Faculty of Engineering – Ain Shams University

Fall 2021

CESS – Sophomore

CSE111: Logic Design

Major Task

Presented to

Dr. Hossam AbdelMunim

Eng. Kyrelloss Nashaat

Presented by

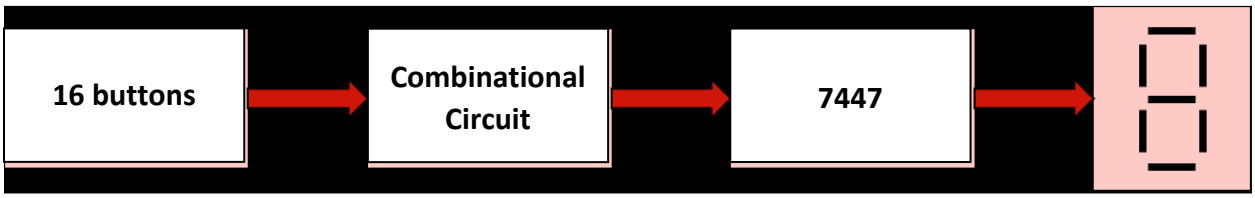
Ahmed Galal

Ahmed Tarek

Mina Fadi

Phase 1

The purpose of the circuit shown in the figure is to display the pressed key on the 7-segment display, with the aid of a combinational circuit designed between the 16 buttons and the BCD to 7-segment display decoder 7447.



We first started by the truth table of the 16 buttons, and the BCD numbers as outputs (functions)

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Enable	W	X	Y	Z
1	0	0	0	0
1	0	0	0	1
1	0	0	1	0
1	0	0	1	1
1	0	1	0	0
1	0	1	0	1
1	0	1	1	0
1	0	1	1	1
1	1	0	0	0
1	1	0	0	1
0	X	X	X	X
0	X	X	X	X
0	X	X	X	X
0	X	X	X	X
0	X	X	X	X
0	X	X	X	X

The resulted functions

W = 8 + 9
X = 4 + 5 + 6 + 7
Y = 2 + 3 + 6 + 7
Z = 1 + 3 + 5 + 7 + 9

E = 0 + 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9

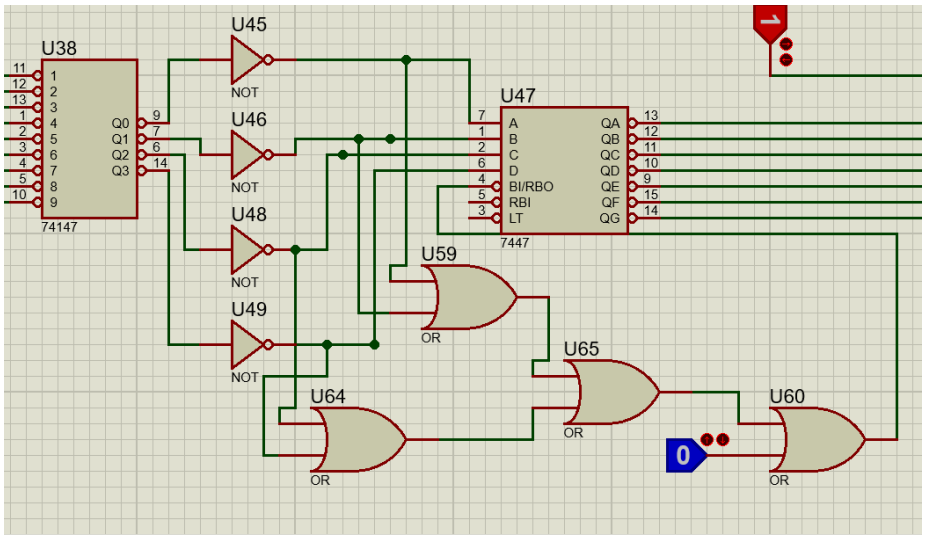
Let's start with numbers from 1 to 9. It's simple using a priority encoder as shown below:

	9	8	7	6	5	4	3	2	1	A	B	C	D
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	1	0	0	0	1
2	0	0	0	0	0	0	0	1	X	0	0	1	0
3	0	0	0	0	0	0	1	X	X	0	0	1	1
4	0	0	0	0	0	1	X	X	X	0	1	0	0
5	0	0	0	0	1	X	X	X	X	0	1	0	1
6	0	0	0	1	X	X	X	X	X	0	1	1	0
7	0	0	1	X	X	X	X	X	X	0	1	1	1
8	0	1	X	X	X	X	X	X	X	1	0	0	0
9	1	X	X	X	X	X	X	X	X	1	0	0	1

At this point, there's a problem; when no button is pressed, the 7-segment display shows the number 0. Using OR gates between the encoder outputs, the BCD decoder enable pin, and the ZERO button, the problem is solved.

Now, if no button is pressed, the output of the OR gates is ZERO, and the BCD decoder is OFF. When ZERO is pressed, the decoder is ON, and ZERO is displayed on the 7-segment display.

