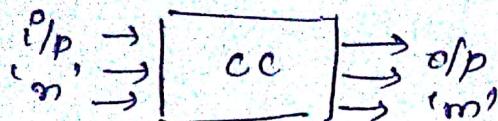


Sec -

Q1 Design procedure of combinational circuit?

CC: It consists of logic gates whose output at any time are determined directly from the present combination of inputs.



2^n possible i/p 2^m possible o/p

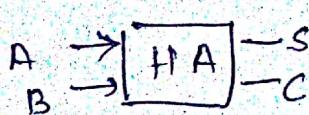
Step 1: Define the problem statement

- 1:- Define no. of i/p variables & no. of o/p variables.
- 2:- Assign letter or symbol to i/p & o/p variables
- 3:- Define truth-table which gives you reln b/w i/p & o/p.
- 4:- Define simplified boolean funⁿ for each & every o/p.
- 5:- Draw the logic diagram.

Q2

• Half Adder:

Definition- HA is a CC with two binary Pinputs & two binary o/p. It adds the two Pinput & produces the sum & the carry.



A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

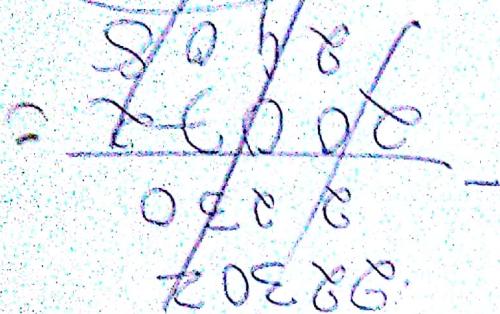
A	B
0	0
1	0

S :- $\begin{array}{|c|c|} \hline A & B \\ \hline 0 & 0 \\ \hline 0 & 1 \\ \hline 1 & 0 \\ \hline 1 & 1 \\ \hline \end{array}$ \Rightarrow XOR Gate
 $= A \oplus B$

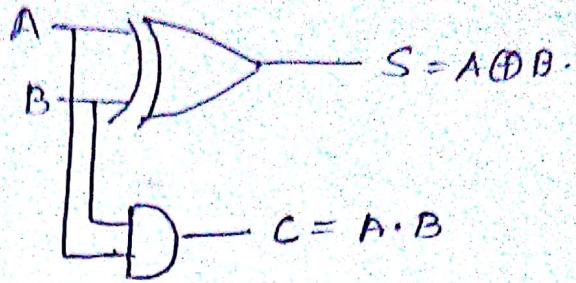
A	B
0	0
1	0

C :- $\begin{array}{|c|c|} \hline A & B \\ \hline 0 & 0 \\ \hline 0 & 1 \\ \hline 1 & 0 \\ \hline 1 & 1 \\ \hline \end{array}$ $\Rightarrow A \cdot B$.

Circuit diagrams.

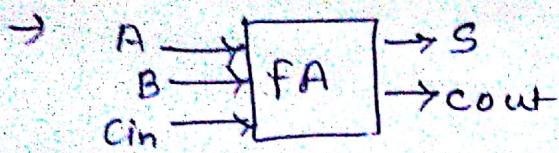


1.7



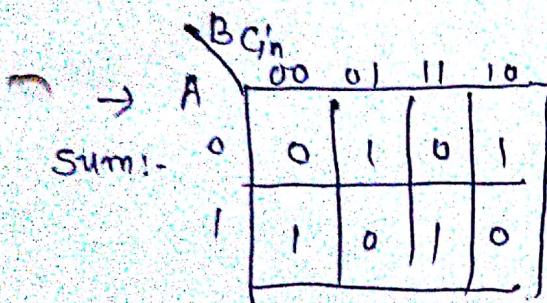
v] full adder:-

A full Adder is a CC that adds two bits & a carry & outputs a sum bit & a carry bit.



