

## **Lab 8**

### **To Practice Binary to Gray Code & BCD to Excess-3 Conversion**

*Note: You may draw all the logic diagrams with hand and paste the pictures here. Also, the conversions in the tasks can be done by hand if doing it in soft seems difficult to you.*

*Use Logicly software with your name, roll number & section mentioned in your workspace. Make sure that all of your connections are clearly visible and distinguishable. In logicly, use **“text”** label to point out/show all your inputs & outputs.*

#### **Task 1**

- a) Convert the given binary codes into respective gray codes. Mention each step while converting the codes.

1111, 1101, 1001, 010, 011, 110

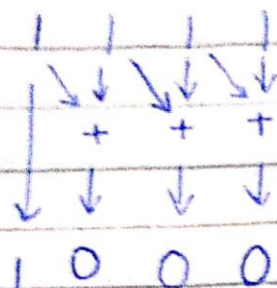
Saad Ahmad

(X) DLD-Lab

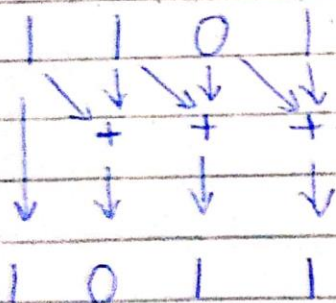
Q1 Binary codes to gray code:

①  $(1111)_2$

MSB

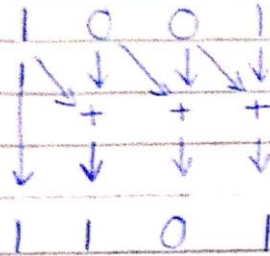


②  $(1101)_2$

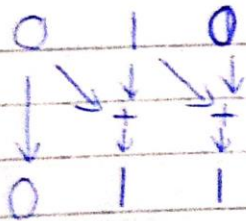


Saad Ahmad (20P-0051)

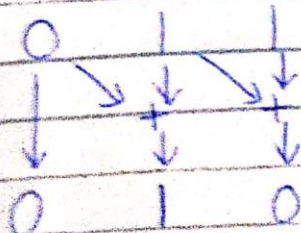
②  $(1001)_2$



④  $(010)_2$

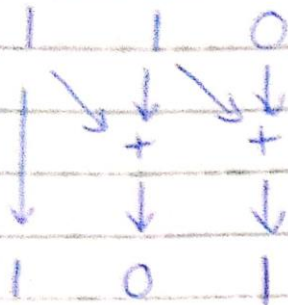


⑤  $(011)_2$



Saad Ahmad (20P-0051)

⑥  $(110)_2$



b) Convert the given gray codes into respective binary codes. Mention each step while converting the codes.

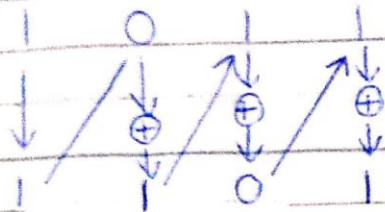
1011, 0101, 1100, 0100



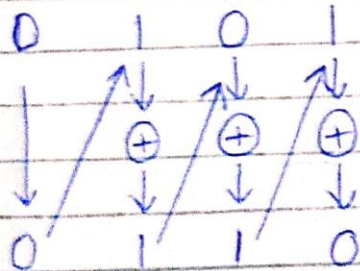
Saad Ahmad (20P-0051)

Q2, Gray codes to binary codes:

①  $(1011)_4$

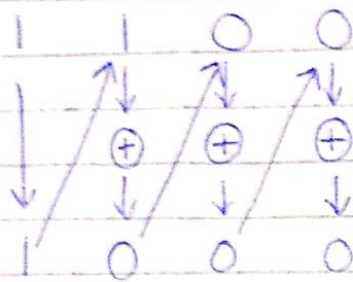


②  $(0101)_4$

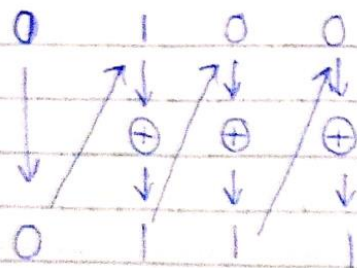


Saad Ahmad (20P-0051)

