Two-Variable K-Map

→ In two-variable map, there are four minterms for two variables, hence the map consists of four squares, one for each minterm.

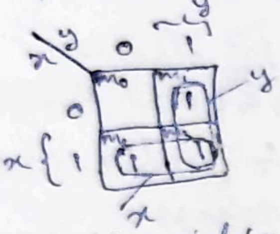
m_0	m_1
m_2	m_3

x \ y	0	1 (y)
0	m_0 $x'y'$	m_1 $x'y$
1 (x)	m_2 xy'	m_3 xy

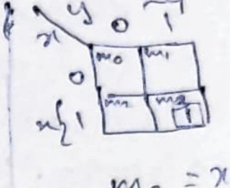
→ The 0 and 1 marked in each row and column designate the values of variables.

→ Variable x appears primed in row 0 and unprimed in row 1. Similarly, y appears primed in column 0 and unprimed in column 1.

→ If we mark the squares whose minterms belong to a given function, the two-variable map becomes another useful way to represent any one of the 16 Boolean functions of two variables.



$$m_1 + m_2 + m_3 = x'y + xy' + xy$$
$$= x + y$$



$m_3 = xy$

Three-Variable K-Map

→ In three-variable K-map, there are eight minterms for three binary variables, therefore, the map consists of eight squares.

→ The minterms are arranged, not in a binary sequence, but in a sequence similar to the Gray code.

→ The characteristic of this sequence is that only one bit changes in value from the adjacent column next to it.

m_0	m_1	m_3	m_2
m_4	m_5	m_7	m_6

x \ yz	00	01	11	10
0	m_0 $x'y'z'$	m_1 $x'y'z$	m_3 $x'yz$	m_2 $x'yz'$
1 (x)	m_4 $xy'z'$	m_5 $xy'z$	m_7 xyz	m_6 xyz'

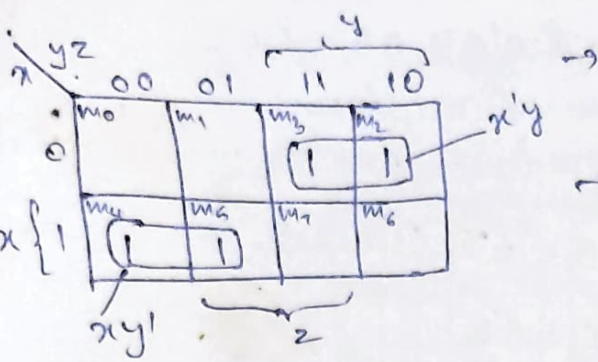
(y spans columns 11 and 10; z spans columns 01 and 11)

→ Any two adjacent squares in the map differ by only one variable, which is primed in one square and unprimed in the ~~(map differ by)~~ other.

→ From the postulates of Boolean algebra, the sum of two minterms in adjacent squares can be simplified to a single product term consisting of only two literals.

→ Thus, any two minterms in adjacent squares (vertically or horizontally, but not diagonally, adjacent) that are ORed together will cause a removal of the dissimilar variable.

Q) Simplify the Boolean function $F(x,y,z) = \sum(2,3,4,5)$



→ First, a 1 is marked in each minterm square that represents the function.

→ The next step is to find possible adjacent squares. These are indicated in the map by two (shaded) rectangles each enclosing two 1's

→ The upper right rectangle represents the area enclosed by $x'y$. Similarly, the lower left rectangle represents the product term xy'

→ The sum of four minterms can be replaced by a sum of only two product terms.

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

*→ The number of adjacent squares that may be combined must always represent a number that is a power of two, such as 1, 2, 4 and 8.

→ As more adjacent squares are combined we obtain a product term with fewer literals.

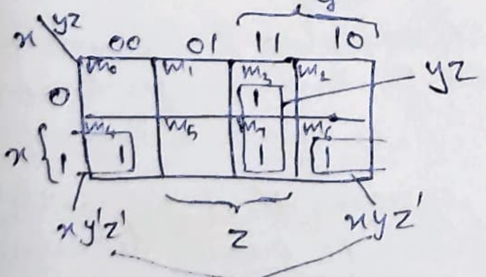
↳ One square represents one minterm, giving a term with three literals.

↳ Two adjacent squares represent a term with two literals

↳ ~~Eight~~ Four adjacent squares represent a term with one literal.

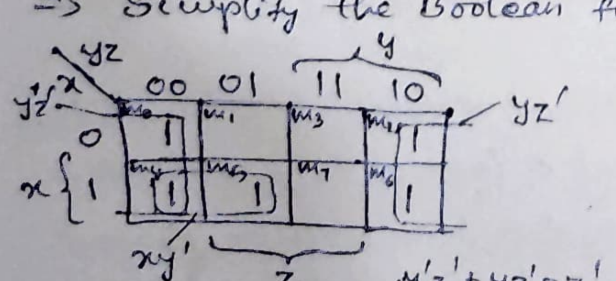
↳ Eight adjacent squares encompass the entire map and produce a function that is always equal to 1.

