

Unit - II

(1)

Data processing circuits

Multiplexers, demultiplexers, decoder, encoder, parity generator/checker, magnitude comparator, PAL & PLA.

Combinational Logic ckt. \rightarrow

The logic circuit having n input variables and m output variables. The output is function of present value of inputs. It does not depend on previous value of o/p.

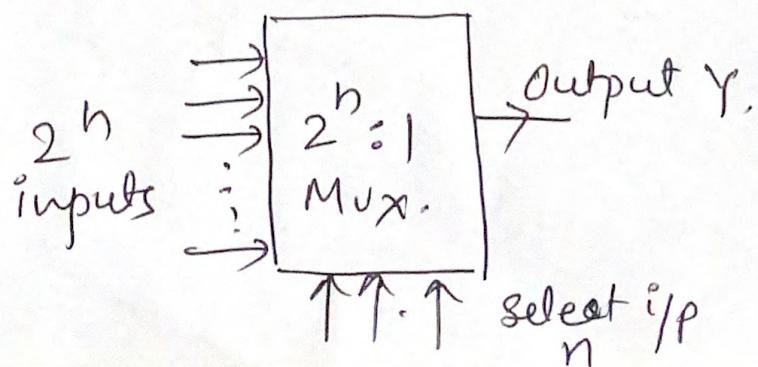
There is no clock or memory element.

Exa. Full adder, decoder, multiplexer - etc.

Multiplexer: Multiplexer is a combinational logic circuit having 2^n inputs and single output and n select input. For any select input only one input is connected to output.

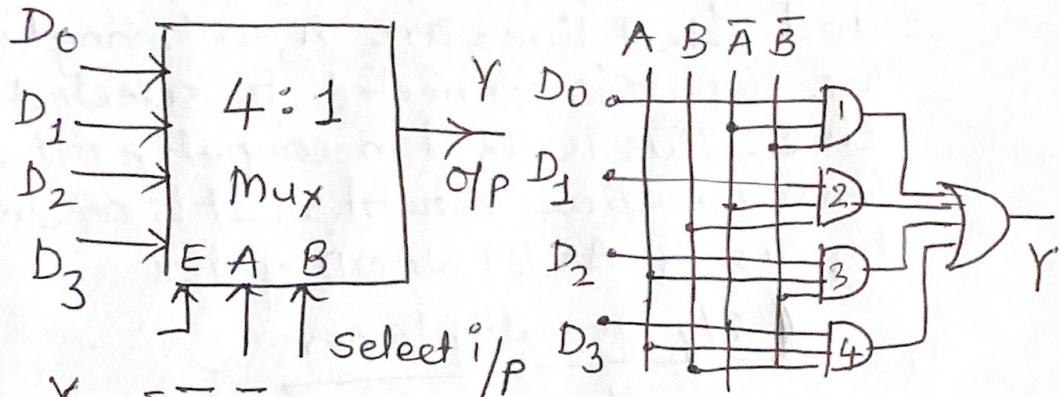
For exa. for 8:1 multiplexer with $D_0 D_1 \dots D_7$ are data inputs, A, B, C are select inputs and Y is output.

