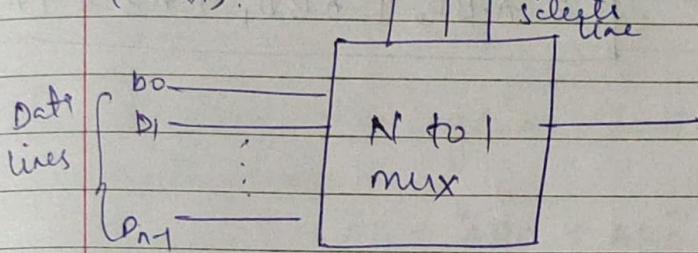


01-12-21

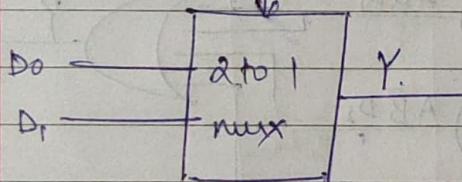
Digital circuit

- * Multiplexer \rightarrow Many to 1. (many inputs but 1 output). (mux).



$$\begin{array}{c} \checkmark \\ \text{select lines} \end{array} \boxed{2^m = n} \quad \xrightarrow{\text{data lines}}$$

\rightarrow 2 to 1 multiplex.

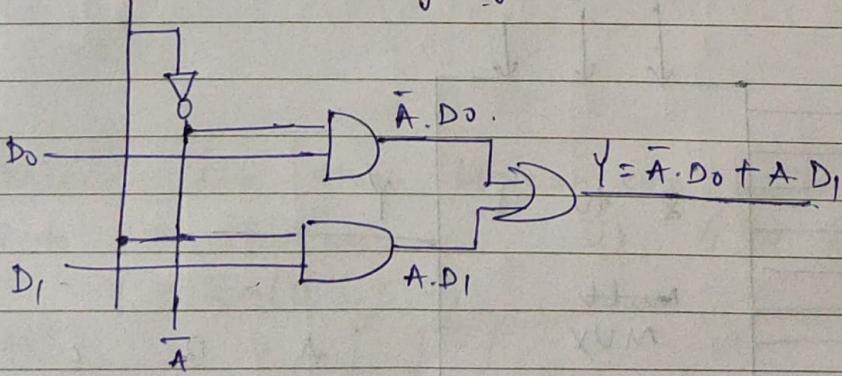


A	Y
0.	D ₀
1	D ₁

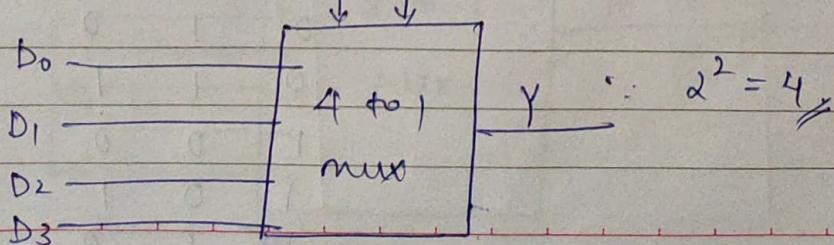
$$\begin{array}{c} \text{select line} \\ \checkmark \\ 2^1 = 2 \\ \text{data lines} \end{array}$$

$$Y = \bar{A} \cdot D_0 + A \cdot D_1$$

Realization using gates



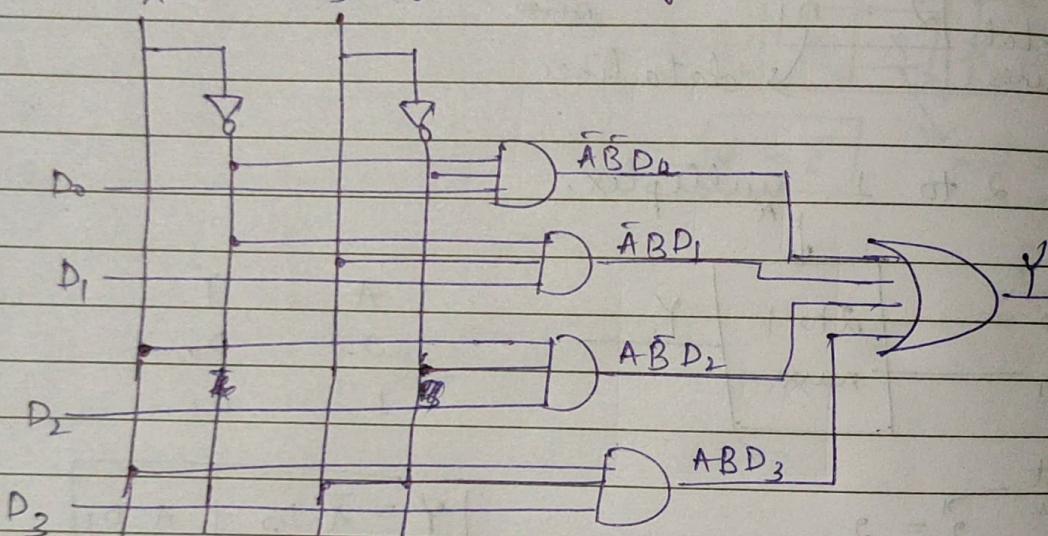
\rightarrow 4 to 1 mux.



