ECE 385

Spring 2022

Experiment #1

Lab 1

Yulun Wu
(Teammate: Tianyu Yu)
TA: Curtis Yu

Purpose of Circuit

The primary purpose of the Lab 1 circuit is to design a 2-to-1 multiplexer with 2-input NAND gate chips. Furthermore, we redesign the circuit to solve the “glitches” problem which is caused by the properties of logic gate itself, and the performance of the circuit is improved after that.

Written Description of Circuit

The circuit is designed based on the concept of 2-to-1 multiplexer, which is supposed to have the following K-map.

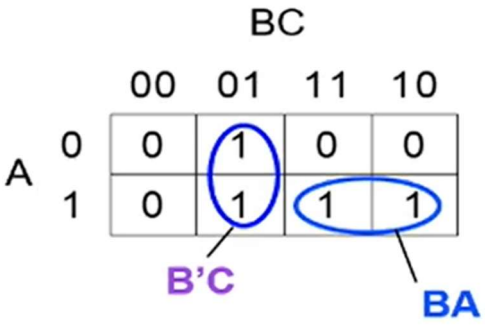

Figure 1. K-map of the circuit

After correctly circling in the K-map, we can come up with the logic expression: Q = AB + B’C, and the circuit design with 2 AND gate and 1 OR gate to accomplish the multiplexer. However, under the preference of using NAND gate only instead of AND and OR, we shall use de Morgan’s Law to further adjust the expression to: Q = ((AB)'(B'C)')'. Notice that we need a converter for B’, but this function can be achieved by connecting both pins of NAND gate with B’s input. Thus, only 4 NAND gates are needed for the circuit design, and we shall have the logical diagram and layout as below.

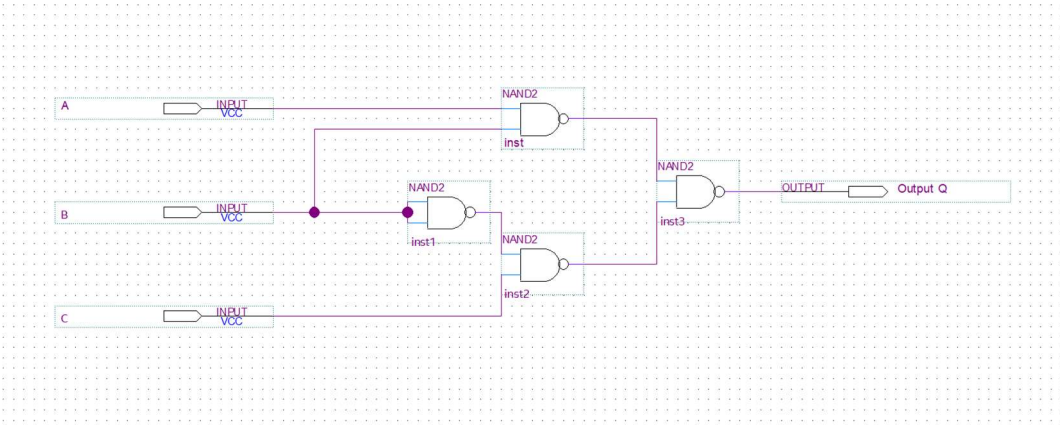

Figure 2. Logic Diagram for Multiplexer

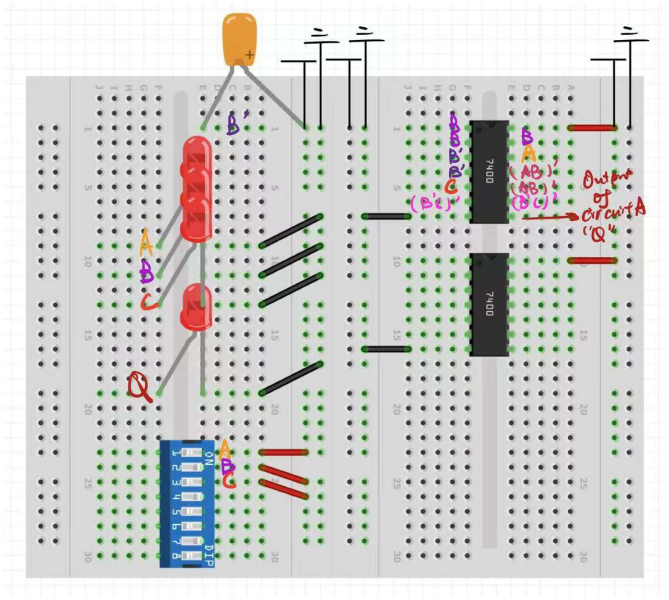

Figure 3. Circuit component layout (circuit A)

