

Q. Realize the given boolean function  $f(A, B, C, D)$

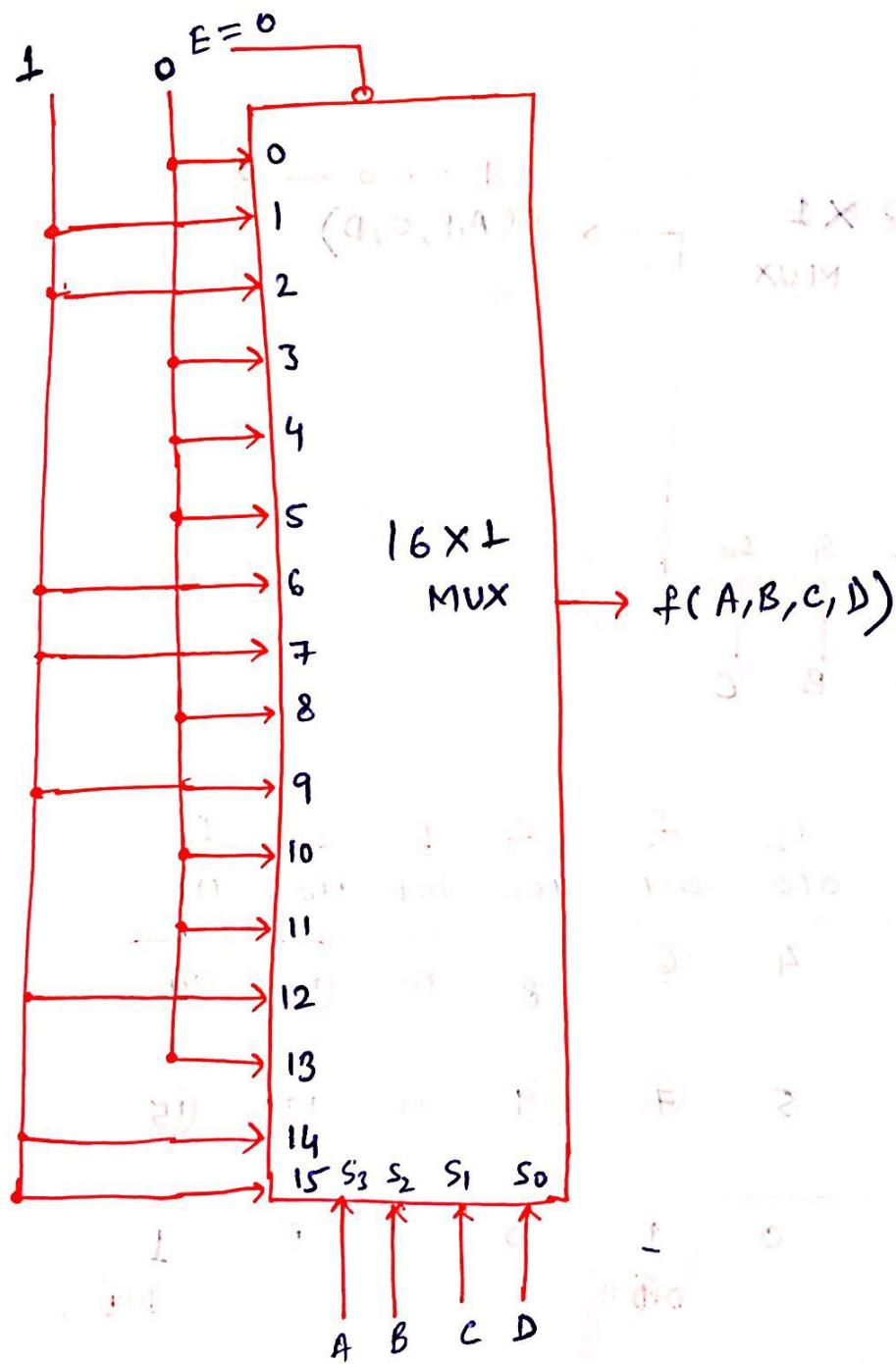
$= \sum m(1, 2, 6, 7, 9, 10, 12, 14, 15)$  using :

(i) 16 to 1 MUX with enable at 0

(ii) 8 to 1 MUX

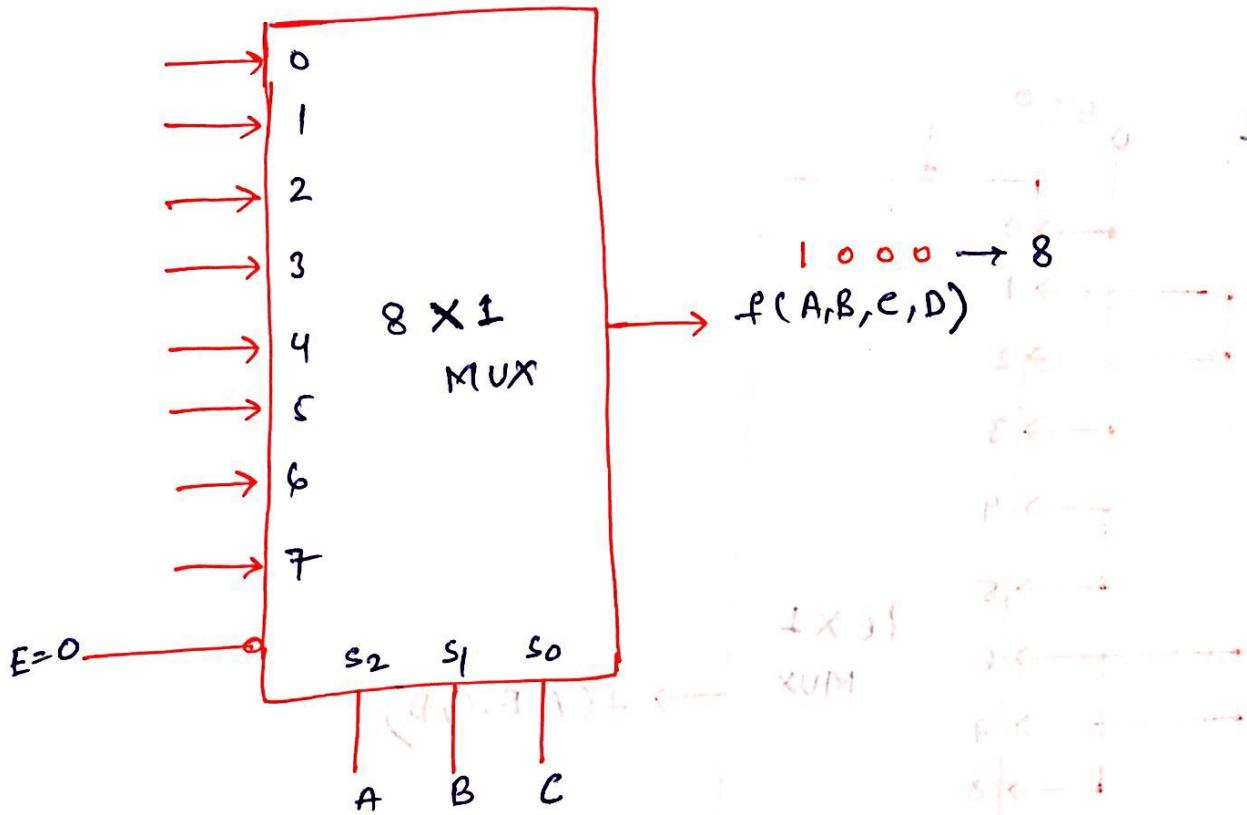
(iii) 4 to 1 MUX

(i)



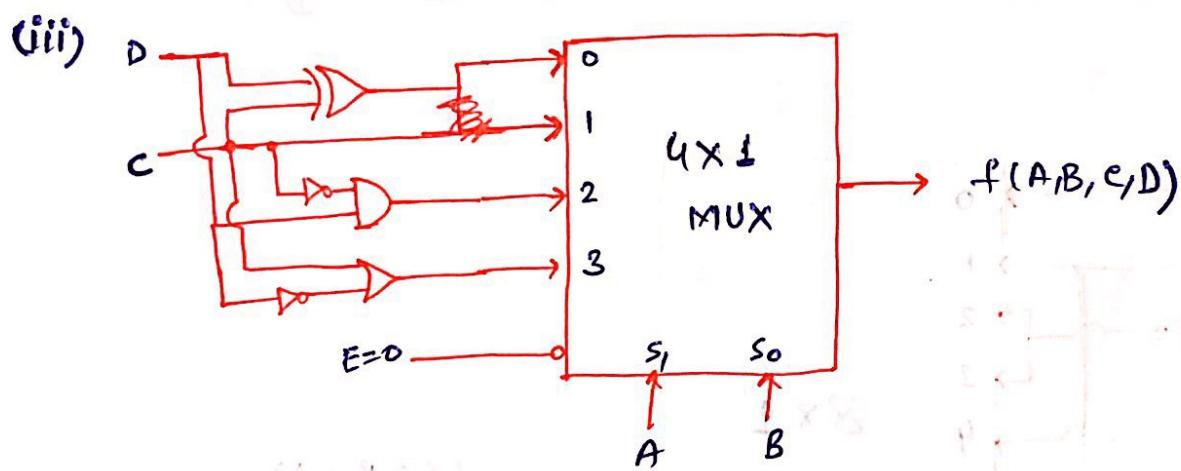
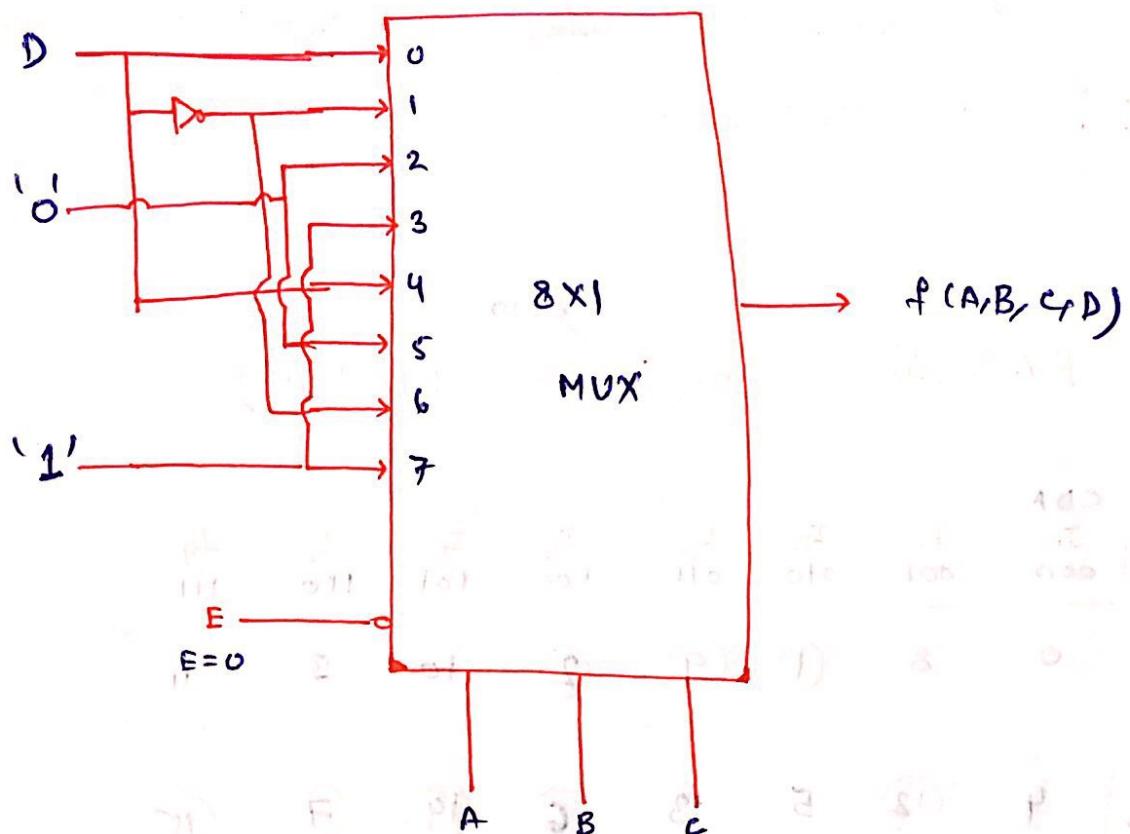
(ii) The variables which are to be connected to select lines are column heading to the tables.

