Combinational Logic Circuit Design: Date of Birth

By Om Patel

Table of Contents

Client:	Local Company
Designer:	Om Patel
Problem Statement:	Given the worldwide birthrates result in a relatively high likelihood of people sharing the same date of birth, called the “Birthday Paradox”. To prove this theory, a company wants a display that showcases a person’s birthday so people can compare.
Design Statement:	Create a circuit that is capable of displaying a person’s date of birth in a seven-segment display, which should accept a variety of inputs to display a person’s month, day, and year. The circuit must only use AOI, NAND, and NOR logic.
Constraints:	Must only use resistors of 150 Ω - 270 Ω Must only use 2-input gates The Karnaugh mapping technique must be used to obtain the simplified logic expression for each of the seven segments. One segment must be implemented with NAND only logic One segment must be implemented with NOR only logic Must be completed by Friday, October 20th, 2023
Deliverables:	Design Brief Proof of working circuits A simulation of my circuit in MultiSIM A simulation of my circuit in TinkerCAD A prototype of my circuit on a breadboard Justification as to the type of circuits used Breadboard must be able to display 82-73-64 Project Reflection which describes the process I took to design my circuit and any errors/problems I faced A portfolio which describes the process I took to design my circuit

Design Brief

In this project we were challenged to build a display that showcases a person’s birthday so people can compare them, testing the "Birthday Paradox"

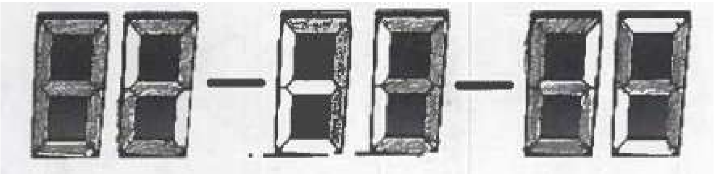
Creating a design brief helps organize my work, helping me see what I have to do and allowing others to see what I have done.

Truth Table

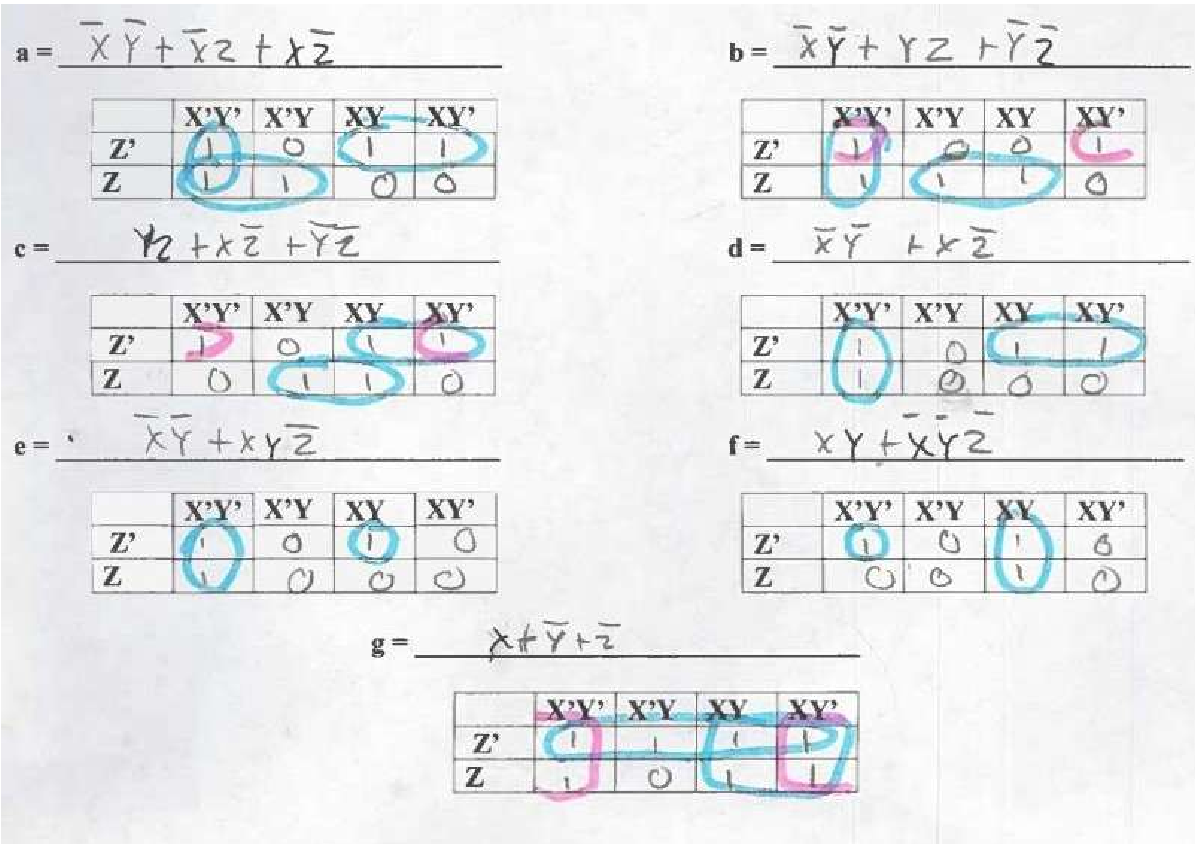
A truth table is a diagram of all possible inputs in a system and their outcomes. If the output of the chart is 1 then the decision in this scenario is positive.

