ECE 385: Digital Systems Laboratory

Fall 2022

Experiment #1

Introductory Experiment

Jialiang Xu (jx17)
TA: Neo Yuan
September 7, 2022

1 Introduction

In this lab, the phenomenon of Static Hazards is studied with the example of an 2 to 1 MUX. To begin with, a naïve version of the 2 to 1 MUX is implemented using only the 7400 NAND gate chips. This implementation possess the weakness of potential static-1 hazard ("glitch"). Then the circuit is re-designed to include redundancy in K-map terms to avoid a state transfer between adjacent minterms, which eliminates the possibility of any glitch.

2 Circuit Operation Description

In both parts of the prelab, the target of the circuits is to implement a 2 to 1 MUX with two inputs (A, C), one output (Z) and select (B) with NAND gates. The basic functionality is that when the select B is at a logic one, the output Z should be directly outputting the value of A; and when B is at a logic zero, the output Z should be outputting that of C.

2.1 Prelab Part A

In part A of the prelab, a naïve design is adopted by following intuitive logic implementation and minimum gates utilization. The K-map corresponding to this behavior is shown in Figure 1. The corresponding logic expression derived from the K-map is:

$$Z = BA + B'C \tag{1}$$

Figure 1: The K-map for the output Z in the part A circuit. The two orange circles, which are adjacent to each other, represent the chosen minterm implementations. The adjacency implies a potential static-1 hazard.

A \ BC	00	01	11	10
0	0	1	0	0
1	0	1	1	1

For implementation with NAND gates, De Morgan's law is applied to Equation 1 to convert the logic expression into a cascade of NAND operations:

$$Z = ((AB)'((BB)'C)')' \tag{2}$$

2.2 Prelab Part B

In part B of the prelab, a re-designed implementation is adopted by adding a redundant logic term AC to remove the discontinuity between the two adjacent minterms. The K-map corresponding to this behavior is shown in Figure 2.

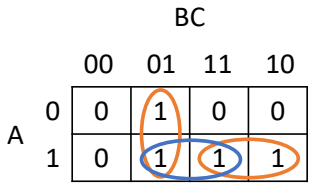

Figure 2: The K-map for the the output Z in the part B circuit. In addition to the two orange circles, one redundant blue circle is added to cover the state transfer between two adjacent minterms. This modification eliminates the static-1 hazard.

The corresponding logic expression derived from the K-map is:

$$Z = BA + B'C + AC \tag{3}$$

As a result, the NAND-compatible logic expression is then correspondingly updated to the following:

$$Z_0 = ((AB)'((BB)'C)')' \tag{4}$$
$$Z = ((Z_0 Z_0)'(AC)')' \tag{5}$$

where Z_0 is the original output in Part A, and Z is the current Part B output.

3 Component Layout Sheet

In this seciton the component layout sheets of the two parts in prelab are provided. The component layout sheet of Part A can be found in Figure 3, and that of Part B can be found in Figure 4.

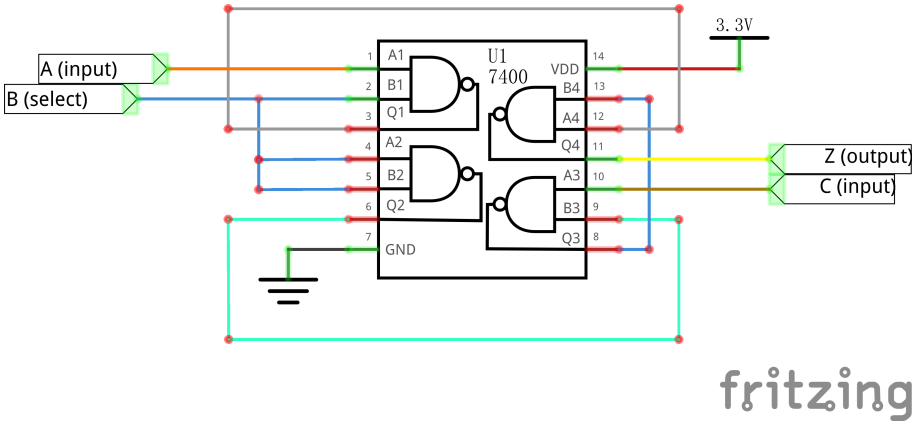

Figure 3: The Component Layout Sheet for the Part A circuit.