③  $(1100)_4$



④  $(0100)_4$



## **Task 2**

**Devise a truth table for 3-bit Binary to Gray code converter. Write simplified logical expressions and simulate the logic diagram in Logicly software to verify your results. Show KMap simplification as well.**

a) Truth Table

A	B	C	G2	G1	G0
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0

b) Boolean Expression (Simplified using KMap)

$$G2 = AB'C' + AB'C + ABC' + ABC$$

$$G1 = A'BC' + A'BC + AB'C' + AB'C$$

$$G0 = A'B'C + A'BC' + AB'C + ABC'$$

KMap for G2

		BC			
A					
		1	1	1	1

$$G2 = A$$



KMap for G1

A	BC			
		1		1
		1		1

$$G1 = AB' + A'B$$

KMap for G0

A	BC			
			1	1
	1	1		

$$G0 = B'C + BC'$$

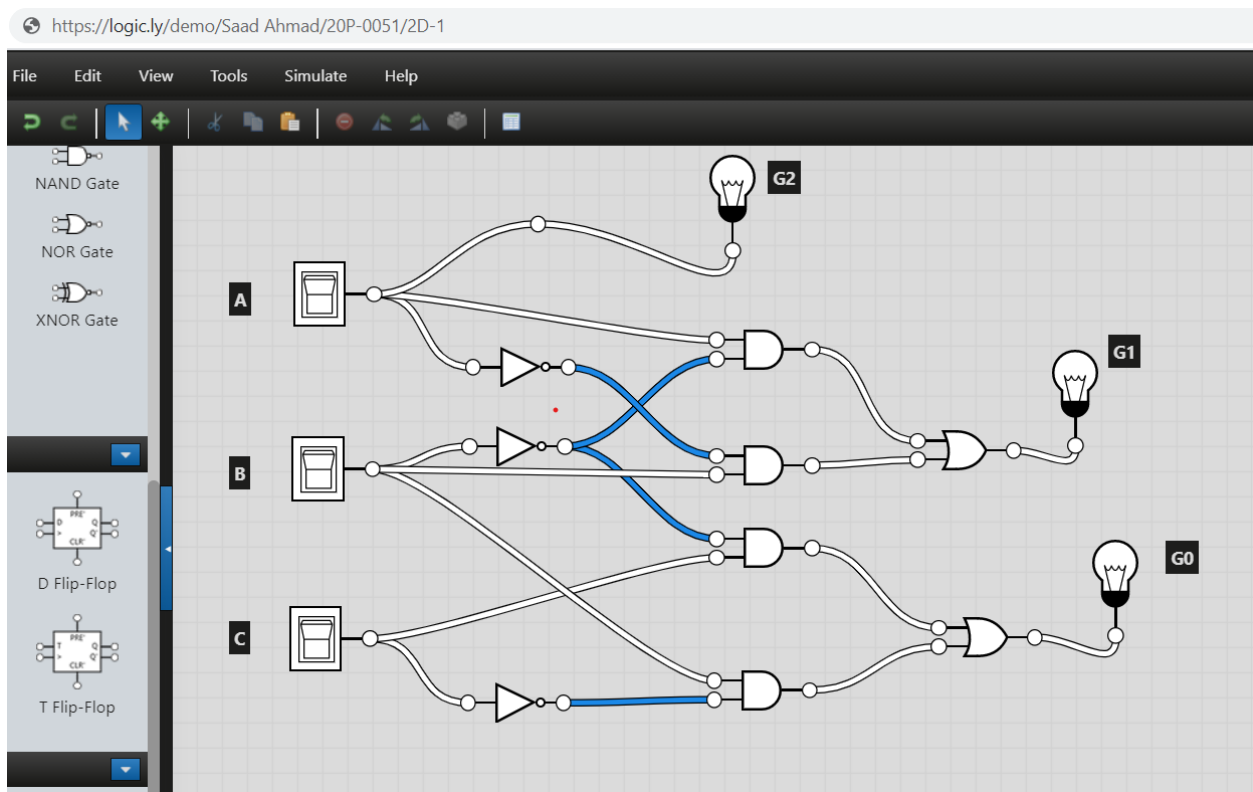
Simplified expression for G2, G1 and G0 is:

$$G2 = A$$

$$G1 = AB' + A'B$$

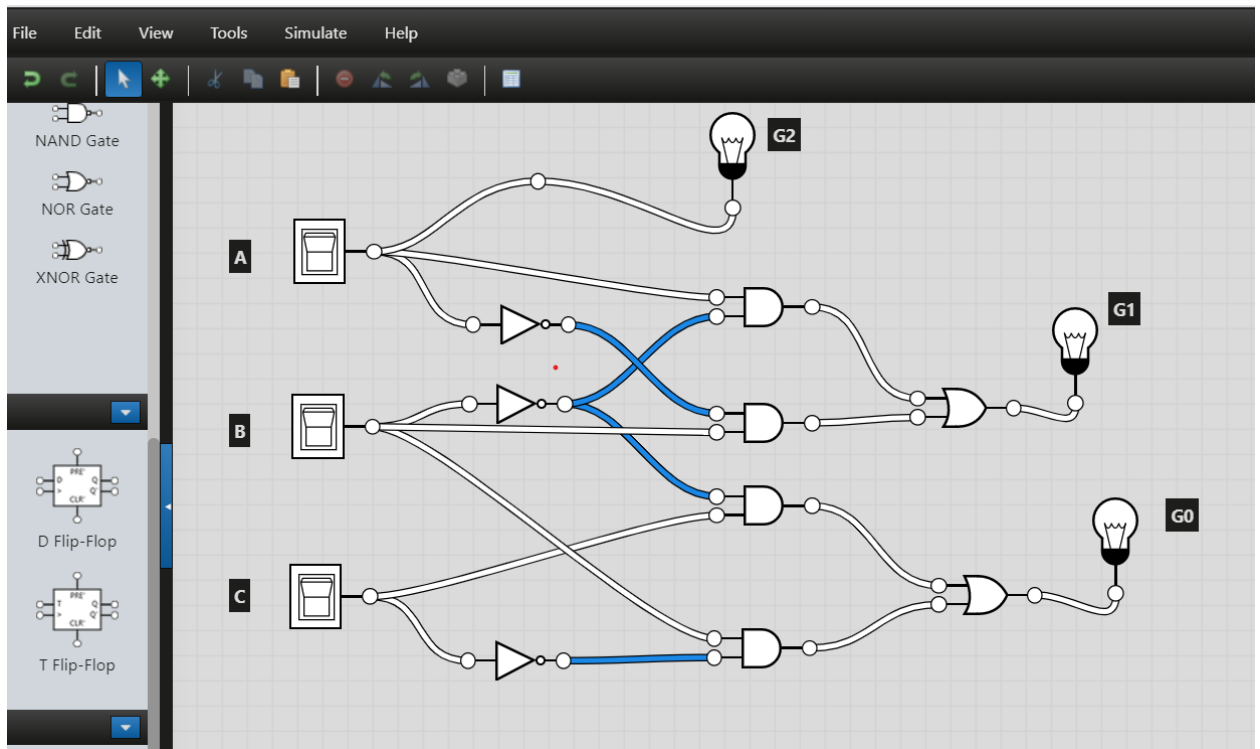
$$G0 = B'C + BC'$$

### c) Logic Diagram

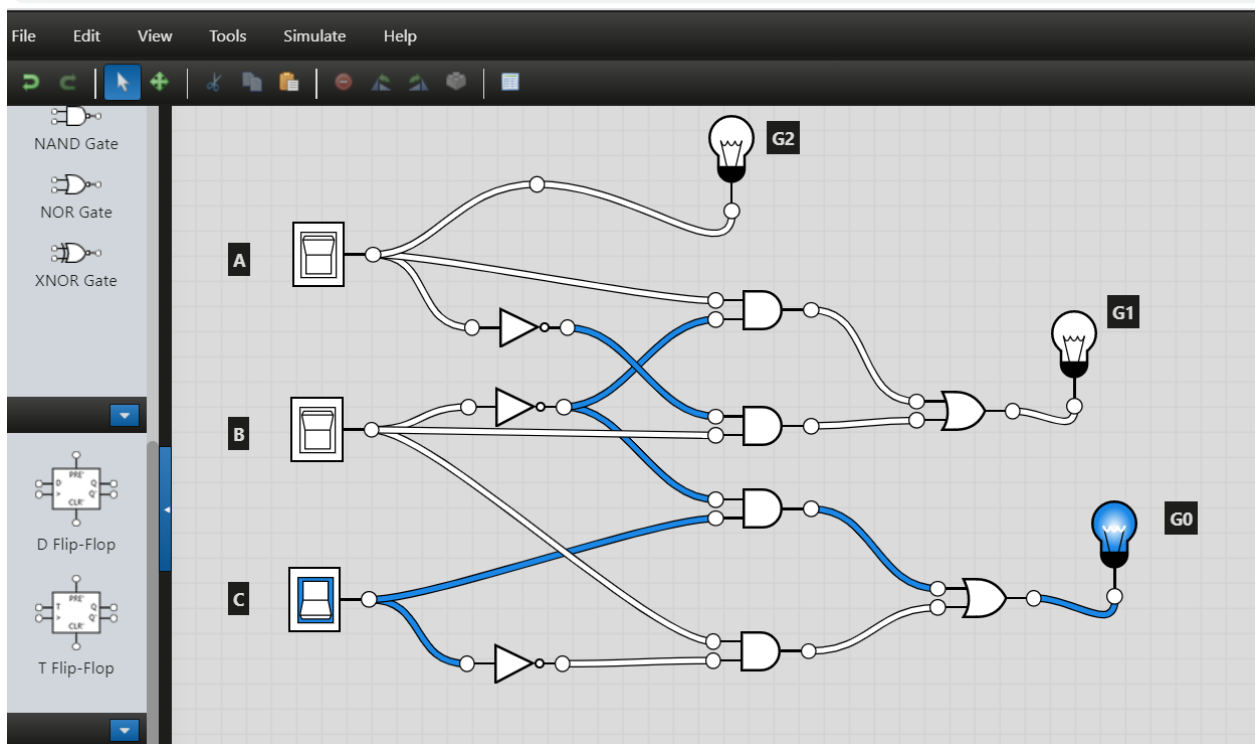