→

A	B	Cin	S	C
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



$$\Rightarrow A\bar{B}\bar{C}_{in} + \bar{A}\bar{B}C_{in} + A\bar{B}C_{in} + \bar{A}B\bar{C}_{in}$$

$$\Rightarrow \overline{C_{in}}(A\bar{B} + \bar{A}B) + C_{in}(\bar{A}\bar{B} + AB)$$

$$\Rightarrow \bar{C}_{in}(A \oplus B) + C_{in}(A \oplus B)$$

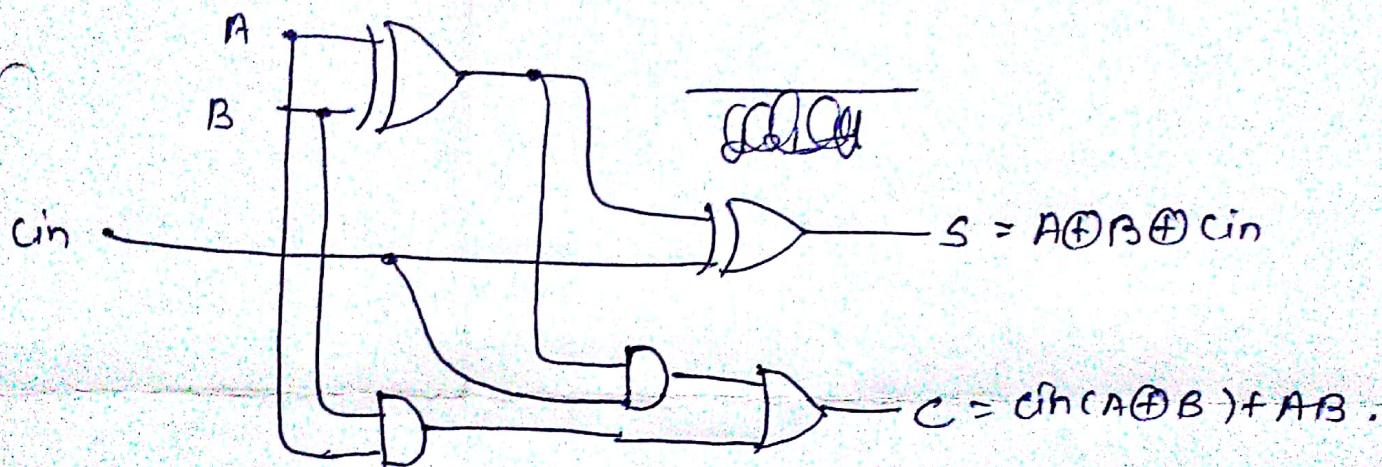
$$\Rightarrow \bar{C}_{in} \oplus X$$

$$S. \Rightarrow \bar{C}_{in} \oplus A \oplus B$$

$$\rightarrow \text{Carry: } \bar{A}B\text{cin} + A\bar{B}\text{cin} + AB\bar{\text{cin}} + AB\text{cin}$$

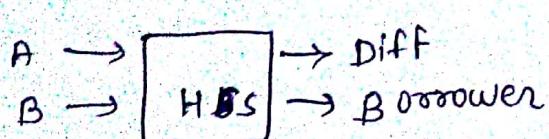
$$\therefore (\text{cin}(A \oplus B) + AB)$$

\rightarrow Circuit diagram:-



(iii) Half Subtractor:-

\rightarrow A H/S is a CC that subtracts one bit from the other eg produces the difference.



A	B	D	B	B
0	0	0	0	0
0	1	1	1	1
1	0	1	1	0
1	1	0	0	0

$\rightarrow \text{Diff: } \begin{array}{|c|c|} \hline A & B \\ \hline 0 & 0 \\ \hline 0 & 1 \\ \hline 1 & 0 \\ \hline 1 & 1 \\ \hline \end{array} \Rightarrow A \oplus B = \text{XOR.}$

\rightarrow Circuit diagram:-

$\Rightarrow \text{Borrower: } \begin{array}{|c|c|} \hline A & B \\ \hline 0 & 0 \\ \hline 0 & 1 \\ \hline 1 & 0 \\ \hline 1 & 1 \\ \hline \end{array} \Rightarrow \bar{A}B$



Ques (iv) full subtractor -

~~Ans~~ A fs ps or cc that performs a subtraction b/w two bits, taking into account that a & may have been borrowed by a lower significant stage.

