ECE 385

Fall 2020

Experiment #1

Introductory Experiment

Xiao Shuhong

3180111551

9/21

E201

Total Introduction

This lab, as the first of total nine ECE385 labs, act as an introduction to both laboratory and equipment. more specific, have a basic idea about how to use the Quartus prime, to be familiar with integrated circuit(chips) especially 7400 series, rise the idea real world design are not as perfect as ideal way, such as possible glitches, and know how to eliminate it.

Purpose of Circuit

Circuit of this lab is just a simple example that help student realize the real-world situation of how electronic circuit performance are different from ideal case. To be more general, how delays act on signal to cause static hazards(glitches). Also, students should analysis which point may cause potential glitches as well as seek the solution about how to eliminate them,

Description of Circuit

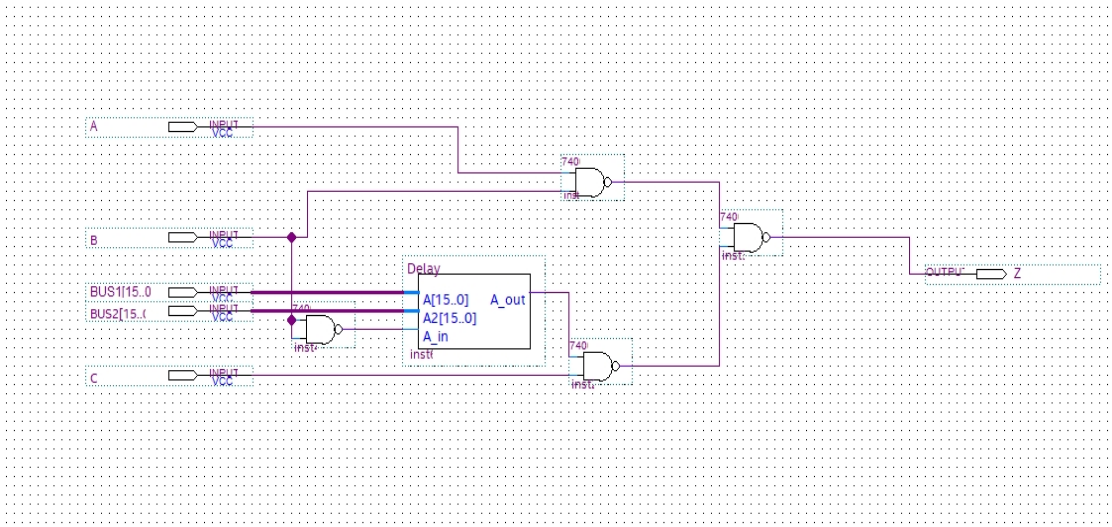
The circuits in both A and B, act as a 2 to 1 mux, which accept two input signal and one select signal and decide one of the two input to be output based on the choice of select signal. The difference of these two parts is part A is built with the simplest expression generated by K-map but would cause potential static hazards in real word. While part B is a strength of A which eliminate the static hazard and achieve a much reliable behave.

For part A, choose A and C as two input, B as select signal and Z as output. the expression is Z=AB+B'C, which need four two-input NAND gate to build. So, one 7400 chip (generate four NAND gates in one chip) should be used. When B=1, input A is selected and output, otherwise, input C is selected and output. When signal B change, due to the time delay, glitches may happen.

For part B, the idea to eliminate glitches happened in A is to add term AC to our expression, that is, Z= AB+B'C+AC. Now, totally six two-input NAND gates are needed, which is two 7400 chips.

