ECE 385

Fall 2021

Lab 1

Static Hazards

Matthew Fang

ABE

Abigail Wezelis

Introduction:

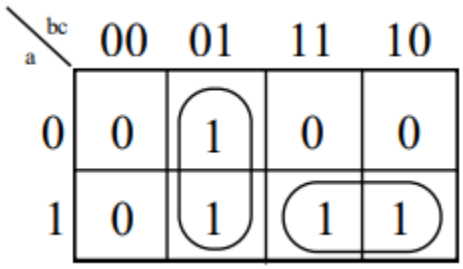
Static hazards are incorrect outputs produced by digital logic circuits when the input logic changes. Because each logic gate has a certain delay, this can lead to short periods of discrepancy between the expected output value and the actual output value as the input data travels through the logic process.

Static hazards may not always be apparent because the glitch can be very small depending on the size of the circuit and how many transistors different inputs must go through before an output can be processed. To make these hazards more apparent, an odd number of inverters can be chained together in place of a single inverter, or a small capacitor can be added to the output. Both examples here would increase the time needed for the input to be processed, which in return allow for the static hazards to show up.

In order to analyze these static hazards, I implemented a circuit given by the karnaugh map and truth table shown below.

Documentation:

Karnaugh map:



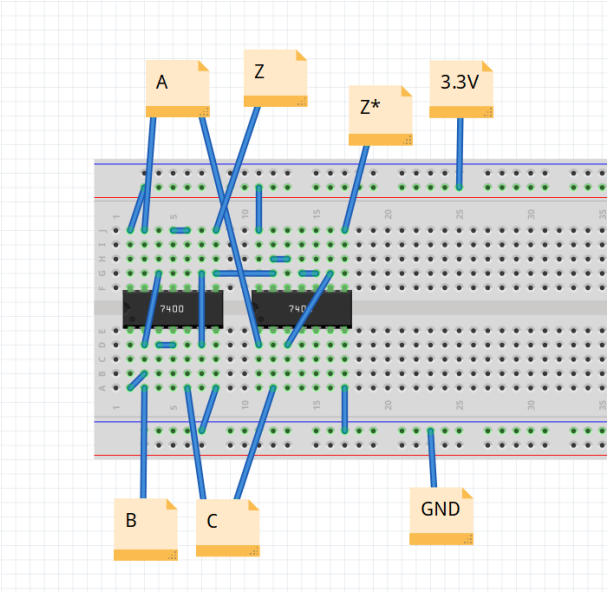
Truth Table:

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

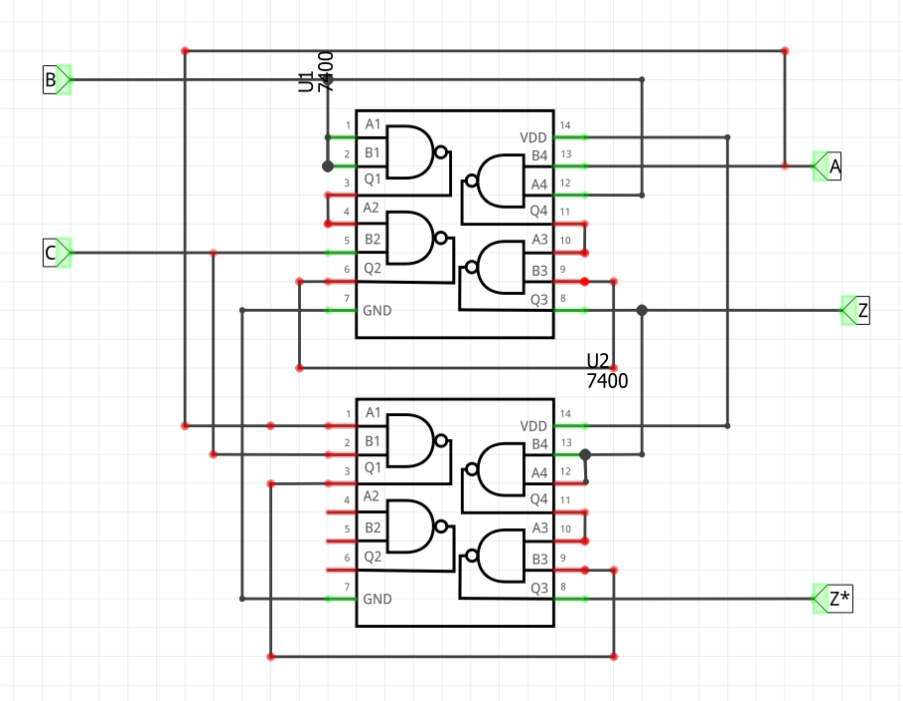
Circuit operation:

The circuit I designed outputs Z for given inputs A, B, and C. However, it is prone to static hazards. To prevent static hazards, I cover all adjacent min terms on the karnaugh map, in this case AC. In the following circuit documentation, Z is the output with static hazards while Z* is the output with minimized static hazards.

