Unit-3
Simplification of Boolean Functions

Karnaugh Map (K-Map) method for simplification of logic expression: K- map is a simple and straight forward graphical technique for simplifying Boolean functions. It is a diagram consisting of squares. It is pictorial form of truth table. Each square of map represents a minterm and logic expression can be written as SOP, i.e. sum of minterms.

K-Map for n variables is made up of 2^n squares. We can write either 1s or 0s in the K-Map square considering grouping of 1s gives the minimal SOP and grouping of 0s gives minimal POS.

Depending on the number of 1s and 0s available for grouping, the types of groups are:

1. **Pair:** Grouping of two 1s or two 0s is called pair. It consists of 1 Row x 2 Columns and
2 Rows x 1Column

1	1
1	

2. **Quad:** Grouping of four 1s or four 0s. It consists of 1 Row x 4 Columns and 4 Rows x 1 Column.

1	1	1	1
1			
1			
1			

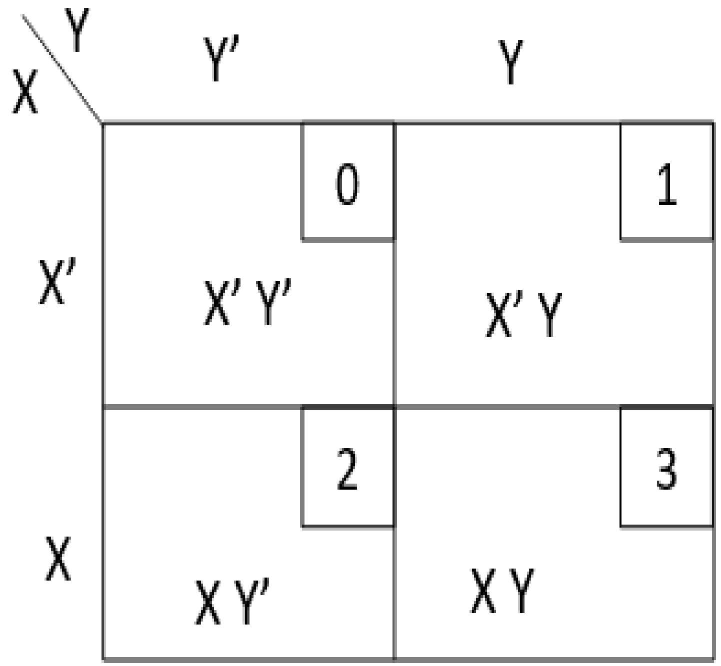
3. **Octet**: Grouping of eight 1s or eight 0s. It consists of 2 Rows x 4 Columns and 2 columns x 4 rows.

1	1	1	1
1	1	1	1
1	1		
1	1		

Rules for making groups in SOP:

1) No zeros are allowed in grouping.
2) No diagonals are allowed in grouping.
3) Groups should be as larger as possible.
4) Overlapping is allowed.

Two Variable K-Map: There are four minterms in two variable K-map. ($2^n = 2^2 = 4$). Hence map consists of 4 squares one for each minterm.

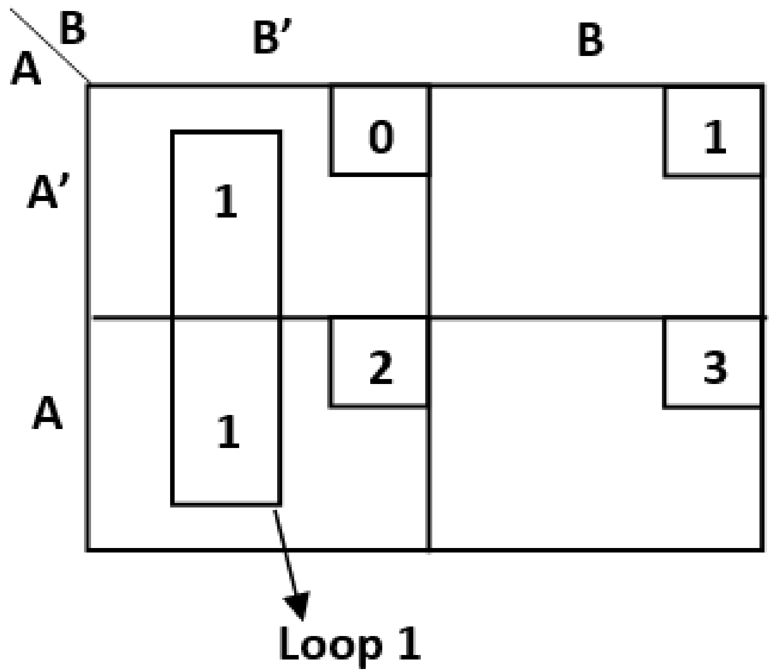


Example: Simplify: $Y = A'B' + AB'$ using K-map.

Solution;

Here,

$A'B' + AB' = B'(A' + A)$



Therefore; $F = B'$

Example: Simplify: $F(A,B) = \sum(0,1,3)$ using K-map.

