EEE3330.02-00 (Computer Architecture)

Project 1-1 Report

2021142180 Kim Min Chan

Date: 2023.04.05

[1. Logic Design]

This project is to design a combinational logic that distinguishes the number that goes into Student ID. I will design a combinational logic that receive a number between 0 and 9, and if that number is included in my Student ID, it will return 1 and if not, it will return 0. Corresponding numbers can be expressed into a 4-bit binary code as follows.

Decimal	Binary (ABCD)
0	0000
1	0001
2	0010
4	0100
8	1000

Figure 1. Decimal into 4-bit Binary

Since my Student ID is 2021142180, the number in student ID is 0, 1, 2, 4, 8. To design combinational circuit that returns 1 if 0, 1, 2, 4, 8 is input, and return 0 otherwise, we must express such logic in truth table as follows.

INPUT				OUTPUT
A	B	C	D	X
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Figure 2. Truth table of the logic

To design combinational logic circuit corresponding to the truth table, we need to express the logic in Boolean algebra. To reduce the number of gates, minimizing the Boolean algebra is inevitable. Since the input is 4-bit binary, by using Karnaugh map, I can minimize the Boolean algebra. Based on Karnaugh map, I expressed Boolean algebra and logic gate as follows.

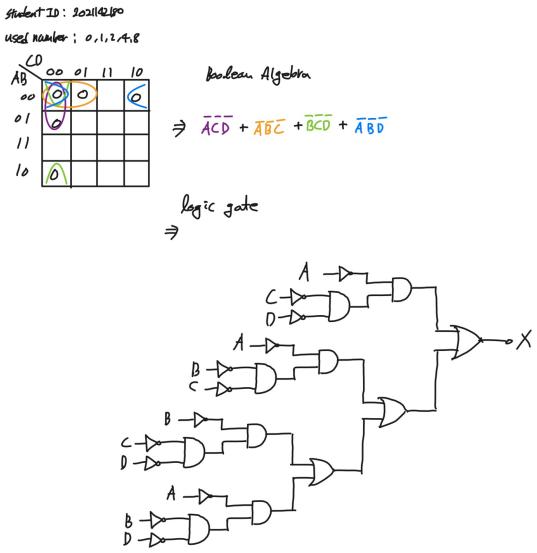

Figure 3. Karnaugh map, Boolean algebra, and design of Combinational Circuits.

[2. Simulating Design]

Now Using Intel Quartus, we can simulate the combinational logic that we designed. Figure 4 is the schematic of the circuit.

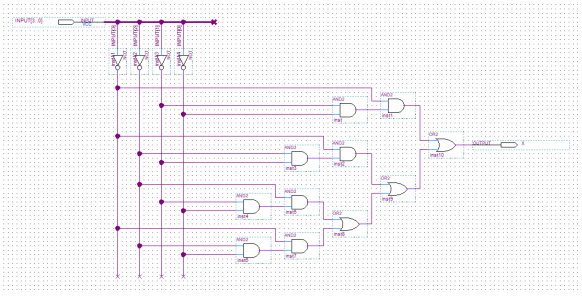

Figure 4. Schematic of the combinational ckt.

The basic logic is designed based on Figure 3. INPUT [3..0] has value of decimal number and Each of input of not gate, INPUT [3], INPUT[2], INPUT[1], INPUT[0], is corresponding to A, B,

C, D of binary code. By creating VWF file, we can simulate the design. The input value will be increased from 0 to 15 by increment 1 per 10ns. The simulation result is shown in Figure 5.

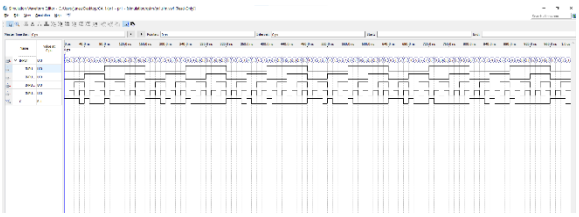

Figure 5. Simulation result (1)

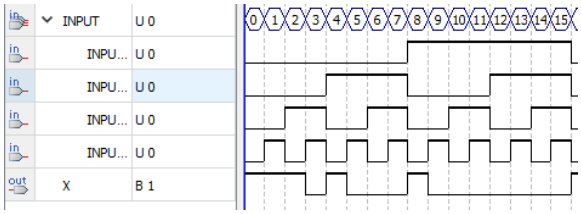

Figure 6. simulation result (2)

Figure 5 represents the whole simulation result and Figure 6 represents the simulation result of one period. In Figure 6, the variable names shown on the left indicate INPUT[3..0], INPUT[A], INPUT[B], INPUT[C], INPUT[D], OUTPUT in order. We can see the output X is 1 for INPUT[3..0] = 0, 1, 2, 4, 8 and 0 otherwise. Through this, we confirmed that our design based on Boolean algebra is it was confirmed that our logic design based on Boolean algebra works according to our expectations.

[3. Discussion]

In this project, we theoretically designed a combinational logic that is independent to time and conducted a simulation using Intel Quartus. This confirmed that the Boolean algebra-based logic design works with the design purpose. The combinational logic is also used to design a sequential circuit which is time dependent. Therefore, more complex logic can also be implemented by utilizing the logic design and simulation learned through this project.