However, even it's subtle, we will meet static hazard due to the nonzero propagation delay from NAND gate's input to output. As a result, in a very short period, the output of circuit is incorrect. It's more obvious under oscilloscope. In current circuit, both A and C provides high-voltage (1) input, and we will expect the result stays high (1) no matter B's on or off. But when we switch B to low-voltage (0) input, there is a gap appear in the output, indicating a temporary wrong output (0). The fixing process will be shown in the Prelab question B.

Figure 4. Waveform of circuit that contains a short period of wrong output

Prelab Questions

Note: Some logic diagram and layout sheet are shown in previous sections.

A.

a) Why not all groups may observe static hazards?

Because the static hazard happens in a pretty short period, and the period of it also depends on the specific property (delay and glitch) of the chip we use.

b) Why does the hazard appear when you do this?

Because it cause a longer delay time to signify the hazard. We utilize a capacitor which cause a

short period of time to charge. We count this into the hazard in our lab.

B. Redesign the circuit of part A to eliminate all static-1 hazards (glitches) at the output.

To fix the problem, we look back to the K-map again, and circle an additional term:

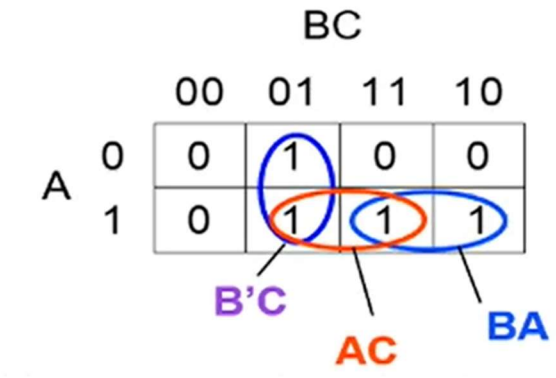

Figure 5. K-map of the circuit with additional AC term

Now the logic expression from the K-map is: Q = AB + B'C + AC, and if we further change it to adapt the NAND-only circuit with de Morgan's Law, we will have: Z=((((AB)'(B'C)')')' (AC)')'. The advantage to do so is to overlap more logic gates in the circuit to reduce the period of static hazard. The newly optimized logic diagram and circuit layout are in Figure 5 and 6:

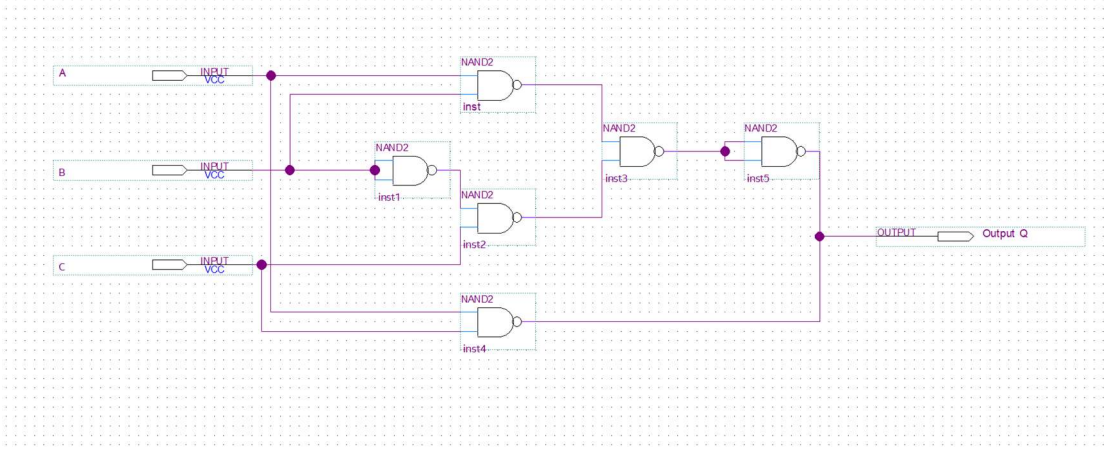

Figure 6. Logic Diagram for Multiplexer (optimized with AC)