In this project, I used this chart to create my logic equation, to turn off or on each section for each individual combination of inputs.

I also used my truth table to check whether anything I simulated, built, or drew works.



Count	X	Y	Z	Date	A	B	C	D	E	F	G
1	0	0	0	8	1	1	1	1	1	1	1
2	0	0	1	2	1	1	0	1	1	0	1
3	0	1	0	-	0	0	0	0	0	0	1
4	0	1	1	7	1	1	1	0	0	0	0
5	1	0	0	3	1	1	1	1	0	0	1
6	1	0	1	-	0	0	0	0	0	0	1
7	1	1	0	6	1	0	1	1	1	1	1
8	1	1	1	4	0	1	1	0	0	1	1



a = $\bar{X}\bar{Y} + \bar{X}Z + X\bar{Z}$

	X'Y'	X'Y	XY	XY'
Z'	1	0	1	1
Z	1	1	0	0

b = $\bar{X}\bar{Y} + YZ + \bar{Y}\bar{Z}$

	X'Y'	X'Y	XY	XY'
Z'	1	0	0	1
Z	1	1	1	0

c = $YZ + X\bar{Z} + \bar{Y}\bar{Z}$

	X'Y'	X'Y	XY	XY'
Z'	1	0	1	1
Z	0	1	1	0

d = $\bar{X}\bar{Y} + X\bar{Z}$

	X'Y'	X'Y	XY	XY'
Z'	1	0	1	1
Z	1	0	0	0

e = $\bar{X}\bar{Y} + XY\bar{Z}$

	X'Y'	X'Y	XY	XY'
Z'	1	0	1	0
Z	1	0	0	0

f = $XY + \bar{X}\bar{Y}\bar{Z}$

	X'Y'	X'Y	XY	XY'
Z'	1	0	1	0
Z	0	0	1	0

g = $X + \bar{Y} + \bar{Z}$

	X'Y'	X'Y	XY	XY'
Z'	1	1	1	1
Z	1	0	1	1

Karnaugh Mapping

Using a method called Karnaugh Mapping also called K-Mapping I can easily derive a simplified equation from a set of outputs.