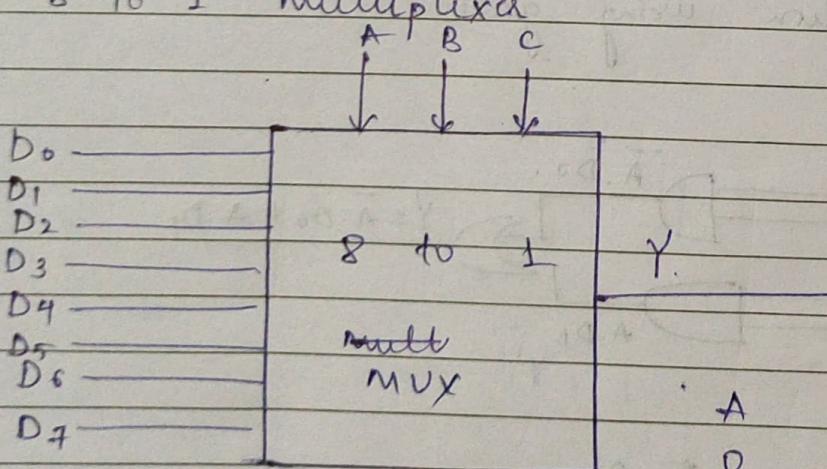
A	B	Y
0	0	D ₀
0	1	D ₁
1	0	D ₂
1	1	D ₃

$$Y = \bar{A} \cdot \bar{B} D_0 + \bar{A} B D_1 + A \bar{B} D_2 + A B D_3$$

Realization using basic gates



→ 8 to 1 multiplexer



A	B	C	Y
0	0	0	D ₀
0	0	1	D ₁
0	1	0	D ₂
0	1	1	D ₃
1	0	0	D ₄
1	0	1	D ₅
1	1	0	D ₆
1	1	1	D ₇

$$Y = \bar{A}\bar{B}\bar{C}D_0 + \bar{A}\bar{B}C\bar{D}_1 + \bar{A}B\bar{C}\bar{D}_2 + \bar{A}BC\bar{D}_3 + A\bar{B}\bar{C}D_4 \\ + A\bar{B}C\bar{D}_5 + AB\bar{C}\bar{D}_6 + ABC\bar{D}_7.$$

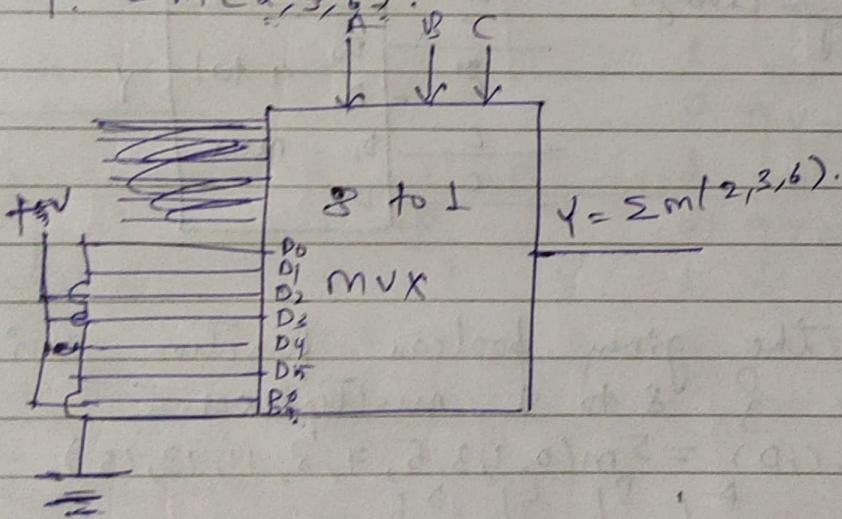
Q) Realise the given boolean exp. using 2 to 1 multiplexer.

$$Y = \bar{A}B + \bar{A}BC + ABC\bar{C}.$$

Converting to std. SOP.

