UNIT 4 :Simplification of Boolean Algebra

Presented by

Er.Harendra Bikram Shah

(Lecturer)

OUTLINES

- K-Map of 2,3,4 variables
- Simplification and Realization using NAND and NOR gates
- Practical Design Steps

K-Map

Introduction to K-map:

- In previous chapters, we have simplified the Boolean functions using Boolean postulates and theorems.
- It is a time consuming process and we have to re-write the simplified expressions after each step.
- To overcome this difficulty, Karnaugh introduced a method for simplification of Boolean functions in an easy way.
- **A Karnaugh map (K-map) is a pictorial method used to minimize Boolean expressions without having to use Boolean algebra theorems and equation manipulations.**
- A K-map can be thought of as a special version of a truth table .
- It is a graphical method, which consists of 2n cells for ‘n’ variables.
- In 1953, the American Physicist Maurice **Karnaugh** invented K-Map

K-Map

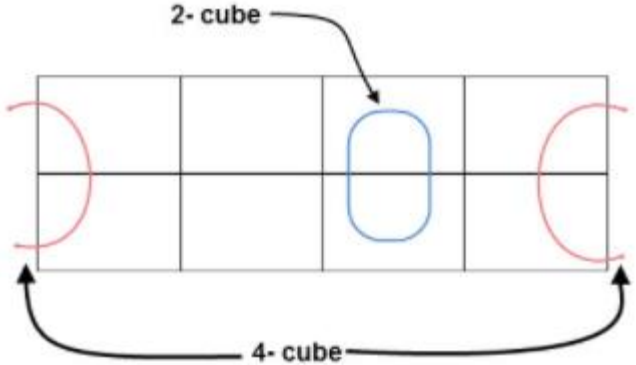
Significance/Application/Importance of K-map:

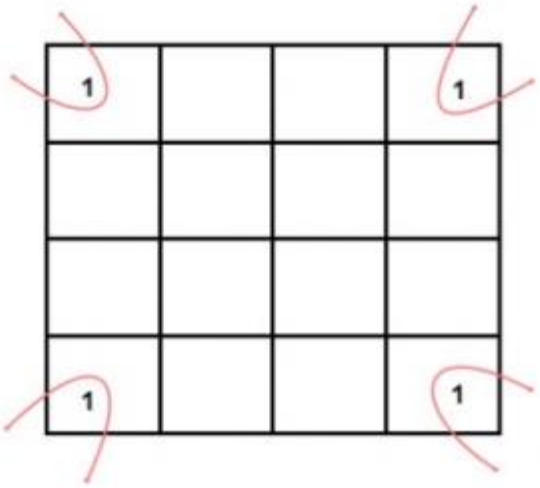
- Boolean algebra can be simplified systematically
- K-map reduces logic functions quickly and simply.
- K-maps are both faster and Easier for more number for variables like 4 or more numbers of variables.