This is our circuit simulation on Proteus:



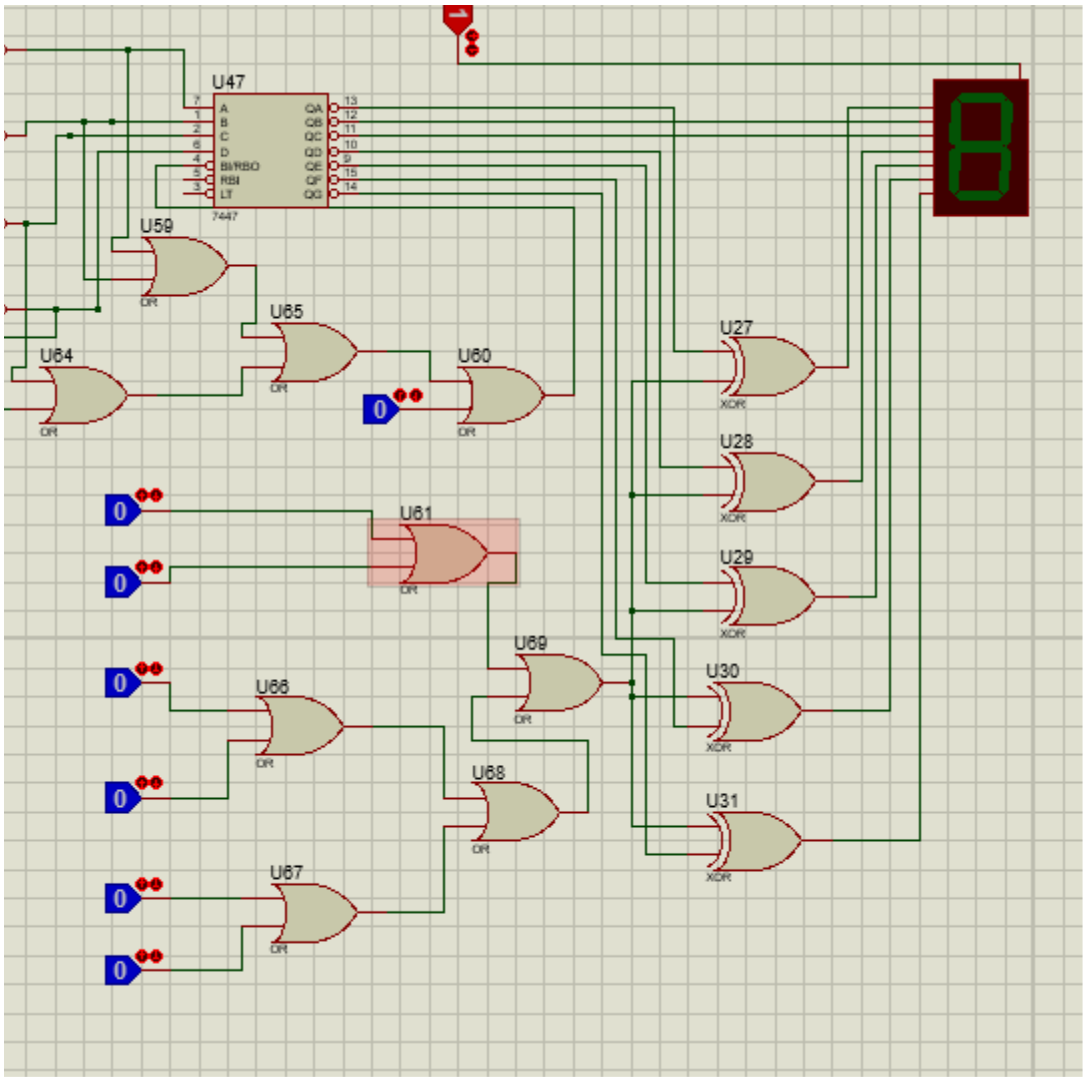
What about the 6 unused buttons?

We need to display the letter E (Error) on the 7-segment when pressed. But the 7-segment does not have the option to show letters, so we will connect directly to the 7-segment.

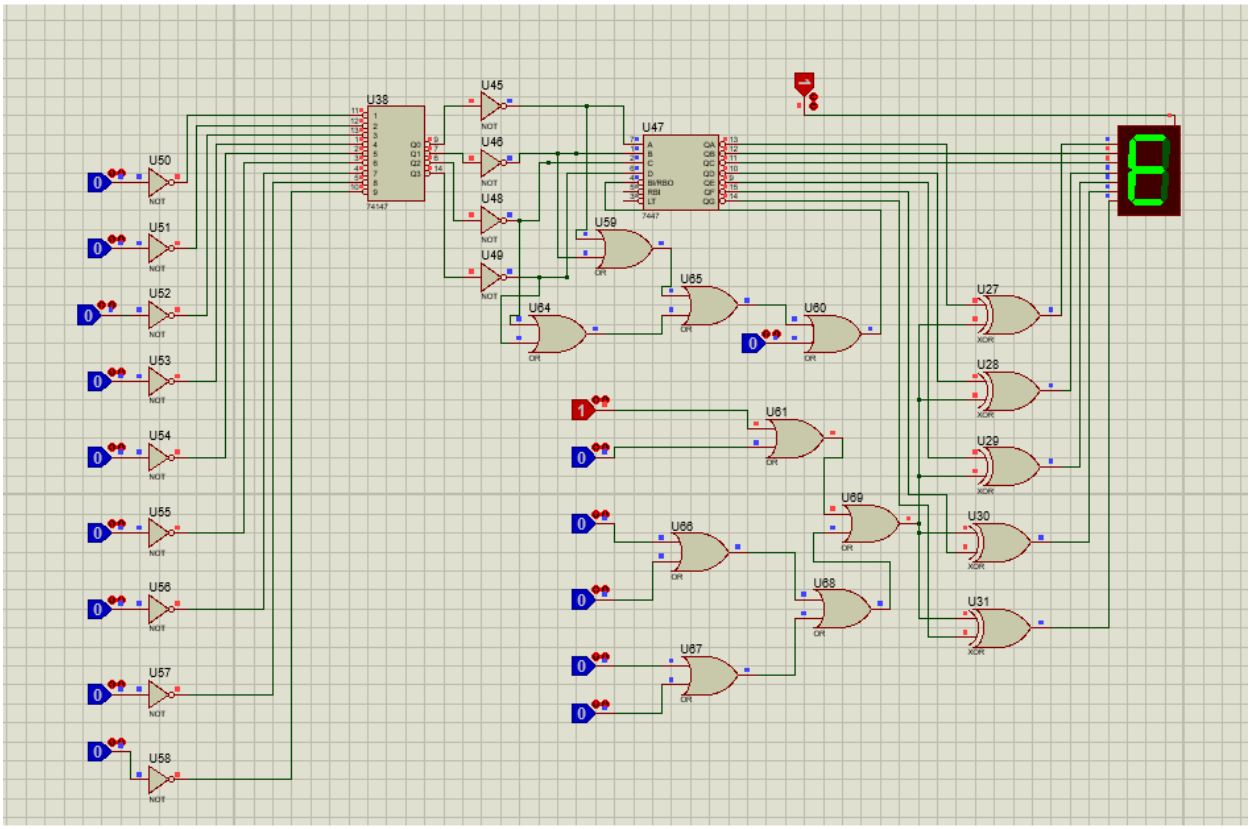
We need a f g e d segments to be ON when any of these buttons is pressed

