Combinational Circuit Minimization

- **Canonical sum and product logic expressions do not provide a circuit realization with the minimum number of gates.**
- **Minimization methods reduce the cost of two level AND-OR, NAND-NAND, OR-AND, NOR-NOR circuits in three ways:**

1. **By minimizing the number of first level gates**
2. **By minimizing the number of inputs of each first-level gate.**
3. **Minimizing the inputs of the second level gate**

- **Most minimization methods are based on the combining theorems T10, T10':**

given product term.Y + given product term.Y' = given product term

(given sum term+Y).(given sum term + Y') = given sum term

Karnaugh Maps

- A Karnaugh Map or (K-map for short) is a graphical representation of the truth table of a logic function.
- The K-map for an n-input logic function is an array with 2^n cells or squares, one for each input combination or minterm.
- The rows and columns are labeled so that the input combination for any cell is determined from the row and column headings.
- The row and columns of the map are ordered in such a way that each cell differs from an adjacent cell in only one input variable:
  - Thus for an n-variable K-map, each cell has n adjacent cells.
- The K-map for a function is filled by putting:
  - a ‘1’ in the square corresponding to a minterm
  - a ‘0’ otherwise (maybe omitted)

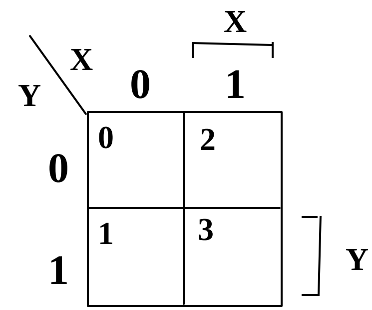
2-Variable K-map

For a 2-variable logic function F(X,Y):

Truth Table:

Row	X	Y	F	Minterm
0	0	0	F(0,0)	X'.Y'
1	0	1	F(0,1)	X'.Y
2	1	0	F(1,0)	X.Y'
3	1	1	F(1,1)	X .Y

K-map

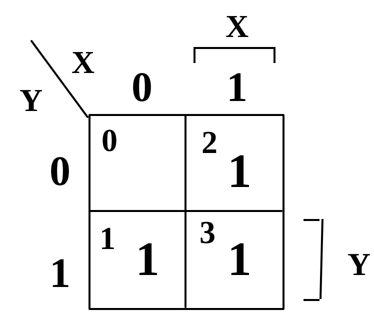


Example: For the function $F(X,Y) = \Sigma_{X,Y}(1,2,3)$

Truth Table:

Row	X	Y	F
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	1

K-map



3-Variable K-map

For a 3-variable logic function F(X,Y,Z):

Truth Table:

Row	X	Y	Z	F	Minterm
0	0	0	0	F(0,0,0)	X'.Y'.Z'
1	0	0	1	F(0,0,1)	X'.Y'.Z
2	0	1	0	F(0,1,0)	X'.Y.Z'
3	0	1	1	F(0,1,1)	X'.Y.Z
4	1	0	0	F(1,0,0)	X.Y'.Z'
5	1	0	1	F(1,0,1)	X.Y'.Z
6	1	1	0	F(1,1,0)	X.Y.Z'
7	1	1	1	F(1,1,1)	X.Y.Z

K-map

(X spans columns 11, 10; Y spans columns 01, 11; Z spans row 1)

Z \ XY	00	01	11	10
0	0	2	6	4
1	1	3	7	5

Example: For the function F(X,Y,Z) = s $_{X,Y,Z}$ (1,2,5,7)

Truth Table:

Row	X	Y	Z	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

K-map

(X spans columns 11, 10; Y spans columns 01, 11; Z spans row 1)

Z \ XY	00	01	11	10
0	0	2: 1	6	4
1	1: 1	3	7: 1	5: 1

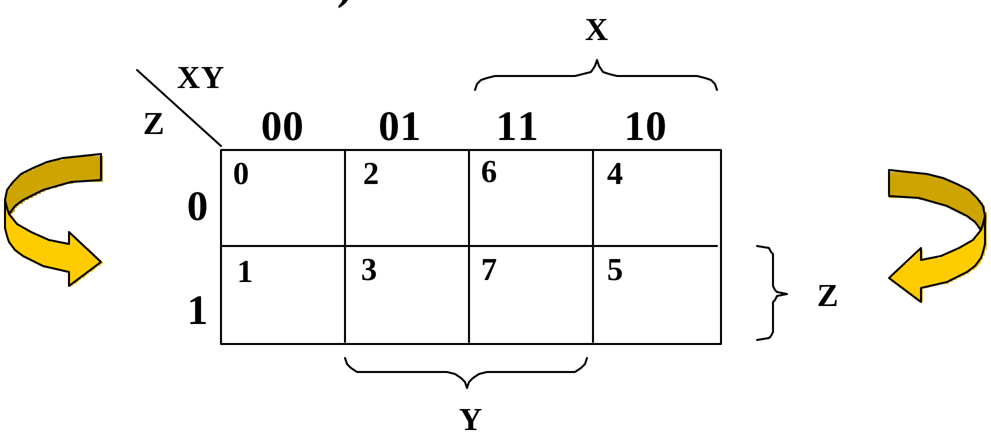
3-Variable K-map (continued)

- There is a horizontal adjacency wrap-around in the 3-variable K-map:

  For example:

  – Cell 0 (minterm 0 = X’.Y’.Z’) is adjacent to:
    - cell 4 (minterm 4, = X.Y’.Z’) by wrap-around.
    - in addition to being adjacent to cells 1, 2 (minterm 1 = X’.Y’.Z minterm 2, = X’.Y.Z’)

  – Cell 1 (minterm 1, X’.Y’.Z) is adjacent to:
    - cell 5 (minterm 5, X.Y’.Z) by wrap-around.
    - in addition to being adjacent to cells 0 , 2 (minterm 0 = X’.Y’.Z’ minterm 3 = X’.Y.Z)



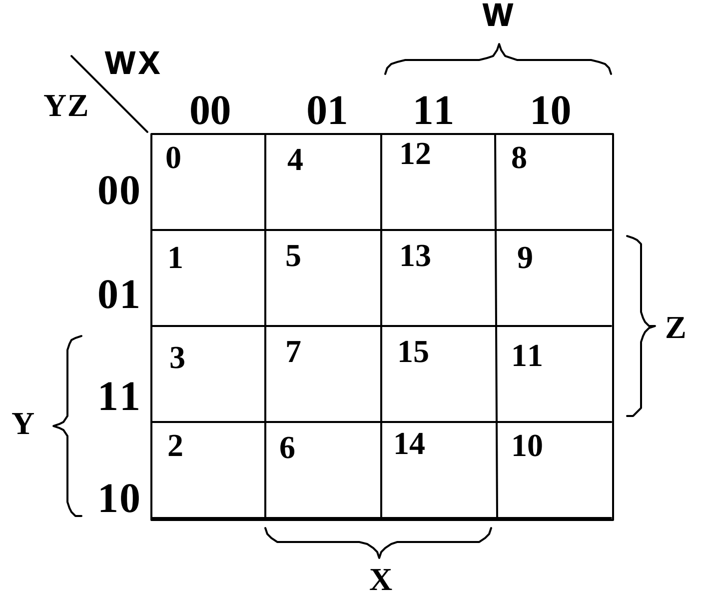
4-Variable K-map

For a 4-variable logic function F(W,X,Y,Z):