And the variable / variables which are not connected to select lines are row heading to the table,



	$I_0$	$I_1$	$I_2$	$I_3$	$I_4$	$I_5$	$I_6$	$I_7$
$\bar{D}$	0	0	2	4	6	8	10	12
D	1	1	3	5	7	9	11	15

$D + \bar{D} = 1$

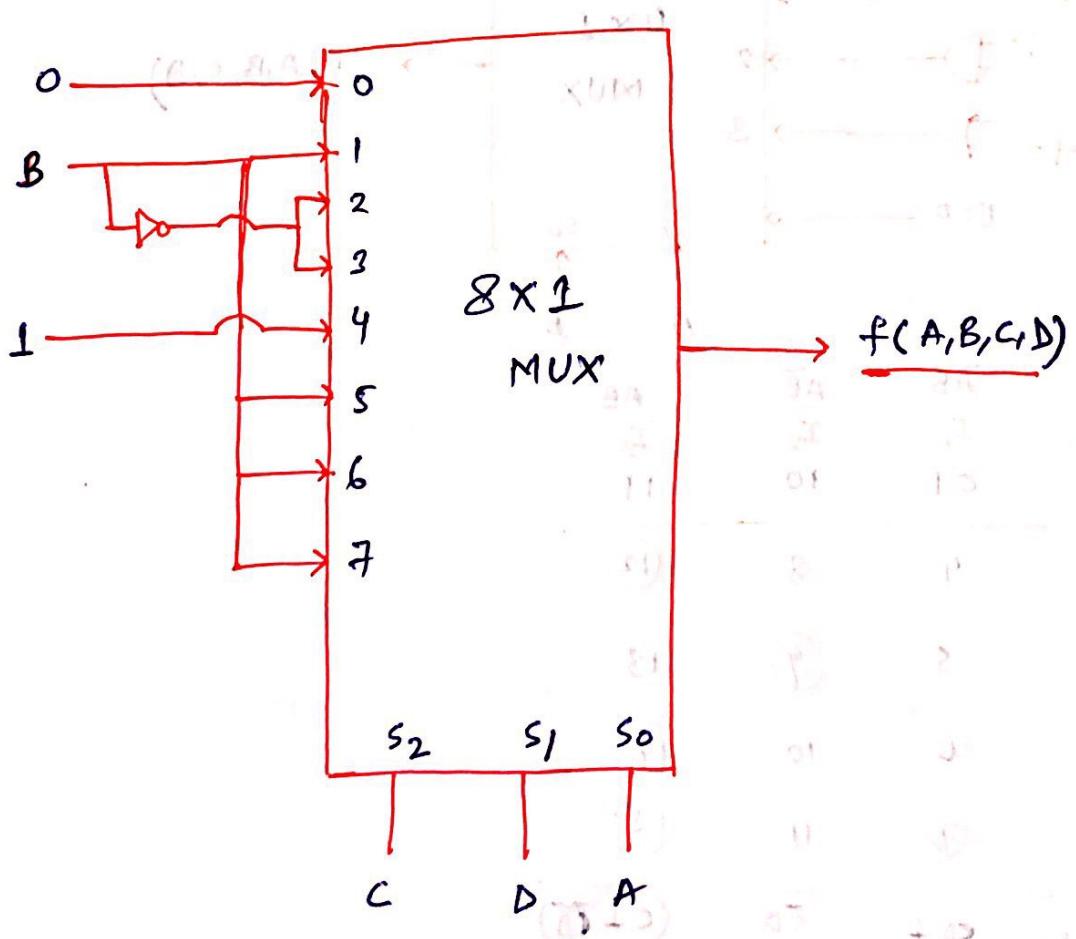


$\bar{A}\bar{B}$	$\bar{A}B$	$A\bar{B}$	$AB$
$I_0$	$I_1$	$I_2$	$I_3$
00	01	10	11
$\bar{C}\bar{D}\ 00$	0	4	8
$\bar{C}D\ 01$	1	5	9
$C\bar{D}\ 10$	2	6	10
$CD\ 11$	3	7	11
$C \oplus D$	$\frac{ep +}{CD}$	$\bar{C}D$	$(C + \bar{C}D)$
	$= C$		

Q. Realize the given function using  $8 \times 1$  MUX while connecting  
 $C, D, A$  to select line  
 $\uparrow \uparrow \uparrow$   
 $s_2 \ s_1 \ s_0$

(Ans. Given)  
 $F(A, B, C, D) = \sum m(1, 3, 6, 7, 9, 12, 14, 15)$

CDA		$I_0$ 000	$I_1$ 001	$I_2$ 010	$I_3$ 011	$I_4$ 100	$I_5$ 101	$I_6$ 110	$I_7$ 111
$\bar{B}$	0	0	8	1	9	2	10	3	11
B	1	4	12	5	13	6	14	7	15



Q. Realize boolean func<sup>n</sup> using 8 4x1 MUX while connecting  $C'$ ,  $A$ .

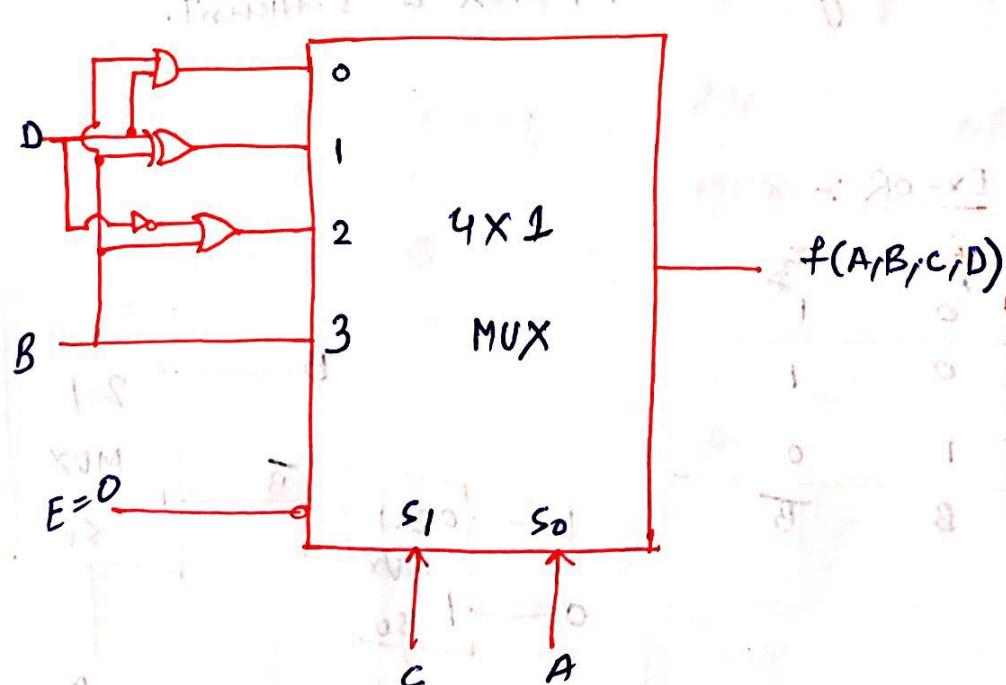
$C'$ ,  $A$

$S_1$ ,  $S_0$

Sum = 111

$$f(A, B, C, D) = \Sigma m(1, 2, 6, 7, 9, 12, 14, 15)$$

	$I_0$	$I_1$	$I_2$	$I_3$
$\bar{B}D$	$\bar{C}\bar{A}$	$\bar{C}A$	$C\bar{A}$	$CA$
00	0	8	2	10
01	1	9	3	11
10	4	12	6	14
11	5	13	7	15



NOTE:-

8:1 MUX  $n=4, m=3 \Rightarrow n-m=1$  (1 NOT)

4:1 MUX  $n=4, m=2 \Rightarrow n-m=2$  (2 NOT)

2:1 MUX  $n=4, m=1 \Rightarrow n-m=3$

- Don't care are treated as MAXTERMS whenever the realization is with MUX.