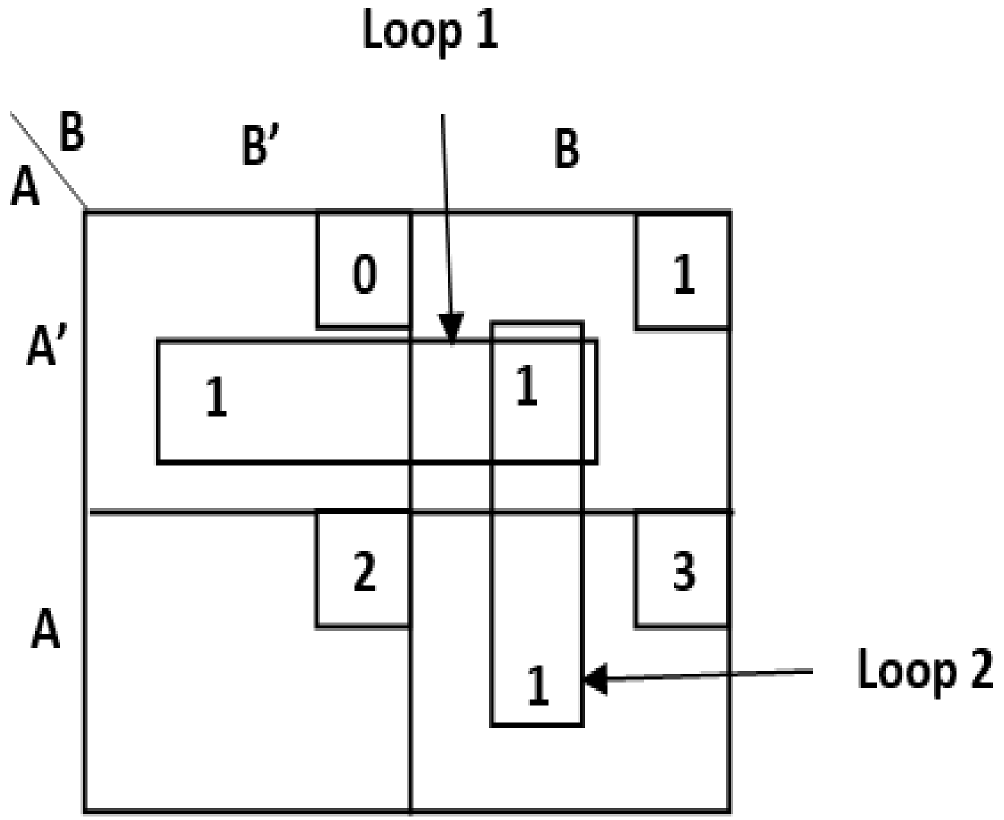
Solution;

Here,

0 = 0 0 = A’ B’

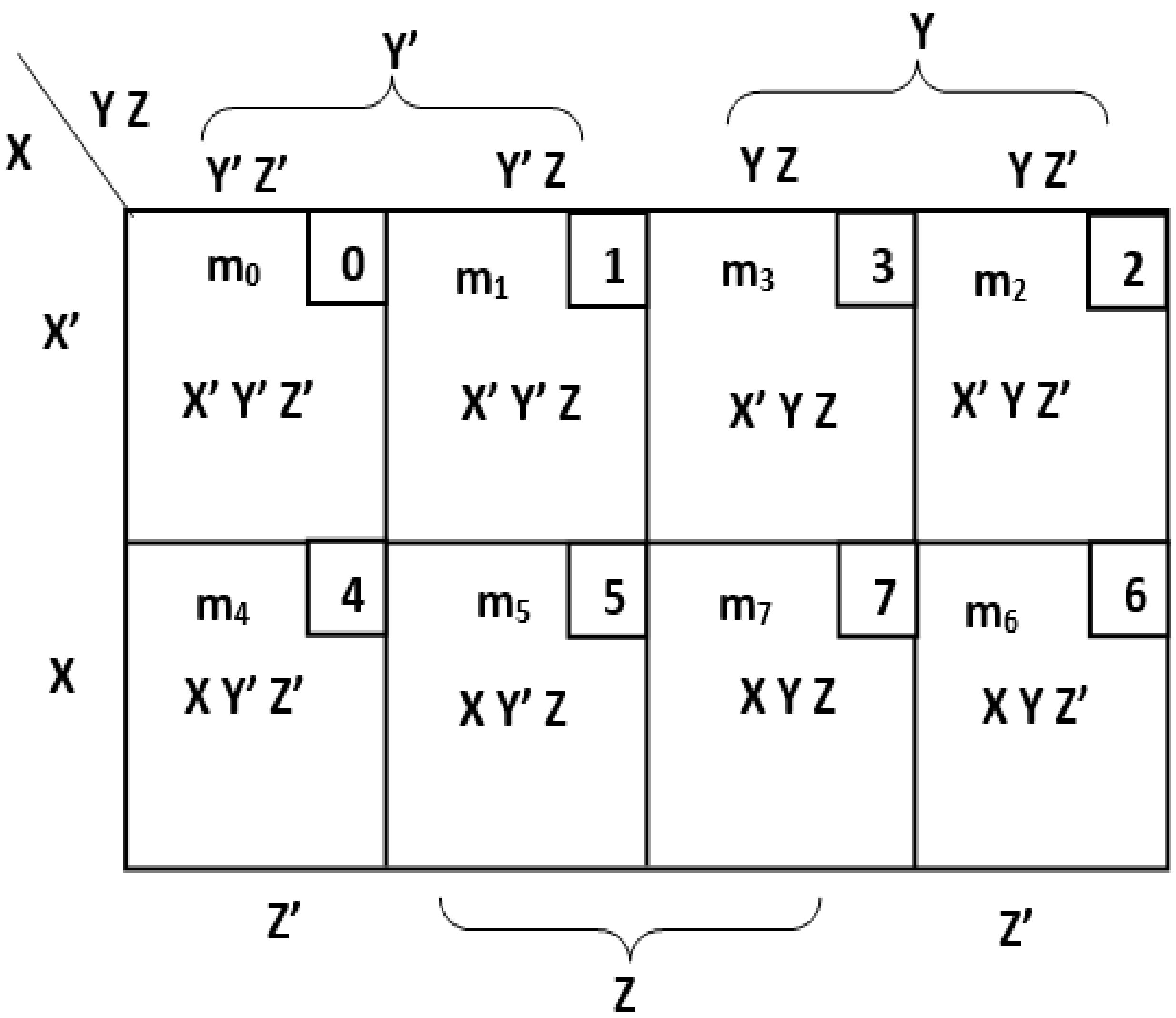
1 = 0 1 = A’ B

3 = 1 1 = A B



Therefore; F = Loop1 + Loop2 = A’ + B

Three Variable K-Map: there are eight minterms for three variable K-Map (i.e. $2^n = 2^3 = 8$). Hence map consists of eight squares for each minterm.