Truth Table:

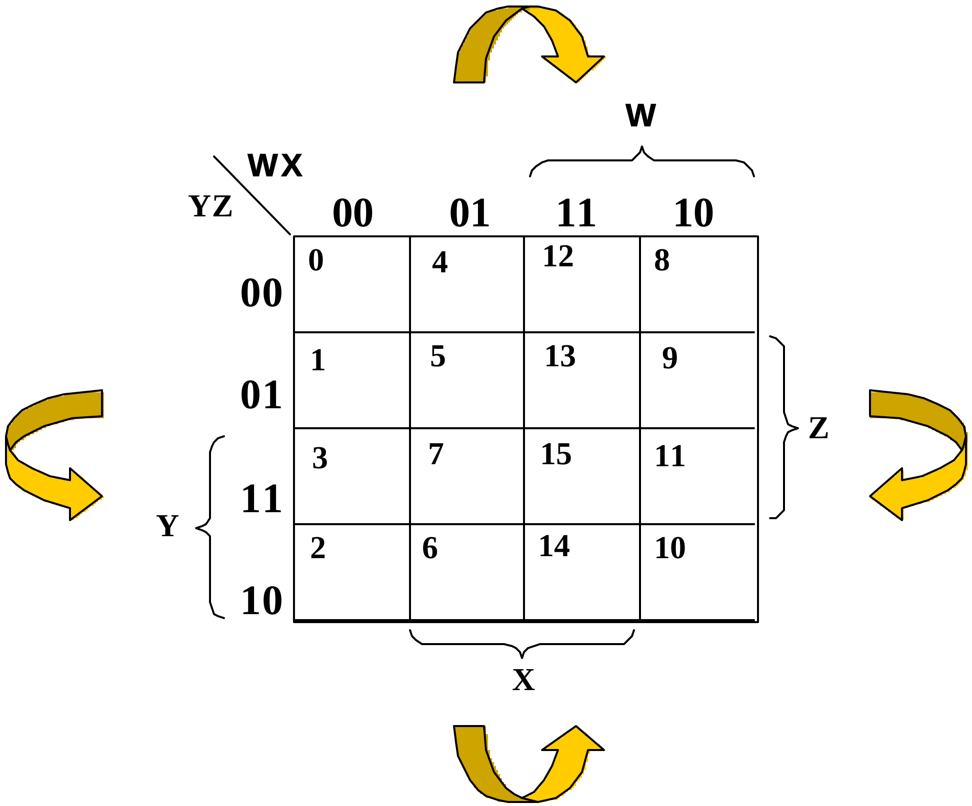
Row	W	X	Y	Z	F	Minterm
0	0	0	0	0	F(0,0,0,0)	W'.X'.Y'.Z'
1	0	0	0	1	F(0,0,0,1)	W'. X'.Y'.Z
2	0	0	1	0	F(0,0,1,0)	W'. X'.Y.Z'
3	0	0	1	1	F(0,0,1,1)	W'. X'.Y.Z
4	0	1	0	0	F(0,1,0,0)	W'. X.Y'.Z'
5	0	1	0	1	F(0,1,0,1)	W'.X.Y'.Z
6	0	1	1	0	F(0,1,1,0)	W'.X.Y.Z'
7	0	1	1	1	F(0,1,1,1)	W'.X.Y.Z
8	1	0	0	0	F(1,0,0,0)	W.X'.Y'.Z'
9	1	0	0	1	F(1,0,0,1)	W.X'.Y'.Z
10	1	0	1	0	F(1,0,1,0)	W.X'.Y.Z'
11	1	0	1	1	F(1,0,1,1)	W.X'.Y.Z
12	1	1	0	0	F(1,1,0,0)	W.X.Y'.Z'
13	1	1	0	1	F(1,1,0,1)	W.X.Y'.Z
14	1	1	1	0	F(1,1,1,0)	W.X.Y.Z'
15	1	1	1	1	F(1,1,1,1)	W.X.Y.Z

K-map



4-Variable K-map (continued)

- **There are 2 adjacency wrap-arounds in the 4-variable K-map : a horizontal wrap-around and a vertical wrap-around.**
- **Every cell thus has 4 neighbours. For example, cell 0 corresponding to minterm 0 is adjacent to: cells *1, 2, 4, 8***



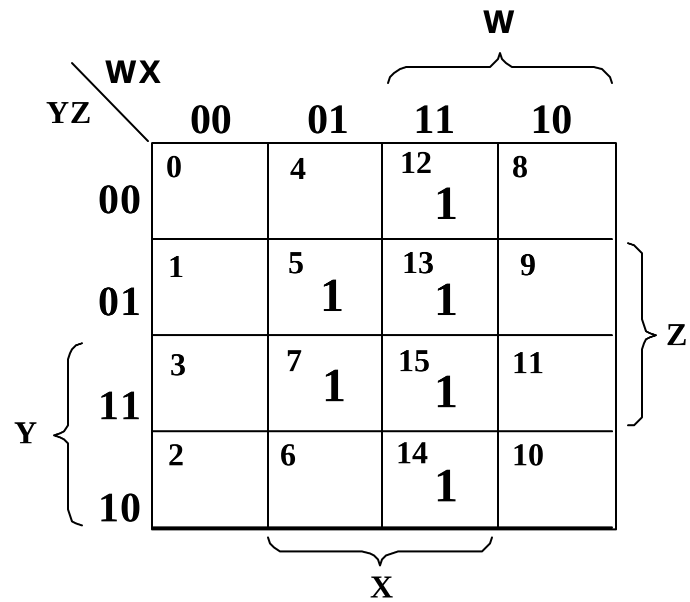
4-Variable K-map Example

For the function $F(W,X,Y,Z) = \Sigma_{W,X,Y,Z}(5,7,12,13,14,15)$

Truth Table:

Row	W	X	Y	Z	F
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	0
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	1

K-map



Minimizing Sum of Products using K-maps

- Each input combination with "1" in a Karnaugh map or truth table correspond to a minterm in the function's canonical sum representation.
- Pairs of adjacent "1" cells in the Karnaugh map indicate minterms that differ in only one variable.
- Using the generalization of T10, such adjacent minterm pairs can be combined into a single product term.
- In general, one can simplify a logic function by combining pairs of adjacent 1-cell minterms and writing a sum of products expression to cover all of the 1-cells.

