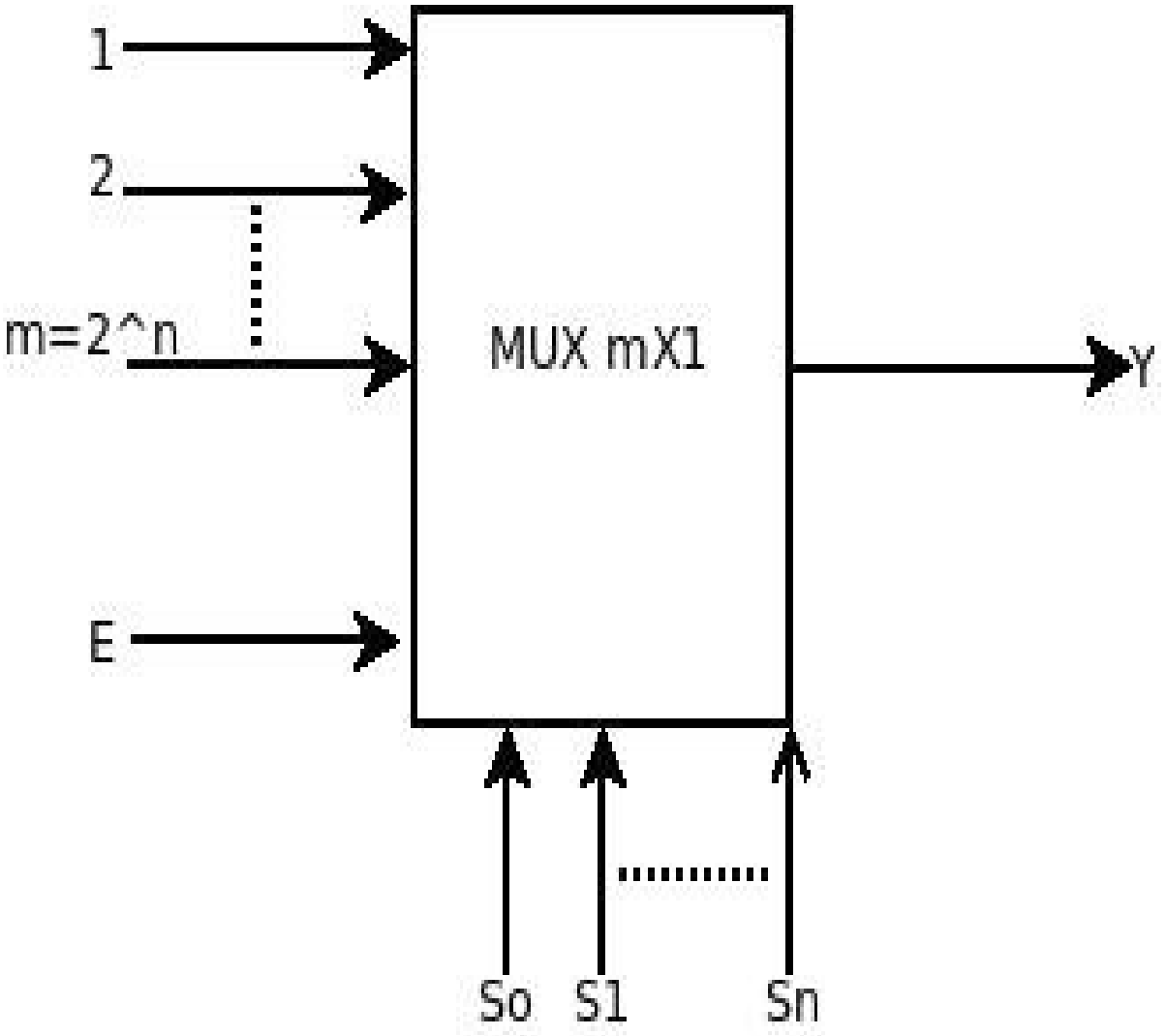
ADE- UNIT IV
Combination Circuits

Topics
1. Multiplexer
2. Demultiplexer
3. Encoder
4. Decoder

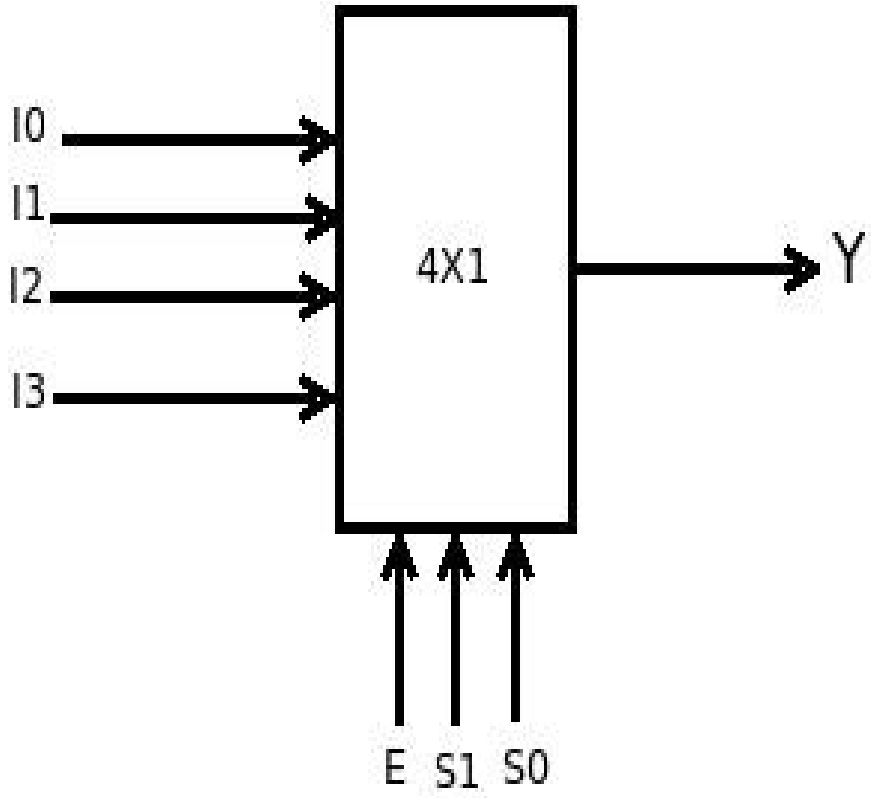
Intoduction and implementation ,Tree & Logicatical circuit implementation

Multiplexer

- Electronic device consist of
- 2^m input terminals
- One output output terminals
- M numbers of selection line


