**EEL4740 – Embedded Systems**

Project Title: Christmas Lights



Florida International University

**Project Objectives:**

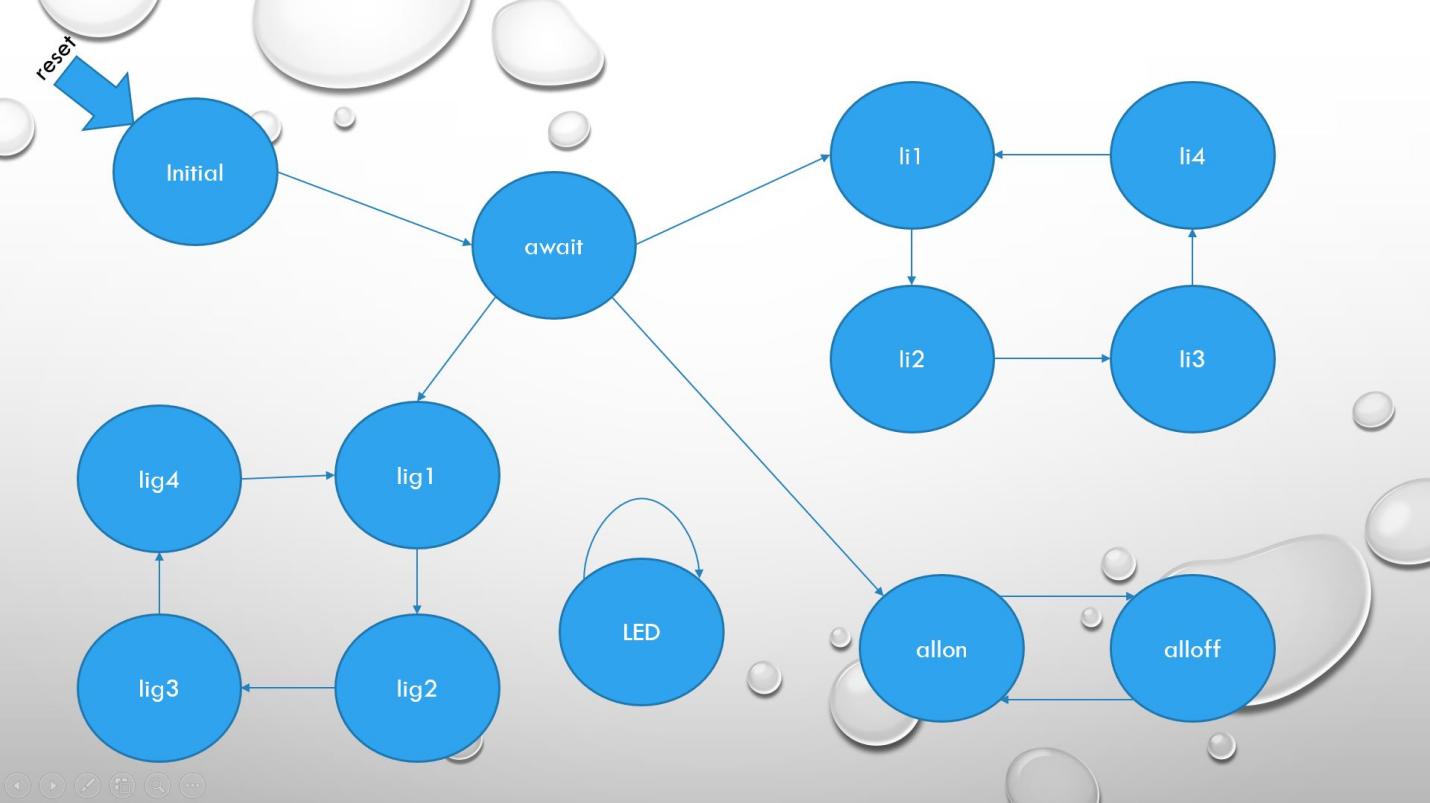
The project is expected to flash different color lights in two different modes and a static/on and off mode. This will be inputted by the user with individual buttons on the bread board. The row of lights is expected to light up in sequences depending on which state the machine is. Different rows are controlled by different states. An indicator LED will flash on the FPGA board to indicate the code is working and the parts that make up the project should be working. The purpose of this project is to understand how lights can be controlled by a case statement, a slowed down clock, and states in a state machine. The project is useful for many real life uses such as Christmas lights, ambulance/police lights, car blinkers, landing strip lights, flashing crosswalk signs, traffic lights, etc.