$$\text{if } ABC = 100, Y = D_4$$

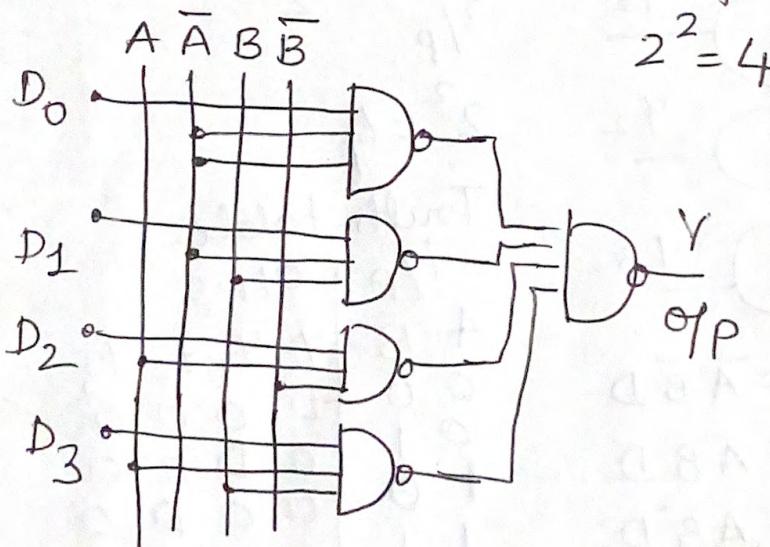


(2)

Multiplexer: 2^n inputs and one output.
Logic ckt. using AND gates
Data i/p/s



NAND realisation of 4:1 Mux $\xrightarrow{\text{SOP}}$



For $32:1$ mux. no. of select i/p $\rightarrow ?$

$2^5 = 32$, 5 select i/p. are required.

$64:1$ mux $\rightarrow ?$ $2^6 = 64$

6 select i/p/s are required.

MSI ckt.

(3)

Implementation of Boolean function using multiplexer.

MEV - map entered variable

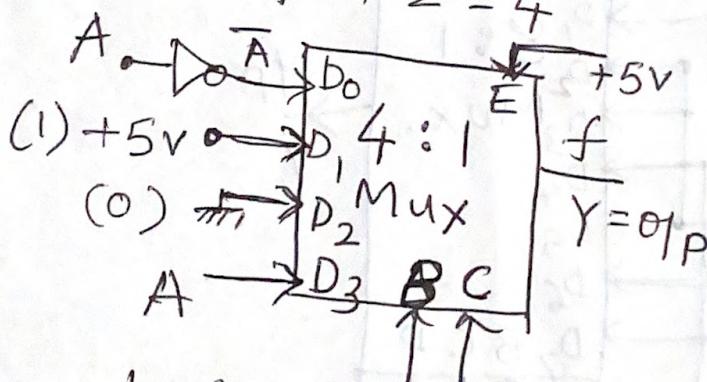
$$1) f(A, B, C) = \sum m(0, 1, 5, 7)$$

3 variables $\rightarrow 2^3 = 8$ T.T.

A as MEV: Implementation

	D ₀	D ₁	D ₂	D ₃	Table A B C Y		
\bar{A}	0	1	2	3	0	0	0
A	4	5	6	7	0	1	1
\bar{A}	1	0	A		0	1	0
A					1	0	0
					1	0	1
					1	1	0
					1	1	1

$$A + \bar{A} = 1 \Rightarrow 2^2 = 4$$



Logic diagram for implementation using 4:1 Multiplexer.

(4)

$$2^4 = 16 \Rightarrow 16:1 \text{ mux}$$

$$f(A, B, C, D) = \sum m(0, 1, 6, 9, 12, 14, 15)$$

$$4-1 \rightarrow 3 \Rightarrow 2^3 = 8:1 \text{ mux}$$

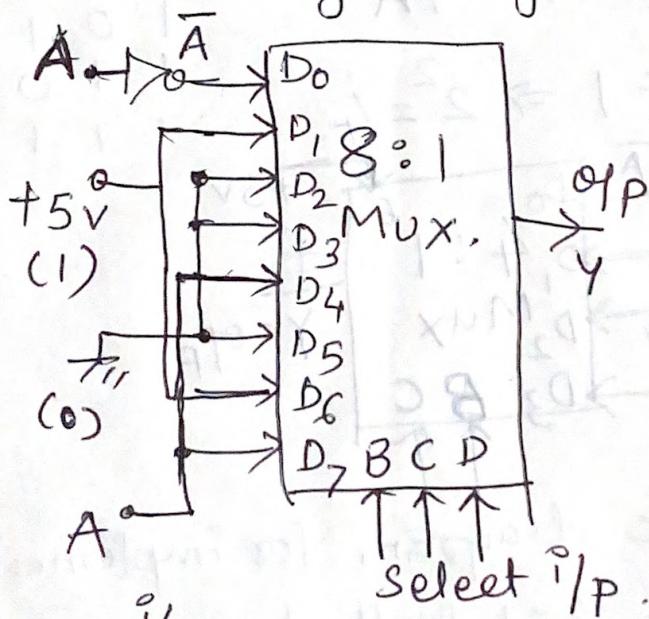
A is MEX (MSB)

