ECE 385

Fall 2019

Experiment #1

Introductory Experiment

Eric Dong, Yifu Guo (partner)
Section AB3, Thursday 2pm
Gene Shiue

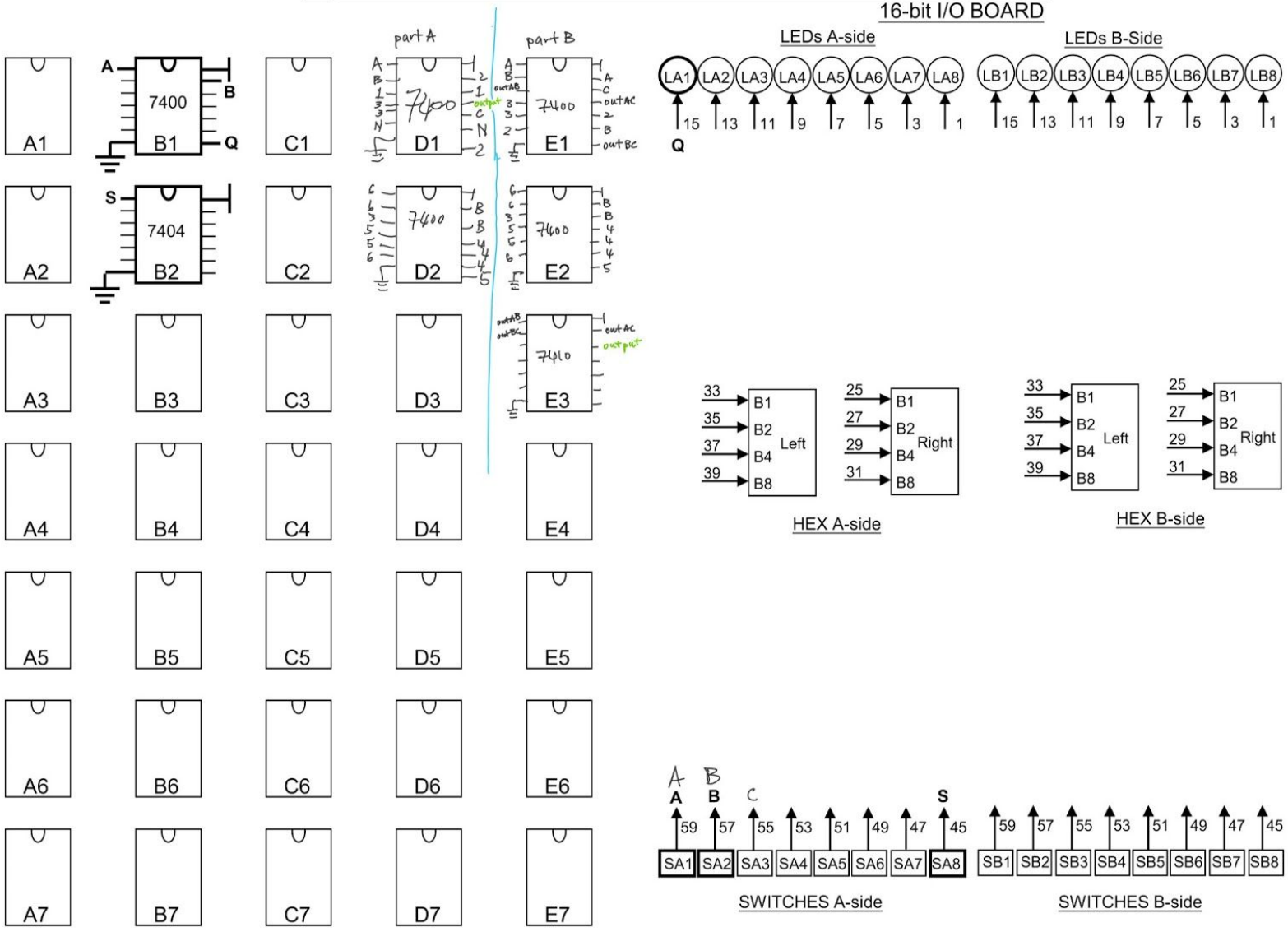
Introduction

For this lab we built a circuit to visualize a static hazard that is caused by gate delays. This lab shows us that even with the simplest of circuits, a MUX, may not be ideal and can have imperfections, which goes to show that we can not base our designs on ideal situations but rather we need to take into consideration of real life events. For a processor architecture to be stable, we have to consider every possible imperfection that may impact the performance of chip.

Written description

The MUX that we designed consisted of 4 NAND gates, and 3 inputs, A, B, and C with B being the selector pin. The boolean expression of the MUX is ((B'C)'(AB)')'. The problem with the MUX was that it had a static hazard whenever we would switch B on from logic 1 to logic 0. To visualize this static hazard, we had to add 4 more NAND gates in series to B'. This increases the gate delay and we can see clearly on the oscilloscope that there's brief static hazard. In order to resolve this issue, we added another minterm (AC) so the resulting boolean logic would be ((B'C)'(AB)'(AC)')'. By adding a (AC)' term, it prevents a jump between the two minterms which ensures that we don't get a static hazard since (AC)' will stay 0 which prevents the output Z from dropping to 0 as well.

