4.6 Boolean Expressions For Truth Table

All Boolean expressions, regardless of their form, can be converted into either of two standard forms: the sum-of-products form or the product-of-sums form. Standardization makes the evaluation, simplification, and implementation of Boolean expressions much more systematic and easier.

4.6.1 The Sum-of-Products (SOP) Form (Minterm)

This form is sometimes called "minterm". A product term that contains each of the n-variables factors in either complemented or uncomplemented form for output digits "1" only, is called SOP. For example for the truth table below:

Input			Output	
A	B	C	F	
0	0	0	1	$\bar{A}\bar{B}\bar{C}$
0	0	1	0	$\bar{A}\bar{B}C$
0	1	0	1	$\bar{A}B\bar{C}$
0	1	1	1	$\bar{A}BC$
1	0	0	0	$A\bar{B}\bar{C}$
1	0	1	0	$A\bar{B}C$
1	1	0	1	$AB\bar{C}$
1	1	1	1	ABC

The Logical SOP expression for the output digit "1" is written as"

$$F = \bar{A}\bar{B}\bar{C} + \bar{A}B\bar{C} + \bar{A}BC + AB\bar{C} + ABC$$

This function com be put in another form such as:

$$F = \sum 0, 2,3,6,7$$

Since F= 1 in rows 0, 2,3,6,7 only**.**

The second form is called the Canonical Sum of Products (Canonical SOP).

4.6.2 The Product-of-Sum (POS) Form (Maxterm)

A Logical equation can also be expressed as a product of sum (POS) form (sometimes this method is called "Maxterm". This is done by considering the combination for F=0 (output = 0).

So for the above example from the truth table F=0 is in rows 1, 4, 5 hence:

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

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

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

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

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

This is POS form. POS form can be expressed as:

$$F = \prod 1, 4, 5$$

This form is called the Canonical Product of Sum (Canonical POS).

Example**:** Put F in SOP and POS form and simplifying it:

A	B	F
0	0	1
0	1	1
1	0	0
1	1	1

Sol.

$$SOP: \quad F(A,B) = \sum 0,1,3$$

$$= \bar{A}\bar{B} + \bar{A}B + AB$$

$$= \bar{A}(\bar{B} + B) + AB$$

$$= \bar{A} + AB$$

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

POS: $F(A,B) = \prod 2$

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

Example**:** Put in canonical SOP form

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

Sol.

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

101 011 111

$$F(A,B,C) = \sum 3,5,7$$

Example**:** Put in canonical POS form and draw the truth table, then determine canonical SOP and SOP form

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

Sol.

$F(A,B,C) =$ 001 010 111 110

M_1 M_2 M_3 M_4

$$F(A,B,C) = \prod 1,2,6,7$$

A	B	C	F
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

$$F(A,B,C) = \sum 0,3,4,5$$

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

Example: Represent F_1, F_2 in SOP & POS forms then simplified F_1 and F_2 using Boolean algebra.

A	B	C	F_1	F_2
0	0	0	0	1
0	0	1	1	0
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

Sol.

In SOP:

$$F_1(A,B,C) = \sum 1,2,3,5,6,7$$

$$= \bar{A}\bar{B}C + \bar{A}B\bar{C} + \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$$

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

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

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

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

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

$$F_1(A,B,C) = B + C$$

In POS:

$$F(A,B,C) = \prod 0,4$$

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

$$= A\bar{A} + AB + AC + \bar{A}B + BB + BC + C\bar{A} + CB + CC$$

$$= AB + AC + \bar{A}B + B + BC + \bar{A}C + BC + C$$

$$= AB + AC + \bar{A}B + B(1 + C) + \bar{A}C + C(1 + B)$$

$$= AB + AC + \bar{A}B + B + \bar{A}C + C$$

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

$$= B + C + B + C$$

$$F_1(A, B, C) = B + C$$

H.W.: Solution for F_2

4.6.3 Converting SOP to POS and Vice Versa

The binary values of the product terms in a given SOP expression aren't present in the equivalent POS expression. Therefore to convert from standard SOP to standard POS the following steps may be used:

Step 1: Evaluate each product term in the SOP expression that determines the binary numbers representing the product term.

Step 2: Determine all 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 it in POS form.