Fritzing breadboard:

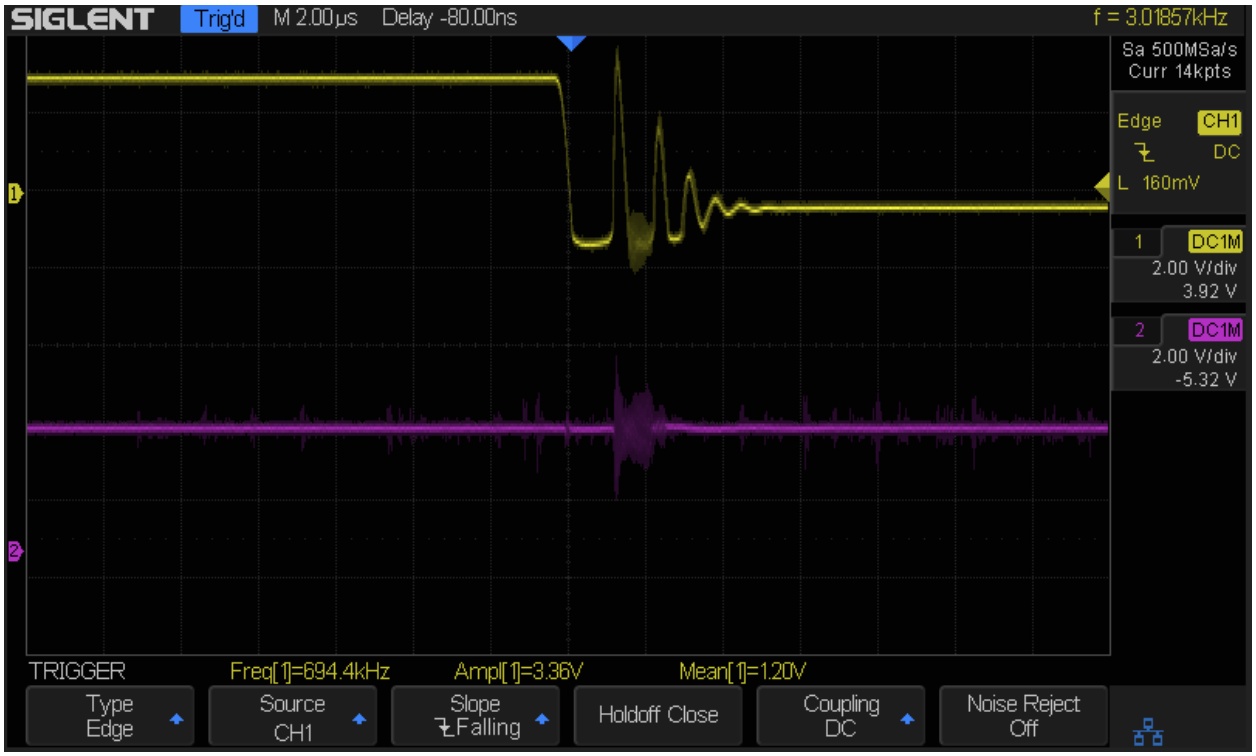


Fritzing schematic:



