Unit – 4
Combinational Logic

Combinational Logic circuit Implementation:

Combinational Logic Circuit: A combinational circuit is one in which the state of the output at any instant is entirely determined by the states of the inputs at that time. The output occurs immediately after a slight propagation delay once the input is given.

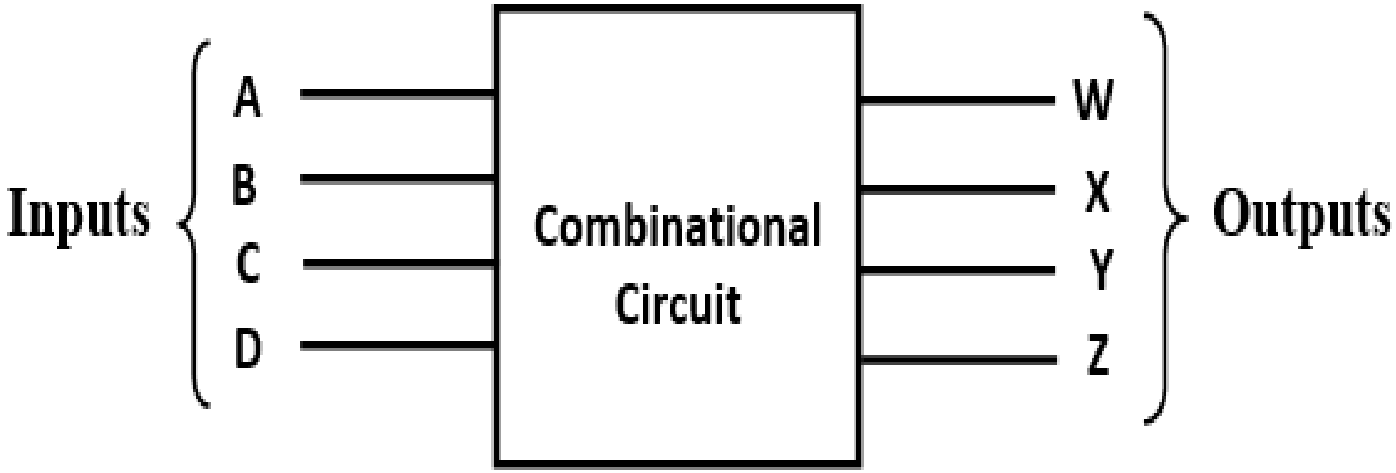
There is no memory in combinational circuits. There will be 2^n combinations of input variable for n inputs. The output will be different for each input combination. Adders, subtractors, decoders, encoders, XOR, XNOR gates, Multiplexer, De-multiplexer etc.

Combinational circuit design procedure:

1) Problem definition. That is problem is stated.
2) Determine the number of input and output variables.
3) Assign each input and output variable with letter symbol.
4) Find the relationship between input and output variables using truth table and logic functions.
5) Simplify the logic function using K-map or Boolean algebra.
6) Draw logic diagram for the simplified logic expression.

Example: Design a combinational circuit with four input lines that represent a decimal digit in BCD and four output lines that generate the 9's compliment of the input digit.