Note: *A Standard SOP expression is one in which all the variables in the domain appear in each term of the expression. If any variable is missing from any term, we must add these missing variables to that term, by multiplying the term by the variables missing.*

For example, if variable B is missing from the term AC, we must multiply this term AC, by $B + \bar{B}$ to make the expression standard SOP.

$$AC(B + \bar{B})$$

Note*: using a similar procedure explained above (steps 1, 2, and 3) we can convert from standard POS to standard SOP. If there is missing any variable from any term, we must add the missing variable multiplied by its complement to that term.*

For example if variable A is missing from the term $(B + \bar{C})$ we must add $A\bar{A}$

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

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

Example: Put in canonical POS form and draw the truth table, then determine canonical SOP and SOP form

$$F(A,B,C)=B+AC$$

Sol.

1st method

$$F(A,B,C)=B+AC$$

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

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

$$=ABC+AB\bar{C}+\bar{A}BC+\bar{A}B\bar{C}+ABC+A\bar{B}C$$

$$=ABC+AB\bar{C}+\bar{A}BC+\bar{A}B\bar{C}+A\bar{B}C$$

111 110 011 010 101

$$\therefore F(A,B,C)=\sum 2,3,5,6,7$$

$$\therefore F(A,B,C)=\prod 0,1,4$$

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

2nd method:

$$F(A,B,C)=B+AC$$

A	B	C	AC	F= B+AC
0	0	0	0	0
0	0	1	0	0
0	1	0	0	1
0	1	1	0	1
1	0	0	0	0
1	0	1	1	1
1	1	0	0	1
1	1	1	1	1

$\therefore F(A,B,C) = \sum 2,3,5,6,7$

$\therefore F(A,B,C) = \prod 0,1,4$

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

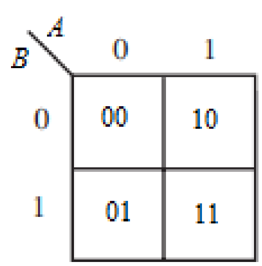
H.W.: Convert the POS form to SOP form and find these canonical:

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

4.7 The Karnaugh Map (K-map)

A K- map provides a systematic method for simplifying Boolean expressions and, if properly used, will produce the simplest SOP or POS expression. As you have seen, the effectiveness of algebraic simplification depends on your familiarity with all the laws, rules, and theorems of Boolean algebra and on your ability to apply them. The K-map is an array of cells in which each cell represents a binary value of the input variables. The cells are arranged in a way so that simplification of a given expression is simply a matter of properly grouping the cells. The K-maps can be used for expressions with two, three, four, and five variables, but we will discuss only 2, 3, and 4 variables. The number of cells in a K-map, as well as the number of rows in a truth table.

For 2 input variables, the number of cells is $2^2 = 4$ cells



For 3 input variables, the number of cells is $2^3 = 8$ cells

$AB \backslash C$	0	1
00		001
01		
11		111
10	100	

And for 4 input variables, the number of cells is $2^4 = 16$ cells

$CD \backslash AB$	00	01	11	10
00				
01			1101	
11				1011
10		0110		

4.7.1 The 2-variebles K - map

1.

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

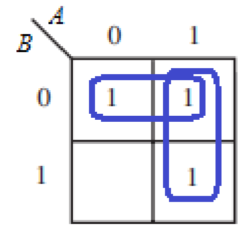
$F(A,B) = B$ POS

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

2.

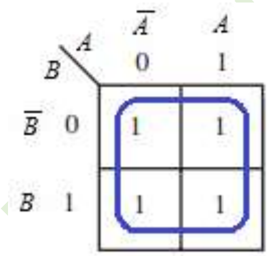
$F(A,B) = \bar{B} + A \quad SOP$

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



3.

$F(A,B) = 1$

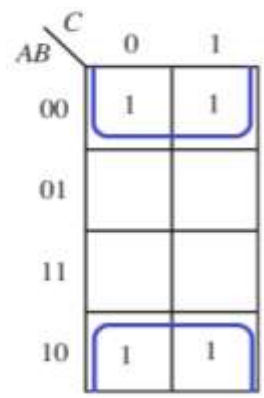


4.7.2 The 3-variebles K - map

1.

