

Challenge Problem 1

AVVARU BHARAT

Download all latex-tikz codes from

<https://github.com/Bharat437/EE5803-FPGA-LAB/tree/main/Challenge-1>

1 Problem

Obtain and implement an algorithm to convert any truth table to NAND logic.

2 Algorithm

1. Obtain K-Map for given truth table.
2. From K-Map obtain NAND Logic instead of SOP form.

3 Explanation

For example the given truth table is as below

X	Y	Z	G(X,Y,Z)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Table 1: Given Truth table

Now the K-Map for given truth table 1 is as below.

		YZ			
		00	01	11	10
X	0	0	0	0	1
	1	1	1	1	0

Figure 1: K-Map for given truth table

Now the NAND logic from K-Map in figure 1 is as follows.

$$G = \overline{(X.\bar{Y})} . \overline{(X.Y.Z)} . \overline{(\bar{X}.Y.\bar{Z})} \quad (3.0.1)$$

Now we will draw logic circuit according to the above expression.

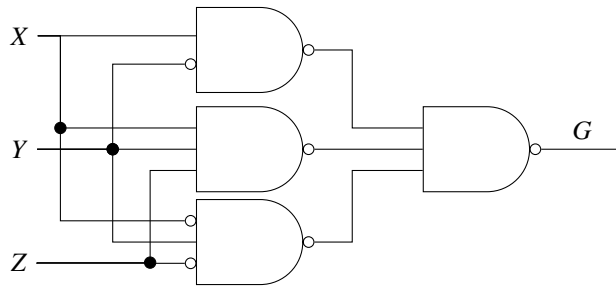


Figure 2: Logic circuit using NAND gate for given truth table

Whereas SOP form from K-Map in figure 1 is given as below.

$$G = X.\bar{Y} + X.Y.Z + \bar{X}.Y.\bar{Z} \quad (3.0.2)$$

Now the logic circuit for above SOP form expression is as given below.

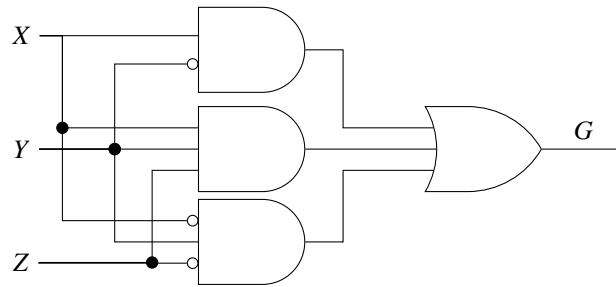


Figure 3: Logic circuit from SOP form

4 Conclusion

From the above circuit figures 2 and 3, we can say that SOP form i.e. AND-OR logic is equivalent to NAND-NAND logic. The Verification is also done using a c code.