Karnaugh Maps Simplification

Rab Nawaz Khan Jadoon

Lecturer

COMSATS Lahore

Pakistan

Department of Computer Science

DCS

COMSATS Institute of
Information Technology

Digital Logic and Computer Design

K-Map Simplification

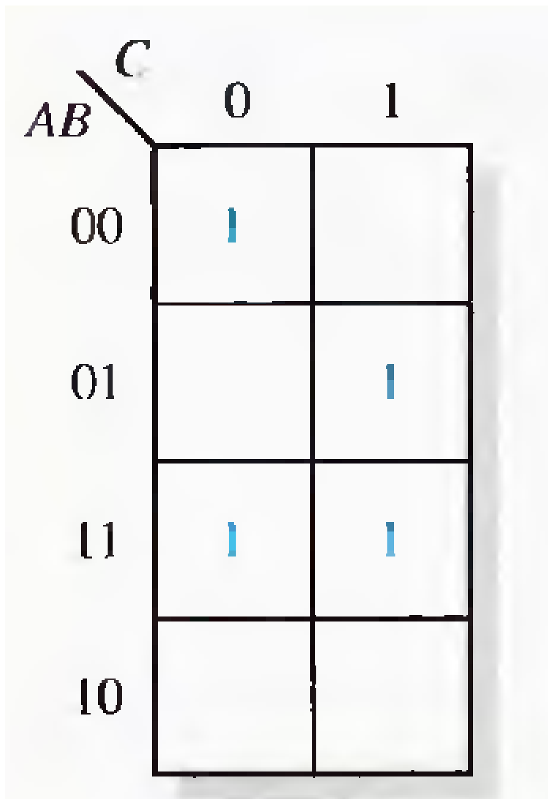
- After an SOP expression has been mapped, a minimum SOP expression is obtained by grouping the Is and determining the minimum SOP expression from the map.
- You can group Is on the Karnaugh map according to the following rules by enclosing those adjacent cells containing 1s.
- The goal is to maximize the size of the groups and to minimize the number of groups.

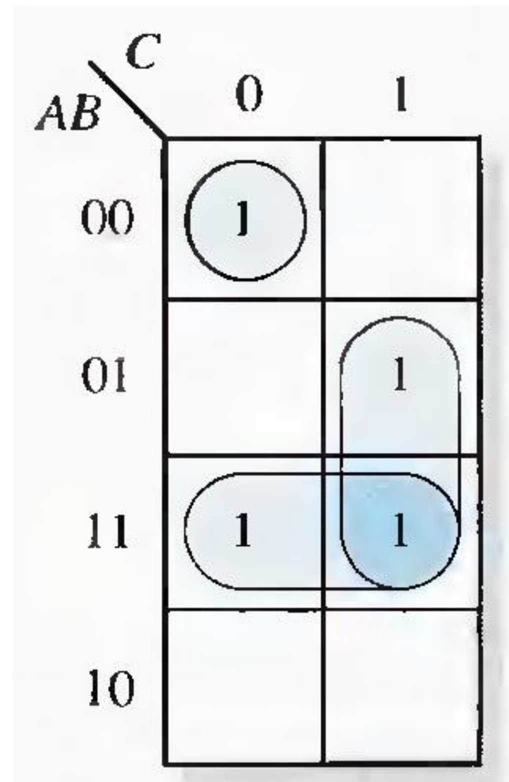
K-Maps

- Karnaugh map is an array of cells in which each cell represents a binary value of the input variables.
- The cells are managed in a way so that simplification of a given expression is simply a matter of properly grouping the cells.
- Karnaugh maps can be used for expressions with two, three, four. and five variables, but we will discuss only 3-variable and 4-variable situations to illustrate the principles.

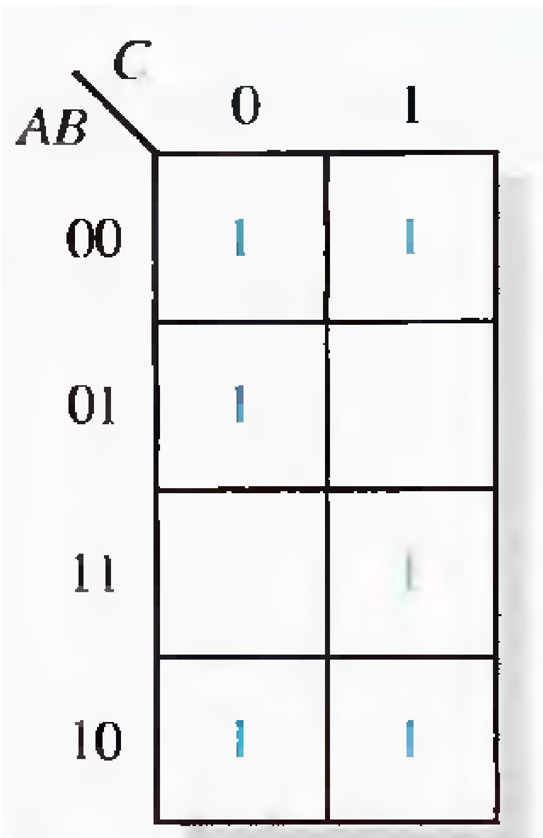
- Steps for Grouping
  - A group must contain either 1, 2, 4, 8, or 16 cells, which are all powers of two. In the case of a 3-variable map, $2^3 = 8$ cells is the maximum group.
  - Each cell in a group must be adjacent to one or more cells in that same group. but all cells in the group do not have to be adjacent to each other.
  - Always include the largest possible number of 1's in a group in accordance with rule 1.
  - Each 1 on the map must be included in at least one group. The Is already in a group can be included in another group as long as the overlapping groups include noncommon 1's.