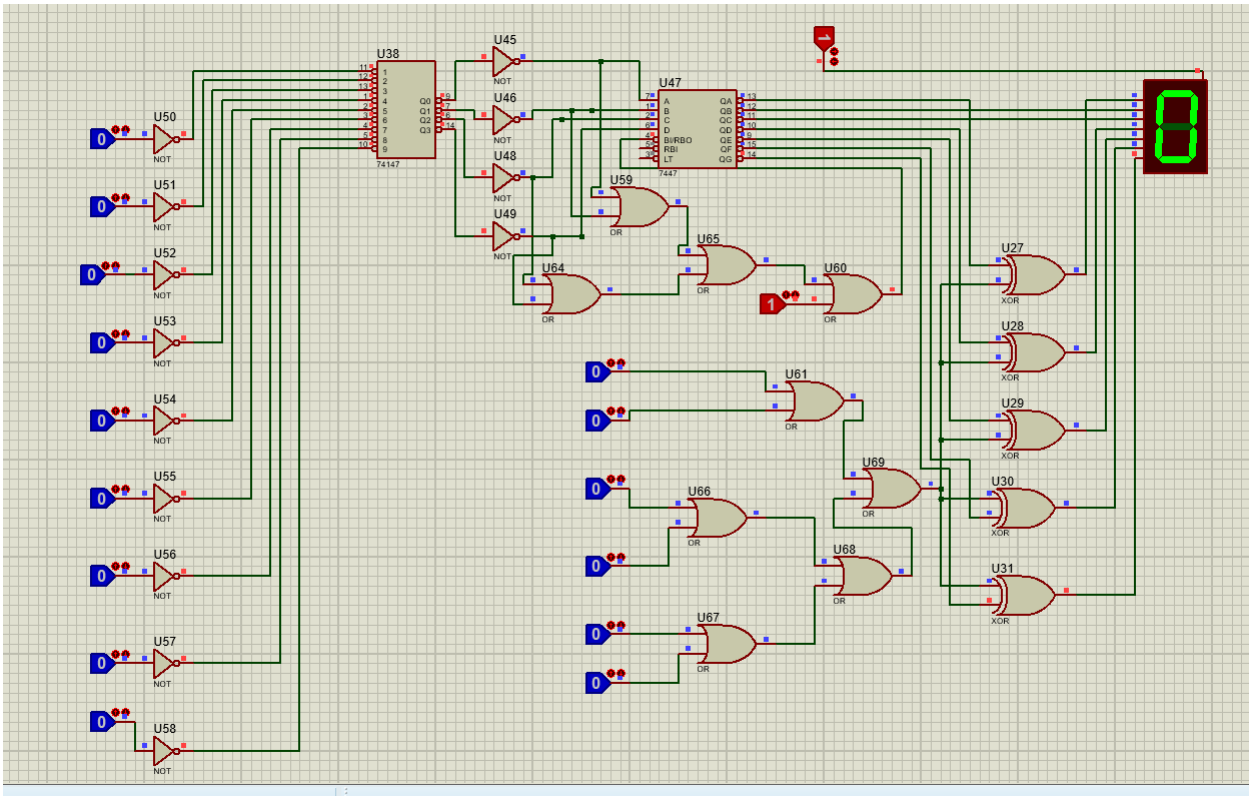
encoder	button	7 segment
0	0	0
0	1	1
1	0	1
1	1	X