A	B	B ^{bin}	Diff	Bout
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

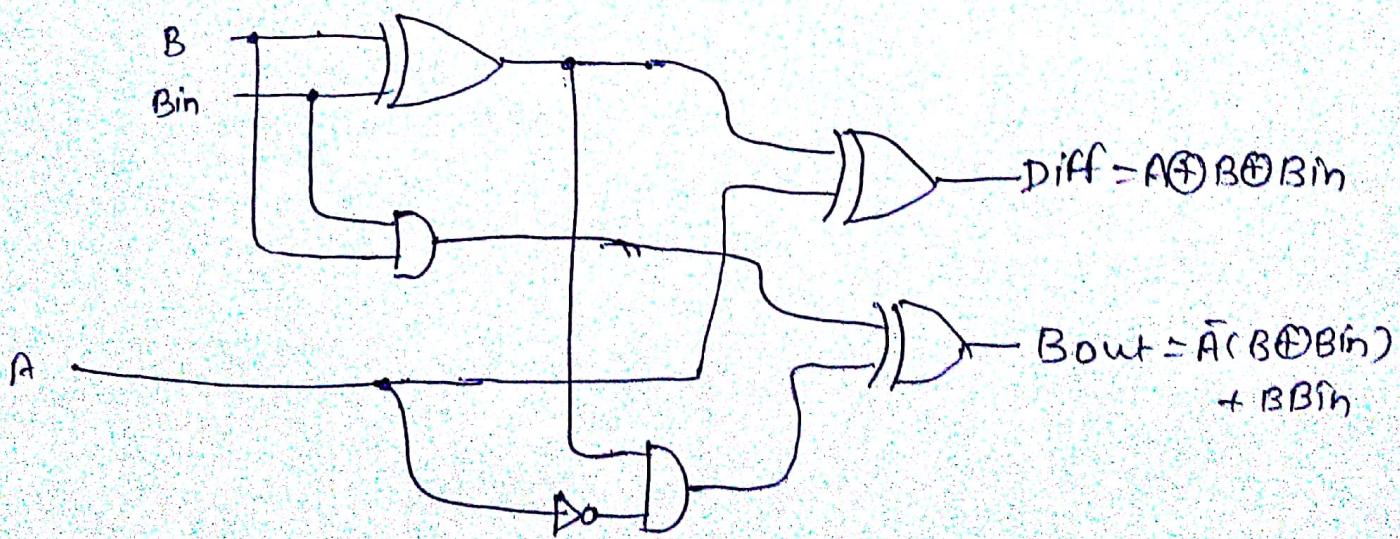
B ^{bin}		0	1	1	1
A		0	0	1	0
B ^{bin}	A	0	0	1	0
1	1	1	0	1	0

$$\Rightarrow A \oplus B \oplus B^{\text{bin}}$$

$$\begin{aligned} \text{Bout: } & \bar{A} \bar{B} B^{\text{bin}} + \bar{A} B \bar{B}^{\text{bin}} \\ & + \bar{A} B B^{\text{bin}} + A B \bar{B}^{\text{bin}} \end{aligned}$$

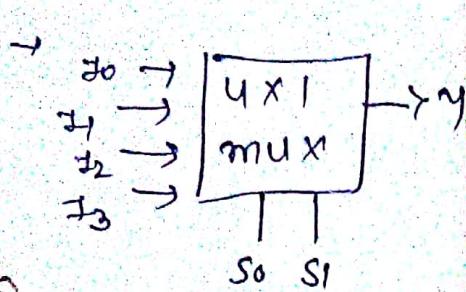
$$\Rightarrow \bar{A} (B \oplus B^{\text{bin}}) + B B^{\text{bin}}.$$

Circuit diagram:



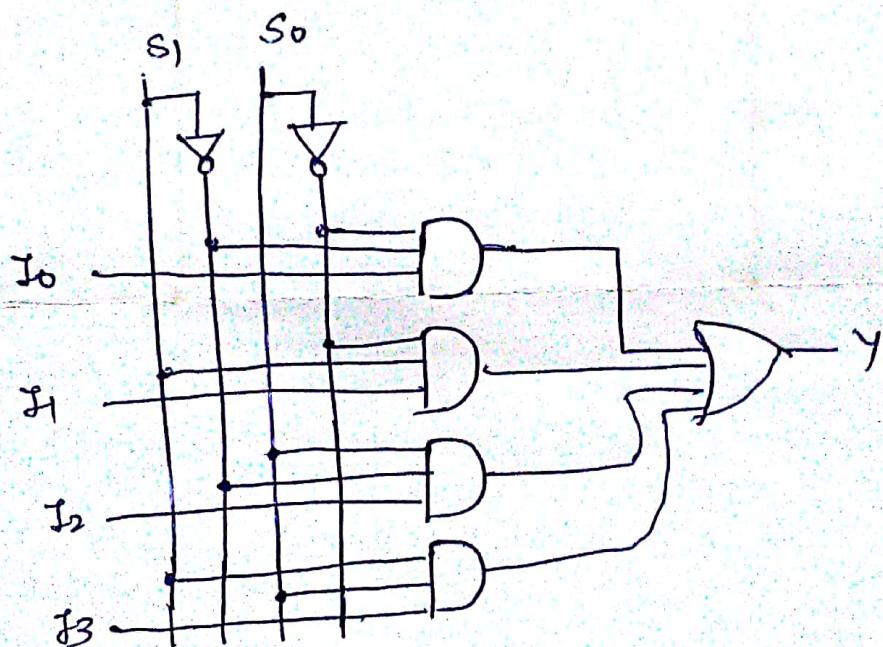
(v) Multiplexers:- (4x1).