Simplify: F= X’Y’Z +X’YZ’ + XY’Z’+XY’Z using k-map.

Solution;

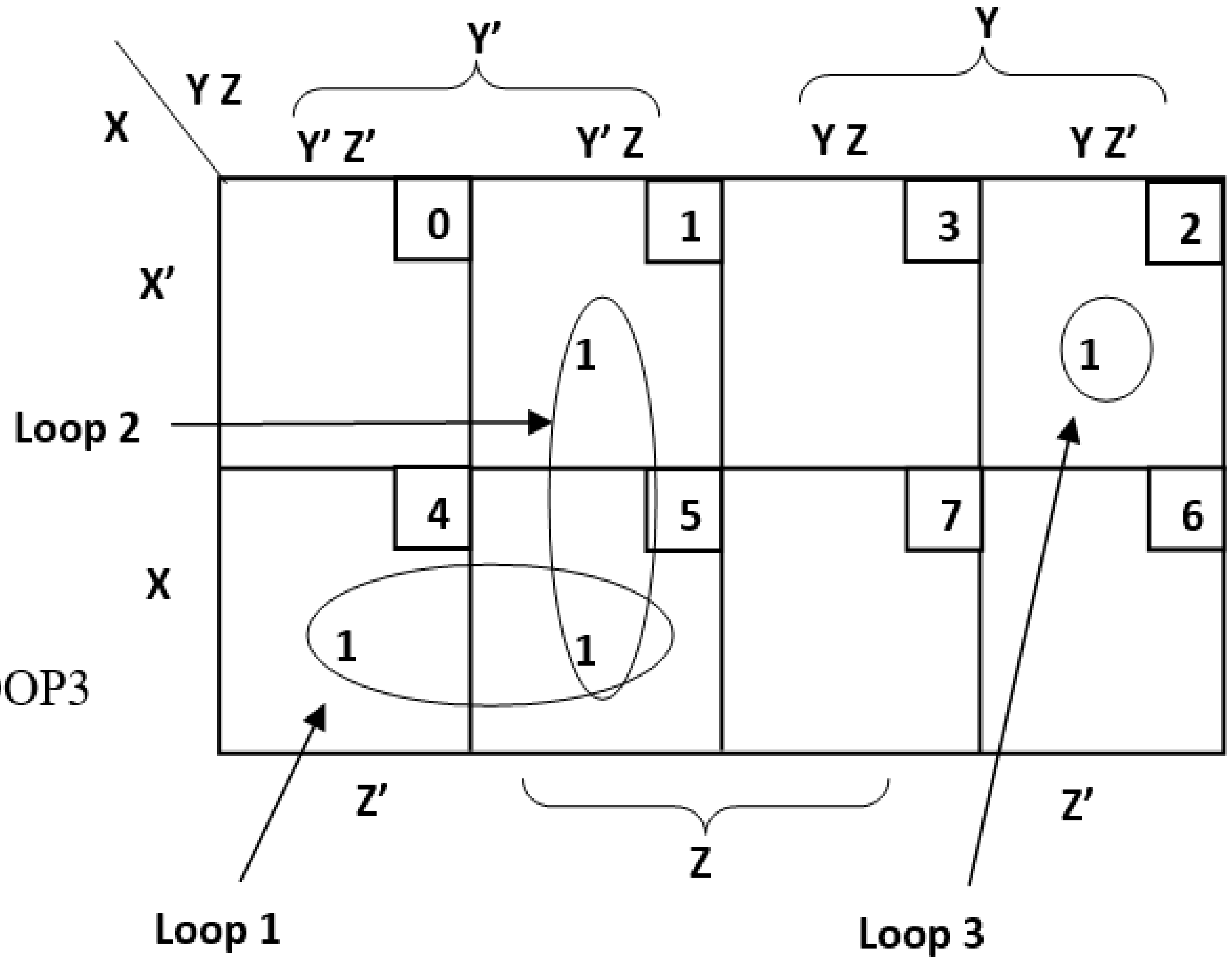
Here,

X’ Y’ Z = 0 0 1 = 1

X’ Y Z’ = 0 1 0 = 2

X Y’ Z’ = 1 0 0 = 4

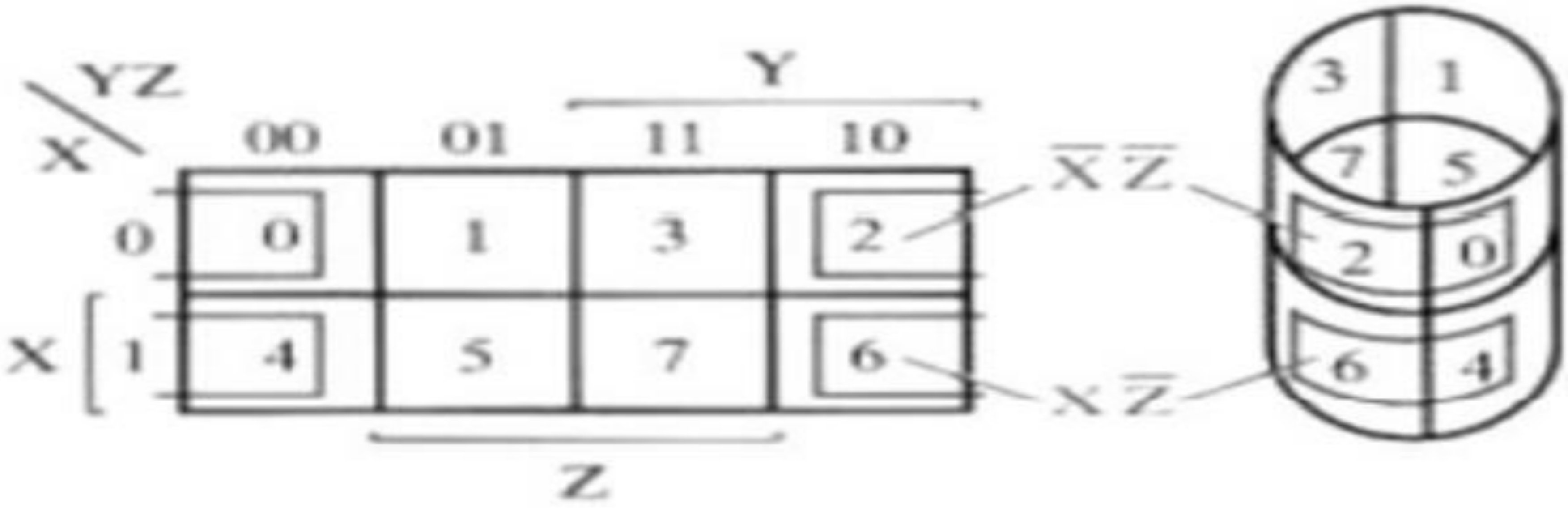
X Y’ Z = 1 0 1 = 5



