Unit No. 3
Combinational Logic Design

A combinational logic circuit is a circuit whose output depends only on the present input values, not on past inputs (no memory element).
eg. Adder, subtractors, Multiplexers, Demultiplexer
encoders, decoders.

Input → $\boxed{\text{Combinational Logic Circuit}}$ → output

General Characteristics

- It doesn't have feedback element (i.e. output not fed to input.)
- Output is determined by a logic expression of ip
- These are represented using Boolean functions.

Standard Representations for logic functions

a) Boolean Function

A Boolean function maps binary input values (0 or 1) to a binary output.

eg. $Y = \overline{A}B + A\overline{B}$

b) Truth table
It is a Tabular Form listing all input combina and their corresponding outputs.

Page No.
Date

2> Product of sum (POS)

It is a unique, standard boolean expression where each sum term (a maxterm) contains every variable of the function, and the entire expression a product of these maxterms.

It's derived from truth table by identifying the rows where the output is 0.
Forming a maxterm for each (a sum of the complemented variable if the input is 0, and the uncomplemented variable if the input is 1) and then logically ANDing these maxterms together.

lets learn how to write POS form with an examp consider the following truth table.

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

- Identify maxterms for o/p '0'.
  look for the rows where the function's output Y is '0'

⇒ A, (0,0,1), (0,1,0), (1,0,1)

Page No.
Date

Don't care condition in k-map

- In digital design, some input conditions never occur or the output doesn't matter for those cases.

Instead of fixing them as 0 or 1, we mark them as X (don't care)

In simplification →
we can use them as 1's (for SOP minimization)

Or use them as 0's (for POS minimization)

or ignore them if they don't help.

- SOP example with Don't care

$F(A,B,C) = \Sigma(1,2,7) + d(0,3)$

1 at minterms m_1, m_2, m_7
Don't care at m_0, m_3

- Plot K map → (3 variables)

AB\C	0	1
00	X(m_0)	1(m_1)
01	1(m_2)	X(m_3)
11	0	1(m_7)
10	0	

use m_0, m_1, m_2, m_3 → group of 4 → gives A'
use m_3, m_7 ⇒ group of 2 → gives BC

Final expression → $\boxed{F = A' + BC}$

POS example with d Don't care

$F(A,B,C,D) = \pi(0,1,2,3,8,9) + d(10,11)$

Mark 0's and X's
0's at minterms 0, 1, 2, 3, 8, 9
X's at minterms 10, 11

Grouping
use X's and 0's to form larger groups.

4 X 4 Kmap

AB\CD	00	01	11	10
00	m_0	m_1	m_3	m_2
01	m_4	m_5	m_7	m_6
11	m_{12}	m_{13}	m_{15}	m_{14}
10	m_8	m_9	m_{11}	m_{10}

0	0	0	0
1	1	1	1
1	1	1	1
0	0	X	X

Top row (m_0, m_1, m_2, m_3) - one group of 4
Bottom-left (m_8, m_9, m_{10}, m_{11}) - another group of 4 (including dont cares at m_{10}, m_{11})

- Expressions from groups.
group 1 ($m_0 - m_3$ entire top row)
AB = 00 (so A=0, B=0 → A'B')
CD varies → eliminated.

In POS form, row of 0's → factor = (A+B)
Group 2 (entire bottom row m_8, m_9, m_{10}, m_{11})
AB = 10 (so A=1, B=0 → AB')
CD in 00 - 11 range, but we use X's eliminated.
so group factor = (A'+B)
∴ F = (A+B)(A'+B)

Page No.		
Date		

4 Variable K-map

16 cells (A, B across top, C, D along side)

AB\CD	00	01	11	10
00	F_0	F_1	F_3	F_2
01	F_4	F_5	F_7	F_6
11	F_{12}	F_{13}	F_{15}	F_{14}
10	F_8	F_9	F_{11}	F_{10}

or

m_0	m_1	m_3	[illegible]
m_4	m_5	m_7	[illegible]
m_{12}	m_{13}	m_{15}	[illegible]
m_8	m_9	m_{11}	[illegible]

Gray code order is used for column (00, 01, 11, 10) and rows (00, 01, 11, 10)