1
2
m=2^n
E
MUX mX1
Y
So
S1
Sn

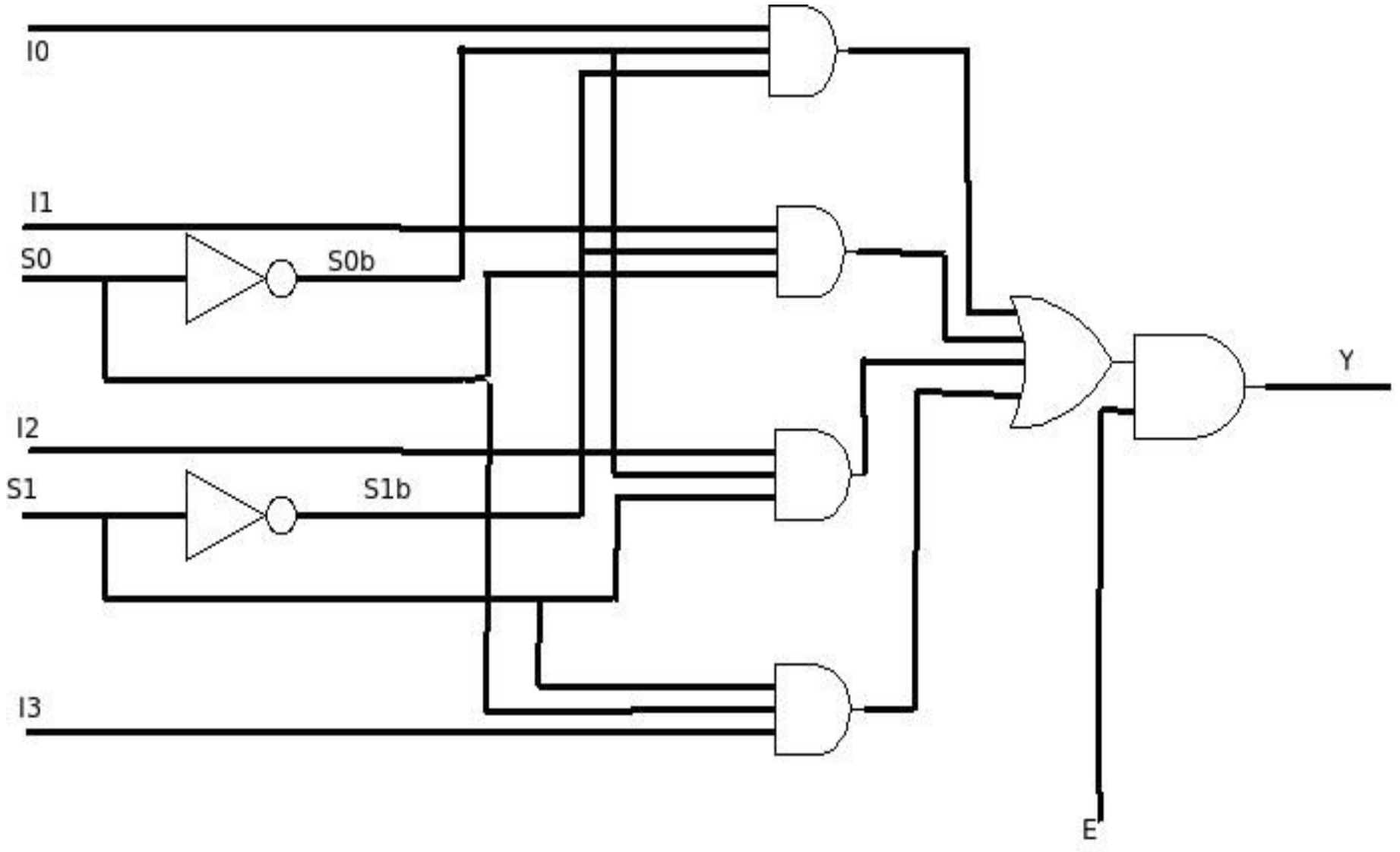
MUX 4X1



E	S1	S0	Y
1	0	0	I0
1	0	1	I1
1	1	0	I2
1	1	1	I3
0	X	X	0

$$Y = E(S1b\ S0b\ I0 + S1b\ S0\ I1 + S1\ S0b\ I2 + S0\ S1\ I3)$$

Logical Implementation



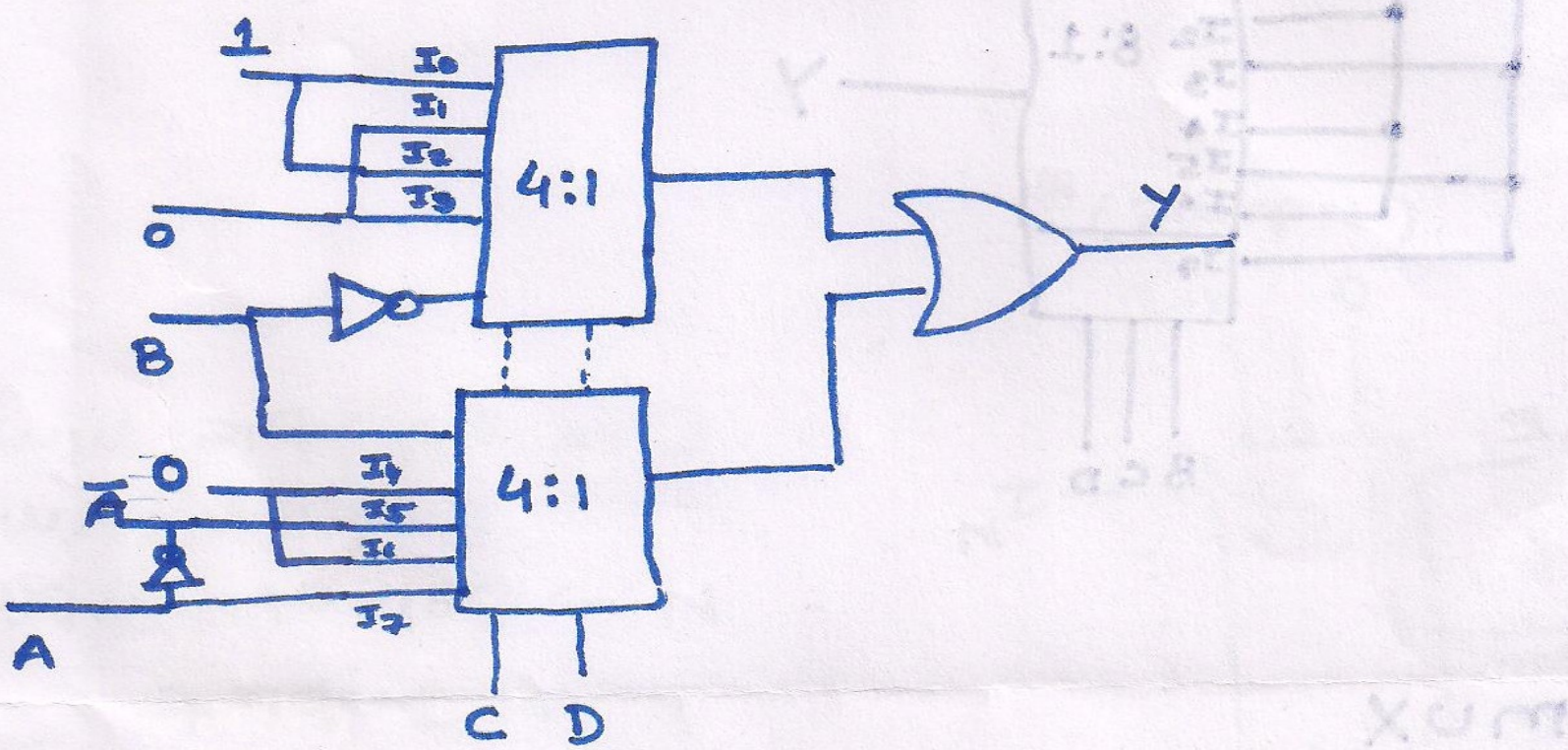
①

Implement the function using 4:1 mux

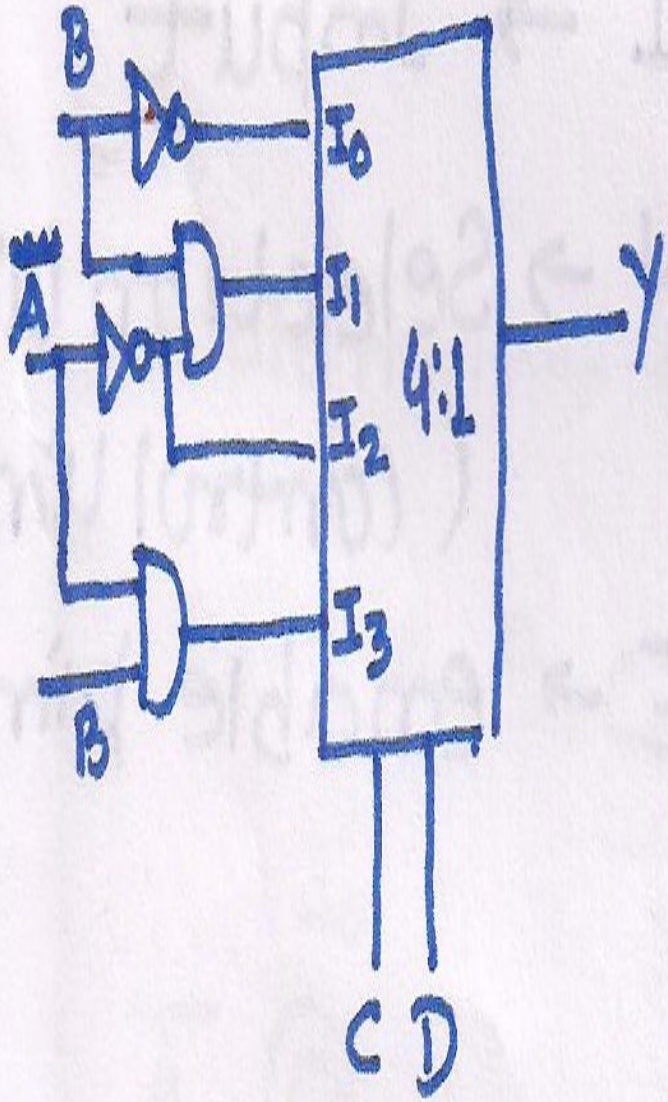
$$F(A,B,C,D) = \Sigma(0,2,5,8,10,15)$$

	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
$\bar{A}$	(0)	1	(2)	3	4	(5)	6	7
A	(8)	9	(10)	11	12	13	14	(15)

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



(2) 2nd way



$\bar{A}\bar{B}$	(0)	1	(2)	3
$\bar{A}B$	4	(5)	6	7
$A\bar{B}$	(8)	9	(10)	11
AB	12	13	14	(15)