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

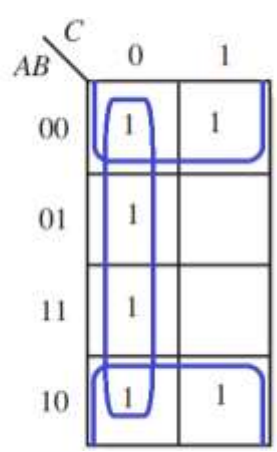
$F(A,B,C) = \bar{B} \quad POS$



2.

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

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

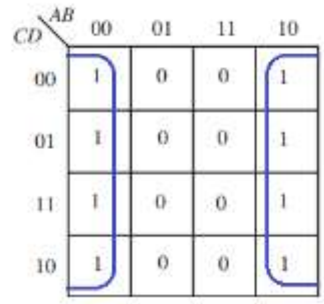


4.7.3 The 4-variebles K – map

1.

$F(A,B) = \bar{B}$ $\quad SOP$

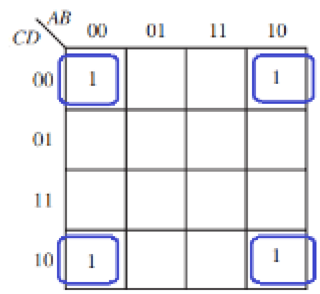
$F(A,B) = B$ $\quad POS$



2.

$F(A,B) = \bar{B}\bar{D}$ $\quad SOP$

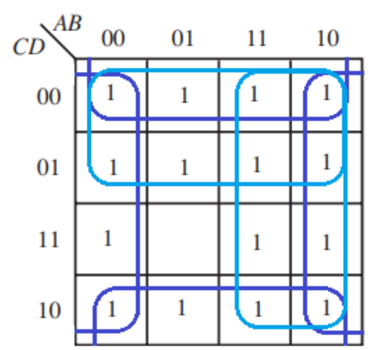
$F(A,B) = (\bar{B})\cdot(\bar{D})$ $\quad POS$



3.

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

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



Note:

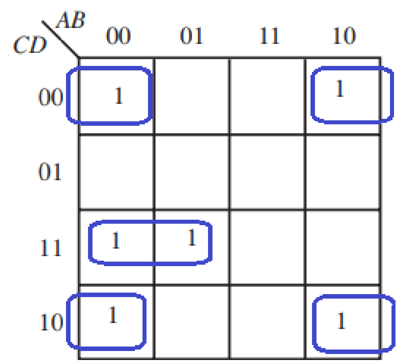
1. *Number of 1's or 0's in one group must be 1, 2, 4, 8, and 16.*
2. *We must take maximum number of 1's or 0's in one group.*

Example: Simplify the following SOP expression on a Karnaugh map:

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

Sol.

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

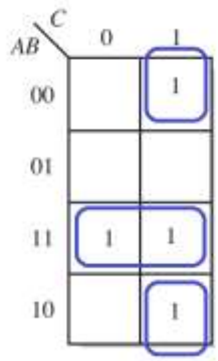


Example: Determine the simply expression by the truth table below using Karnaugh map method.

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

Sol.

$$F = AB + B\bar{C}$$



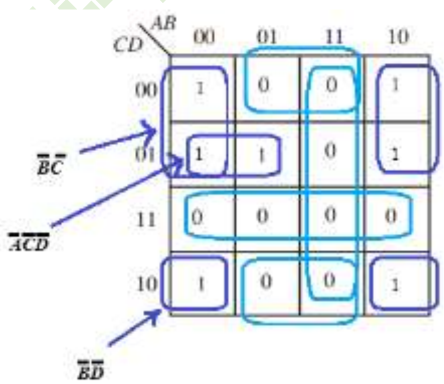
HW: Implement the Logic function specified in the above example.

Example: Simplify the following Boolean function in:

(a) SOP form (b) POS form

$$F(A,B,C,D) = \sum(0,1,2,5,8,9,10)$$

Sol.



