Logical Design

Lecture 7: SOP, POS, and K-Map

Produced By: Dr. Haleema Essa

2023-2024

Sum of Product (SOP)

The ***Sum of Product (SOP)*** expression comes from the fact that two or more products (AND) are summed (OR) together. That is the outputs from two or more AND gates are connected to the input of an OR gate so that they are effectively OR'ed together to create the final AND-OR logical output. For example:

$$Q = (A.B) + (\bar{B}.C) + (A.1)$$

However, Boolean functions can also be expressed in a nonstandard sum of product forms like that shown below but they can be converted to a standard SOP form by expanding the expression. So:

$$Q = A.\bar{B}(\bar{C} + C) + ABC$$

Becomes in sum-of-product terms:

$$Q = A.\bar{B}.\bar{C} + A.\bar{B}.C + ABC$$

<u>Example:</u> The following Boolean Algebra expression is given as:

$$Q = \bar{A}(\bar{B}C + BC + B\bar{C}) + ABC$$

1. Convert this logical equation into an equivalent SOP term.
2. Use a truth table to show all the possible combinations of input conditions to produce an output **1**.
3. Draw a logic gate diagram for the expression.

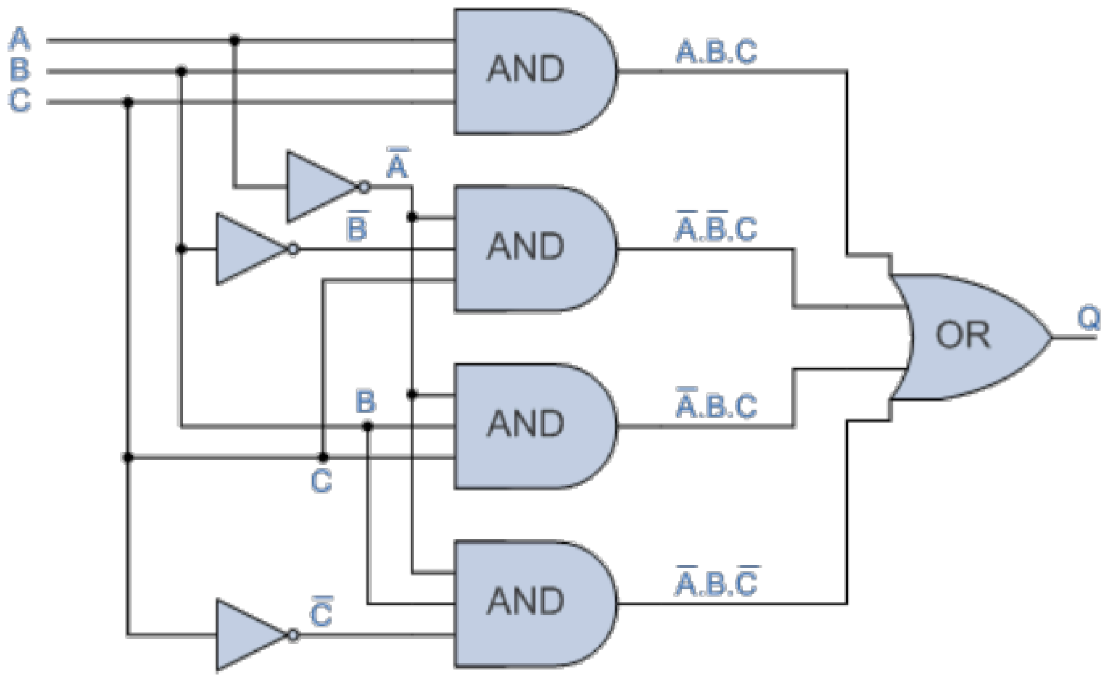
1. Convert to SOP term

$$Q = A.B.C + \bar{A}.\bar{B}.C + \bar{A}.B.C + \bar{A}.B.\bar{C}$$

2. Truth Table: Sum of Product Truth Table Form

inputs			output	Product
A	B	C	Q	
0	0	0	0	
0	0	1	1	$\bar{A}.\bar{B}.C$
0	1	0	1	$\bar{A}.B.\bar{C}$
0	1	1	1	$\bar{A}.B.C$
1	0	0	0	
1	0	1	0	
1	1	0	0	
1	1	1	1	$A.B.C$

3. Logic Gate SOP Diagram



Product of Sum (POS)

The ***Product of Sum (POS)*** expression comes from the fact that two or more sums (OR's) are added (AND'ed) together.
That is the outputs from two or more OR gates are connected to the input of an AND gate so that they are effectively AND'ed together to create the final (OR AND) output.