K-Map Minimization Rules and Definitions

- **A minimal sum of a logic function $F(X_1, X_2, ..X_n)$ is a sum-of-products expression for F such that no other similar expression for F has fewer product terms, and other expressions with the same number of product terms have at least the same number of literals.**
- **A set of 2^i 1-cells are combined into a single square or rectangle if i variables take all 2^i possible combinations within the set while the remaining variables have the same value.**
- **The corresponding product term for the combined cells has n-i literals.**
- **Only the variables that have the same value appear in the resulting product term:**
  - **A variable in the resulting product term is complemented if it appears as 0 in all the 1-cells, and uncomplemented if it appears as 1.**

Minimization Using K-maps

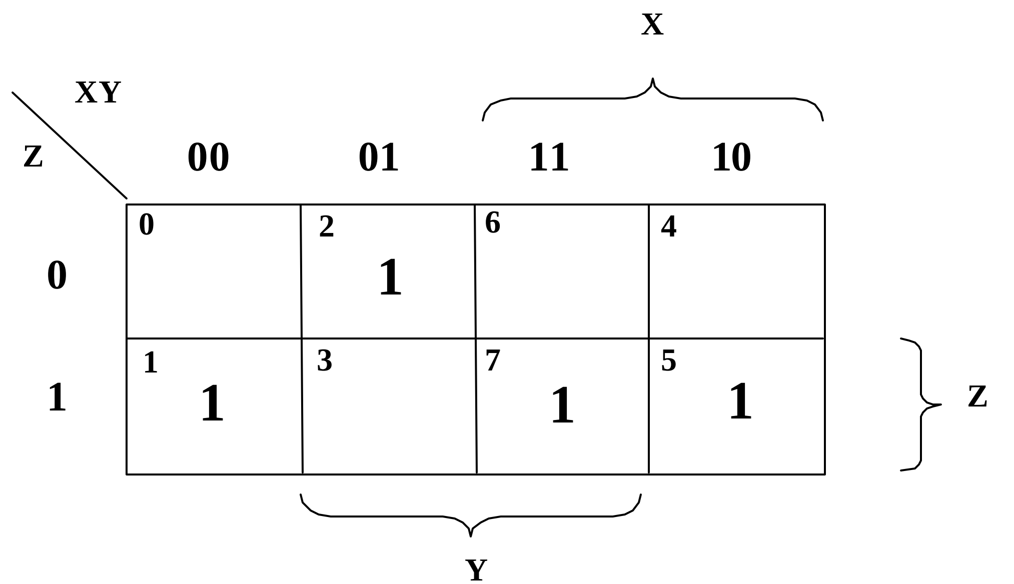
- **Group or combine as many adjacent 1-cells as possible:**
  - **The larger the group is, the fewer the number of literals in the resulting product term.**
  - **Each group of combined adjacent 1-cells must have a number of cells equal to *powers of two*: 1, 2, 4, 8, …**
  - **Grouping 2 adjacent 1-cells eliminates 1 variable, grouping 4 1-cells eliminates 2 variables, grouping 8 1-cells eliminates 3 variables, and so on. In general, grouping 2^n squares eliminates *n* variables.**
- **Select as few groups as possible to cover all the 1-cells (minterms) of the function:**
  - **The fewer the groups, the fewer the number of product terms in the minimized function.**

3-Variable K-map Minimization Example

- **Using K-map, find a minimal sum of products (SOP) expression expression for the function:**

$$F(X,Y,Z) = S_{X,Y,Z}\ (1,2,5,7)$$

K-map



Truth Table

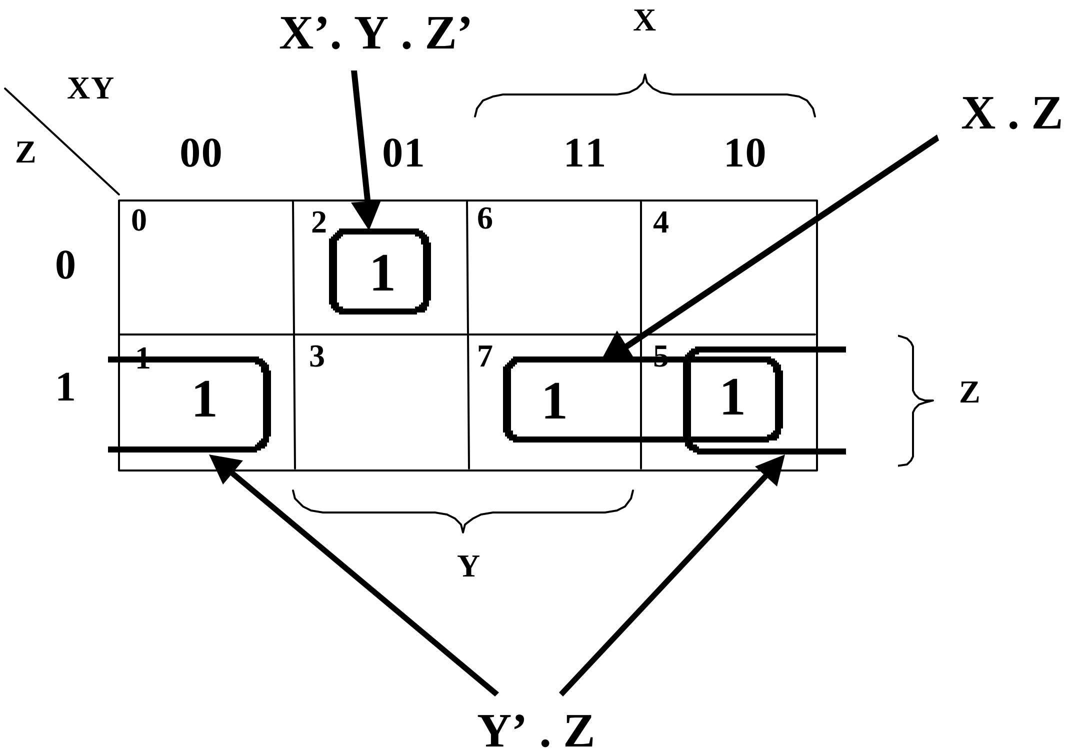
Row	X	Y	Z	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

3-Variable K-map Minimization Example

- Using K-map, find a minimal sum of products (SOP) expression expression for the function:

$$F(X,Y,Z) = S_{X,Y,Z}\ (1,2,5,7)$$

K-map



Truth Table

Row	X	Y	Z	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

Minimum SOP for F = X'. Y . Z' + X . Z + Y' . Z

3-Variable K-map Minimization Example

- **Using K-map, find a minimal sum of products (SOP) expression expression for the function:**

$$F(X,Y,Z) = S_{X,Y,Z}\ (0,1,4,5,\ 6)$$

K-map

X (columns 11, 10) · Y (columns 01, 11) · Z (row 1)

