Practical exercise 6

Aim :- To minimize and realize functions using K-maps and its implementation by constructing the circuit on bread-board.

Theory :-

Karnaugh map :-

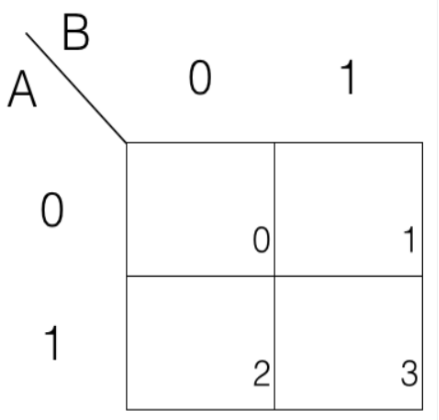
A Karnaugh Map, also known as a K-map, is a graphical representation used in digital logic design and simplification of Boolean algebra expressions. It is named after the American mathematician and engineer, Maurice Karnaugh, who developed this tool in the 1950s.

Karnaugh Maps are primarily used for simplifying and optimizing Boolean functions, which are expressions composed of logical AND, OR, and NOT operations. These maps are particularly useful for reducing the number of gates and variables in digital circuits, which can lead to more efficient and cost-effective designs.

A Boolean expression consisting n number of variables so the number of cell required in K-map is 2^n .

Two variable K map :-

- Two variable K map is drawn for a Boolean expression consisting of two variables
- The number of cells present in two variable
- K map = 2^2 = 4 cells.
- So for a Boolean function consisting of two variables we draw 2 x 2 K map.



- Here A and B are two variables of given function

Three variable K map :-

- Three variable K map is drawn for a Boolean expression consisting of three variable.
- The number of cells present in three variable K map = 2^3 = 8cells.
- So for a Boolean function consisting of three variable we draw a 2 x 4 K map.

X \ YZ	00	01	11	10
0	0	1	3	2
1	4	5	7	6

- Here X , Y and Z are three variable of given Boolean function.

Four variable K map :-

- Four variable K map is drawn for a Boolean expression consisting of three variable.
- The number of cells present in four variable K map = 24 = 8cells.
- So for a Boolean function consisting of three variable we draw a 4 x 4 K map.

BC \ DE	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

- Here B , C , D and E are three variable of given Boolean function.

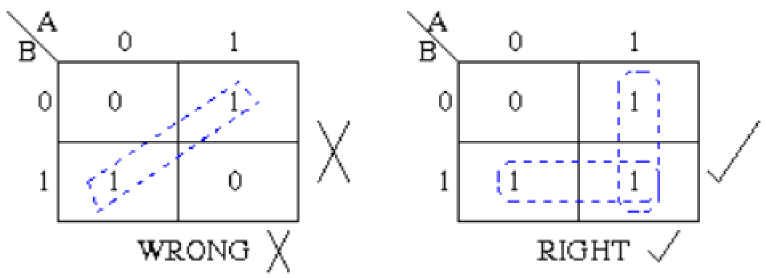
Karnaugh map simplification rules:-

- We draw a K map according to the number of variable contains.
- We fill the K map with 0's and 1's according to its function.
- Then, we minimize the function in accordance with the following rules.

Rule 1:

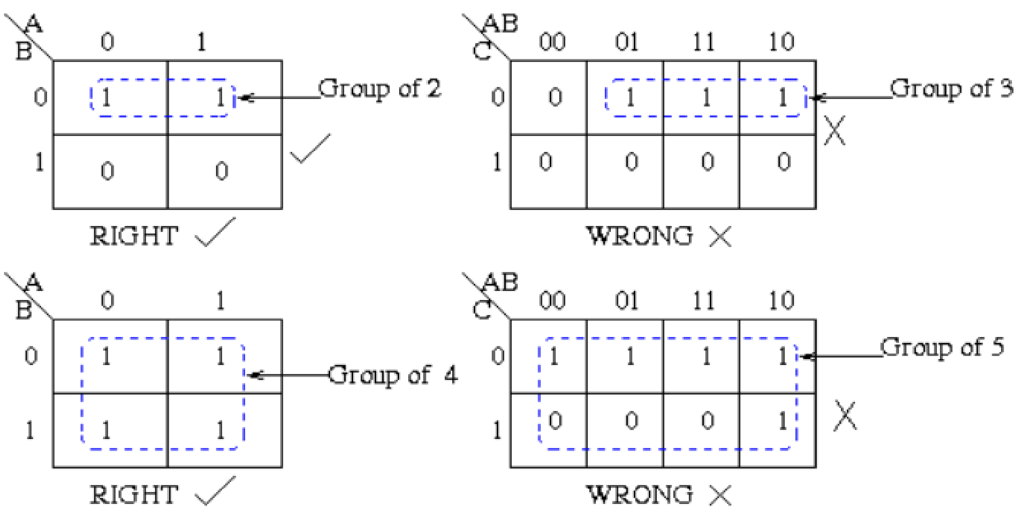
Grouping Ones: The primary goal is to identify and group adjacent ones (1s) in the K-map. Groups can be formed in the following ways:

- Horizontal Grouping: You can group ones horizontally in rows.
- Vertical Grouping: You can group ones vertically in columns.
- Wraparound Grouping: Groups can wrap around the edges of the K-map if needed.



