

ECE385

Spring 2016

Experiment #1

2-to-1 Multiplexer TTL Circuit

Eric Meyers

Section ABG

January 26, 2016

1 Introduction

The purpose of this lab is to demonstrate the impact of using a glitch-free 2-to-1 multiplexer using Transistor-Transistor Logic (TTL) chips on a protoboard. This experiment consisted of two parts: (1) Designing and constructing a glitch-prone 2-to-1 multiplexer, connecting a 1 MHz square-wave input to the select-line, and analyzing its output waveform and (2) designing and constructing a glitch-free 2-to-1 multiplexer, connecting a 1 MHz square-wave input to the select-line, and analyzing its output waveform. Static hazards were also explored during this lab and are explained when necessary below.

2 Description of Circuits

As stated in the introduction, there are two circuits created in this experiment. The first one being a glitch-prone 2-to-1 multiplexer and the second being a glitch-free 2-to-1 multiplexer. Both Part A and Part B are explained in detail below.

Part A: The glitch-prone 2-to-1 multiplexer was constructed using one SN7400 Quad 2-Input NAND chip. In the pre-lab, a 2-to-1 multiplexer was designed by first creating a Karnaugh-Map (shown below in Figure 1), then selecting the optimal minterms and constructing an AND-OR circuit ($Z = (A \wedge B) \vee (\neg B \wedge C)$). This was then converted to a NAND-NAND circuit using deMorgan's Laws, and constructed on a protoboard.

Power was then connected to the protoboard and the A and C input were driven to high, while a 1 MHz square-wave was connected to the select line (B). The reason a square-wave was connected to the select-line was in order to test the reliability of the output and to, specifically, check for a glitch occurring on the falling edge of the input.

	bc	00	01	11	10
a	0	0	1	0	0
	1	0	1	1	1

Figure 1: Glitch-Prone K-Map for Part A

Part B: The glitch-free 2-to-1 multiplexer was constructed using two SN7400 Quad 2-Input NAND chips. The pre-lab adequately described the process for creating a glitch-free circuit by constructing the Karnaugh-Map first (shown below in Figure 2), selecting the optimal-minterms, then including a redundant minterm to avoid glitching. The redundant minterm is $A \wedge C$. This brings our function to $Z = (A \wedge B) \vee (\neg B \wedge C) \vee (A \wedge C)$. The reason this additional minterm was added in was to ensure redundancy and to allow the circuit to operate glitch-free.

An identical procedure to Part A was performed on Part B, and a 1MHz square-wave was connected to the select-line (B), while the A and C lines were driven to high.