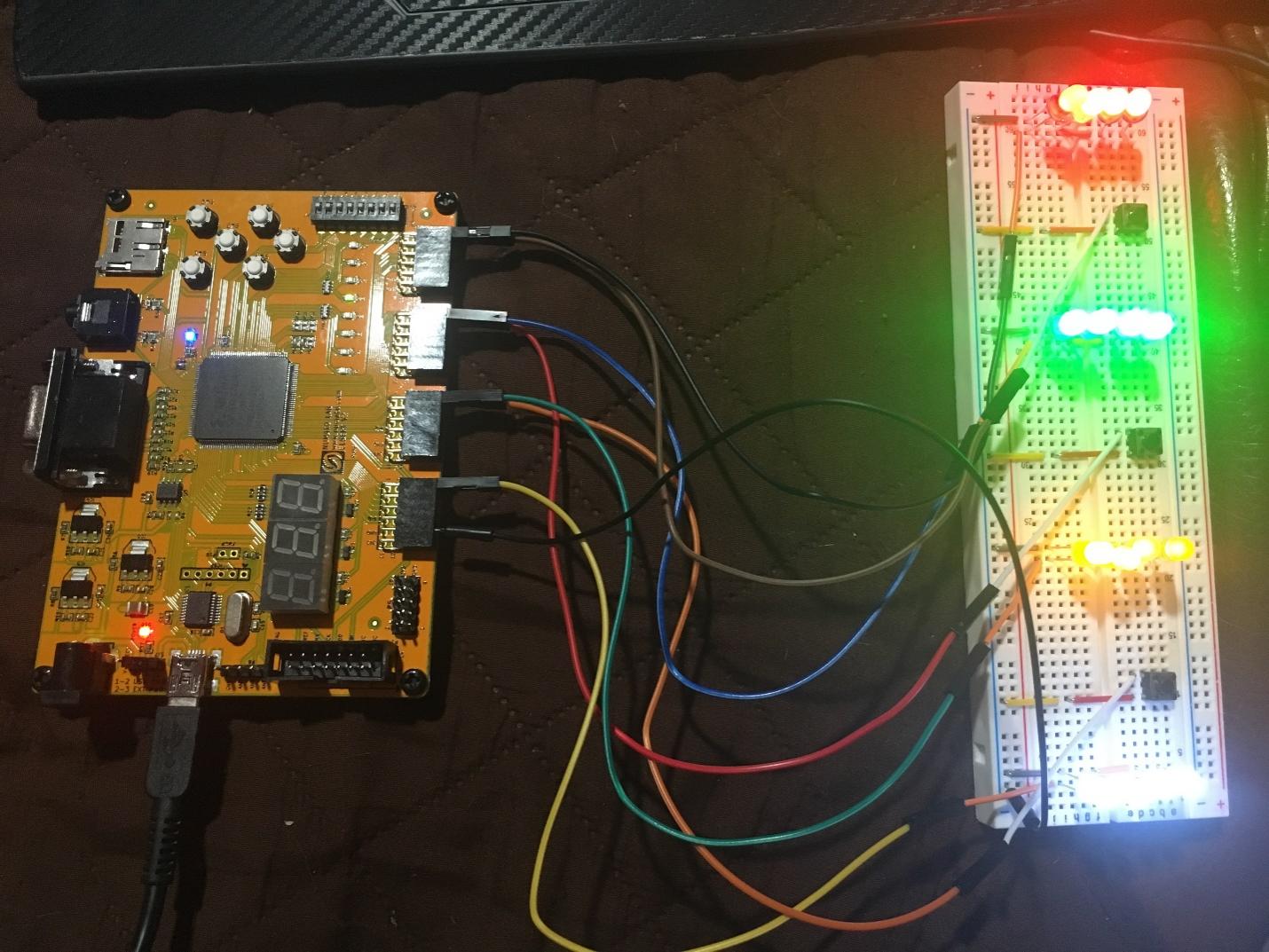