Implementation

		Table $2^4 = 16$							
\bar{A}	A	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0	0	0	1	2	3	4	5	6	7
1	1	8	9	10	11	12	13	14	15
		\bar{A}	1	0	0	A	0	1	A

$$\bar{A} + A = 1$$

Logic diagram



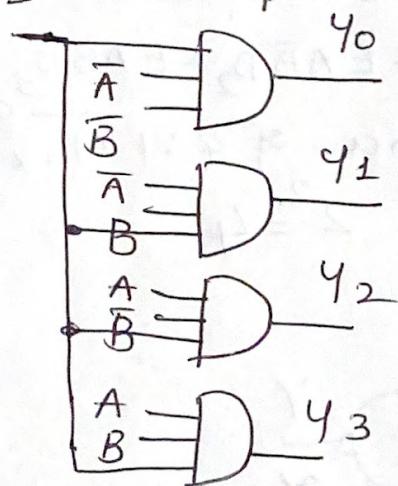
$I/P \rightarrow 0000, O/P = 1$
 $I/P \rightarrow 0010, O/P = 0$

(5)

(2)

Demultiplexer: It is a combinational logic circuit with single input and many outputs. The output lines are 2^n and select lines are n , with single input. The input is connected to selected output line. This is used in computer network and communication networks. It is constructed by using AND/NAND gates.

1 : 4 Demultiplexer



$$O/P = 4$$

Select
i/p
 $2^2 = 4$

Truth-table

i/p	O/Ps
A 0 0	y_0, y_1, y_2, y_3 D 0 0 0
A 0 1	0 D 0 0
A 1 0	0 0 D 0
A 1 1	0 0 0 D

$$y_0 = \bar{A} \bar{B} D$$

$$y_1 = \bar{A} B D$$

$$y_2 = A \bar{B} D$$

$$y_3 = A B D$$

$$AB = 01, y_1 = D$$

(6)

③ Decoder: n inputs and 2^n outputs. The input combination will select one output out of 2^n outputs.

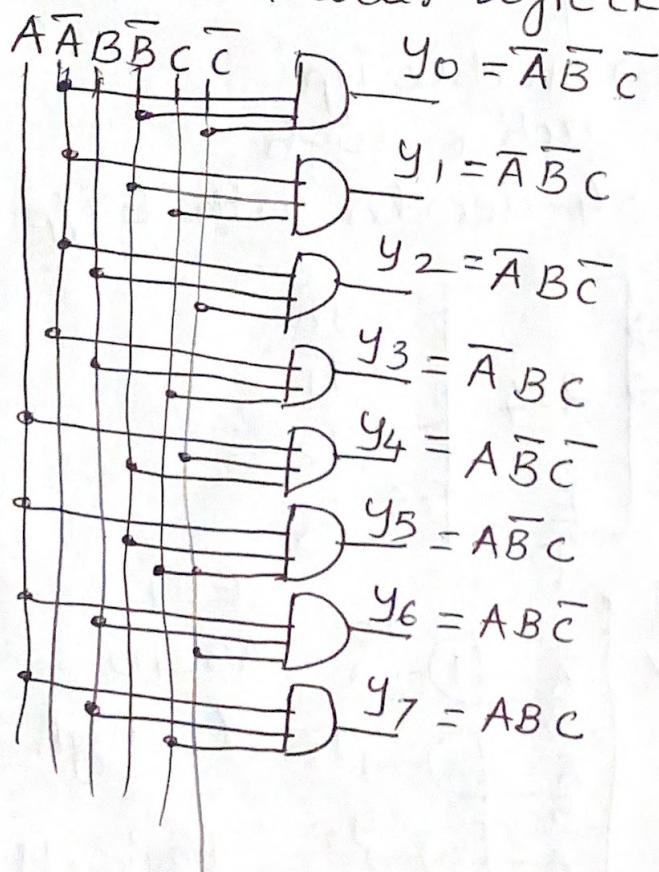
i/p $\rightarrow 001$, o/p $= y_1$ will be selected.