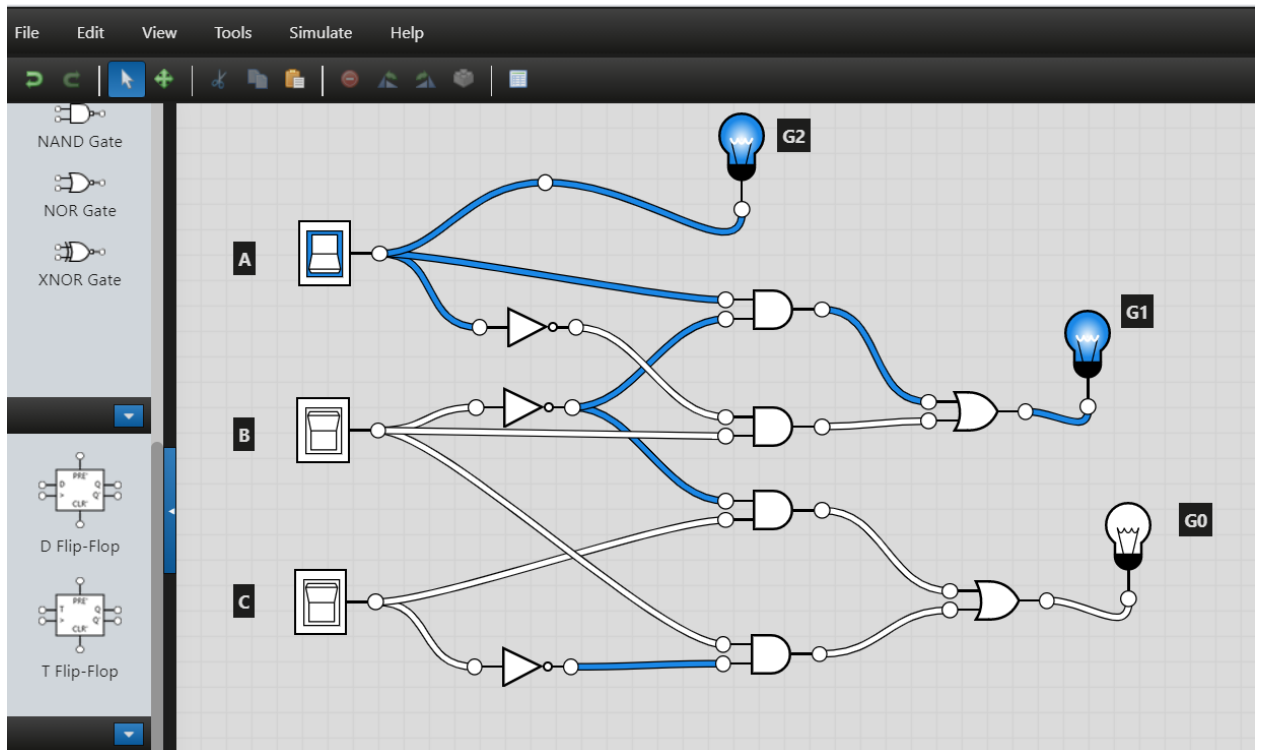
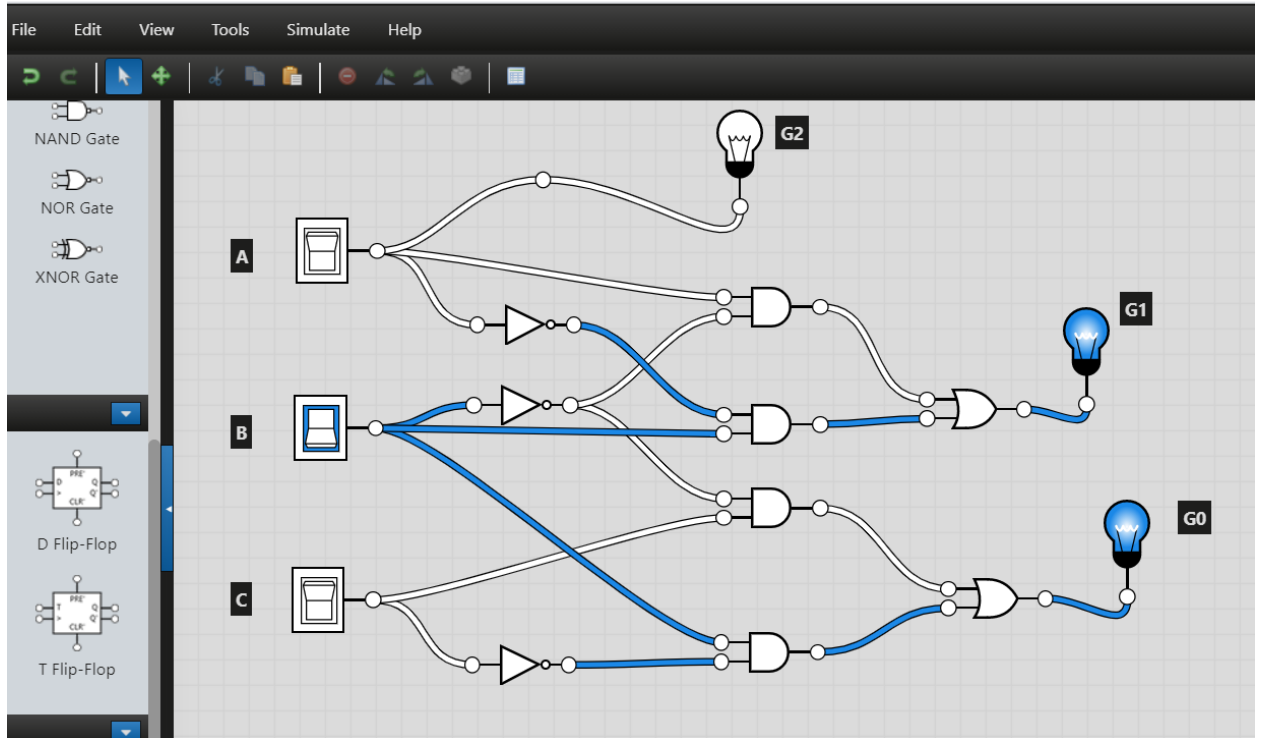
a) Software Simulation (Show here your results for a few combinations to verify the circuit)