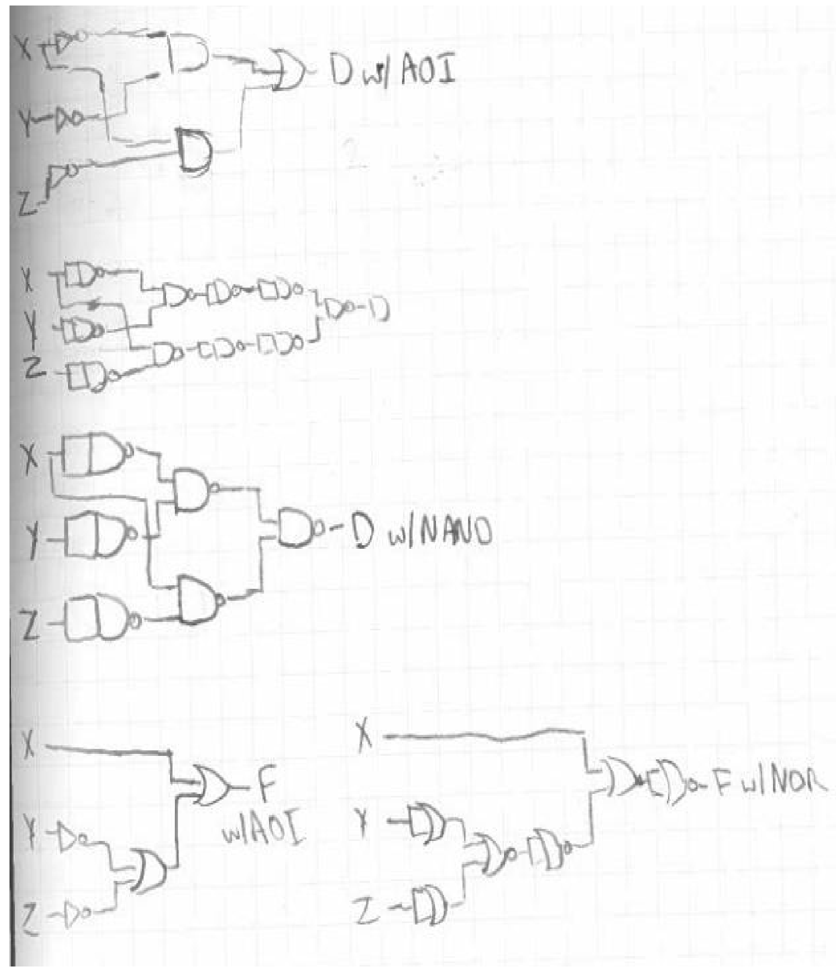
To set up each of my K-Mapping tables I used the outputs from the columns labeled A-G.

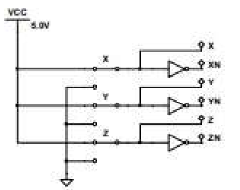
To create the equation from a table you have to group 1s in a way to eliminate any excess inputs.

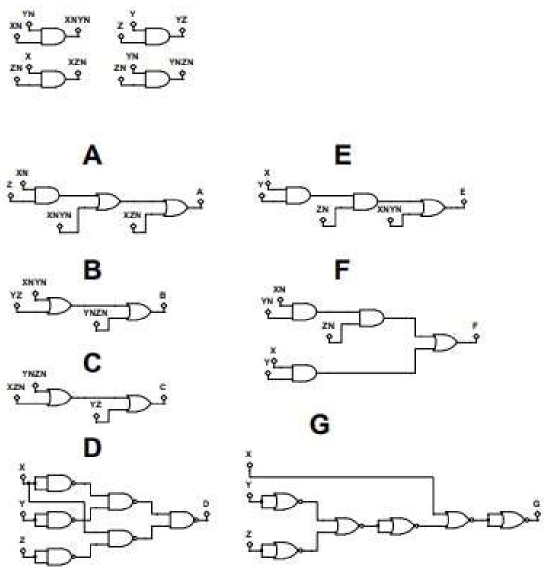
NAND & NOR Sketch

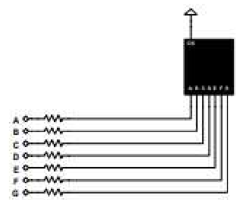
NAND and NOR circuits are beneficial as they can allow for a decrease in the use of IC chips, in turn allowing for a faster and more efficient chip.

To create a circuit with NAND or NOR logic I first created the AOI logic circuit from the derived logic equations. This then allows me to derive its NAND/NOR versions due to their AOI equivalents.









MultiSIM Sketches/Prototype

With my NAND and NOR circuits sketch I could then test them using a program called MultiSIM.

I also sketched my AOI circuits in MultiSIM allowing me to test them faster.

In this schematic, I separate the most repeatedly used AND gates to make my circuit more efficient.