$$Y = \bar{A}B(C + \bar{C}) + \bar{A}BC + ABC\bar{C} \\ = \bar{A}B\bar{C} + \bar{A}B\bar{C} + \bar{A}BC + \bar{A}BC \\ = \bar{A}B\bar{C} + \bar{A}B\bar{C} + \bar{A}BC. \\ = \Sigma m(3, 2, 6).$$

$$Y = \Sigma m(2, 3, 6).$$

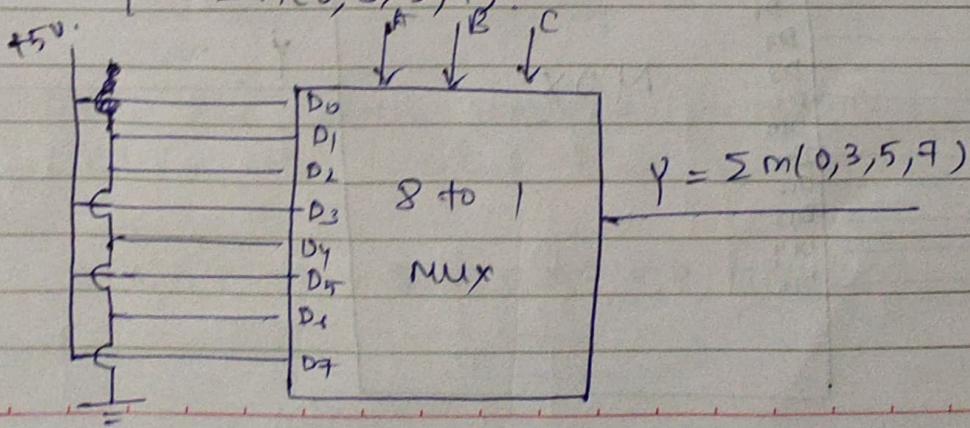


Q) Realise the given boolean exp. using 8 to

i) 8 to 1 multiplexer.

ii) 4 to 1 multiplexer

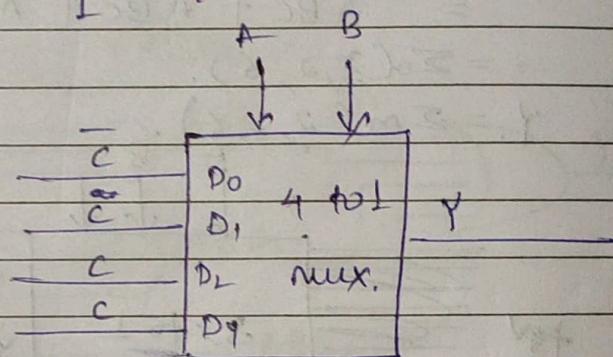
$$Y = \Sigma m(0, 3, 5, 7).$$



ii) 4 to 1 multiplexer.

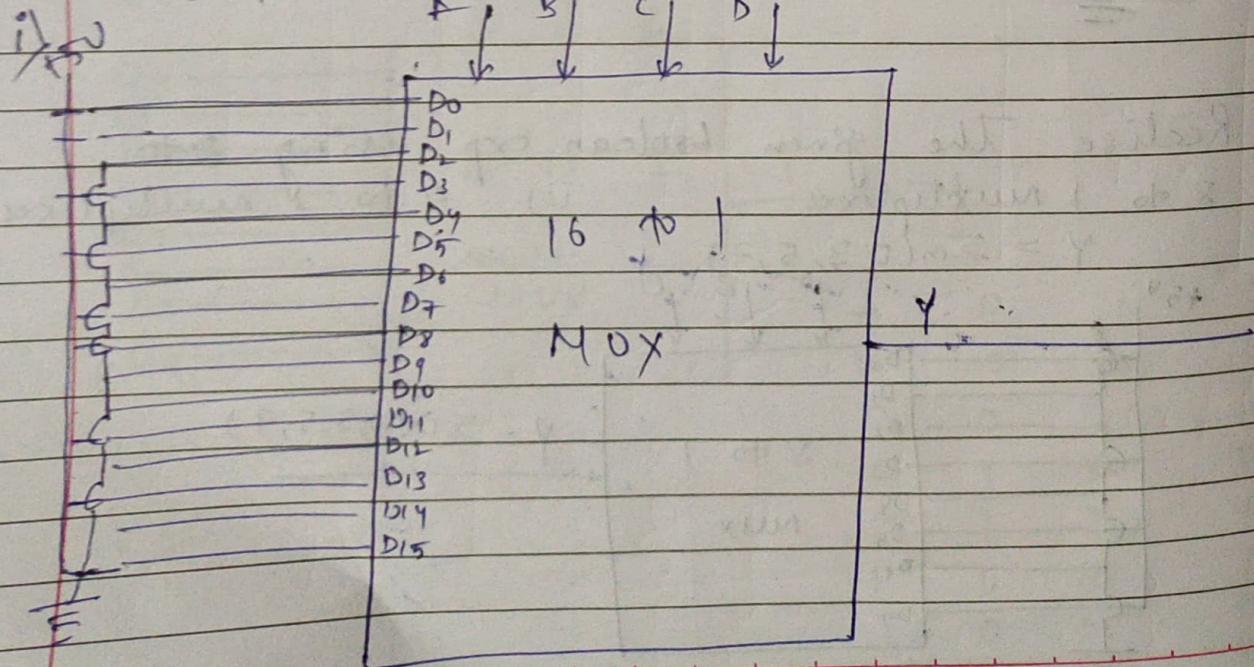
A	B	C	Y
0	0	0	1] $Y = \bar{C}$
0	0	1	0]
0	1	0	0
0	1	1	1 $Y = C$
1	0	0	0
1	0	1	1 $Y = C$
1	1	0	0 $Y = \bar{C}$
1	1	1	1

