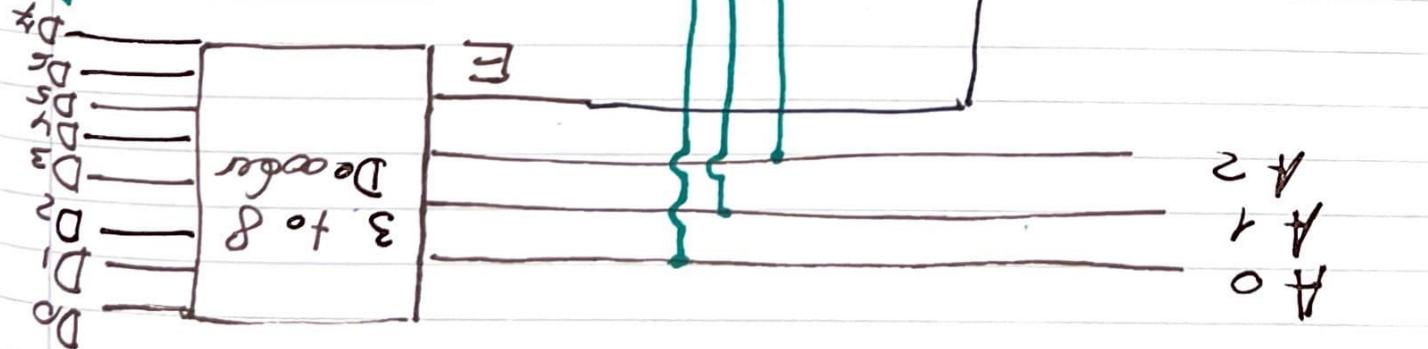
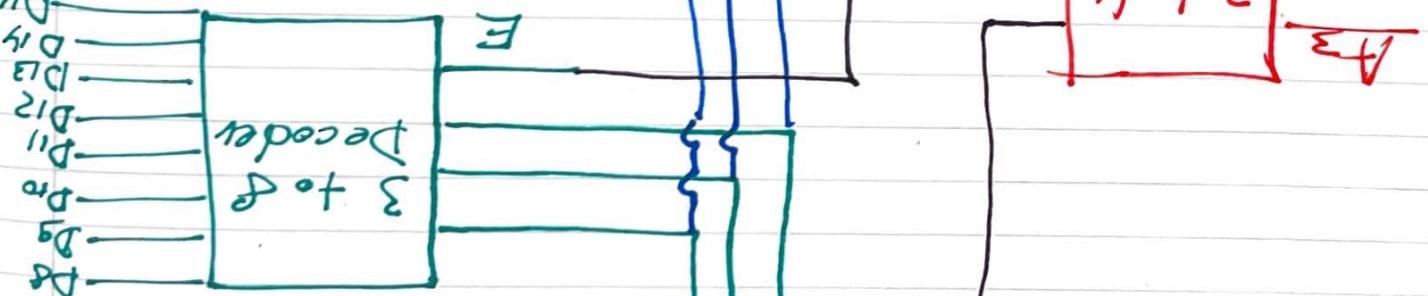
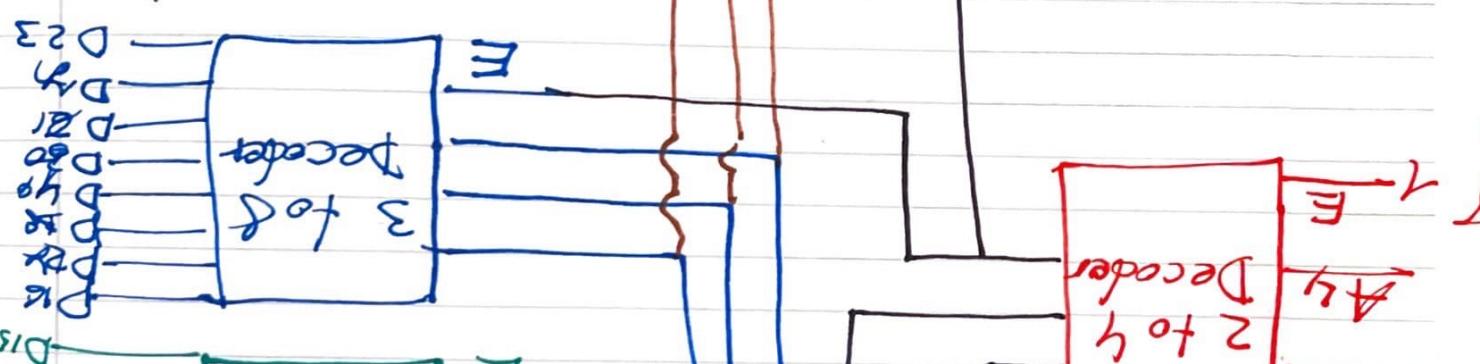
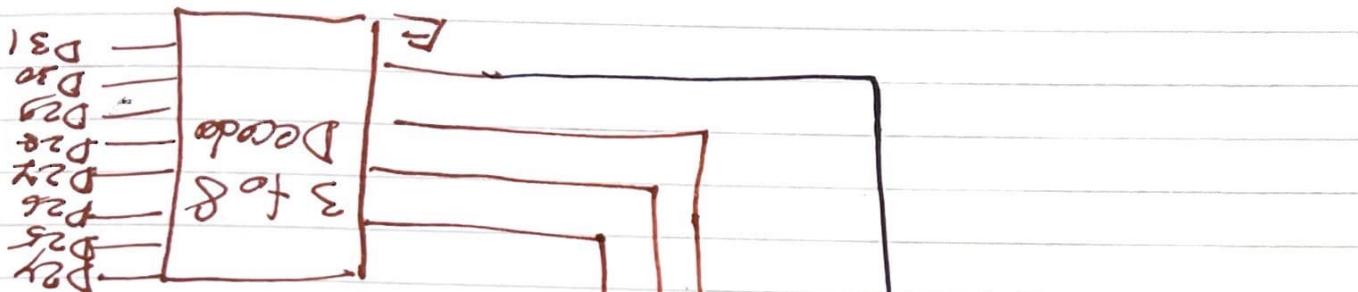