$\bar{B}\bar{A}$ $\bar{A}B$ $\bar{A}\bar{B}$ AB

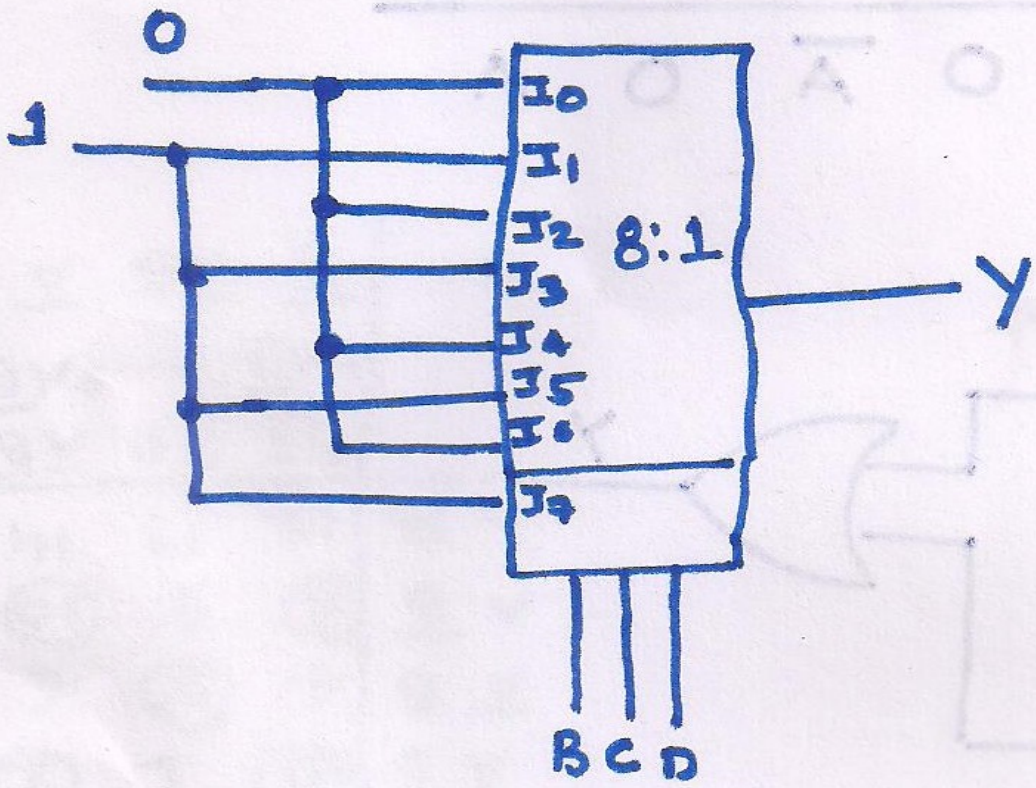
$+A\bar{B}$ $+A\bar{B}$

$= \bar{B}$ $\bar{A}B$ $\bar{A}$ AB

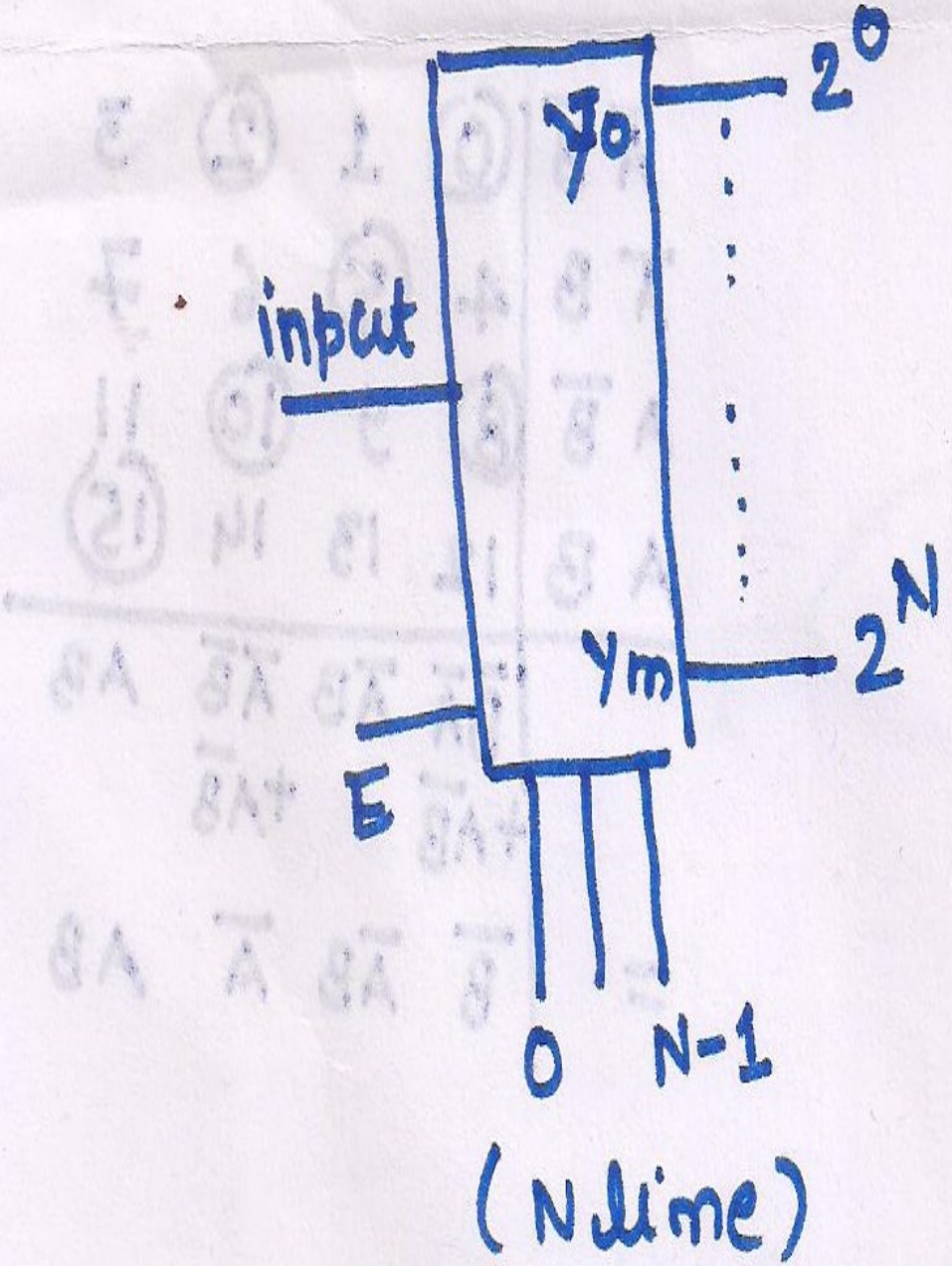
③

a $\pi(0,2,4,6,8,10,12,14)$ mux (8:1)

	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
$\bar{A}$	0	(1)	2	(3)	4	(5)	6	(7)
A	8	(9)	10	(11)	12	(13)	14	(15)
	0	1	0	1	0	1	0	1



(31) DEMUX



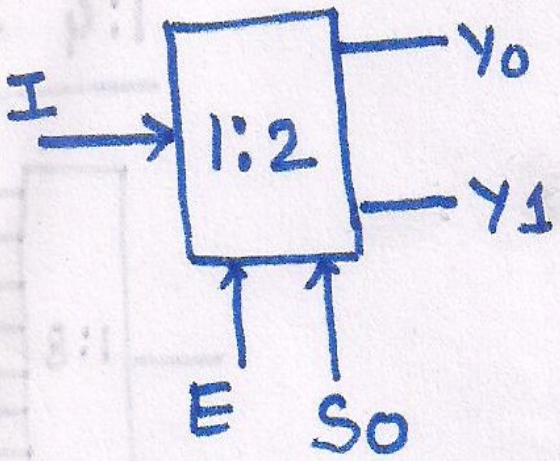
2^N → output

1 → Input

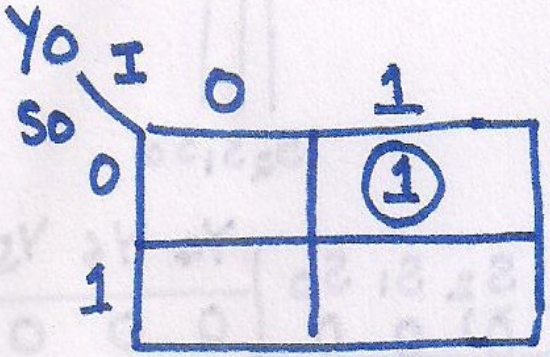
N → Selection line
(control line)

E → enable pin

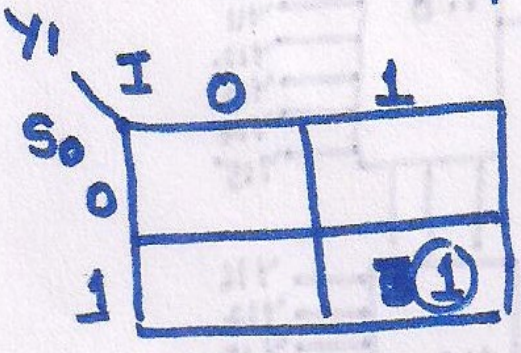
1:2 Demux



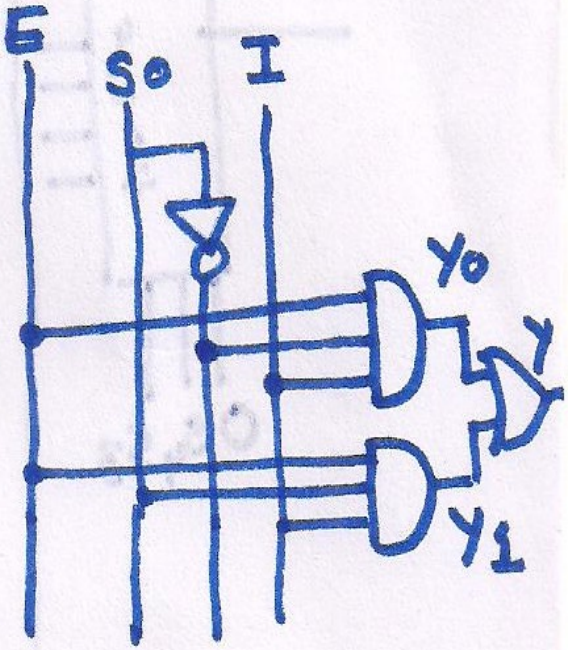
E	I	S_0	Y_1	Y_0
0	X	X	0	0
1	0	0	0	0
1	1	0	0	1
1	0	1	0	0
1	1	1	1	0