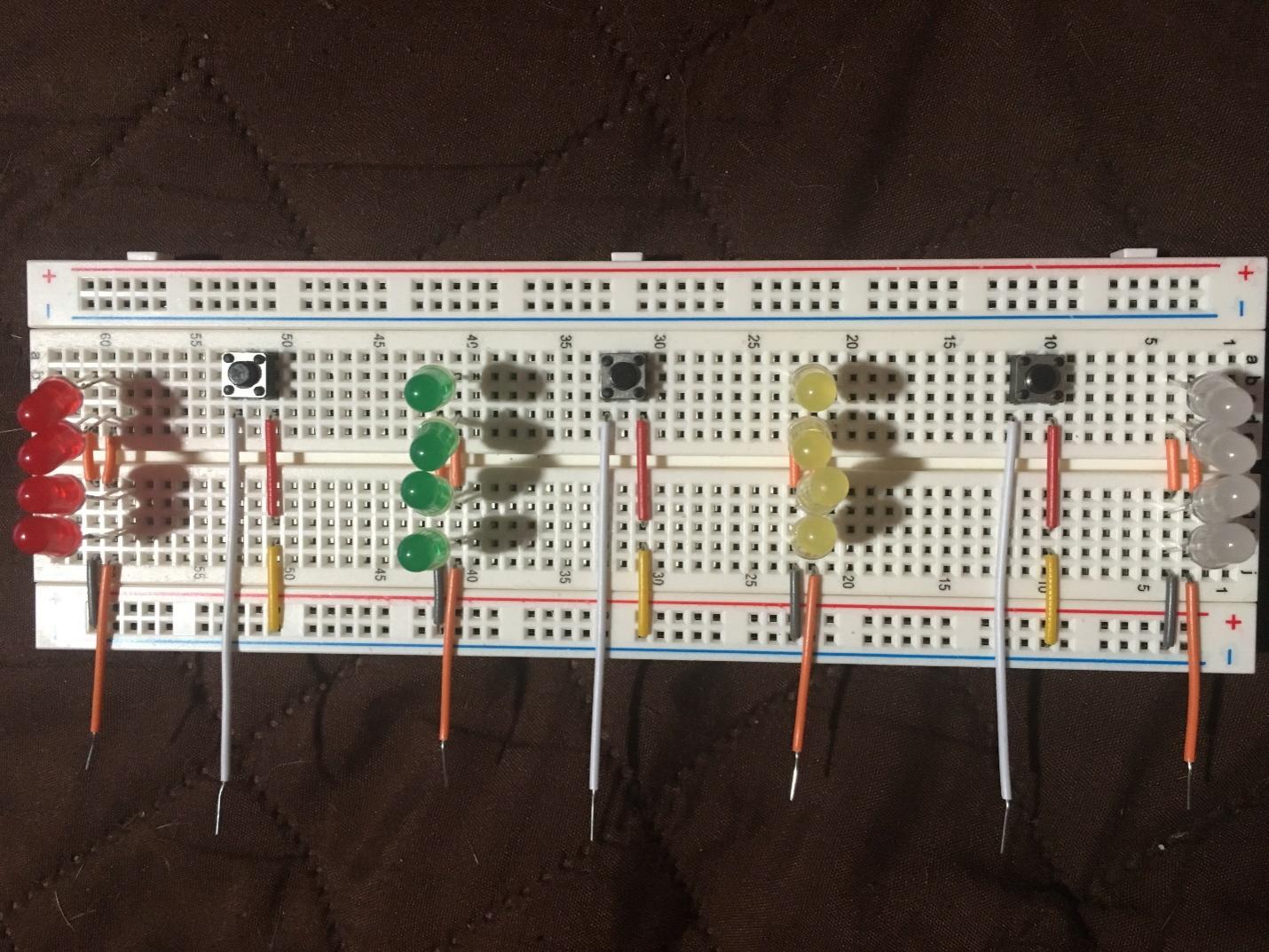
**Components used**