Q Realize 2 IP AND, OR, NAND, NOR, EX-OR and EX-NOR,

using:

(i) 4:1 MUX only

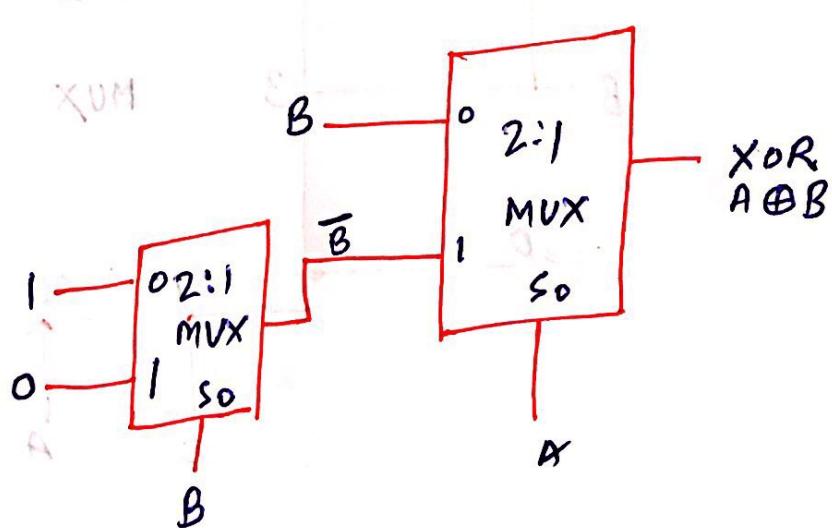
(ii) 2:1 MUX only

(i) Since there are two select lines  $m=2$  and  $n=2$  so, only one 4:1 MUX is sufficient.

(ii)

Ex-OR :-

	$\bar{A}$	$A$	$\bar{B}$	$B$
$\bar{B}$	0	0	1	
$B$	1	1	0	



2 MUX needed

### Ex-NOR :

A	B	f
0	0	1
0	1	0
1	0	0
1	1	1

2 MUX needed

### NAND :

A	B	f
0	0	1
0	1	1
1	0	1
1	1	0

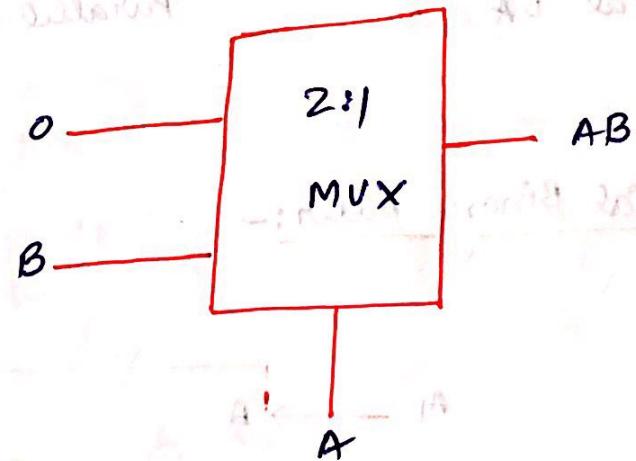
$\bar{A}$	$A$
0	1
1	0

$\bar{B}$	$B$	$\bar{A}\bar{B} + \bar{A}B$	$\bar{AB}$
0	0	0	0
1	1	1	1

### AND :

A	B	f
0	0	0
0	1	0
1	0	0
1	1	1

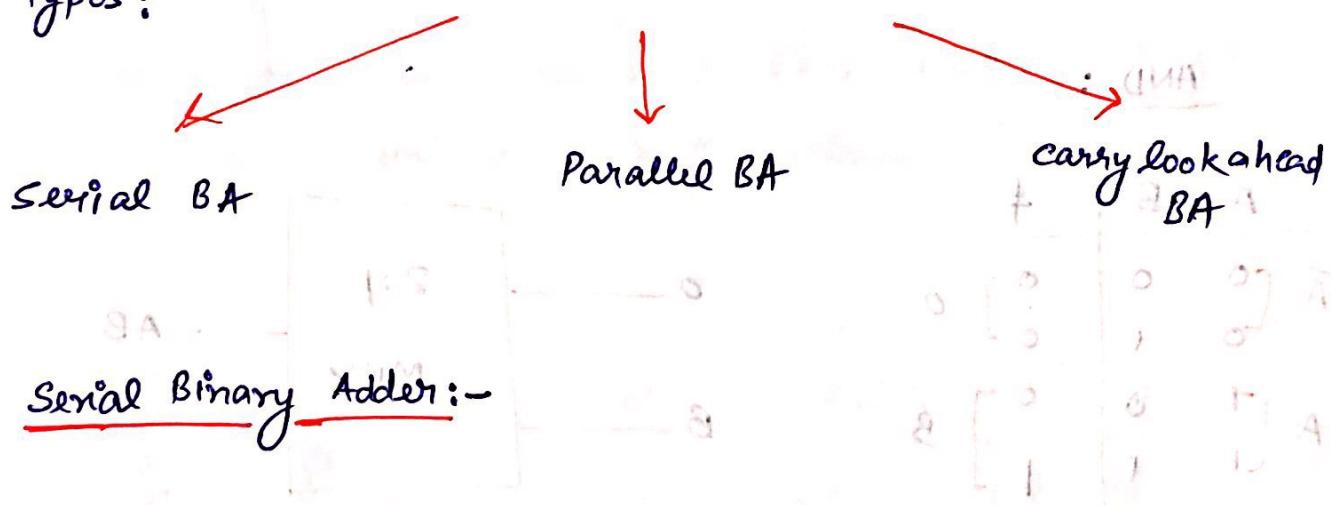


Q. By using only 4:1 MUX it is possible to realize:

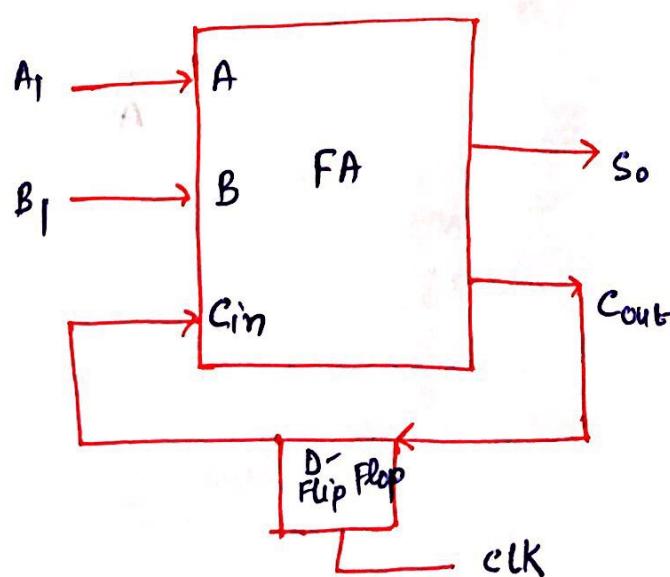
- (a) any three variable func<sup>n</sup>.
- (b) only 2 variable func<sup>n</sup>.
- (c) ~~any~~ 2 and 3 variable func<sup>n</sup>
- (d) few 3 variable func<sup>n</sup>.

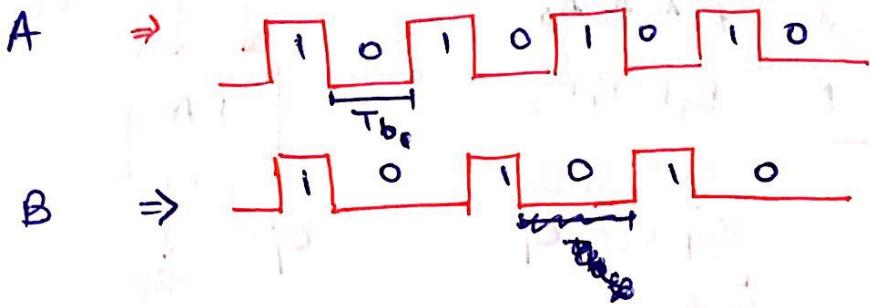
### Binary Adder:-

It is a ckt, which adds  $\overline{2}$  multi-bit numbers. It is of 3 types:



### Serial Binary Adder:-





$$T_b = \frac{T_{pb}}{2}$$

bit repetition time

Assume  $T_p$  is propagation delay

$$T_p \approx 1\text{ns}$$

$$\text{and } T_b = 1\text{μs}$$

- 'D-Flip flop' is used to generate a delay for 'carry in' signal until A and B I/p are available.

$$(T_c \text{ (clk signal)} \approx T_b)$$

Serial BA	$n=4$	$n$
	$3T_b$	$(n-1)T_b$

o/p generating time

\* So, o/p time depends upon size of I/p which is not good w.r.t CPU clock.