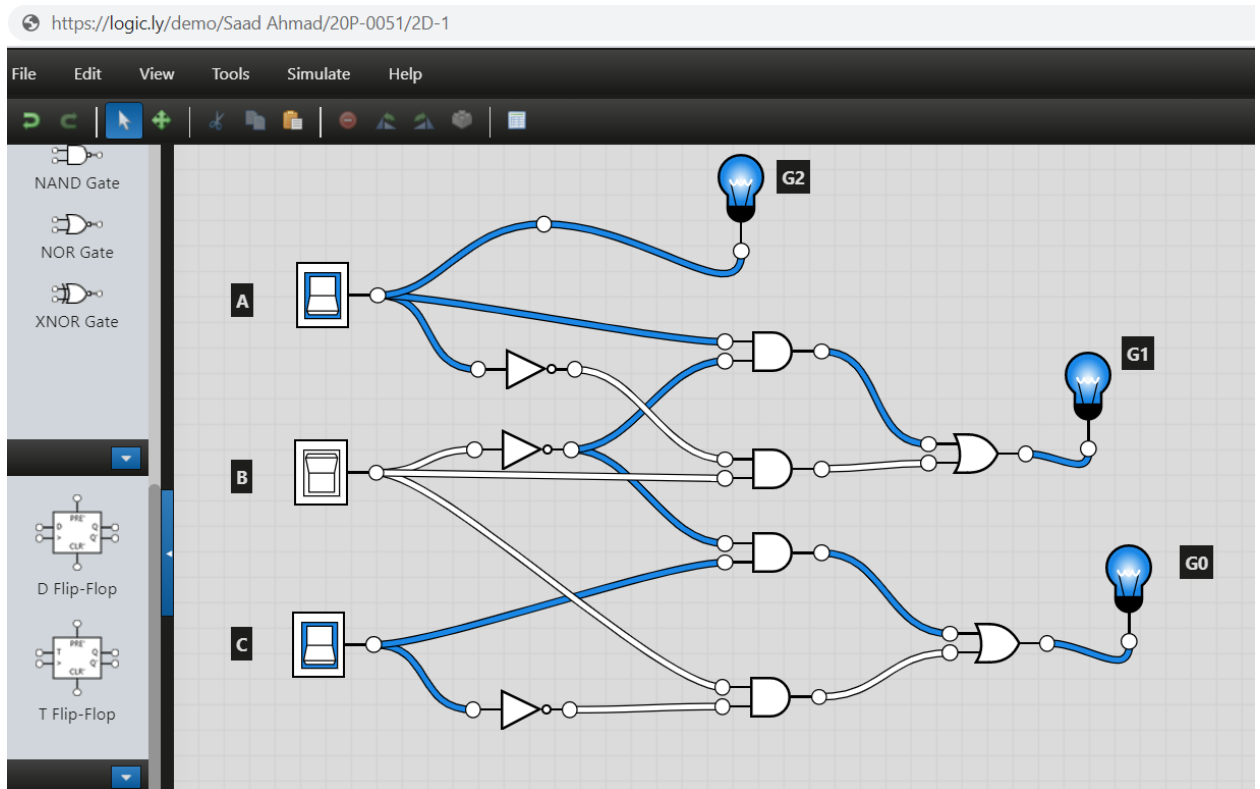
[https://logic.ly/demo/Saad Ahmad/20P-0051/2D-1](https://logic.ly/demo/Saad%20Ahmad/20P-0051/2D-1)



[https://logic.ly/demo/Saad Ahmad/20P-0051/2D-1](https://logic.ly/demo/Saad%20Ahmad/20P-0051/2D-1)







### **Task 3**

a) For the given numbers, find excess-3 codes. Mention all the steps of conversion.

4, 9, 25, 50, 250, 405.