i/p $A B C$, o/p $\rightarrow y_0 y_1 \dots y_7$

$2^3 = 8 \Rightarrow 3:8$ decoder

AND / NAND gates.

3:8 decoder logic ckt.



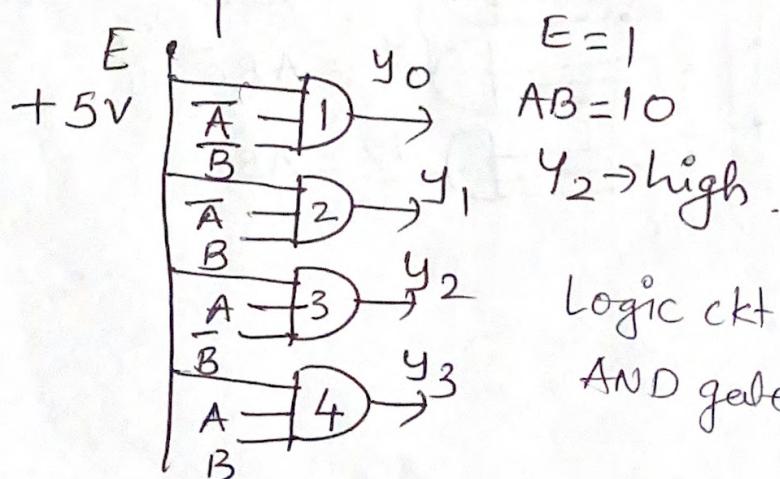
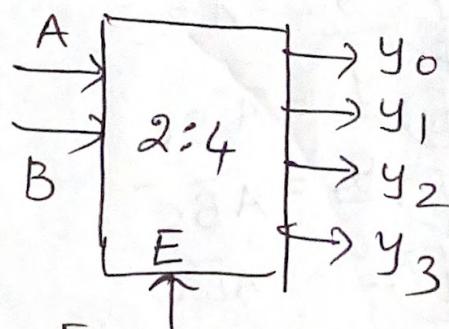
(7)

Truth-table for
3 : 8 decoder

i/p	o/p									
A	B	C	y_0	y_1	y_2	y_3	y_4	y_5	y_6	y_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

$\bar{E} \rightarrow$ enable i/p
active high

2 : 4 decoder with E i/p



Logic ckt. using
AND gates.

* Implementation of Boolean function ⑧ using decoder :

Decoder generates minterms (product term)

By using external OR gate any boolean function can be implemented in SOP form.

Implement following Boolean functions in SOP form using suitable decoder.

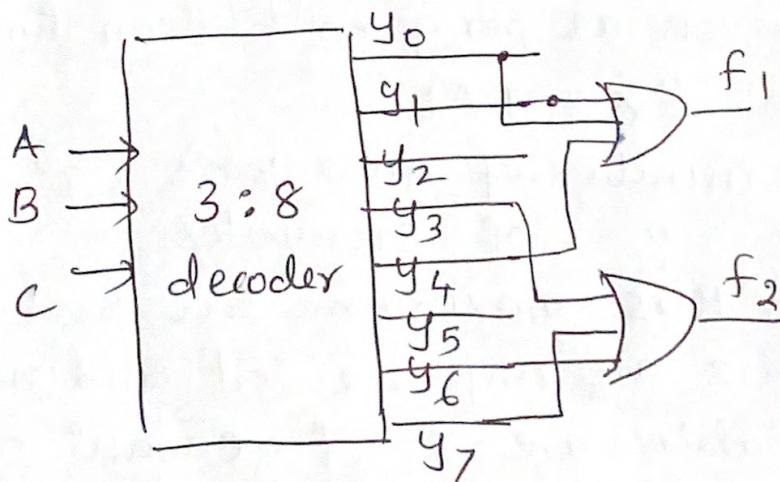
$$f_1(A, B, C) = \sum m(0, 1, 4)$$

$$f_2(A, B, C) = \sum m(3, 6, 7)$$

Use 3:8 decoder as there are 3 i/p variables.