The components used are:

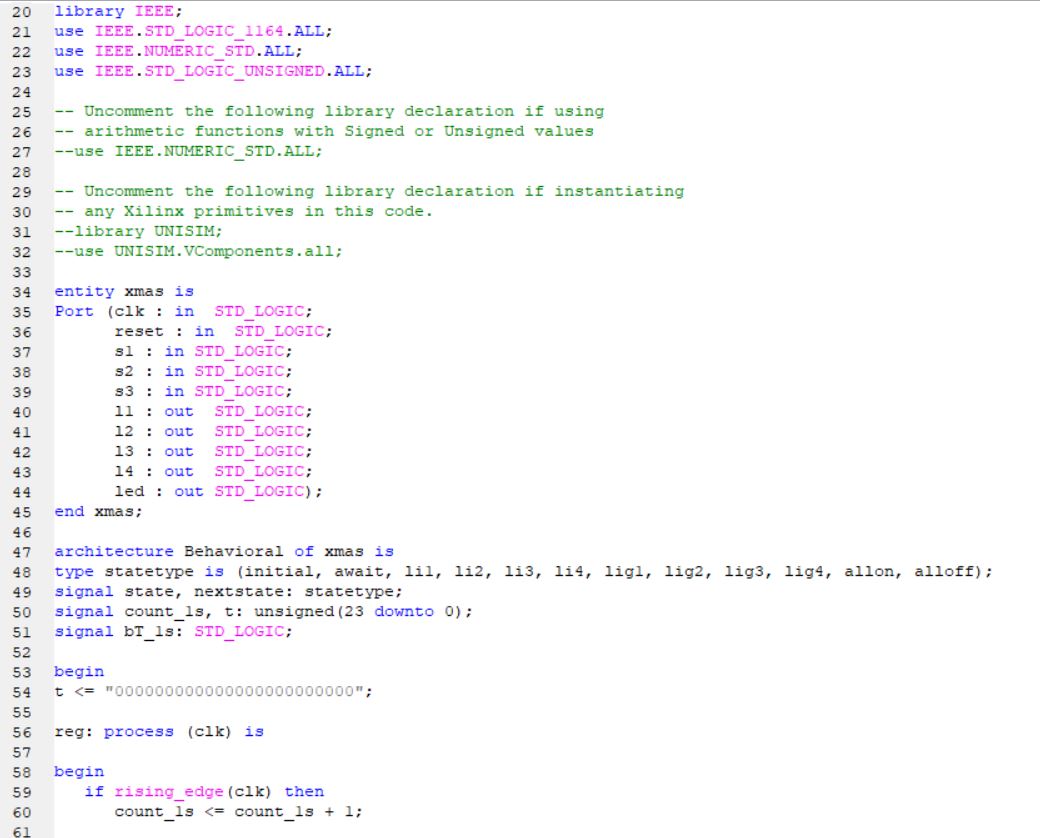
* 1 FPGA board
* 1 USB type A to Mini B 5-pin cable
* 1 laptop
* 1 breadboard
* 8 solderless wires
* 16 Solderless LEDs: 4 yellow, 4 green, 4 red, 4 white
* 3 push buttons