a \ bc	00	01	11	10
0	0	1	0	0
1	0	1	1	1

Figure 2: Glitch-Free K-Map for Part B

3 Component Layout Sheet

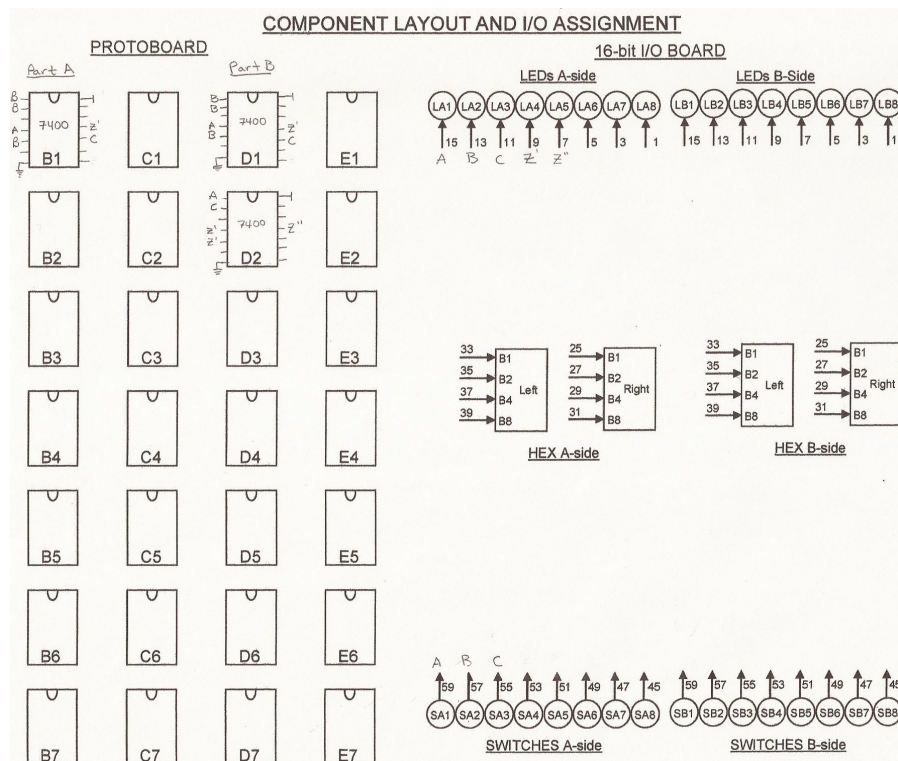


Figure 3: Component Layout Assignment and I/O Diagram

4 Logic Diagrams

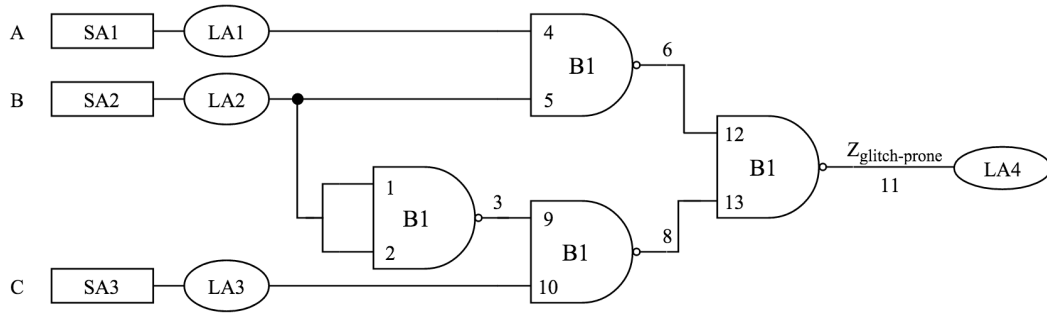


Figure 4: Glitch-Prone Logic Circuit

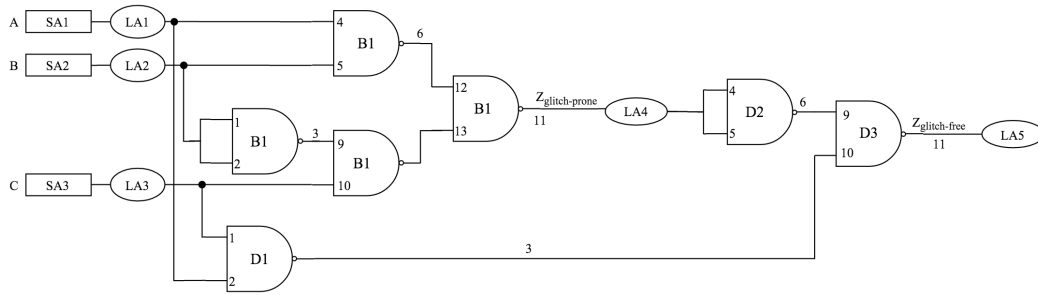


Figure 5: Glitch-Free Logic Circuit

5 Experiment Documentation and Results

The results of the experiment are documented below. The first two tables (Table 1 and Table 2) are truth-tables summarizing the outputs of the constructed circuits in both part A and part B. The remaining two figures (Figure 6 and Figure 7) are readings generated from the oscilloscope of the respective circuits.