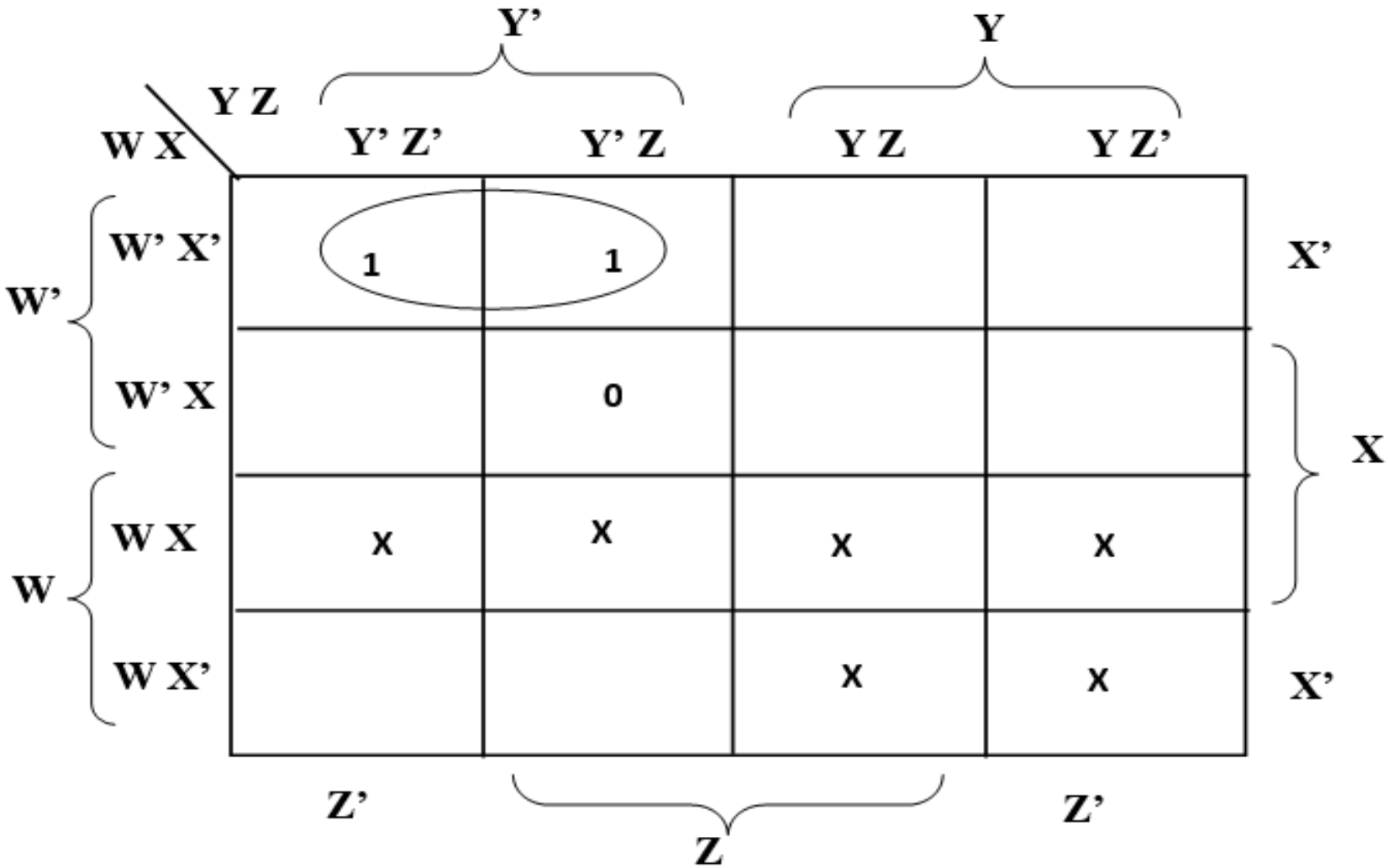
Solution:



Truth table:

