Corner group of 4 will give term with 2 literals remain unchanged i.e., $B' D'$.

The last group of 4 will give $A' D'$ because they remain unchanged in the group.

So the expression will be $F(A, B, C, D) = C' + B' D' + A' D'$

2.4.14 Another Method for Minimization of Boolean Expression

There are two basic steps for minimizing functions namely, determining prime implicants and then finding subsets such implicants that cover all product terms of a function.

2.4.15 Definition (Implicant) : An implicant of a function is a product term that is included in the function.

For ~~intence~~ instance : $x y z$ is an implicant of $f(x, y, z) = x y$, for $x y = x y z + x y z'$

2.4.16 Definition (Prime implicant) : A prime implicant of a function is an implicant that is not included in any other implicant of the function.

For instances, xyz is not a prime implicant of $f(x, y, z) = x y + x' y' z'$; because in $x y z$ is contained in xy.

$x y$ is a prime implicant of $f(x, y, z)$ because it is not contained in $x' y' z'$. So, if an implicant is not prime, then it is possible to obtain prime implicant of by removing some literals from it.

2.4.17 Definition : (Essential prime implicant). If a prime impliant includes a min term that is not included in any other prime impliant, then it called an essential prime impliant.

For example : $f(x, y, z) = x y + x' y' z'$ has two prime implant normally $x y$ and $x' y' z'$. prime implicant $x y$ is essential because $x y$ contains $x y z$ and $x y z'$ which are not contained in any other prime implicant (*i.e.* $x' y' z'$).

In another function $f(x, y, z) = x y + x y' + x z'$, the prime impliants are $x y$, $x y'$ and $x z'$. Among them $x z'$ is not essential because $x z' = x y' z' + x y z'$ and $x y' z'$ is in $x y'$ and $x y z'$ is in xy and $x y' z'$ is in $x y'$ and $x y z'$ is in $x y$.

2.4.18 Quine-McCluskey Tabular Method (Quine-Mc Cluskey Minimizing Technique)

In the previous section, we discussed K-map method, which is a convenient method for minimizing Boolean functions upto 5 variables. But, it is difficult to simplify the Boolean functions having more than 5 variables by using this method.

Quine-McCluskey Tabular Method is a tabular method based on the concept of prime implicants. We know that prime implicant is a product (or sum) term, which can't be further reduced by combinding with any other product (or sum) term of the given

Boolean function. This tabular method is useful to get the prime implicants by repeatedly using the following Boolean identity

$$xy + xy' = x(y + y') = x \cdot 1 = x.$$

2.4.19 Procedure of Quine-McCluskey Tabular Method

Follow these steps for simplifying Boolean functions using Quine-McCluskey Tabular Method

Step 1. Arrange the given min terms in an ascending ording and make the groups based on the number of ones present in their binary representation. So, there will be ***atmost "n + 1" groups*** if these are n-Boolean variables in a Boolean function or 'n' bits in the binary equivalent of min terms.

Step 2. Compare the min terms present in ***successive groups***. If there is a change in only one bit position, then make the pair of these two min terms. Placed this symbol '–' in the differed bit position and keep the remaining bit as it is.

Step 3. Repeat step 2 with newly formed terms till we get all ***prime implicants.***

Step 4. Formulate the ***prime implicant*** table. It consist of set of rows and columns. Prime implicants can be placed in row wise and min terms can be placed in column wise. Place '1' in the cells corresponding to the min terms that are covered in each prime implicant.

Step 5. Find the essential prime implicants by observing each column. If the min term is covered only by one prime implicant then it is ***essential prime implicant.*** Those essential prime implicants will be part of the simplified Boolean function.

Step 6. Reduce the prime implicant table by removing the row of each essential prime implicant and the column corresponding to the min terms that are covered in that essential prime implicant. Repeat step 5 for reduced prime implicant table. Stop this process when all min terms of given Boolean functions are over.

