@default

# ECE 210 - Combinational Logic Design
# Lab 2

David Lenfesty
lenfesty@ualberta.ca

Radomir Wasowski
wasowski@ualberta.ca

2018-10-24

# Contents

# 1 Abstract

Various logic circuit designs can be combined together to create functional applications.

Multiplexers and demultiplexers can be used to route specific incoming signals to a specified destination. In this lab a simulated digital signal from three different receivers was routed towards one of three ”engineers”.

As well, logical elements can be used to design a simplistic access control system. In this lab such a circuit was designed and tested on an FPGA.

# 2 Introduction

The purpose of this lab was to design a simple Multiplexer and Demultiplexer circuit, as well as to design a circuit to control access to a lab.

In order to validate the Multiplexer/Demultiplexer circuit, a useful circuit was designed using Xilinx Vivado software, and this circuit was simulated against the required inputs.

In order to test the Lab Access Control circuit, the circuit was again designed using Xilinx Vivado, and simulated against required inputs. However, for this section, the design was also uploaded to a physical board where various inputs could be physically tested and validated.

# 3 Design Section

In order to design the following circuits, the Xilinx Vivado software was used to write VHDL that described the logic circuit.

**MUX / DEMUX Circuit**

```
entity lab2_part1 is
    Port ( clk : in STD_LOGIC;
           I  : in STD_LOGIC_VECTOR (2 downto 0);
           O  : out STD_LOGIC_VECTOR (2 downto 0);
           EI : out STD_LOGIC_VECTOR (2 downto 0);
           S  : in STD_LOGIC_VECTOR (1 downto 0);
           DS : in STD_LOGIC_VECTOR (1 downto 0) );
end lab2_part1;

architecture Behavioral of lab2_part1 is
    signal M  : STD_LOGIC := ’0’;
    signal SEL : STD_LOGIC_VECTOR (2 downto 0) := "000";

begin

    M <= ((not S(1) and not S(0)) and I(0)) or ((not S(1) and S(0)) and
         I(1)) or (( S(1) and not S(0) ) and I(2));
    SEL(0) <= (not DS(1) and not DS(0));
    SEL(1) <= (not DS(1) and DS(0));
    SEL(2) <= (DS(1) and not DS(0));

    O(0) <= M and SEL(0);
    O(1) <= M and SEL(1);
    O(2) <= M and SEL(2);

    EI(0) <= SEL(0);
    EI(1) <= SEL(1);
    EI(2) <= SEL(2);

end Behavioral;
```

**Lab Access Control Circuit**

## 4 Procedure

We typed things

**Part 1** Simu

**Part 2**

## 5 Results

Lights flashed

**Part 1**

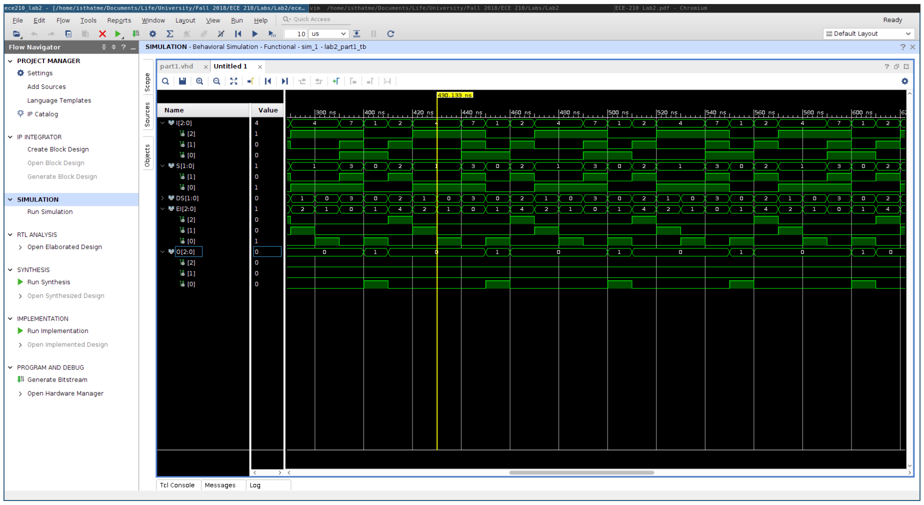
**Part 2**

## 6 Discussion

Dogs are better than cats. End of discussion

## 7 Conclusion

Blinkies are cool.



| Name | Value |
|---|---|
| I[2:0] | 4 |
| [2] | 1 |
| [1] | 0 |
| [0] | 0 |
| S[1:0] | 1 |
| [1] | 0 |
| [0] | 1 |
| DS[1:0] | 0 |
| EI[2:0] | 1 |
| [2] | 0 |
| [1] | 0 |
| [0] | 1 |
| O[2:0] | 0 |
| [2] | 0 |
| [1] | 0 |
| [0] | 0 |

Figure 1: Part 1 Simulation.