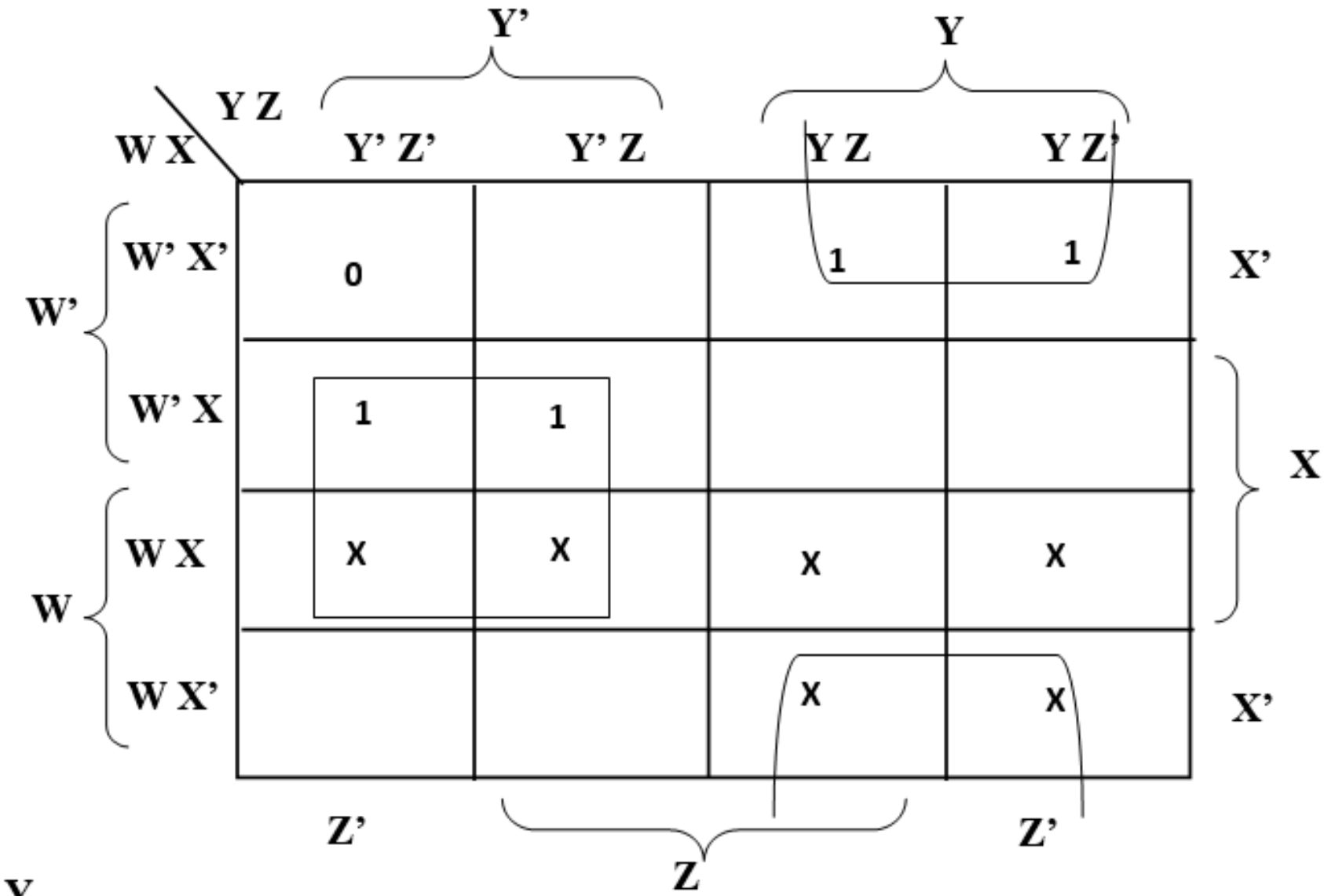
Inputs		Outputs
Decimal	BCD	9's complement
	W X Y Z	A B C D
0	0 0 0 0	1 0 0 1
1	0 0 0 1	1 0 0 0
2	0 0 1 0	0 1 1 1
3	0 0 1 1	0 1 1 0
4	0 1 0 0	0 1 0 1
5	0 1 0 1	0 1 0 0
6	0 1 1 0	0 0 1 1
7	0 1 1 1	0 0 1 0
8	1 0 0 0	0 0 0 1
9	1 0 0 1	0 0 0 0

K-Map for A:



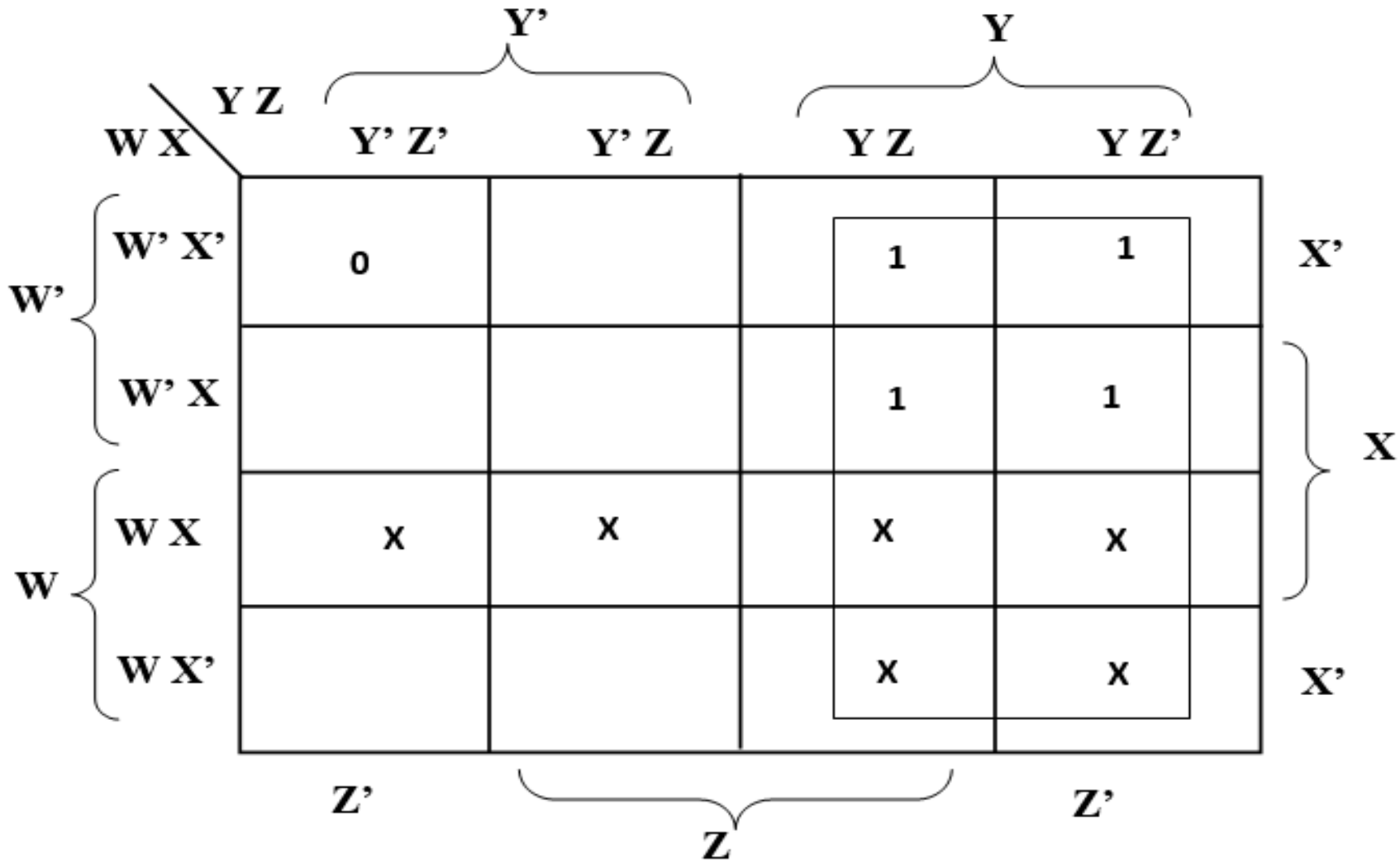
A = W' X' Y'

K-Map for B:



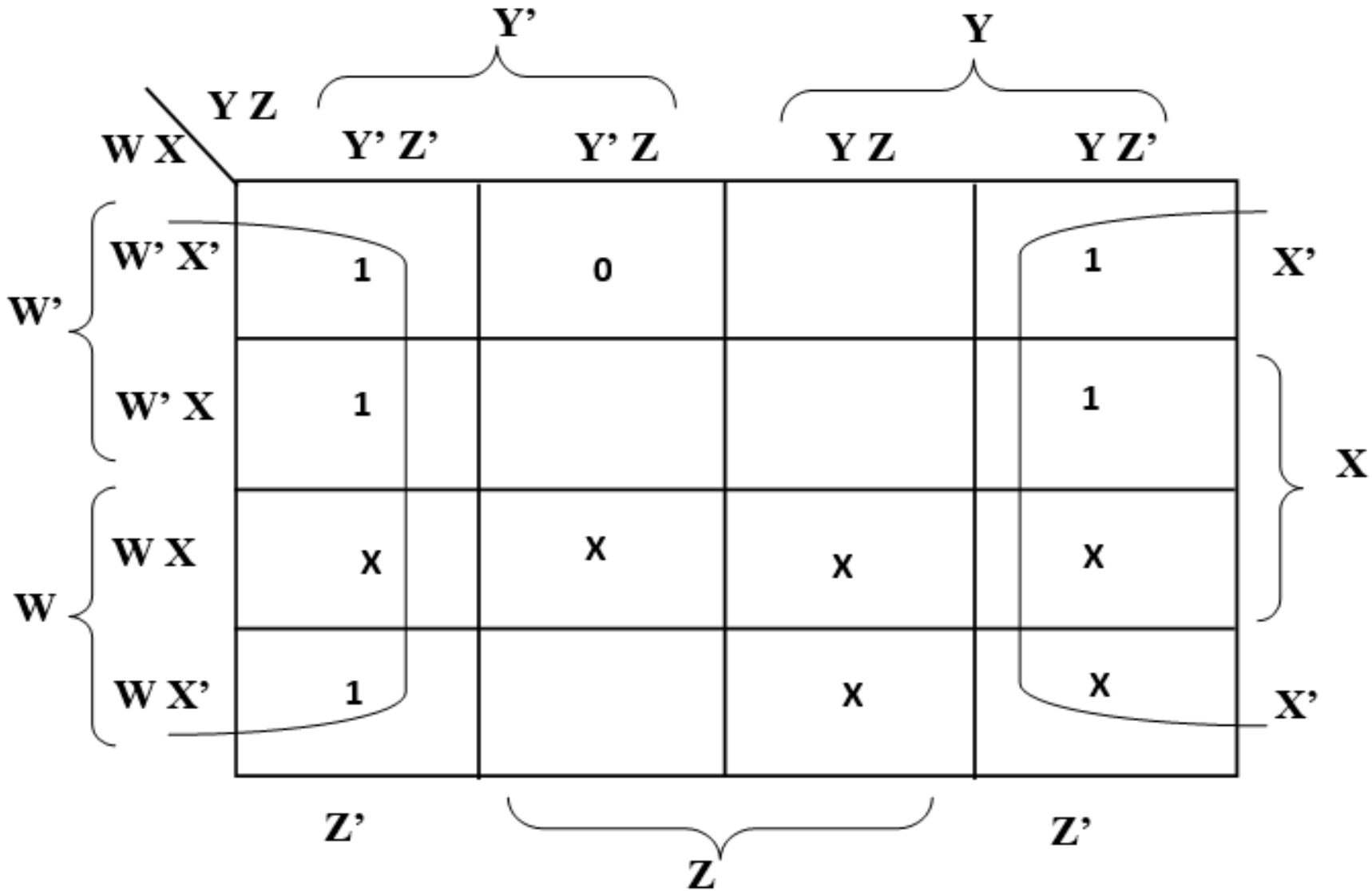
B = X Y' + X' Y

K-Map for C:



C = Y

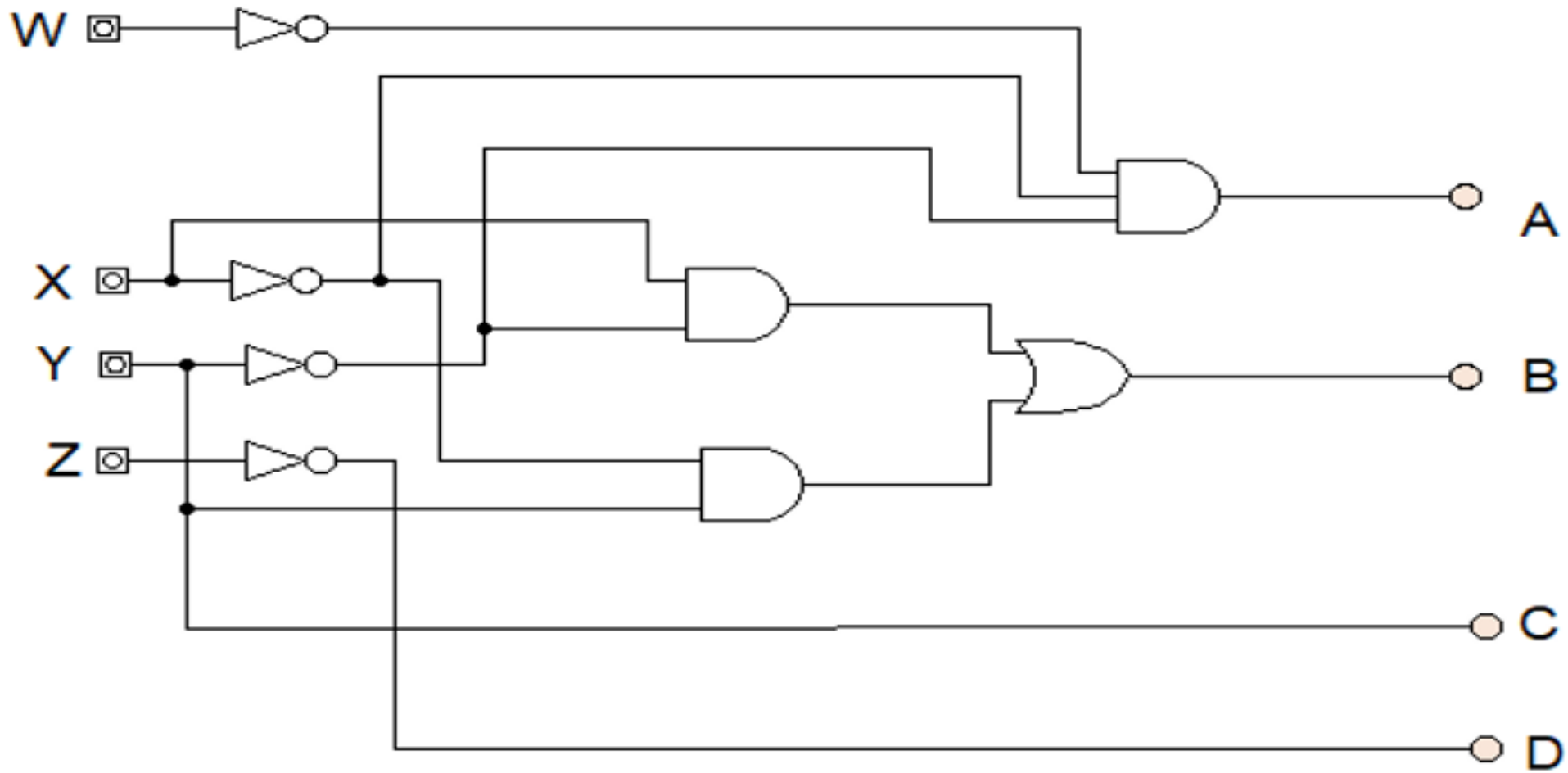
K-Map for D:



D = Z'

Logic diagram implementation:

$A = W' X' Y' \quad B = X Y' + X'Y \quad C = Y \quad D = Z'$



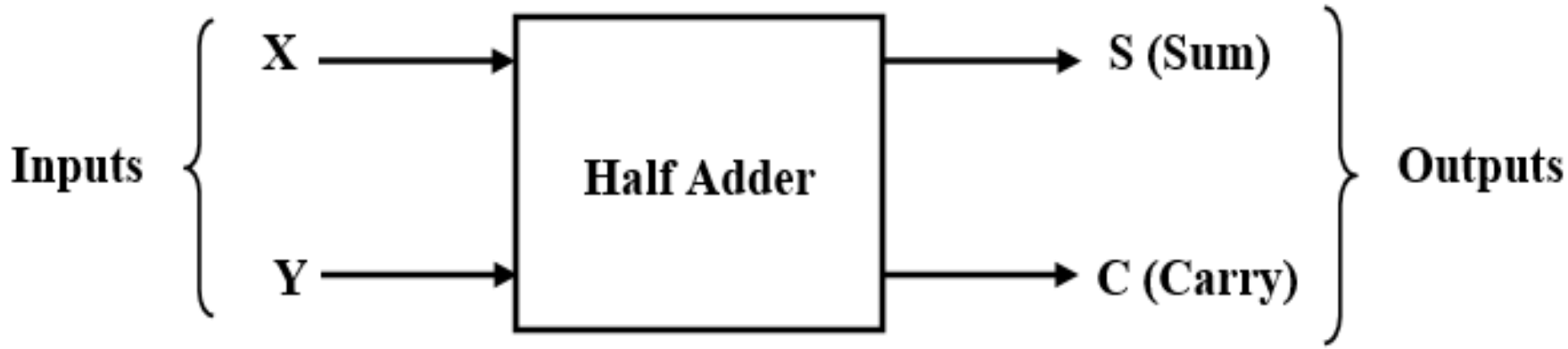
Adders: Adders are the combinational logic circuit, which is used to add two or more than two bits at a time.