$$(O/P \propto \text{input size } (n))$$

$$\propto T_b$$

## (ii) Parallel Binary adder :-

$$A = A_{n-1} \ A_{n-2} \ \dots \ A_1 \ A_0$$

$$B = B_{n-1} \ B_{n-2} \ \dots \ B_1 \ B_0$$

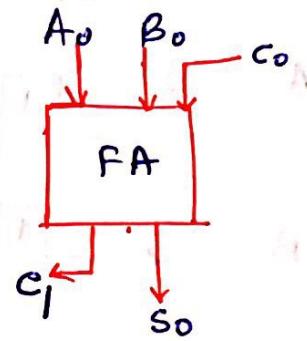
for  $n = 4$ ;

$$A = A_3 A_2 A_1 A_0 \leftarrow C_0$$

$$B = B_3 B_2 B_1 B_0$$

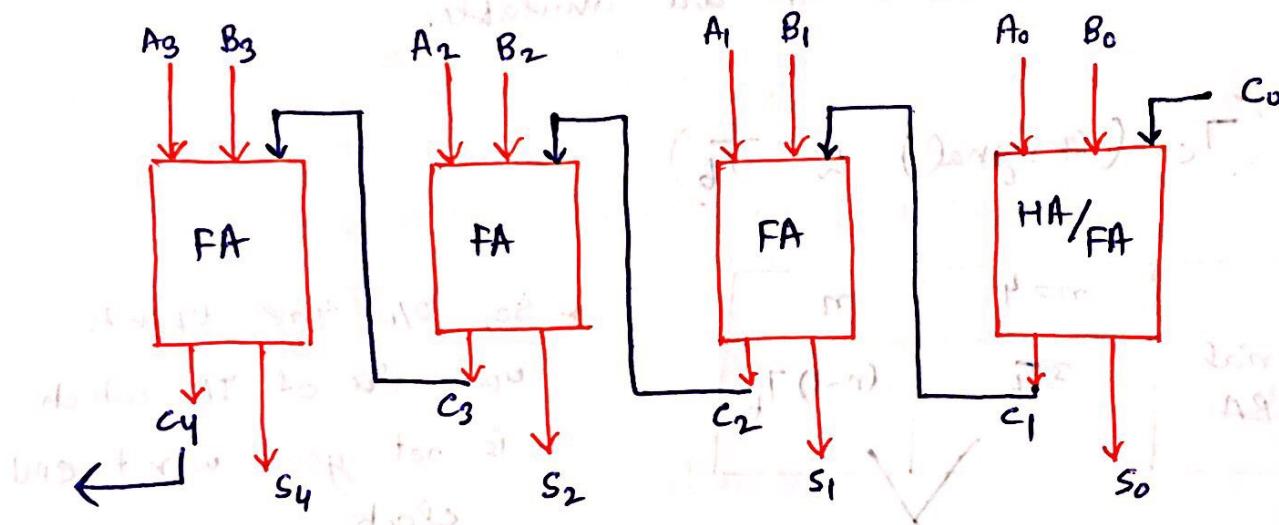
$$\begin{matrix} S_3 & S_2 & S_1 & S_0 \\ C_4 & C_3 & C_2 & C_1 \end{matrix}$$

final carry



Read out =  $C_4 S_3 S_2 S_1 S_0$

2-level ckt is used (no XOR gate is used)



All bits of A and B are simultaneously on respective FA units.

So here o/p delay is only depends upon the propagation delay of FA ckt's not on  $T_f$  (bit representation time).

H/W cost increases and performance increases.

It is also known as ripple carry adder (carry generating at LSB travels all to MSB).

$$(HA_0) \Rightarrow S_0 = \overline{A_0}B_0 + A_0\overline{B_0} \quad 2T_{pd}$$

$$c_1 = A_0B_0 \quad 1T_{pd}$$

$$FA_i \quad S_i = \overline{A_i}B_iC_i + \overline{A_i}B_i\overline{C_i} + A_i\overline{B_i}\overline{C_i} + A_iB_iC_i \quad 2T_{pd}$$

$$c_{i+1} = A_iB_i + B_iC_i + A_iC_i \quad 2T_{pd}$$

Q. To realize 4-bit Parallel BA, at LSB HA is used.

At other place FA is implemented. Each logic gates used in the ckt is having propagation delay of  $1T_{pd}$ . Variables are available primed and un-primed form. What is the op<sup>n</sup> time of 4-bit parallel BA.

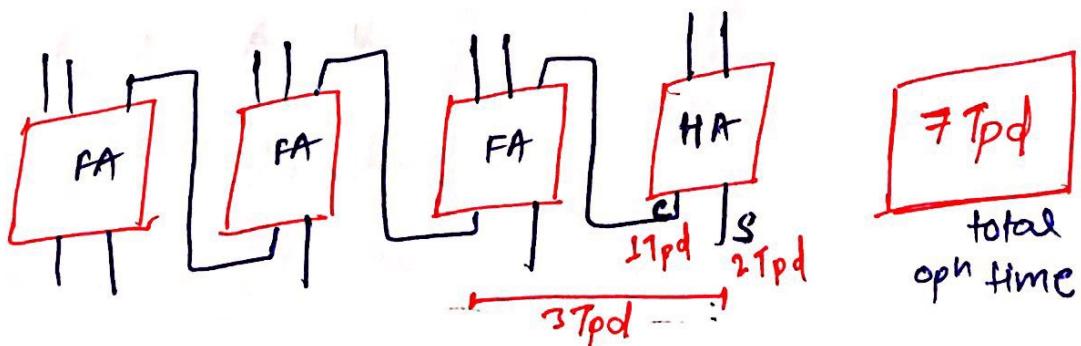
$$HA_0 \Rightarrow S_0 = \overline{A_0}B_0 + A_0\overline{B_0} \quad 2T_{pd}$$

$$c_1 = A_0B_0 \quad 1T_{pd}$$

Since  $c_1$  needs to transfer to next FA

Both  $S_0$  and  $c_1$   
are available

so,



Adder	Input size ( $n$ )
SBA	$(n-1) T_b$
PBA	$(2n-1) T_p$

- \* In PBA, op<sup>n</sup> time depends also upon input size which is a drawback.

(iii) Carry look ahead Binary adder:-

$$C_1 = A_0 B_0 \quad \text{--- (i)}$$

$$\begin{aligned} C_2 &= A_1 B_1 + B_1 C_1 + A_1 C_1 \\ &= A_1 B_1 + A_0 B_0 B_1 + A_0 A_1 B_0 \quad \text{--- (ii)} \end{aligned}$$

$$C_3 = A_2 B_2 + B_2 C_2 + A_2 C_2$$

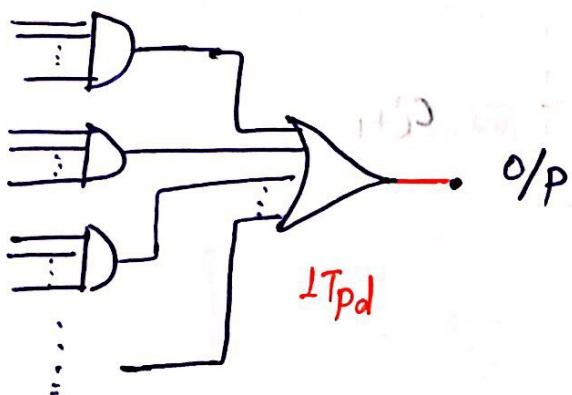
$$= A_2 B_2 + (B_2 + A_2) (A_1 B_1 + A_0 B_0 B_1 + A_0 A_1 B_0)$$

$$\begin{aligned} &= A_2 B_2 + A_1 B_1 B_2 + A_1 A_2 B_1 + A_0 B_0 A_2 B_0 B_1 + \\ &\quad A_0 B_0 B_1 B_2 + A_0 A_1 B_0 B_2 + A_0 A_1 A_2 B_0 \end{aligned}$$

(iii)

- At FA input carry is entering with a delay of  $2T_{pd}$ . After knowing carry FA is performing addition with a delay of  $2T_{pd}$ . Hence effective delay is  $4T_{pd}$ .

NOTE:- op<sup>n</sup> delay of SOP equation is always  $2T_{pd}$ .



- In CLA, XOR gates are used to implement SUM & CARRY.

$$S_i = \underbrace{A_i \oplus B_i}_{P_i} \oplus C_i$$

$$C_{i+1} = \underbrace{(A_i \oplus B_i) C_i}_{P_i} + \underbrace{A_i B_i}_{Q_i}$$

$$A = A_3 \ A_2 \ A_1 \ A_0$$

$$+ B = B_3 \ B_2 \ B_1 \ B_0$$

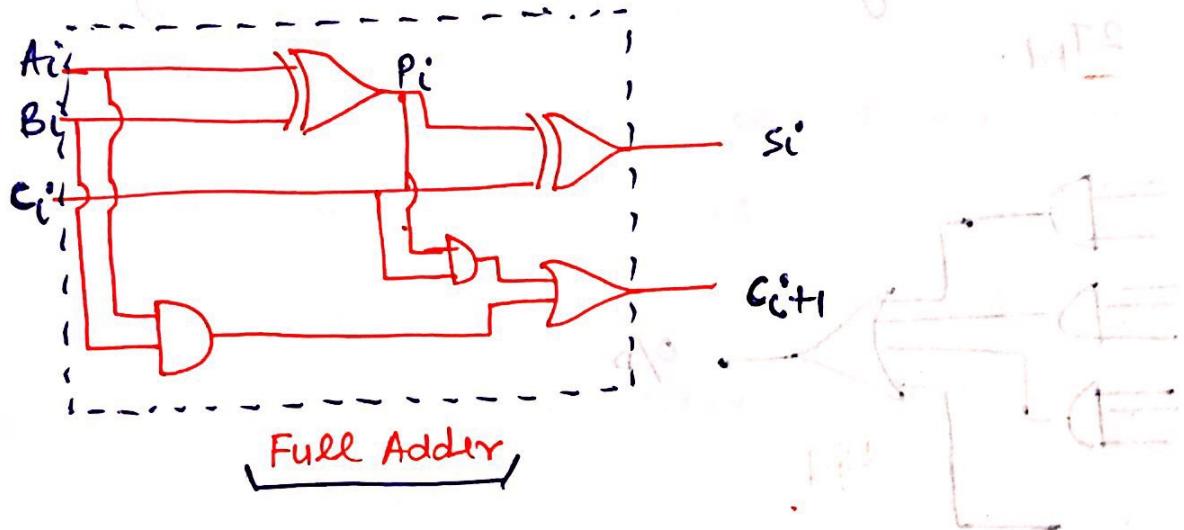
$$\overline{\boxed{C_4 \ S_3 \ S_2 \ S_1 \ S_0}} \rightarrow \text{Read out}$$