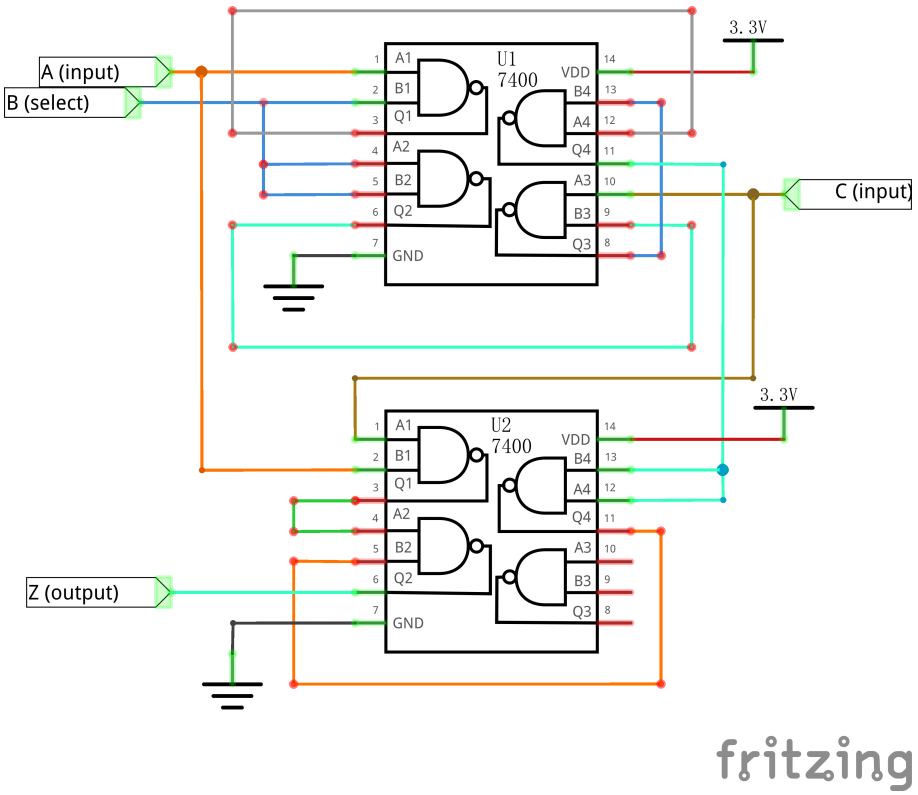

Figure 4: The Component Layout Sheet for the Part B circuit.

4 Circuit Diagrams

In this section the circuit diagrams of the two parts in prelab on a breadboard are provided. The cuicuit diagram of Prelab Part A can be found in Figure 5 and that of Part B can be found in Figure 6.

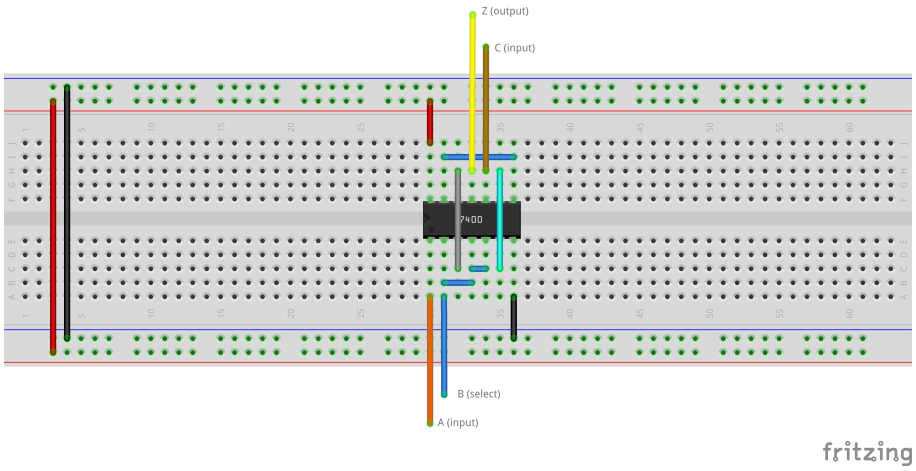

Figure 5: The Circuit Diagram for the Part A circuit.