Q, Simplify the Boolean function $F(x,y,z) = \sum(3,4,6,7)$



$$F = yz + xz'$$

Q, Simplify the Boolean function $F(x,y,z) = \sum(0,2,4,5,6)$

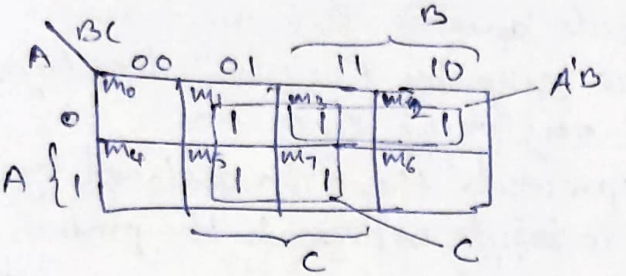


$$F = z' + xy'$$

Q) For the Boolean function $F = A'C + A'B + AB'C + BC$

a) Express this function as a sum of minterms.

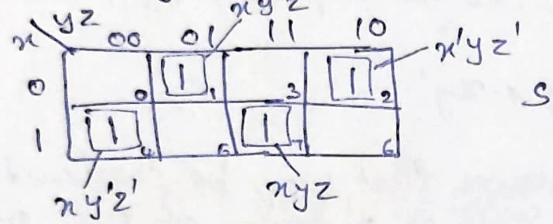
b) Find the minimal sum-of-products expression.



$F(A,B,C) = \Sigma(1,2,3,5,7)$

$F = C + A'B$

Q) Simplify the Boolean expression $S(x,y,z) = \Sigma(1,2,4,7)$



$S(x,y,z) = x'y'z + x'yz' + xy'z' + xyz$

Four-Variable K-Map

→ In four-variable K-map, there are 16(~~eight~~) minterms for four binary variables, therefore, the map consists of 16 squares.

m_0	m_1	m_3	m_2
m_4	m_5	m_7	m_6
m_{12}	m_{13}	m_{15}	m_{14}
m_8	m_9	m_{11}	m_{10}

wx \ yz	00	01	11 (y)	10 (y)
00	m_0 $w'x'y'z'$	m_1 $w'x'y'z$	m_3 $w'x'yz$	m_2 $w'x'yz'$
01 (x)	m_4 $w'xy'z'$	m_5 $w'xy'z$	m_7 $w'xyz$	m_6 $w'xyz'$
11 (w, x)	m_{12} $wxy'z'$	m_{13} $wxy'z$	m_{15} $wxyz$	m_{14} $wxyz'$
10 (w)	m_8 $wx'y'z'$	m_9 $wx'y'z$	m_{11} $wx'yz$	m_{10} $wx'yz'$

(z spans columns 01 and 11)

→ The map minimization of four-variable Boolean function is similar to the method used to minimize three-variable functions.

→ The combination of adjacent squares that is useful during the simplification process is easily determined from inspection of the four-variable map:

↳ One square represents one minterm, giving a term with four literals.

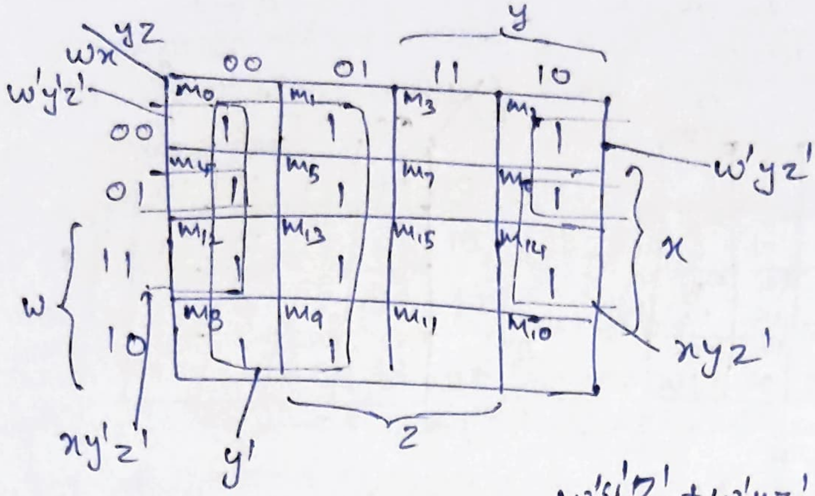
↳ Two adjacent squares represent a term with three literals.

↳ Four adjacent squares represent a term with two literals.

↳ Eight adjacent squares represent a term with one literal.

↳ Sixteen adjacent squares procedure a function that is always equal to 1.