$Y_0 = \bar{S}_0 I$

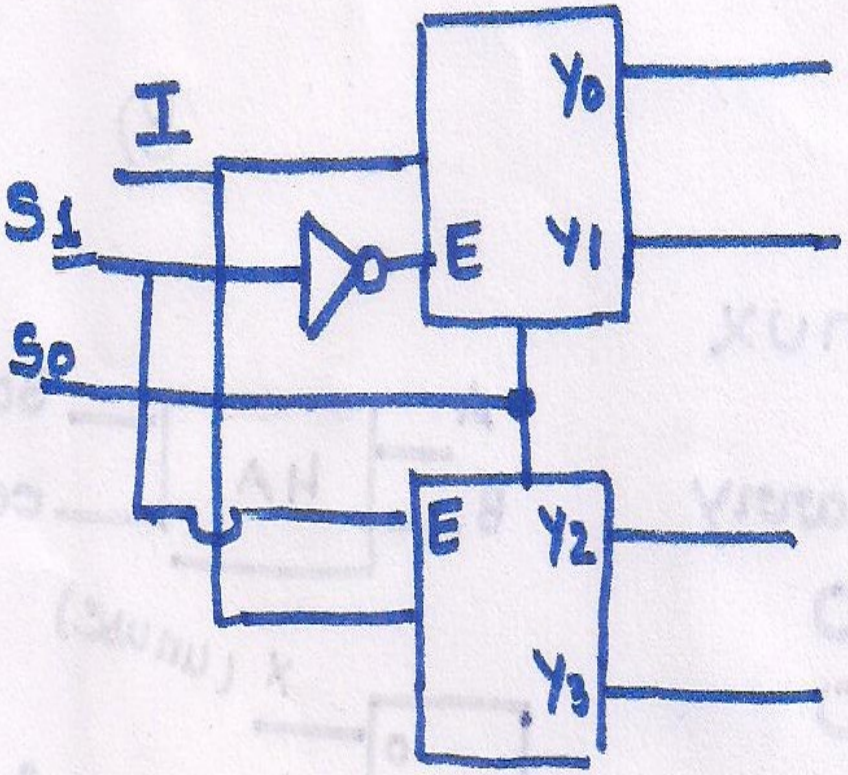


$Y_1 = S_0 I$



DEMUX TREE

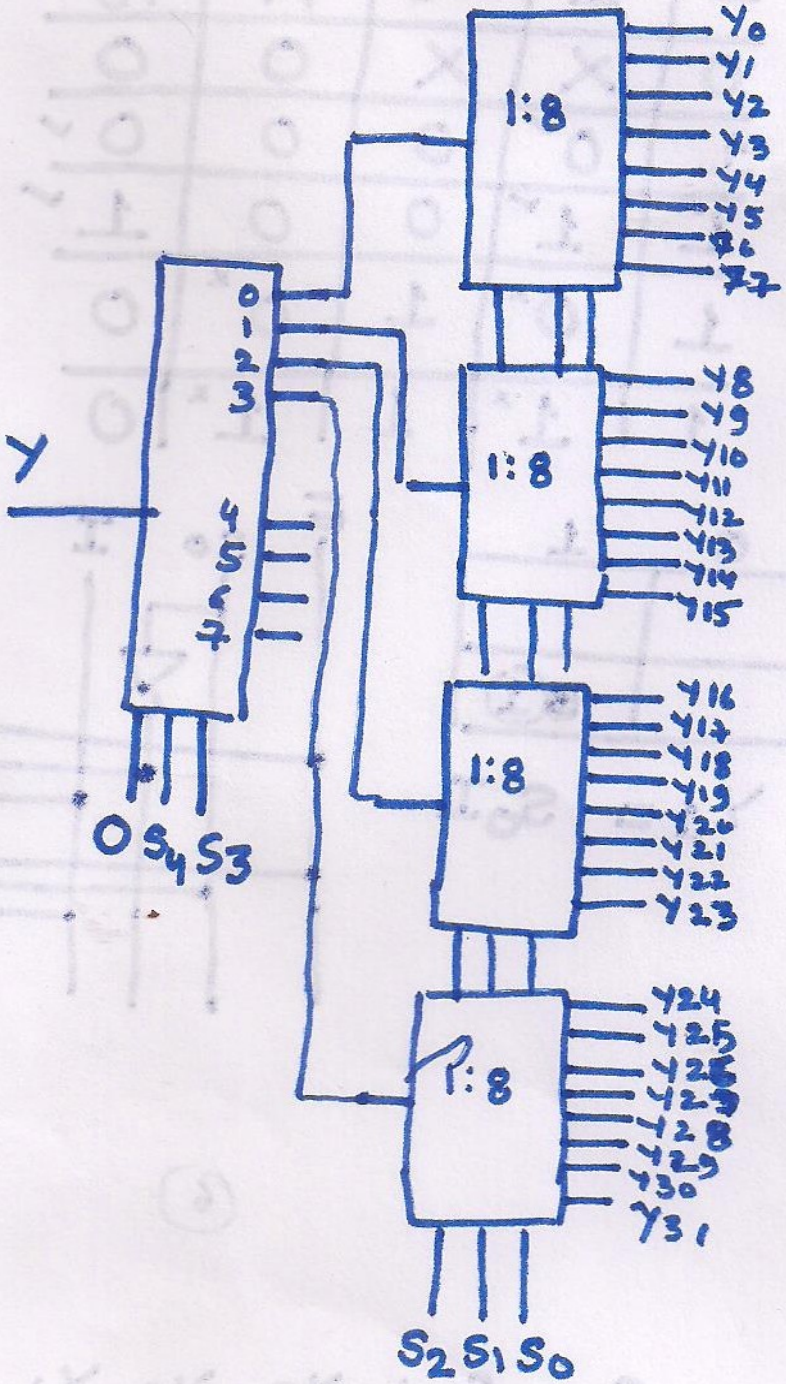
1:4 using 1:2



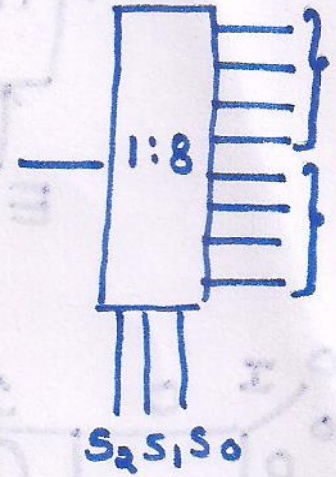
S_1	S_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	I
0	1	0	0	I	0
1	0	0	I	0	0
1	1	I	0	0	0

Q:- Implement 1:8 using 1:4

Q:- 1:32 using 1:8



1:4 from 1:8



S_2	S_1	S_0	Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	0	0	0	I
0	0	1	0	0	0	0	0	0	I	0
0	1	0	0	0	0	0	0	I	0	0
0	1	1	0	0	0	0	I	0	0	0
1	0	0								
1	0	1								
1	1	0								
1	1	1								