Product of Sum Expressions: $Q = (A + B).(B + C).(A + 1)$

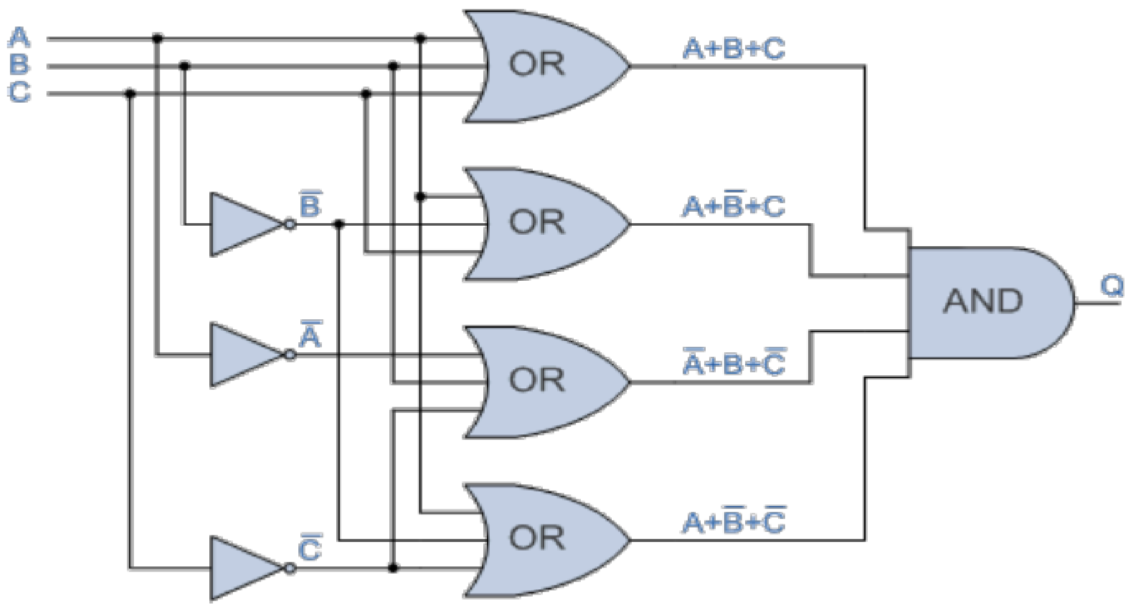
$Q = A + (BC)$ ➔ using the distributive law : $Q = (A + B)(A + C)$

<u>Example:</u> The following boolean algebra expression is given as:

$$Q = (A + B + C)(A + \bar{B} + C)(A + \bar{B} + \bar{C})(\bar{A} + B + \bar{C})$$

1. Use a truth table to show all the possible combinations of input conditions to produce output **0**.
2. Draw a logic gate diagram for the POS expression.

inputs			output	Product
A	B	C	Q	
0	0	0	0	A+B+C
0	0	1	1	
0	1	0	0	$A + \bar{B} + C$
0	1	1	0	$A + \bar{B} + \bar{C}$
1	0	0	1	
1	0	1	0	$\bar{A} + B + \bar{C}$
1	1	0	1	
1	1	1	1	



Karnaugh Map (K-map)

A ***Karnaugh map (K-map)*** is a method for minimizing Boolean expressions without applying Boolean algebra theorems and equation manipulations by translating a truth table to its equivalent logic circuit in a simple orderly process, as presented by Maurice Karnaugh in 1953.

Advantages and Disadvantages of Karnaugh Map

The advantages of K-map are:

1. K-map simplification does not demand the knowledge of Boolean algebraic theorems.

2. It requires a smaller number of steps when compared to the algebraic minimization technique.

3. It is easy to convert a truth table to a k-map and a k-map to the Sum of Products form equation.

4. The K-map method is faster and more efficient than other simplification techniques of Boolean algebra.

Disadvantages of K-map are:

1. The complexity of the K-map simplification process increases with the increase in the number of variables.
2. The minimum expression obtained might not be unique.