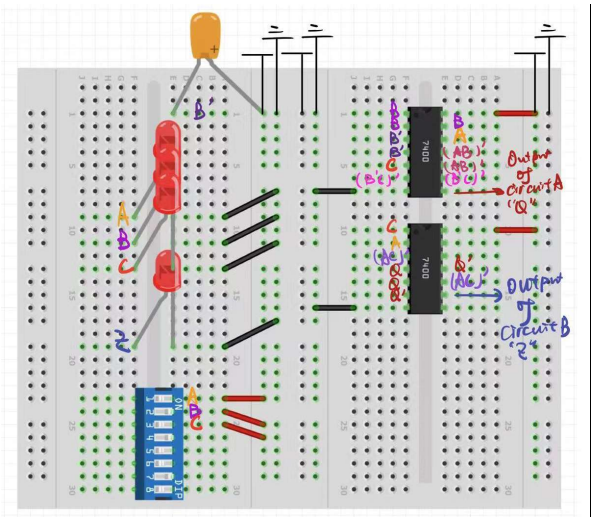

Figure 7. Circuit component layout (green output)

If we observe the oscilloscope now, the static hazard problem is reduced. Even though there are certain level of noise indicating the switch of B's input, the overall output stays in the same level.

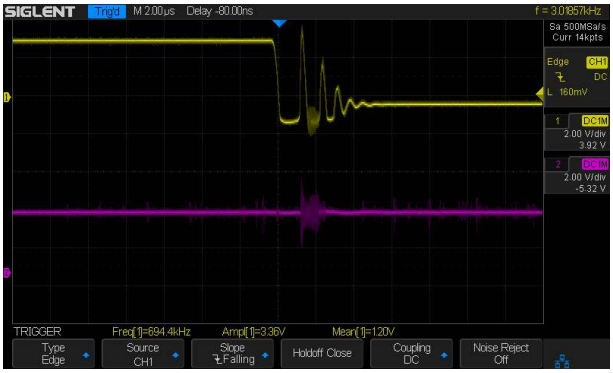

Figure 8. Waveform of circuit that is correct by AC

Lab Questions

1. Unit Test

Test has been done, and all gate within the chip works normally.

2. Test part A circuit and complete truth table.

A	B	C	Q
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0

Figure 9. Truth table of part A circuit

The A circuit's performance same way as the truth table.

3. Waveform of circuit A

Figure 10. Waveform of circuit A with 0-5V, 1MHz square wave input to B

4. Test part B circuit and complete truth table, take 0-5V, 1MHz square wave input to B and observe the oscilloscope.

A	B	C	Q
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0

Figure 11. Truth table of part B circuit

The B circuit's truth table is same with A circuit's one. We can observe that A and B's performance under 0-5V, 1MHz square wave input to B are quite different.

Figure 12. Waveform of circuit B with 0-5V, 1MHz square wave input to B

We can observe that in B's waveform, there's barely no pattern indicating a glitch happening when we switch between 5V and 0V to input B. In A's waveform, we can see that in every falling edge of input, the output displays a falling edge (static hazard) due to the delay. And it's noticeable that the period of delay is longer than 1MHz input wave, so the output will raise when the input signal raise from 0V to 5V, showing same frequency. So I think in the falling edge of input B, we are more likely to observe a glitch at output.

Post-Lab Questions

1. How long does it take the output Z to stabilize on the falling edge of B (in ns)? How long does it take on the rising edge (in ns)? Are there any potential glitches in the output, Z? If so, explain what makes these glitches occur.

