ECE 210 - Combinational Logic Design

Lab 2

David Lenfesty lenfesty@ualberta.ca

Radomir Wasowski wasowski@ualberta.ca

2018-10-24

1 Abstract

It is often the case that more fundamental elements in any field can be combined to produce more complex and useful structures. This is certainly true in the field of circuit design, where discrete elements are used to create functional applications all the time.

Multiplexers and demultiplexers can be used together to route specific incoming signals to a specified destination one at a time through a shared transmission line. The requirement of this lab was to design boolean expressions realizing the functionality of multiplexers and demultiplexers in the context of a real-life application.

Another real-life application of logical elements is in an access control system. A simplistic version of such a system was also designed in this lab session and tested on an FPGA afterwards.

2 Introduction

The purpose of this lab was to design control circuits according to the provided specifications, and then verify their operation using a simulation or an FPGA.

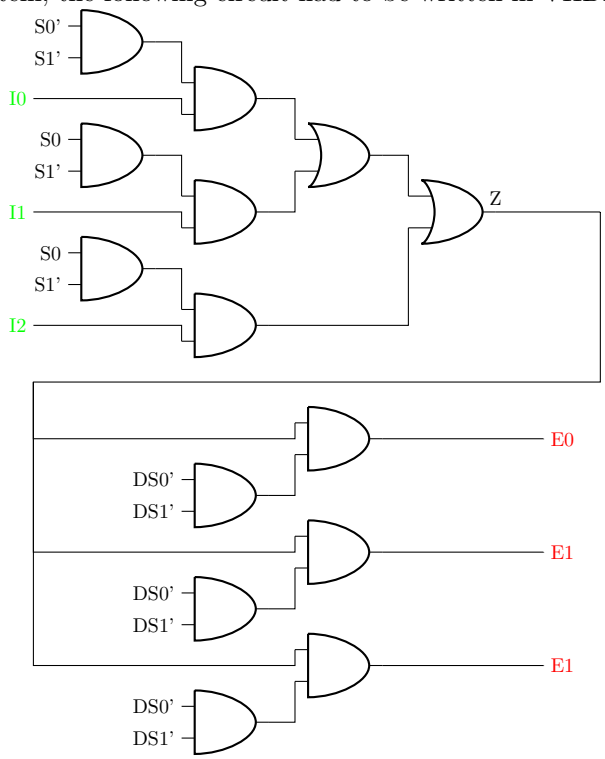
In the first part of the lab, a series of boolean expressions were designed to implement a Multiplexer/Demultiplexer circuit, intended to route data from one of three radio recievers to one of three engineers, and signal which engineer was currently recieving data. First, Xilinx Vivado Software was used to produce a circuit to fulfill this objective. Then, using the same software, the circuit was simulated against input combinations which would be encountered during normal use, for verification.

For the second part of the lab, an Access Control circuit was to be designed, allowing lab entry only if a valid ID was provided alongside a proper keypad combination. Otherwise, an alarm signal was to be sent out. The method of designing this circuit was very similar to the method in part one: Again using Xilinx Vivado, the circuit was designed and simulated against inputs to verify if the outputs matched those in the specification. However, for this section, the design was also uploaded to a physical FPGA board where various could be manually tested and validated.

3 Design Section

In order to design the desired systems, the Xilinx Vivado software was used to write VHDL code that described the operation of each circuit.

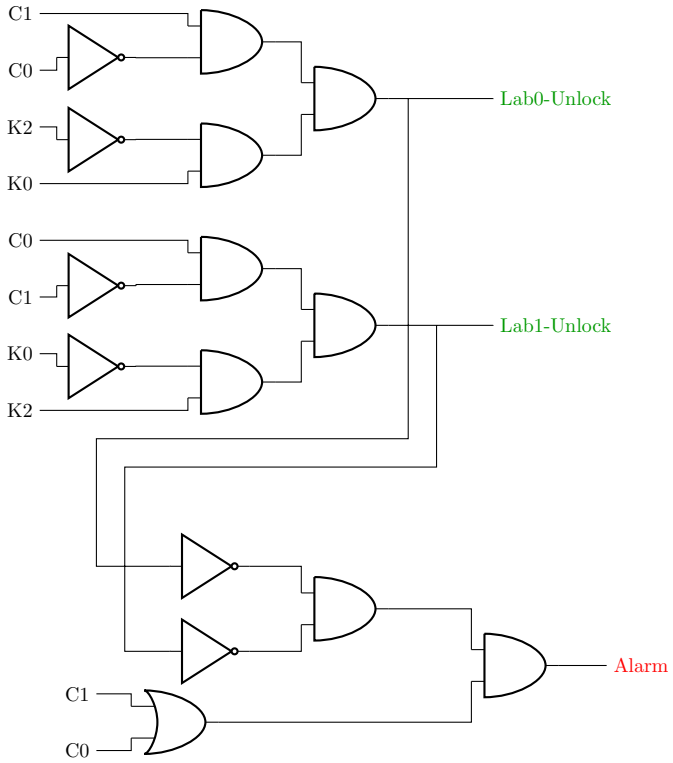
MUX / DEMUX Circuit To implement the multiplexing/demultiplexing system, the following circuit had to be written in VHDL.