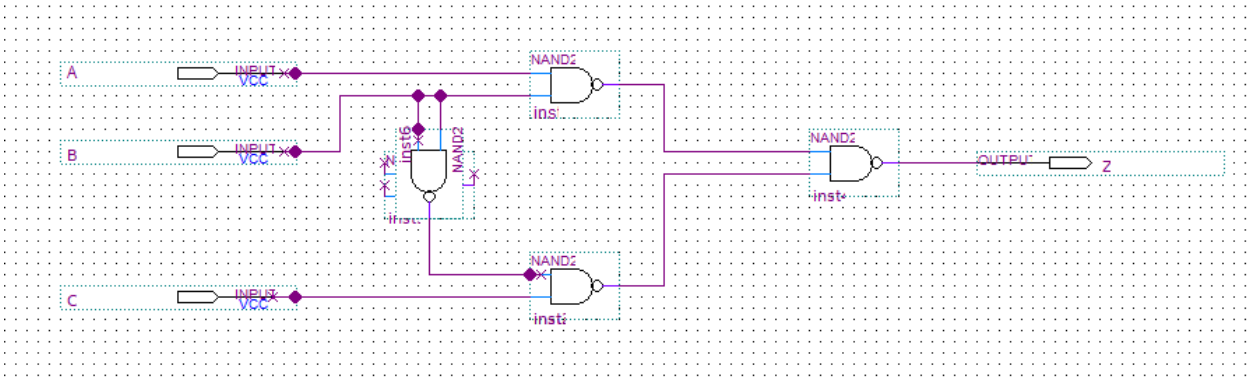
Z output (static hazard):

Z* output (no static hazard):

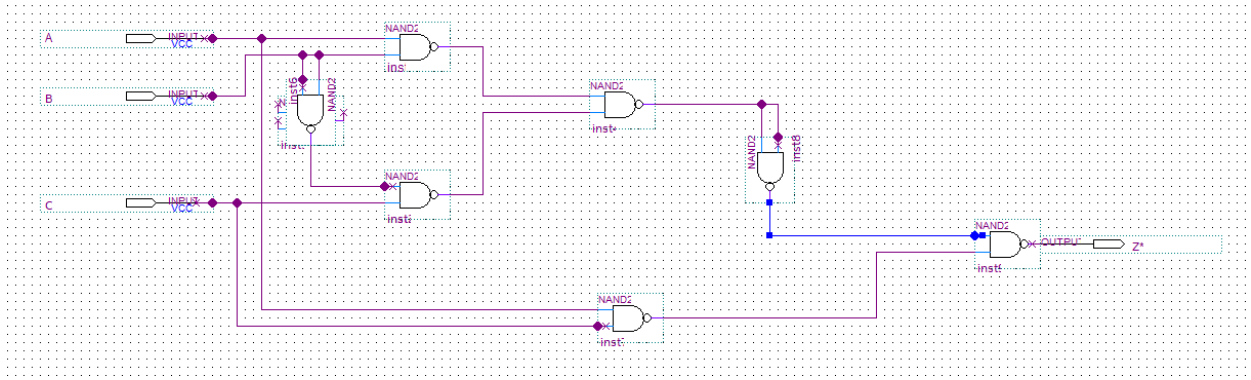


Logic Diagrams:

Z output (static hazard)

