# CS 152A Introductory Digital Lab Design

**Lab 4 Creative Project: Connect Four**

# Date: 6/12/15

# Grade: \_\_\_\_\_\_\_

# TA Name: Behnam Shabazi

# Name 1: Alexander Wang

# SID: 103889014

# Name 2: Alan Kha

# SID: 904030522

# Introduction

For our final project, we decided to implement the game Connect Four using an FPGA. Connect Four is a board game played by two people. The game is played on a vertical 8x8 game board, with checker pieces of two different colors. Players take turns dropping pieces down vertical columns of the board. Since the board is vertical, pieces automatically drop to the lowest possible position. The game ends when there are four in a row, any direction (vertical, horizontal, or diagonal) of pieces on the board.

# Design Details

Design for this project was done mostly on the fly, building modules from the bottom up as necessary.

The first thing implemented was the VGA. We used a core-generated 50 Mhz clock to set the horizontal sync and vertical sync. Next, we built the game controller module. This controlled the game, and set each pixel to output whatever color we wanted it to. Inside the game controller, our game state was stored using two 8x8 arrays. That created the 64 game squares, by using one array to indicate whether a piece was presence, and the other to determine the color of the piece. In order to control the switches, we used the debouncer module from Lab 3, to activate the buttons. We also took the priority encoder from Lab 2 to find where the next piece goes in each column (the highest available position without a piece currently). The game controller was then used to draw the board on pieces using the VGA module.

# Simulation and Testbench

Most of our testing was done directly with the FPGA. The VGA in particular, was tested by trying to implement the code, and then programming the FPGA. If the monitor that the VGA was plugged into changed color, then the code was a success.

Simulation was used to test the clocks used for the project. We used ISIM in order to make sure the clocks were working as intended.

For most of our testing after that, we would just write code, and program the FPGA with it, to check if it worked. Since synthesizing takes a long time, we would try to do as much as possible before testing it with the FPGA.

We did not get to use all of the extreme test cases since we did not have time to complete it. However, we would have tested all the possible edge cases, including tie games, full boards, full columns, etc.

# Conclusion

The biggest problem we had was not finishing the project for various reasons.

The largest issue we had was figuring out how to get VGA to work. We spent over three lab sessions just to get the module working (not even drawing anything important with it yet). For the future, it would be nice to have a VGA tutorial or sample code for us to use, because we could not proceed until we got it to work, and by the time we did, we were extremely far behind.

Another issue we had was with the debouncers. For some reason, because we were using a core-gen 50 Mhz clock, we were unable to create a clock signal slow enough for proper debouncing. This resulted in a single button press that was registered multiple times, which prevented us from being able to properly play the game, despite the logic working correctly.

By the end, we did not have a completed project. We only had the board implemented, and we could drop pieces. Most of the game rules that we planned to implement went unfinished, since we could not do any of it without the rest of the lab complete.

# Individual Contribution

We both contributed to the lab at all parts. However, the majority of the design work was done by Alex, with Alan helping him. For the Verilog code, Alan did most of the coding work, with Alex on the side bouncing ideas and determining the math needed to draw pixels. Finally, the report was mostly written by Alex, with Alan doing most of the editing. Overall, the work split was 60/40, with Alan doing more.