The VHDL architecture below was written to implement this circuit in hardware.

```
1  architecture Behavioral of lab2_part1 is
2      signal M   : STD_LOGIC := '0';
3      signal SEL : STD_LOGIC_VECTOR (2 downto 0) := "000";
4  begin
5      M <=   ((not S(1) and not S(0)) and I(0))
6          or ((not S(1) and S(0)) and I(1))
7          or (( S(1) and not S(0) ) and I(2));
8
9      SEL(0) <= (not DS(1) and not DS(0));
10     SEL(1) <= (not DS(1) and DS(0));
11     SEL(2) <= (DS(1) and not DS(0));
12     O(0) <= M and SEL(0);
13     O(1) <= M and SEL(1);
14     O(2) <= M and SEL(2);
15     EI(0) <= SEL(0);
16     EI(1) <= SEL(1);
17     EI(2) <= SEL(2);
18 end Behavioral;
```

Lab Access Control Circuit To implement lab access control as described, the following circuit was developed.



The VHDL architecture written to implement this circuit in hardware is next:

```
1  architecture Behavioral of lab2_part2 is
2      signal Lab0_Correct : STD_LOGIC := '0';
3      signal Lab1_Correct : STD_LOGIC := '0';
4  begin
5
6      Lab0_Correct <= (C(1) and not C(0))
7                  and (K(0) and not K(2));
8      Lab1_Correct <= (not C(1) and C(0))
9                  and (not K(0) and K(2));
10
11     Lab0_Unlock <= Lab0_Correct;
12     Lab1_Unlock <= Lab1_Correct;
13
14     Alarm <= (C(1) or C(0))
15          and (not Lab0_Correct and not Lab1_Correct);
16 end Behavioral;
```

4 Procedure

In order to test that each circuit worked properly, the corresponding VHDL programs were simulated. Then, a physical FPGA was programmed with the access control code and the design was verified using buttons.

The circuit for part one was tested using the supplied simulation file. The outputs were then checked against the expected truth table.

For part two, the circuit was simulated and run against another supplied simulation file. Additionally, it was uploaded to an FPGA board, using the provided constraints, where the functionality of the access system was verified by using the physical buttons and switches on the board.

5 Results

Part 1 The VHDL design worked perfectly according to the specifications laid out. Below is the output from the simulation:

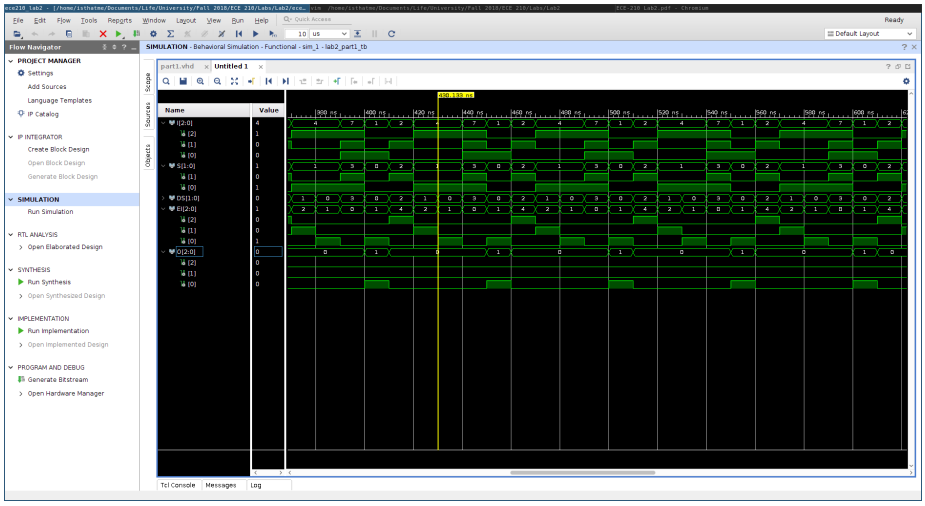


Figure 1: Results from Multiplexer/Demultiplexer circuit simulation in Xilinx Vivado

Part 2 The VHDL design was able to correctly control the lab access and alarm signals as specified, both in the simulated results and the tests on the physical board. Below are images of the simulation waveform and a physical test of one of the input combinations.