With all my subcircuits complete I could then wire them to a seven-segment display to see if they work in sequence.

TinkerCAD Prototype

Utilizing a digital breadboard software called TinkerCAD I can visualize what a breadboard of this circuit would look like without the repercussion of physically making it.

To color code I made my signal:

Z - White

Y - Orange

X - Yellow

Z' - Green

Y' - Light Blue

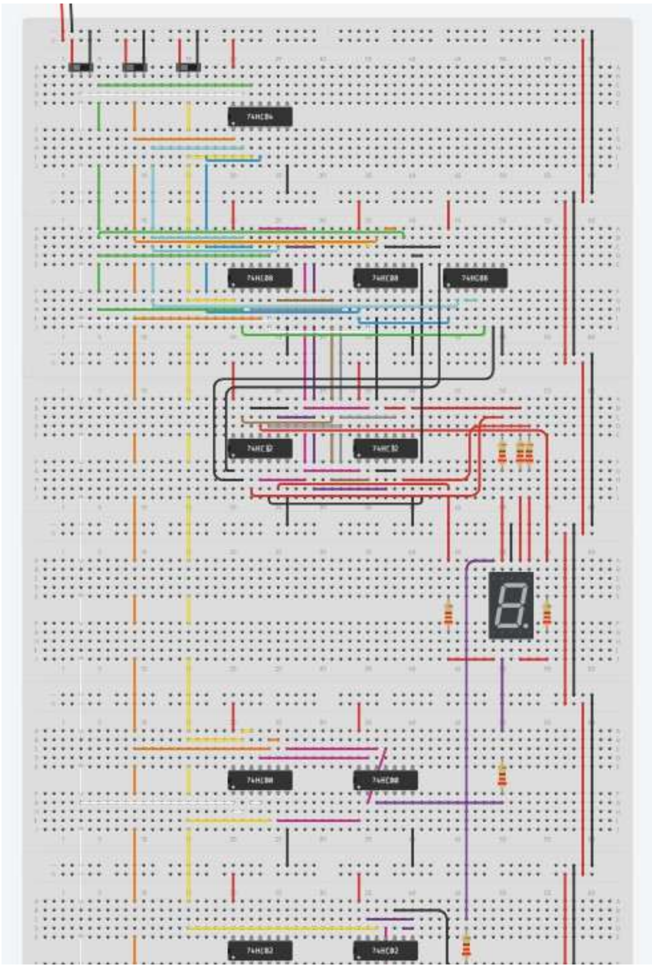
X' - Blue

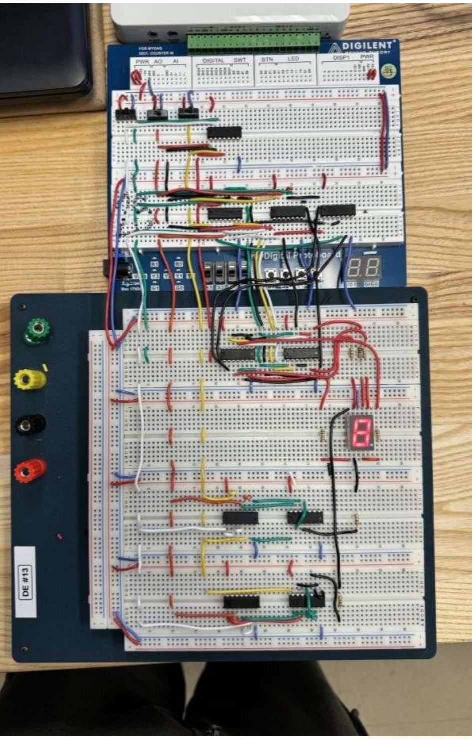
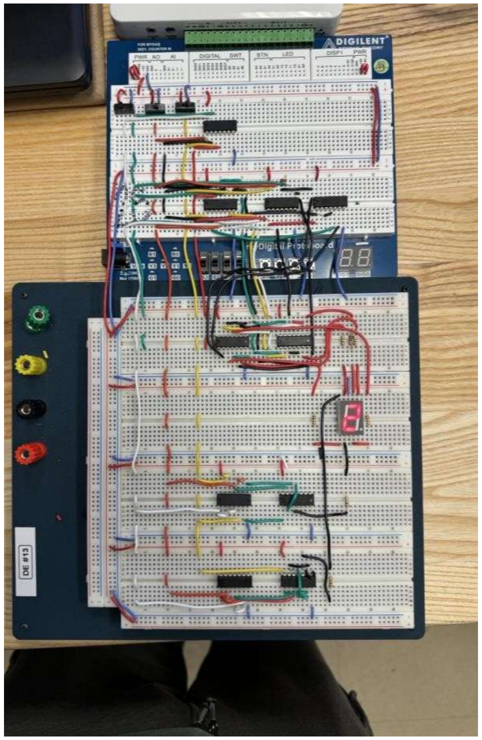
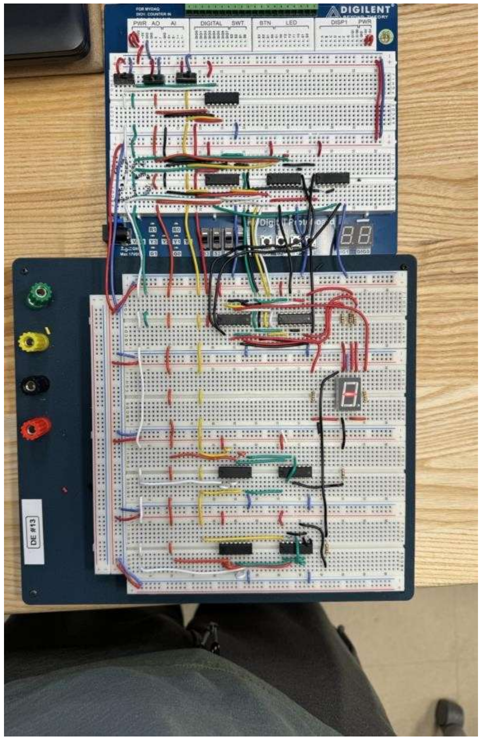
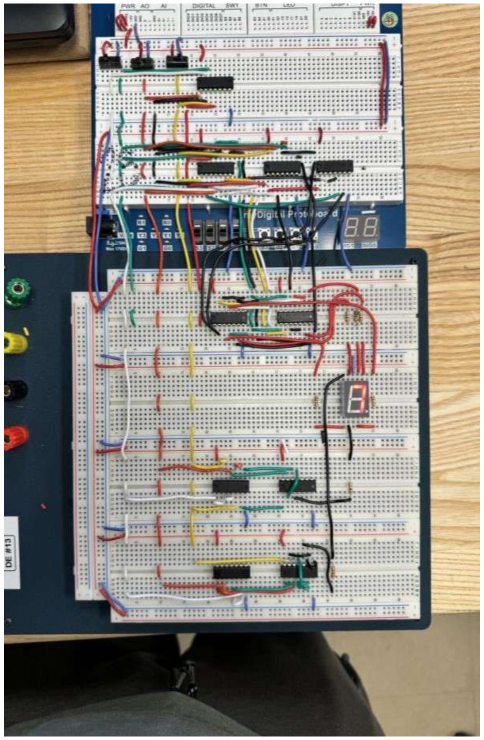
X'Y' - Pink