Z \ XY	00	01	11	10
0	0: 1	2:	6: 1	4: 1
1	1: 1	3:	7:	5: 1

Truth Table

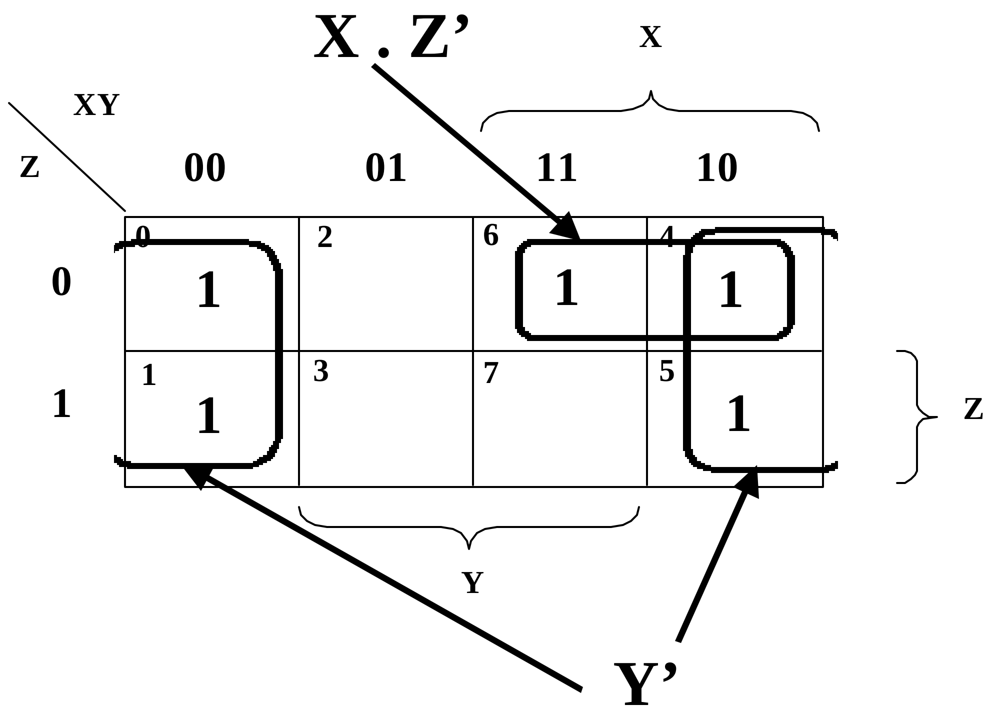
Row	X	Y	Z	F
0	0	0	0	1
1	0	0	1	1
2	0	1	0	0
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

3-Variable K-map Minimization Example

- **Using K-map, find a minimal sum of products (SOP) expression expression for the function:**

$$F(X,Y,Z) = S_{X,Y,Z}\ (0,1,4,5,\ 6)$$

K-map



Truth Table

Row	X	Y	Z	F
0	0	0	0	1
1	0	0	1	1
2	0	1	0	0
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

Minimum SOP for F = Y' + X . Z'

4-Variable K-map Minimization Example

- Using K-map, find a minimal sum of products (SOP) expression expression for the function:

$$F(N3,N2,N1,N0) = S_{N3,N2,N1,N0}\ (1,2,3,5,7,11,13)$$

K-map

Truth Table:

Row	W	X	Y	Z	F
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	0

N3 spans columns 11 and 10; N2 spans columns 01 and 11; N0 spans rows 01 and 11; N1 spans rows 11 and 10.

N1 N0 \ N3 N2	00	01	11	10
00	0	4	12	8
01	1: 1	5: 1	13: 1	9
11	3: 1	7: 1	15	11: 1
10	2: 1	6	14	10

4-Variable K-map Minimization Example

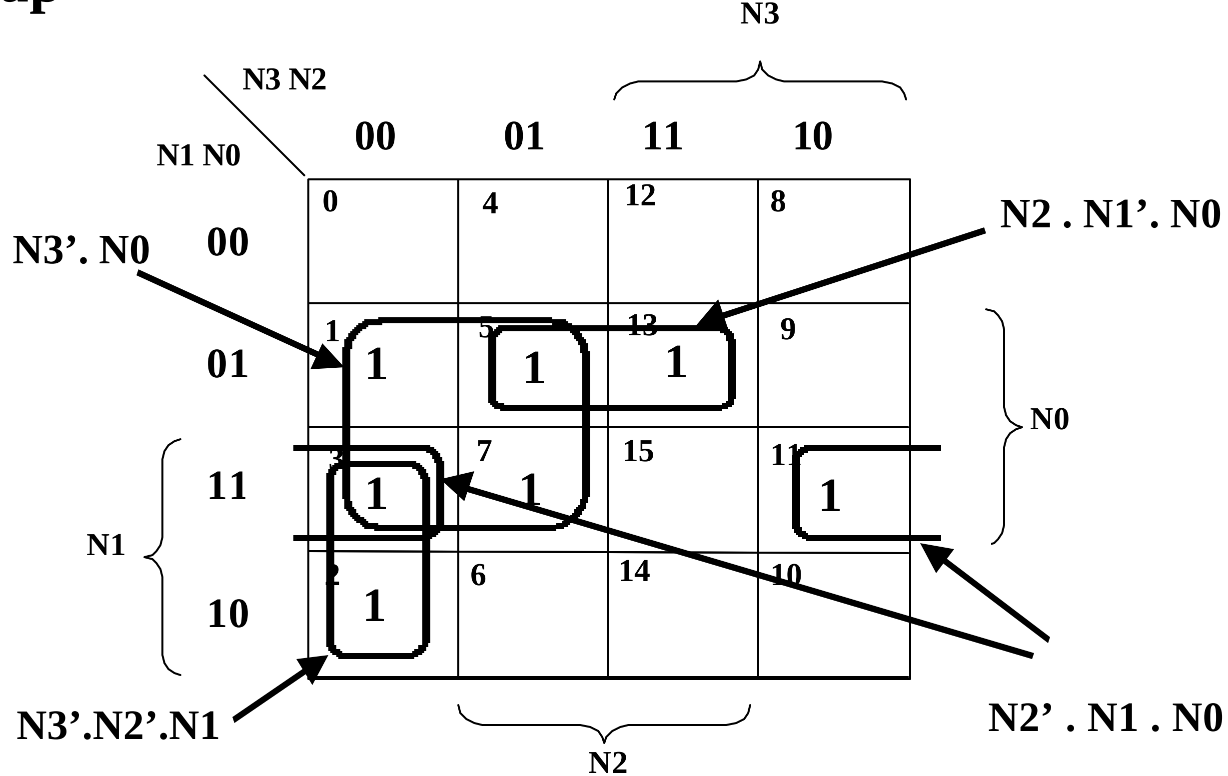
- Using K-map, find a minimal sum of products (SOP) expression expression for the function:

$$F(N3,N2,N1,N0) = \Sigma_{N3,N2,N1,N0}(1,2,3,5,7,11,13)$$

Truth Table:

Row	W	X	Y	Z	F
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	0

K-map



Minimum SOP for F = N3'. N0 + N3'. N2' . N1 + N2'. N1 . N0 + N2 . N1'.N0

K-Map Minimization Rules and Definitions

- **A logic function $P(X_1, X_2, ..X_n)$ implies a logic function $F(X_1, …, X_n)$ if for every input combination such that P=1, then F=1 (F includes P , or F covers P).**
- **A prime implicant of a logic function $F(X_1, ..X_n)$ is a normal product term $P(X_1, ..X_n)$ that implies F, such that if any variable is removed from P, the the resulting product term does not imply F.**
- **A minimal sum is a sum of prime implicants (not necessarily all of them).**
- **A distinguished 1-cell of a logic function is an input combination that is covered by only one prime implicant.**
- **An essential prime implicant of a logic function is a prime implicant that covers one or more distinguished 1-cells and *must* be included every minimal sum expression for the function.**

4-Variable K-map Minimization Example

- **Using K-map, find a minimal sum of products (SOP) expression expression for the function:**

$$F(W,X,Y,Z) = \Sigma_{W,X,Y,Z}(2,3,4,5,6,7,11,13,15)$$

- **Also identify all prime implicants, distinguished 1-cells and the corresponding essential prime implicants that cover them.**

K-map

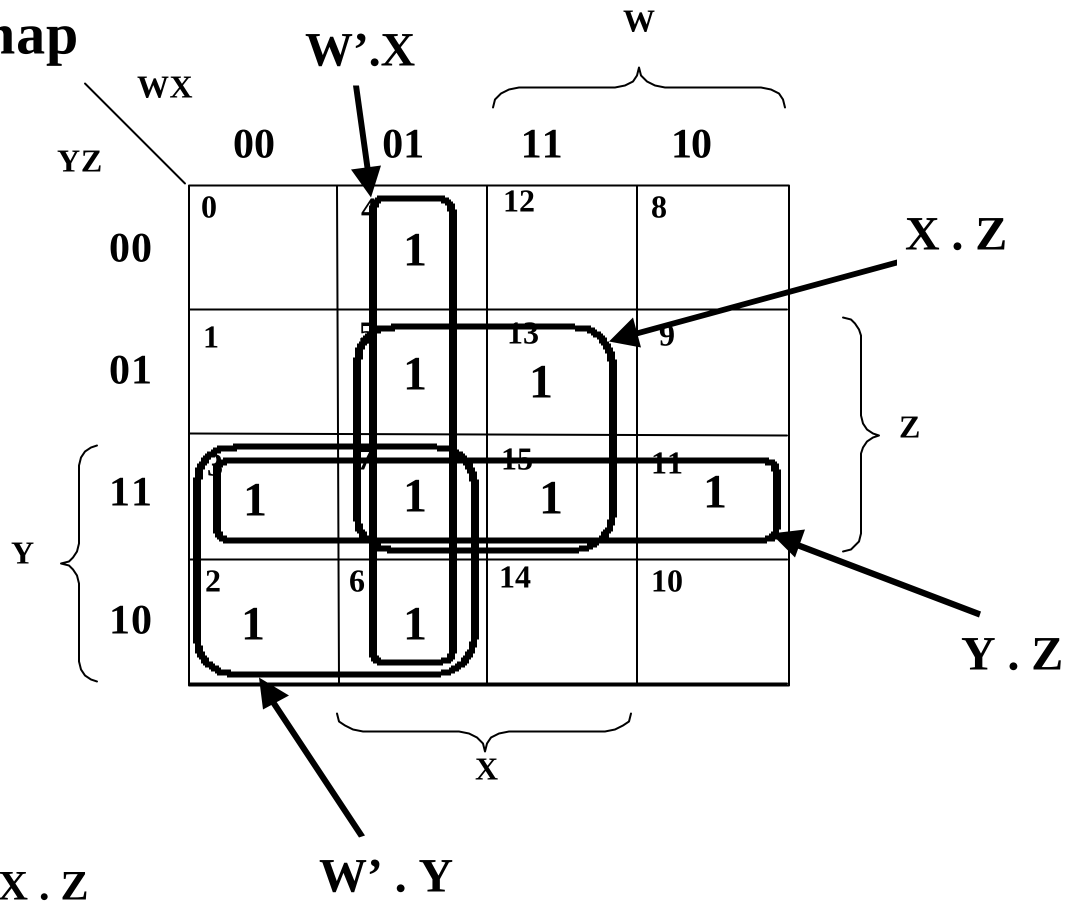
W (columns 11, 10); X (columns 01, 11); Y (rows 11, 10); Z (rows 01, 11)

YZ \ WX	00	01	11	10
00	0	4 1	12	8
01	1	5 1	13 1	9
11	3 1	7 1	15 1	11 1
10	2 1	6 1	14	10

4-Variable K-map Minimization Example

- **Using K-map, find a minimal sum of products (SOP) expression expression for the function: F(W,X,Y,Z) = $\Sigma_{W,X,Y,Z}$ (2,3,4,5,6,7,11,13,15)**
- **Also identify all prime implicants, distinguished 1-cells and the corresponding essential prime implicants that cover them.**

K-map



From K-map:

Prime Implicants:

W'. Y W' . X Y . Z X . Z

Distinguished 1-cells:

Cell 2 covered by W' . Y
Cell 4 covered by W' . X
Cell 11 covered by Y . Z
Cell 13 covered by X . Z

Here all prime implicants are essential prime implicants and all of them must be included in minimum SOP expression:

F = W' . Y + W'. X + Y . Z + X . Z

Minimization with Don’t care Input Combinations

- **In some cases, the output of a combinational circuit doesn’t matter for certain input combinations.**

- **Such combinations are called don’t cares and the output is represented in the truth table and K-maps as “d”.**

- **When using K-maps to minimize such functions:**
  - **Allow d’s to be included when circling sets of 1’s to make the sets as large as possible.**
  - **Do not circle any set that only contains d’s.**

4-Variable K-map Minimization Example With Don't cares

- **Using K-map, find a minimal sum of products (SOP) expression for prime BCD-digit detector which gives 1 when the input BCD digit is prime,**
- **Since the values 10-15 do not occur in a BCD digit minterms 10-15 are treated as don't cares giving the expression:**

$$F(N3,N2,N1,N0) = S_{N3,N2,N1,N0}(1,2,3,5,7) + d(10,11,12,13,14,15)$$

From K-map:

Prime Implicants:

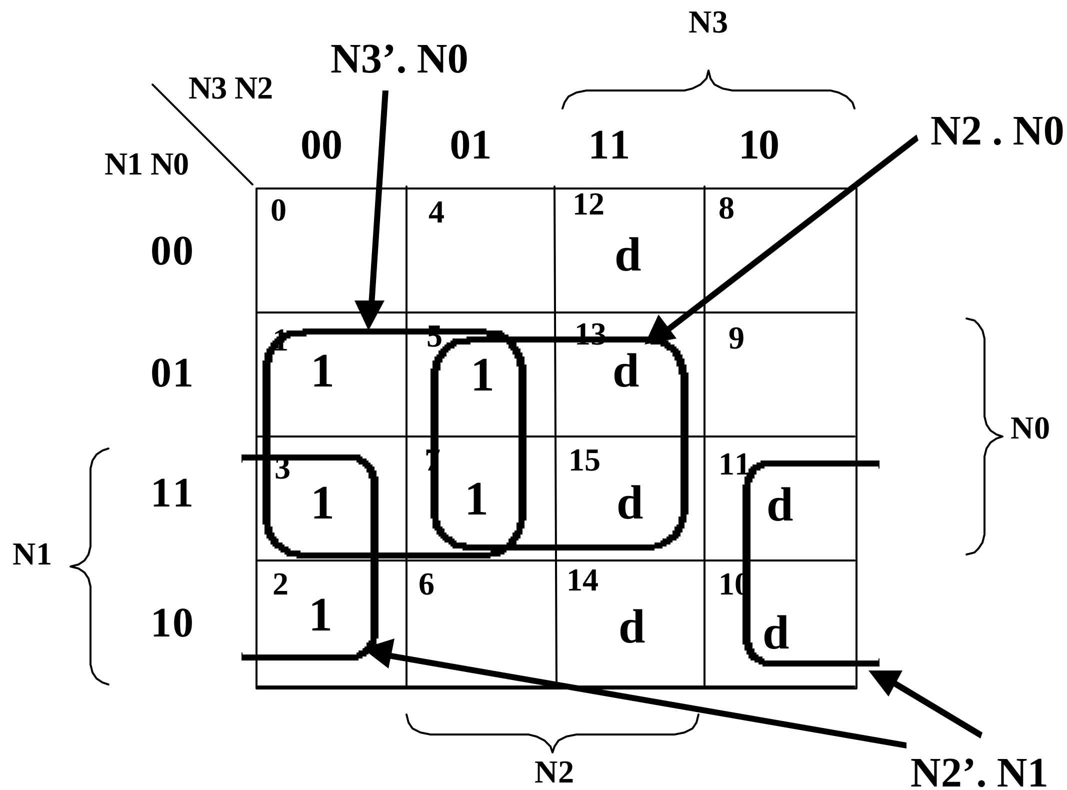
N3'. N0 N2'. N1 N2 . N0

Distinguished 1-cells:

Cell 1 covered by N3'. N0
Cell 2 covered by N2'. N1

Here not all prime implicants are essential prime implicants that must be included minimum SOP expression:

F = N3' . N0 + N2' . N1



K-map Minimization of Product of Sums

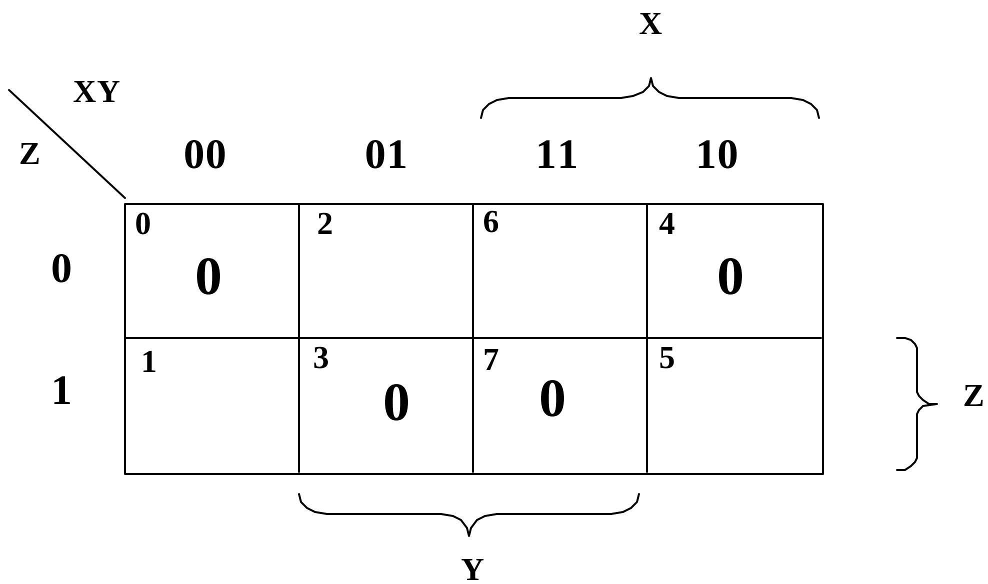
- **Similar to K-map minimization of sum of products by using duality and looking at 0-cells instead of 1-cells.**
- **A set of 2^i 0-cells may be combined if i variables take all 2^i possible combinations within the set while the remaining variables have the same value.**
- **In the resulting n-i literals sum term, a variable is complemented if it appears as 1 in all the 0-cells, and uncomplemented if it appears as 0.**
- **A prime implicate of a logic function $F(X_1, ..X_n)$, is a normal sum term $S(X_1, ..X_n)$ implied by F, such as if any variable is removed from S, then the resulting sum term is not implied by F.**
- **A minimal product is a product of prime implicates.**