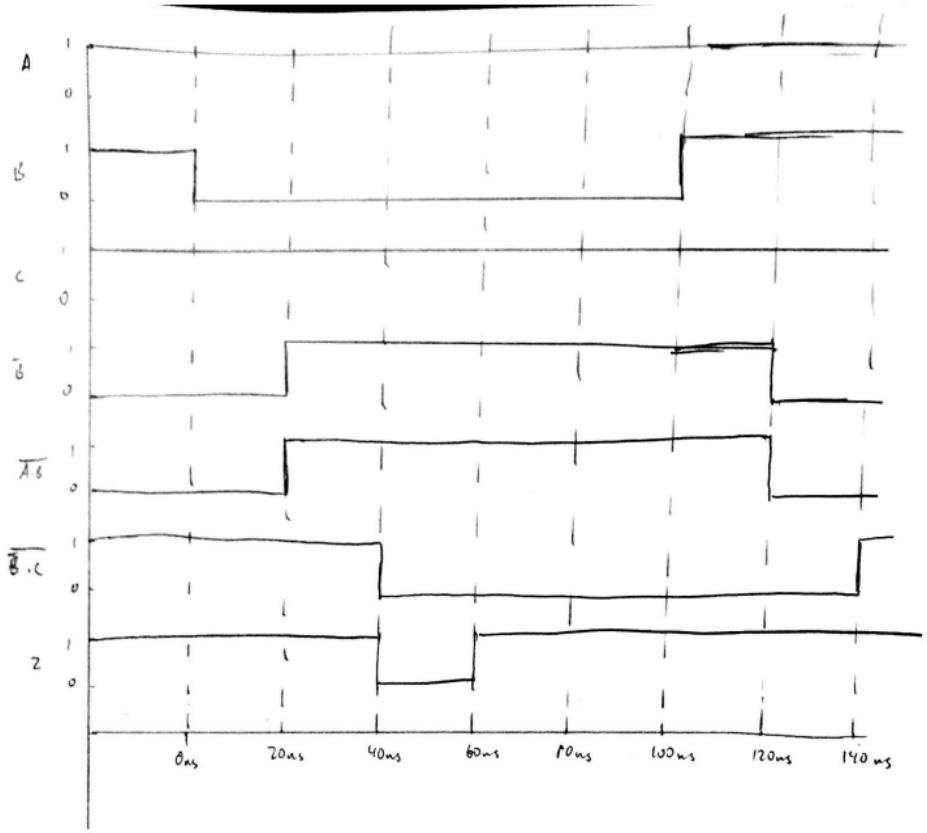


Z* output (no static hazard)

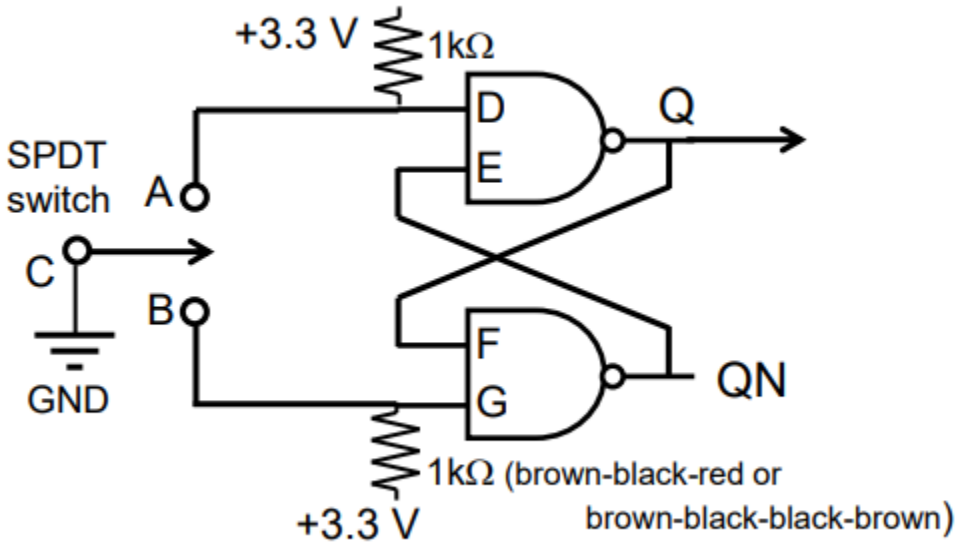


Post-lab:

1) If the guaranteed minimum propagation delay of a 7400 is 0 ns and its maximum delay time is 20 ns



2) Debouncer circuit



When normal mechanical switches are closed, the electrical contacts of the switches are slammed together, which causes the contacts to momentarily lose contact and "bounce." For a digital logic circuit, this bouncing can cause errors as multiple signals are sent which may cause counters to advance more times than desired. The debouncer circuit, shown above, eliminates the bouncing issue.

