BOOLEAN ALGEBRA AND DE-MORGAN'S THEOREMS

Boolean Algebra

In the middle of the 19^{th} century, an English mathematician George Boole developed rules for manipulations of binary variables, known as Boolean algebra. This is the basis for all digital systems like computers, calculators, etc. Binary variables can be represented by a letter symbol such as A, B, X, Y, etc. The variable can have only one of the two possible values at any time either 0 or 1. The following are the operators used in Boolean algebra.

Operator	**Operation**
=	Equal (Assignment)
+	OR (Logic addition)
.	AND (Logic multiplication)
(Bar)	NOT (Complement)

Boolean postulates and laws

Theorems

S.No	Theorem
1.	$\overline{\overline{A}} = \mathrm{A}$
2.	$\overline{\mathrm{A}}\,(A + \mathrm{B}) = \mathrm{A\,B}$
3.	$\mathrm{A + A\,B = A}$
4.	$\overline{(}\mathrm{A + B)\,(A} + B) = \mathrm{A}$
5.	$\overline{\mathrm{A\,B}} + A\,B = \mathrm{A}$
6.	$\mathrm{A\,(A + B) = A}$
7.	$\overline{\mathrm{A}} + \mathrm{A}\,B = \mathrm{A + B}$
8.	$\mathrm{A + B\,C = (A + B)\,(A + C)}$

Laws of complementation (NOT Laws)

S.No	Law
1.	0 = 1
2.	1 = 0
3.	$\overline{\mathrm{I}}$f A = 0, A = 1
4.	$\overline{\mathrm{I}}$f A = 1, A = 0
5.	$\overline{\overline{A}} = \mathrm{A}$

AND Laws

S.No	Law
1.	$A \cdot 0 = 0$
2.	$A \cdot 1 = A$
3.	$A \cdot A = A$
4.	$\overline{A} \cdot A = 0$

OR Laws

S.No	Law
1.	$A + 0 = A$
2.	$A + 1 = 1$
3.	$A + A = A$
4.	$\overline{A} + A = 1$

Commutative Laws

The commutative laws allow the change in position of an AND or OR variable.

S.No	Law
1.	$A + B = B + A$
2.	$A \cdot B = B \cdot A$

Associative Laws

The associative laws allow the grouping of variables.

S.No	Law
1.	$A + (B + C) = (A + B) + C$
2.	$A \cdot (B \cdot C) = (A \cdot B) \cdot C$

Distributive Laws

The distributive laws allow the factoring or multiplying out of expressions.

S.No	Law
1.	$A \cdot (B + C) = (A \cdot B) + (A \cdot C)$
2.	$A + (B \cdot C) = (A + B) \cdot (A + C)$

DeMorgan’s theorems

De Morgan contributed a lot for the Boolean algebra. Out of them, the following twotheorems are very important. It allows transformation from a sum-of-products form to a products-of-sums form. De Morgan‟s theorems are also useful in simplifying Boolean equations.

First theorem

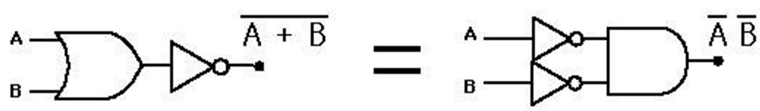
Statement:

The complement of a sum of variables is equal to the product of their complements.

Equation:

$$\overline{A + B} = \bar{A} \cdot \bar{B}$$

Logic diagram :



Proof:

We have to show the left side equals the right side for all possible values of A and B. The following table proves the De Morgan‟s first theorem.

A	B	A + B	$\overline{A + B}$ (LHS)	$\bar{A}$	$\bar{B}$	$\bar{A} . \bar{B}$ (RHS)
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

In the above table, LHS = RHS. Hence, the De Morgan‟s first theorem is proved.

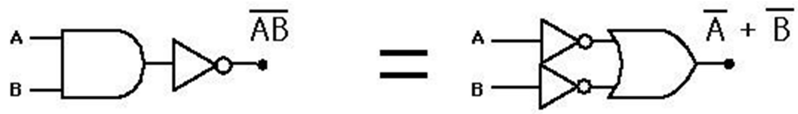
Second theorem

Statement:

The complement of a product of variables is equal to the sum of their complements. Equation:

$$\overline{A \cdot B} = \bar{A} + \bar{B}$$

Logic diagram :



Proof:

We have to show the left side equals the right side for all possible values of A and B. The following table proves the De Morgan‟s second theorem.

A	B	A . B	$\overline{A.B}$ (LHS)	$\overline{A}$	$\overline{B}$	$\overline{A}+\overline{B}$(RHS)
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

In the above table, LHS = RHS. Hence, the De Morgan‟s second theorem is proved.