XZ' - Brown

YZ - Grey

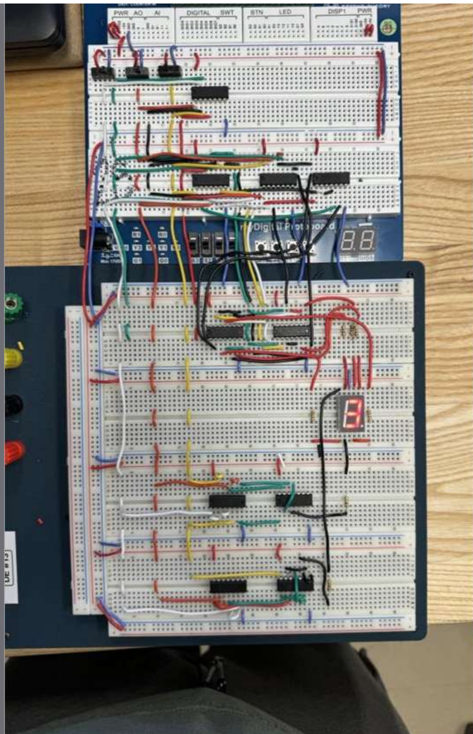
Y'Z' - Purple

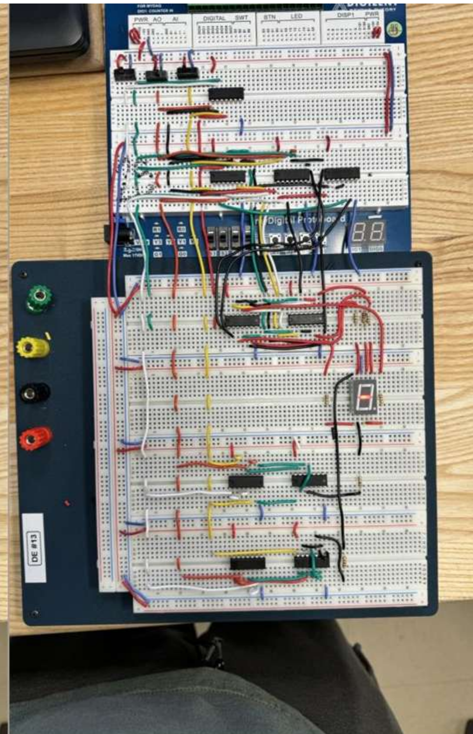


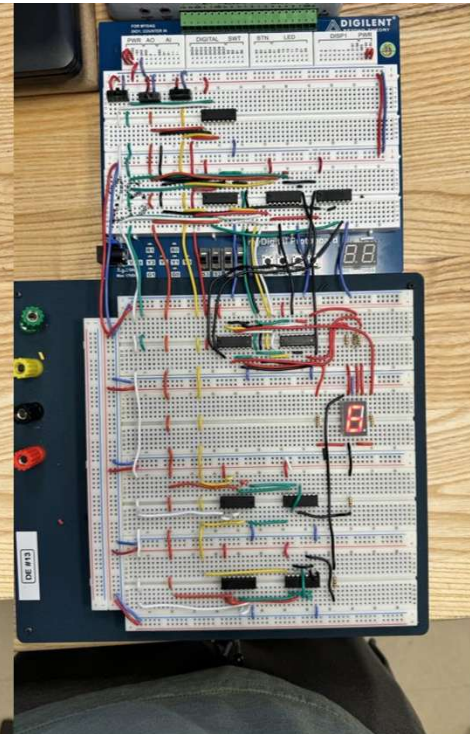





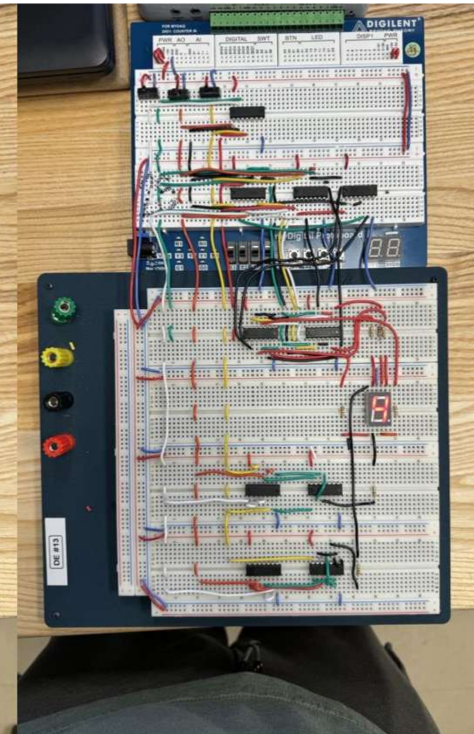
Breadboard Prototype

Using my TinkerCAD simulation and the building techniques that I've learned I created my circuit in real life using a breadboard. The prototype worked flawlessly with no errors, displaying "82-73-64", just as I intended.









Breadboard Prototype