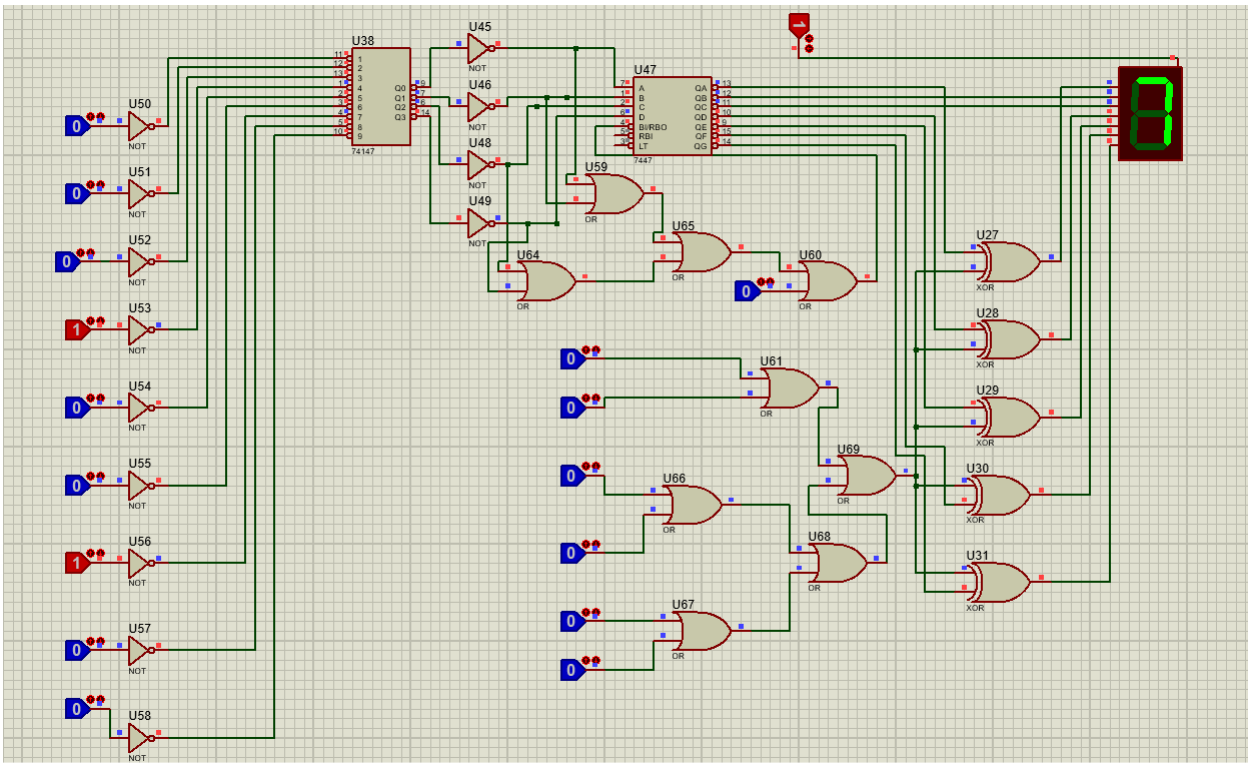
And this is our simulation on Proteus:

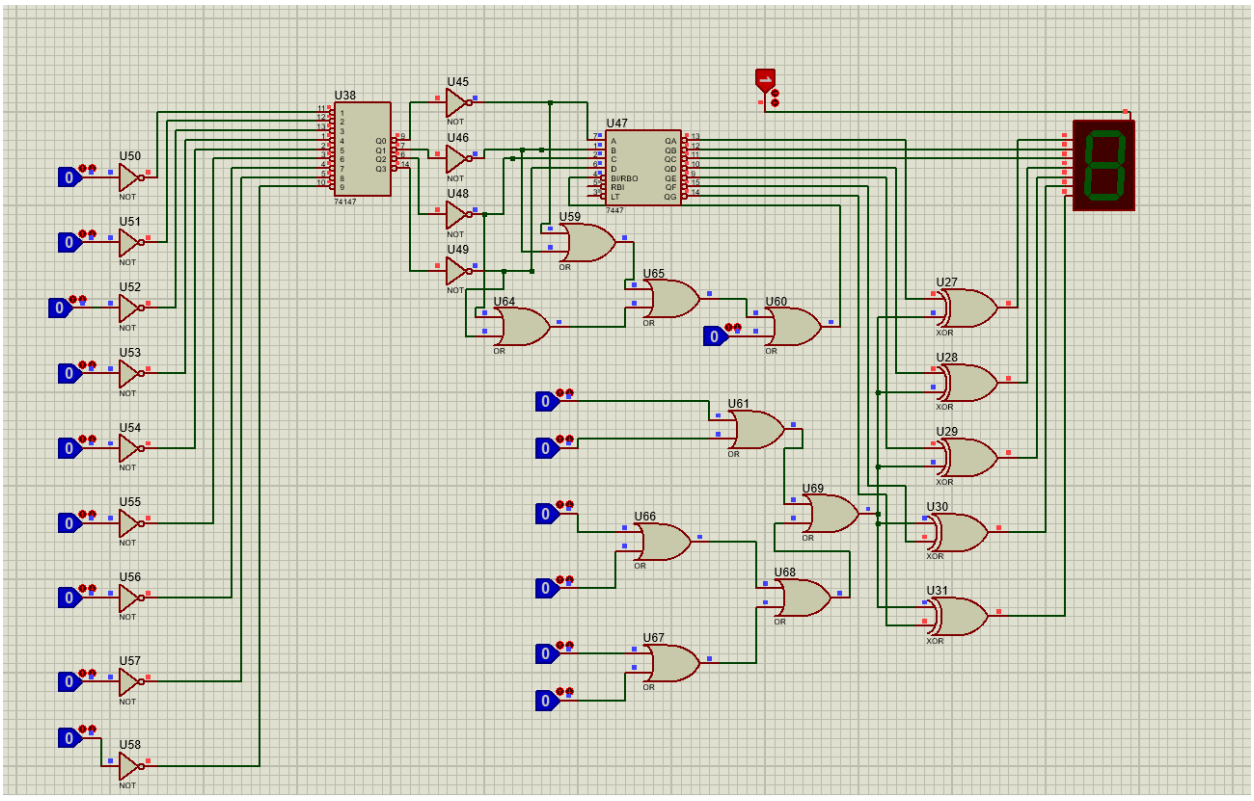

U47
7447
U59
U65
U64
U60
U61
U69
U66
U68
U67
U27
U28
U29
U30
U31
OR
XOR
0

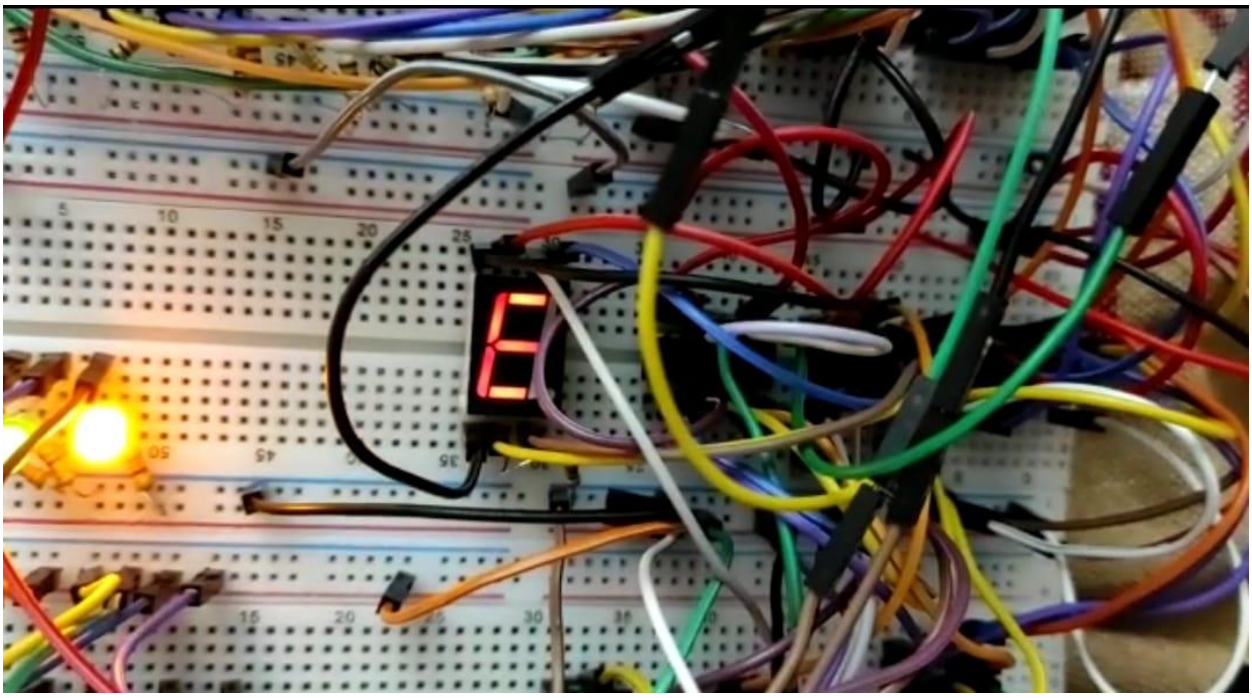
Some screenshots of the whole circuit simulation