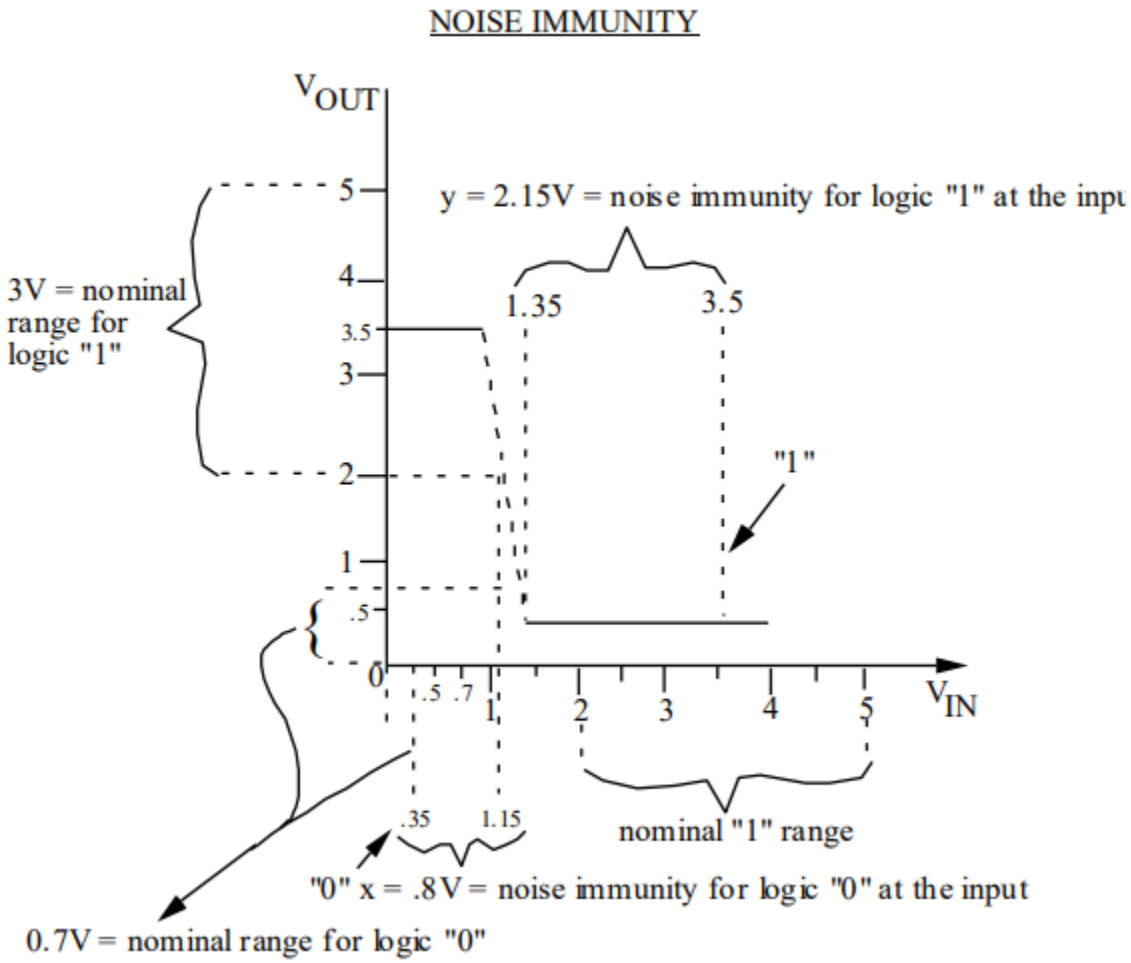
When the switch is flipped, both A and B are momentarily high, due to the pulling high design of the circuit. When the switch connects, one of the two becomes grounded, let's say A. Because NAND outputs 0 only if both inputs are 1, Q becomes 1 and QN becomes 0. When the switch is flipped to B. QN becomes 1 and Q becomes 0. Because an output of one NAND is the input of another NAND, when the bounce occurs, the 0 from either Q or QN keeps the output of the other NAND gate stable for the brief moment that A or B bounces between high and low.

<u>General Guide:</u>

GG.6

The advantage to larger noise immunity is that the circuit won't be as affected if the digital logic has to travel through a lot of components. Other components used in the circuit will cause voltage to drop, which may affect the actual voltage to deviate from the desired "high" or "low" voltage level. Even the logic gate chips contain small resistors, which also contain voltage drops.

To determine the logic “0” and “1,” several inverter gates can be connected in series and the last inverter's voltage level can be observed. The reason the last inverter is measured instead of the first is so that the signal can go through multiple gates, and thus experience more noise.



To calculate noise immunity, find the input of a gate. Add a positive pulse to a “0” voltage level until the value at the output changes. The value of the positive pulse just before the output changed is the noise immunity. On the graph, the “1” voltage level is 3.5. When 1.35V is imputed into the inverter, the “0” V level is reached, showing that the noise immunity for logic “1” is 2.15V. On the other hand, when logic “0” is the input, the output doesn’t start changing until .7V, meaning .7-0 = .7V is the noise immunity for logic “0.”

GG.31

It is bad practice to share resistors because the purpose of a resistor is to limit the current that will flow through the LED. When a resistor is shared, different amounts of resistors will flow through the diodes, which may clamp the signal voltage.

Conclusion:

When working with sensitive circuits, it is important to make sure there are no static hazards. Static circuits could create brief unwanted outputs which can affect the rest of the circuit because of the discontinuous output. These hazards can be fixed by adding adjacent minterms to the karnaugh maps, however, the trade off is a more complex circuit. Switches can also be a problem with sensitive circuits as usual mechanical switches will bounce before achieving a steady output. This can be fixed with hardware by including a debouncing circuit, which again makes the circuit more complex.