Saad Ahmad (20P-0051)

Q3(a) Excess-3 code:

① 4:

$$4 = (0100)_{BCD}$$

0 1 0 0

0 0 1 1

0 1 1 1

$$4 = (0111)_{E-3}$$

② 9:

$$9 = (1001)_{BCD}$$

0 0  
1 0 0 1

0 0 1 1

1 1 0 0

$$9 = (1100)_{E-3}$$

③ 25:

$$25 = (0010 \ 0101)_{BCD}$$

0 0 1 0

0 0 1 1

0 1 0 1

0 0 1 0 1

0 0 1 1

1 0 0 0

$$25 = (0101 \ 1000)_{E-3}$$



Saad Ahmad (20P-0051)

④ 50:

$$50 = (0101 \ 0000)_{BCD}$$

$$\begin{array}{r} \textcircled{0}\textcircled{0}\textcircled{0} \\ 0101 \ 0000 \\ 0011 \ 0011 \\ \hline 1000 \ 0011 \end{array}$$

$$50 = (1000 \ 0011)_{E-3.}$$

⑤ 250:

$$250 = (0010 \ 0101 \ 0000)_{BCD}$$

$$\begin{array}{r} \textcircled{0} \quad \textcircled{0}\textcircled{0}\textcircled{0} \\ 0010 \ 0101 \ 0000 \\ 0011 \ 0011 \ 0011 \\ \hline 0101 \ 1000 \ 0011 \end{array}$$

$$(250) = (0101 \ 1000 \ 0011)_{E-3}$$

Saad Ahmad (20P-0051).

① 405

$$405 = (0100 \quad 0000 \quad 0101)_{BCD}$$

		000
0100	0000	0101
0011	0011	0011
<u>0111</u>	0011	1000

$$405 = (0111\ 0011\ 1000)_{E-3}$$

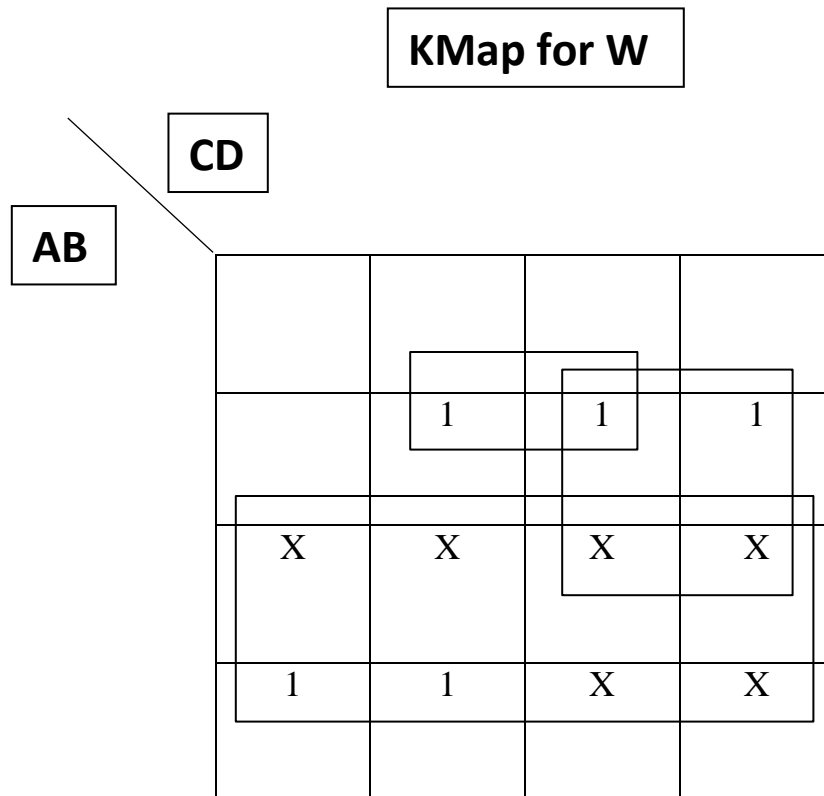
- b)** Devise a truth table for BCD to Excess-3 conversion (as discussed in the lecture). Write simplified expressions and simulate the logic circuit in Logically software to verify your truth table. You may paste screenshots of a few combinations from the truth table to check your circuit. Show your KMap simplification as well.

