BINARY REPRESENTATION OF A STANDARD PRODUCT TERM

A standard product term is equal to 1 for only one combina of variable values. For example, the product term $A\bar{B}C\bar{D}$ is equal to 1 when A = 1, B = 0, C = 1, D = 0, and is 0 for all other combinations of values for the variables.

$A\bar{B}C\bar{D} = 1\cdot\bar{0}\cdot1\cdot\bar{0} = 1$

In this case, the product term has a binary value of 1010 (decimal 10).

A product term is implemented with an AND gate whose output is 1 only if each of its inputs is 1.

An SOP expression is equal to 1 only if one or more of the product terms in the expression is equal to 1.

BINARY REPRESENTATION OF A STANDARD SUM TERM

A standard sum term is equal to 0 for only one combination of variable values. For example, the sum term $A+\bar{B}+C+\bar{D}$ is 0 when A = 0, B = 1, C = 0, D = 1

$A+\bar{B}+C+\bar{D} = 0+\bar{1}+0+\bar{1} = 0$

The sum term has a binary value of 0101 (decimal 5).

A sum term is implemented with an OR gate whose output is 0 only if each of its inputs is 0.

A POS expression is equal to 0 only if one or more of the sum terms in the expression is equal to 0.

Converting Standard SOP to Standard POS

Step 1: Evaluate each product term in the SOP expression. That is, determine the binary numbers that represent the product terms.

Step 2: Determine all of the binary numbers not included in the evaluation in Step 1.

Step 3: Write the equivalent sum term for each binary number from Step 2 and express in POS form.

Using a similar procedure, POS can be converted to SOP.

$$\left.\begin{aligned} \overline{A.B} &= \bar{A} + \bar{B} \\ \overline{A+B} &= \bar{A} \cdot \bar{B} \end{aligned}\right\} \text{De Morgan's}$$

KARNAUGH MAP REPRESENTATION OF LOGIC FUNCTIONS

Karnaugh Map Technique is a graphical technique for systematically simplifying and manipulating Boolean expressions. It will produce the simplest SOP or POS expression possible, known as the minimum expression. It is also known as K-map.

Although the technique may be used for any number of variables, it is generally used up to six variables beyond which it becomes very cumbersome.

In an n-variable K-map, there are 2^n cells. Each cell corresponds to one of the combinations of n variables, since there are 2^n combinations of n variables.

Therefore, we see that for each row of the truth table, for each minterm and for each maxterm there is one specific cell in the K-map. In each map the variables and all possible values of the variables are indicated (the first bit corresponds to the first variable and the second bit corresponds to the second variable) to identify the cells. Gray code has been used for the identification of cells.

2-Variable K-map

A B	0	1
0	0	2
1	1	3

A B	0	1
0	$\bar{A}\bar{B}$	$A\bar{B}$
1	$\bar{A}B$	AB

A B	0	1
0	$A+B$	$\bar{A}+B$
1	$A+\bar{B}$	$\bar{A}+\bar{B}$

3- Variable K-map

AB C	00	01	11	10
0	0	2	6	4
1	1	3	7	5

AB C	00	01	11	10
0	$\bar{A}\bar{B}\bar{C}$	$\bar{A}B\bar{C}$	$AB\bar{C}$	$A\bar{B}\bar{C}$
1	$\bar{A}\bar{B}C$	$\bar{A}BC$	ABC	$A\bar{B}C$

AB C	00	01	11	10
0	$A+B+C$	$A+\bar{B}+C$	$\bar{A}+\bar{B}+C$	$\bar{A}+B+C$
1	$A+B+\bar{C}$	$A+\bar{B}+\bar{C}$	$\bar{A}+\bar{B}+\bar{C}$	$\bar{A}+B+\bar{C}$

4-variable K-map

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

AB CD	00	01	11	10
00	$\bar{A}\bar{B}\bar{C}\bar{D}$	$\bar{A}B\bar{C}\bar{D}$	$AB\bar{C}\bar{D}$	$A\bar{B}\bar{C}\bar{D}$
01	$\bar{A}\bar{B}\bar{C}D$	$\bar{A}B\bar{C}D$	$AB\bar{C}D$	$A\bar{B}\bar{C}D$
11	$\bar{A}\bar{B}CD$	$\bar{A}BCD$	$ABCD$	$A\bar{B}CD$
10	$\bar{A}\bar{B}C\bar{D}$	$\bar{A}BC\bar{D}$	$ABC\bar{D}$	$A\bar{B}C\bar{D}$

AB CD	00	01	11	10
00	$A+B+C+D$	$A+\bar{B}+C+D$	$\bar{A}+\bar{B}+C+D$	$\bar{A}+B+C+D$
01	$A+B+C+\bar{D}$	$A+\bar{B}+C+\bar{D}$	$\bar{A}+\bar{B}+C+\bar{D}$	$\bar{A}+B+C+\bar{D}$
11	$A+B+\bar{C}+\bar{D}$	$A+\bar{B}+\bar{C}+\bar{D}$	$\bar{A}+\bar{B}+\bar{C}+\bar{D}$	$\bar{A}+B+\bar{C}+\bar{D}$
10	$A+B+\bar{C}+D$	$A+\bar{B}+\bar{C}+D$	$\bar{A}+\bar{B}+\bar{C}+D$	$\bar{A}+B+\bar{C}+D$