② A combinational circuit is defined. To do this  
the following three Boolean functions are  
and extended gates. Draw the equations  
Design the circuit with a decoder  
follow using three Boolean functions



④ Construct a 5-to-32 line decoder  
with four 3-to-8 line decoders  
enable all inputs and outputs  
with four 3-to-8 line decoders  
enable all inputs and outputs

Home work 5

② Based on Boolean function, design circuit with a decoder and external gates. Draw the diagram and label all inputs

$$F_1 = X'Y'Z' + XZ$$

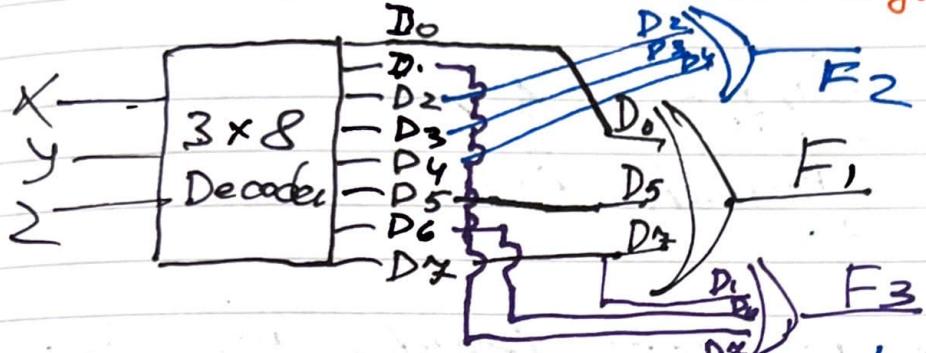
Function can be represented as decoder + OR gate.