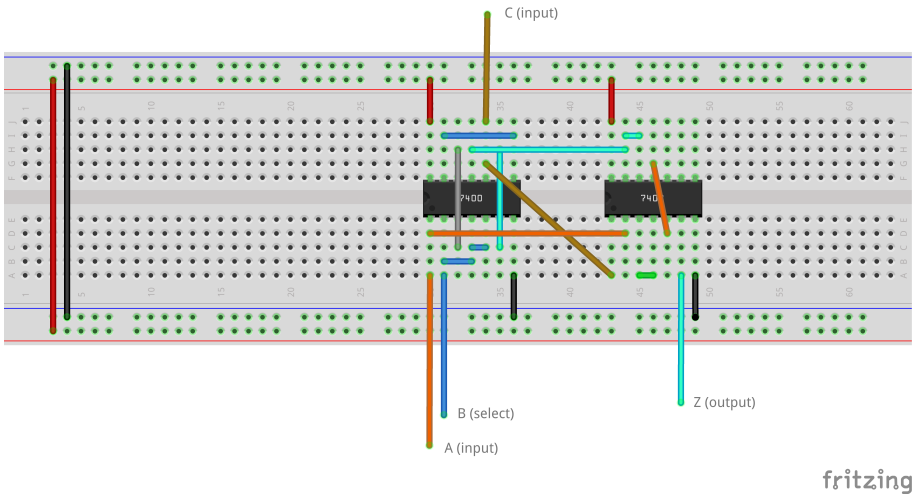

Figure 6: The Circuit Diagram for the Part B circuit.

5 Lab Documentations

5.1 Pre-lab Part A Truth Table

The truth table for Part A can be found in Figure 7. This truth table is derived from the logic expression in Equation 1.

A	B	C	Z
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Figure 7: The Truth Table of Prelab Part A.

5.2 Pre-lab Question Answers

5.2.1 Why some groups won't observe a glitch?

Due to the variance in the inverter chip's propagation delay, some chips won't produce an observable glitch. According to the 7400 NAND gate datasheet[1] the gate possesses a delay time ranging from 0 to 22 ns under the reported experiment environment. In the cases where the glitch is not observed, it is possible that the chip is having a lower propagation delay.

5.2.2 Why glitches will be observed if one chain an odd number of inverters or add a small capacitor to the output of the inverter?

When an odd number of inverters are chained together, their delay adds up to a distribution with higher mean and variance, which gives rise to a higher probability of glitches lasting sufficient time to be observed.

When a capacitor is added to the output of the inverter, it creates a "buffer" for electric charges. Each time the output of the inverter is set to logic zero, the charges accumulated on the capacitor get released (potentially only a portion, but the buffering effect still exists), and next time when the output is switched back to logic one, the charges start to accumulate on the capacitor again. These extra charges on the capacitor requires extra time for the input of NAND to

[1] https://web.mit.edu/6.131/www/document/7400.pdf

rise to logic one, which can be viewed as an extended time before the output of the inverter rising back to logic one (the glitch time is increased).

Thus, the glitches are easier to be observed with the two approaches above.

5.3 Pre-lab Part B Truth Table

The truth table for Part B can be found in Figure 8. This truth table is derived from the logic expression in Equation 3.

A	B	C	Z
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Figure 8: The Truth Table of Prelab Part B.

5.4 Oscilloscope Printout

The oscilloscope printout for Prelab Part A can be found in Figure 9, and that of Prelab Part B can be found in Figure 10. Note that despite the noise during the switching, the glitch observed in Part A has been eliminated in Part B.

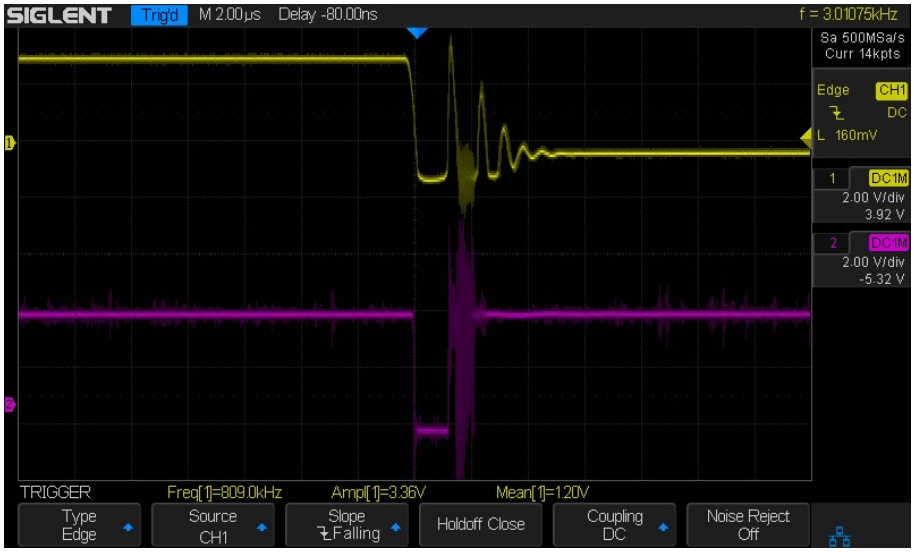

Figure 9: The Oscilloscope Printout for the Part A circuit.