Therefore, F = LOOP1 + LOOP2 + LOOP3

= X Y’ + Y’ Z + X’ Y Z’

Adjacent loop:

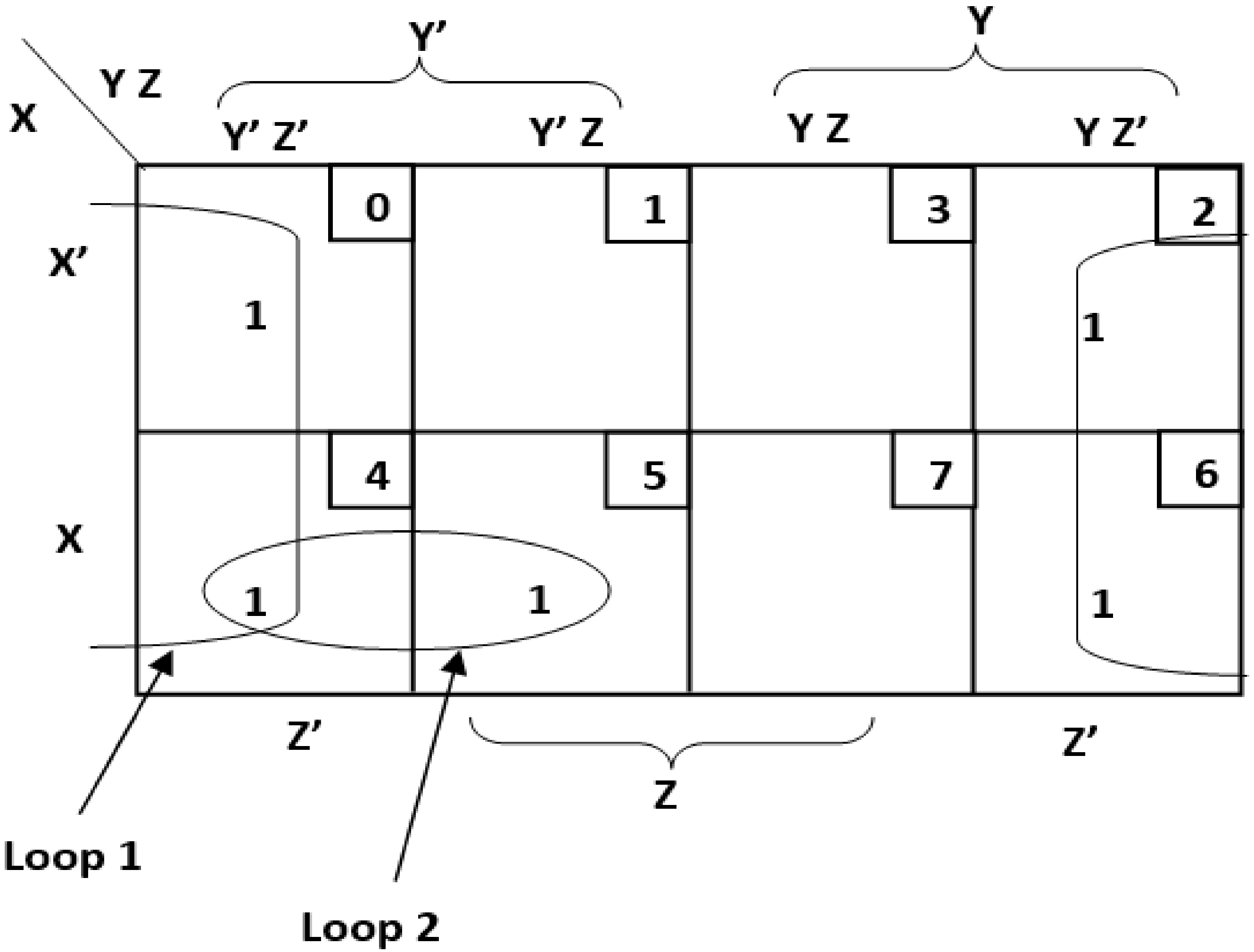


(a) (b)

Minterm adjacencies are circular in nature.
This figure shows Three-Variable Map in Flat and on a Cylinder to show adjacent squares.