here A & B
* & C are same
only C changes



Q3) Realise the given boolean function using 16 to 1 multiplexer of 8 to 1 multiplexer.

$$Y(A, B, C, D) = \sum m(0, 1, 3, 5, 7, 8, 11, 13, 15)$$

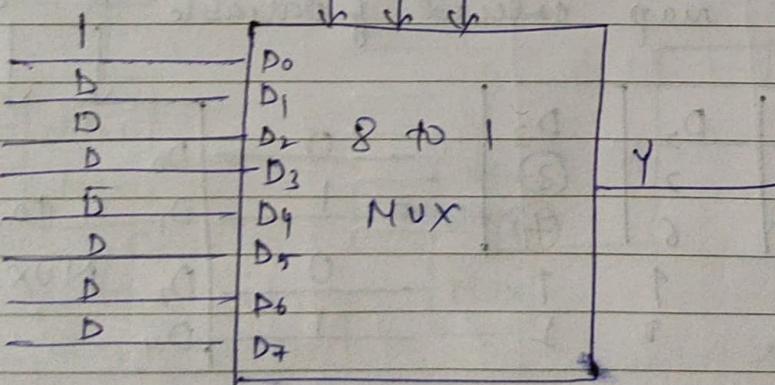


MSB

LSB

A	B	C	D	Y
0	0	0	0	1] Y = 1
0	0	0	1	1]
0	0	1	0	0] Y = D
0	0	1	1	1]
0	1	0	0	0] Y = D
0	1	0	1	1]
0	1	1	0	0] Y = D
0	1	1	1	1]
1	0	0	0	1] Y = D
1	0	0	1	0]
1	0	1	0	0] Y = D
1	0	1	1	1]
1	1	0	0	0] Y = D
1	1	0	1	1]
1	1	1	0	0] Y = D
1	1	1	1	1]

A B C



2-12-21

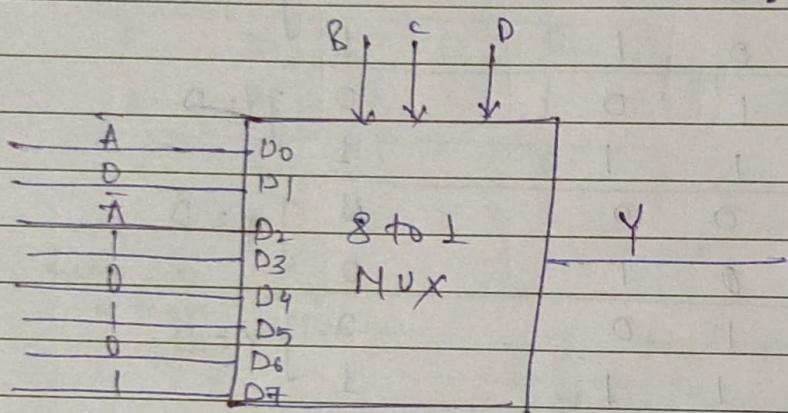
Q4) Realise the given boolean function using 8 to 1 multiplexer. Use select line 'A' as map entered variable.

$$Y(A, B, C, D) = \sum m(0, 2, 3, 5, 7, 11, 13, 15).$$

'A' as map entered variable.

D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	
0	1	2	3	4	5	6	7	
8	9	10	11	12	13	14	15	
A	A	L	O	L	A	D	I	

Implementation
table.