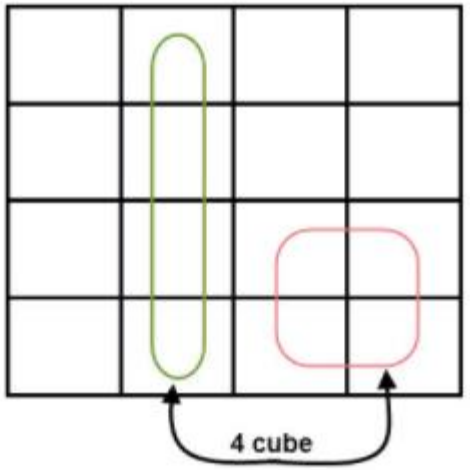
Steps to solve expression using K-map-

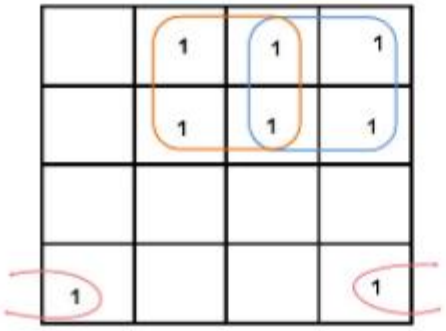
1.Select K-map according to the number of variables.
2.Identify minterms or maxterms as given in problem.
3.For SOP put 1's in blocks of K-map respective to the minterms (0's elsewhere).
4.For POS put 0's in blocks of K-map respective to the maxterms(1's elsewhere).
5.Make rectangular groups containing total terms in power of two like 2,4,8 ..(except 1) and try to cover as many elements as you can in one group.
6. Don't care "x" should also be included while grouping to make a larger possible group.
7.From the groups made in step 5 find the product terms and sum them up for SOP form.