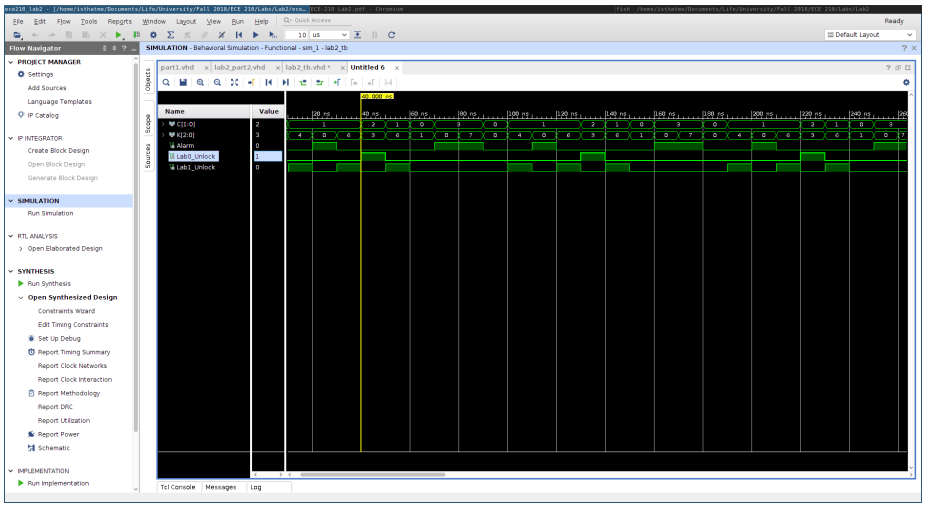


Figure 2: Results from Lab Access Control circuit simulation in Xilinx Vivado

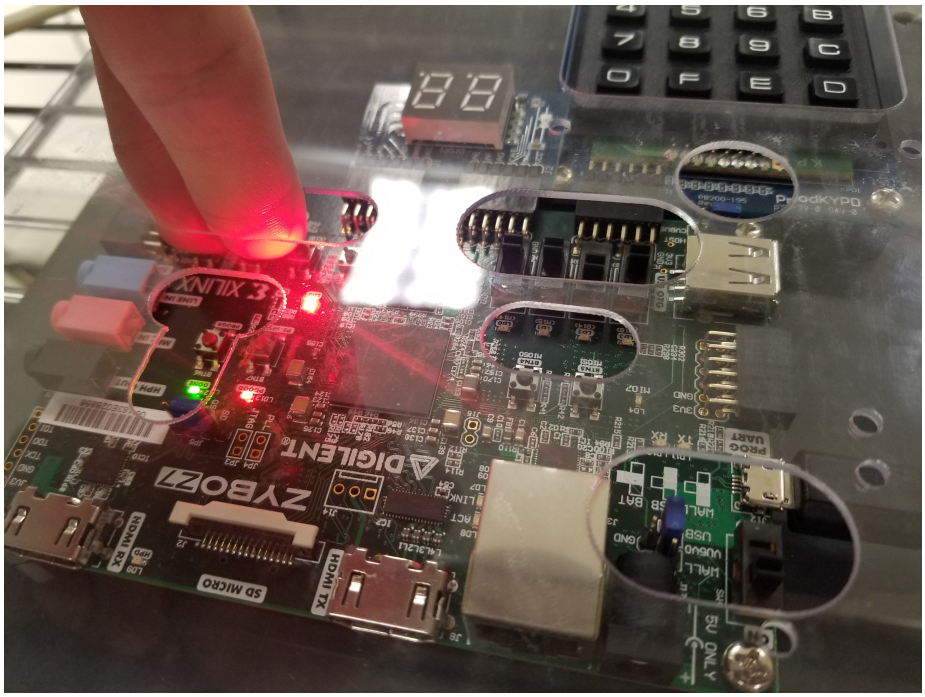


Figure 3: Physically Testing the Lab Access Control system on an FPGA board

6 Discussion

Multiplexing and demultiplexing circuits are designed using 'selection' elements. These elements can also be repurposed to create functional and practical circuits for real life applications.

Mux and Demux Circuit

1. When the mux and demux select signals are in an unused state, There will be no signal outputs going to any of the engineers.
2. Waveform can be found in the results section.
3. The advantage of ICs is that you can physically see the layout of the circuit. As well, you can probe the internal connections. However, it is quite tedious to connect a large number of ICs together, and it can be quite error prone. The advantages of using an FPGA is that you can get a computer to make the connections for you. This allows you to rapidly design and prototype circuits. However, a disadvantage is that you cannot physically inspect the circuit and probe the internals.

Lab Access Control

1. The simulation waveform is provided above in the results section.
2. I would rate the system as somewhat effective. However, it is not at all upgradable or expandable as designed. There are too few possible combinations, and these are set on toggle switches, which is not secure. It performs according to the given specifications.
3. Something with a momentary keypad, as well as a programmable access control system. This would allow access to be given and revoked, passwords to be longer, and easily changable combinations.

7 Conclusion

It is sometimes necessary in real-life applications to use the fundamental elements of circuit technology to implement boolean expressions in hardware, when such hardware may be unavailable. Even if these packages may be purchased from a manufacturer, sometimes a situation may call for an implementation that cannot use these elements. Also, FPGAs are useful tools for the quick design and development of functional circuits. These circuits can borrow ideas and methods of operation from each other to more efficiently and creatively implement the desired functionality.