Karnaugh Map Details:

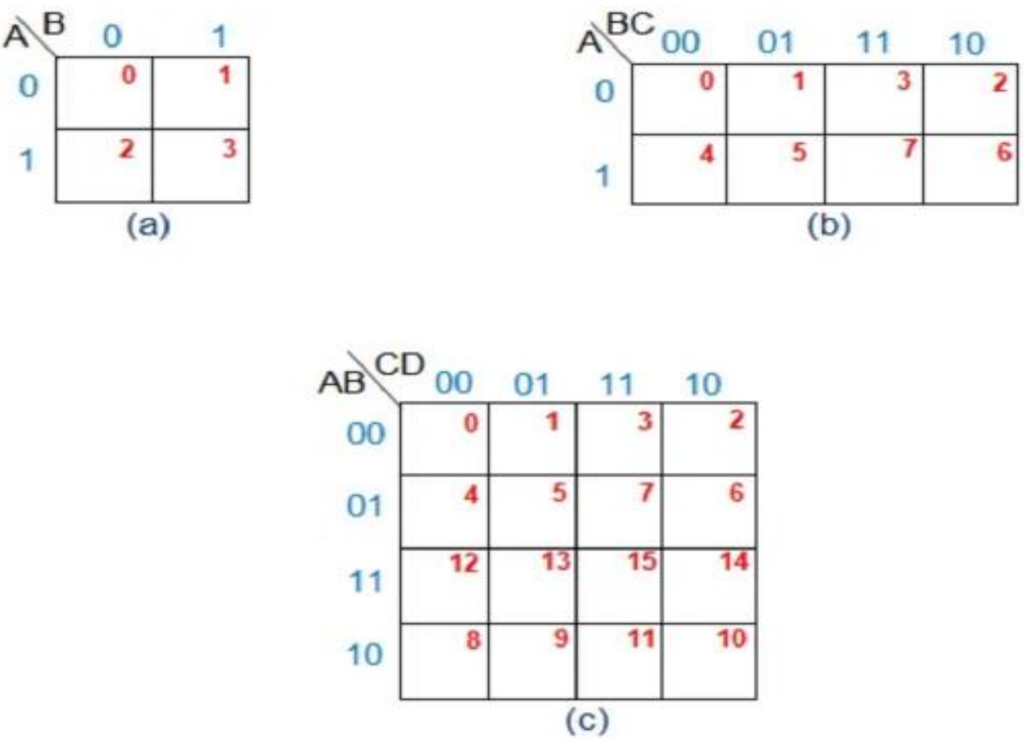
In general, if there are **n** inputs, then the corresponding K-map has to be of $\mathbf{2^n}$ cells.

For example:

• If the number of input variables is 2, then we have to consider a K-map with 4 (2^2) cells.

• While if there are 3 input variables, then we require an 8 (2^3) cell K-map.

• Similarly for 4 inputs one gets 16 (2^4) cell K-map and so on.



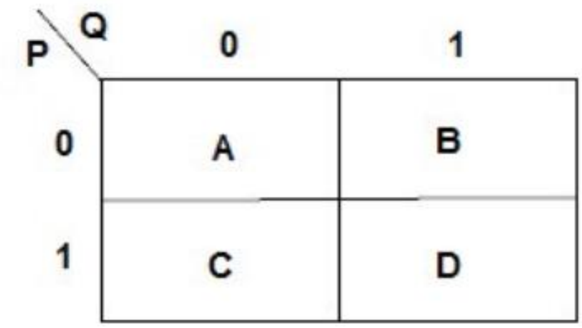
Karnaugh Maps for (a) Two Variables (b) Three Variables (c) Four Variables

Example: Two variable map

For the following truth table:

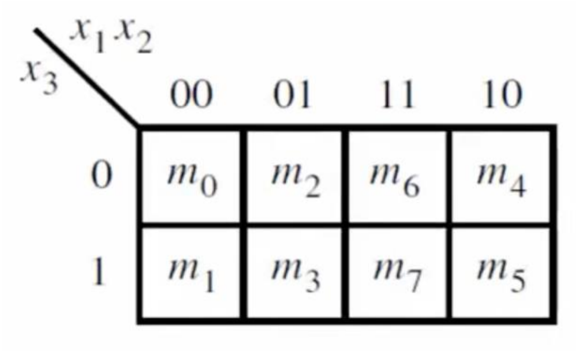
P	Q	output
0	0	A
0	1	B
1	0	C
1	1	D

The K-map for the truth table:



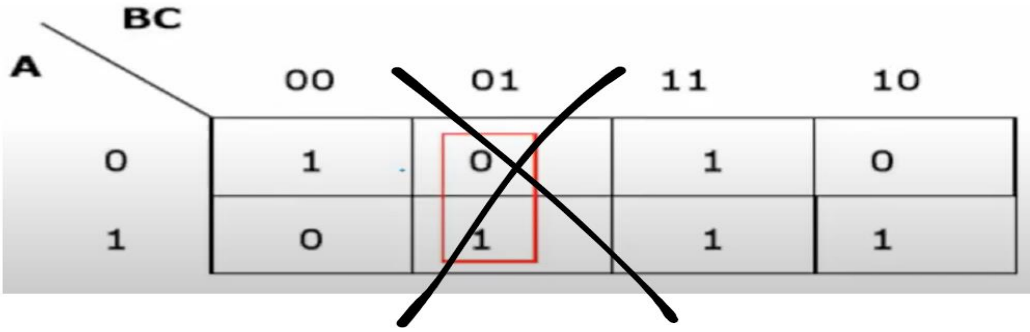
Example: Three-variable map

x_1	x_2	x_3	
0	0	0	m_0
0	0	1	m_1
0	1	0	m_2
0	1	1	m_3
1	0	0	m_4
1	0	1	m_5
1	1	0	m_6
1	1	1	m_7

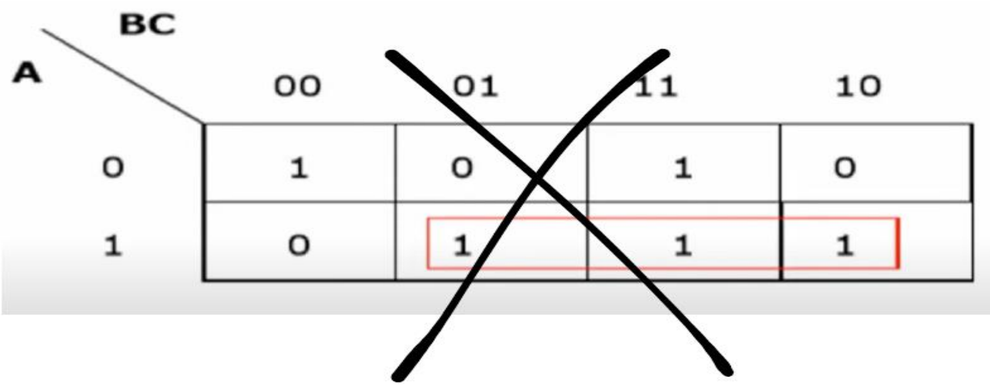


K-map Variables Grouping Rules

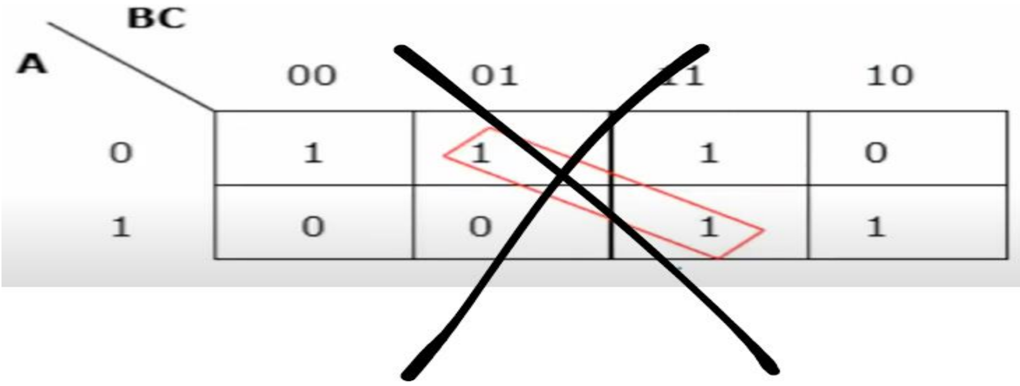
Rule 1: The group ***should not contain any Zeros.***