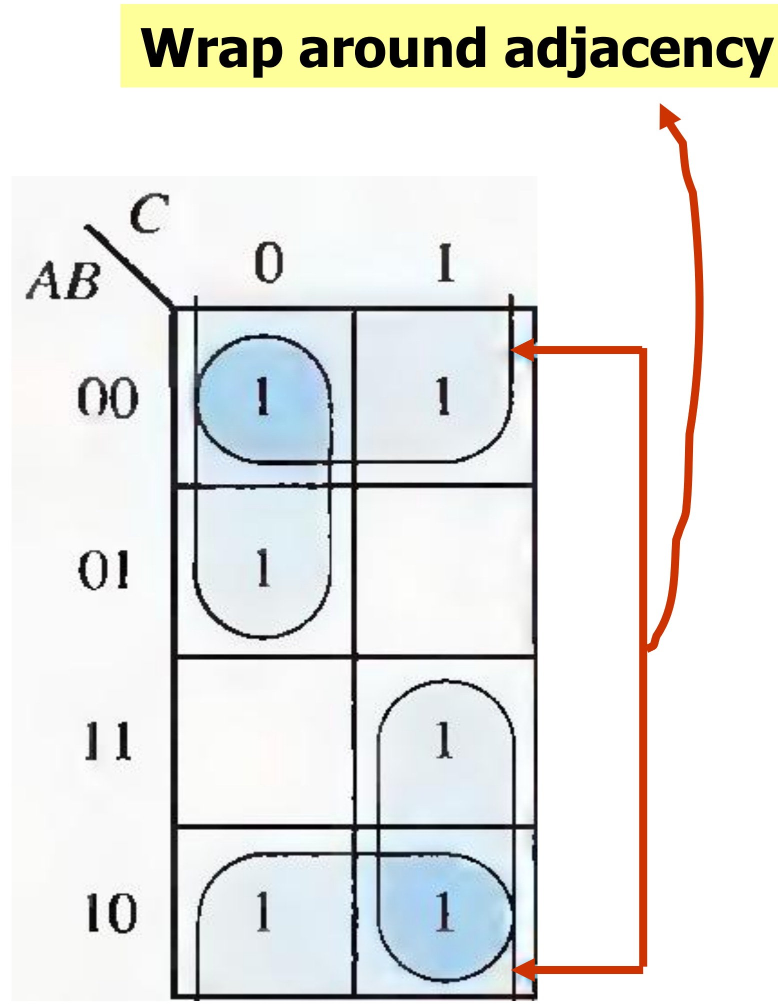
For Example



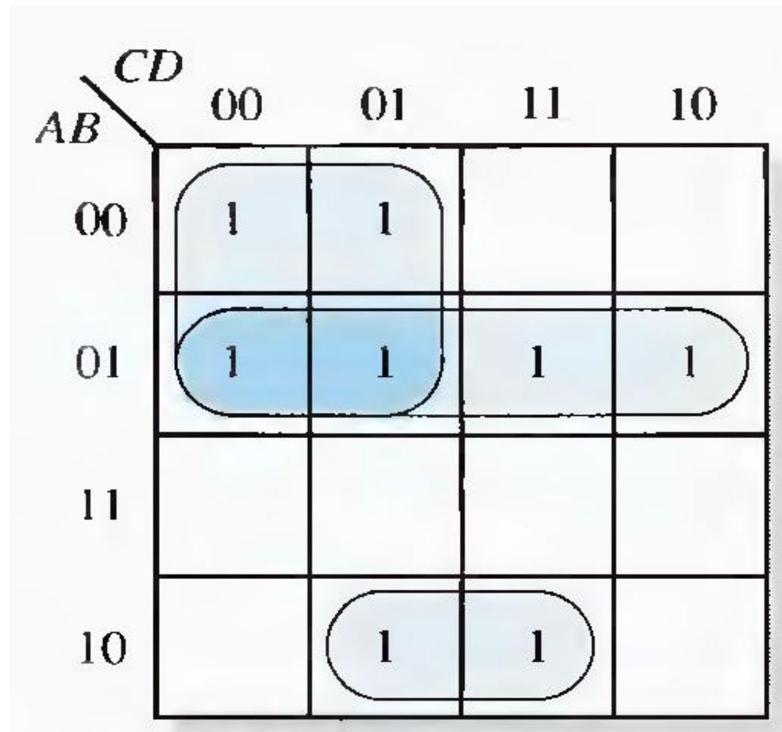


Example

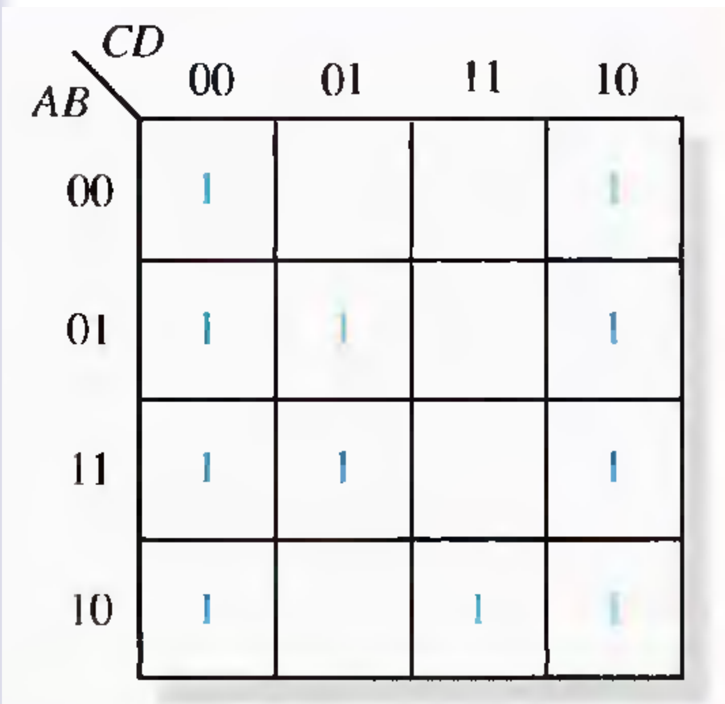

C
AB
0
1
00
01
11
10
1
1
1
1
1
1

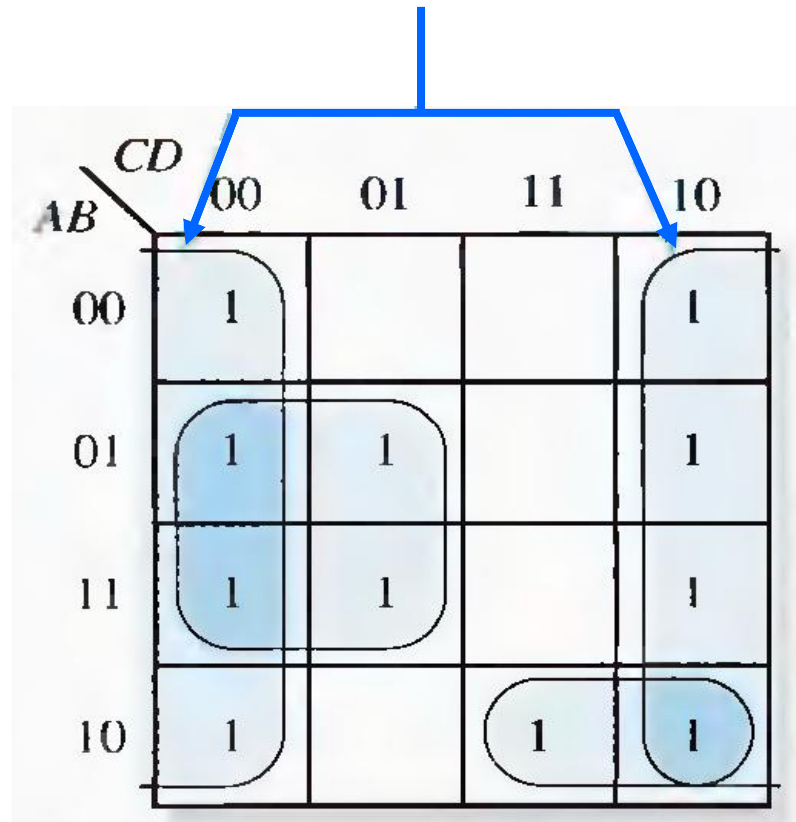

Wrap around adjacency
C
AB
0
1
00
01
11
10
1
1
1
1
1
1

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



Wrap around adjacency





- Determining the Minimum SOP Expression from the Map.
  - When all the 1's representing the standard product terms in an expression are properly mapped and grouped, the process of determining the resulting minimum SOP expression begins.
  - Rules
    - Group the cells that have 1's. Each group of cells containing I s creates one product term composed of all variables that occur in either form within the group.
    - Variables that occur both uncomplemented and complemented within the group are eliminated. These are called contradictory variables.

The 4-Variable Karnaugh Map

- Determining the minimum term for each group
  - For a 3-veriable map.

    (1) A 1-cell group yields a 3-variable product term
    (2) A 2-cell group yields a 2-variable product term
    (3) A 4-cell group yields a 1-variable term
    (4) An 8-cell group yields a value of 1 for the expression

  - For a 4-veriable map

    (1) A 1-cell group yields a 4-variable product term
    (2) A 2-cell group yields a 3-variable product term
    (3) A 4-cell group yields a 2-variable product term
    (4) An 8-cell group yields a 1-variable term
    (5) A 16-cell group yields a value of 1 for the expression

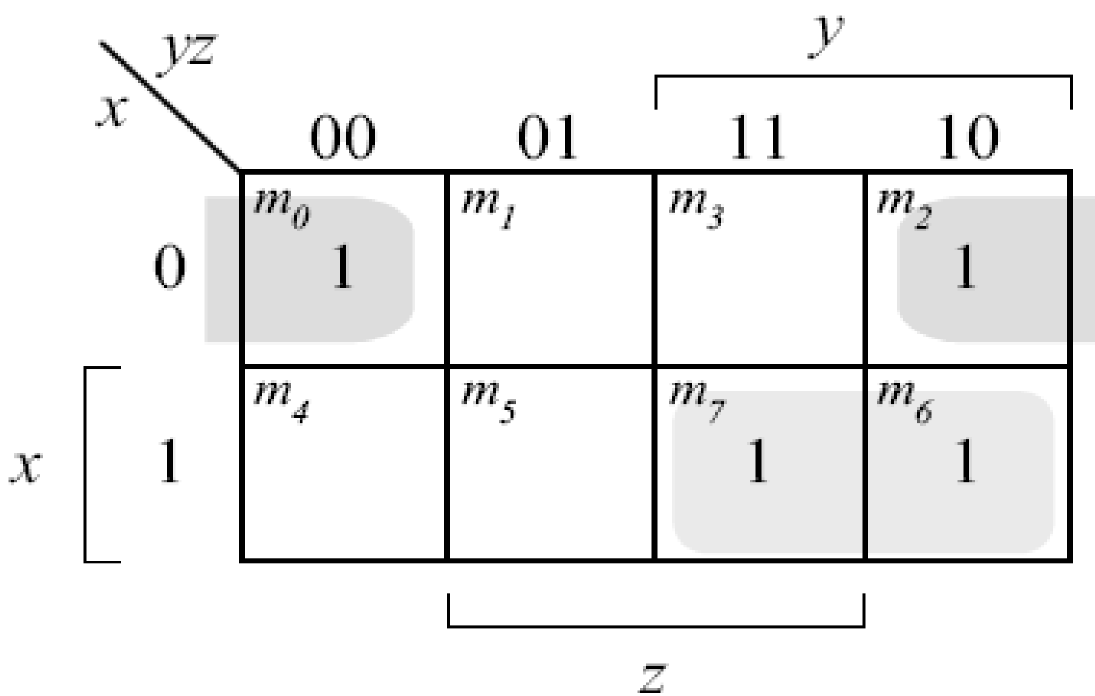
Note: When all the minimum product terms are derived from the Karnaugh map, they are summed to form the minimum SOP expression.

Related Example

- Minimize the following SOP expression,
  - $F(x, y, z) = \Sigma(0,2,6,7)$ → 1
  - $F(x, y, z) = \Sigma(0,2,3,4,6)$ → 2
  - $F(x, y, z) = \Sigma(0,1,2,3,7)$ → 3
  - $F(x, y, z) = \Sigma(3,5,6,7)$ → 4
  - $F(x, y, z) = \Sigma(0,1,5,7)$ → 5
  - $F(x, y, z) = \Sigma(0,1,6,7)$ → 6
  - $F(x, y, z) = \Sigma(1,2,3,6,7)$ → 7

Solution

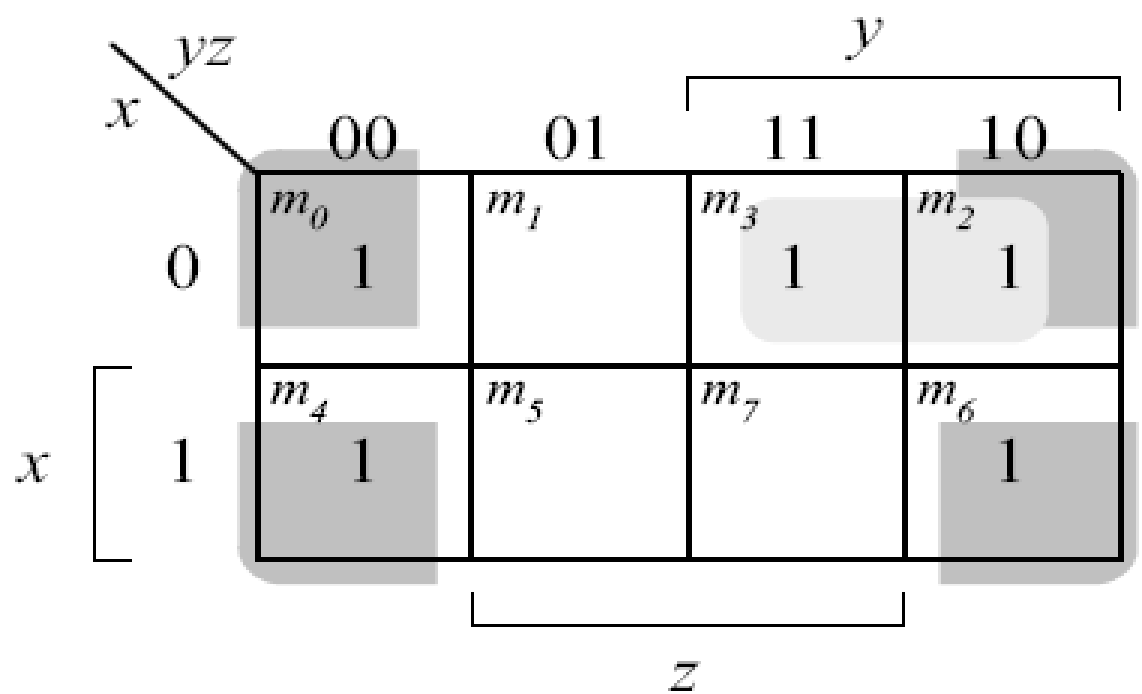
- F(x, y, z) = Σ(0,2,6,7) → 1



F = x y + x′ z'