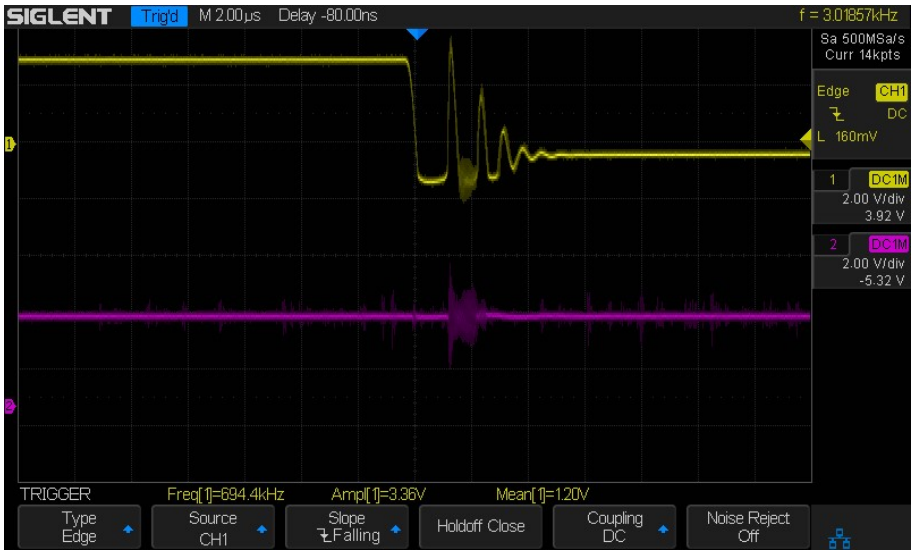

Figure 10: The Oscilloscope Printout for the Part B circuit.

6 Post-lab question answers

6.1 Timing Diagram

6.1.1 Complete the timing diagram for the circuit of part A

The completed timing diagram can be found in Figure 11. For B' and $(AB)'$, the switching of B are delayed by one NAND gate, resulting in a 20 ns period of uncertainty. For $(B'C)'$, the switching are delayed by two NAND gate, resulting in a 40 ns. For Z, the switching are delayed by three NAND gate, resulting in a 60 ns.

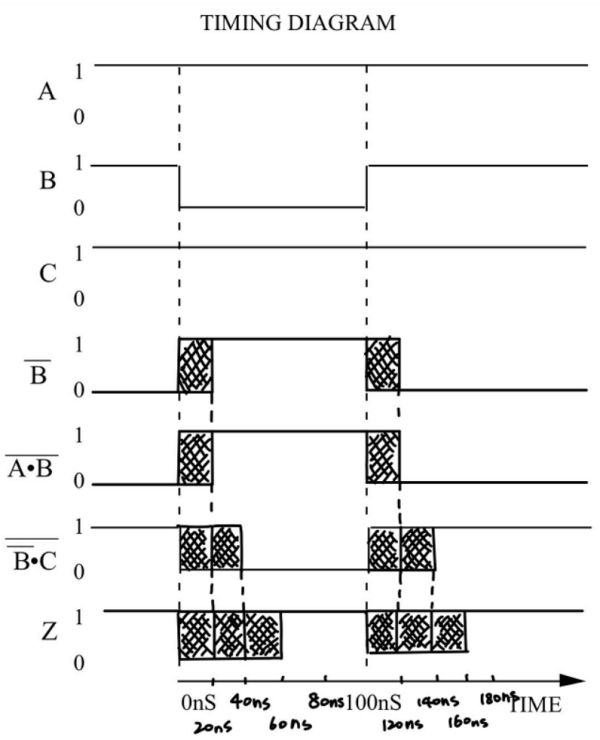

Figure 11: The timing diagram for the circuit of part A.

6.1.2 How long does it take the output Z to stabilize on the falling edge of B (in ns)?

Output Z will stabilize after $\boxed{60ns}$ on the falling edge of B, the reason being that Z was delayed by 3 NAND gates, and only after the third NAND gates having stabilized will Z be stabilized.

6.1.3 How long does it take on the rising edge (in ns)?

Following the reasoning in §6.1.2, the output Z will stabilize after $\boxed{60ns}$ on the falling edge of B.

6.1.4 Are there any potential glitches in the output, Z? If so, explain what makes these glitches occur.

Yes. Potential glitches will happen on the falling edge of B, but not on the rising edge.