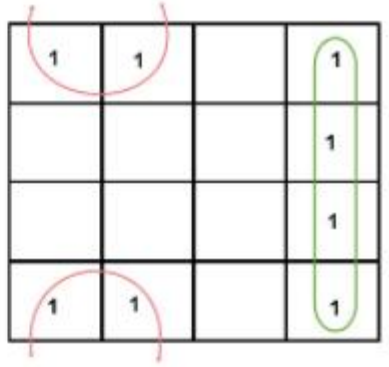
Pairing Ways in k-map





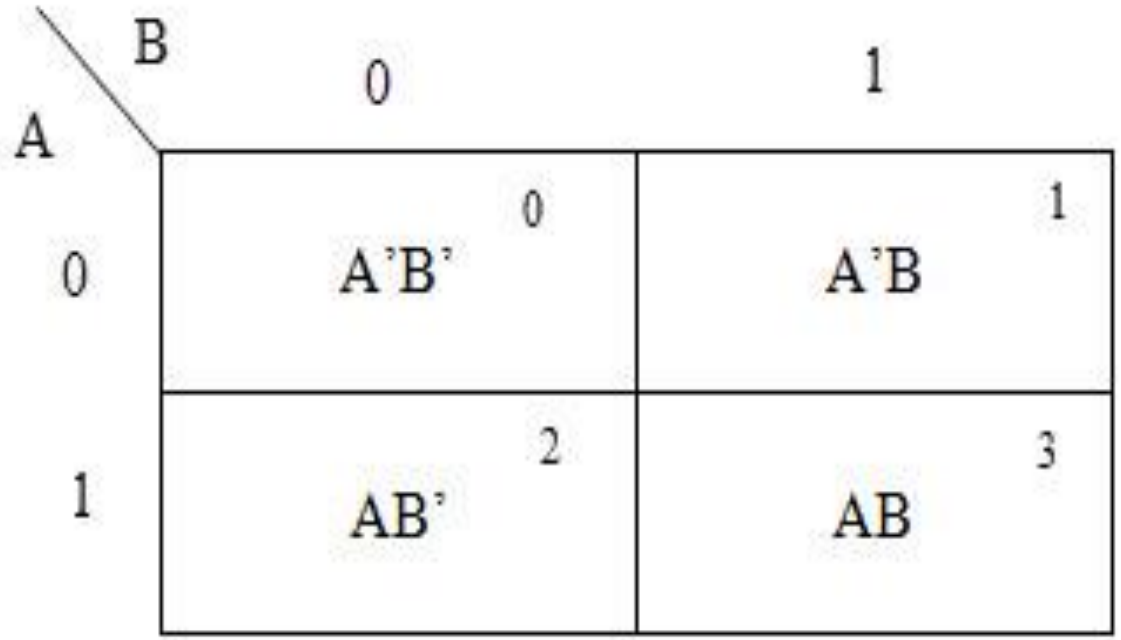






K-Map of 2 variables

SOP FORM

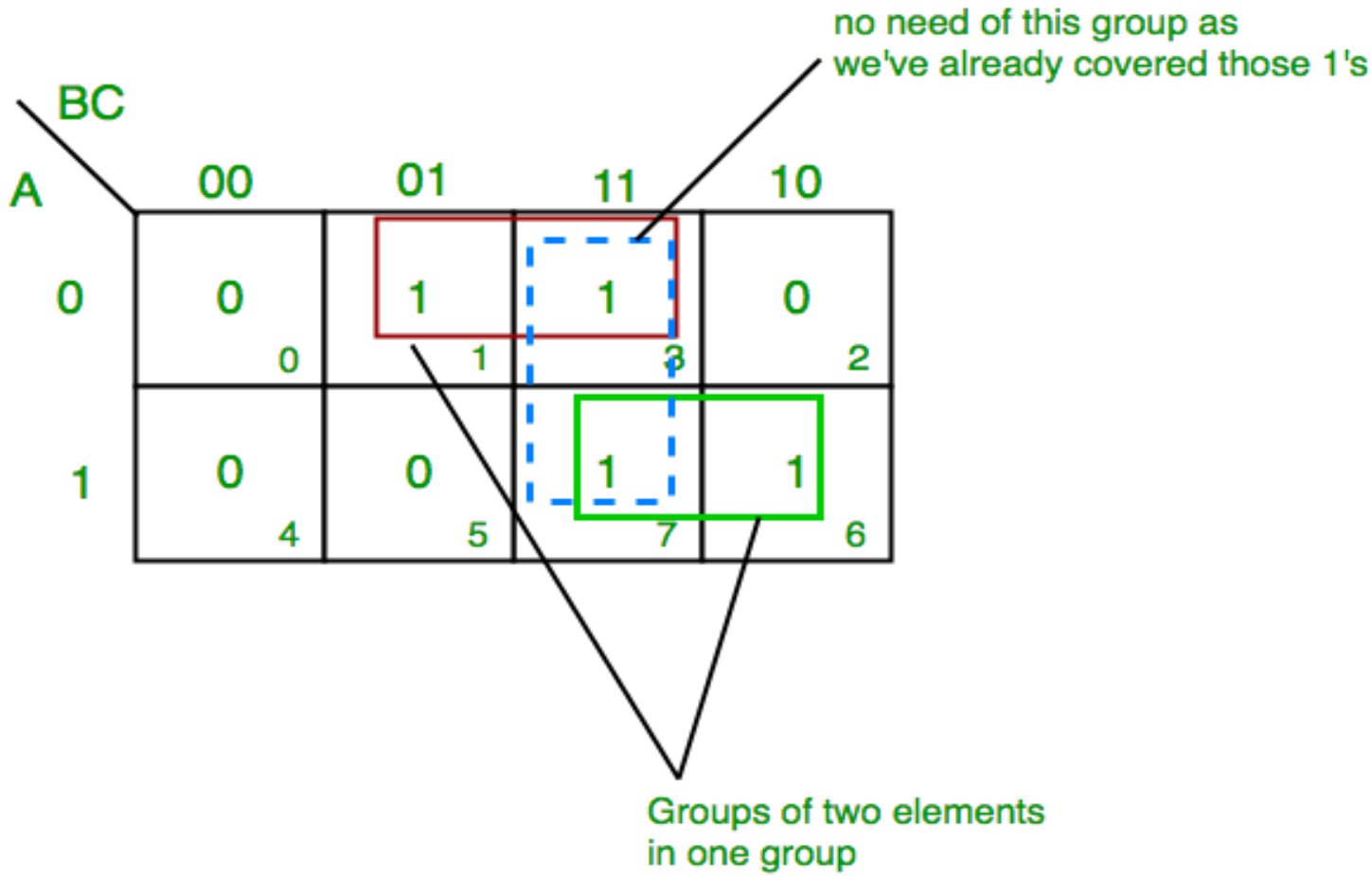


K-Map of 3 variables

SOP FORM

1.K-map of 3 variables-

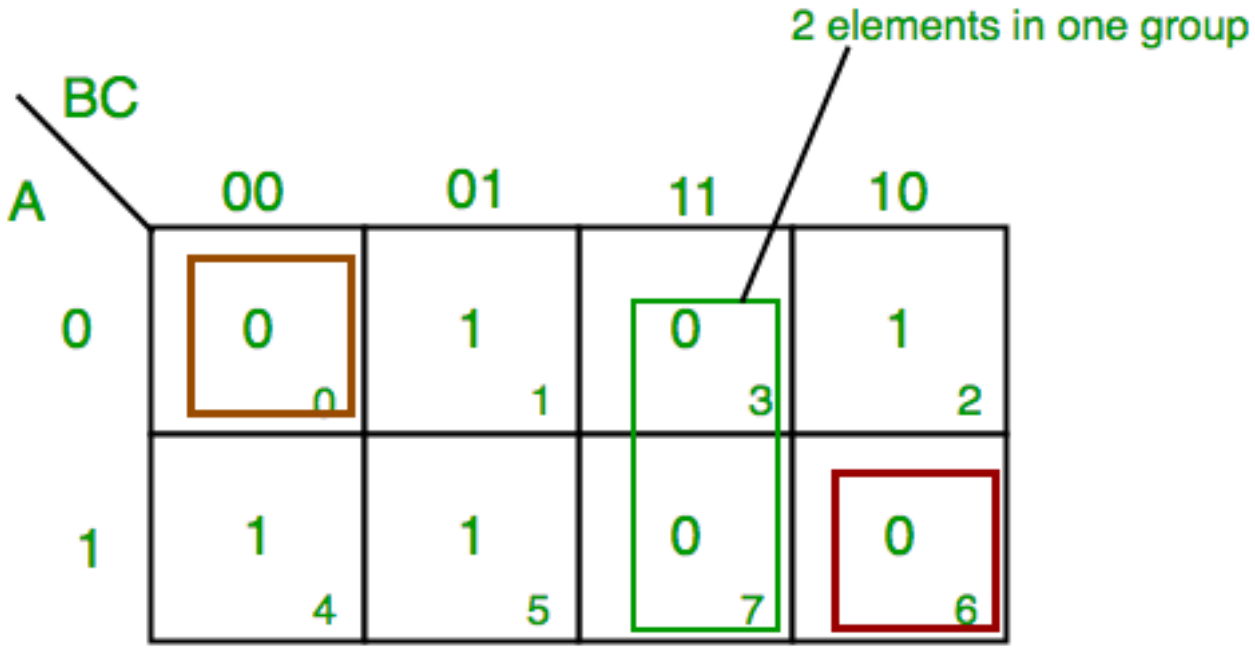
Z= ∑A,B,C(1,3,6,7)



From **red** group we get product term—
A'C
From **green** group we get product term—
AB
Summing these product terms we get- **Final expression (A'C+AB)**

K-Map of 3 variables

SoP FORM



K-Map of 4 variables

SoP FORM

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

K-Map of 4 variables

PoS FORM

A+B \ C+D	C+D	C+D’	C’+D’	C’+D
A+B 0+0	0	1	3	2
A+B’ 0+1	4	5	7	6
A’B’ 1+1	12	13	15	14
A’+B 1+0	8	9	11	10

5 Variable K-Map

The number of cells in 5 variable K-map is thirty-two, since the number of variables is 5.
The following figure shows 5 variable K-Map.

- There is only one possibility of grouping 32 adjacent min terms.
- There are two possibilities of grouping 16 adjacent min terms. i.e.,
- grouping of min terms from m0 to m15 and m16 to m31.
- If v=0, then 5 variable K-map becomes 4 variable K-map.

V=0

WX \ YZ	00	01	11	10
00	m_0	m_1	m_3	m_2
01	m_4	m_5	m_7	m_6
11	m_{12}	m_{13}	m_{15}	m_{14}
10	m_8	m_9	m_{11}	m_{10}

V=1

WX \ YZ	00	01	11	10
00	m_{16}	m_{17}	m_{19}	m_{18}
01	m_{20}	m_{21}	m_{23}	m_{22}
11	m_{28}	m_{29}	m_{31}	m_{30}
10	m_{24}	m_{25}	m_{27}	m_{26}

In the above all K-maps, we used exclusively the min terms notation.
Similarly, you can use exclusively the Max terms notation.

K-Map with Don’t care

- Everything is same except the notation ‘X’ is used for don’t care condition
- Don’t care condition will remain same for both POS and SOP form