Minterms (SOP)

example $m_0 = A'B'C'D'$

$m_1 = A'B'C'D$

$m_{15} = ABCD$

1) SOP example

$F(A, B, C, D) = \Sigma(0, 2, 8, 10)$

Plot 1's in K-map

place 1 at m_0, m_2, m_8, m_{10}

these cells form a rectangle of 4

$m_0 = 0000 \Rightarrow A=0, B=0, C=0, D=0$

$m_2 = 0010 \Rightarrow A=0, B=0, C=1, D=0$

$m_8 = 1000 \Rightarrow A=1, B=0, C=0, D=0$

$m_{10} = 1010 \Rightarrow A=1, B=0, C=1, D=0$

Observe constants across all four minterms

$B=0$, and $D=0$

A and C vary

$\therefore \quad B = B'D'$

Form maxterms

$(0,0,1) \rightarrow A+B+C'$

$(0,1,0) \rightarrow A+B'+C$

$(1,0,1) \rightarrow A'+B+C'$

And the maxterms $\rightarrow$

$$Y = (A+B+C') \cdot (A+B'+C) \cdot (A'+B+C')$$

K-Map representation for logic function

(K-Map $\rightarrow$ Karnaugh map)

- It is a graphical method for simplification of Boolean function.

- Advantages – Reduces complex expressions
  easy visualization
  helps minimize gates.

- A K-map is a truth table arranged in a specia 2D form.
- Adjacent cells differ in only one variable (Gray code Order)
- Group size must be a power of 2 (1, 2, 4, 8 .. etc

2 Variable K-map

- Two input variables $\rightarrow 2^2 = 4$ minterms
- The K-map has 2 rows $\times$ 2 columns = 4 cells.

A\B	0 (B=0)	1 (B=1)
0	m_0	m_1
1	m_2	m_3

$m_0 = A'B'$

$m_1 = A'B$

$m_2 = AB'$

$m_3 = AB$

Page No
Date

Simplify further

$$F = (A+B)(A'+B)$$
$$= (AA' + AB + A'B + BB)$$
$$= (0 + AB + A'B + B)$$
$$= AB + A'B + B$$
$$= B + (AB + A'B)$$

$$\boxed{F = B}$$

Page No. | Date

POS example

Solve

$F(A,B,C,D) = \Pi(1,3,9,11)$

That means the function is 0 at minterms m_1, m_3, m_9, m_{11}

Gray code for rows & columns

AB\CD	00	01	11	10
00	m_0	m_1	m_3	m_2
01	m_4	m_5	m_7	m_6
11	m_{12}	m_{13}	m_{15}	m_{14}
10	m_8	m_9	m_{11}	m_{10}

Plot 0's

$m_1 \Rightarrow (AB=00, CD=01)$

$m_3 \Rightarrow (AB=00, CD=11)$

$m_9 \Rightarrow (AB=10, CD=01)$

$m_{11} \Rightarrow (AB=10, CD=11)$

Grouping 0's

A varies $(0 \leftrightarrow 1) \rightarrow$ eliminated

C varies $(0 \leftrightarrow 1) \rightarrow$ eliminated.

So simplified factor = $B+D$

$\therefore \boxed{F(A,B,C,D) = B+D}$

3-variable K-map

- It has 8 cells (2^3)
- Gray code ordering ensures only 1 bit change beetween adjacent cells.

AB\C	0	1
00	F_0	F_1
01	F_2	F_3
11	F_6	F_7
10	F_4	F_5

Or

AB\C	0(C=0)	1(C=1)
00	m_0	m_1
01	m_2	m_3
11	m_6	m_7
10	m_4	m_5

Minterms (SOP)	Maxterms (POS)
$m_0 = A'B'C'$	$M_0 = A+B+C$
$m_1 = A'B'C$	$M_1 = A+B+C'$
$m_2 = A'BC'$	$M_2 = A+B'+C$
$m_3 = A'BC$	$M_3 = A+B'+C'$
$m_4 = AB'C'$	$M_4 = A'+B+C$
$m_5 = AB'C$	$M_5 = A'+B+C'$
$m_6 = ABC'$	$M_6 = A'+B'+C$
$m_7 = ABC$	$M_7 = A'+B'+C'$

SOP minimization (Using 1's)

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

Step 1 → Plot 1's
place 1's at cells m_1, m_3, m_5, m_7

Step 2 → Grouping
$(m_1, m_3) \rightarrow$ vertical groups $\rightarrow A'C$
$(m_5, m_7) \rightarrow$ — " — $\rightarrow AC$