- **Solution**

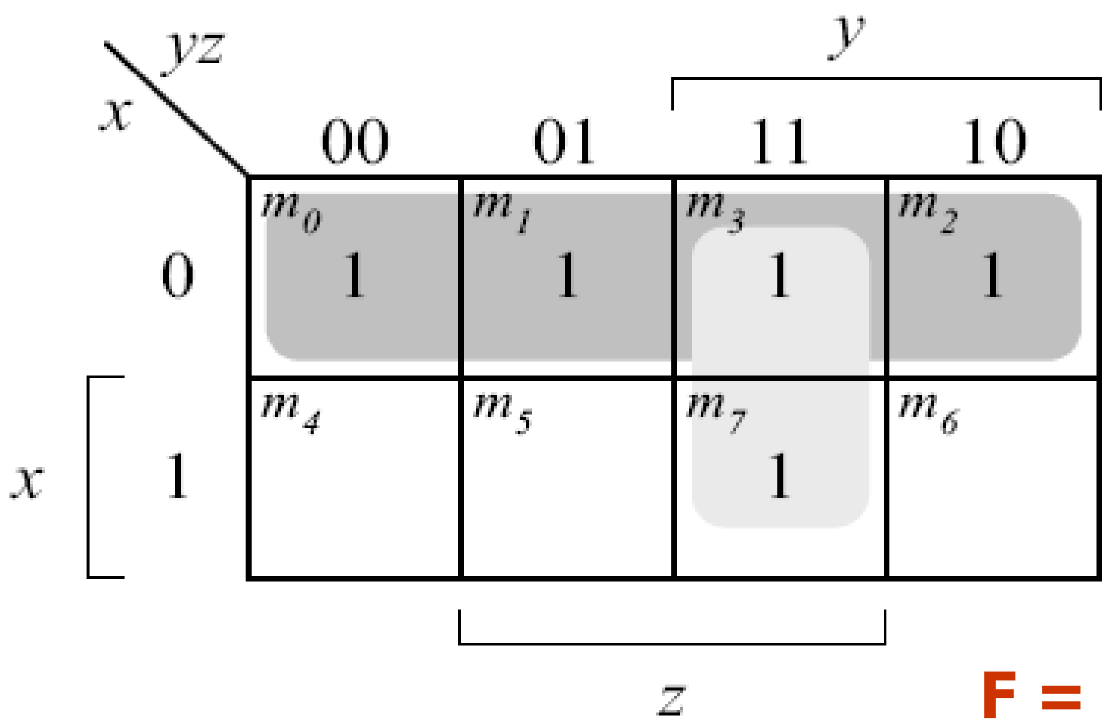
F(x, y, z) = Σ(0,2,3,4,6) ➔ 2



F = z' + x'y

- **Solution**

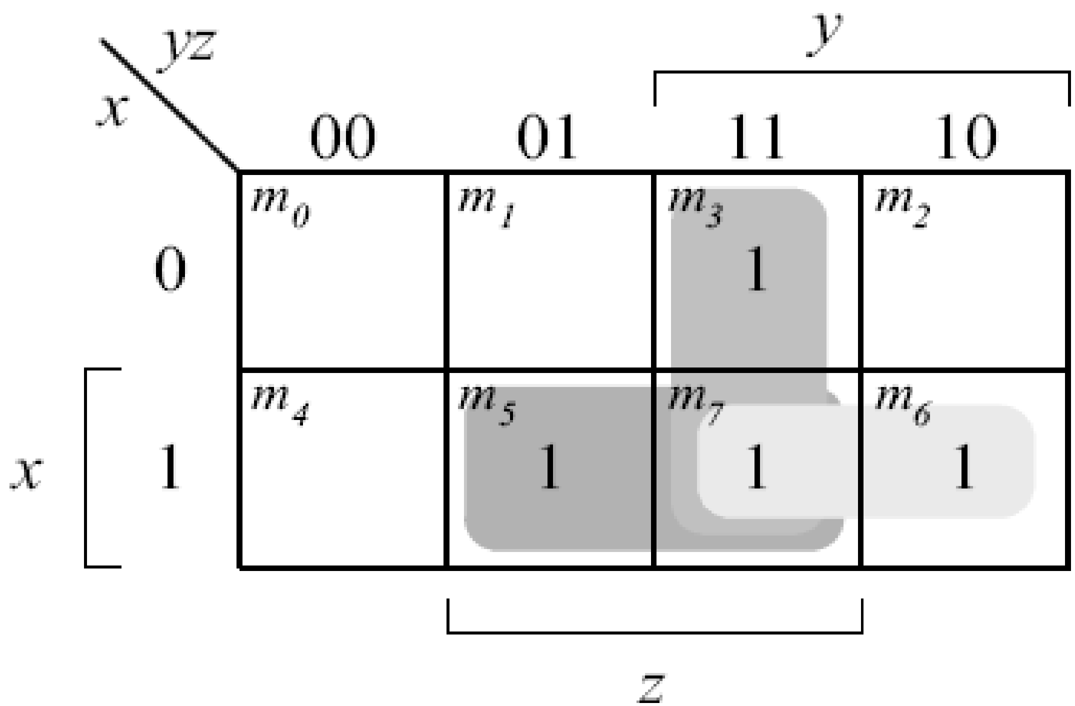
F(x, y, z) = Σ(0,1,2,3,7) → 3



F = x' + y z

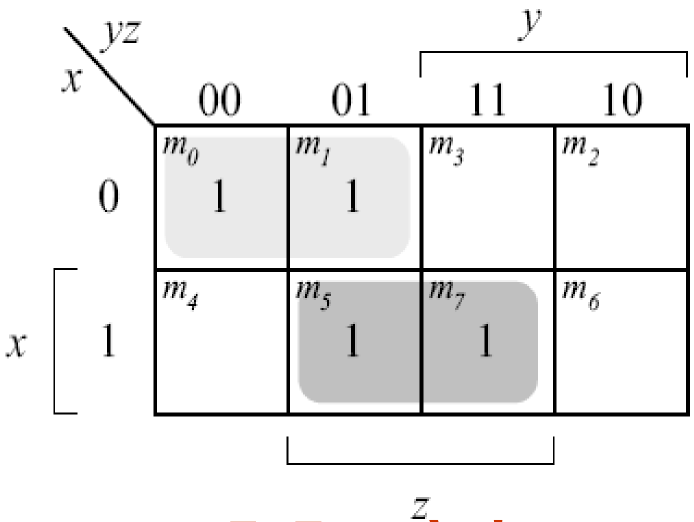
- **Solution**

F(x, y, z) = Σ(3,5,6,7) → 4

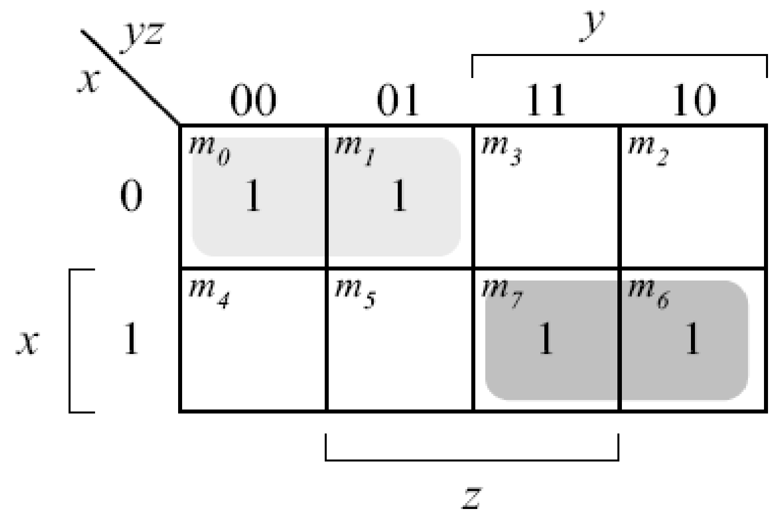


F = xy + xz + yz

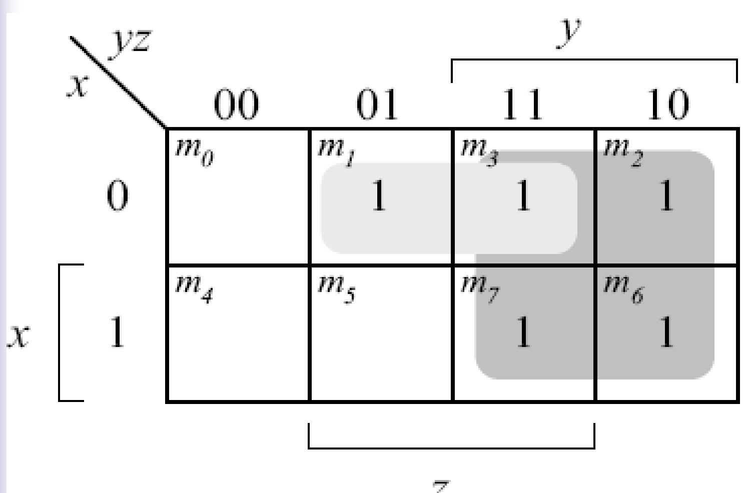
Solution of 5, 6, 7



5. F = x‵ y' + x z

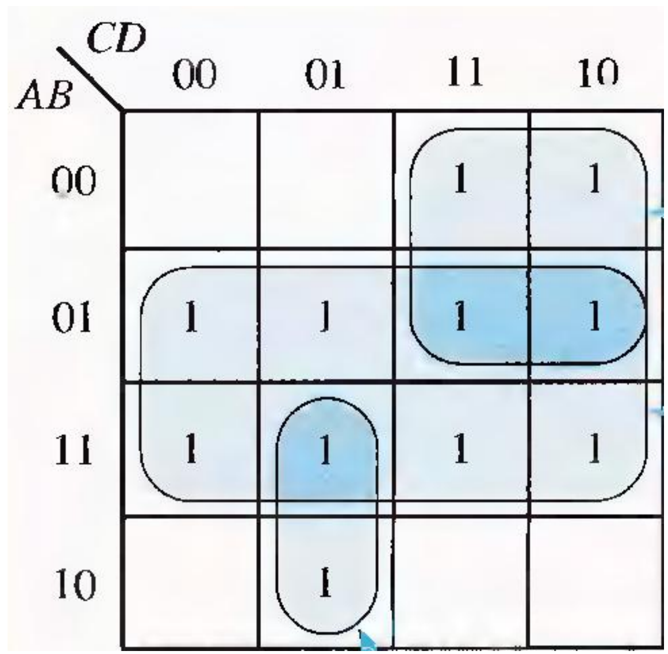


6. F = x‵ y' + x y

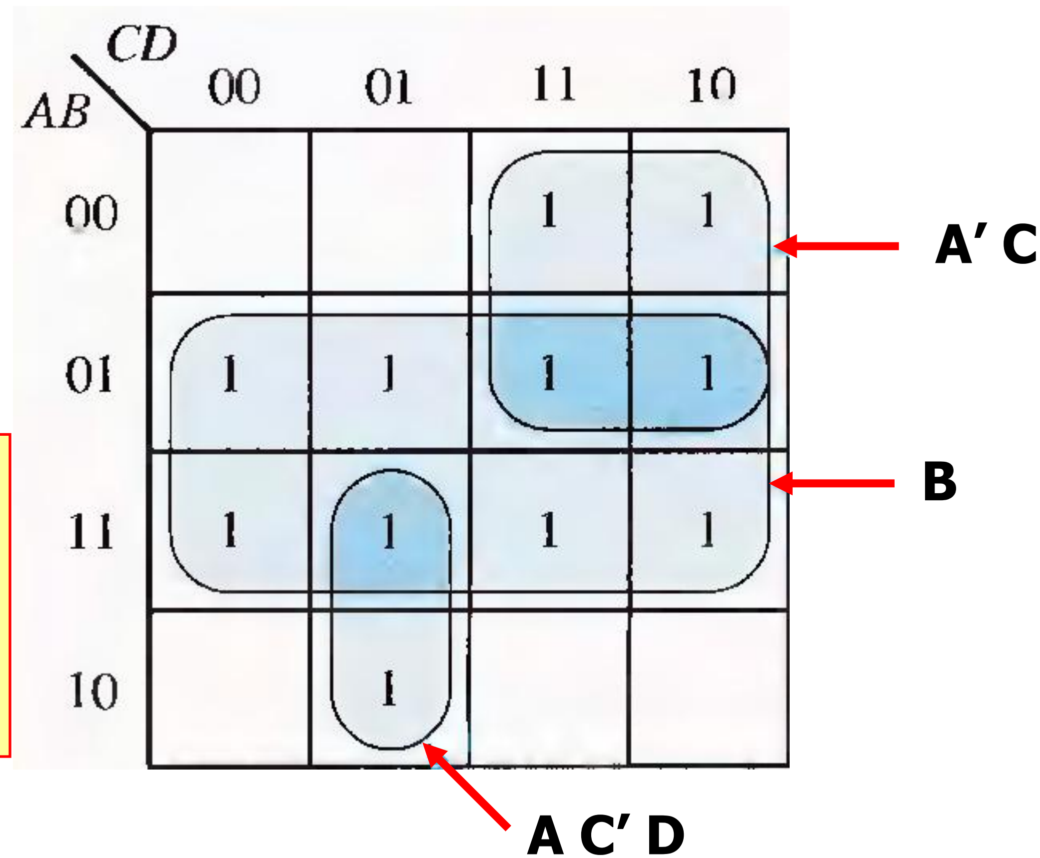


7. F = y + x‵ z

- **Example**
  - Determine the product terms for the Karnaugh map given and write the resulting minimum SOP expression?



- **Solution**

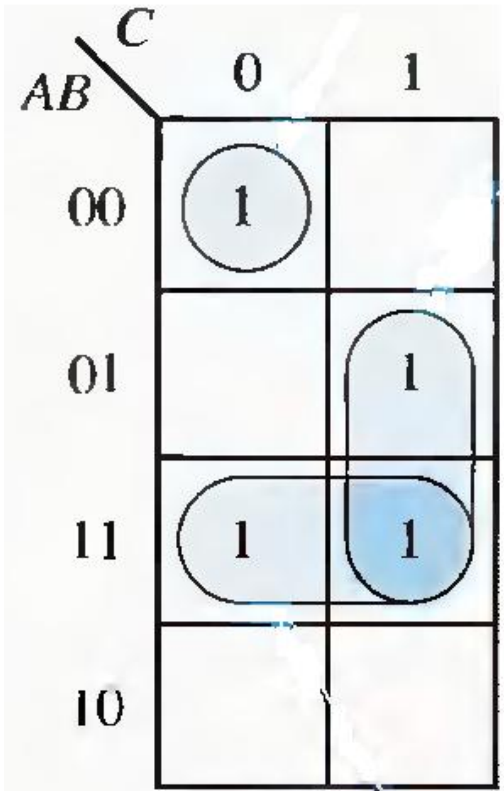


The resulting minimum SOP expression is,