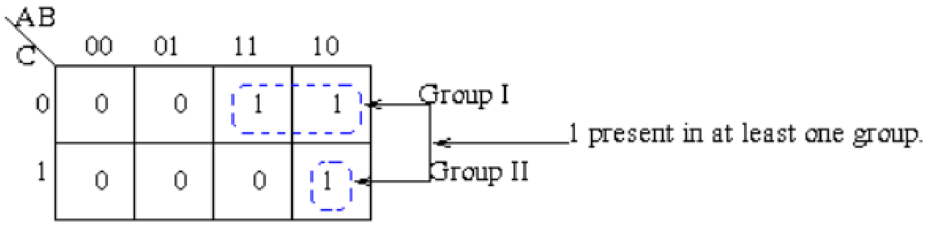
Rule 2:

Largest Groups Possible: Try to form the largest groups possible, as larger groups result in more simplified expressions. Groups should contain 1, 2, 4, 8, etc., ones (powers of 2)



Rule 3:

Cover All Ones: Make sure that every one (1) in the K-map is included in at least one group. Don't leave any ones out.



Rule 4:

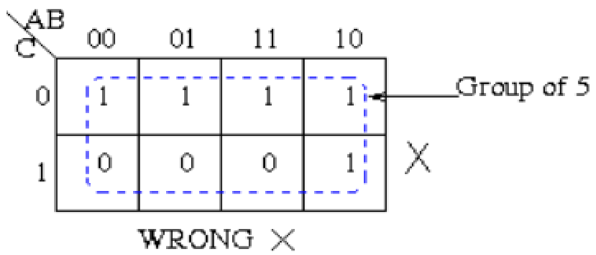
Overlap of Groups: It's acceptable for groups to overlap if it helps simplify the expression further. This can occur when a cell is part of multiple groups.

Rule 5:

Minimize Variables: Aim to minimize the number of variables within each group. If a group spans multiple rows or columns, the variables should change between rows or columns to form the simplest term.

Rule 6:

Don't Include Zeros: Groups should only consist of ones (1s). Zeros (0s) can be ignored when forming groups.



Rule 7:

Gray Code Ordering: In larger K-maps, it's common to use Gray code ordering for variable combinations. Gray code ensures that adjacent combinations differ by only one variable change, making it easier to spot groups.

Rule 8:

Use Don't Cares (X): If the K-map contains "Don't Care" conditions (usually denoted as "X" in a cell), you can include them as either ones or zeros in your groups, depending on which option results in a simpler expression.

- **The truth table for minterm :-**

A	B	C	Min term
0	0	0	$m_0 = \bar{A}\,\bar{B}\,\bar{C}$
0	0	1	$m_1 = \bar{A}\,\bar{B}\,C$
0	1	0	$m_2 = \bar{A}\,B\,\bar{C}$
0	1	1	$m_3 = \bar{A}\,B\,C$
1	0	0	$m_4 = A\,\bar{B}\,\bar{C}$
1	0	1	$m_5 = A\,\bar{B}\,C$
1	1	0	$m_6 = A\,B\,\bar{C}$
1	1	1	$m_7 = A\,B\,C$

- **The truth table for maxterm**

A	B	C	Max term
0	0	0	$M_0 = A + B + C$
0	0	1	$M_1 = A + B + \bar{C}$
0	1	0	$M_2 = A + \bar{B} + C$
0	1	1	$M_3 = A + \bar{B} + \bar{C}$
1	0	0	$M_4 = \bar{A} + B + C$
1	0	1	$M_5 = \bar{A} + B + \bar{C}$
1	1	0	$M_6 = \bar{A} + \bar{B} + C$
1	1	1	$M_7 = \bar{A} + \bar{B} + \bar{C}$

- **Minimize the following expression using K-map :**

- $F(W,X,Y,Z)=\sum m(3,4,7,12)$

WX \ YZ	[00] Y'Z'	[01] Y'Z	[11] YZ	[10] YZ'
[00] W'X'	0 0	0 1	1 3	0 2
[01] W'X	1 4	0 5	1 7	0 6
[11] WX	1 12	0 13	0 15	0 14
[10] WX'	0 8	0 9	0 11	0 10

$$F = W'XY'Z' + WXY'Z' + W'X'YZ + W'XYZ$$

$$F = XY'Z'(W' + W) + W'YZ(X' + X)$$

$$\boxed{F = XY'Z' + W'YZ}$$

<u>CONCLUSION :</u>

Thereby we have understand the use of K map in minimizing equation and have hereby also problem on them.