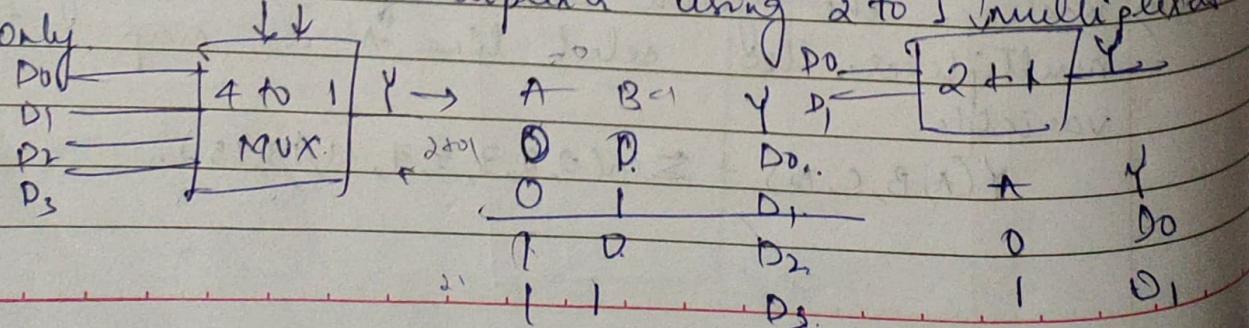


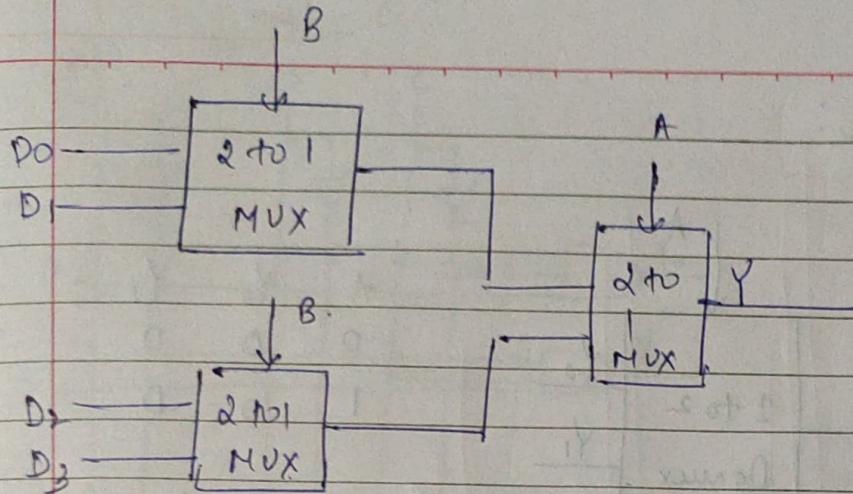
Q5) Realise the given boolean func using 4 to 1 multiplexer. Use select line 'A' as map entered variable.

$$Y(A, B, C) = \sum (1, 3, 5, 7).$$

→ 'A' as map entered f variable. B C

Q6) Realise 4 to 1 multiplexer using 2 to 1 multiplexers only.