A multiplexer or data selector is a logic circuit that accepts several data inputs & allows only one of them at a time to get through the output.



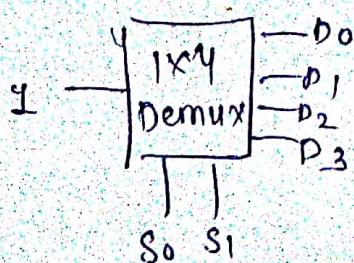
$S_0 \cdot S_1$	Y
0 0	I_0
0 1	I_1
1 0	I_2
1 1	I_3

Circuit diagram:



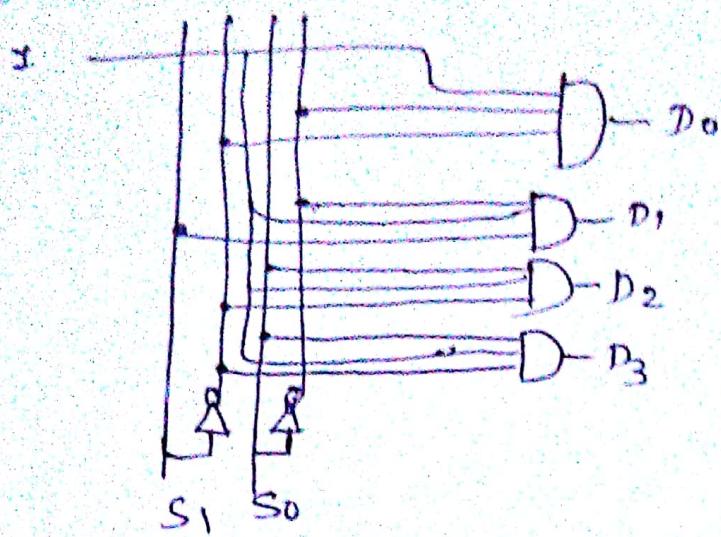
(vi) Demultiplexers:- (1x4)

A demultiplexer performs the reverse operation; it takes a single input & distributes it over several outputs.



S_0	S_1	D_0	D_1	D_2	D_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

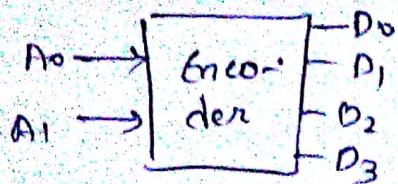
Circuit diagram



(VII) Encoders: (4 to 2) (2^n to n):-

An encoder is a device which converts familiar numbers or symbols into coded format.