Representation of Truth Table on K-map

Let the truth table of a 3-variable logic function be given by

Row No.	Inputs			Output
	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

$Y = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC$ – SOP form

$Y = (A+B+C)(A+\bar{B}+\bar{C})(\bar{A}+B+\bar{C})(\bar{A}+\bar{B}+C)$ – POS form

C \ AB	00	01	11	10
0	0 0	2 1	6 0	4 1
1	1 1	3 0	7 1	5 0

Representation of Canonical SOP form on K-map

A logical equation in canonical SOP form can be represented on a K-map by simply entering 1's in the cells of the K-map corresponding to each minterm present in the equation.

Representation of Canonical POS form on K-map

Logical equation in canonical POS form can be represented on K-map by entering 0's in the cells of K-map corresponding to each maxterm present in the equation.

Cell Adjacency

The cells in a K-map are arranged so that there is only a single-variable change between adjacent cells. Adjacency is defined by a single variable change. The cells that differ by only one variable are adjacent cells. In the 3-variable map the 010 cell is adjacent to the 000 cell, the 011 cell, and the 110 cell. The 010 cell is not adjacent to the 001 cell, the 111 cell, the 100 cell, or the 101 cell. Physically, each cell is adjacent to the cells that are immediately next to it on any of its four sides. The cells with values that differ by more than one variable and a cell that diagonally touches other cells at any of its corners are not adjacent cells. Also, the cells in the top row are adjacent to the corresponding cells in the bottom row and the cells in the outer left column are adjacent to the corresponding cells in the outer right column. This is called "wrap-around" adjacency.

A Boolean expression must be first in a standard form before a K-map is used. If an expression is not in the standard form, then it must be converted to standard form.

SIMPLIFICATION OF LOGIC FUNCTIONS USING K-MAP

Simplification of logic functions with K-map is based on the principle of combining terms in adjacent cells. It can be verified that in adjacent cells one of the literals is same, whereas the other literal appears in uncomplemented form in one and in the complemented form in the other cell. From this it becomes clear that if the Gray code is used for the identification of cells in K-map, physically adjacent cells differ in only one variable. The simplification of logical function is achieved by grouping adjacent 1's or 0's in groups of 2^i, where $i = 1, 2, \ldots, n$ and n is the number of variables.

The process of simplification involves grouping of minterms and identifying prime-implicants (PI) and essential prime-implicants (EPI).

A prime-implicant is a group of minterms that cannot be combined with any other minterm or groups. An essential prime-implicant is a prime-implicant in which one or more minterms are unique; i.e. it contains atleast one minterm which is not contained in any other prime-implicant.

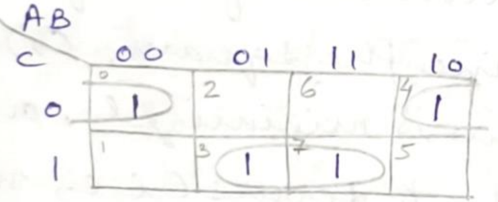
In choosing adjacent squares in a map, we must ensure that (1) all the minterms of the function are covered when we combine the squares, (2) the number of terms in the expression is minimized, and (3) there are no redundant terms (i.e., minterms already covered by other terms). A prime implicant is a product term obtained by combining the maximum possible number of adjacent cells in the map. If a minterm in a square (cell) is covered by only one prime-implicant, that prime-implicant is said to be essential.

The prime implicants of a function can be obtained from the map by combining all possible maximum numbers of cells. This means that a single 1 on a map represents a prime implicant if it is not adjacent to any other 1's. Two adjacent 1's form a prime implica provided that they are not within a group of four adjacent cells. Four adjacent 1's form a prime implicant if they are not within a group of eight adjacent squares, and so on.

Grouping two adjacent ones

If there are two adjacent ones on the map, these can be grouped together and the resulting term will have one less literal than the original two terms.

eg:- Simplify the following K-map



AB / C	00	01	11	10
0	0 1	2	6	4 1
1	1	3 1	7 1	5

The canonical SOP form of equation is given by

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

Combining adjacent cells (0,4) and (3,7)

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

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

1) Identify adjacent ones, then see the values of the variables associated with these cells. Only one variable will be different and it gets eliminated. Other variables will appear in ANDed form in the term, it will be in the uncomplemented form if it is 1 and in the complemented form if it is 0.

2) Determine the term corresponding to each group of adjacent ones. These terms are ORed to get the simplified equation in SOP form.