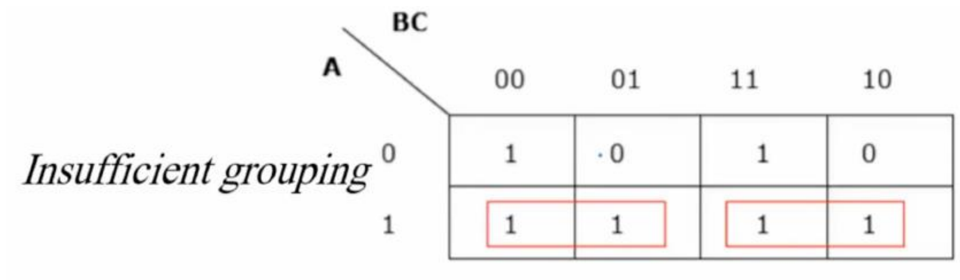
Rule 2: Groups must contain ***2^n*** cells (n = 0,1,2,3, …).

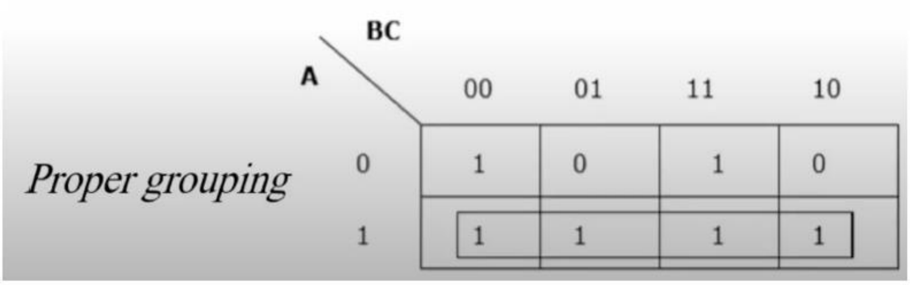


Rule 3: Grouping must be horizontal or vertical but ***not diagonal***.