Example: Simplify: $F(X, Y, Z) = \sum(0, 2, 4, 5, 6)$

Taking adjacent loop of minterms 0, 2, 4 and 6

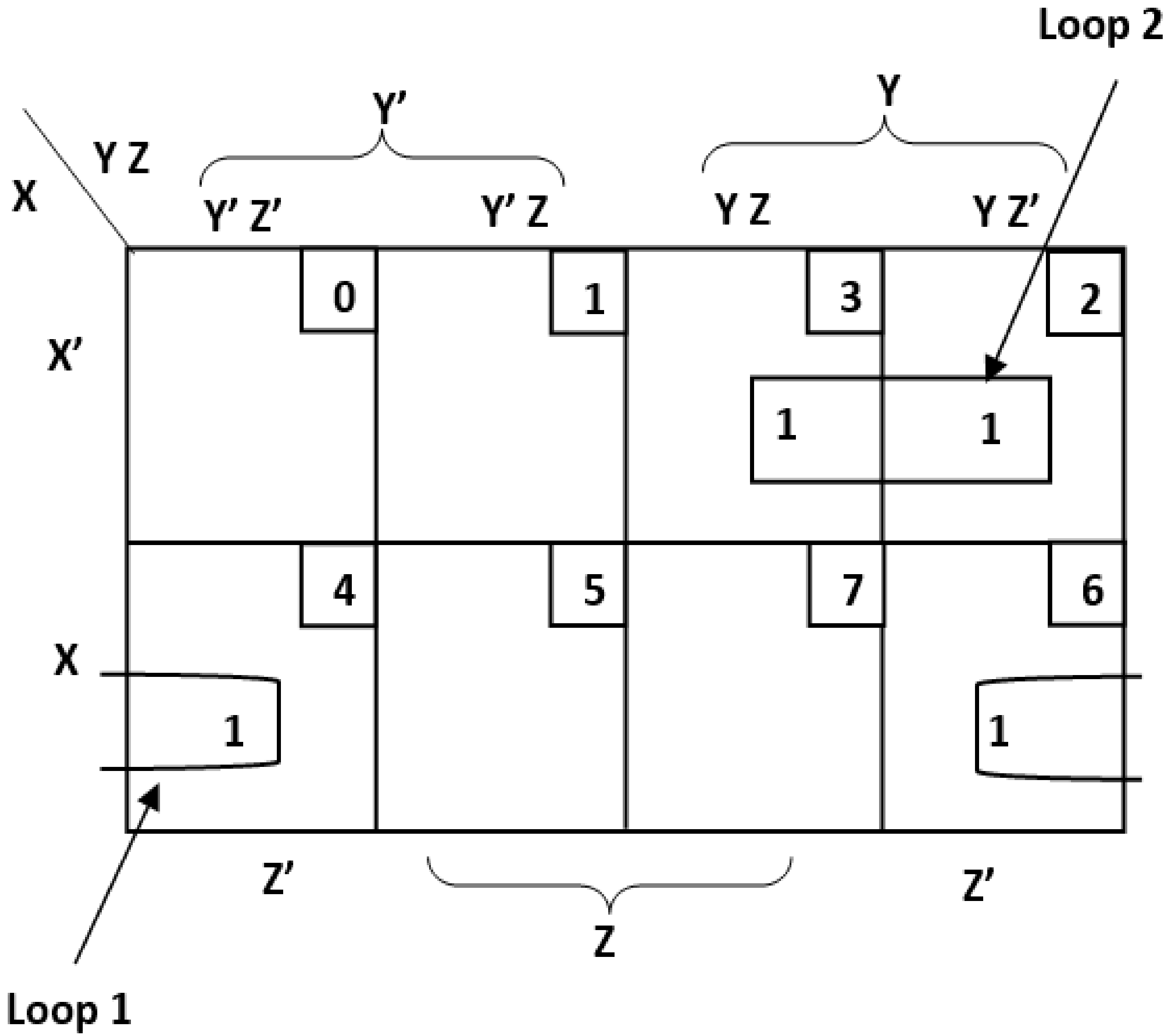


$F = Z' + X\,Y'$

Example: Simplify: $F = X\,Y\,Z' + X\,Z' + X'\,Y + X'\,Y\,Z'$

$XZ'(Y+Y') = XYZ' + XY'Z'$

$X'Y(Z+Z') = X'YZ + X'YZ'$



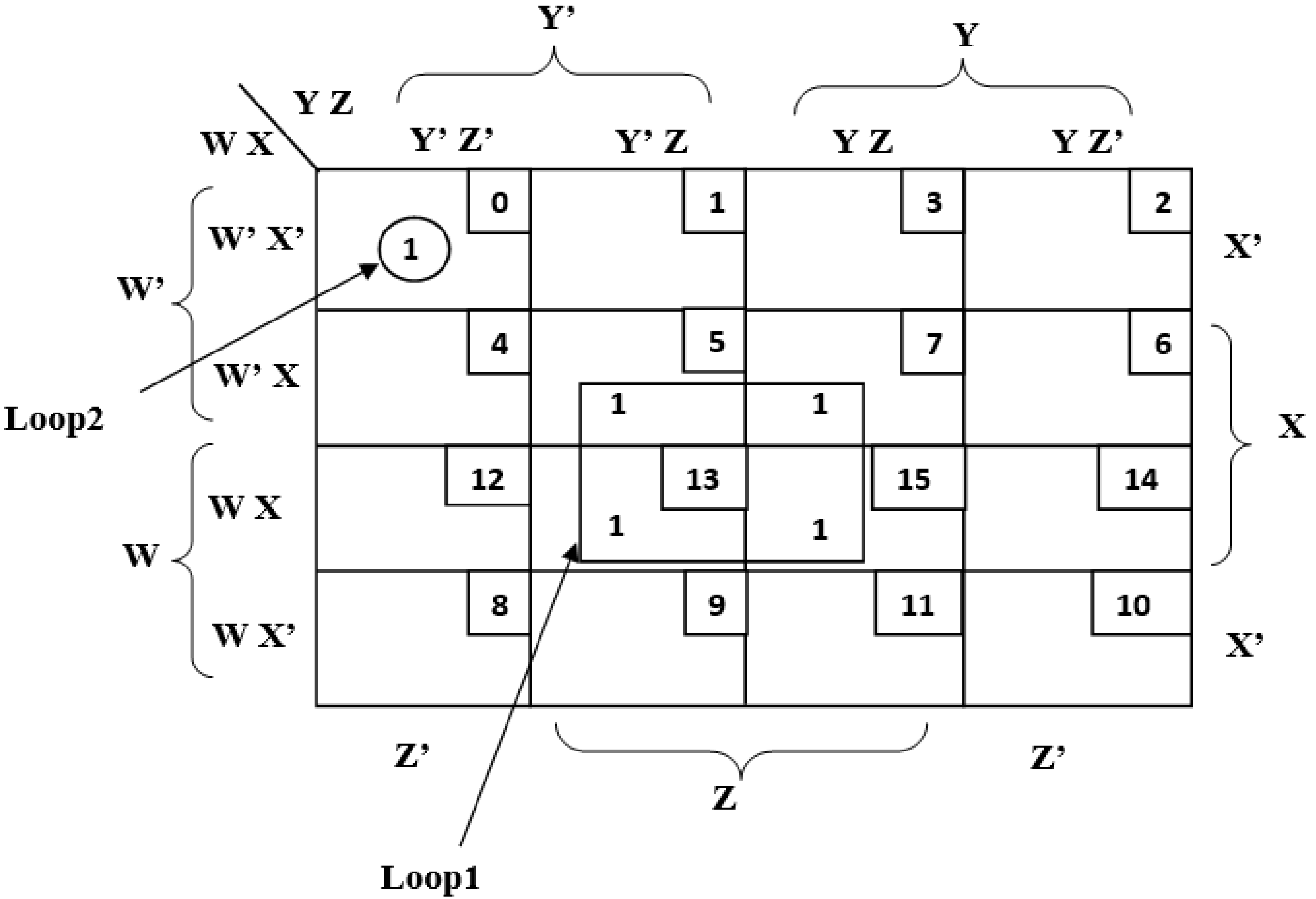
$F = \text{Loop1} + \text{Loop2}$

$= X\,Z' + X'\,Y$

Four Variable K-Map: There are 16 minterms for four variable k-Map (i.e. $2^n = 2^4 = 16$). Hence map consists of 16 squares for each minterm.

WX \ YZ	Y' Z'	Y' Z	Y Z	Y Z'
W' X'	0 m_0 W' X' Y' Z'	1 m_1 W' X' Y' Z	3 m_3 W' X' Y Z	2 m_2 W' X' Y Z'
W' X	4 m_4 W' X Y' Z'	5 m_5 W' X Y' Z	7 m_7 W' X Y Z	6 m_6 W' X Y Z'
W X	12 m_{12} W X Y' Z'	13 m_{13} W X Y' Z	15 m_{15} W X Y Z	14 m_{14} W X Y Z'
W X'	8 m_8 W X' Y' Z'	9 m_9 W X' Y' Z	11 m_{11} W X' Y Z	10 m_{10} W X' Y Z'