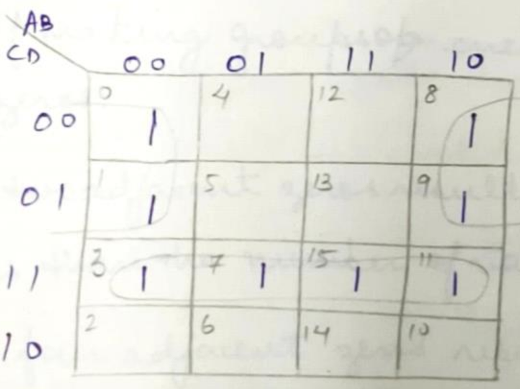
Here, $\bar{B}\bar{C}$ and BC are the two prime-implicants since grouping of the two minterms m_0 and m_4 cannot be combined with anyone of the remaining minterms m_3 and m_7, or the combination of the minterms m_3 and m_7. Similarly, the grouping of m_3 and m_7 is a prime-implicant.

Both the groups are essential prime-implicants, since each one of them contains unique minterms which are not contained in the other group.

Grouping four adjacent ones

Four cells form a group of four adjacent ones if two of the literals associated with minterms/maxterms are not same and the other literals are same.

eg:- Simplify the K-map



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

$$Y = m_0 + m_1 + m_3 + m_7 + m_8 + m_9 + m_{11} + m_{15}$$

$$= (m_0 + m_1 + m_8 + m_9) + (m_3 + m_7 + m_{11} + m_{15})$$

$$m_0 + m_1 + m_8 + m_9 = \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D + A\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}D$$

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

$$= \bar{B}\bar{C}\,[\bar{A}(D+\bar{D}) + A(\bar{D}+D)]$$

$$= \bar{B}\bar{C}\,[(\bar{A}+A)(D+\bar{D})]$$

$$= \bar{B}\bar{C}$$

Similarly, $m_8 + m_9 + m_{11} + m_{15} = CD$

1) Two of the variables appear as $\bar{B}$ and $\bar{C}$ in all four terms.

2) The variable A appears as A in two and as $\bar{A}$ in the other two minterms.

3) The variable D appears as D in two and as $\bar{D}$ in the other two minterms.

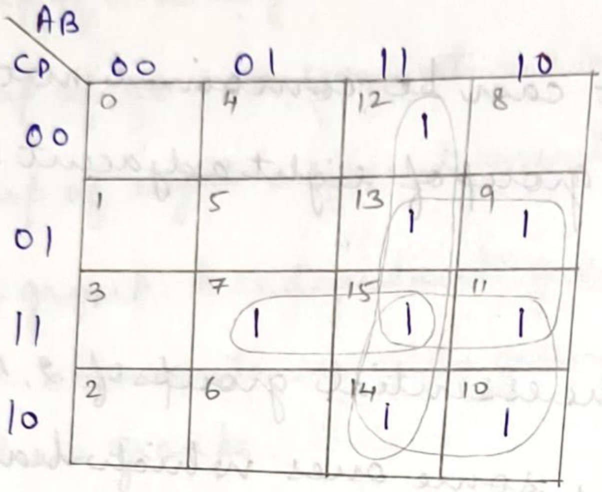
4) The combination of these four mintermes results in one term with two literals which are present in all the four terms.

Similarly, the second term is simplified to CD.

$$Y = \bar{B}\bar{C} + CD$$

2. Find the simplified Boolean equation for the follow

$f(A,B,C,D) = \Sigma m(7,9,10,11,12,13,14,15)$

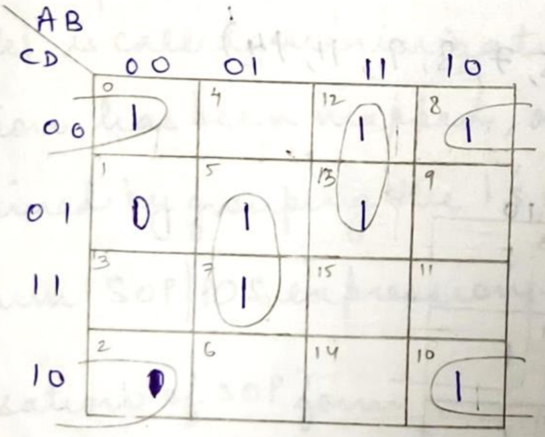


AB CD	00	01	11	10
00	0	4	12 1	8
01	1	5	13 1	9 1
11	3	7 1	15 1	11 1
10	2	6	14 1	10 1

$f(A,B,C,D) = BCD + AC + AD + AB$

3. Simplify the following Boolean expressions using K-map.

$f(A,B,C,D) = \bar{A}\bar{B}\bar{C}\bar{D} + A\bar{C}\bar{D} + \bar{B}C\bar{D} + \bar{A}BCD + B\bar{C}D$



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

$f(A,B,C,D) = \bar{B}\bar{D} + AB\bar{C} + \cancel{B\bar{C}A} + \bar{A}BD$