$$f_1 = m_0 + m_1 + m_4 = \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + A\bar{B}\bar{C}$$

$$f_2 = m_3 + m_6 + m_7 = \bar{A}BC + AB\bar{C} + ABC.$$



Similarly any boolean function can be implemented. For 3 variable boolean function 3:8 decoder and for 4 variable boolean function 4:16 decoders are used.

PAL and PLA :

PAL - Programmable array logic in which AND gates are programmable and OR gates have fixed connection. AND gates generate minterms (product term) and OR gate for summation of minterms. PAL is used for implementation of SOP expressions. As there is inbuilt AND & OR gates, only programming is done before using as per SOP expression. fusible link (x) is given for each input of AND gates and the variables are connected through fusible link. PAL are available in different sizes and are chosen as per given boolean function. for exa. 8x2 PAL

8 → number of AND gates ($2^3 = 8$)
 2 → " of OR gates.

So with this maximum two boolean functions are implemented and number of variables are 3. If we want to implement 4 variable boolean function then $2^4 = 16$, 16x2, 16x4 PAL can be used. Number of inputs to OR gates are fixed. x - used to show fuse link and • - fixed connection.

(10)

* Implement following boolean functions using suitable PAL.

$$f_1(A, B, C) = \sum m(0, 3, 6, 7)$$

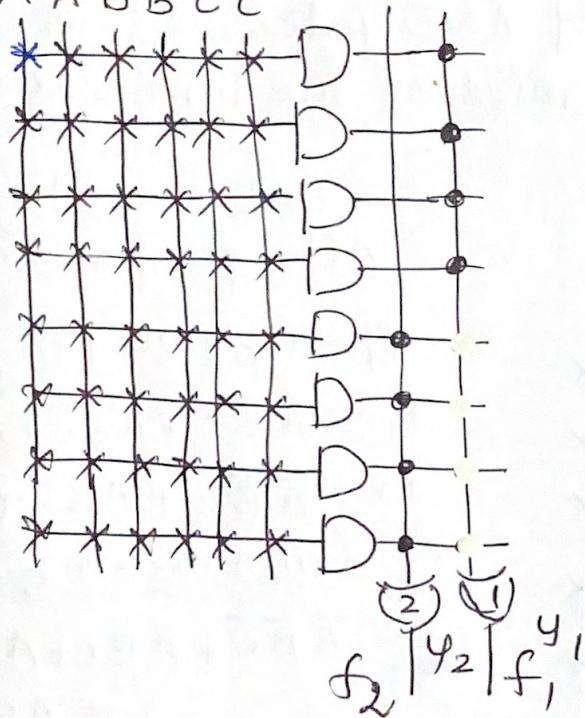
$$f_2(A, B, C) = \sum m(0, 1, 4, 7)$$

use 8×2 PAL as there are two functions of 3 variable. ($2^3 = 8$ minterms)