$$\boxed{s_i = p_i \oplus c_i}$$

$$c_{i+1} = p_i c_i + g_i$$

$$\boxed{p_i = a_i \oplus b_i}$$

$$g_i = a_i b_i$$



$$p_0 = A_0 \oplus B_0$$

$$p_1 = A_1 \oplus B_1$$

$$p_2 = A_2 \oplus B_2$$

$$p_3 = A_3 \oplus B_3$$

$$g_0 = A_0 B_0$$

$$g_1 = A_1 B_1$$

$$g_2 = A_2 B_2$$

$$g_3 = A_3 B_3$$

$$s_0 = p_0 \oplus c_0$$

$$s_1 = p_1 \oplus c_1$$

$$s_2 = p_2 \oplus c_2$$

$$s_3 = p_3 \oplus c_3$$

$$c_0 = A_0 B_0$$

$$c_1 = p_0 c_0 + g_0 \quad \text{--- (i)}$$

$$c_2 = p_1 c_1 + g_1$$

$$= (p_0 c_0 + g_0) p_1 + g_1$$

$$= p_1 p_0 c_0 + p_1 g_0 + g_1 \quad \text{--- (ii)}$$

$$c_3 = p_2 c_2 + g_2$$

$$C_3 = P_2 (P_1 P_0 C_0 + P_1 Q_0 + Q_1) + Q_2$$

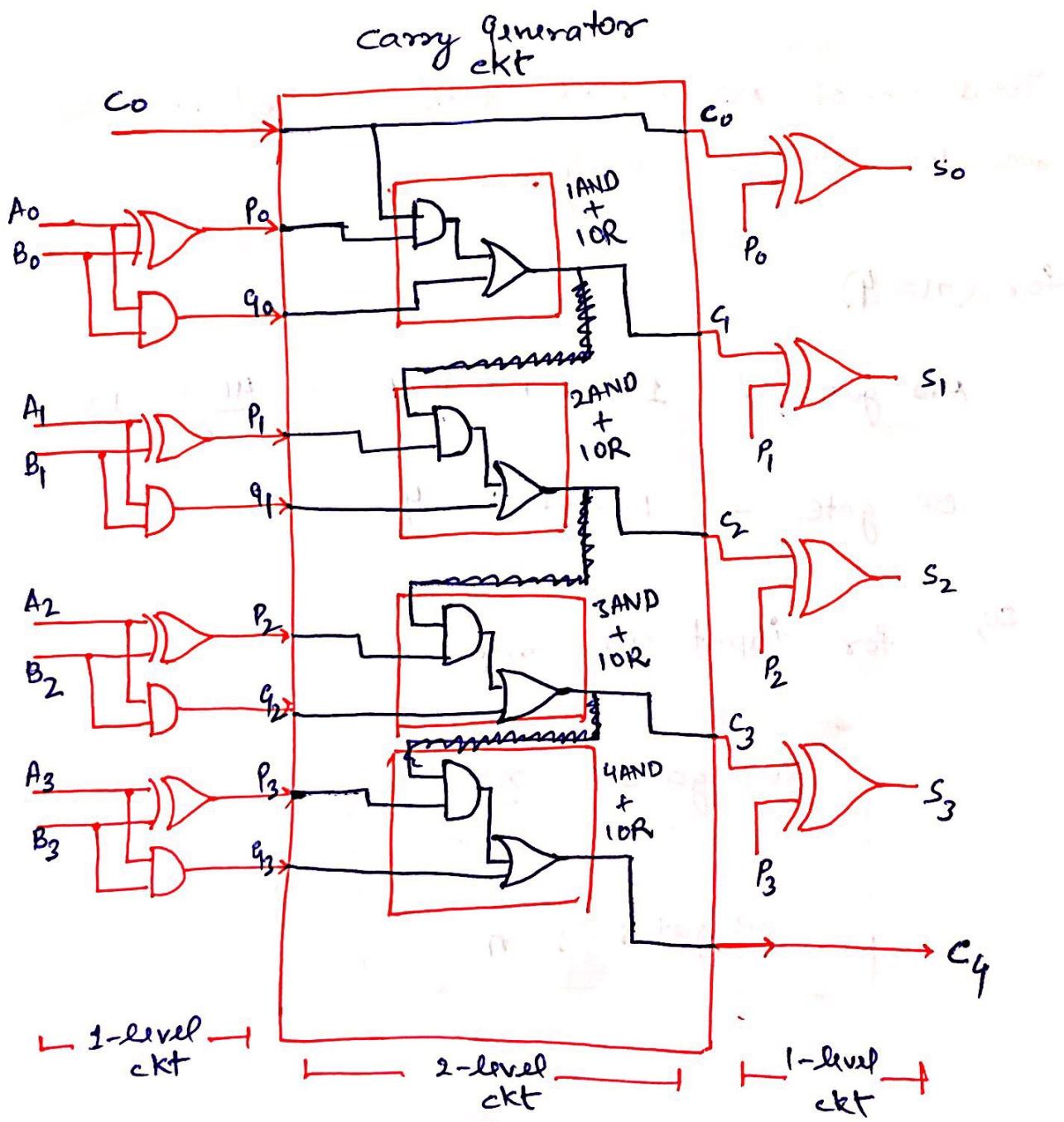
$$= P_2 P_1 P_0 C_0 + P_2 P_1 Q_0 + P_2 Q_1 + Q_2 \quad \text{--- (iii)}$$

$$C_4 = P_3 C_3 + Q_3$$

$$= P_3 (P_2 P_1 P_0 C_0 + P_2 P_1 Q_0 + P_2 Q_1 + Q_2) + Q_3$$

$$= P_3 P_2 P_1 P_0 C_0 + P_3 P_2 P_1 Q_0 + P_3 P_2 Q_1 + P_3 Q_2 + Q_3$$

--- (iv)



Q. Each gate which is used in the circuit is having a propagation delay of  $T_{pd}$ . what is the overall delay of the ckt. (circuit is CLA).

- Since CLA is a 4-level ckt design, so total opn. time will be  $4T_{pd}$ .

- So, opn. delay is independent of input size.

Q. Total no. of AND and OR gates required in carry generator ckt respectively.

for ( $n=4$ )

$$\text{AND gates} : - 1 + 2 + 3 + 4 = \frac{4(5)}{2} = 10$$

$$\text{OR gates} : - 1 + 1 + 1 + 1 = 4$$

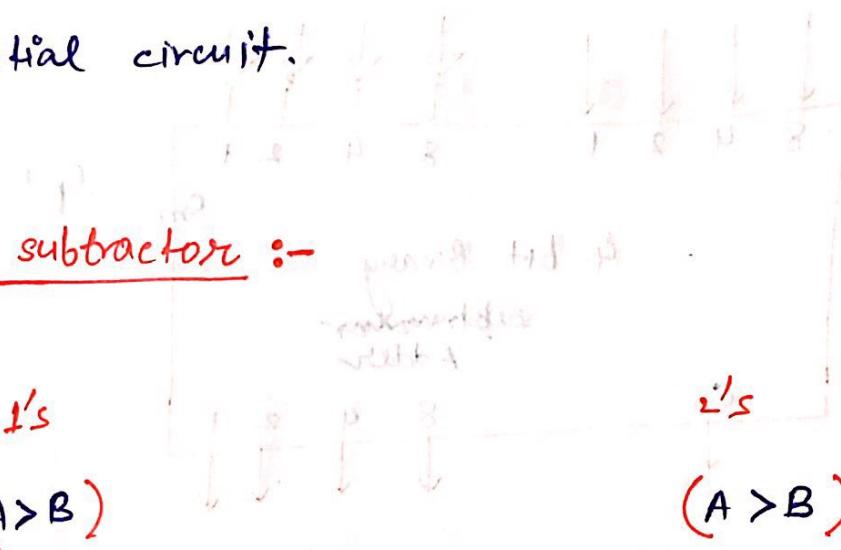
so, for input size  $n$ :

$$\text{AND gates} : \frac{n(n+1)}{2}$$

$$\text{OR gates} : n$$

NOTE:- Serial BA is an example of sequential ckt whereas the other two BA are example of combinational ckt.

As D-Flip Flop are present in SBA so it belongs to sequential circuit.



$$\begin{array}{r}
 A=5 \\
 - B=3 \\
 \hline
 2
 \end{array}
 \qquad
 \begin{array}{r}
 0\ 1\ 0\ 1 \\
 1\ 1\ 0\ 1 \\
 \hline
 1\ 0\ 0\ 1\ 0
 \end{array}$$

discard

True magnitude

$$\begin{array}{r} (A \times B) \\ \hline 0011 \\ 1010 \\ \hline \underline{1101} \end{array}$$

$$\begin{array}{r} A = 3 \\ B = 5 \\ - \\ \hline -2 \end{array}$$

- (0010)

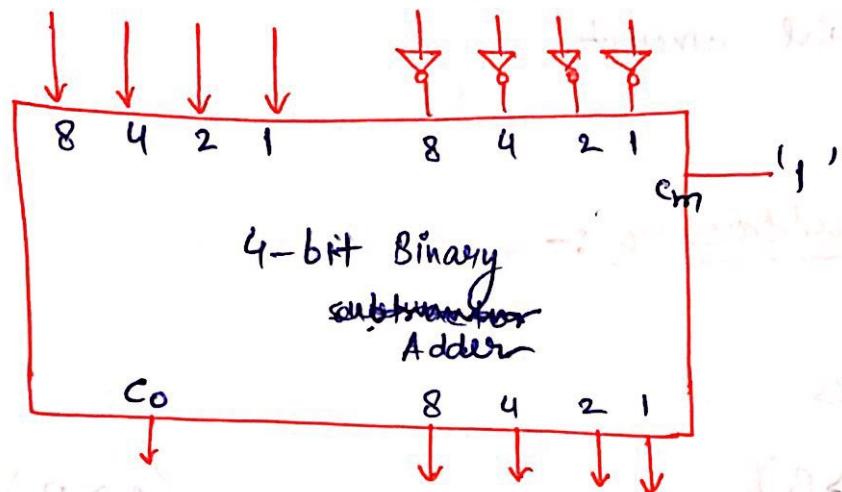
$$\begin{array}{r}
 0011 \\
 1011 \\
 \hline
 1110
 \end{array}$$