### BCD to Excess-3 Converter

- a) Truth Table

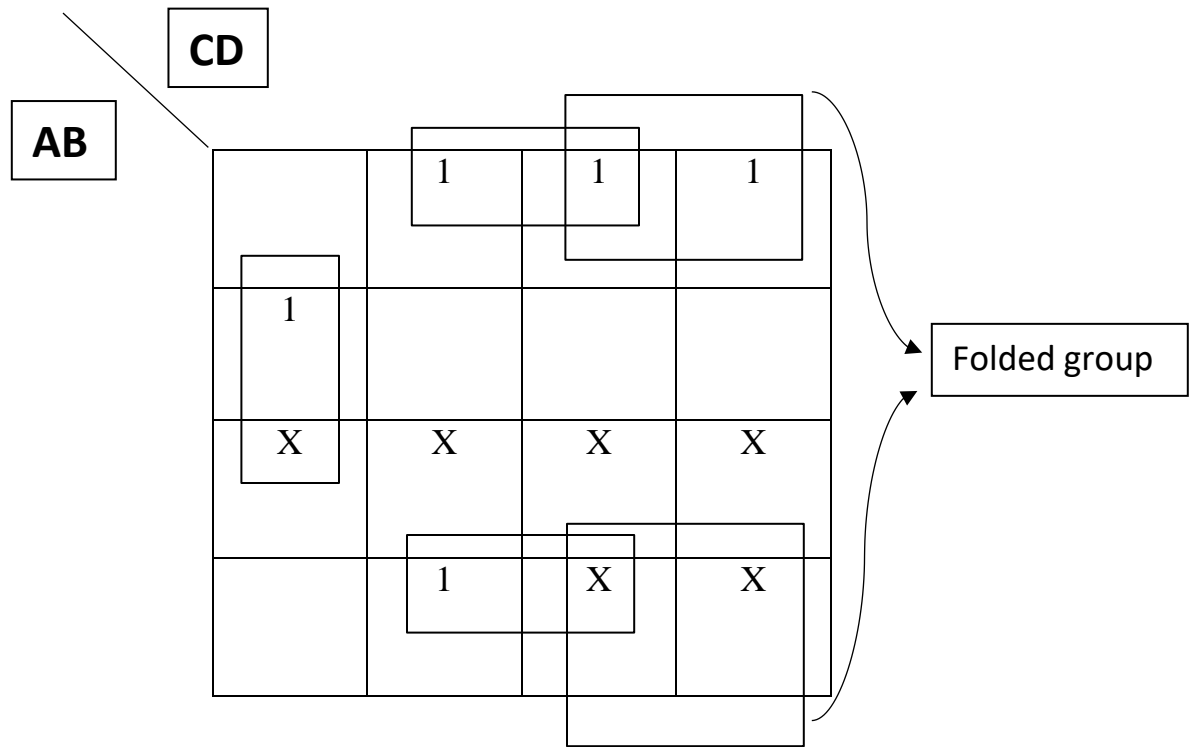
BCD (8421)				Excess- 3			
A	B	C	D	W	X	Y	Z
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0
1	0	1	0	X	X	X	X
1	0	1	1	X	X	X	X
1	1	0	0	X	X	X	X
1	1	0	1	X	X	X	X
1	1	1	0	X	X	X	X
1	1	1	1	X	X	X	X

b) Boolean Expression (Simplified Using KMap)