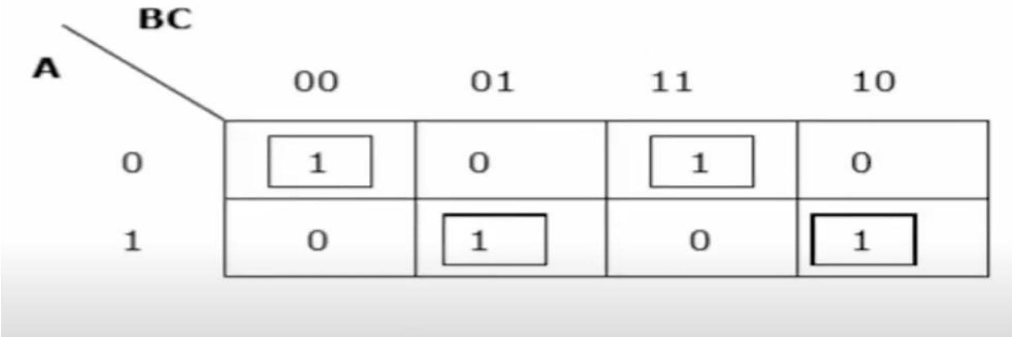


Rule 4: Groups must cover ***as large as possible.***

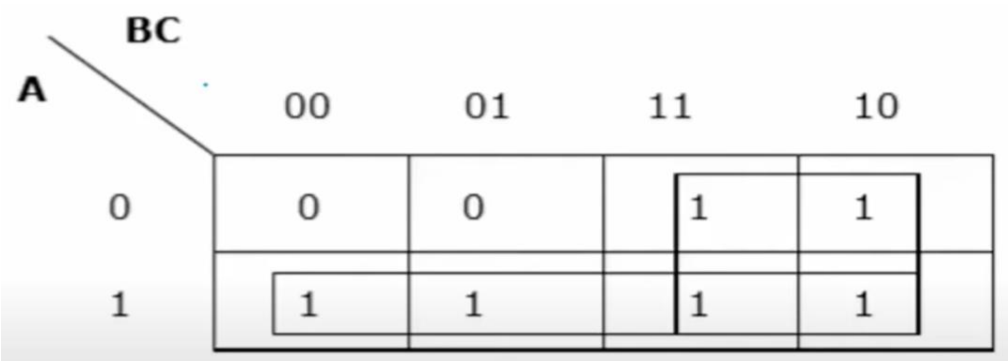




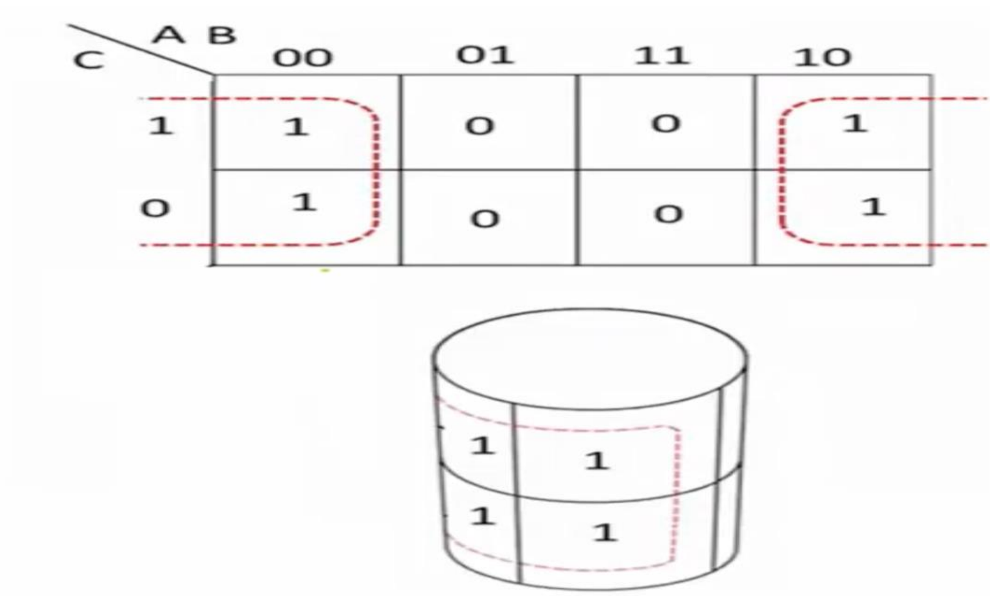
Rule 5: If **1** of any cell cannot be grouped with any other cell, then ***it will act as a group itself.***



Rule 6: Groups ***may overlap*** but there should be as few groups as possible.



Rule 7: The leftmost cell/cells can be grouped with the rightmost cell/cells and the topmost cell/cells can be grouped with the bottommost cell/cells.