→ ~~4x2~~ 4X2

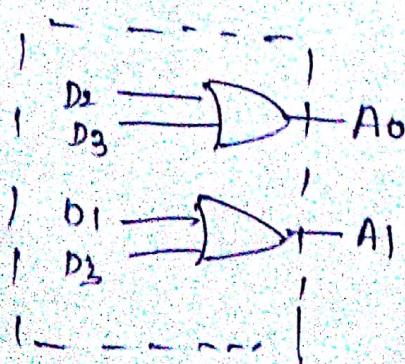


A_0	A_1	γ
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

circuit:-

$$B_P = A_1 \cdot A_0 = D_1 + D_3$$

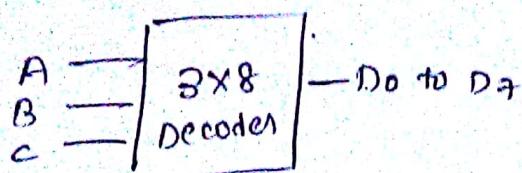
$$A_0 = D_2 + D_3$$



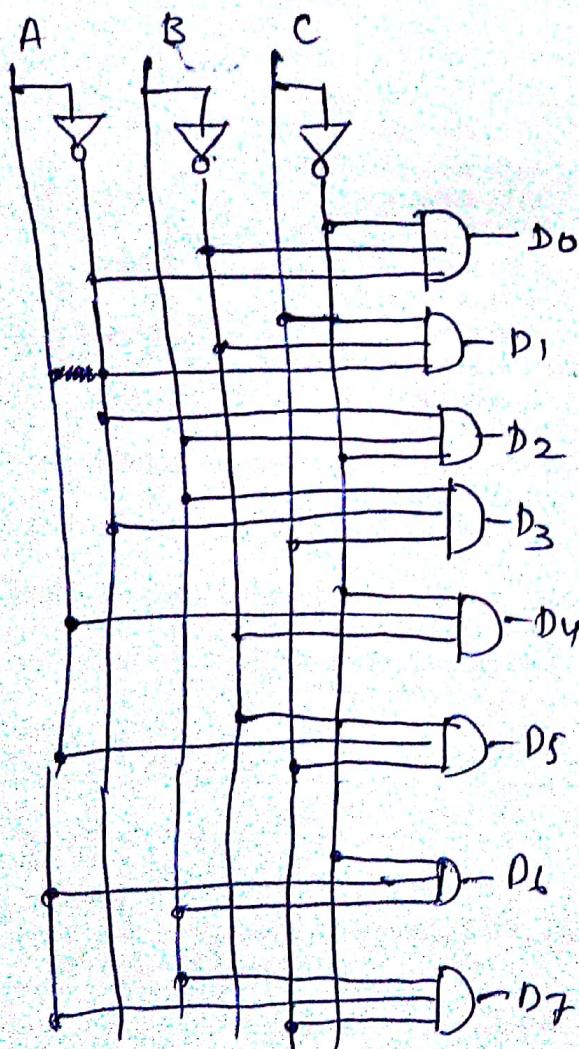
It accepts 4 input
lines & produces a
2 bit output code
corresponding to the
activated input.

(viii) Decoder (3-to 8) (n -to 2^n)

A decoder is a logic circuit that converts an n bit binary input code n into m o/p lines such that only one o/p line is activated for each one of the possible combinations of i/p.



Circuit Diagrams



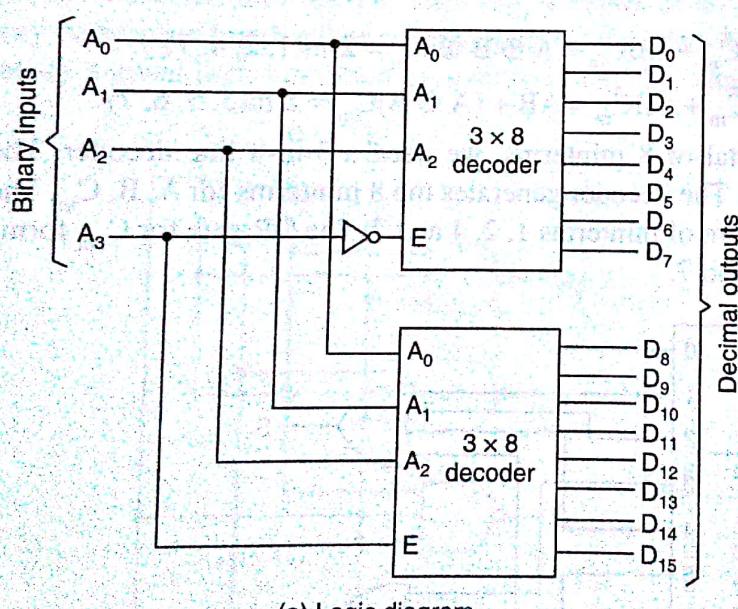
A	B	C	D ₀	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇
0	0	0	1							
0	0	1		1						
0	1	0			1					
0	1	1				1				
1	0	0					1			
1	0	1						1		
1	1	0							1	
1	1	1								1

A decoder with 3 i/p & 8 o/p. it uses an AND gate. therefore the o/p are active high. for active low o/p NAND gate is used. It is also called binary-to-octal decoder. the 3 bit binary i/p code activates one of the eight o/p corresponding to that code.

7.23.6 4-to-16 Decoder from Two 3-to-8 Decoders

Decoders with enable inputs can be connected together to form a larger decoder circuit. Figure 7.72 shows the arrangement for using two 74138s, 3-to-8 decoders, to obtain a 4-to-16 decoder. The most significant input bit A_3 is connected through an inverter to \bar{E} on the upper decoder (for D_0 through D_7) and directly to E on the lower decoder (for D_8 through D_{15}). Thus, when A_3 is LOW, the upper decoder is enabled and the lower decoder is disabled. The bottom decoder outputs all 0s, and top 8 outputs generate minterms. When A_3 is HIGH, the lower decoder is enabled and the upper decoder is disabled. The bottom decoder generates minterms 1000 to 1111 while the outputs of the top decoder are all 0s.