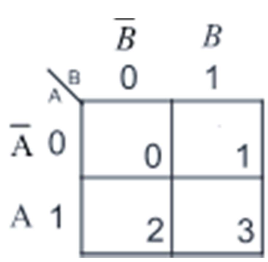
KARNAUGH MAP

Karnaugh map (K-map) is a graphical technique which provides a systematic method for simplifying Boolean expressions. In this technique, the information contained in truth table is represented in a map form called K-map.

Two-variable K-map

Follow the steps given below to draw a two-variable K-map.

1. Draw a table with 2 rows and 2 columns with an extension line at the top left corner.
2. Mark A for the Rows and B for the Columns.
3. Put $\overline{A}$ and 0 for the first row. Put A and 1 for the second row.
4. Put $\overline{B}$ and 0 for the first column. Put B and 1 for the second column.
5. Put the numbers inside each box (at the bottom right corner) as 0, 1, 2 and 3 row wise.



Three-variable K-map

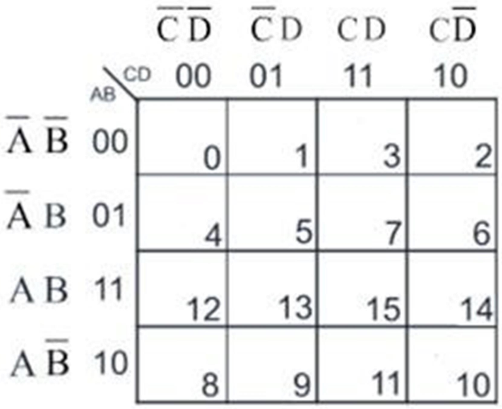
Follow the steps given below to draw a three-variable K-map.

1. Draw a table with 4 rows and 2 columns with an extension line at the top left corner.
2. Mark AB for the Rows and C for the Columns.
3. Put $\overline{AB}$ and 00 for the first row, $\overline{A}$B and 01 for second row, A B and 11 for third row, $A\overline{B}$
and 10 for forth row.
4. Put $\overline{C}$ and 0 for the first column. Put C and 1 for the second column.
5. Put the numbers inside each box (at the bottom right corner) as 0, 1, 2, 3, 6, 7, 4 and 5 row wise.

Four-variable K-map

Follow the steps given below to draw a three-variable K-map.

1. Draw a table with 4 rows and 4 columns with an extension line at the top left corner.
2. Mark AB for the Rows and CD for the Columns.
3. Put $\overline{AB}$ and 00 for the first row, $\overline{A}$B and 01 for second row, A B and 11 for third row, $A\overline{B}$
and 10 for forth row.
4. Put $\overline{CD}$ and 00 for the first row, $\overline{C}$ D and 01 for second row, C D and 11 for third row, C$\overline{D}$ and 10 for forth row.
5. Put the numbers inside each box (at the bottom right corner) as 0, 1, 3, 2, 4, 5, 7, 6, 12,
13, 15, 14, 8, 9, 11 and 10 row wise.



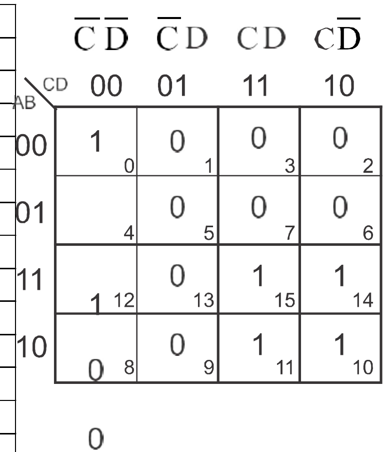
1.1.1 Truth table to K-map

Once the K-map is drawn we have to transfer the truth table information to the K-map. Follow the steps given below to fill up the K-map.

1. Look at the truth table carefully.
2. Identify the input combinations (A, B, C and D) for which the outputs are 1s. (Use complements of the variables for zeros).
3. Put 1s in the corresponding boxes in the K-map.
4. Put 0s in all the remaining boxes.

<u>Example</u>

Inputs				Output
A	B	C	D	Y
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1



Logic equation to K-map

Sometimes, logic equation will be given instead of truth table. Follow the steps given below to fill up the K-map using the logic equation.

1. Look at the logic equation carefully.
2. Each term in the equation is called min-term. Put 1s in the corresponding boxes in the K- map for all the min-terms.
3. Put 0s in all the remaining boxes.

Example

$$F = \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}B\bar{C}\bar{D} + A\bar{B}C\bar{D} + A\bar{B}CD + ABC\bar{D} + ABCD$$

		$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
	AB \ CD	00	01	11	10
$\bar{A}\bar{B}$	00	1 $_{0}$	0 $_{1}$	0 $_{3}$	0 $_{2}$
$\bar{A}B$	01	1 $_{4}$	0 $_{5}$	0 $_{7}$	0 $_{6}$
AB	11	0 $_{12}$	0 $_{13}$	1 $_{15}$	1 $_{14}$
$A\bar{B}$	10	0 $_{8}$	0 $_{9}$	1 $_{11}$	1 $_{10}$

Logic function to K-map

If the logic function is given, follow the steps given below to fill up the K-map.

