# ECE 2700 Lab 2

# Due the week of 01/27 at the beginning of your registered lab session (100 points)

# Objectives

To learn how to build, simulate, and implement simple combinational circuits in Verilog/Xilinx.

# 1 Pre-Lab Preparation

- 1. Construct truth tables for functions G and H in Problem 2.51 from the main textbook.
- 2. Write Boolean equations for the digital system described in Problem 2.21 from the main textbook. Draw the corresponding circuits.

Recall that you will not get the pre-lab credit if the prep work is not finished prior to your assigned lab time.

Note that it may be useful to bring Lab 1 instructions, Taylor's Verilog tutorial, and the secondary text to lab for reference. Pay close attention to Chapter 6, which is a Verilog mini-reference. Also, the example on page 34 shows how to use the **built-in** logic gates in Verilog.

#### 2 Overview

In this lab, you will implement some simple combinational digital systems in Verilog, simulate your designs, and implement one of the systems on the Basys board.

#### 3 Part I

You will use simulations to determine if the two circuits in Problem 2.51 are equivalent. Create a new directory called Lab2 inside the ECE2700 directory you created last week. Start Xilinx ISE and create a new project called CircuitsComparison with 3 new files: circuit1.v, circuit2.v, and testbench.v. Make sure that you specify the files as Verilog files.

Implement the circuit G in circuit1.v and the circuit H in circuit2.v using built-in gates. Inside testbench.v, put in all possible test cases. Hint: there are 8 possible combinations.

Check for syntax errors and simulate your circuits. Verify the outputs and show the results to your TA. Are the circuits equivalent?

## 4 Part II

In the second part of the lab, you will implement the digital system described in Problem 2.21 and load it on the Basys board.

Create a new project titled disco\_system. You will want to create two new files to add to this project: ds\_system.v, which will contain your Verilog code and disco\_test.v, which will contain your test cases.

Inside the file ds\_system.v, write out the Verilog code to implement the disco system described in Problem 2.21 using built-in logic gates. Once you are done, write out all the possible test cases to disco\_test.v. Each test vector should be simulated for 10 ns.

Check for any syntax errors and simulate your system. Show your results to the TA.

Create a UCF file. Locate SWO and SW1 on the Basys board. SWO and SW1 will correspond to the inputs S (sound sensor) and M (motion sensor), respectively. LDO and LD1 will correspond to the outputs L (light) and B (ball), respectively. Perform appropriate pin assignments in the UCF file and take the steps necessary to load your design on the Basys board. Show your working design to the TA.

### 5 TA Checkoff

- (10 points) Complete pre-lab work prior to start of the lab.
- (30 points) Correct implementation of the two circuits (including exhaustive test cases).
- (30 points) Correct implementation of the disco system (including exhaustive test cases).
- (30 points) Disco system correctly working on the Basys board.

Important: Please upload your .v files and .ucf file on Canvas. Failure to do so will result in a zero for this assignment.