This example demonstrates the usefulness of enable inputs in decoders and other combinational logic components. In general, enable inputs are a convenient feature for interconnecting two or more standard components for the purpose of expanding the component into a similar function with more inputs and outputs.



(a) Logic diagram

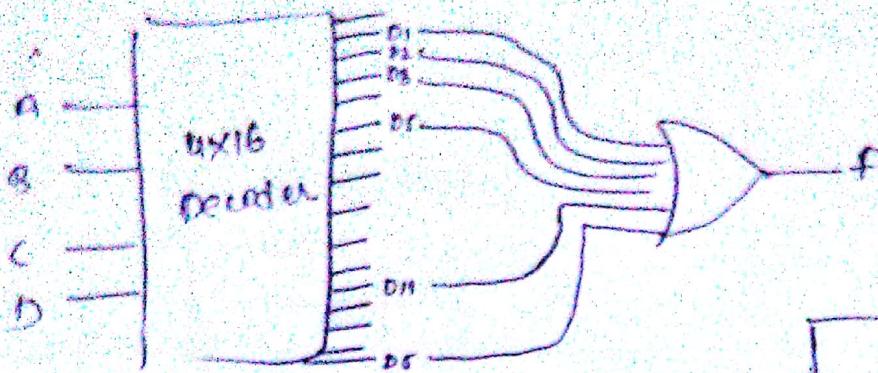
Binary inputs				Decimal output (active low)
A_3	A_2	A_1	A_0	
0	0	0	0	D_0
	0	0	0	D_1
	0	0	1	D_2
	0	0	1	D_3
	0	1	0	D_4
	0	1	0	D_5
	0	1	1	D_6
	0	1	1	D_7
1	0	0	0	D_8
	1	0	0	D_9
	1	0	1	D_{10}
	1	0	1	D_{11}
	1	1	0	D_{12}
	1	1	0	D_{13}
	1	1	1	D_{14}
	1	1	1	D_{15}

(b) Function table

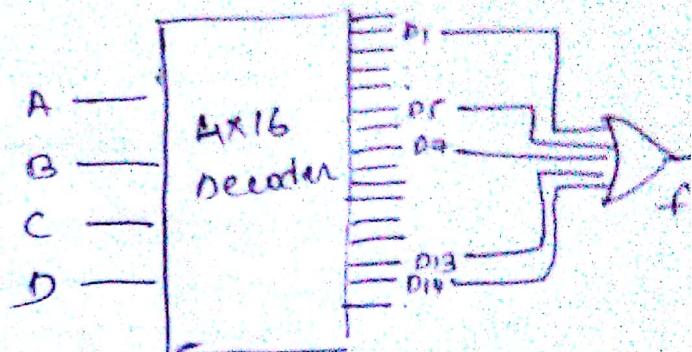
Figure 7.72 Connecting two 74138 3-to-8 decoders to obtain a 4-to-16 decoder.

Q20 Implement using Decoders

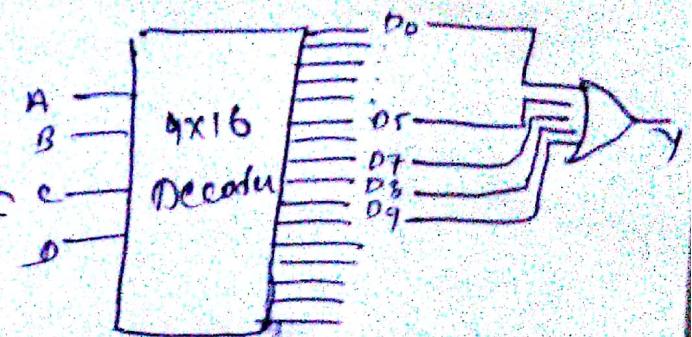
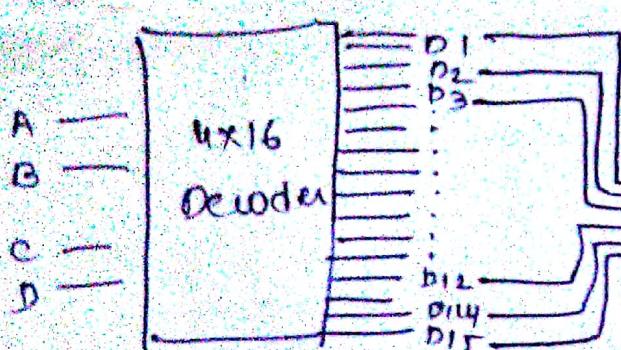
a) $f_1(a, b, c, d) = \Sigma(1, 3, 5, 9, 11, 15)$