B + A′ C + AC′ D
the sum of these product terms:

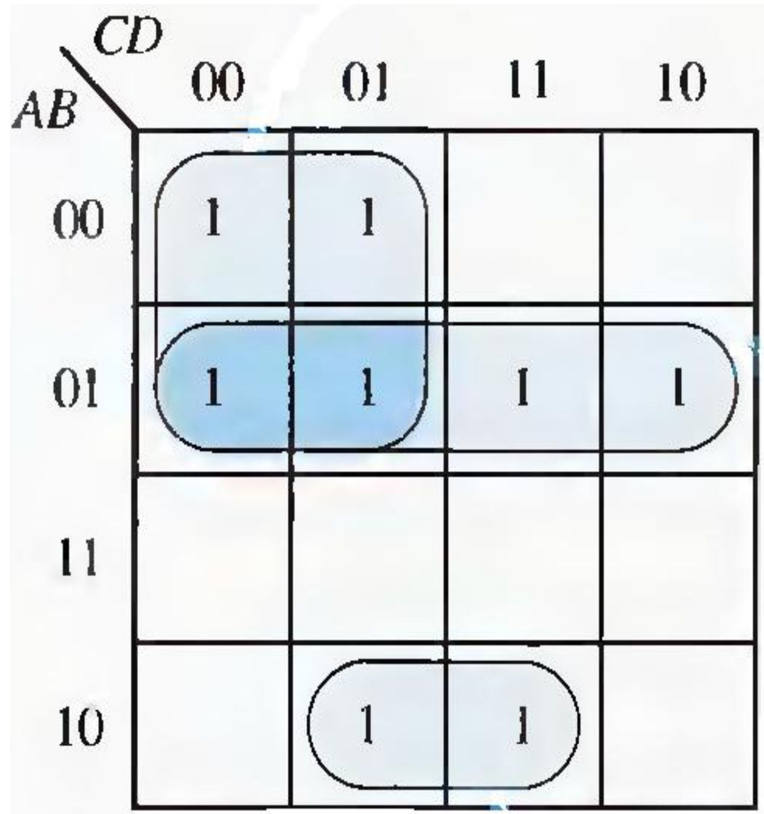
Self Assessment

Problem: For the Karnaugh map on the previous slide, add a 1 in the lower right cell (1010) and determine the resulting SOP expression.

Example 1

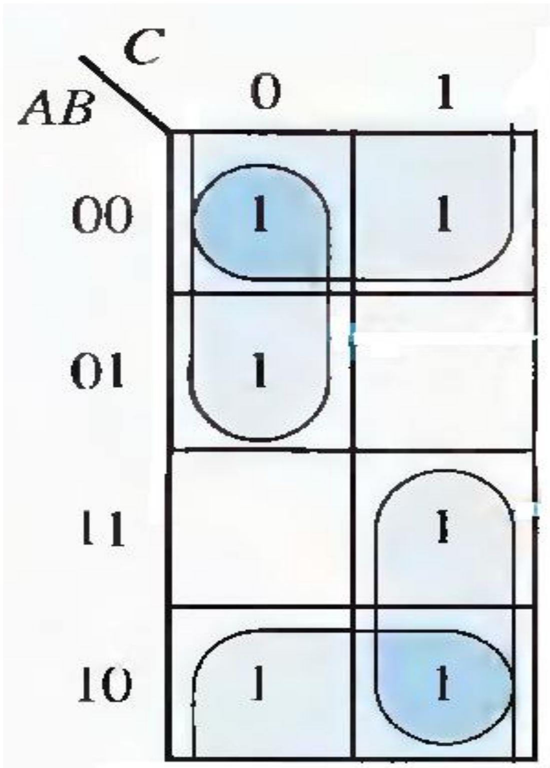


a

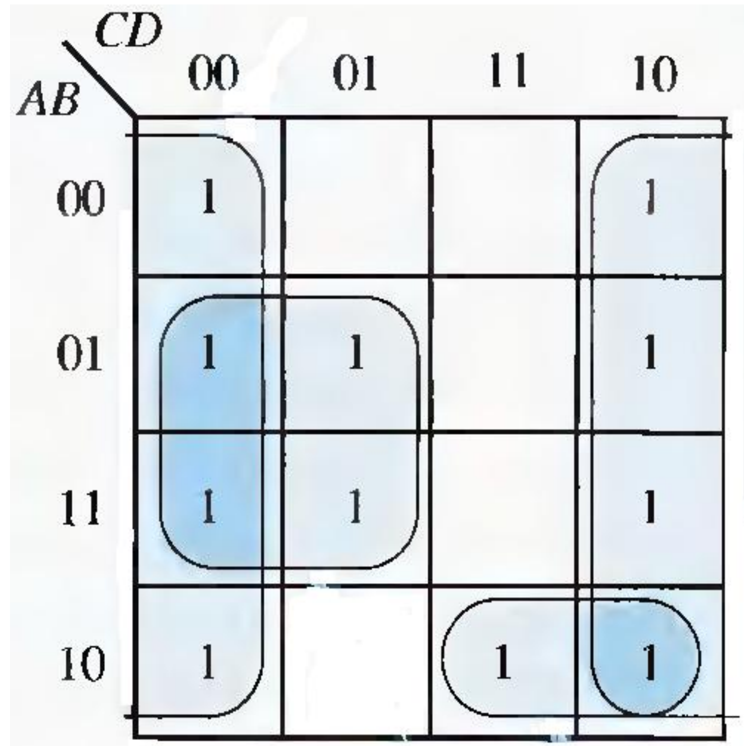


b

Example 2


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

c


CD
AB
00
01
11
10
00
01
11
10
1
1
1
1
1
1
1
1
1
1
1
1

d

Solution

- The resultant SOP expression against a, b, c and d are,
  a. AB+BC+A′ B′ C′
  b. A′ B + A′ C′ + AB′ D
  c. B′ + A′ C′ + AC
  d. D′ + AB′ C + BC′

Class Assignment

- Using k-map simplify the following expression,

  $F(A,B,C,D) = \Sigma(0,2,3,4,6,8,10,11,12,14)$

Just do it in 3min. Only.