In order to get minterms we missing one variable  $y$ , so we can represent as  $(Y + Y')$

$$F_1 = X'Y'Z' + XZ(Y + Y') = \begin{matrix} X'Y'Z' & XZY & X'ZY' \\ 0\ 0\ 0 & 1\ 1\ 1 & 1\ 0\ 1 \end{matrix}$$

minterms

connect inputs of decoder to OR gate



$$\begin{aligned} F_2 &= XY'Z' + X'Y = XY'Z' + X'Y(Z + Z') = \\ &= \begin{matrix} XY'Z' & X'YZ & X'Y'Z' \\ 1\ 0\ 0 & 0\ 1\ 1 & 0\ 1\ 0 \end{matrix} \\ &\quad D_4 \qquad D_3 \qquad D_2 \end{aligned}$$

$$F_3 = X'Y'Z + XY =$$

$$= X'Y'Z + XY(Z + Z') =$$

$$= X'Y'Z + XYZ + XY'Z' =$$

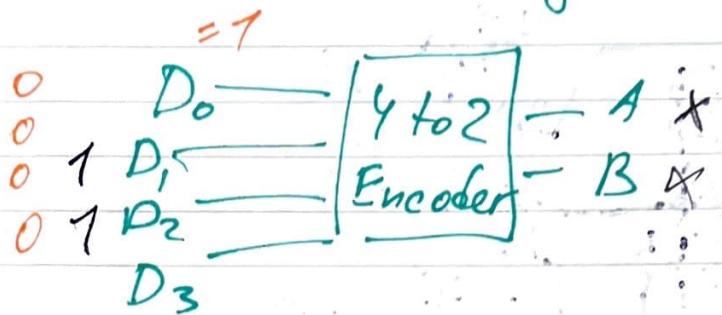
$$= \begin{matrix} X'Y'Z & XYZ & XY'Z' \\ 0\ 0\ 1 & 1\ 1\ 1 & 1\ 1\ 0 \end{matrix}$$

$$= D_1 \quad D_7 \quad D_6$$

$$D_2 \quad D_7 \quad D_6$$

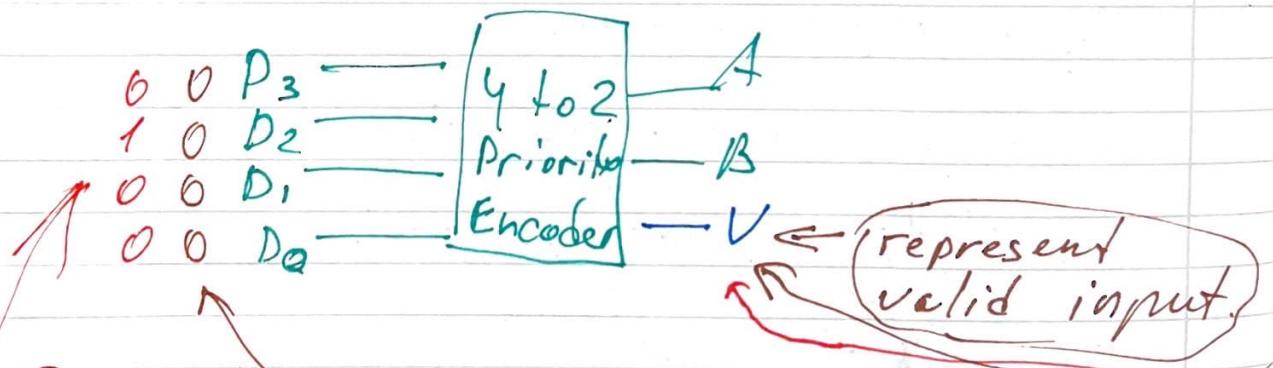
Design a 4 input priority encoder with inputs as in Table 4-8, but with inputs  $D_3$  having the lowest priority and input  $D_0$  the highest priority. Derive truth table, obtain simplified function for each output using K-map and draw logic diagram.