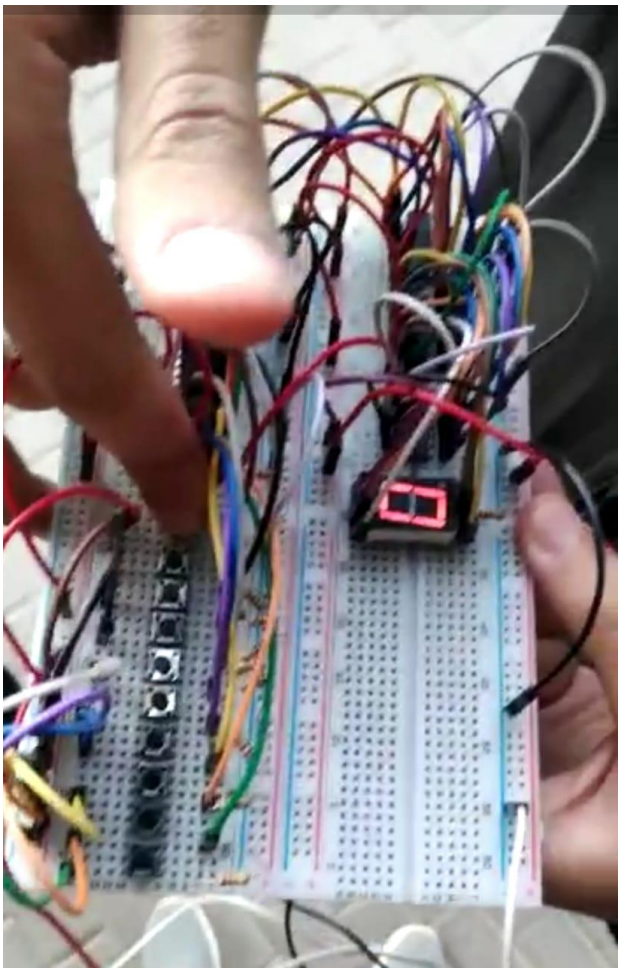


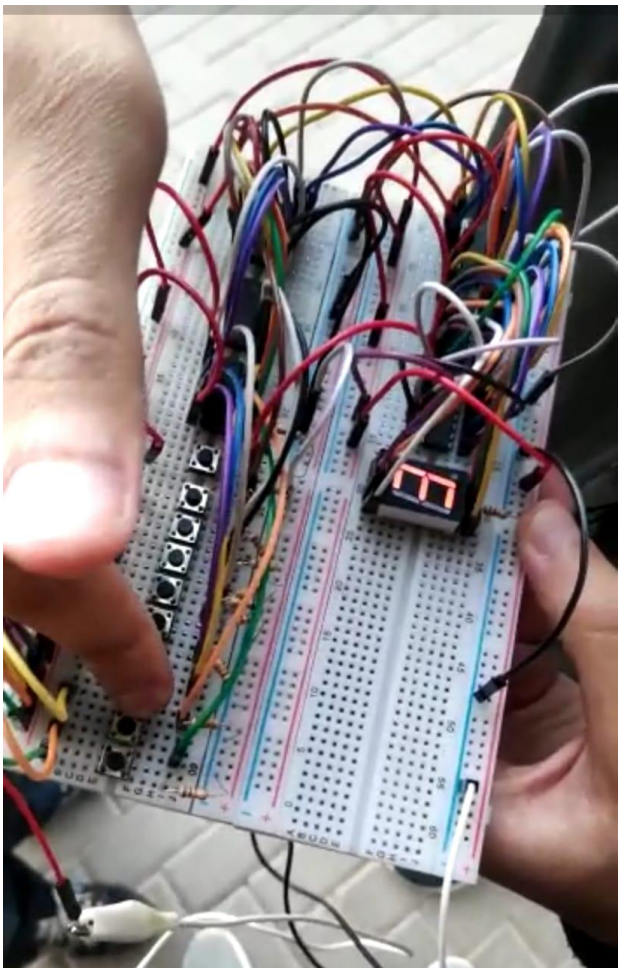



U38
74147
U45
U46
U48
U49
U47
7447
U59
U65
U64
U60
U50
U51
U52
U53
U54
U55
U56
U57
U58
U27
U28
U29
U30
U31
U61
U66
U67
U68
U69
NOT
OR
XOR


U38
74147
U45
U46
U48
U49
U47
7447
U59
U65
U64
U60
U50
U51
U52
U53
U54
U55
U56
U57
U58
U27
U28
U29
U30
U31
U61
U66
U67
U68
U69
NOT
OR
XOR







Phase 2

The purpose of the circuit shown in the figure is to display the sequence of numbers: 1 6 7 5 8 3 4 on the 7-segment display and repeat, with the aid of a sequential circuit, and 4 D-flipflops.

Starting with the state table, K-maps, and functions of Ds of the flipflops

	Present			Next											
A	B	C	D	A8	B4	C2	D1		1	6	7	5	8	3	4
0	0	0	0	0	0	0	1								
0	0	0	1	0	1	1	0								
0	0	1	0	0	0	0	1								
0	0	1	1	0	1	0	0								
0	1	0	0	0	0	0	1								
0	1	0	1	1	0	0	0								
0	1	1	0	0	1	1	1								
0	1	1	1	0	1	0	1								
1	0	0	0	0	0	1	1								
1	0	0	1	0	0	0	1								
1	0	1	0	0	0	0	1								
1	0	1	1	0	0	0	1								
1	1	0	0	0	0	0	1								
1	1	0	1	0	0	0	1								
1	1	1	0	0	0	0	1								
1	1	1	1	0	0	0	1								

Da			C	
	1			B
A				
		D		

Da = A'BC'D

Db			C	
		1	1	
			1	1	B
A				
		D		

Db = A'B'D + A'BC

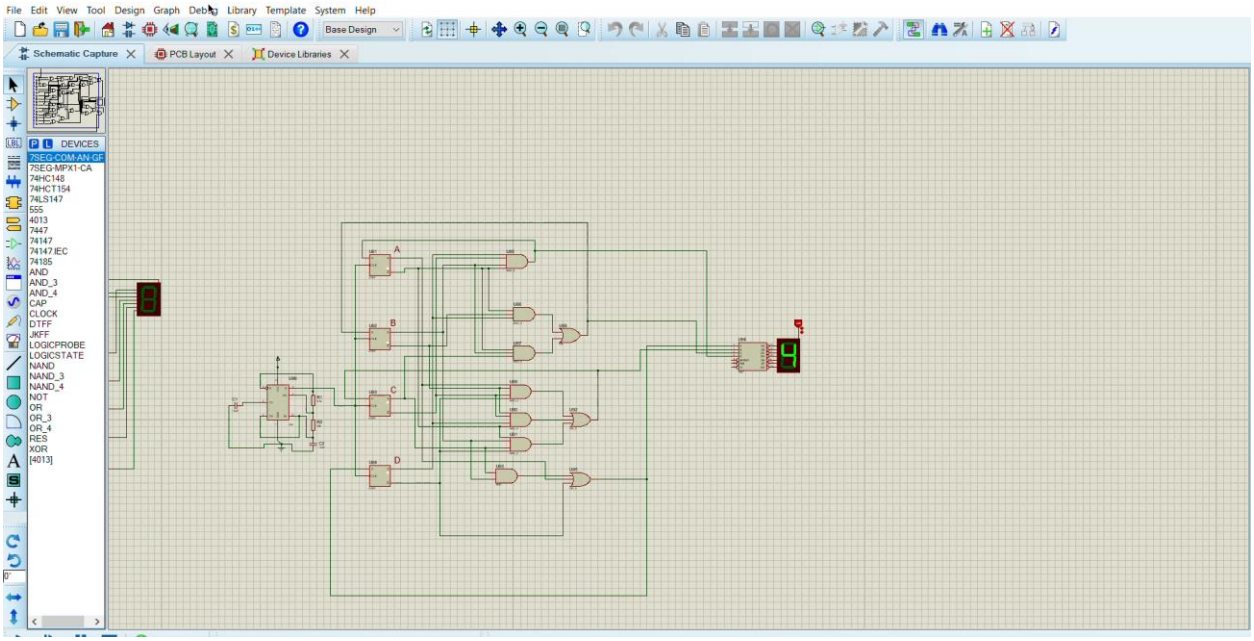
Dc			C	
	1			
			1	B
A				
1				
		D		

Dc = AB'C'D' + A'B'C'D + A'BCD'