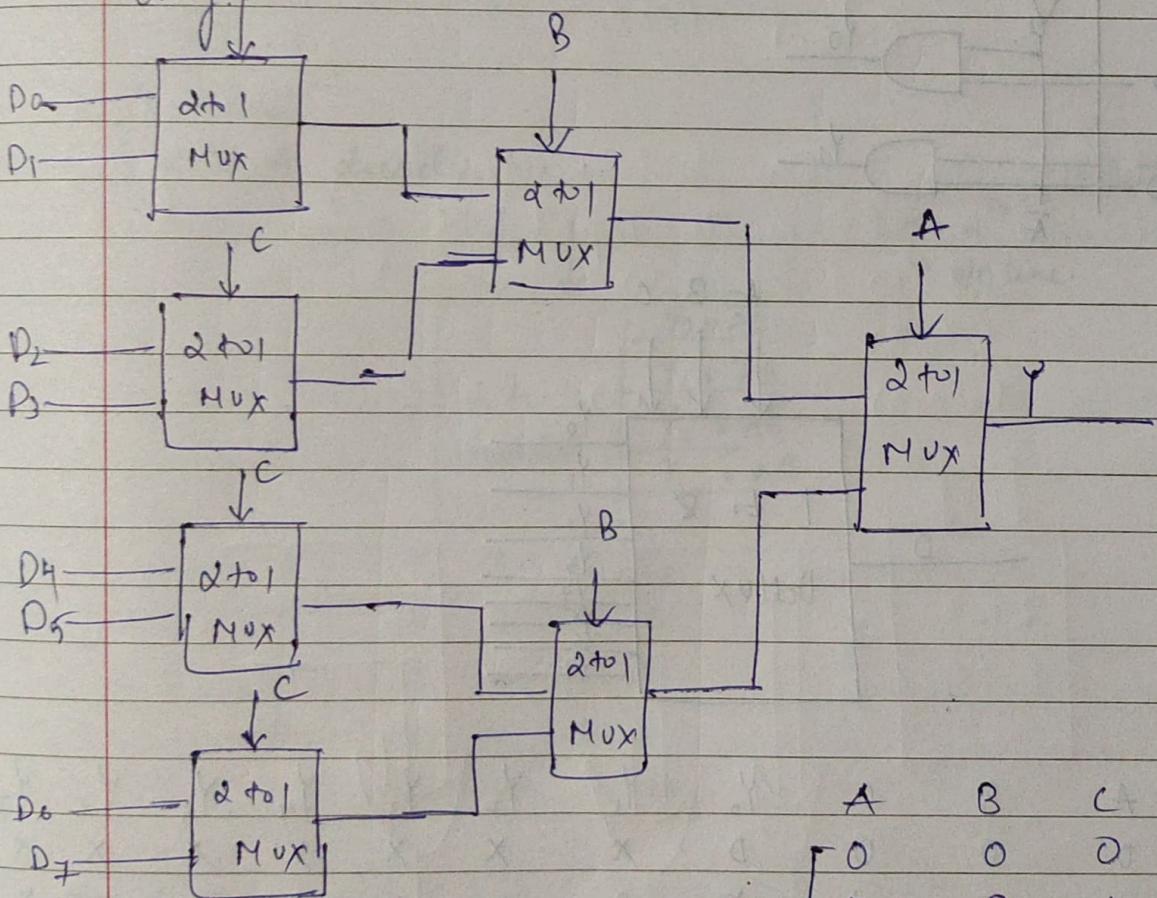




A	B	Y
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

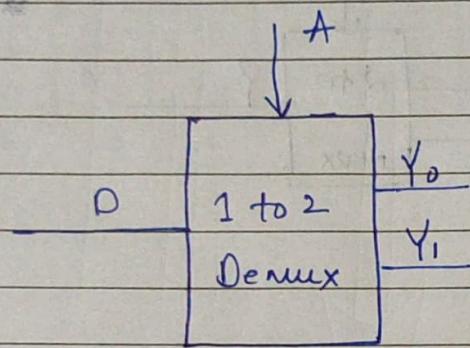
Q7) Realise 8-to-1 multiplexer using 2-to-1 multiplexers.

only C



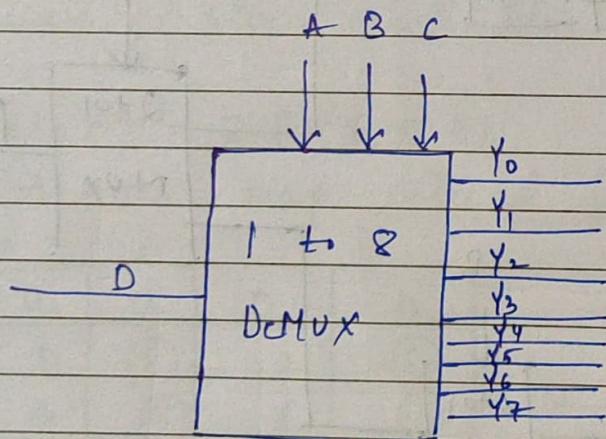
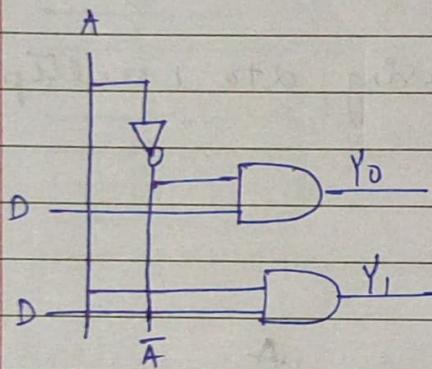
A	B	C	Y
0	0	0	D_0
0	0	1	D_1
0	1	0	D_2
0	1	1	D_3
1	0	0	D_4
1	0	1	D_5
1	1	0	D_6
1	1	1	D_7

* Demultiplexer :



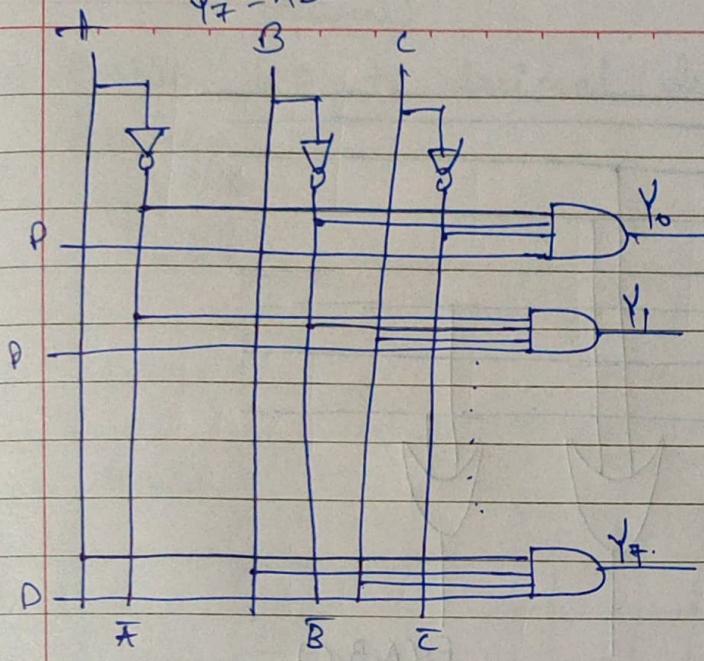
$$\begin{array}{ccc} A & Y_0 & Y_1 \\ 0 & D & 0 \\ 1 & 0 & D \end{array}$$