### Limitation of regular encoder



- ① When two or more inputs are high then result will be invalid.
- ② When all 4 inputs are 0, then output will be the same as when  $D_0 = 1$ .

These two issues can be resolved with priority encoder.



So when all inputs are 0, then  $V = 0$   
 And when  $V = 0$ , then does not matter what is the output of this  $A$  and  $B$   
 And when at least one input are high then  $V$  will be equal to 1;  $V = 1$

So when  $D_2 = 1$ , then output A and B represents valid output.

Since it's priority encoder, then each input has assigned priority and based on specified condition:

$$D_0 > D_1 > D_2 > D_3$$

So when one input is high, then like regular encoder we get output corresponding to that input.

For example if  $D_2 = 1$ , then  $A = 0, B = 1$  or when two inputs = 1 :  $D_2 = 1, D_1 = 1$ , then  $D_1$  has priority, therefore  $A = 1, B = 0$

0	0*	1*	11	2	1	V
$D_3$	$D_2$	$D_1$	$D_0$	A	B	
0	0	0	0	X	X	0
1	0	0	0	0	0	1
X	1	0	0	0	1	1
X	X	1	0	1	0	1
X	X	X	1	1	1	1

Since  $D_2$  has higher priority than  $D_3$  then it doesn't care condition X another words does not matter what a value of  $D_3$  and so on for others.

If  $D_0 = 1$ , then does not matter if  $D_1 = 0$  or  $D_1 = 1$ , same for  $D_2, D_3$ ; output still will be  $A = 1, B = 1$ .

Logic expression only valid when = 1  
So output V is high when any input is high

$$V = D_3 + D_2 + D_1 + D_0$$

$$A = D_0 + D_1 \bar{D}_0 = D_0 + \bar{D}_0 D_1 = (D_0 + \bar{D}_0)(D_0 + D_1) = D_0 + D_1$$

$$B = D_0 + D_2 \bar{D}_1 \bar{D}_0 = D_0 + D_2 \bar{D}_1$$

✓

$$V = D_1 + D_0 + D_1'D_3 + D_1'D_2 = D_0 + D_1 + D_1'(D_3 + D_2)$$

$$= D_0 + D_1 + D_3 + D_2$$

$D_3 D_2$	00	01	11	10	$D_1$	$D_0$	A	B	V
00	0	0	0	0	0	0	X	X	0
01	0	0	0	0	1	1	1	1	1
11	0	0	1	0	0	0	1	0	1
10	0	0	1	1	1	1	1	1	1
00	0	1	0	0	0	1	0	1	1
01	0	1	0	0	1	0	1	0	1
11	0	1	0	1	1	1	1	1	1
10	0	1	0	1	1	0	0	0	1
00	1	0	0	0	0	0	0	0	0
01	1	0	0	0	0	1	1	1	1
11	1	0	0	1	0	1	1	0	1
10	1	0	0	1	1	0	0	1	1
00	1	1	0	0	0	0	0	1	0
01	1	1	0	0	0	1	0	1	1
11	1	1	0	1	0	1	1	0	1
10	1	1	0	1	1	0	0	1	1