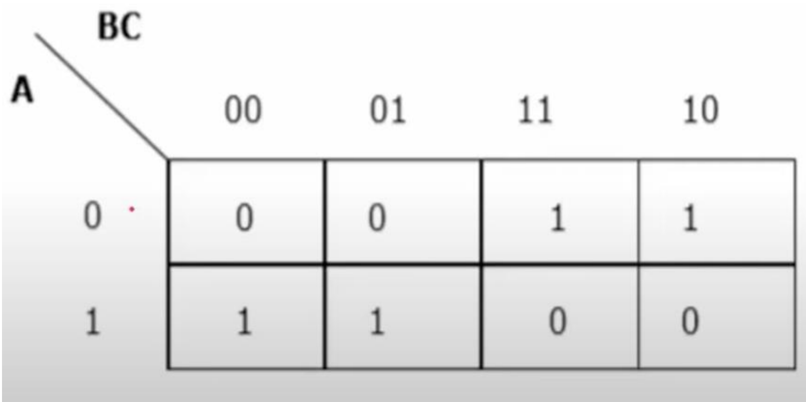


__Example1:__ Simplify the following Boolean expression using k-map method:

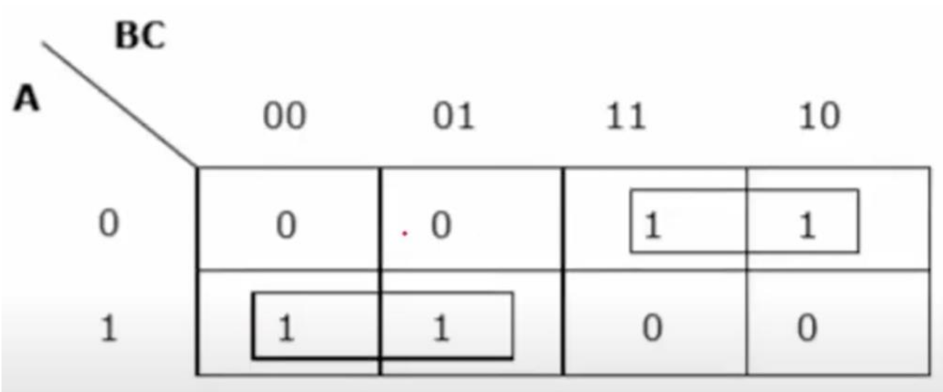
$$F(A,B,C) = \bar{A}BC + \bar{A}B\bar{C} + A\bar{B}\bar{C} + A\bar{B}C$$

__Solution:__ Create the truth table first then k-map

Inputs			Output
A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0



Now the **1s** will be grouped according the above rules

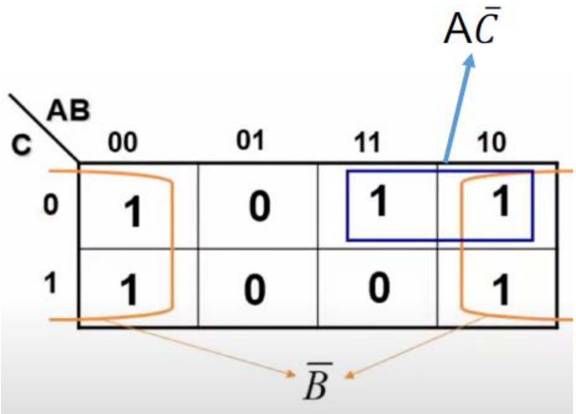


$$F(A,B,C) = \bar{A}B + A\bar{B}$$

__Example1:__ Use k-map to derive the minimal SOP for the output Y(A,B,C) of the given truth table below:

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

C \ AB	00	01	11	10
0	1	0	1	1
1	1	0	0	1



$$Y = \bar{B} + A\,\bar{C}$$

Four variables K-maps

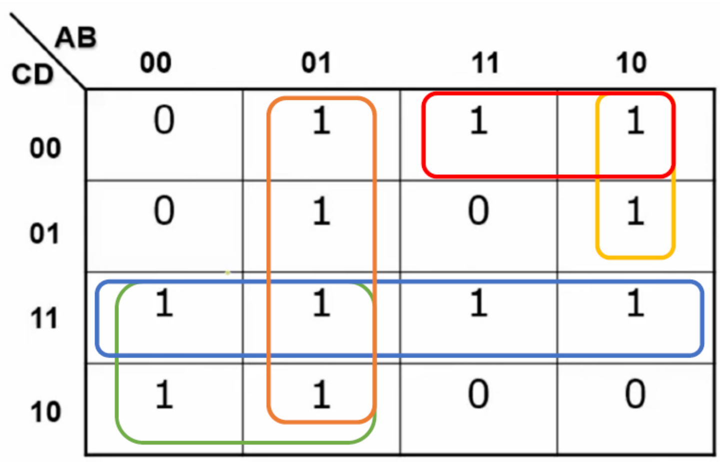
Number of Cells = 2^4=16

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

Example: Using k-map to derive the minimal SOP for the output F(A,B,C,D) of the given truth table:

A	B	C	D	*F*
0	0	0	0	0
0	0	0	1	0
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	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

CD \ AB	00	01	11	10
00	0	1	1	1
01	0	1	0	1
11	1	1	1	1
10	1	1	0	0



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

H.W1: Using K-map, derive minimal SOP for output Y(A,B,C) whose truth table is given below:

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

H.W2: For the following K-map, derive the minimal SOP.

x_3x_4 \ x_1x_2	00	01	11	10
00	0	0	0	0
01	0	0	1	1
11	1	1	1	1
10	1	1	1	1