K-map Product of Sums Minimization Example 1

- **Using K-map, find a minimal product of sums (POS) expression expression for the function:**

$$F(X,Y,Z) = P_{X,Y,Z}\ (0,3,4,7)$$

K-map



Truth Table

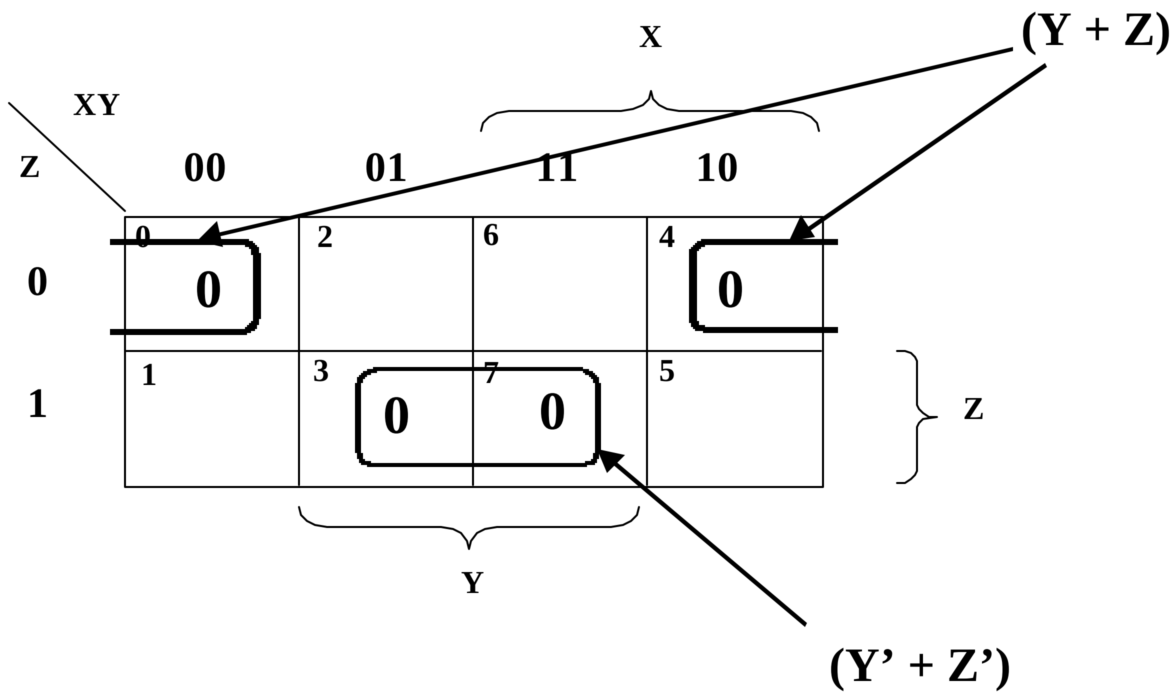
Row	X	Y	Z	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

K-map Product of Sums Minimization Example 1

- **Using K-map, find a minimal product of sums (POS) expression expression for the function:**

$$F(X,Y,Z) = \Pi_{X,Y,Z}(0,3,4,7)$$

K-map



Truth Table

Row	X	Y	Z	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

Minimum POS for F = $(Y + Z) \cdot (Y' + Z')$

K-map Product of Sums Minimization Example 2

- **Using K-map, find a minimal product of sums (POS) expression expression for the function:**

$$F(W,X,Y,Z) = \Pi_{W,X,Y,Z}(1,3,8,10,12,13,14,15)$$

K-map

YZ \ WX	00	01	11	10
00	0	4	12: 0	8: 0
01	1: 0	5	13: 0	9
11	3: 0	7	15: 0	11
10	2	6	14: 0	10: 0

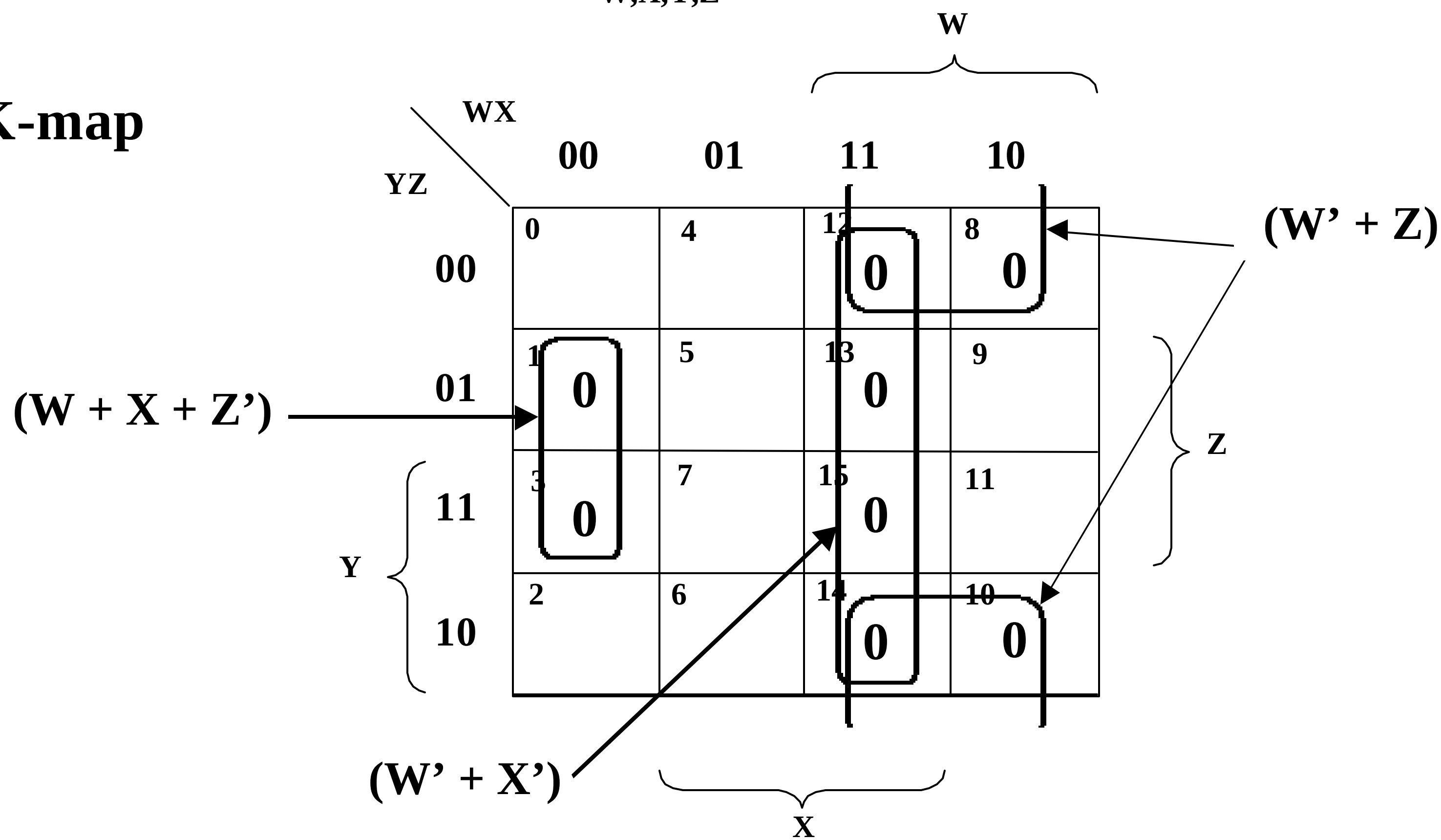
W (columns 11, 10); X (columns 01, 11); Y (rows 11, 10); Z (rows 01, 11)