The resulting minimum SOP expression is

$D' + B'C$

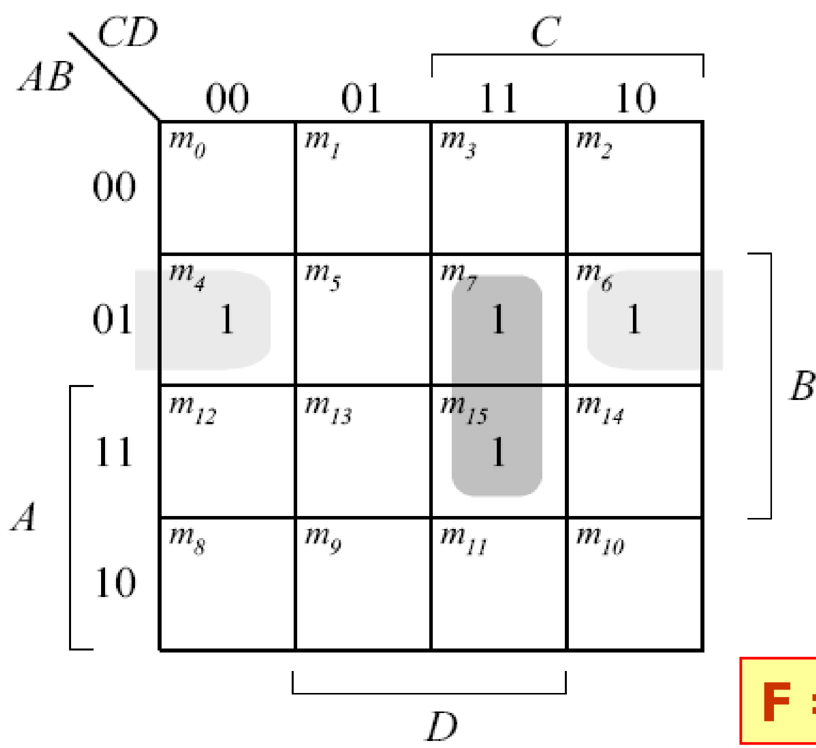
Examples

- Find the minimized SOP expression against the following,

1. $F(A,B,C,D) = \Sigma(4,6,7,15)$
2. $F(A,B,C,D) = \Sigma(3,7,11,13,14,15)$
3. $F(A,B,C,D) = \Sigma(0,1,5,8,9)$
4. $F(A,B,C,D) = \Sigma(1,4,5,6,12,14,15)$

- **Solution**

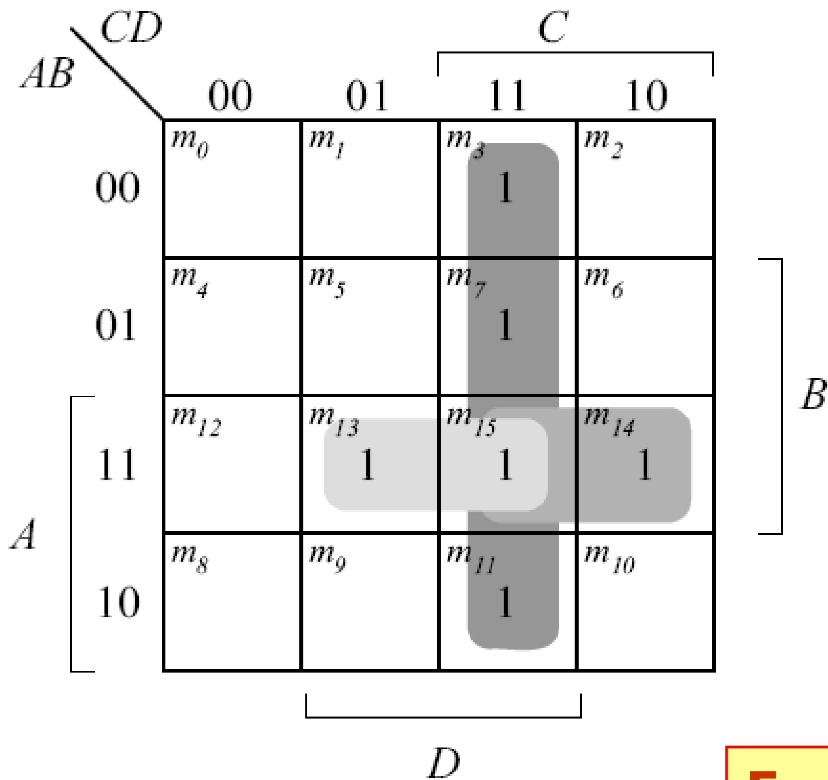
1. F (A,B,C,D) = Σ(4,6,7,15)



F = BCD + A' BD'

- **Solution**

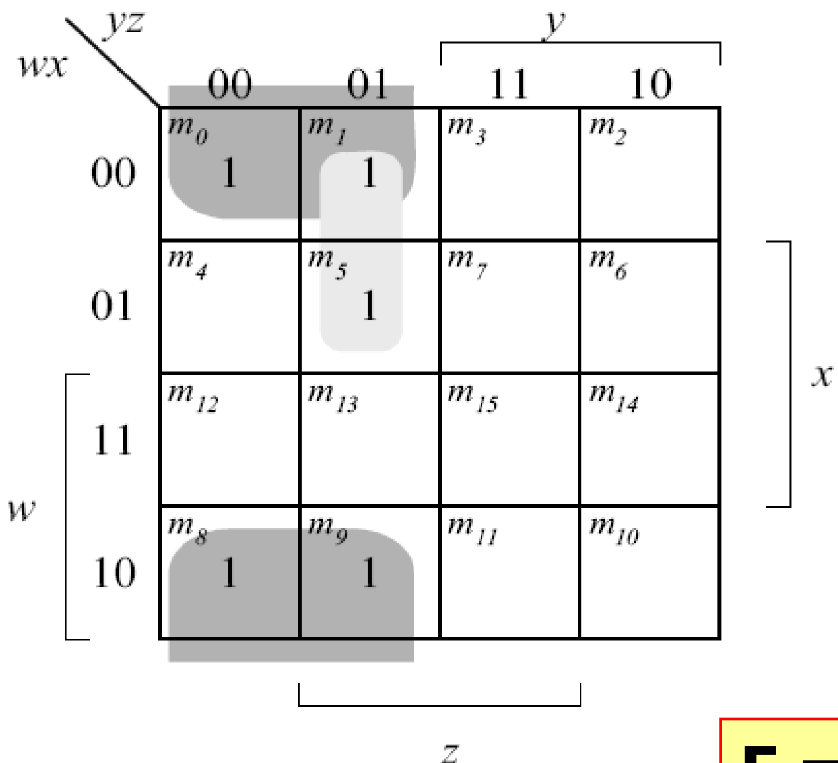
2. $F(A,B,C,D) = \Sigma(3,7,11,13,14,15)$



F =CD + ABD + ABC

- **Solution**

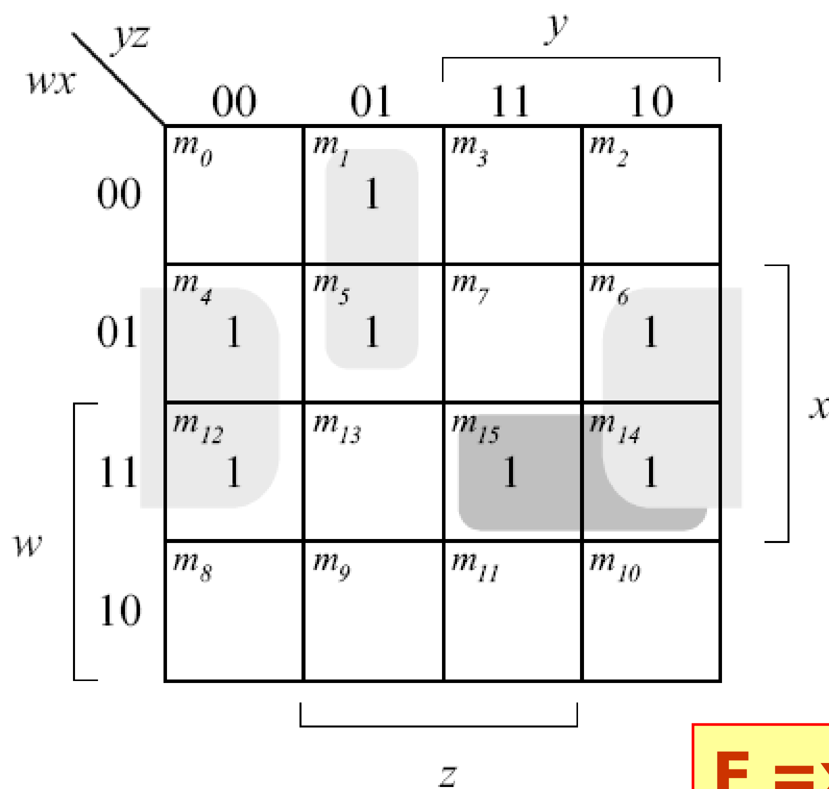
3. F (A,B,C,D) = Σ(0,1,5,8,9)



F = x'y' + w'y'z

- **Solution**

4. F (A,B,C,D) = Σ(1,4,5,6,12,14,15)



F =xz' + w'y'z+ wxy

Assignment# 2

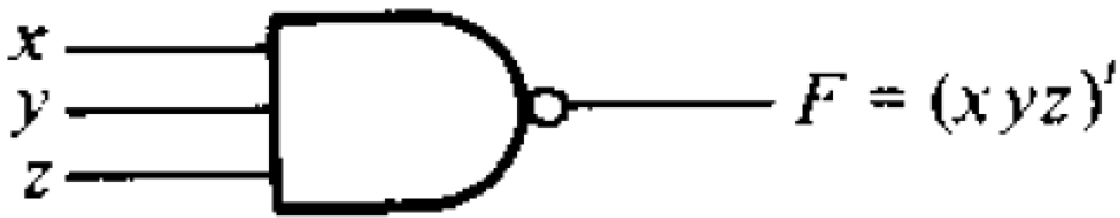
- Simplify the following SOP Expression,

1. F (A,B,C,D) = Σ(1,2,3,5,7,9,10,11,13,15)
2. F (A,B,C,D) = Σ(1,2,3,5,9,10,11,12,13)
3. F (A,B,C,D) = Σ(0,2,3,5,7,8,10,11,14,15)
4. F (A,B,C,D) = Σ(2,3,7,10,11,12,13,14,15)
5. F (A,B,C,D) = Σ (1, 3, 5, 9, 12, 13, 14)
6. F (A,B,C,D) = Σ (0, 2, 4,5,6,7, 8, 10, 13, 15)