b) $\Sigma(1, 5, 7, 14, 13)$

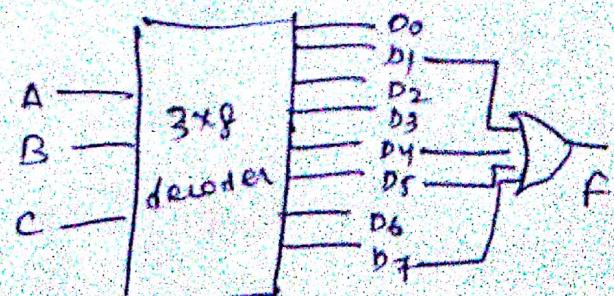


b) $f_1(a, b, c, d) = \Sigma(1, 12, 3, 2, 14, 15) \quad \ominus \quad f_2(a, b, c, d) = \Sigma(0, 5, 7, 8, 9)$

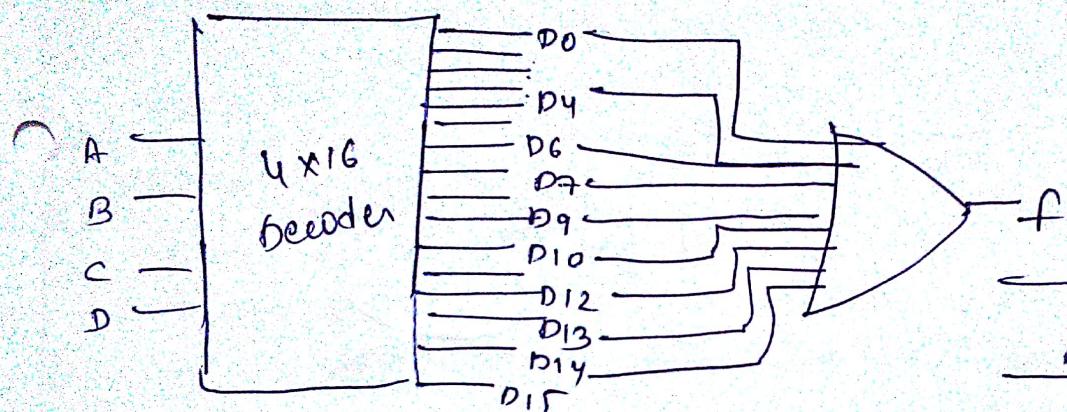


g)
$$\begin{aligned} f &= xy'z + x'y'z + xy'z' + xy'z \\ &= 000 + 001 + 100 + 101 \\ &= 1, 4, 5 \\ &= \Sigma(1, 4, 5, 7) \end{aligned}$$

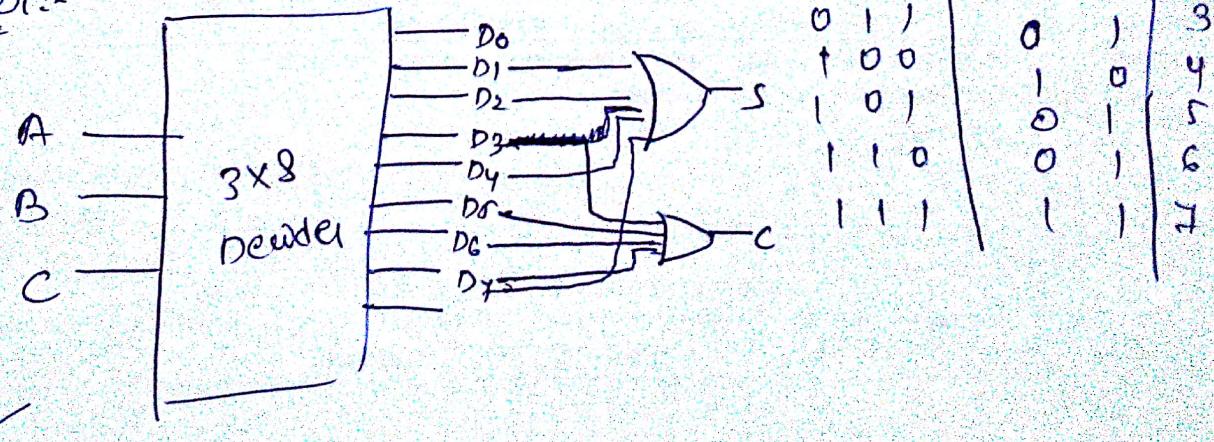
Ex21



d) $f(w, x, y, z) = \pi(1, 2, 3, 5, 8, 11, 15)$
 $= m(0, 4, 6, 7, 9, 10, 12, 13, 14)$