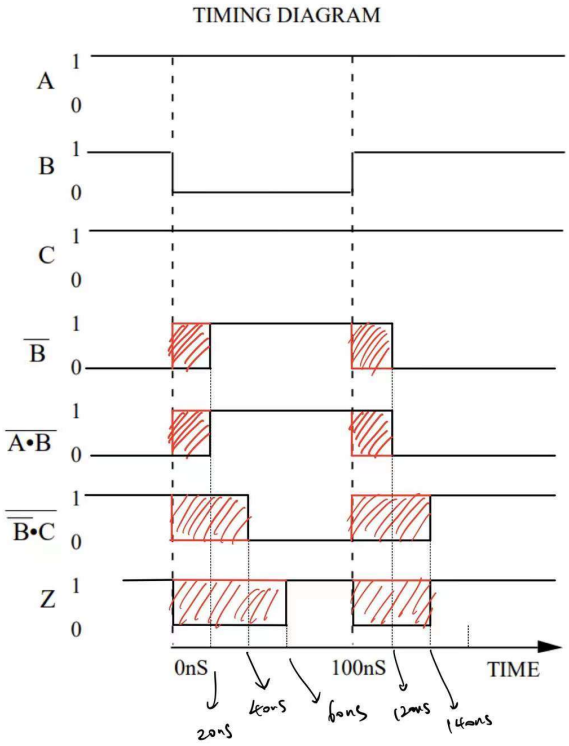

Figure 13. Timing diagram

As shown in the final row of timing diagram, it takes 60ns for the circuit to stabilize and have a predictable value on the falling edge of B, whereas it takes 40ns on the raising edge. The glitches are potential on both raising and falling edges due to the delay of our NAND gate, which makes the response time of each logic gate in the circuit uncertain. In falling edge, we see the term (AB)' tends to response 20ns faster (earlier to be stabilized to a certain value) than the (B'C)' term, so it will cause 60ns potential glitch, and this also causes potential glitch in raising edge, but it may only last 40ns. So, in raising edge, it a little less likely for glitch to occur.

2. Explain how and why the debouncer circuit given in General Guide Figure 17 (GG.32) works. Specifically, what makes it behave like a switch and how the ill effect of mechanical contact bounces is eliminated?

We know the mechanical switch normally brings "contact bounce", the state of switch jumping between 0 and 1 for several times, because of the electrical contacts inside the switch. To make a clear and stable switch, the debouncer circuit applies two NAND gates within the switch, making a clear logic output whether switch is in position A or B.

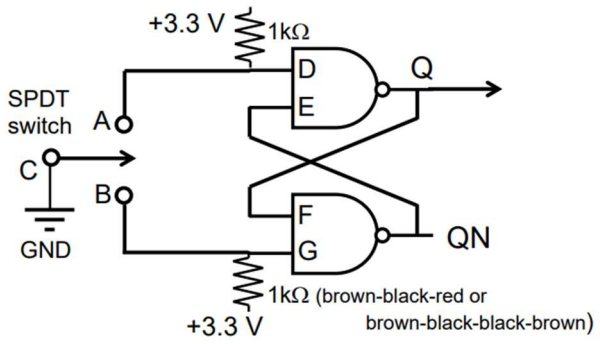

Figure 14. Debouncer circuit

It always stores a logic state, so let's set Q = 1 when switch is at position A. Now input to D is 0V due to connection to GND through A and C. So E is 0V now, which is just Q'. Because G is directly connected to 3.3V, it's logic input is 1 and the value of F, same with Q, is 1. When switch comes to position B, G first became 0V, making Q' and E be 1, while D becomes 1, so output Q becomes 0. Thus, no matter which side the switch at, there will always be a stable logic output that doesn't affected by electrical contact.

Additional General Guide Questions

1. What is the advantage of a larger noise immunity? Why is the last inverter observed rather than simply the first? Given a graph of output voltage (VOUT) vs. input GG.7 voltage (VIN) for an inverter, how would you calculate the noise immunity for the inverter?

1) With larger noise immunity, the chip can prevent the output from the effect of unclear and unstable pulse. Thus the logic output will tend to be correct with greater chance. 2) Because in a series of inverters, the noises are being overlapped again and again, and if the last one still keeps logic output correctly, its voltage level is the closest to the gate's nominal logic level. 3) The immunity for inverter shall be 0.8V. Since the nominal "0" level is 0.35V, and the largest input level that keeps the output identified as "0" is 1.15V, so we have 1.15V-0.35V=0.8V, as the noise immunity for the inverter.

2. Irrespective of which polarity you choose for your LEDs, it is important that each LED has its own resistor. If we have two or more LEDs to monitor several signals, why is it bad practice to share resistors?

Because each logic gates may have different maximum sink current value, the practice that connecting LEDs that indicate different output with a single resistor may lose the flexibility of adjusting the current flow to each LED, creating potential risk of mis-display of logic result or break the LED.

Conclusion

In this intro lab of ECE 385, I reviewed how to build logic circuit based on truth table and K-map. I also studied the causes and phenomenon of static-1 hazard which can potentially lead to wrong output of circuit. Through the learning of the lab, I've figured out how to avoid it when it's necessary in circuit design.