↓

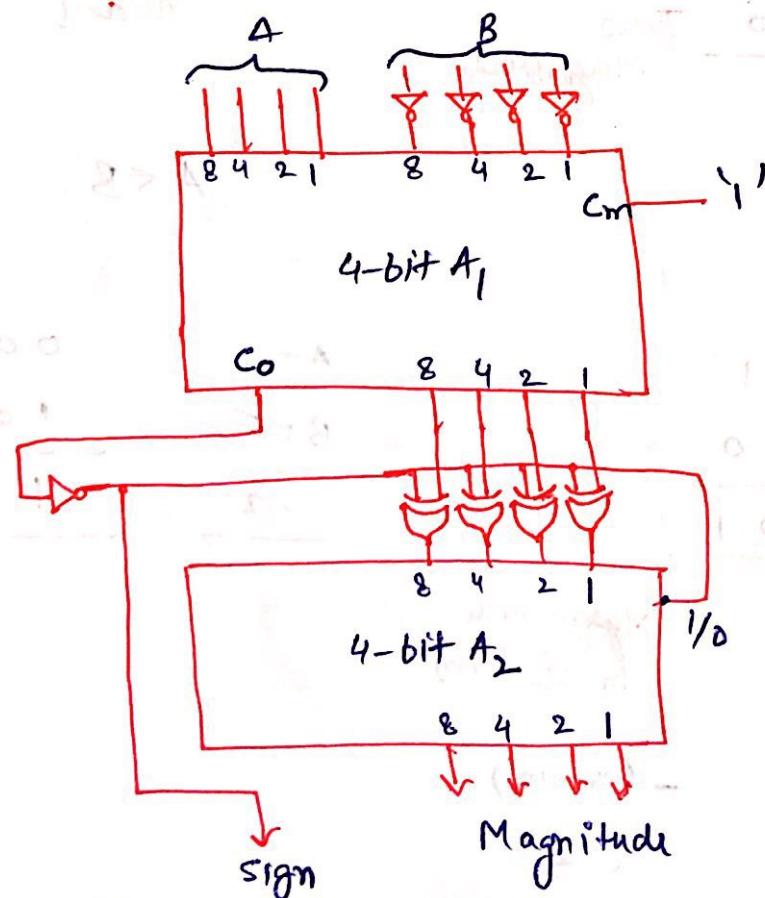
Magnitude  
in 2's form

- (0010)

Q. By using 4-bit binary adder & other required logic gates, realize 4-bit Binary subtractor circuit using 2's complement procedure.



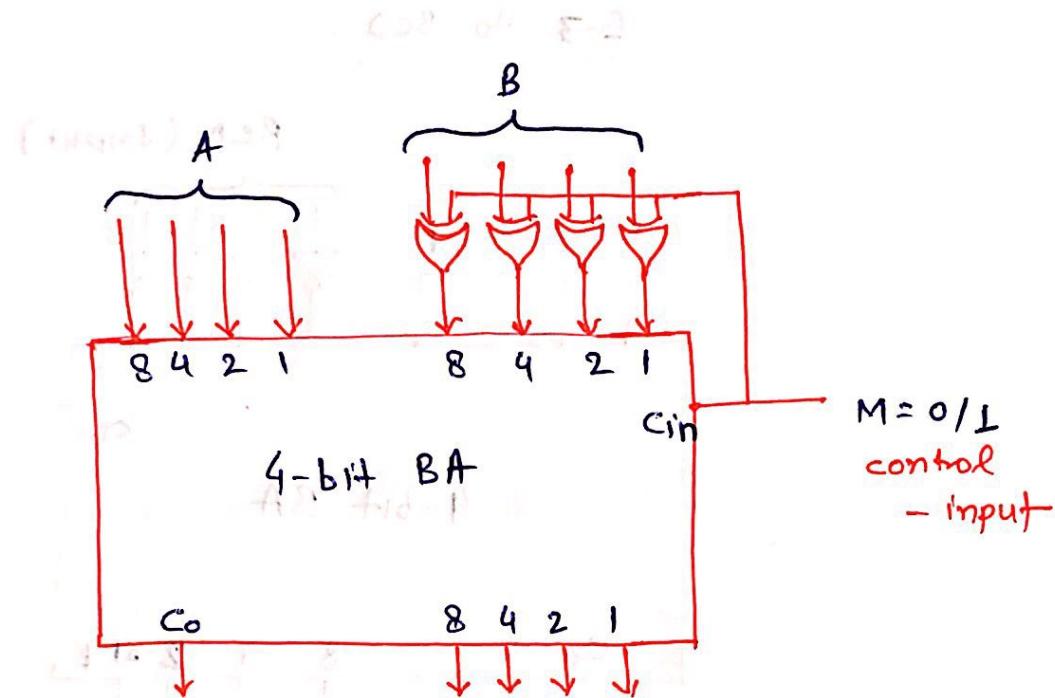
- In a Binary subtractor ckt the result must true magnitude along with the sign bit.
- So, we need 4-bit adder ckt and a 2's ckt.



Q By using 4-bit binary - adder & other, required minimum no. of logic gates, realize -

- controllable 4-bit binary adder/ subtractor
- controllable BCD to E-3 / E-3 to BCD
- BCD to 9's complement

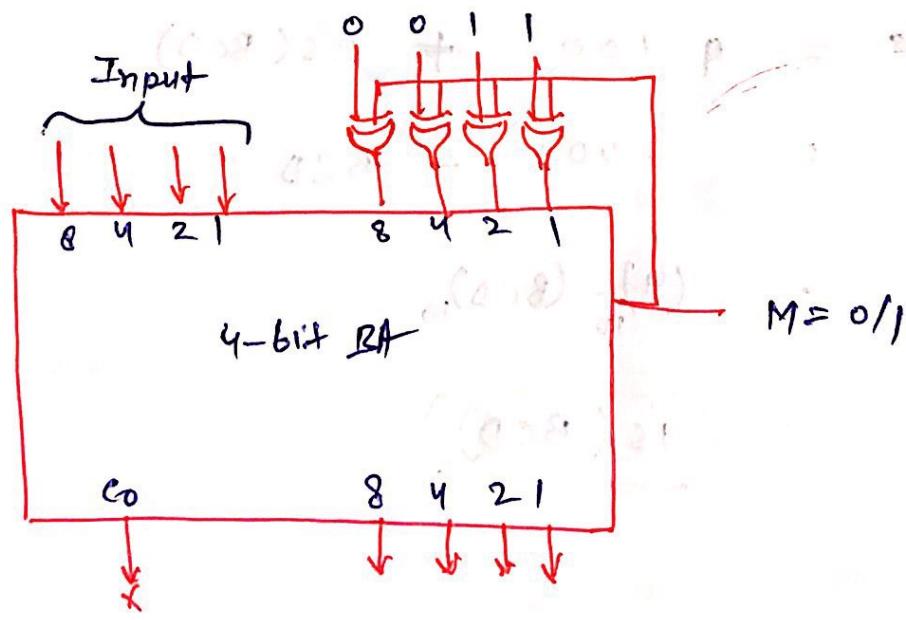
(a)



when  $M=0$  (MSB of B) then addition takes place

"  $M=1$  then subtraction takes place

(b)

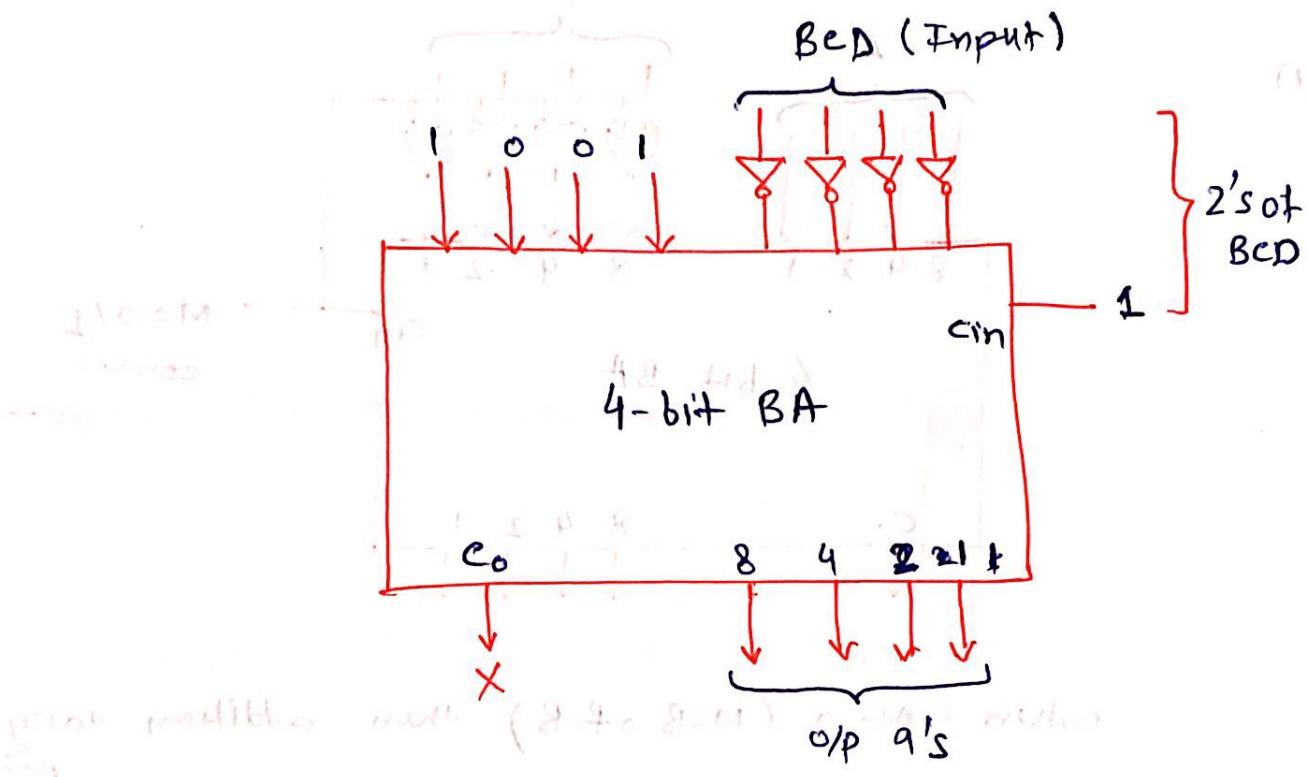


if ( $M=0$ )      Input + 3  
BCD to E-3

if ( $M=1$ )      Input + 2's of 3  
 Input - 3

E-3 to BCD

(c)



$$* = 9 \ 1001 + 2's(BCD)$$

$$= 1001 - BCD$$

$$= (9)_{10} - (BCD)_{10}$$

$$= \boxed{9's(BCD)}$$

## BCD - Adder :-

$$\begin{array}{r} A = 8 \\ + B = 5 \\ \hline 13 \end{array}$$

BCD output =  $\begin{array}{r} 1 \\ \hline 0001 \end{array} \quad \begin{array}{r} 3 \\ \hline 0011 \end{array}$

$$\begin{array}{r} A: 1000 \\ + B: 0101 \\ \hline 1101 \leftarrow \text{BCD} \end{array}$$