1. Look at the logic function carefully.
2. Put 1s in the corresponding boxes in the K-map for all the numbers given in the equation.
3. Put 0s in all the remaining boxes.

Example

$$F = \sum m(0,4,10,11,14,15)$$

		$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
	AB \ CD	00	01	11	10
$\bar{A}\bar{B}$	00	1 $_{0}$	0 $_{1}$	0 $_{3}$	0 $_{2}$
$\bar{A}B$	01	1 $_{4}$	0 $_{5}$	0 $_{7}$	0 $_{6}$
AB	11	0 $_{12}$	0 $_{13}$	1 $_{15}$	1 $_{14}$
$A\bar{B}$	10	0 $_{8}$	0 $_{9}$	1 $_{11}$	1 $_{10}$

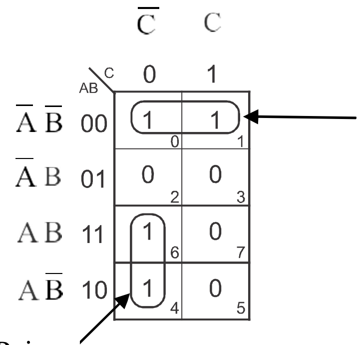
1.1.2 Pairs, Quads and Octets

Once the K-map is filled with 1s and 0s, we have to identify the pairs, quads and octets in order to simplify the Boolean expressions.

Pairs

If there are two 1s adjacent to each other, vertically or horizontally, we can form a pair. A pair eliminates one variable and its complement.

Pair



Pair

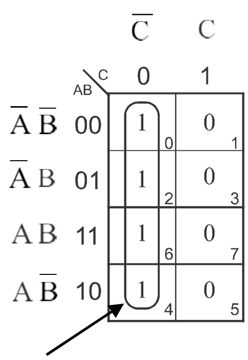
Here, for the boxes 0 and 1 ie. first row and first column & second column, C is changing from complement form to uncomplement form. Hence, C is eliminated and we get AB.For the boxes 4 and 6 ie.first column and third row & forth row, B is changing from

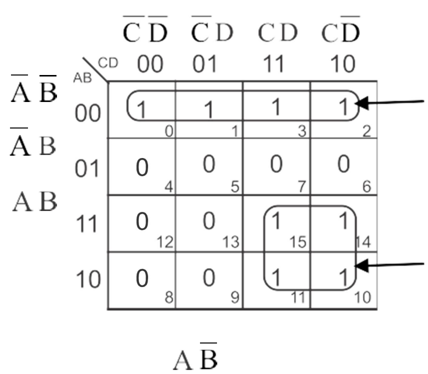
uncomplement form to complement form. Hence, B is eliminated and we get AC. The simplified equation is AB + AC.

Quads

If there are four 1s adjacent to each other, vertically or horizontally, we can form a quad. Two variables and their complements are eliminated.

Quad

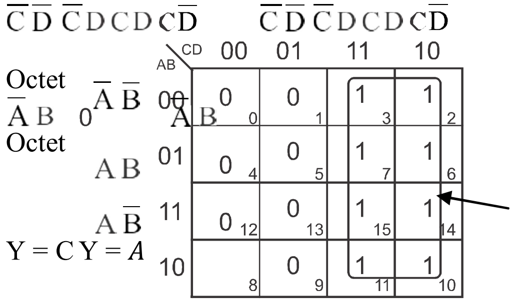


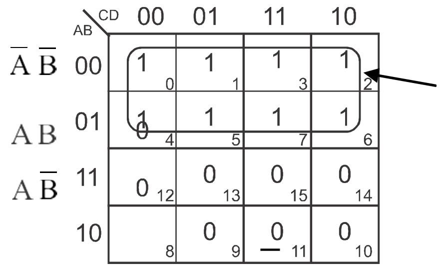


Quad

Octets Quad Y = $\overline{C}$ Y = $\overline{AB}$ + A C

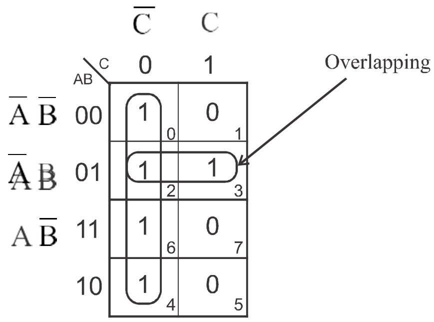
If there are eight 1s adjacent to each other, vertically or horizontally, we can form an octet. Three variables and their complements are eliminated.





