-segment display is implemented with two Verilog modules in the provided file:

- module sevenSegDisplay
  - This module is completed, and does not require any editing. It contains the sequential logic for switching between the four digits of the seven-segment display.
- module segmentFormatter
  - This module is currently empty, and will need to be completed.
  - input [3:0] num_pi
    - A four bit number to be displayed
  - output [6:0] disp_po

        - The seven bits of the seven-segment display digit
        - The mapping for the bits of disp_po is given in comments above the module, but you will have to implement the logic for each bit, based on the input num_pi
        - disp_po is active low

Hint: To test this step before you move on, assign the switch bits directly to the ‘display_num’ wire in the top ‘calculator’ module, e.g.:

```
assign display_num = {8’h00, SW};
```

Note that if you do this, display_num will need to be changed from a ‘reg’ to a ‘wire’ for this statement. You use this line to test your segmentFormatter code. You will need to remove it before proceeding to Step 2.

Step 3: Calculator

In lab1.v implement the calculator module.

The calculator should take the input SW[7:0] bits and perform one of several calculations described below, depending on which push buttons are pressed. The output should be assigned to display_num, which drives the display. Note that depending on your implementation, display_num may need to be changed to a ‘wire’ type. In all cases, prepend the output with enough zeros to make the full 16---bit display_num.

- NO CALC NUMBER
    - If no buttons are pressed, display SW[7:0]. No additional calculation is required.
- ADDITION
    - If the first button is pressed, display the sum of SW[7:4] and SW[3:0]
- MULTIPLICATION
    - If the second button is pressed, display the product of SW[7:4] and SW[3:0]
- XOR
    - If the third button is pressed, display the bit---wise XOR of SW[7:4] and SW[3:0]
- DIVISION
    - If the fourth button is pressed, display the result of integer division SW[3:2]/SW[1:0]
        - You cannot use the division operator in Verilog, as it could result in a divide by zero when SW[1:0] is 2’h0
        - The result of this division is a two-bit number
        - You will need to write the logical relationship between the two bits of output and the four bits of input. Use a Karnaugh map to simplify the equations
        - In your Karnaugh map, set the result of division for the case that SW[1:0] is zero to “don’t care”
- COUNTER
    - If more than one button is pressed, display the counter wire, which is an output of the ‘increment’ module in the Verilog provided

Step 4:

For the Synthesized Design, capture screenshots of the Device layout and Schematics (including the RTL Schematic and Synthesis Schematic). **Document these in your lab report and what significance the RTL Schematic has.**

Step 5:

Demonstrate your working implementation to one of the lab graders during staffed lab times.

Due Date --- Lab Demonstration and Code: Friday October 5, 2018 by 5:00pm. Note projects need to be demonstrated before the due date during scheduled hours. Upload the code and note the group members and lab grader in the Canvas submission notes. Please work in self---selected groups of 2 (or maximum 3) students. Only one student needs to upload the Verilog and the same student must also upload the report.

Due Date --- Lab Report: Sunday October 7, 2018 by 11:59pm.

LAB 1 – Report: See the example report on Canvas. Submit as a PDF by the same student that uploaded the code. Include figures and descriptions / explanations on the output schematics and the synthesized design. Also, briefly describe how the seven---segment display works in general with the development board and in relation to the project.