$$Y_0 = \overline{AD} \quad Y_1 = AD.$$



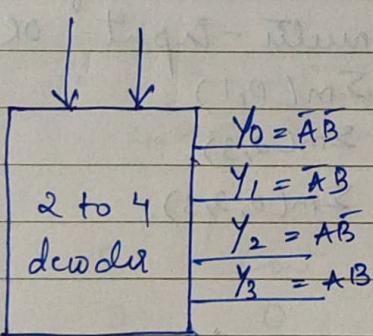
$$\begin{aligned} Y_0 &= \overline{A}\overline{B}C \cdot D \\ Y_1 &= \overline{A}\overline{B}C \cdot D \\ Y_2 &= ABC \cdot D \end{aligned}$$

} write all
exp:



* Decoder:

2 to 4 decoder.



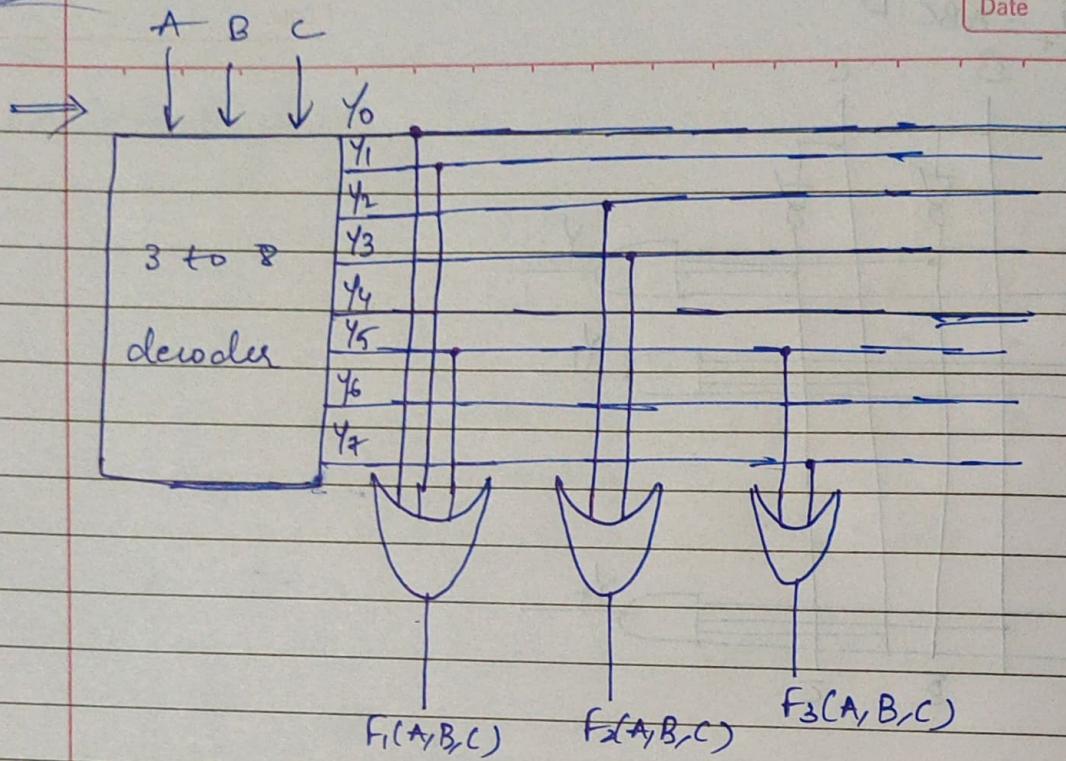
A	B	Y_0	Y_1	Y_2	Y_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

Q) Realise the following boolean expressions using suitable decoder & multi-input OR gates.