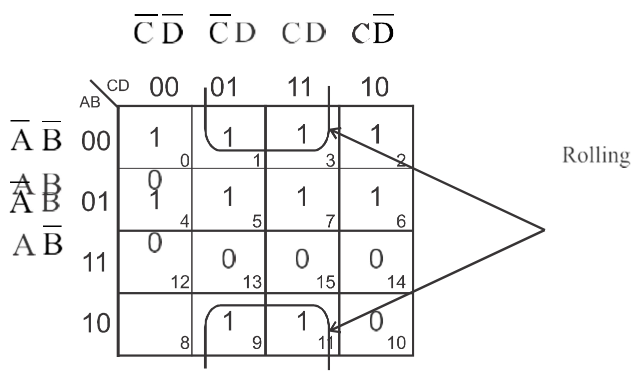
Overlapping groups

We can use 1s in more than one loop of pairs, quads and octets. This type of groups is called overlapping group.



Rolling

It is obvious that when we roll the map, the first row and last row are adjacent to each other. Similarly, the first column and last column are adjacent to each other. This is called rolling the map. By this way also, we can form pairs, quads and octets.



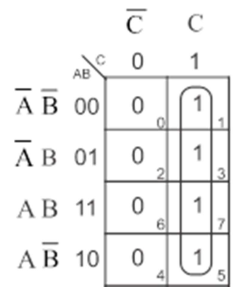
Redundant group

It is a group whose all 1s are overlapped by other groups. Redundant groups must be removed.

1.1.3 Simplification using K-map

- Note down the corresponding variables for the pairs, quads and octets from the K-map.
- If any variable goes from un-complemented form to complemented form, the variable can be eliminated to form the simplified equation. Write the simplified products.
- It is noted that one variable & their complement will be eliminated for pairs, two variables & their complements for quads and three variables & and their complements for octets.
- By OR ing all the simplified products, we get the Boolean equation corresponding to the entire K-map.

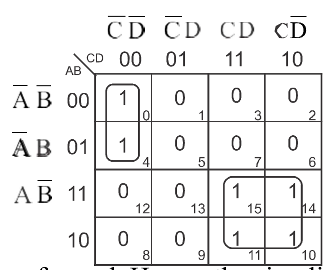
Example 1 :
Simplify the following logic equation Y = $\overline{AB}$ C +$\overline{A}$BC + A$\overline{B}$C + ABC Solution:
It is a three variable equation. The K-map for the given equation is,



ng Here, only one quad is formed. Hence, the simplified logic equation is,
Y = C
Example 2 :
Simplify the following logic equation

F = $\overline{ABCD}$ + $\overline{A}$ B $\overline{CD}$ + A $\overline{B}$ C $\overline{D}$ + A $\overline{B}$ C D + A B C $\overline{D}$ + A B C D

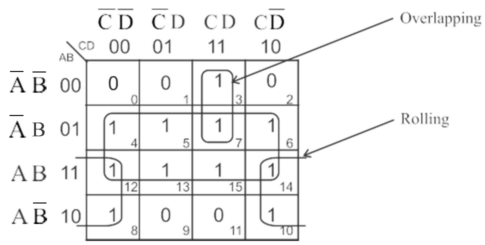
It is a four variable equation. The K-map for the given equation is,



Here, one pair and one quad are formed. Hence, the simplified logic equation is,

Y = $\overline{ACD}$ + AC

Example 3 :
Simplify the following min term function Y = Σm (3, 4, 5, 6, 7, 8, 10, 12, 13, 14, 15)
The maximum number in the function is 15. Hence, we have to draw a four variable K-map.



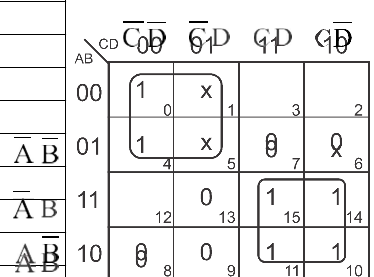
Here, we have one pair, one quad and one octet. For pair, the equation is $\bar{A}CD$. For quad, $A\bar{D}$.For octet, B. Hence, the simplified equation is,

$Y = \bar{A}CD + A\bar{D} + B$

1.1.4 Don't care conditions

In some logic circuits, certain input conditions do not produce any specified outputs ie. neither 0 nor 1. These conditions are called don"t care conditions and they are marked as „X" in the truth table. The same may be marked in the K-map as „X" in the corresponding boxes. We can also use „X" as „1" to form pairs, quads and octets, if necessary.
Consider the following truth table.

Inputs				Output
A	B	C	D	Y
0	0	0	0	1
0	0	0	1	X
0	0	1	0	X
0	0	1	1	0
0	1	0	0	1
0	1	0	1	X
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1



When we consider „X" condition, we get two quads. Hence, the simplified equation is,

$Y = \bar{A}\bar{C} + AC$