1) Sum of Products (SOP)

The SOP represents minterms in the Boolean Algebra.
SOP forms are OR-ed minterms (product terms)

It is a Boolean expression where each product term (or minterm) includes every variable of the function, either in its true or complemented form

It represents the function as a sum of these minterms for which the output is 1.

let learn how to write SOP form, with an example consider following truth table,

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

- identify '1's : The o/p Y is '1' for inputs → (0,0,0), (0,1,1), (1,0,0), (1,1,0), & (1,1,1)

- write minterms → (0,0,0) → $A'B'C'$, (0,1,1) → ($A'B$
(1,0,0) → $AB'C'$, (1,1,0) → ABC', (1,1,1) → (A

- Sum the minterms →
$Y = A'B'C + A'BC + AB'C + ABC' + ABC$

$m_0 = A'B'$

$m_2 = AB'$

A\B	0	1
0	1	0
1	1	0

$$F(A,B) = m_0 + m_2$$
$$= A'B' + AB'$$

A changes from 0 to 1, B is constant = 0

$\therefore$ A is eliminated

$$\therefore \boxed{F(A,B) = B'}$$

POS minimization Example.

Problem

Simplify $F(A,B) = \pi(0,2)$

Step 1 Place 0's in k-map

$0 = 0 \ (A=0, B=0)$

$2 = 0 \ (A=1, B=0)$

$M_0 = (A+B)$: 00

$M_1 = (A+B')$ 01

$M_2 = (A'+B)$ 10

$M_3 = (A'+B')$ 11

$$F(A,B) = M_0 \cdot M_2$$
$$= (A+B) \cdot (A'+B)$$
$$= AA' + AB + A'B + BB$$

A/B	0	1
0	0	1
1	0	1

A changes to 0

B remains constant

$\therefore F = B$

but $AA' = 0$, $BB = B$

$$F = AB + A'B + B$$

Absorption law $AB + B = B$

$\therefore F = B$

eg.

A	B	$Y = \bar{A}B + A\bar{B}$
0	0	0
0	1	1
1	0	1
1	1	0

Canonical Form

Canonical form refers to a standardized representation of a mathematical or computer science object that is unique within its equivalence class.

Properties -

- Uniqueness - when two objects are equivalent, their canonical forms will be identical.
- Simplification - Converting an object to its canonical form often simplifies it, revealing its fundamental properties and making it easier to analyze.
- Standardization - Canonical forms provide a consistent, standard representation that elimina ambiguity across various contents
- Equivalence - Canonical form is defined in the context of an equivalence relation, where all equivalent objects share the same unique canonical representation.

There are two canonical forms are —

a> Sum of Products (SOP)

b> Product of sum (POS)

Minterms →

$m_0 = A'B'$

$m_1 = A'B$

$m_2 = AB'$

$m_3 = AB$

- SOP Minimization Example

1> Problem.

Simplify $F(A,B) = \Sigma(1,3)$

Step 1. Place 1's in K-map

$m_1 = 1$ $(A=0, B=1)$

$m_3 = 1$ $(A=1, B=1)$

$F(A,B) = m_1 + m_3$

$= A'B + AB$

A\B	0	1
0	0	1
1	0	1

Step 2 Group adjacent 1's

- m_1 and m_3 form a vertical group

Step 3 Write simplified expression

- In this group, $B=1$ is constant, A changes $(0 \to 1)$ so it is eliminated. B is constant $=1$

$\therefore$ $\boxed{F(A,B) = B}$

2> $F(A,B) = \Sigma(0,2)$

$m_0 = 1$ $(A=0, B=0)$

$m_2 = 1$ $(A=1, B=0)$

Page No
Date

Step 3 — Expression

$$F = A'C + AC = C$$

$\therefore$ Answer $\boxed{F = C}$

POS minimization (Using 0's)

solve

$F(A,B,C) = \Pi(0,2,4,6)$

Step 1 Plot 0's

Place 0 at m_0, m_2, m_4, m_6

Step 2 Grouping

$(m_0, m_2) \rightarrow$ vertical group $\rightarrow (A+C)$

$(m_4, m_6) \rightarrow$ vertical group $\rightarrow (A'+C)$

expression

$F = (A+C)\cdot(A'+C)$

$F = C$

$\boxed{\text{Answer} \rightarrow F = C}$