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

Table 1: Truth Table for Glitch-Prone Circuit (Part A)

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

Table 2: Truth Table for Glitch-Free Circuit (Part B)

The first figure shows a static-1 hazard occurring at the falling edge of the 1 MHz square wave. A static-1 hazard is when the signal jumps/glitches down to a logic-0, when the actual output should instead be a logic-1.

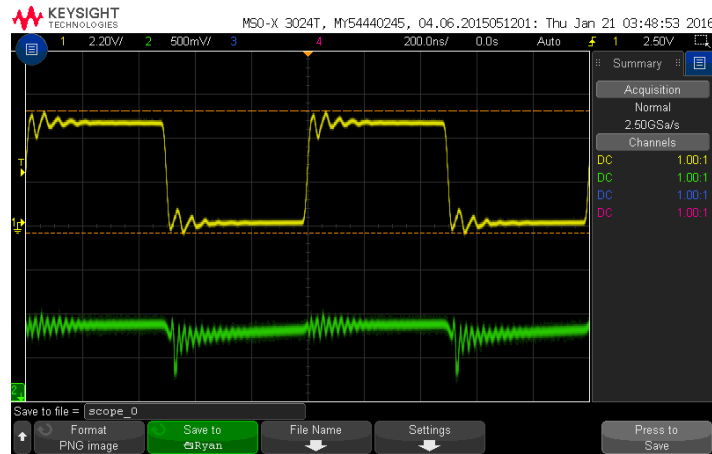


Figure 6: Glitch-Prone Oscilloscope Reading for Part A

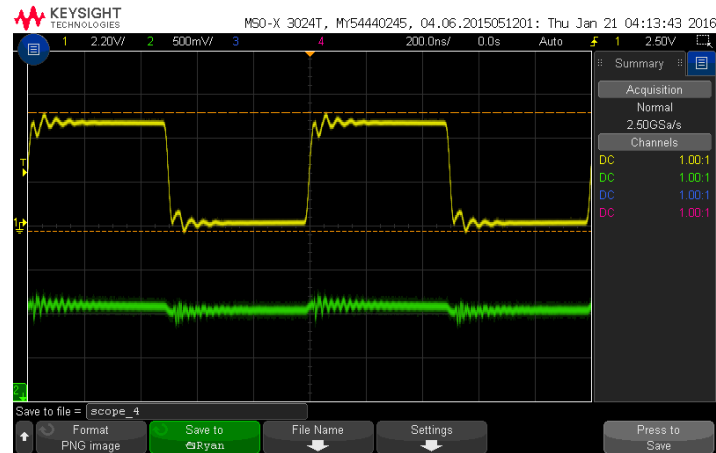


Figure 7: Glitch-Free Oscilloscope Reading for Part B

6 Post-Lab Responses

- 1.) Given that the guaranteed minimum propagation delay of a 7400 is 0ns and that its guaranteed maximum delay time is 20ns, complete the timing diagram below for the circuit of part A.