Circuit diagrams and Documentation

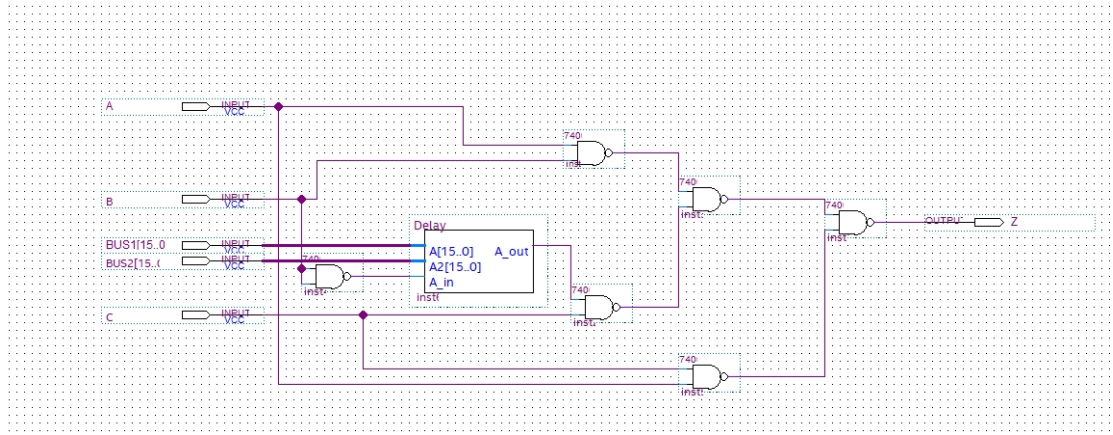
Part A:



Truth Table of 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

Part B



Truth Table of B

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

question in pre-lab

1. No groups may observe static hazards (why?)

as the delay of a single NOT (NAND) is still too small to be observed

2. if you do not observe a static hazard, chain an odd number of inverters together in place of the single inverter from Figure 16. But no static hazard is find, why?

for Quartus prime, when we do the waveform simulation, it automatically optimizes the circuit according to the circuit principle. That is, odd number of inverters may be optimized to just one inverter, and could not offer enough time delay as before.

question in lab

1. Does it respond like the circuit of part A? Describe and save the output and explain any differences between it and the results obtained in part 2.

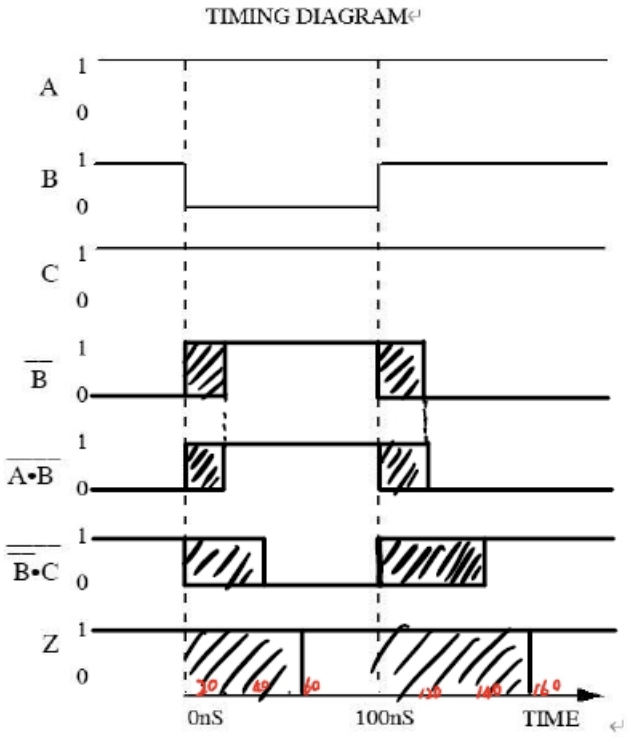
waveform in part B act different as A, for all the glitches shown in part A disappear and output Z in part B just show high (1).

2. for the circuit of part A of the pre-lab, at which edge (rising/falling) of the input B are we more likely to observe a glitch at the output?

a glitch more likely to be observed at falling edge, that is, from B=1 to B=0. Because the part B'C has an extra adder(to enhance the delay) to go through than part AB, that is, for A=1,C=1, initially B=1, AB=1,B'C=0. When B suddenly change to 0, AB change faster than B'C, so in a short time we have B' still 0 and B'C 0.

question in post-lab

1. complete the timing diagram below for the circuit of part A



2. 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.

it takes at least 0ns, at most 60ns for output Z to stabilize on falling edge of B

it takes at least 0ns, at most 60ns for output Z to stabilize on rising edge of B