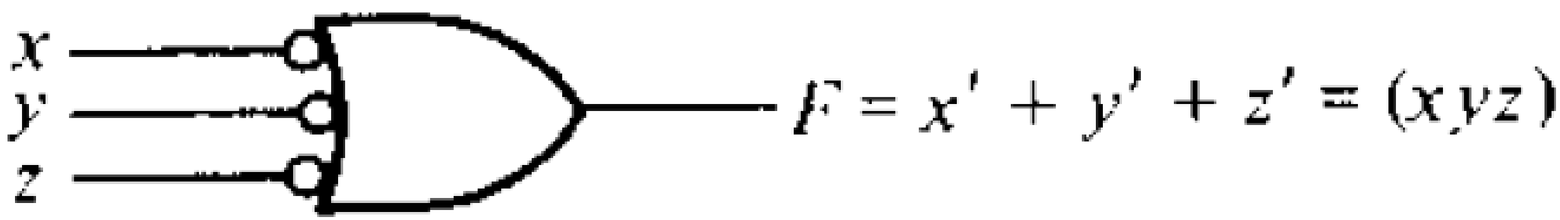
Due date: Next Incoming Class

No Copy/Past Material should be,

NAND and NOR Implementation

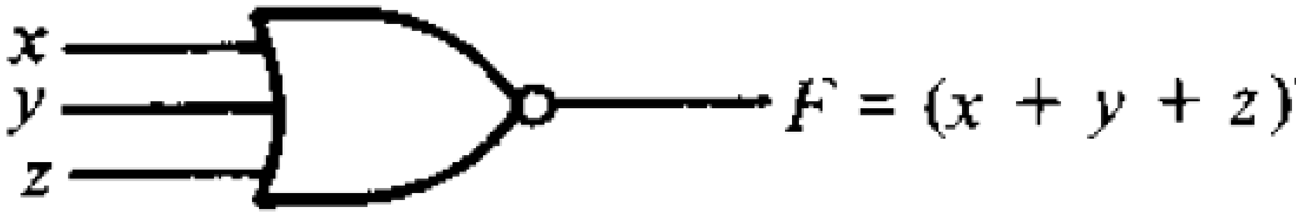


AND-invert

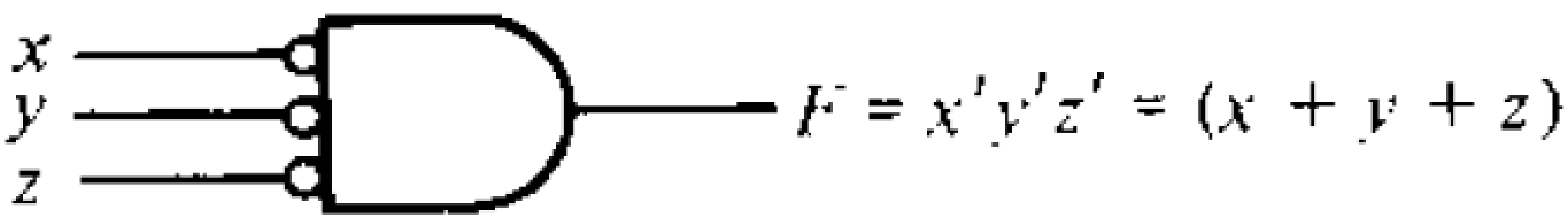


Invert-OR

NOR Equivalent



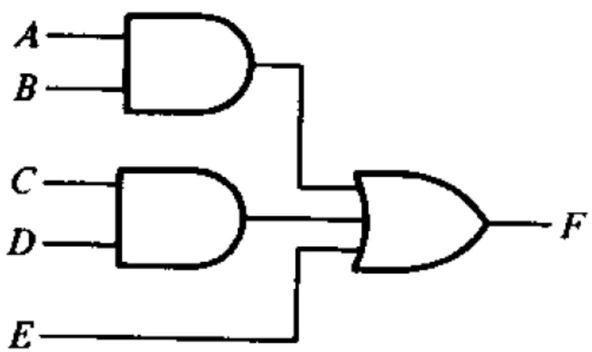
OR-invert



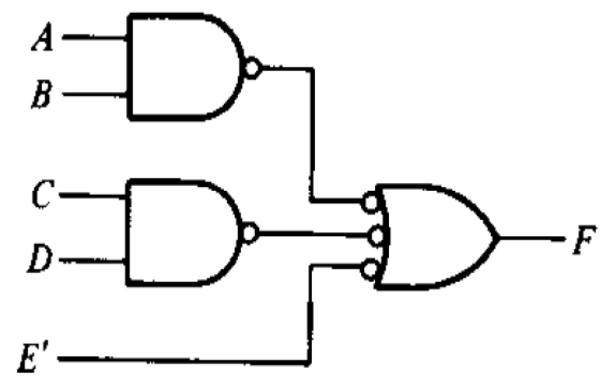
Invert-AND

NAND Implementation

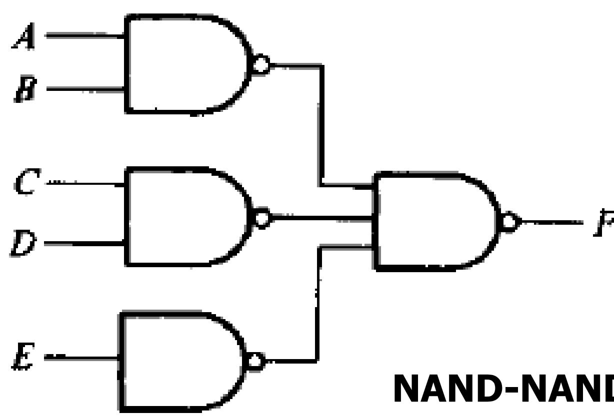
F=AB + CD + E



AND-OR



NAND-NAND



NAND-NAND

NAND implementation

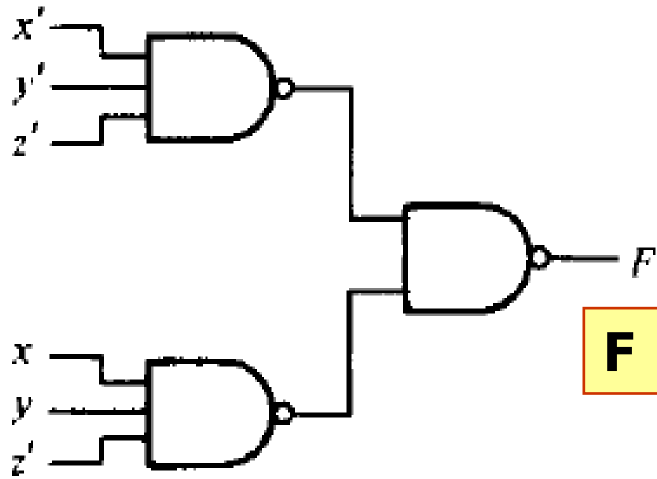
- Implement the following function using NAND,

  $F(x, y, z) = \Sigma(0,6)$

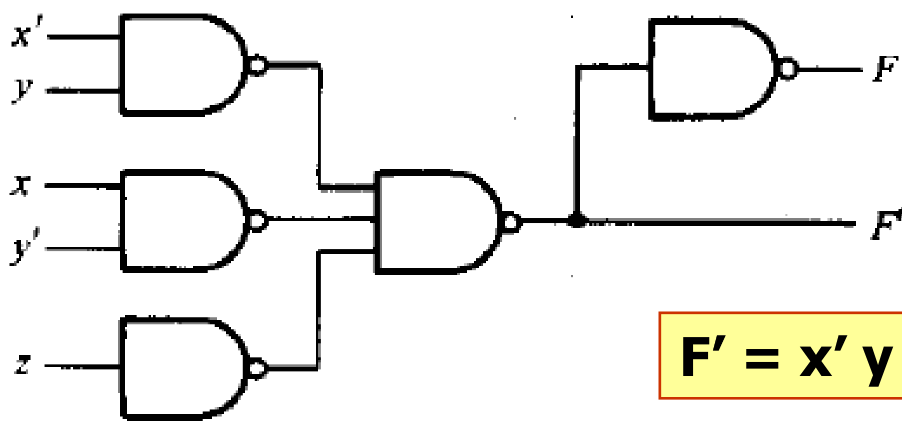
x \ yz	00	01	11	10
0	1	0	0	0
1	0	0	0	1

$F = x'y'z' + xyz'$
$F' = x'y + xy' + z$

Cont…



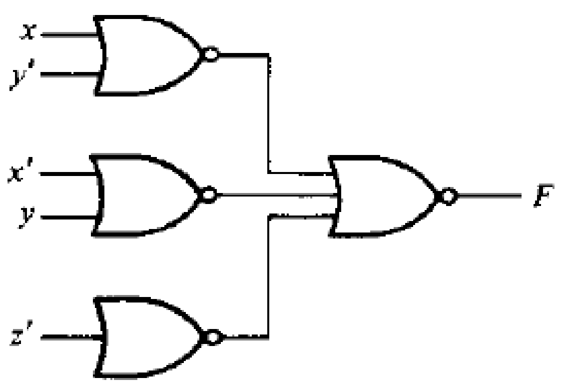
F = x′ y′ z′ + xyz′



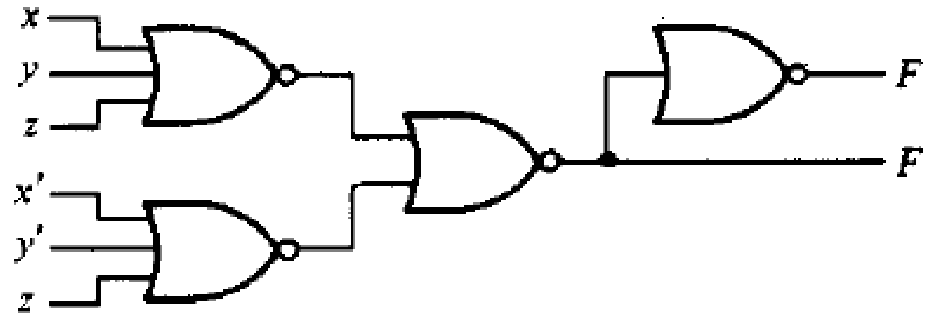
F′ = x′ y + x y′ +z

NOR Implementation

- Taking the function and expand it using demorgan theorem we, get,
  - **F′ = x′ y + x y′ +z**
  - **F=(x+y′)(x′+y)z′**