Example 4 : Simplity the following Boolean function,

f (W, X, Y, Z) = Σ m (2, 6, 8, 9, 10, 11, 14, 15) using Quine-McCluskey tabular method.

Sol. The given Boolean function is in sum of min terms form. It is having 4 variables W, X, Y and Z. The given min terms are 2, 6, 8, 9, 10, 11, 14 and 15. The ascending order of these min terms based on the number of ones present in their binary equivalent is 2, 8, 6, 9, 10, 11, 14 and 15. The following table shows these min terms and their equivalent binary representation

Group Name	*Min terms*	W	X	Y	Z
GA1	2	0	0	1	0
	8	1	0	0	0
GA2	6	0	1	1	0
	9	1	0	0	1
	10	1	0	1	0
GA3	11	1	0	1	1
	14	1	1	1	0
GA4	15	1	1	1	1

The given min terms are arranged into 4 groups based on the numbers of ones present in their binary equivalents. The following table shows the possible **merging of min terms** from the adjacent group

Group Name	*Min terms*	W	X	Y	Z
GB1	2, 6	0	—	1	0
	2, 10	—	0	1	0
	8, 9	1	0	0	—
	8, 10	1	0	—	0
GB2	6, 14	—	1	1	0
	9, 11	1	0	—	1
	10, 11	1	0	1	—
	10, 14	1	—	1	0
GB3	11, 15	1	—	1	1
	14, 15	1	1	1	—

The min terms, which are differed in only one bit position from adjacent group are merged. That differed bit is represented by the symbol '—'. In this case, there are three groups and each group contains combinations of two min terms. The following table shows the possible **merging of min terms pair** from adjacent groups.

Group Name	*Min terms*	W	X	Y	Z
GB1	2, 6, 10, 14	—	—	1	0
	2, 10, 6, 14	—	—	1	0
	8, 9, 10, 11	1	0	—	—
	8, 10, 9, 11	1	0	—	—
GB2	10, 11, 14, 15	1	—	1	—
	10, 14, 11, 15	1	—	1	—

The successive groups of min terms pair, which are different in only one-bit position are merged. That differed bit is represented with this symbol, '—'. In this case, there are two groups and each group contains combinations of four min terms. Here, these combinations of 4 min terms are available in two rows. So, we can remove the repeated rows. The reduced table after removing the redundant rows is shown below

Group Name	*Min terms*	W	X	Y	Z
GC1	2, 6, 10, 14	—	—	1	0
	8, 9, 10, 11	1	0	—	—
GC2	10, 11, 14, 15	1	—	1	—

Further merging of the combinations of min terms from adjacent groups is not possible, since they are different in more than one-bit position. There are three rows in the above table. So, each row will give one prime implicant. Therefore, the prime implicants are YZ′, WX′ and WY.

The prime implicant table is shown below :

Min terms/ Prime implicants	2	6	8	9	10	11	14	15
YZ′	1	1			1		1	
WX′			1	1	1	1		
WY					1	1	1	1

The prime implicants are placed in row wise and min terms are placed in column wise. 1's are placed in the common cell of prime implicant rows and the corresponding min term columns. The min terms 2 and 6 are covered only by one prime implicant YZ′. So, it is an essential prime impliant. This will be part of simplified Boolean function. Now, we remove this prime implicant row and the corresponding min term columns. The reduced prime implicant table is shown below :

Min terms/ Prime implicants	8	9	11	15
WX′	1	1	1	
WY			1	1

The min terms 8 and 9 are covered only by one prime implicant WX′. So, it is an essential prime implicant. This will be part of simplified Boolean function. Now, remove this prime implicant row and the corresponding min term column. The reduced prime implicant table is shown below :

Min terms/ Prime implicants	15
WY	1

The min term 15 is covered only by one prime inplicant WY. So, it is an **essential prime implicant**. This will be part of simplified Boolean function.

In this example problem, we got three prime implicants and all the three are essential. Therefore, the **Simplified Boolean Function** is

$$f(W, X, Y, Z) = Y Z' + W X' + W Y$$