$$W = A + BC + BD$$

# KMap for X



$$X = B'C + B'D + BC'D'$$

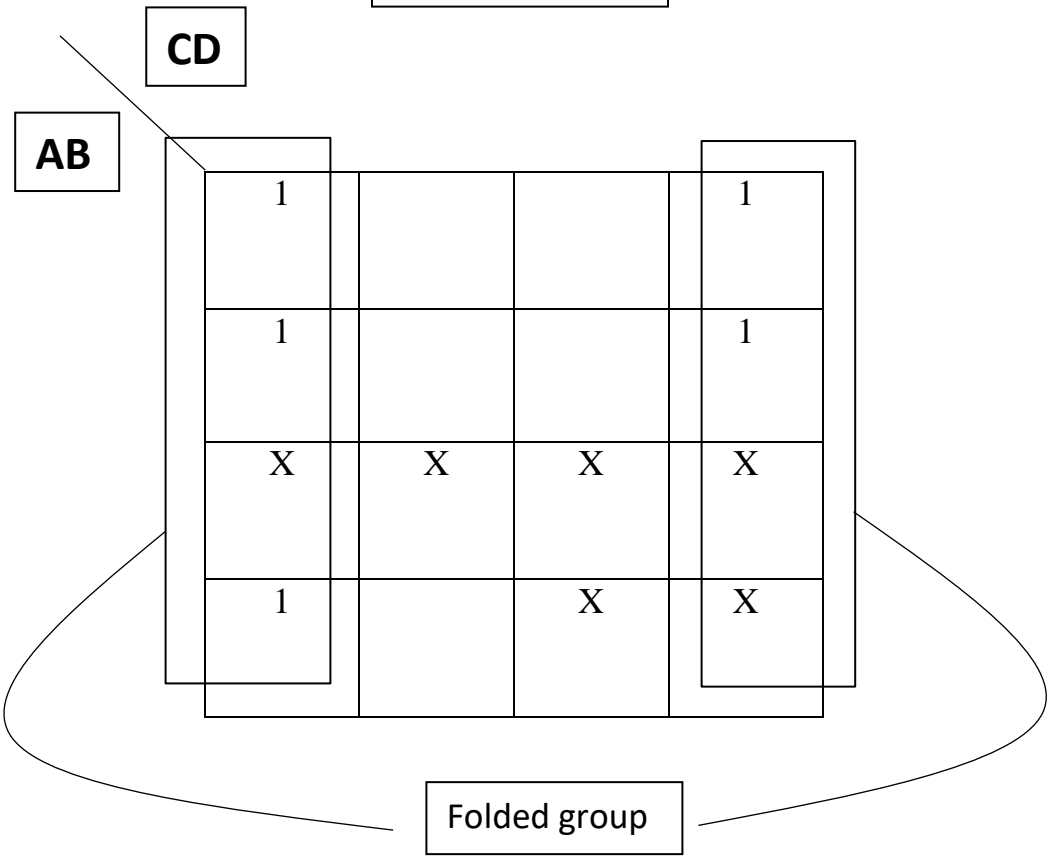
# KMap for Y

AB	CD			
	1		1	
	1		1	
	X	X	X	X
	1		X	X

$$Y = CD + C'D'$$

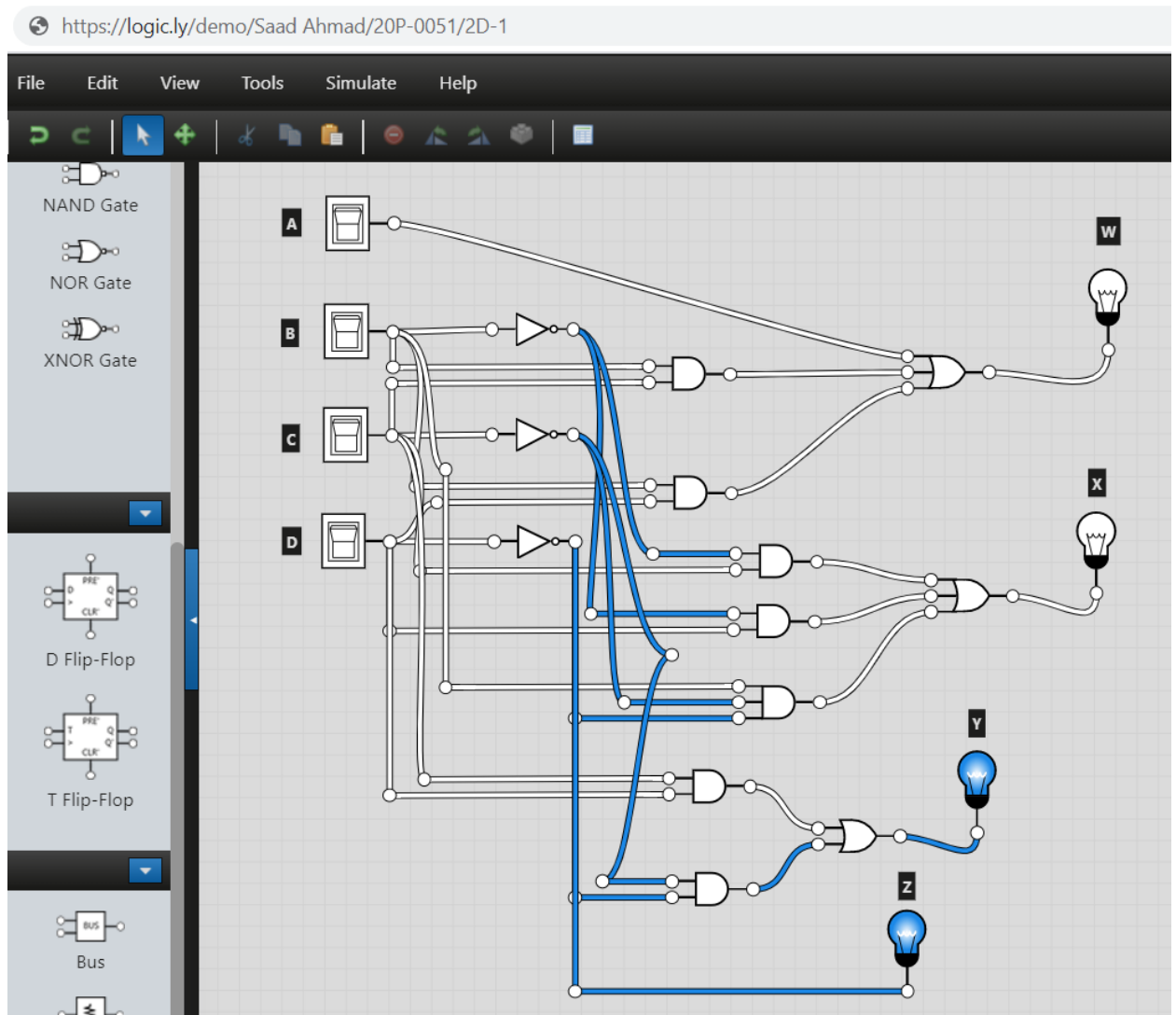


KMap for Z



$Z = D'$

c) Logic Diagram



d) Software Simulation (Show here your results for a few combinations to verify the circuit)