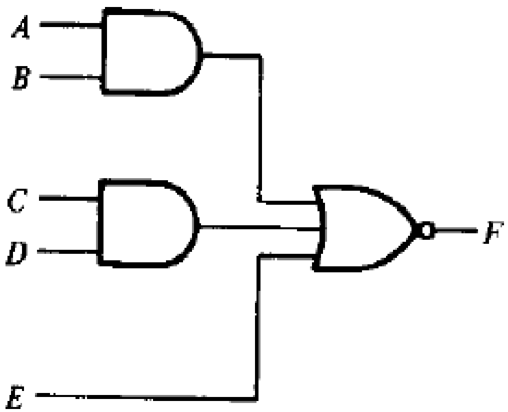
F=(x+y′)(x′+y)z′



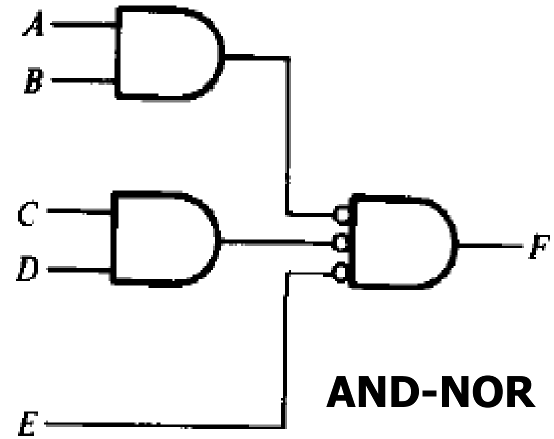
F = x′ y′ z′ + xyz′
F′ = (x+y+z)(x′+y′+z)

AND-OR-INVERT Implementation

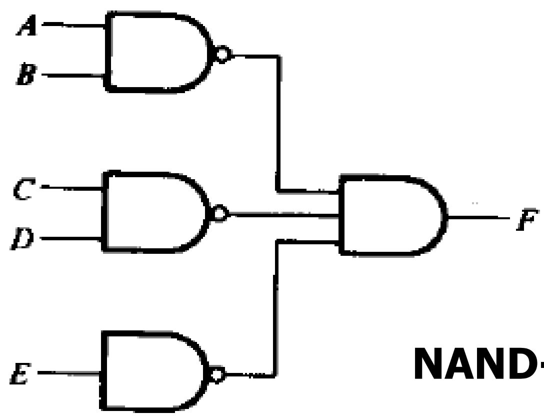
- Let the function is
  - F=(AB+CD+E)'



AND-NOR



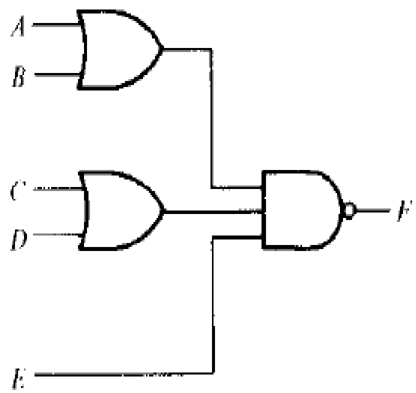
AND-NOR



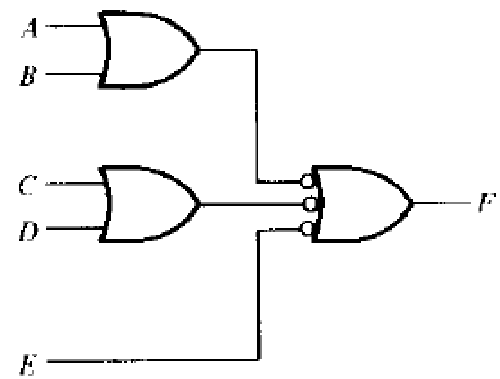
NAND-AND

OR-AND-Invert

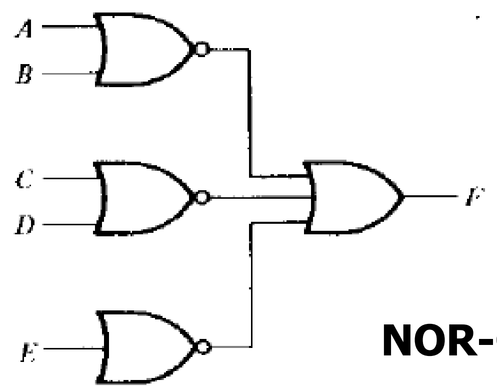
- Let the function is
  - F=[(A+B)(C+D)E]'



OR-NAND



OR-NAND



NOR-OR

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.
  - The "don't care" terms can be used to advantage on the Karnaugh map.

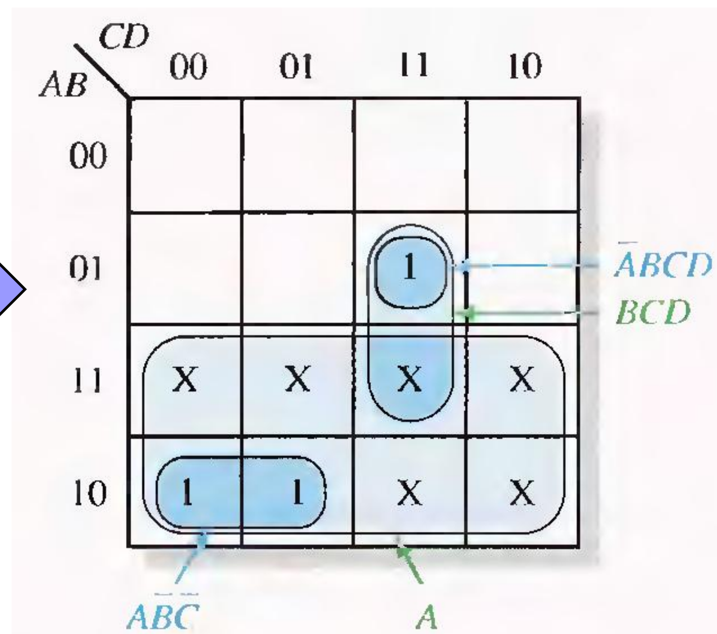
- For each "don't care" term, an X is placed in the cell.
  - When grouping the 1's, the X's can be treated as I s to make a larger grouping or as O's if they cannot be used to advantage.
  - The larger a group, the simpler the resulting term will be.

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 care conditions

- The truth table in Figure below describes a logic function that has a I output only when the BCD code for 7,8, or 9 is present on the inputs. if the "don't cares" are used as Is, the resulting expression for the function is A + BCD.

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



Without "don't cares"
Y = AB'C' + A'BCD
With ""don't cares"
Y = A + BCD

Related Problem

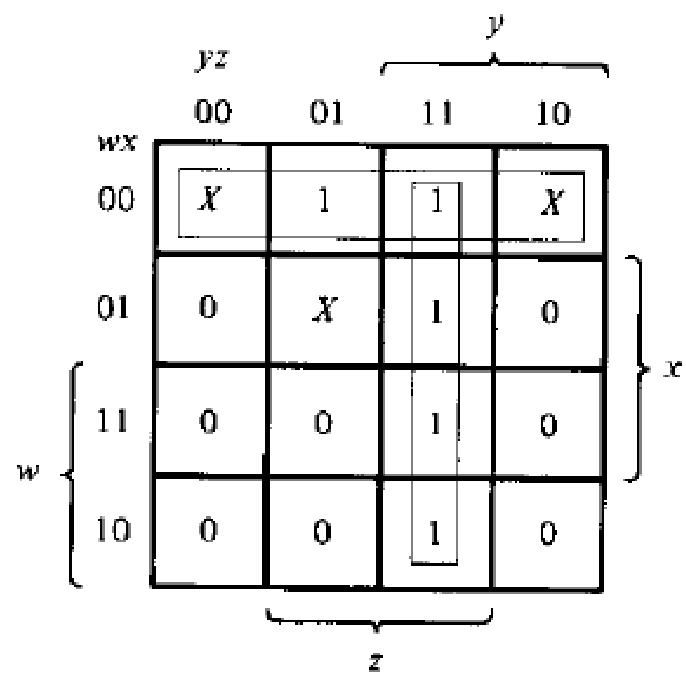
- Simplify the Boolean functions

F(w,x,y,z)=Σ(1.3.7.11.15)

Don't care conditions are,

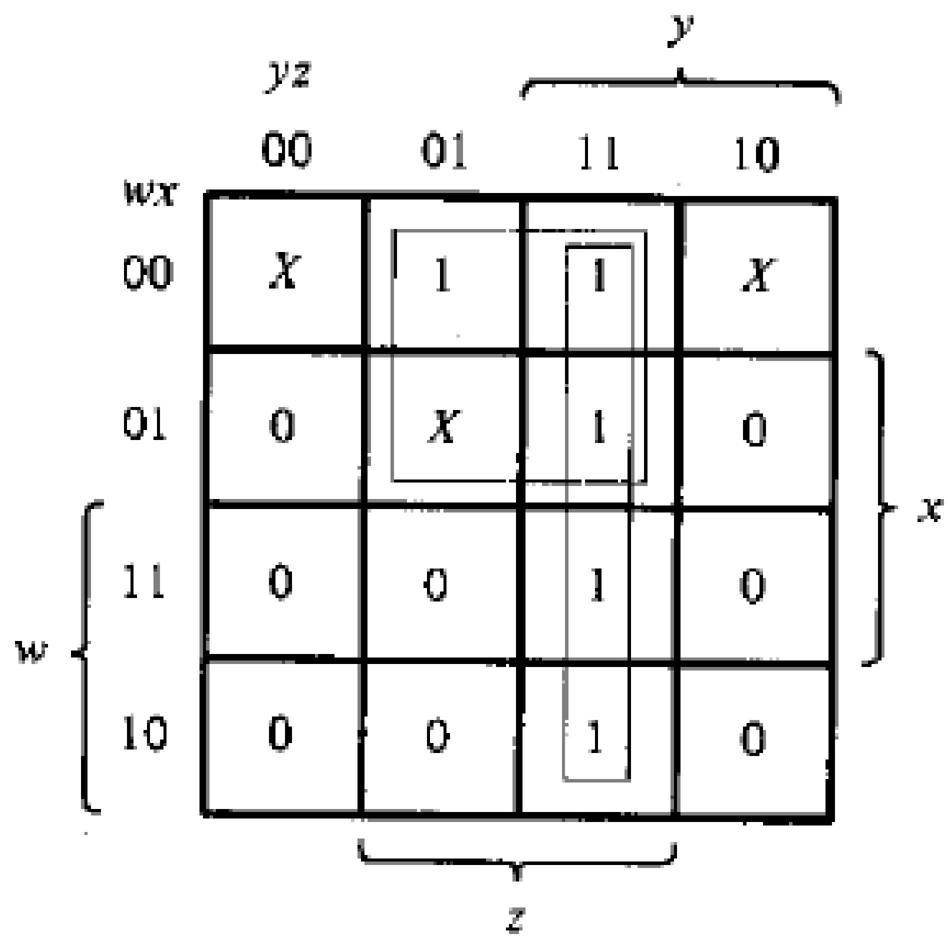
d(w,x,y,z)= Σ(0.2.5)