K-map Product of Sums Minimization Example 2

- **Using K-map, find a minimal product of sums (POS) expression expression for the function:**

$$F(W,X,Y,Z) = \prod_{W,X,Y,Z}(1,3,8,10,12,13,14,15)$$

K-map



Minimum POS for F = (W + X + Z') . (W' + Z) . (W' + X')

5-variable K-maps

- **The K-map for a 5-variable logic function F(V,W,X,Y,Z) is organized as two 4-variable K-maps:**

YZ \ WX	00	01	11	10
00	0	4	12	8
01	1	5	13	9
11	3	7	15	11
10	2	6	14	10

W: columns 11, 10; X: columns 01, 11; Y: rows 11, 10; Z: rows 01, 11

V = 0

YZ \ WX	00	01	11	10
00	16	20	28	24
01	17	21	29	25
11	19	23	31	27
10	18	22	30	26

W: columns 11, 10; X: columns 01, 11; Y: rows 11, 10; Z: rows 01, 11

V = 1

Corresponding squares of each map are adjacent. Can be visualised as being one 4-variable map on top of another 4-variable map.

5-Variable K-map SOP Minimization Example

- Using K-map, find a minimal sum of products (SOP) expression expression for the function:

$$F(V,W,X,Y,Z) = \Sigma_{V,W,X,Y,Z}(4,5,6,7,9,11,13,15,25,27,29,31)$$

K-map

V = 0 (W spans columns 11, 10; X spans columns 01, 11; Y spans rows 11, 10; Z spans rows 01, 11)

YZ \ WX	00	01	11	10
00	0	4 1	12	8
01	1	5 1	13 1	9 1
11	3	7 1	15 1	11 1
10	2	6 1	14	10

V = 0

V = 1 (W spans columns 11, 10; X spans columns 01, 11; Y spans rows 11, 10; Z spans rows 01, 11)

YZ \ WX	00	01	11	10
00	16	20	28	24
01	17	21	29 1	25 1
11	19	23	31 1	27 1
10	18	22	30	26

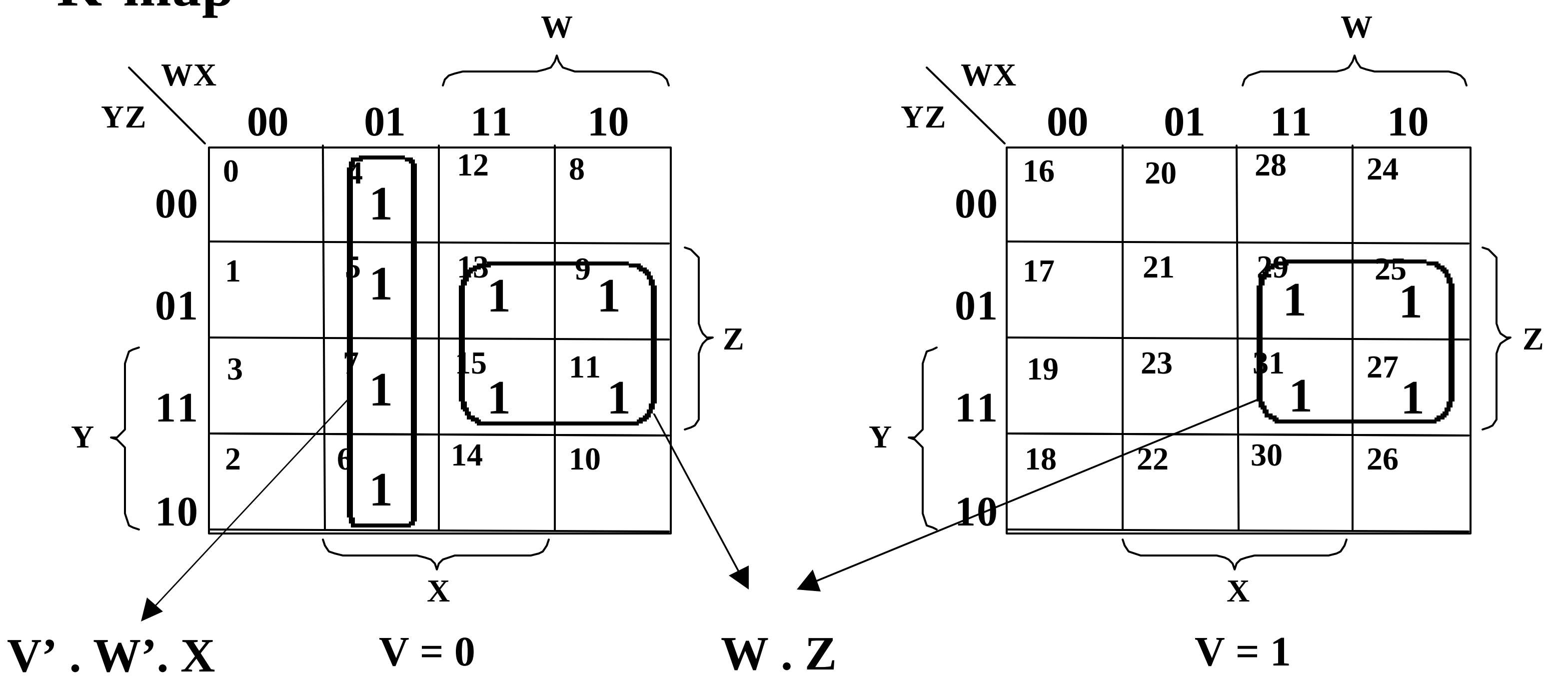
V = 1

5-Variable K-map SOP Minimization Example

- **Using K-map, find a minimal sum of products (SOP) expression expression for the function:**

$$F(V,W,X,Y,Z) = \Sigma_{V,W,X,Y,Z}(4,5,6,7,9,11,13,15,25,27,29,31)$$

K-map



Minimum SOP for F = V' . W'. X + W . Z

6-variable K-maps

K-map for a 6-variable logic function F(U,V,W,X,Y,Z) is organized as two 5-variable K-maps:

U,V = 0,0

YZ \ WX	00	01	11	10
00	0	4	12	8
01	1	5	13	9
11	3	7	15	11
10	2	6	14	10

U,V = 0,1

YZ \ WX	00	01	11	10
00	16	20	28	24
01	17	21	29	25
11	19	23	31	27
10	18	22	30	26

U,V = 1,0

YZ \ WX	00	01	11	10
00	32	36	44	40
01	33	37	45	41
11	35	39	47	43
10	34	38	46	42

U,V = 1,1

YZ \ WX	00	01	11	10
00	48	52	60	56
01	49	53	61	57
11	51	55	63	59
10	50	54	62	58

(In each map: W spans columns 11 and 10; X spans columns 01 and 11; Y spans rows 11 and 10; Z spans rows 01 and 11.)