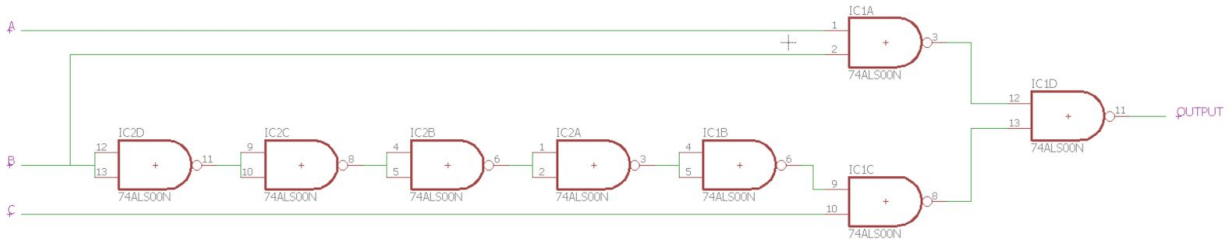
Figure 8. SAMPLE COMPONENT LAYOUT AND I/O ASSIGNMENT



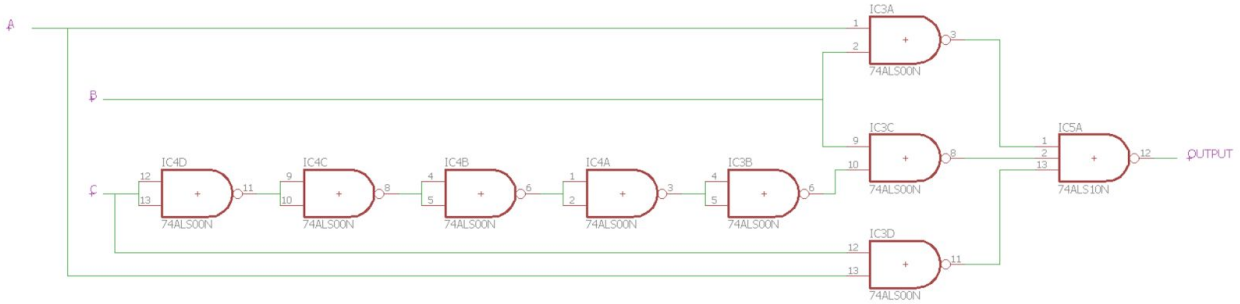
Component Layout Sheet

Circuit Diagrams

Prelab part A



Prelab part B



Documentation

Part A Truth Table

A	B	C	OUT
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

PreLab Questions

1. Not all groups may observe the static hazards since the gate delay on each chip may vary between the maximum 20ns or the minimum gate delay which is 0ns. Also the capacitance at the end of the probe to the oscilloscope also may affect the observation of the static hazard.
2. The static hazard appears after you chain multiple gates together because you are adding up the gate delays until it is obvious enough to observe on the oscilloscope.

Part B truth Table

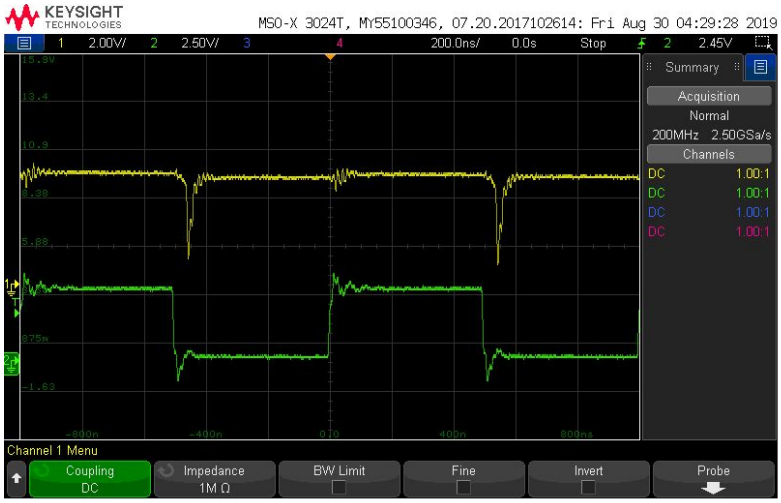
A	B	C	OUT
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Lab Questions