Solution



F=yz+w'x'

Example

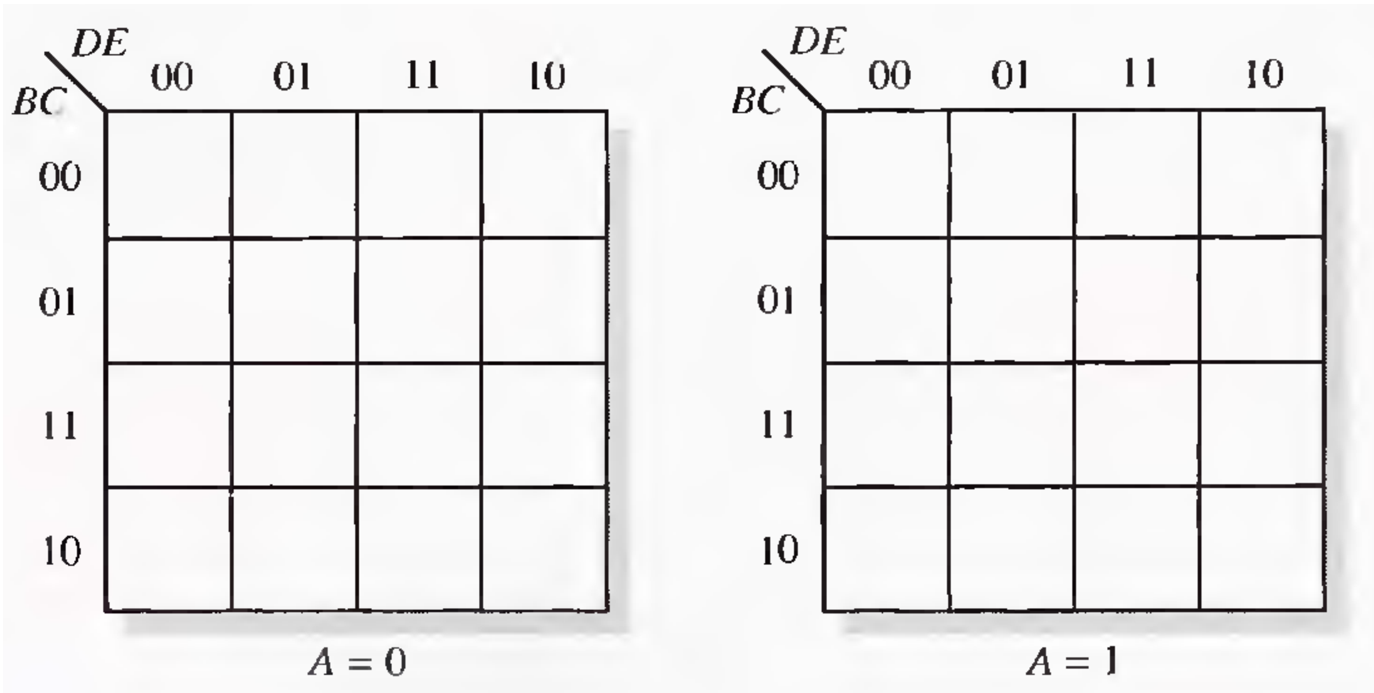


(b) $F = yz + w'z$

Five veriable K map

- Boolean functions with five variables can be simplified using a 32-cell Karnaugh map.
- Actually, two 4-variable maps (16 cells each) are used to construct a S-variable map.
- A Karnaugh map for five variables (ABCDE) can be constructed using two 4-variable maps with which you are already familiar.
- Each map contains 16 cells with all combinations of variables 8, C, D, and E.
- One map is for A = 0 and the other is for A = I, as shown on the next slide.

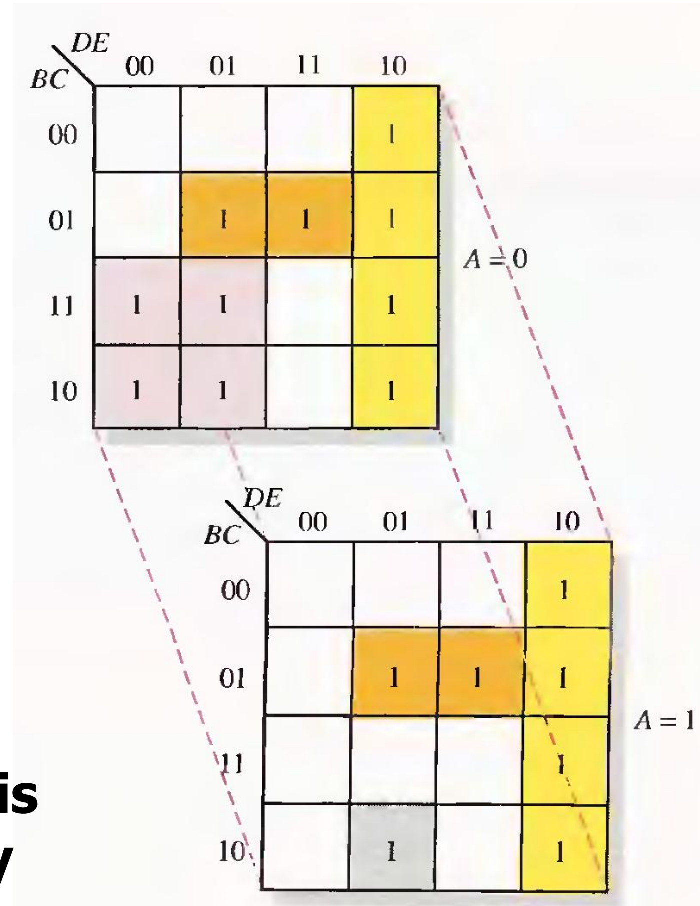
5 veriable K map



Cell Adjacency

- The best way to visualize cell adjacencies between the two 16-cel1 maps is to imagine that the A = 0 map is placed on top of the A = 1 map.
- Each cell in the A = 0 map is adjacent to the cell directly below it in the A = 1 map.

Cell Adjacency



Each cell in the A = 0 map is adjacent to the cell directly below it in the A = I map.

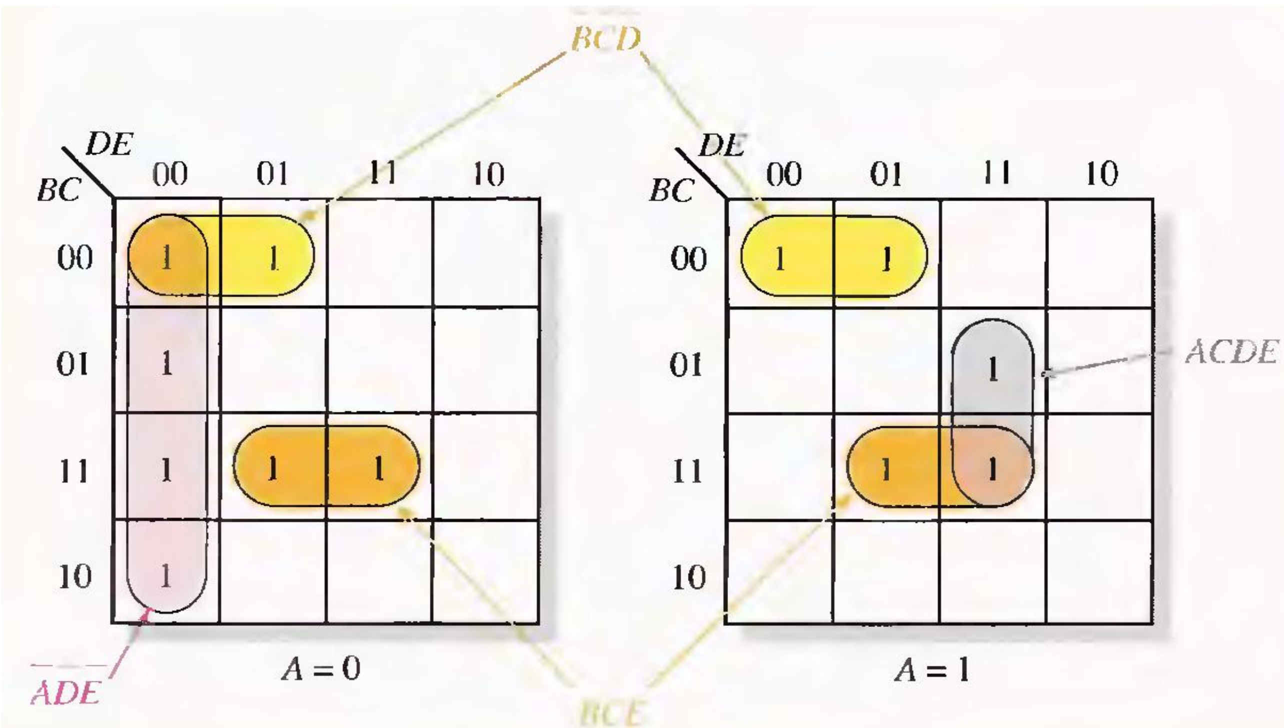
5 variable K Map

- The simplified expression taken from the map is developed as follows
  - The term for the yellow group is DE′.
  - The term for the orange group is B′CE.
  - The term for the light red group is A′BD′.
  - The term for the gray cell grouped with the red cell is BC′ D′E.
  - The final SOP expression is,
    - **X = DE′ + B′CE + A′BD′ + BC′ D′E**

Example

Use a Karnaugh map to minimize the following standard SOP 5-variable expression:

$$X = \overline{A}\,\overline{B}\,\overline{C}\,\overline{D}\,\overline{E} + \overline{A}\,\overline{B}C\overline{D}\,\overline{E} + \overline{A}BC\overline{D}\,\overline{E} + \overline{A}B\overline{C}\,\overline{D}\,\overline{E} + \overline{A}\,\overline{B}\,\overline{C}\,\overline{D}E + \overline{A}BC\overline{D}E$$
$$+ \overline{A}BCDE + A\overline{B}\,\overline{C}\,\overline{D}\,\overline{E} + A\overline{B}\,\overline{C}\,\overline{D}E + ABC\overline{D}E + ABCDE + A\overline{B}CDE$$



- Final Simplified Expression is,

$$X + \overline{A}\,\overline{D}\,\overline{E} + \overline{B}\,\overline{C}\,\overline{D} + BCE + ACDE$$


thanks
for the
tolerance