Q→ Simplify the Boolean function $F(w,x,y,z) = \sum(0,1,2,4,5,6,8,9,12,13,14)$

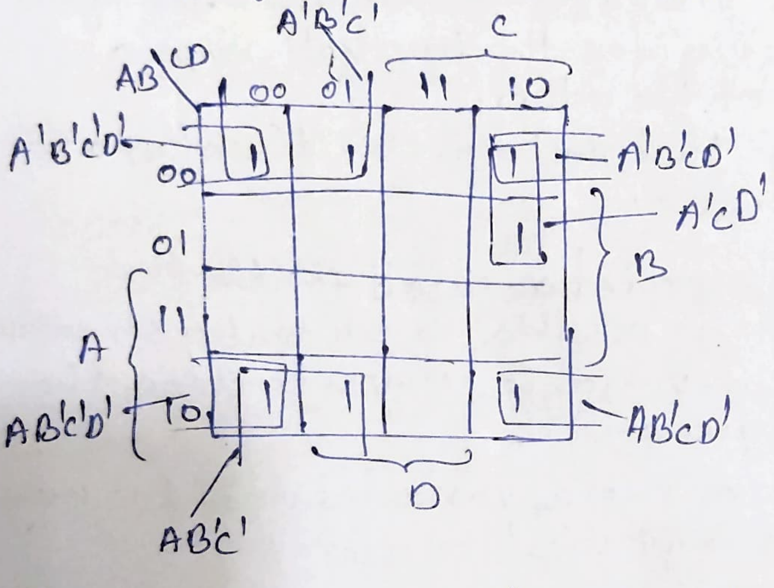


$w'y'z' + w'yz' = w'z'$

$xy'z' + xyz' = xz'$

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

Q→ Simplify the Boolean function $F = A'B'C' + B'CD' + A'BCD' + AB'C'$



$A'B'C'D' + A'B'CD' = A'B'D'$

$AB'C'D' + AB'CD' = AB'D'$

$A'B'D' + AB'D' = B'D'$

$A'B'C' + AB'C' = B'C'$

$F = B'D' + B'C' + A'CD'$

- Five-Variable Map and Six-Variable Map

→ A five-variable map needs 32 squares and a six-variable map needs 64 squares.

→ When the number of variables becomes large, the number of squares becomes excessive and the geometry for combining adjacent squares becomes more complex.

→ The five-variable map consists of 2 four-variable maps with variables A, B, C, D, and E.

→ The left-hand four-variable map represents the 16 squares in which A = 0 and the other four-variable map represents the squares in which A = 1.

→ Minterms 0 through 15 belong to A = 0 and minterms 16 through 31 to A = 1

A = 0

BC \ DE	00	01	11	10
00	m_0 0	m_1 1	m_3 3	m_2 2
01	m_4 4	m_5 5	m_7 7	m_6 6
11	m_{12} 12	m_{13} 13	m_{15} 15	m_{14} 14
10	m_8 8	m_9 9	m_{11} 11	m_{10} 10

A = 1

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

(D spans columns 11, 10; E spans columns 01, 11; B spans rows 11, 10; C spans rows 01, 11)

→ Each four-variable map retains the previously defined adjacency when taken seperately

→ In addition, each square in the A=0 map is adjacent to the corresponding square in the A=1 map.

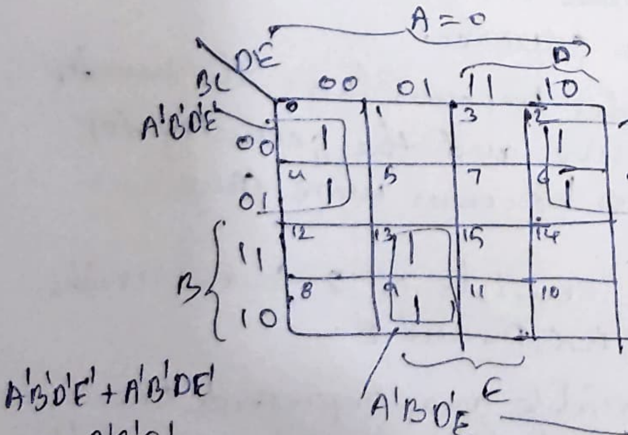
→ The best way to visualize the new rule for adjacent squares is to consider the two half maps as being one on top of the other.

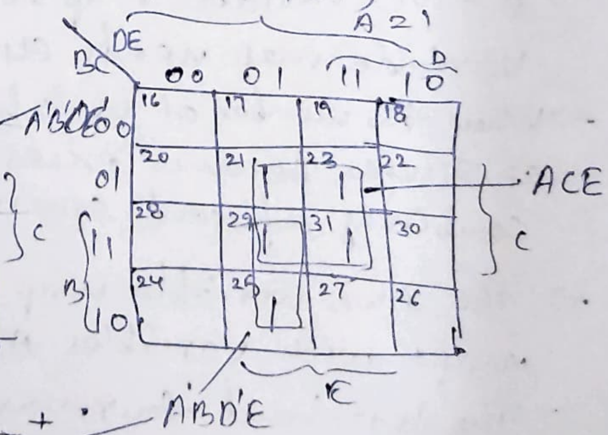
→ Any two squares that fall one over the other are considered adjacent.

→ By following the procedure used for the five-variable map, it is possible to construct a six-variable map with 4 four-variable maps to obtain the required 64 squares.

→ Maps with six or more variables need too many squares and are impractical to use.

Q → Simplify the Boolean function $F(A,B,C,D,E) = \Sigma(0,2,4,6,9,13,21,23,25,29,31)$





+ BD'E

F = A'B'E' + BD'E + ACE