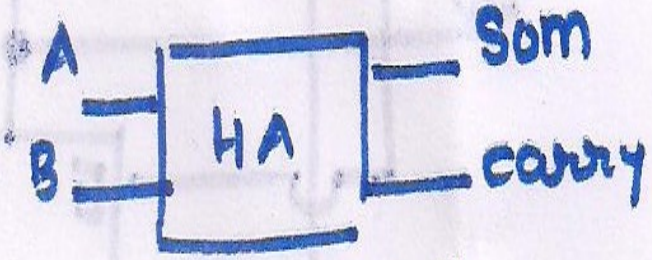
Q:- Compare mux & DEmux

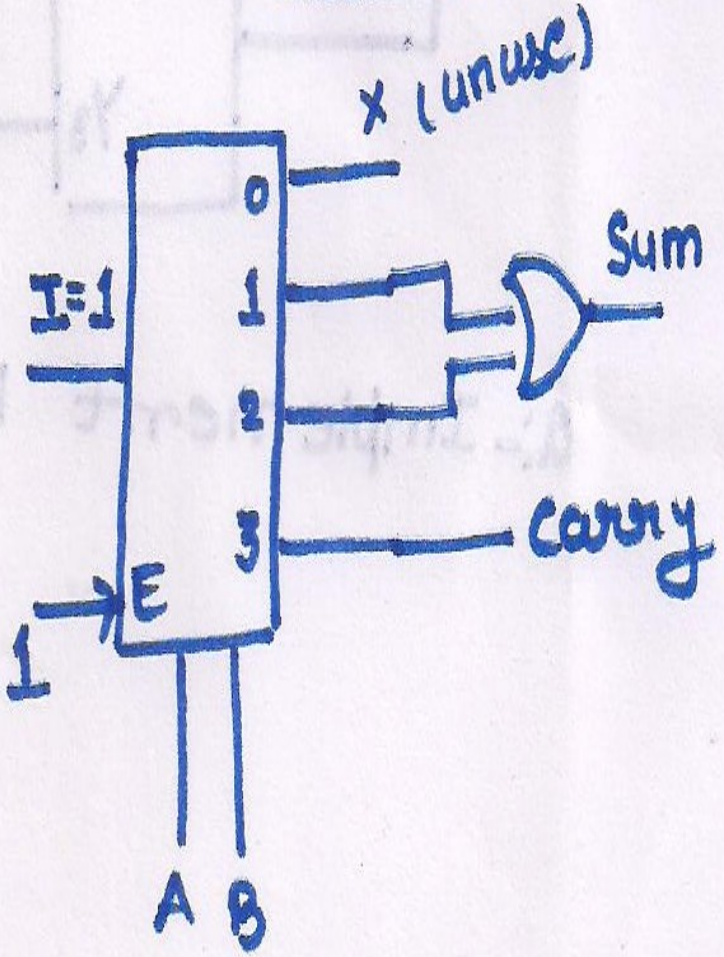
Half adder

A	B	Sum	carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$Sum = \Sigma(1,2)$

$Carry = \Sigma(3)$



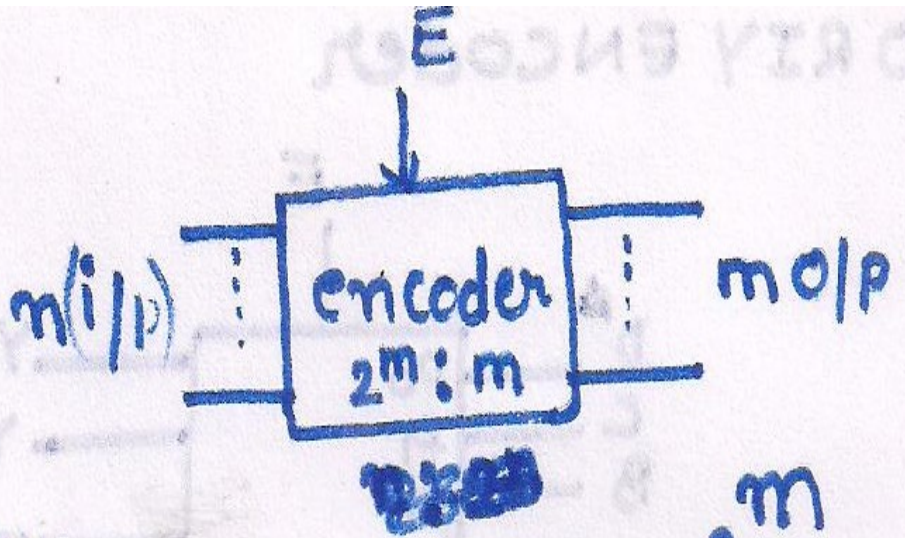


Do same with mux

ENCODERS

n input

m output



$n = 2^m$

1. Octal to binary
2. Decimal to BCD
3. Hexadecimal-to binary
4. Priority encoder

1. $2^3 = 8$ (octal) $(D_0, D_1, \ldots D_7)$

   3 bit binary $(B_2, B_1 B_0)$

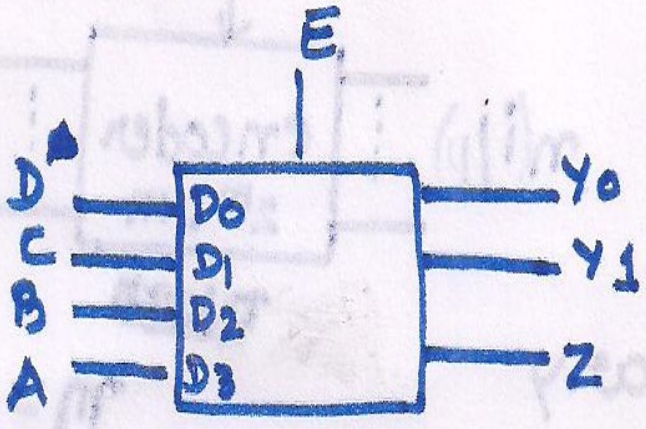
D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	B_2	B_1	B_0
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

$$B_2 = D_4 + D_5 + D_6 + D_7$$

$$B_1 = D_2 + D_3 + D_6 + D_7$$

$$B_0 = D_1 + D_3 + D_5 + D_7$$

PRIORIY ENCODER



Let A > B > C > D

Y_1Y_0	Y_1Y_0	Y_1Y_0	Y_1Y_0
11	10	01	00

	Y_1	Y_0	
0000	X	X	Z = 1

no choice are available

A	B	C	D	Y_1	Y_0
0	0	0	1	0	0

if only D is available

0	0	1	0	0	1

Only C avaiable

0	1	1	0	

??.