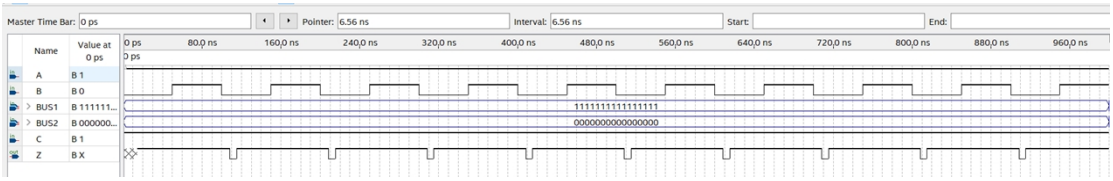
yes, glitches may happen between 0 to 60ns and 100ns to 160ns

when B is falling at t=0ns, initially B=1 and B'=0, when B tend to be 0, change of B' maybe later than B because of the time delay, so at this time, B=0 and B'=0, in a short time(happen in 0ns to 60ns), Z will change to 0. Also, when B rising at t=100ns, initially B=0 and B'=1, because of the same delay, there is a short time interval happen between 100ns to 160ns, when B change to 1, we have B=1 and B'=1, cause Z to be 0.

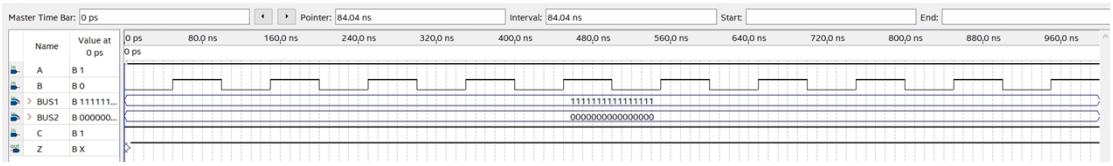
That is, if the length from different input to output is different, the different time delay may cause glitches.

Simulation Result

for part A



for part B



we can see clearly see that after AC part is added into our expression, the glitches happened in part A is eliminated in Part B

Answer Question from GG

1. advantage of large

as digital signal use two different amplitude levels to represent HIGH and LOW. Basically HIGH=5V and LOW=0V. If a noise is to be superimposed on the signal, the small voltage fluctuations may change the signal level from one to another. That is, a larger noise immunity can tolerate stronger noise while not cause the signal to change unwillingly.

2. why observe from last inverter rather than the first one

while the input contains noise or is not good as we thought (around 0V and 5V), then the output of first inverter may be not falling on the flat part but shaded part from the figure. for example, input with noise have peak around 0.8V, the output will fall between 2-3V and even worse. that is, we should invert it several times to make sure our output finally fall at flat part which will be the logic 0 and 1 we want.

3. how to calculate noise immunity for the inverter

consider the figure, the two flat part give us the two logical output we want (y=3.5V and y=0.35V).

for high voltage(define between 2 to 5V) give us logic 1, the input should be constrained between 0 and 1.15V, otherwise, the output will be lower than 2V and output logic 0 rather that logic 1. that is, noise immunity for logic 0 at the input is 1.15-0.35=0.8V.

with the same idea, voltage low is defined smaller than 2V, that is, the input should be no smaller than 1.15V to make sure we output logic 0 rather than 1, so noise immunity for logic 1 at the input is 3.5-1.35=2.15V.

4.bad practice of share resistors for LED

share resistor for more LED cause these LED to be parallel with each other, then cause the voltage drop on resistor to increase, the current may become too large and destroy the circuit. In addition, the voltage on LED may become too small to light it on.

Conclusions：

In this lab, actually help us establish the concept that a group of extra consideration should be taken in real world design. In this lab, for example, how to eliminate the glitches. the core concept is the delay of circuit may not be ignored. in this lab, we solve it by adding the common items for overlapping term observing from K map, but in product manufacture,

this is still not a applicable approach, as the extra part may cost more energy and money and may have other problems. so, actually the problem I have at the end of lab 1 is, what is the industry method to eliminate or avoid such glitches when build circuit? are they use the idea lake apply latches to hold value when glitches happened or some else?

in addition, in lab 1, we have the basic idea how to design a circuit using professional software like Quartus Prime, have the skill to debug our circuit by simulation with wave form.