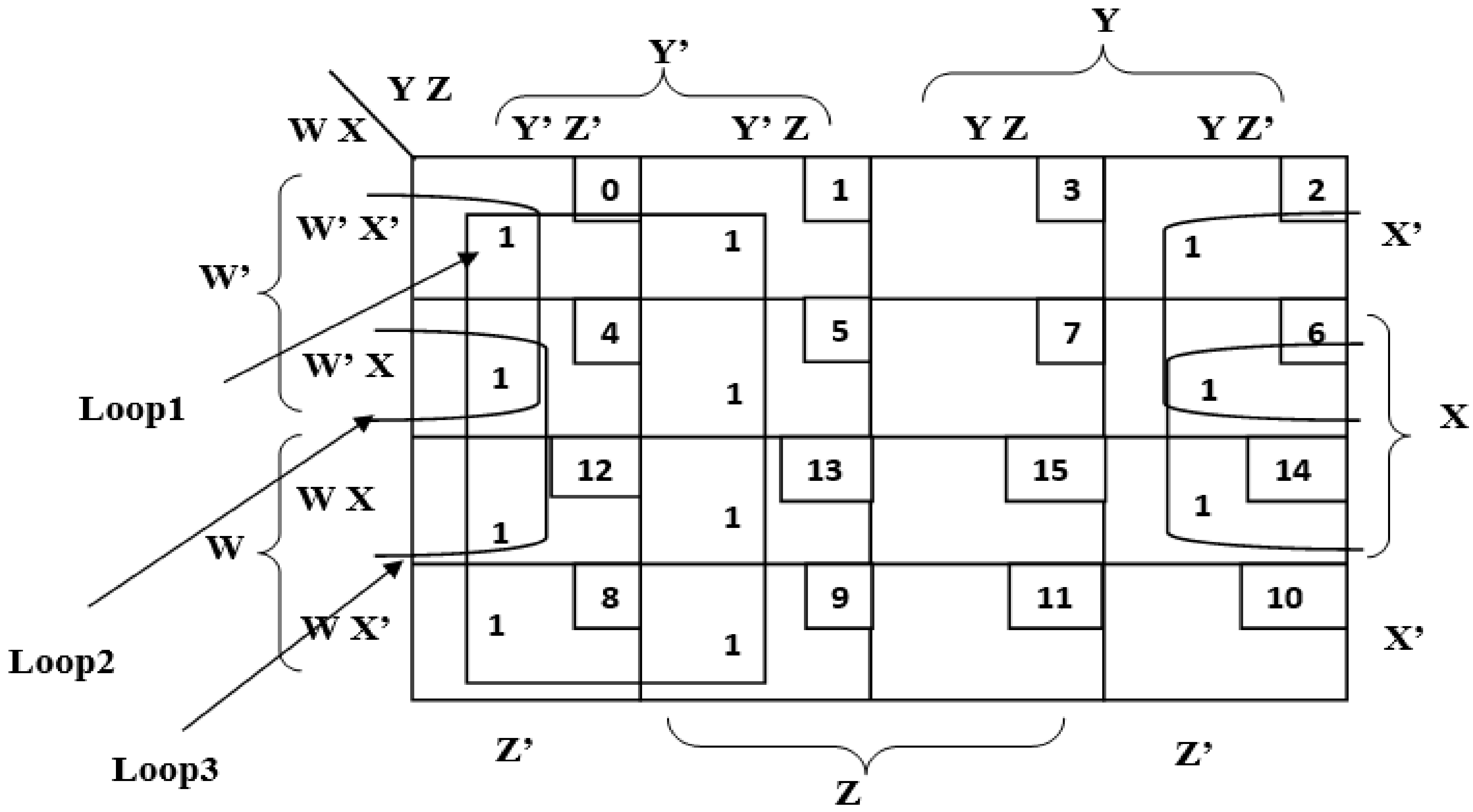
Example: Simplify: F= W'XY'Z + WXY'Z + W'XYZ + WXYZ + W'X'Y'Z'



F = Loop1 + Loop2

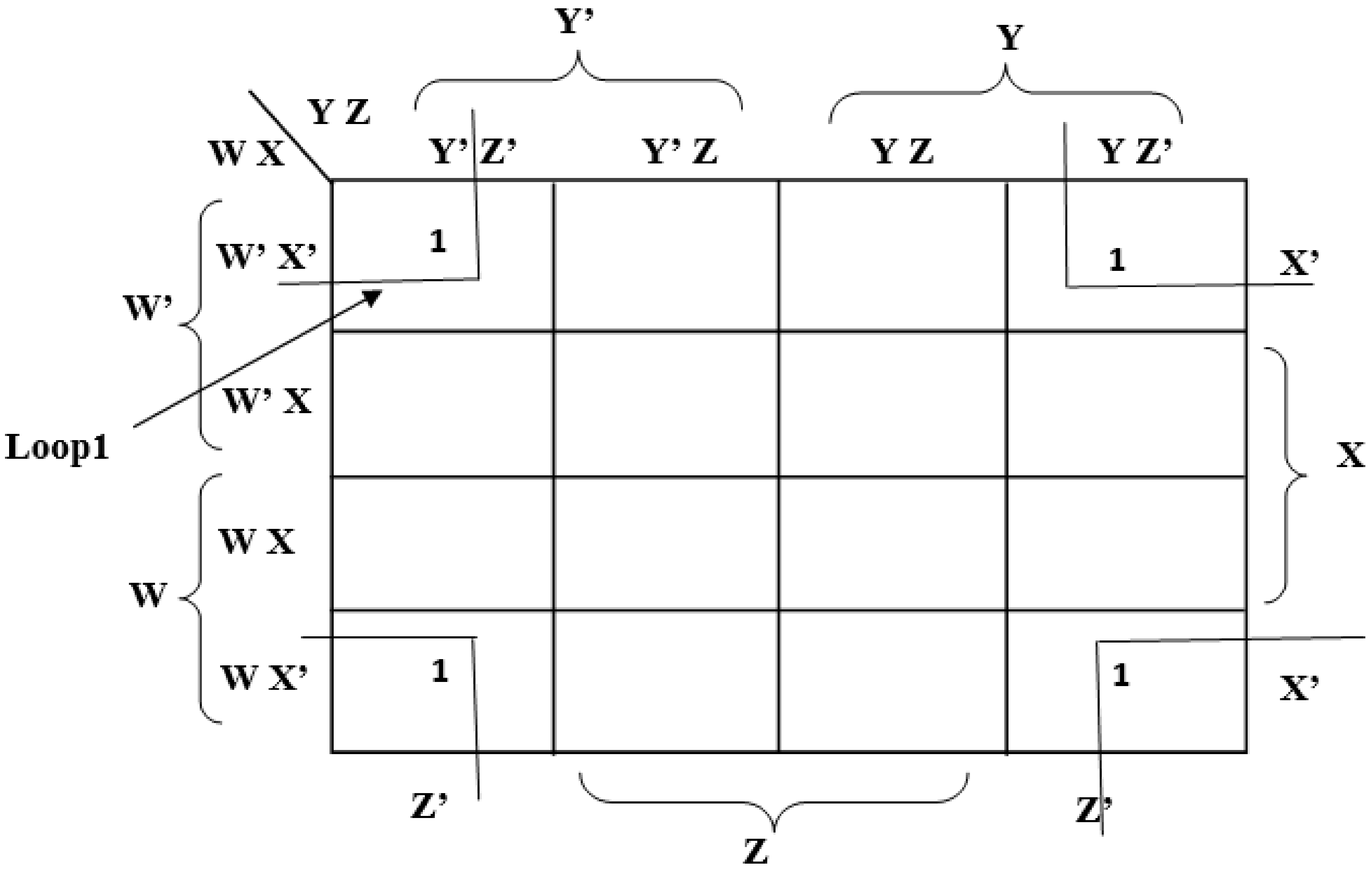
= X Z + W' X' Y' Z'