B & C available

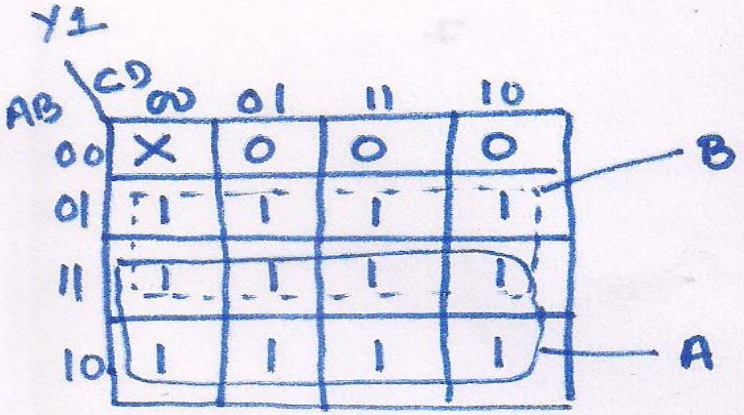
B > C ⇒ 0 1 1 0 | 10

A	B	C	D	Y_1	Y_0	Z
0	0	0	0	X	X	1
0	0	0	1	0	0	0
0	0	1	0	0	1	0
0	0	1	1	0	1	0
0	1	0	0	1	0	0
0	1	0	1	1	0	0
0	1	1	0	1	0	0
0	1	1	1	1	0	0
1	0	0	0	1	1	0
1	0	0	1	1	1	0
1	0	1	0	1	1	0
1	0	1	1	1	1	0
1	1	0	0	1	1	0
1	1	0	1	1	1	0
1	1	1	0	1	1	0
1	1	1	1	1	1	0