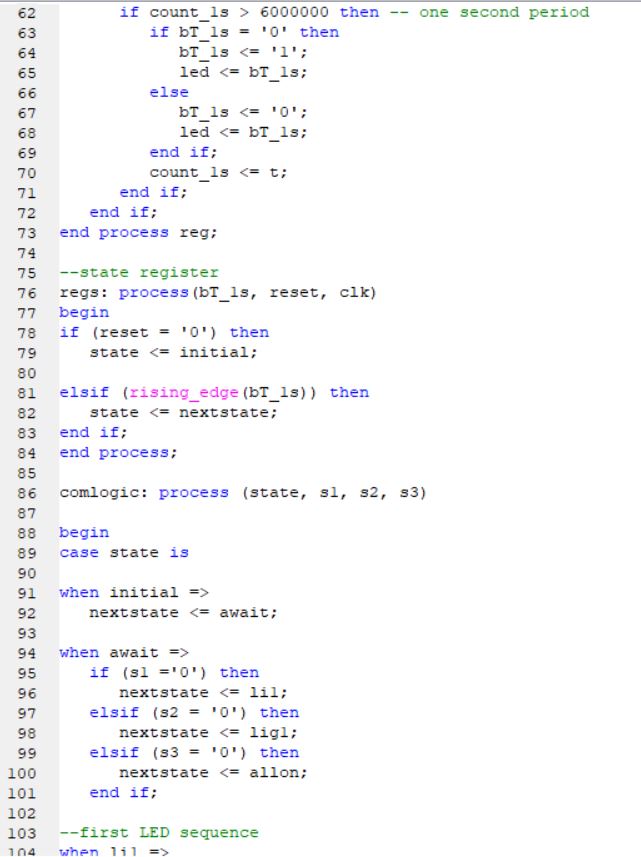
**Picture or Schematic of the System**

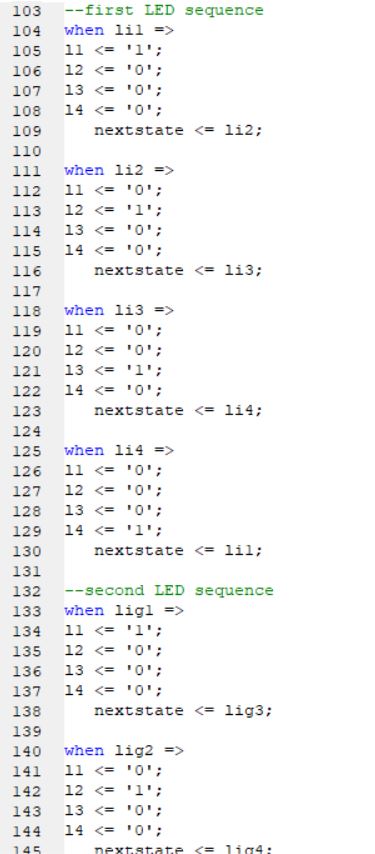


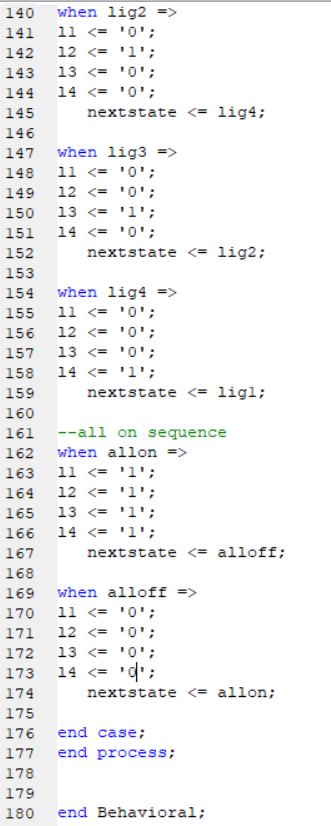


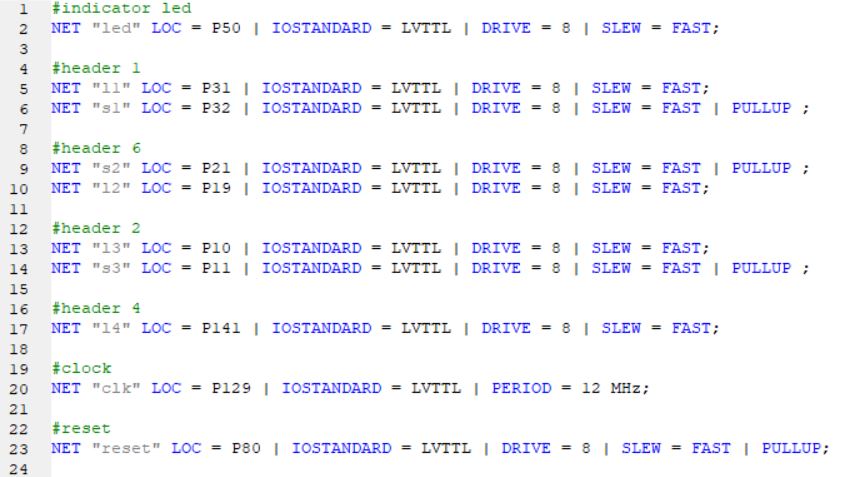
**VHDL Code**

****

****

****

****

****

**Troubleshooting**

This project had multiple problems while creating and implementing the code. The first issue encountered was an issue with the clock. I was unable to correctly slow down the clock to an acceptable rate. The LEDs would either blink too fast or blink to slow. Finding the correct middle was a large difficulty due to an error in code. The code had to be adjusted to > 6000000 for a one second period.

The second issue was a delay from when the button is pressed to when the lights begin to flash. I resolved this by creating less states in the code causing faster switches between states with the clock but that was only a temporary fix. I resolved this fully by putting the push buttons on the breadboard rather than using the ones on the FPGA board. This allows me to hold down the push button for much less time than the wait from push button to LEDs.

The third issue was adding more states with more push buttons would cause a large delay of ten seconds or more. This would be very unrealistic in a real world setting so it was the most annoying issue to cope with. I realized after a while my state machine was incorrect. Rather than going from initial to a wait state where the buttons would be pressed, I was expecting the push buttons to work from initial. Once I added the extra state the issue seemed to have stopped.