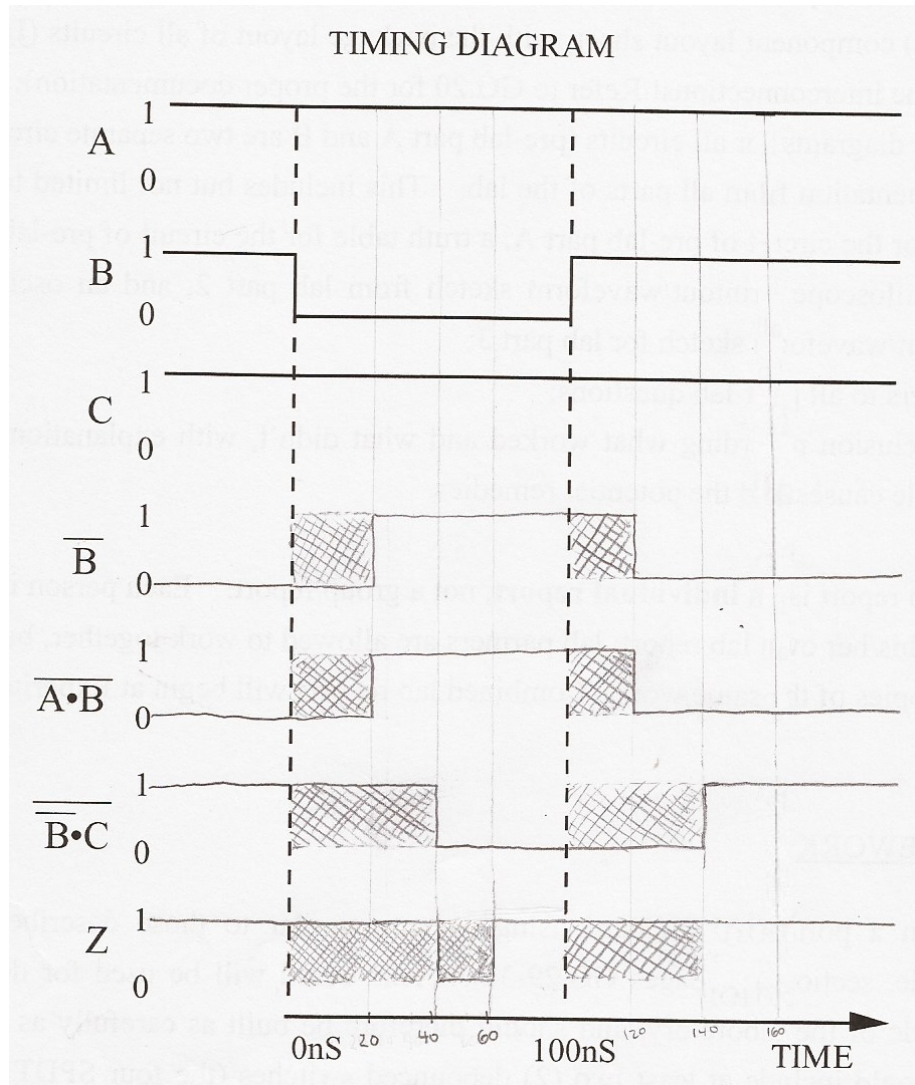


Figure 8: Timing Diagram for Part-A

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

It take approximately a maximum of 60 ns for the output Z to stabilize on the falling edge of B.

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

The output, Z, takes approximately a maximum of 40 ns to stabilize.

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

Yes. There are potential glitch (a static-1 hazard) on the falling edge of the input B, as it is transitioning from 1 to 0. This is due to the fact that the propagation delay time for the A & B is quicker than the propagation delay time for \tilde{B} & C. This is known as a race-condition.

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 a mechanical contact bounces are eliminated?

The debouncer circuit is an S-R Latch with pull-down resistors of $5k\Omega$. The S-R Latch has the special property that if either A = 1 or B = 1, it holds the stable state of either one. So, for example, if both A is 1 and B is 0, or vice-versa (which occurs when the switch is bouncing), the S-R NAND Latch holds the state of the previously contacted switch (so if the switch was previously at A (A = 1, B = 0), then it would hold the state of A=1 and B=0 - the same for the opposite case). This is because the output on the top NAND gate is feedback into the input of the bottom NAND gate and the same is done for the bottom NAND gate.

7 Conclusion

As stated previously, both glitch-prone and a glitch-free 2-to-1 multiplexer TTL circuits were constructed in this lab. The circuit constructed in part A was prone to glitches (specifically a static-1 hazard) during the transition from 1 to 0 (on the falling edge) and this is shown in Figure 6 (the oscilloscope output). This is due to the propagation delay on TTL chips and depending on the number of NAND gates the signal must pass through to reach the output. These race-conditions must be eliminated at all costs due to their potential risks on a system.

Another interesting experiment to perform would be to analyze the propagation delay and glitches using different chips and complex circuit arrangements. However, overall, for the purposes of this lab, many properties can be understood about TTL circuits and static hazards.