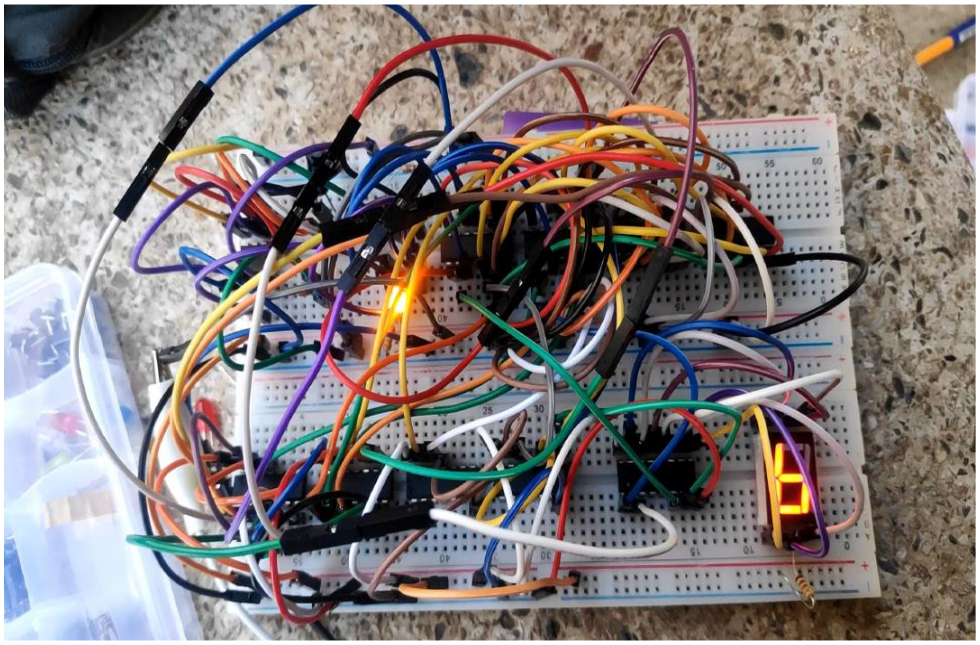
Dd			C	
1			1	
1		1	1	B
A	1	1	1	
1	1	1	1	
		D		