e) full adder:-



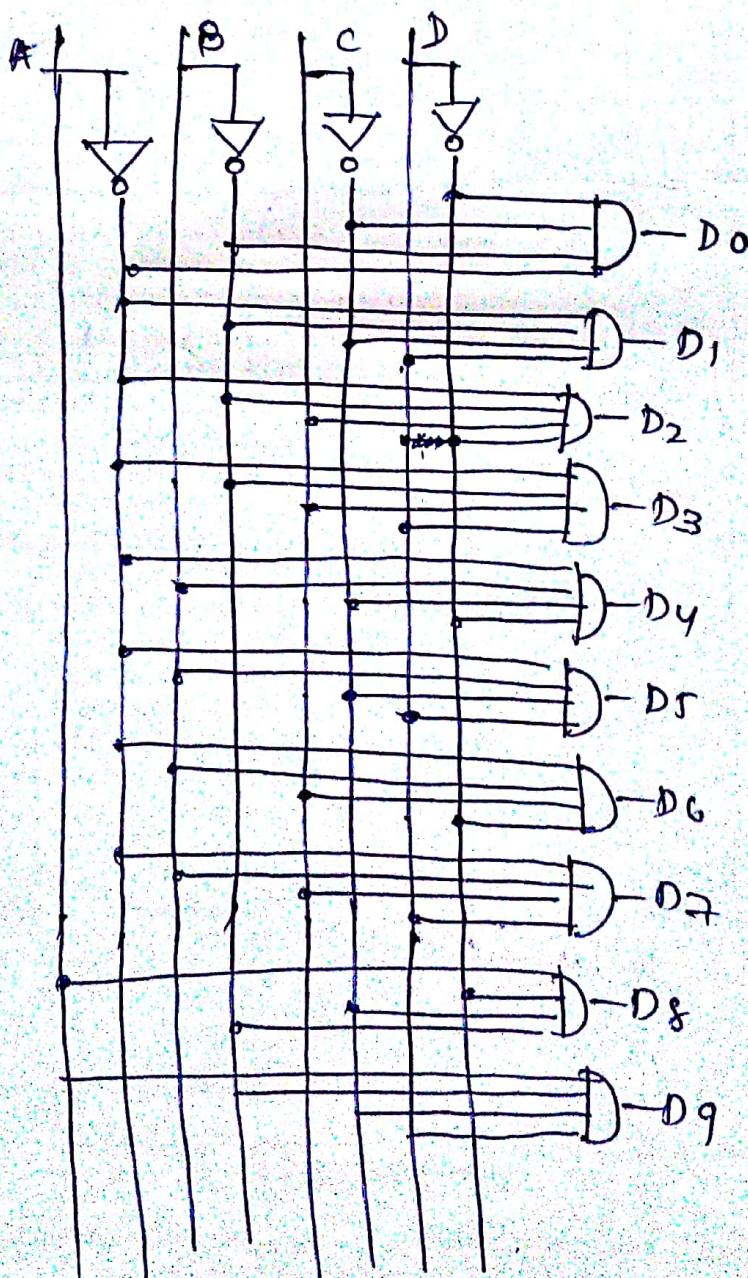
Q14. 4x16 Decoder with two 3x8 decoders.

1B:- 342

Q18 Design BCD to decimal decoder:-

- The decoder should have 4 I/p to accept the coded digit & ten o/p, one for each decimal digit. This will give a 4 line to 10 line BCD to decimal decoder.
- There are 6 don't care conditions here. They will never occur so we mark their corresponding minterms with X.

A	B	C	D	Y
0	0	0	0	D ₀
0	0	0	1	D ₁
0	0	1	0	D ₂
0	0	1	1	D ₃
0	1	0	0	D ₄
0	1	0	1	D ₅
0	1	1	0	D ₆
0	1	1	1	D ₇
1	0	0	0	D ₈
1	0	0	1	D ₉
1	0	1	0	X
1	0	1	1	X
1	1	0	0	X
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X



Q2B Full Adder from 4:1 mux to few Applications:-

A	B	C	S	C
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