Types of adders:

1) **Half Adder**
2) **Full Adder**

Half-Adder: It is a combinational logic circuit, which is used to find the sum of two binary digits at a time. Circuit needs two inputs and two outputs. The input variables designate the augend (x) and addend (y) bits; the output variables produce the sum (S) and carry (C).

Block diagram of Half-Adder:



Now we formulate a Truth table to identify the function of half-adder.

x	y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

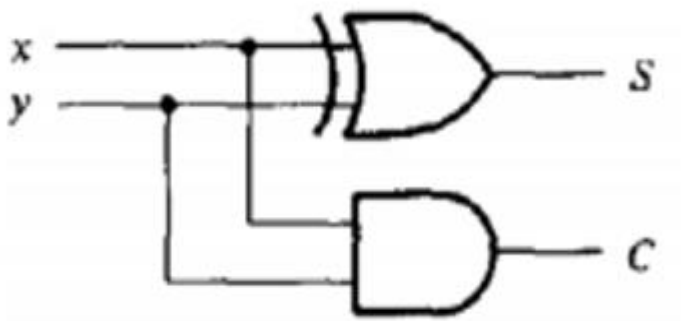
The simplified Boolean functions for the two outputs can be obtained directly from the truth table. The simplified **sum of products** expressions are:

$$S = x'y + xy' \quad \text{or} \quad S = x \oplus y$$

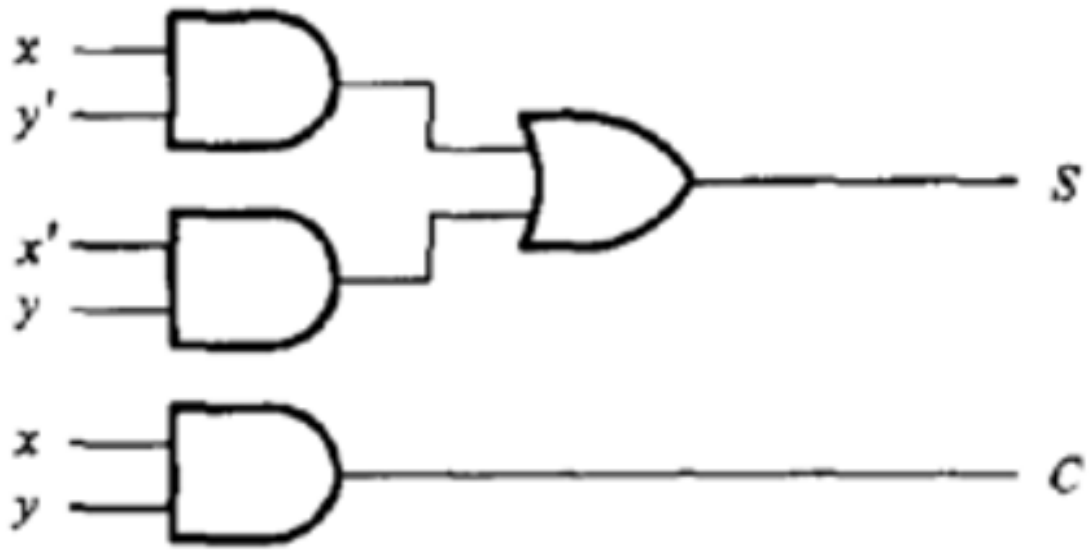
$$C = xy$$

Logic Diagram of Half Adder:

Or



(c) $S = x \oplus y$
$C = xy$

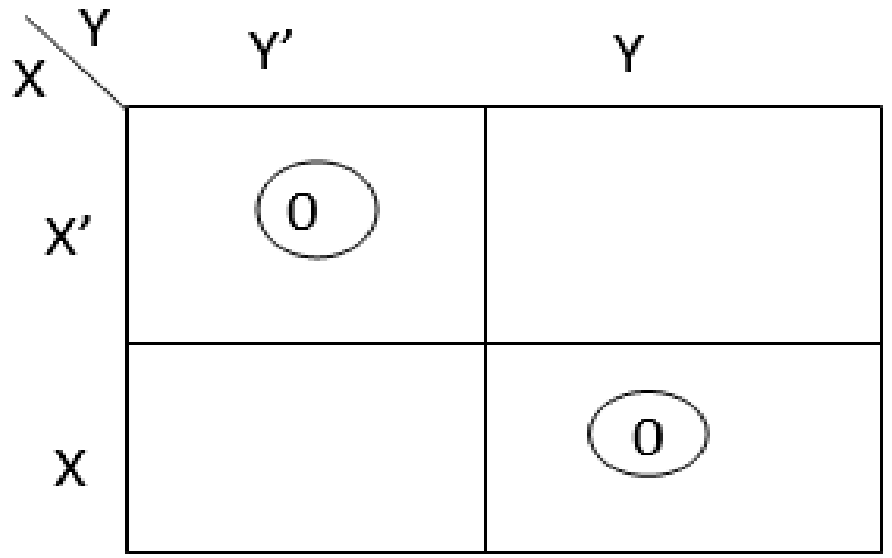


(a) $S = xy' + x'y$
$C = xy$

Ac
Go

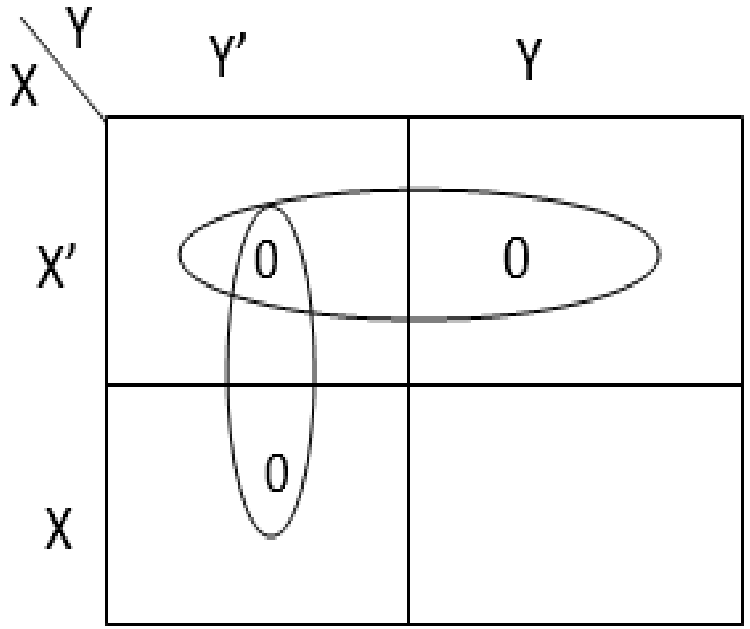
The Sum and Carry can be realized in Product of Sums form.

K-map for simplified expression in POS for Sum:



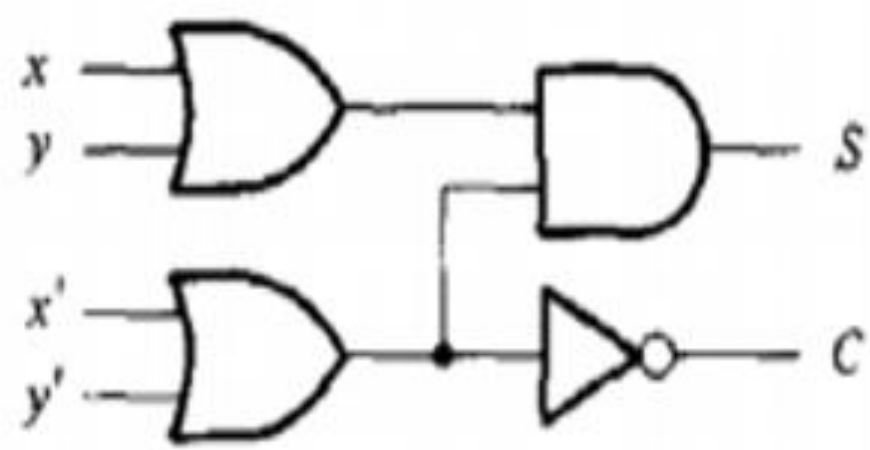
Sum (S) = (x + y) (x' + y')

K-Map for Carry:



Carry (C) = (x' + y')'

Logic diagram of Half-adder for sum and carry in POS:



$$S = (x + y)(x' + y')$$
$$C = (x' + y')'$$