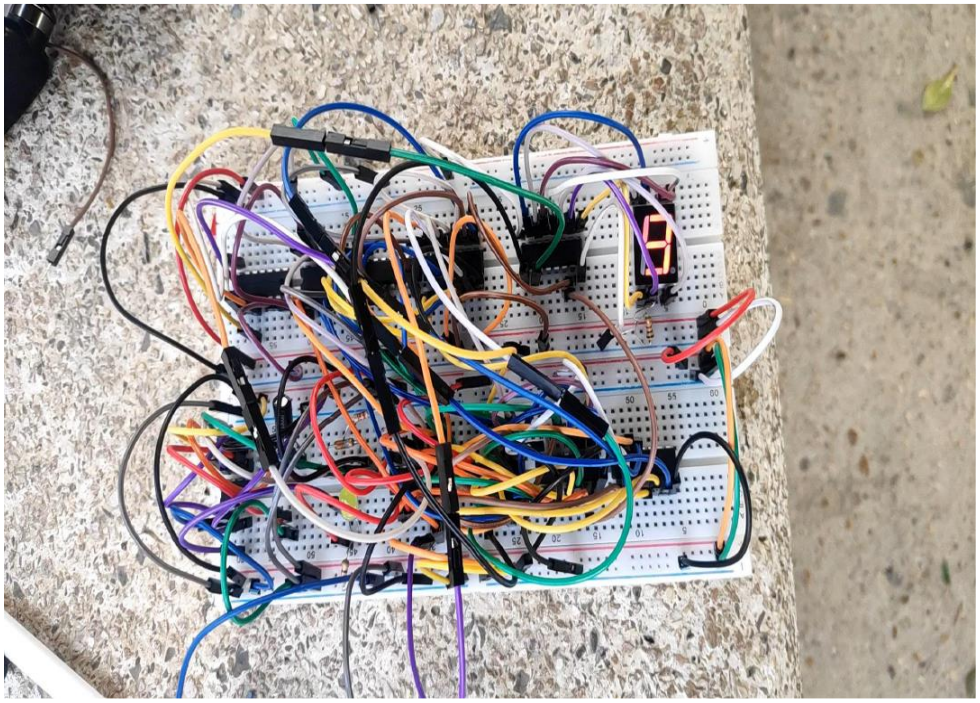
Dd = A + D' + BC

Implementing on Proteus



Screenshots of the hardware implementation





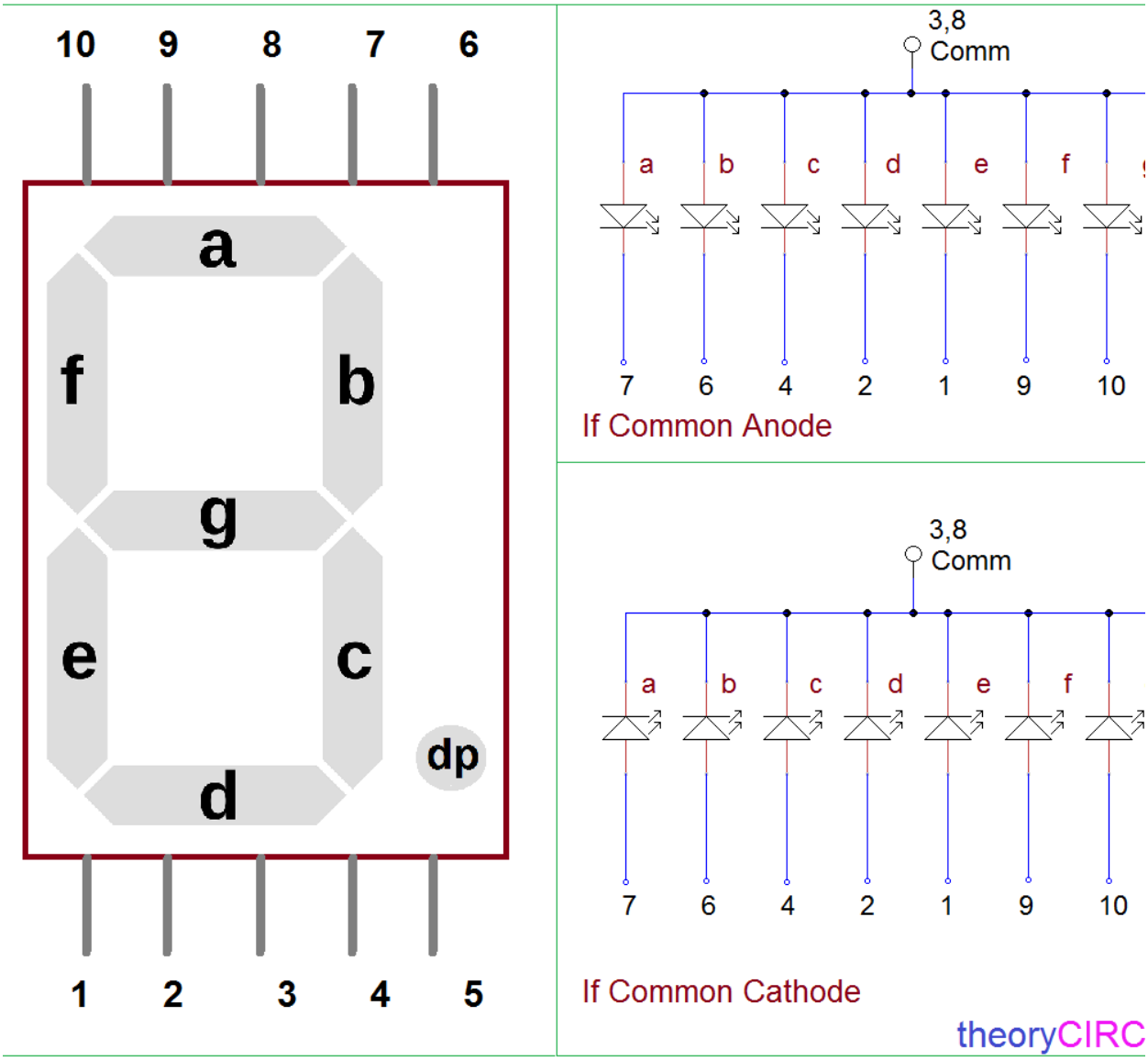
Conclusion

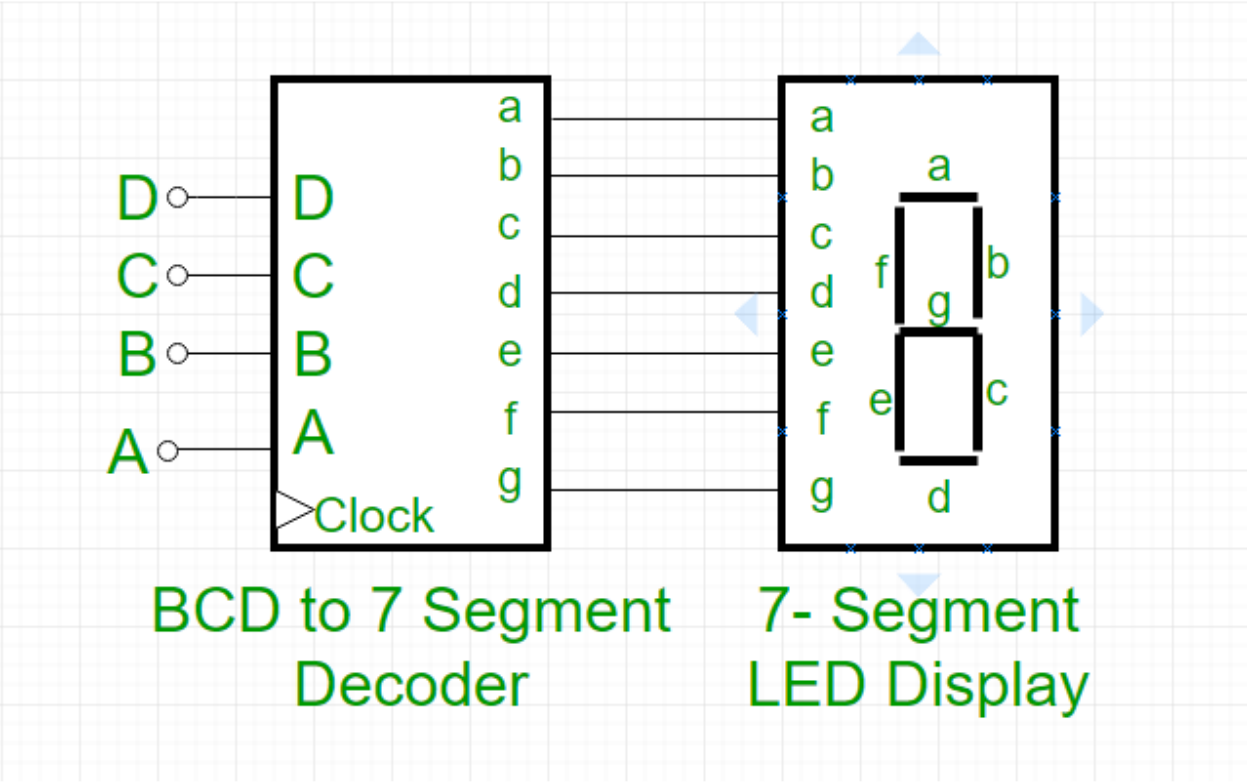
Implementing hardware circuits is much harder than the software simulation. So the circuit must be simplified as much as possible, and the components used must be in a good condition as well, to avoid any kind of unexpected errors.

Thank You!

Pin Diagrams of some hardware components used

7 Segment Display Pinout





Truth Table of BCD to 7-segment decoder

A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1

The rest of cases are taken as Don’t Care