ADD 6

$$\begin{array}{r} 1101 \\ 0110 \\ \hline \boxed{1} \boxed{0011} \\ \text{carry } 10^1 \quad 10^0 \end{array}$$

$$\begin{array}{r} A = 9 \\ + B = 8 \\ \hline 17 \end{array}$$

carry  $10^1$

ADD 6

$$\begin{array}{r} 1001 \\ 1000 \\ \hline \boxed{1} 0001 \\ \text{carry } 10^1 \end{array} \quad \begin{array}{r} 00001 \\ 0110 \\ \hline 0111 \\ 10^0 \end{array}$$

$$\begin{array}{r} A = 5 \\ + B = 3 \\ \hline 8 \end{array}$$

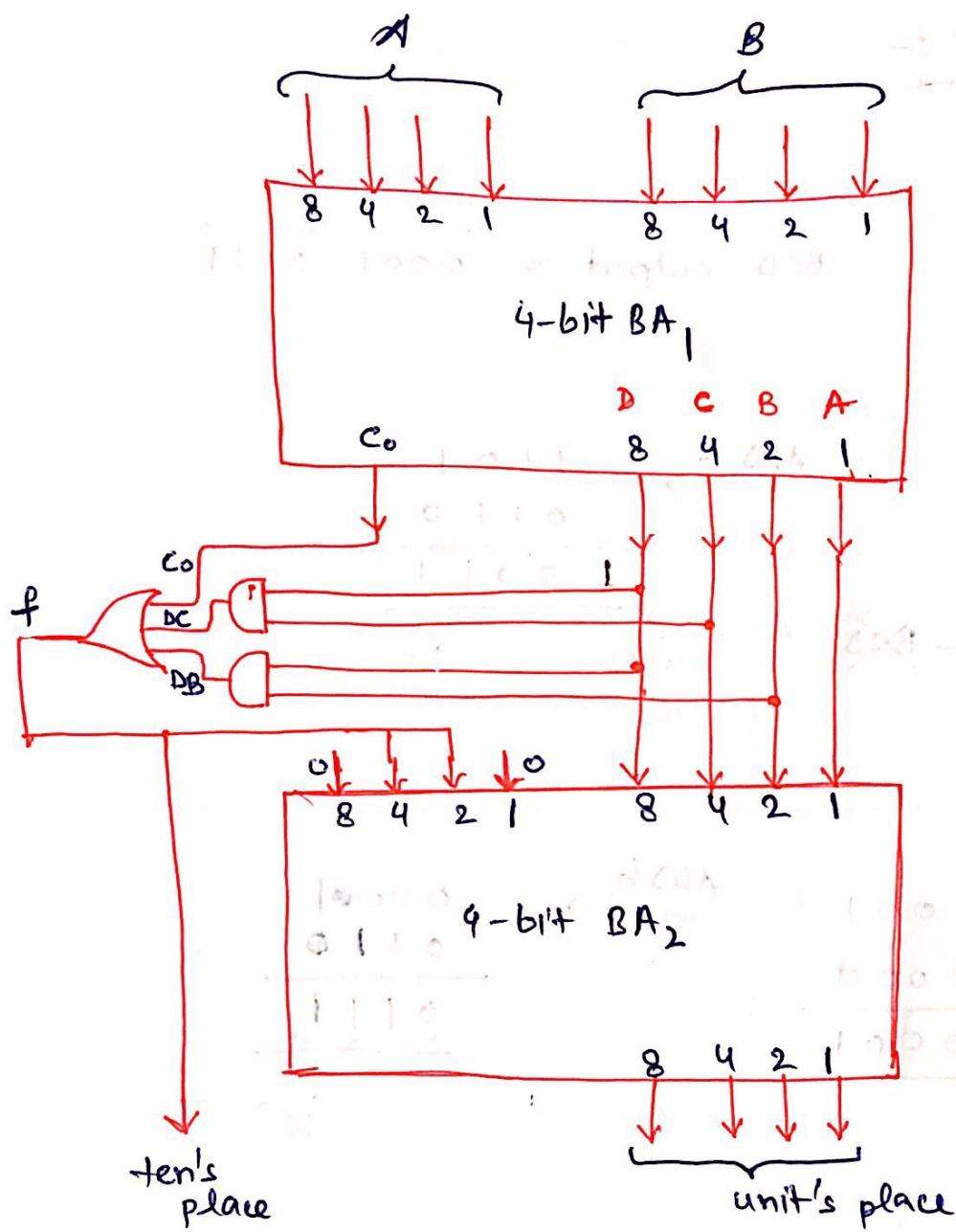
BCD

checking ckt for  
BCD valid or  
invalid:  
if  $f=1$ ; ADD 6  
else if  $f=0$ ; ADD 0

Implement BCD - Adder using 4-bit Binary Adder :-

$$f = C_0 + f_1$$

carry of  $A+B$       cal from sum-bits (DCBA)



Invalid BCD:

	$D \ C \ B \ A$	$f_1$
10	1 0 1 0	1
11	1 0 1 1	1
12	1 1 0 0	1
13	1 1 0 1	1
14	1 1 1 0	1
15	1 1 1 1	1

$$f_1 = DC + DB$$

$$\text{so, } f = C_0 + f_1$$

$$f = C_0 + DC + DB$$

control bit

## BCD Subtractor :-

$$\begin{array}{r}
 A \\
 - B \\
 \hline
 \end{array}
 \quad \text{or} \quad
 \begin{array}{r}
 A \\
 + 9's(B) \\
 \hline
 \end{array}
 \Rightarrow
 \begin{array}{l}
 = A - B \\
 = (A + 9's(B))
 \end{array}$$

$$\begin{array}{r}
 A = 6 \\
 - B = 4 \\
 \hline
 2
 \end{array}
 \quad \text{so, } A = 6 \quad 9's(B) = 5 \quad \Rightarrow \quad
 \begin{array}{r}
 0110 \\
 0101 \\
 \hline
 1011
 \end{array}
 \leftarrow \text{BCD X (Invalid)}$$

carry = 1

$$\begin{array}{r}
 + 6 \\
 \hline
 0110 \\
 10001 \\
 \hline
 0010
 \end{array}
 \quad \text{Add End carry}$$

if  $A > B$

$$\begin{array}{r}
 A = 9 \\
 - B = 1 \\
 \hline
 \end{array}
 \quad \text{so, } \quad
 \begin{array}{r}
 1001 \\
 1000 \\
 \hline
 10001
 \end{array}
 \quad \text{carry = 1} \quad \text{Invalid BCD}$$

carry

$$\begin{array}{r}
 0001 \\
 + 0110 \\
 \hline
 0111
 \end{array}
 \quad \xrightarrow{\quad + 1 \quad} \quad
 \begin{array}{r}
 \hline
 1000
 \end{array}$$

if  $A < B$

$$\begin{array}{r}
 A = 4 \\
 - B = 6 \\
 \hline
 -2
 \end{array}
 \quad
 \begin{array}{r}
 0100 \\
 0011 \\
 \hline
 0111
 \end{array}
 \quad \text{carry = 0} \quad \text{Valid BCD (Magnitude is in 9's form)}$$

True magnitude

$$\begin{array}{r}
 \hline
 0010
 \end{array}$$

Implement using Binary adder, BCD adder, a's IC:

- (i) a's ckt
- (ii) BCD adder
- (iii) Binary adder

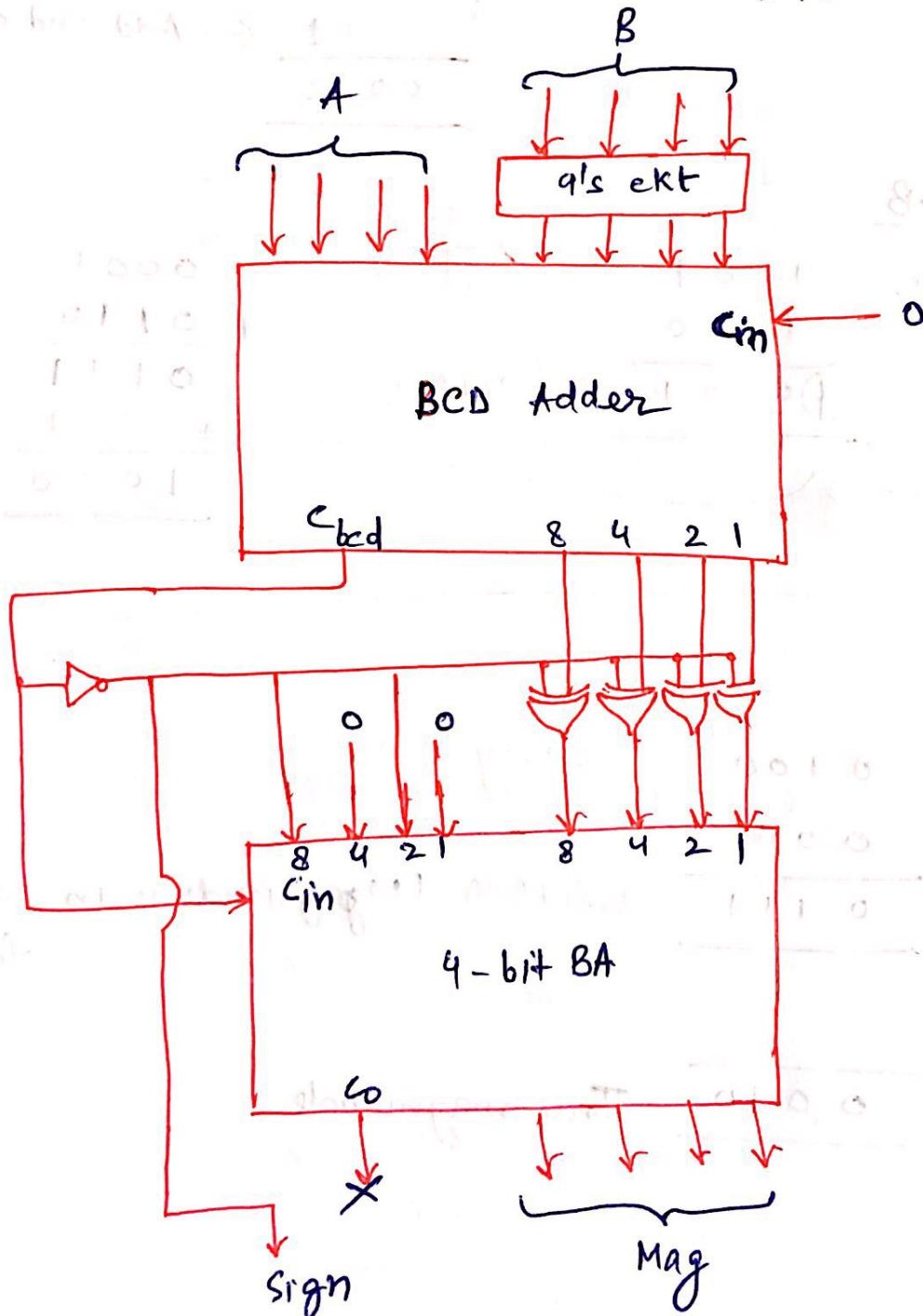
if Carry from BCD ( $C_{bcd} = 0$ )  
i.e.  $A < B$

a's ckt

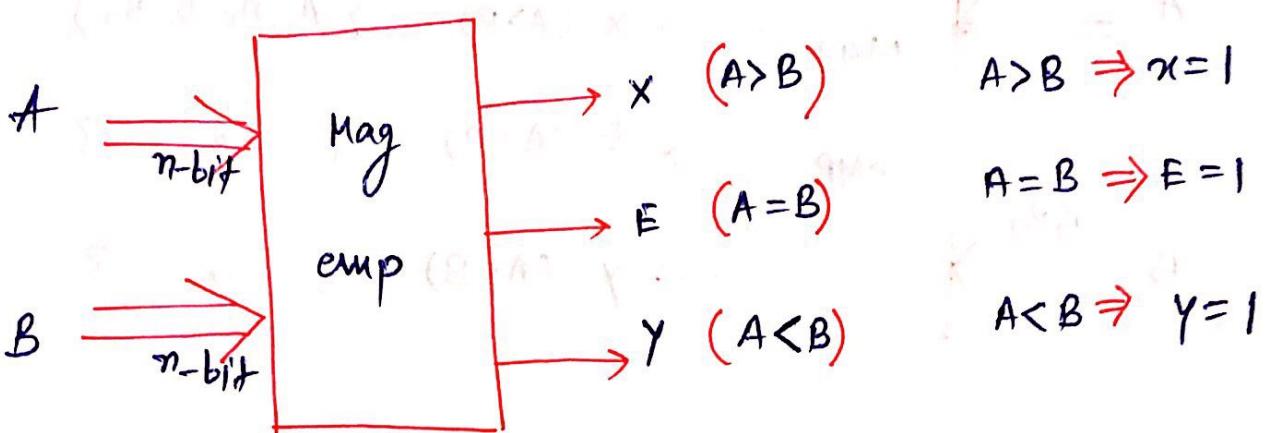
4-bit BA

if ( $C_{bcd} = 1$ ) i.e.  $A > B$

Add 1



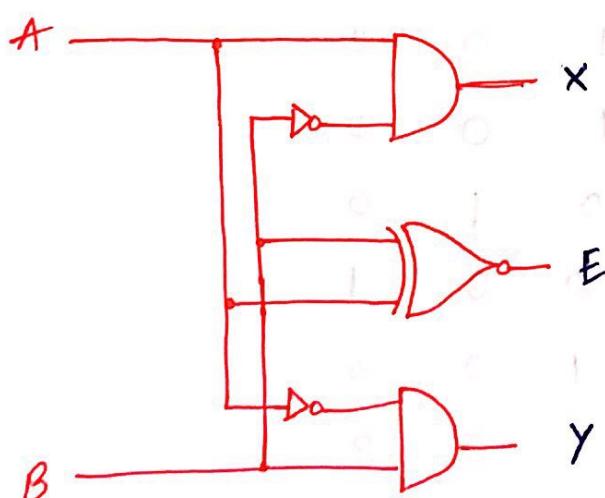
## Magnitude comparator :-



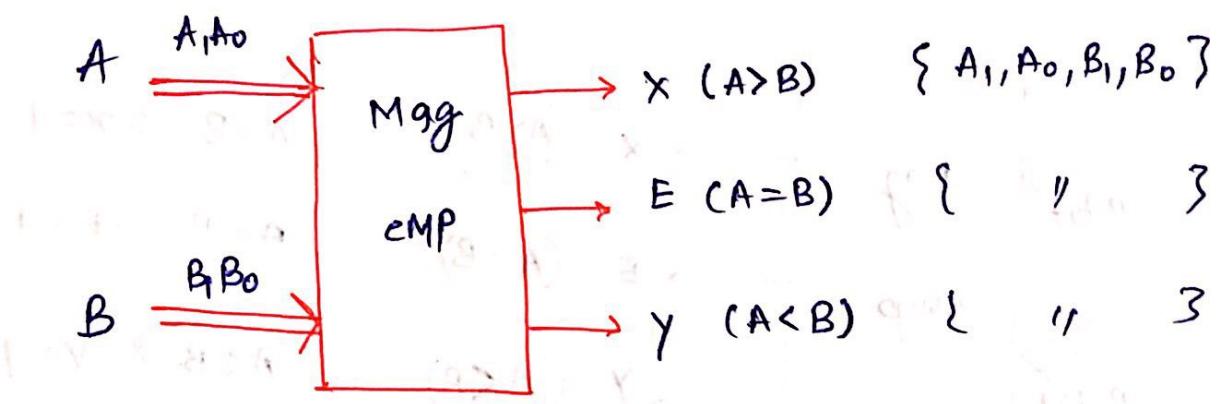
For ( $n = 1$ ) :-

A	B	X	E	Y
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

$$\boxed{\begin{aligned} E &= A \oplus B \\ X &= \bar{A}B \\ Y &= \bar{A}\bar{B} \end{aligned}}$$

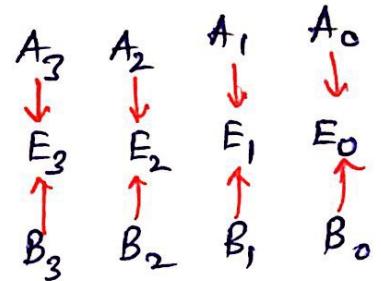
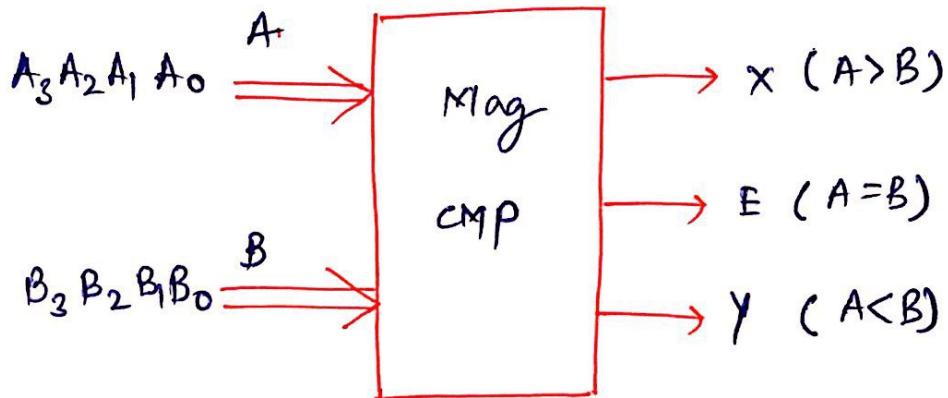


$\Rightarrow$  for  $(n=2)$  :-

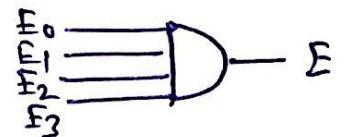


$A_1$	$A_0$	$B_1$	$B_0$	$X$	$E$	$Y$
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

$\Rightarrow$  for ( $n=4$ ):-



$$E = E_3 \cdot E_2 \cdot E_1 \cdot E_0$$



$$(A = B) :- \underline{E = E_0 \cdot E_1 \cdot E_2 \cdot E_3}$$

$(A > B) :-$  if  $A_3 > B_3$

or  $A_3 = B_3$  but  $A_2 > B_2$

or  $A_2 = B_2$  but  $A_1 > B_1$   
 $A_3 = B_3$

or  $A_1 = B_1$  but  $A_0 > B_0$   
 $A_2 = B_2$   
 $A_3 = B_3$

$$\underline{X = A_3 \bar{B}_3 + E_3 A_2 \bar{B}_2 + E_3 E_2 A_1 \bar{B}_1 + E_3 E_2 E_1 A_0 \bar{B}_0}$$

$$(A < B) :- \underline{Y = \bar{A}_3 B_3 + E_3 \bar{A}_2 B_2 + E_3 E_2 \bar{A}_1 B_1 + E_3 E_2 E_1 \bar{A}_0 B_0}$$