(a) The 1's marked in the map represent all minterm of the function. The cells marked with 0's represent the Maxterm not included in the function and therefore the function will be:

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

(b) If the squares marked with 0's are combined we obtain the simplified POS form or the complement of *F:*

$$\overline{F} = AB + CD + B\overline{D}$$

Applying DeMorgan's theorem by taking the complement of each side, we obtain the simplified function in POS form:

$$\overline{\overline{F}} = \overline{AB + CD + B\overline{D}}$$

$$\overline{\overline{F}} = (\overline{AB}) \cdot (\overline{CD}) \cdot (\overline{B\overline{D}})$$

$$\overline{\overline{F}} = (\overline{A} + \overline{B}) \cdot (\overline{C} + \overline{D}) \cdot (\overline{B} + \overline{\overline{D}})$$

$$\overline{\overline{F}} = (\overline{A} + \overline{B}) \cdot (\overline{C} + \overline{D}) \cdot (\overline{B} + D)$$

Note: *To use K-map for simplification a function expressed in POS form, follow these rules:*

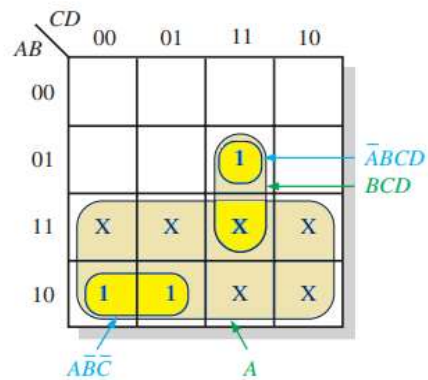
1. *Take the complement of the function.*
2. *From the results write "0" in the Squares of POS form. Or convert the POS to SOP form, then follow the standard rules used to enter the 1's in the cells of K-map.*

4.7.4 Don't Care Conditions

Sometimes a situation arises in which some input variable combinations are not allowed. For example, recall that in the BCD code, there are six invalid combinations: 1010, 1011, 1100, 1101, 1110, and 1111. Since these unallowed states will never occur in an application involving the BCD code, they can be treated as “don’t care” terms with respect to their effect on the output. That is, for these “don’t care” terms either a 1 or a 0 may be assigned to the output; it really does not matter since they will never occur. The “don’t care” terms can be used to advantage on the Karnaugh map. The figure below shows that for each “doesn’t care” term, an X is placed in the cell. When grouping the 1s, the Xs can be treated as 1s to make a larger grouping or as 0s if they cannot be used to advantage. The larger a group, the simpler the resulting term will be.

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

Don't cares



The truth table describes a logic function that has a 1 output only when the BCD code for 7, 8, or 9 is present on the inputs. If the "don't care" are used as 1s, the resulting expression for the function is A + BCD, as indicated in K-map. If the "don't care" is not used as 1s, the resulting expression is ABC + ABCD; so you can see the advantage of using "don't care" terms to get the simplest expression.

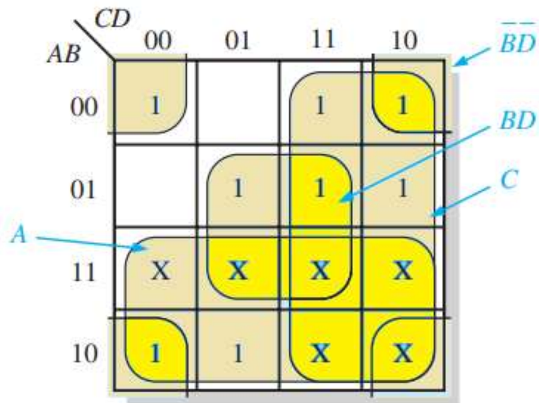
Example: In a 7-segment display, each of the seven segments is activated for various digits. For example, segment-a is activated for the digits 0, 2, 3, 5, 6, 7, 8, and 9, as illustrated in the figure below. Since each digit can be represented by a BCD code, derive an SOP expression for segment-a using the variables ABCD and then minimize the expression using a K - map.

Sol.

The expression for segment-a is:

$$a = \overline{A}\,\overline{B}\,\overline{C}\,\overline{D} + \overline{A}\,\overline{B}C\overline{D} + \overline{A}\,\overline{B}CD + \overline{A}B\overline{C}D + \overline{A}BC\overline{D} + \overline{A}BCD + A\overline{B}\,\overline{C}\,\overline{D} + A\overline{B}\,\overline{C}D$$

Each term in the expression represents one of the digits in which segment-a is used. The Karnaugh map minimization is shown in the figure below. X's (don't care) are entered for those states that do not occur in the BCD code.



From the K - map, the minimized expression for segment-a is:

$$a = A + C + BD + \overline{B}\,\overline{D}$$