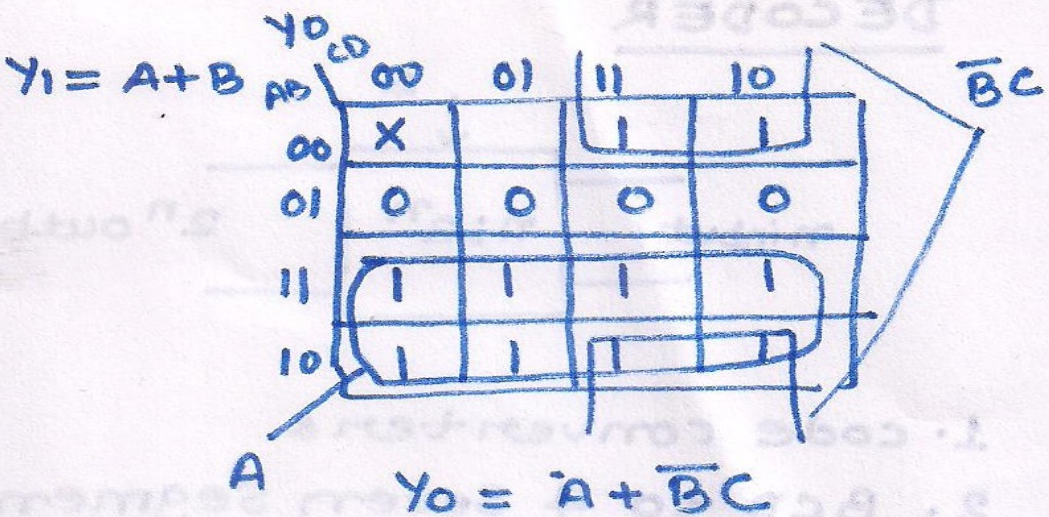
no choice available

A	B	C	D	Y_1	Y_0	Z
0	0	0	0	X	X	1
⇒ 0	0	0	1	0	0	0
0	0	1	X	0	1	0
0	1	X	X	1	0	0
1	X	X	X	1	1	0

5



$Y_1 = A + B$



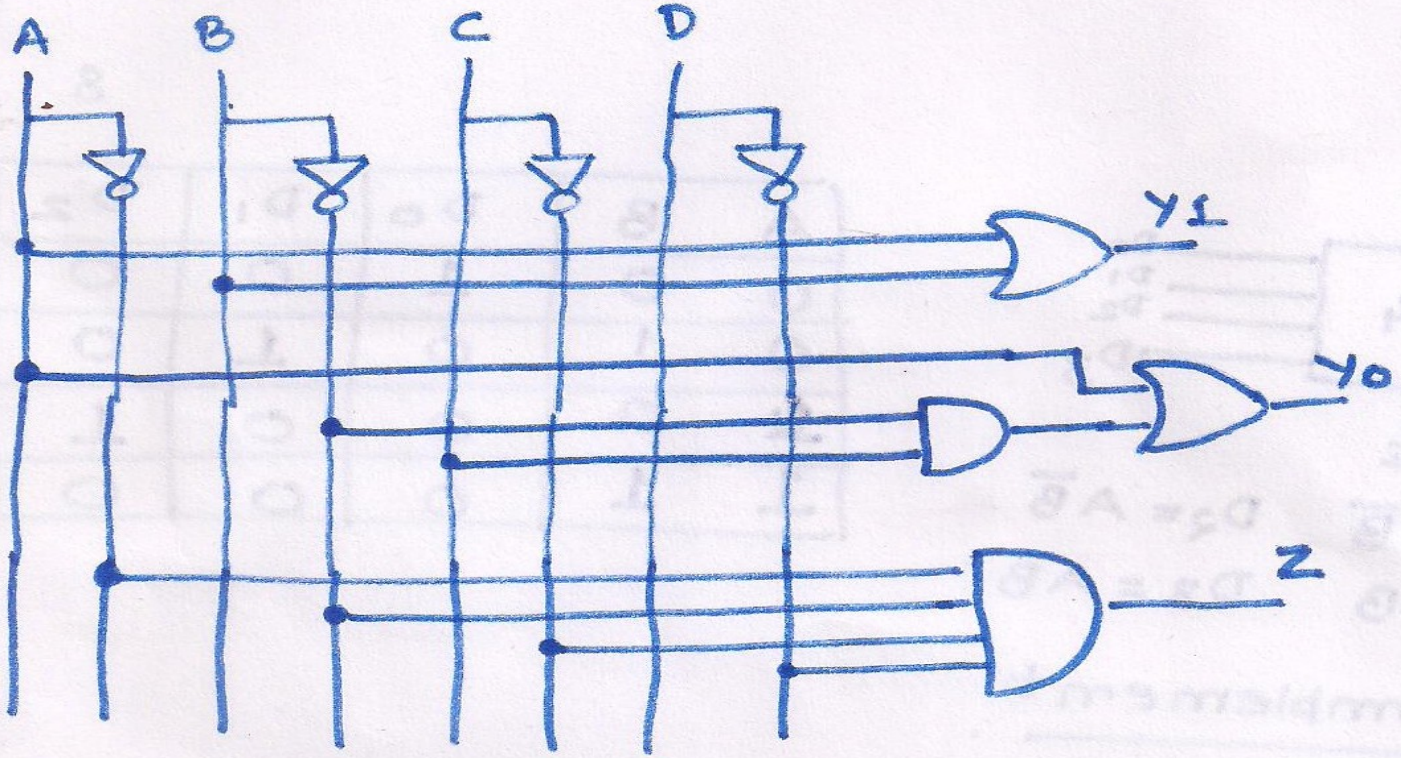
$Y_0 = A + \overline{B}C$

$Y_1 = A + B$

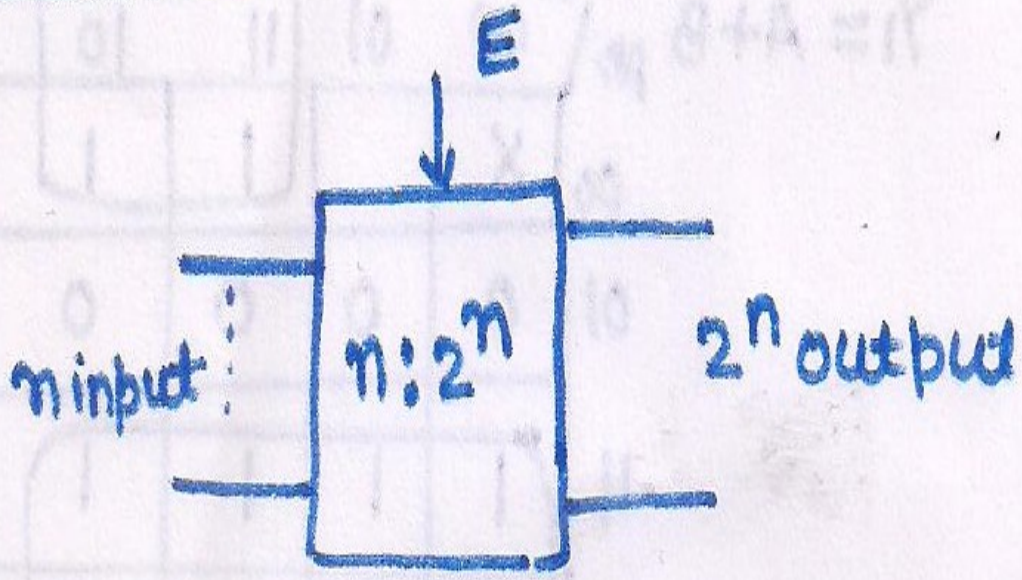
$Y_0 = A + \overline{B}C$

$Z = \overline{A}\,\overline{B}\,\overline{C}\,\overline{D}$

6

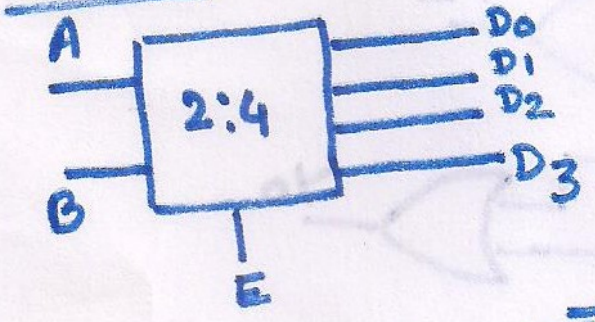


DECODER



1. code converters
2. BCD to 7 Seven Segment

1. **2 : 4**



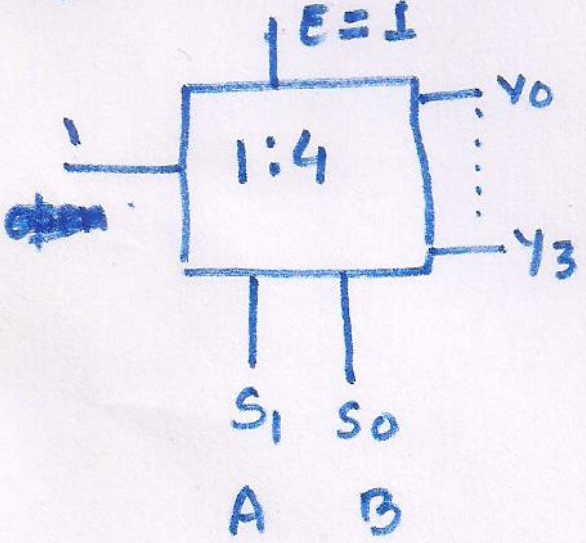
A	B	D_0	D_1	D_2	D_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