Example: Simplify: F(W, X, Y, Z) = $\sum(0, 1, 2, 4, 5, 6, 8, 9, 12, 13, 14)$



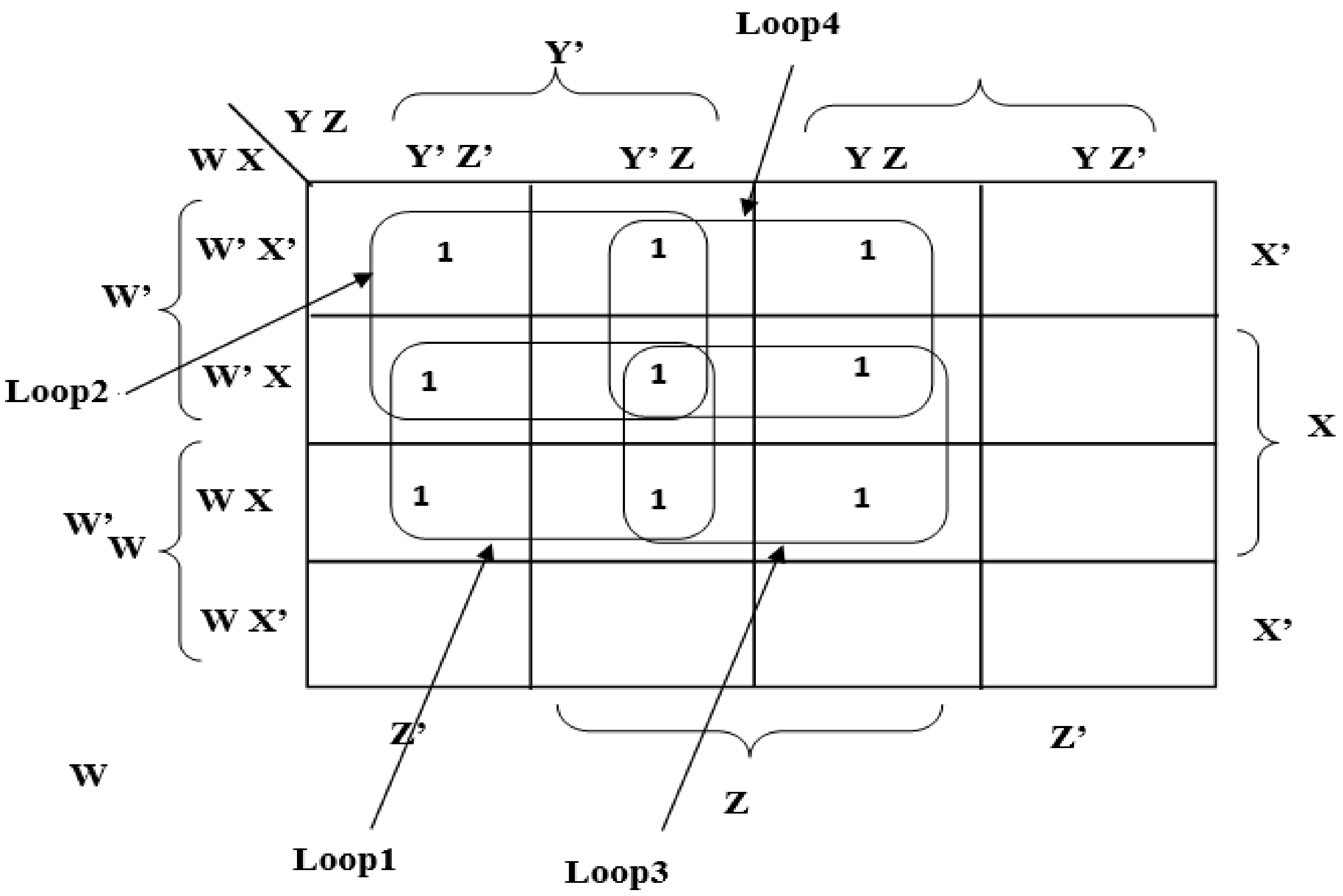
F = Y' + W' Z' + X Z'

Example: F = W'X'Y'Z' + W'X'YZ' + WX'Y'Z' + WX'YZ'



F= X' Z'

Example: F = W'X'Y'Z' + W'X'Y'Z + W'X'YZ + W'XY'Z' + W'XY'Z + W'XYZ + WXY'Z' + WXY'Z + WXYZ



F= XY' + W' Y' + XZ + W' Z