$$F_1(A, B, C) = \Sigma_m(0, 1, 5)$$

$$F_2(A, B, C) = \Sigma_m(2, 3)$$

$$F_3(A, B, C) = \Sigma_m(5, 7).$$



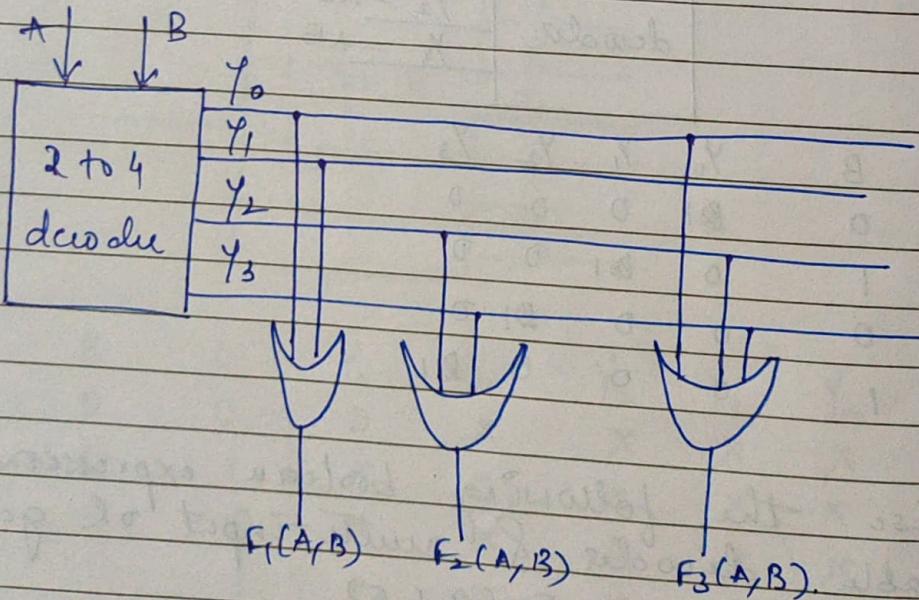
Q2) Realise the given boolean functions using suitable decoder and multi-input OR gates.

$$F_1(A, B) = \Sigma m(0, 1)$$

$$F_2(A, B) = \Sigma m(2, 3)$$

$$F_3(A, B) = \Sigma m(0, 2, 3).$$

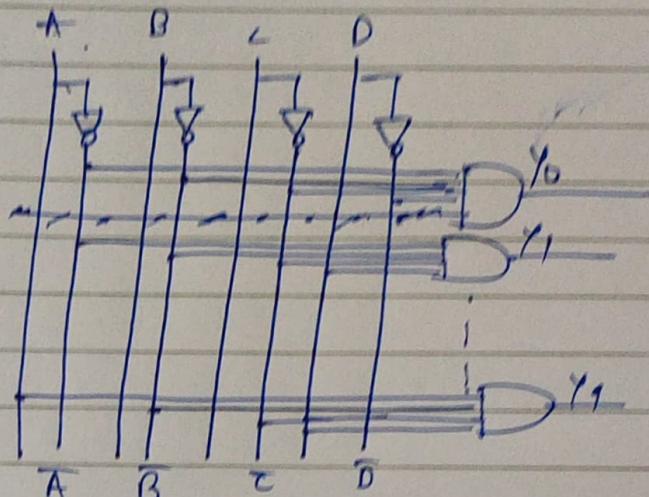
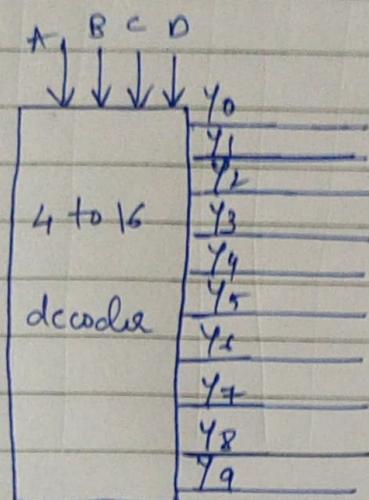
Sol:



(Binary coded decimal \rightarrow D-9)

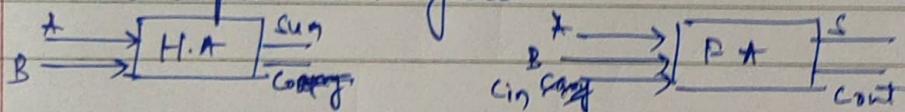
Q3) Realise BCD to decimal decoder using suitable decoder.

Sol:

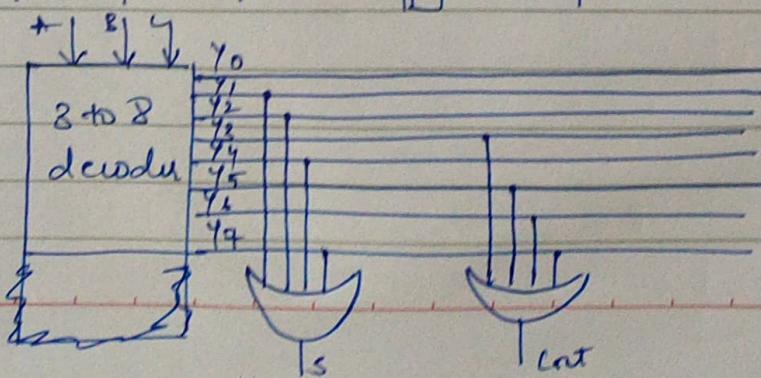


Q4) Realise one-bit full adder using suitable decoder & multi-input OR gate.

Sol:



Inputs			Outputs	
A	B	Cin	S	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



Hardik