$D_0 = \bar{A}\bar{B}$ $D_2 = A\bar{B}$

$D_1 = \bar{A}B$ $D_3 = AB$

Just implement

1 : 4 Dmux as De coder

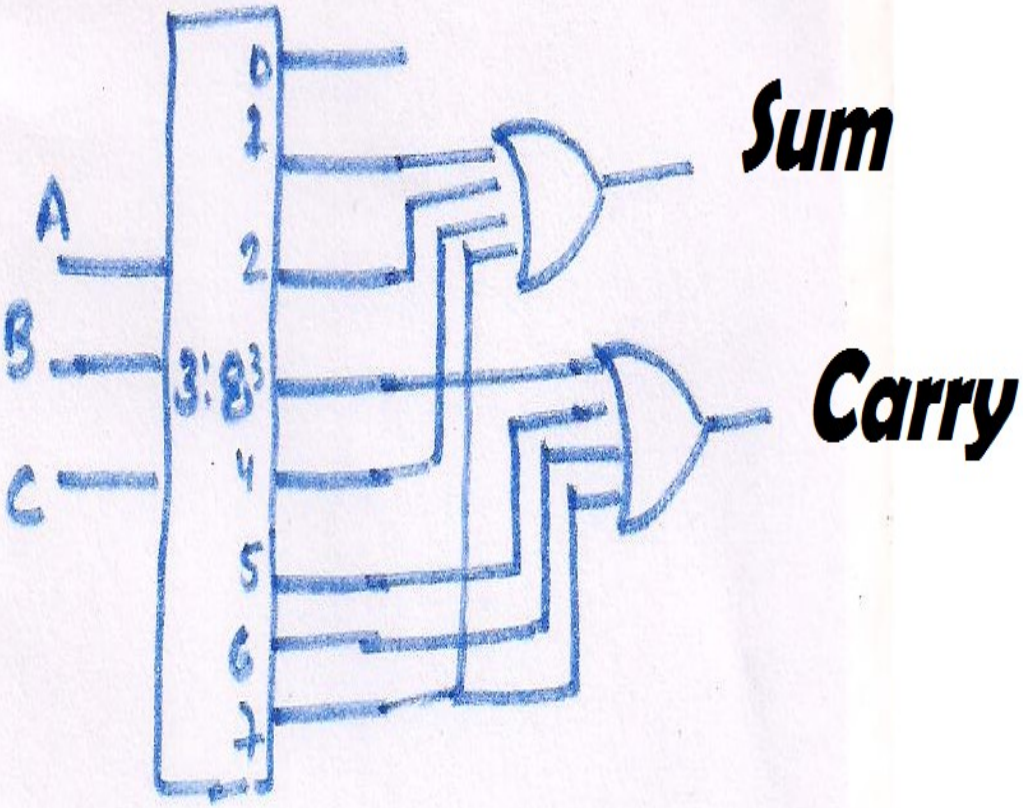


Fulladder using 3:8

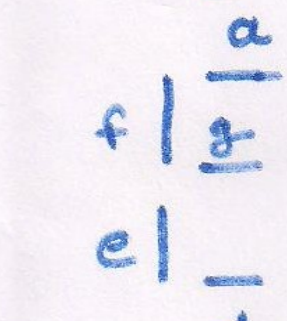
A	B	C	sum	carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Sum = Σ (1, 2, 4, 7)

carry = Σ (3, 5, 6, 7)



7 Segment (BCD to 7) common cathode



A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	0	1	1	1
1	0	1	0	x	x	x	x	x	x	x
1	0	1	1	x	x	x	x	x	x	x
1	1	0	0	x	x	x	x	x	x	x
1	1	0	1	x	x	x	x	x	x	x
1	1	1	0	x	x	x	x	x	x	x
1	1	1	1	x	x	x	x	x	x	x