The glitches occur due to the possibility that the falling edge of $(B'C)'$ occurs later than the rising edge of $(AB)'$ due to the propagation delay, resulting in a period of time where the output Z becomes temporarily logic zero. On the rising edge of B, the aforementioned misplacement won't affect the output because there isn't an overlap between the two aforementioned signals.

6.2 Debouncer

6.2.1 Explain how and why the debouncer circuit given in General Guide Figure 17 (GG.32) works. Specifically, what makes it behave like a switch?

It works like a switch because the connection of C determines the output of Q. When C is connected to A, Q outputs logic one, and when C is connected to B, Q outputs logic zero. Without loss of generality, we discuss the situation of C being connected to A, we show that Q will output logic one and Q' will output logic zero:

When C is connected to A, D gets tied directly to `GND` (logic zero), and G gets pulled up to `VCC` (logic one); There are two situations:

- Current Q is logic one: then F gets logic one, which results in the output of the lower NAND gate Q' being logic zero. This implies the output of the upper NAND gate Q being logic one, so the circuit is stable.
- Current Q is logic zero: then the output of the lower NAND gate Q' gets logic one, which leads to E getting logic one. However, then the output of the upper NAND gate changes to logic one, resulting in F being updated to logic one. The change to F further updates Q' to logic zero, and also E to logic zero. At this point, the output of the upper NAND gate is logic one, and the circuit gets stabilized.

6.2.2 How the ill effect of mechanical contact bounces is eliminated?

The assumption of the elimination is the bouncing effect was not severe enough for the SPDT switch to contact the throw other than the intended one. The way it eliminates the ill effect is that the latch stores the current circuit output even if the pole of the switch leaves the intended throw. Without loss of generality,

we discuss the situation of C bouncing off A, and we will show that the bounce won't affect the current status of circuit.

Before C leaves A, both D and E are both at logic zero (as analyzed in §6.2.1). When the pole bounces off from D, D gets pulled up to `VCC`, which results in it being logic one. However, due to the property of the NAND gate, the output of the upper NAND gate Q is still logic one and not affected.

Thus, the ill effect of mechanical contact bounces is eliminated.

7 General Guide question answers

7.1 GG.6: Noise Immunity

7.1.1 What is the advantage of a larger noise immunity?

A larger noise immunity provides the circuit with more robustness. Larger noise immunity prevents the status of the circuit to be easily changed to another level by tolerating more external perturbation.

7.1.2 Why is the last inverter observed rather than simply the first?

When multiple inverter are chained together, the effect of chip-wise parameter variance gets attenuated. In the chain, every inverter is outputting a signal level closer to the nominal levels than the previous one. If only the first gets observed, the experimental noise has a strong effect on the observation.

7.1.3 Given a graph of output voltage (VOUT) vs. input voltage (VIN) for an inverter, how would you calculate the noise immunity for the inverter?

By definition, the noise immunity is the "the maximum amplitude of a positive going (noise) pulse added to the nominal logic '0' voltage level or a negative-going (noise) pulse added to the logic '1' voltage level at the input of a gate which does not cause the output of that gate to change its logic value". According to the graph, the guarantee of a logic one being not changed ends at input voltage $1.15V$, the norminal range of logic zero is 0 to $0.7V$, so the noise immunity for range X is $1.15 - 0.7 = 0.45(V)$. Following same reasoning, the noise immunity for range Y is $2 - 1.35 = 0.65(V)$. Since we are choosing the minimum value, the final noise immunity is $\boxed{0.45\text{V}}$.

7.2 GG.31: LED

7.2.1 If we have two or more LEDs to monitor several signals, why is it bad practice to share resistors?

When multiple LEDs are connected in parallel, the current is not evenly distributed. There's a danger that the single diode with lowest forward voltage will start conducting while others won't. This further leads to the first diode

being worn out quickly, and the diode with lowest forward voltage in the rest will wear out the next, etc. This will eventually leads to a failure of the circuit. Thus sharing resistor across LEDs is a bad practice.[2]

8 Conclusion

In this experiment, we studied the issue of static-1 hazard (glitch) via an example 2 to 1 MUX. To begin with, an naive design of the desired logic expression is implemented, and output observation is conducted with oscilloscope printouts. The results aligned with our expectation. Then the reasons of the glitch are identified, which is that a short-period circuit state inconsistency occurred due to the propagation delay of the NAND gates. A series of questions related to the Prelab and Postlab are also answered in this report.

[2]Reference: https://electronics.stackexchange.com/questions/22291/why-exactly-cant-a-single-resistor-be-used-for-many-parallel-leds