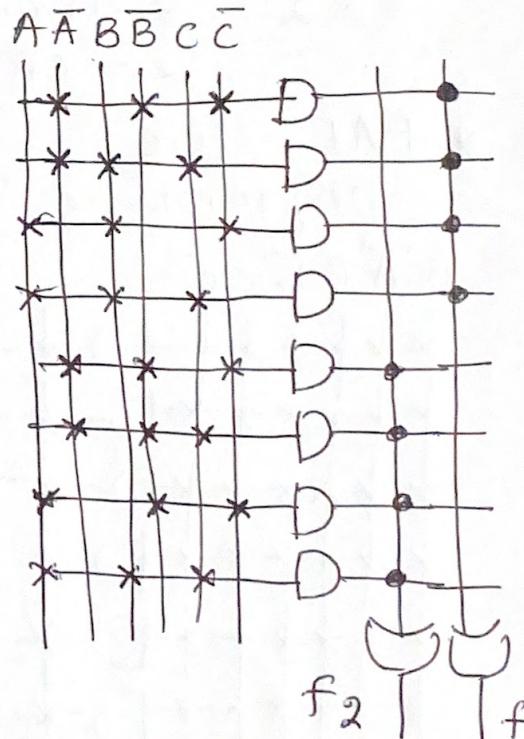
PAL before

programming

$A \bar{A} B \bar{B} C \bar{C}$



PAL after
programming \rightarrow



$$f_1 = m_0 + m_3 + m_6 + m_7$$

$$f_1 = \bar{A} \bar{B} \bar{C} + \bar{A} B C + A B \bar{C} + A B C$$

$$f_2 = m_0 + m_1 + m_4 + m_7$$

$$f_2 = \bar{A} \bar{B} \bar{C} + \bar{A} \bar{B} C + A \bar{B} \bar{C} + A B C$$

* Note: when more than 4 minterms are present in function simplification can be done

(11)

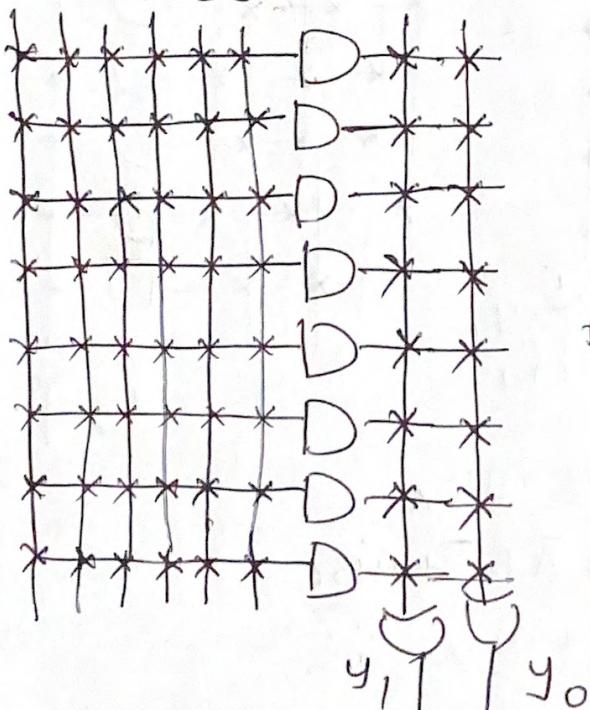
PLA = Programmable Logic Array

In PLA AND array and OR gates both are programmable. So there is no restriction on inputs to OR gates. Depending on SOP expression the input to AND gates and input to OR gate is programmed.

- * Implement given boolean functions using suitable PLA. $f_1(A, B, C) = \sum m(0, 6, 7)$
size required is $f_2(A, B, C) = \sum m(0, 3, 5, 6)$
 8×2 . $8 \rightarrow$ no. of AND gates. $2^3 = 8$
 $2 \rightarrow$ OR gates as two functions.

* PAL before programming

$A \bar{A} B \bar{B} C \bar{C}$



PAL circuit

After programming -

$$f_1 = m_0 + m_6 + m_7$$

$$f_1 = \bar{A} \bar{B} \bar{C} + A \bar{B} \bar{C} + A B \bar{C}$$

$$f_1 = \bar{A} \bar{B} \bar{C} + A B \quad (1)$$

$$f_2 = m_0 + m_3 + m_5 + m_6$$

$$f_2 = \bar{A} \bar{B} \bar{C} + \bar{A} B \bar{C} + A \bar{B} C + A B \bar{C}$$