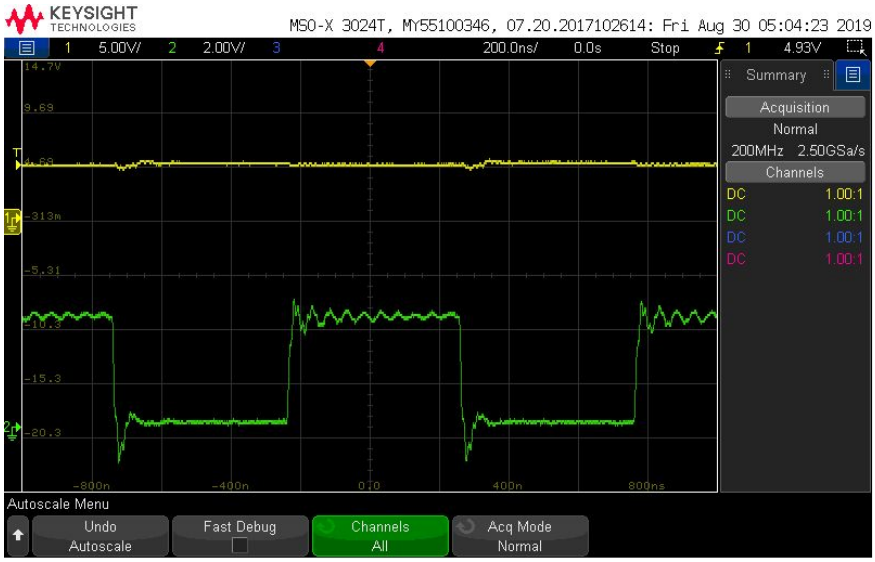
1. The truth table of part B is the same as part A's truth table. With the improved circuit, there are barely any significant differences between part B's result and the predicted result from the truth table. For the circuit of part A, we are more likely to observe the glitch at the falling edge of the pulse generator since at this point, (AB)' and (B'C)' are both logic 1 and if we NAND those two together we will get a logic 0 at the output.

Oscilloscope Pictures

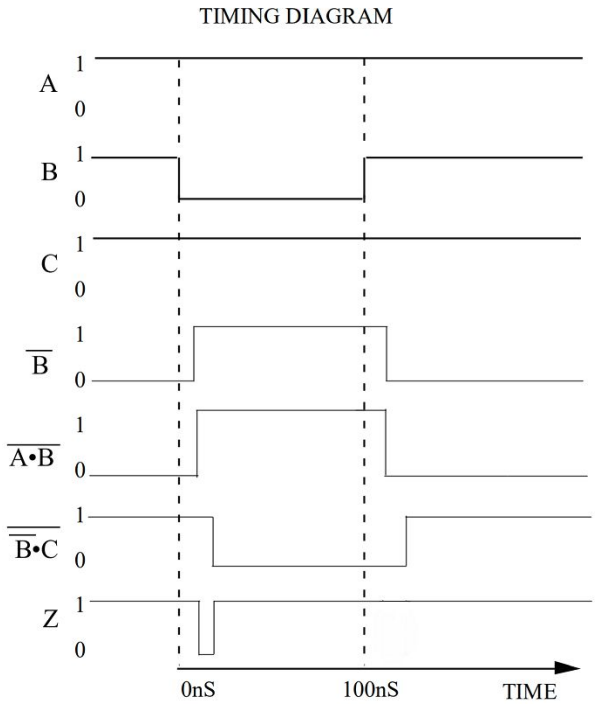
Static Hazard



Redesigned Circuit



Post Lab Questions

1. 

   On the falling edge of B, it takes Z about 0 to 40ns to stabilize, and on the rising edge, it takes Z also around 0 to 40ns to stabilize. There are static hazards that can occur on the output Z. These glitches occur due to the gate delay of the extra NAND gate that inverts B, so momentarily on the inputs of the NAND gate that outputs Z, there is a logic 1 and a logic 1 which outputs a logic low for Z which is the glitch.

2. The debouncer circuit is like a memory latch. Let's assume that Q, F are 0, and D, E and G are 1. When C connects to A, then D becomes 0 and thus Q becomes 1 and remains a 1 until C is connected to B. This debouncer works because it uses feedback, which means once C is connected to Q, Q would stay 1 until C is connected to B. So even if the contacts of C loses connection with A, unless it touches B, Q will stay 1. The debouncer acts like a switch in the sense that it has a mechanical component and can output either a logic 1 or a logic 0. The ill effect of the mechanical switch is eliminated since even if the switch bounces and no longer remains in contact with A, the output stays a 1 instead of jumping from 1 to a 0.

General Guide Questions

GG.6

1. The advantage of having a larger noise immunity is when your input is noisy, the chip will still be able to read the correct values and output the right logic. The last inverter is observed so that if there were to be any noise in the inputs, they will no longer be present at the output. To obtain the noise immunity of the chip, "we use the smallest of the ranges X and Y", with X being the noise immunity for logic 0 and Y being the noise immunity for logic 1 both being input voltage(VIN). To calculate X, we use the x coordinate of the point that would result in the lowest logic 1 voltage, subtract that by half the nominal range for input logic 0. To calculate Y, we use half the nominal range for input logic 1 subtracted by the lowest input voltage that would result in a logic 0 output.

GG.29

1. Real LEDs have different characteristics, and so when you share a resistor and add many LEDs in parallel some LEDs may not conduct while others may have too much current flowing through it. Different LEDs can have different forwarding bias, even if they are the same color LED they can still have slight differences which might cause some LEDs to not even turn on.

Conclusion

At the end of this lab, everything was working fine but we did run into several roadblocks. At the beginning of the lab, while we were trying to detect the static hazard on the oscilloscope, we were unable to identify it since there weren't enough NAND gates. After we added 4 more, we were able to clearly identify the static hazard. We weren't able to observe the static hazard in the first place because there wasn't enough NAND gates to cause a significant gate delay, but afterwards we could see it.