The fourth issue was incorrect mapping of the LEDs and push buttons to the FPGA board. The way this was solved was by asking the professor who cleared up the mapping issues. I had connected everything to voltage and ground rather than the pins I had assigned them to which had kept them permanently on.

The last issue was cable management. Whether or not everyone thinks its an issue, having to press a button under so many cables is an issue. I fixed this by using flat short cables in color coordination for the board and then long cables to reach the FPGA.

**Recommendations and/or Conclusions**

The recommendations I would have for improvement are finding a way to put the buttons and the LEDs at different clock cycles to minimize the delay from pressing the button to the LEDs turning on on the breadboard. Another modification is attempting to make the delay from pressing reset shorter since there is a big delay from pressing reset to the LEDs turning on. The reset must be pressed twice to have it fully work. A last modification would be modifying the actual way the lights flash or adding/taking away certain sequences to best fit the use.

This project is inspired by Christmas lights but towards the end I was able to see other applications for it. The first thought was the blinker in a mustang since it lights up different lights in a certain direction. The other implications just seemed obvious at that point. Things like lights on a runway, disco lights, RGB in computer components, and ambulance/police lights.

I believe the project came out exactly as I expected it to work when I first thought of it. Some disadvantages are the delay and the limit of space on the breadboard for more LEDs or push buttons. I believe with more breadboards the project could’ve been much larger but a limitation on supplies kept me very restricted.