$A \quad P_1 P_0$

$D_3 D_2$	00	01	11	10	$P_1$	$P_0$
00	X	1	1	1	1	0
01	0	1	1	1	0	1
11	0	1	1	1	1	1
10	0	1	1	1	1	1

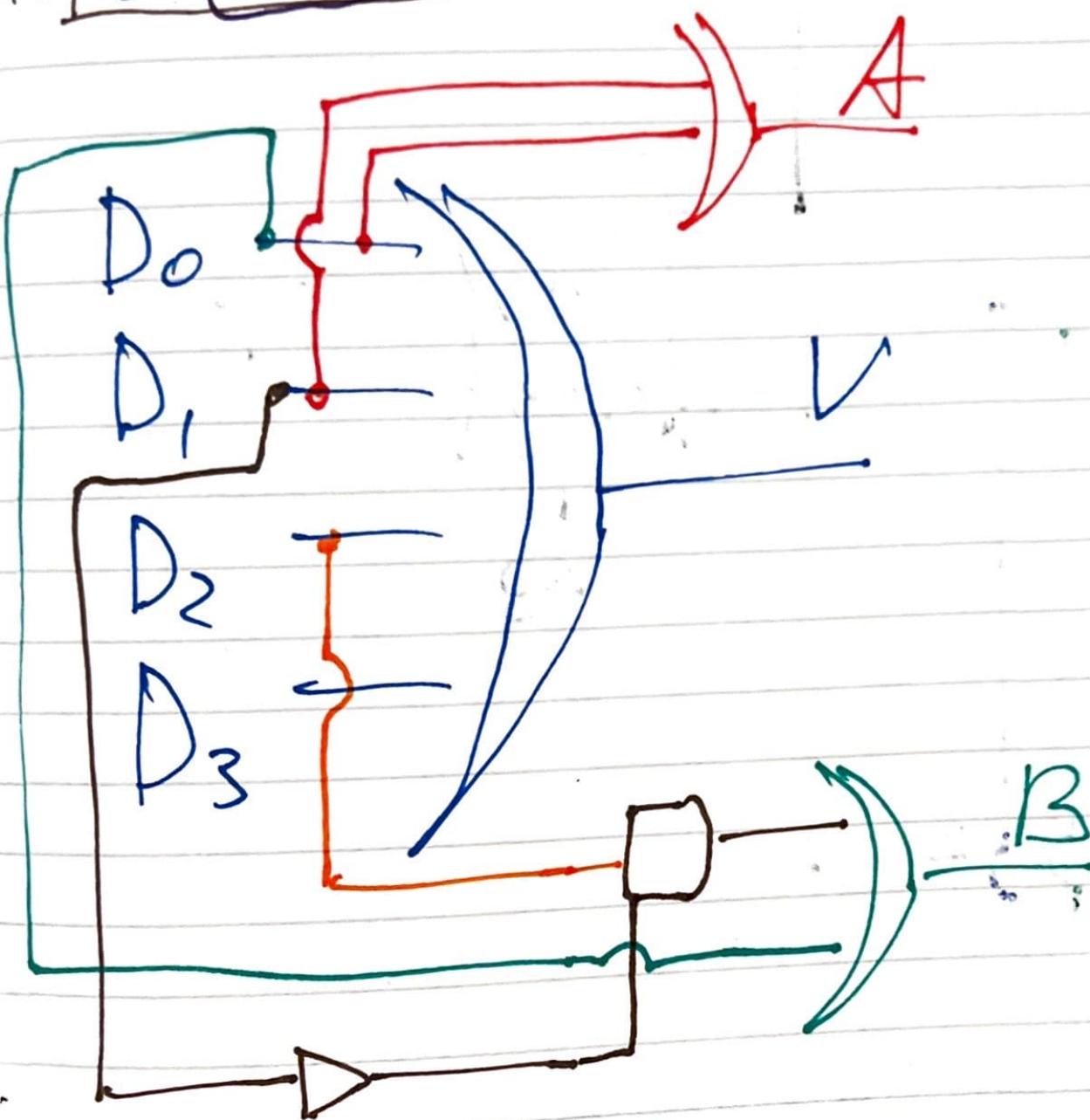
$$A = D_0 + D_1$$

$D_1, D_0$

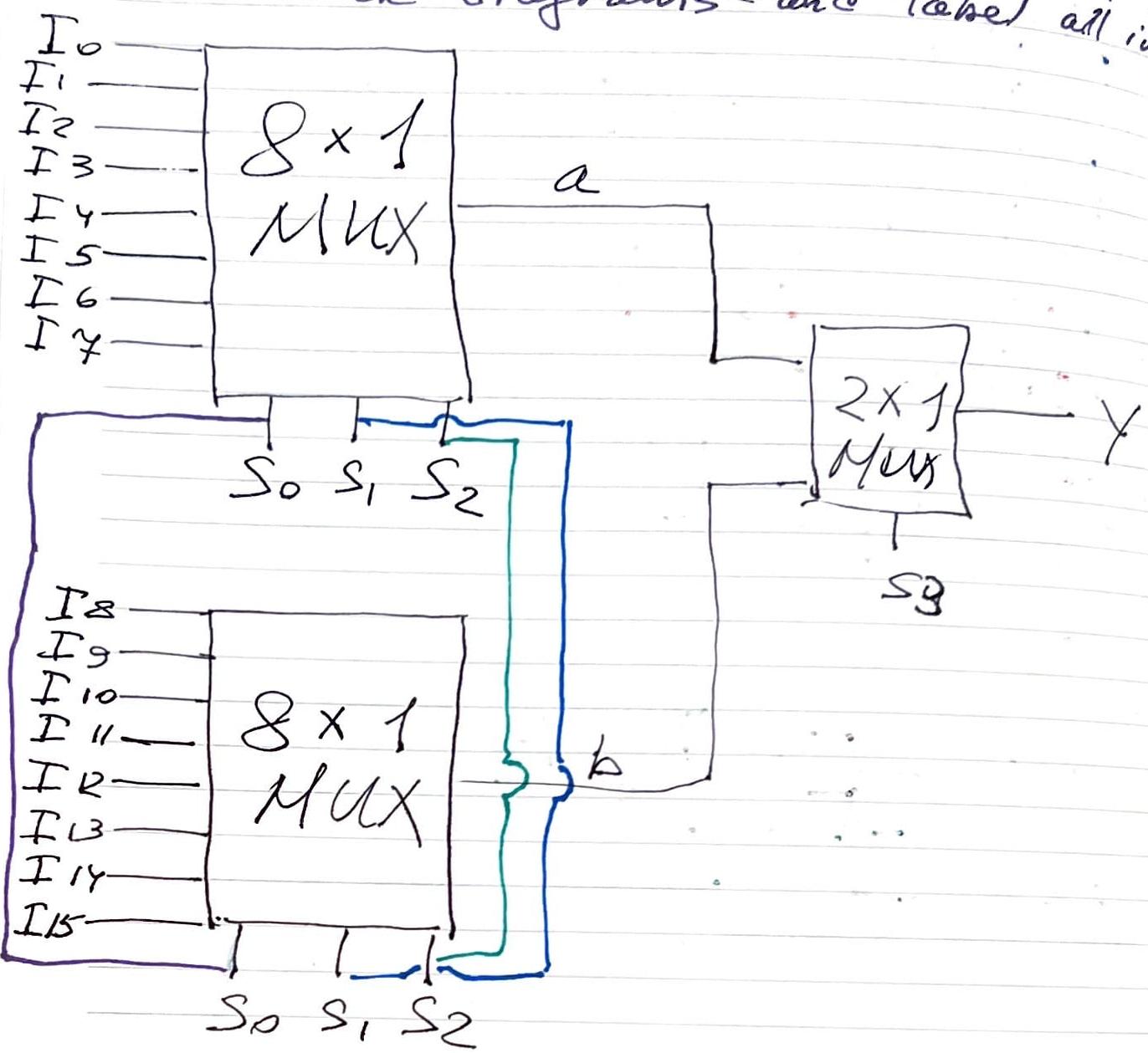
$B$

	00	01	11	10
00	X	1	1	0
01	1	1	1	0
11	10	1	1	0
10	0	1	1	0

$$B = D_0 + D_1' D_2$$



Q. Construct a  $16 \times 1$  multiplexer with two  $8 \times 1$  and one  $2 \times 1$  MUX. Use block diagrams and label all inputs.

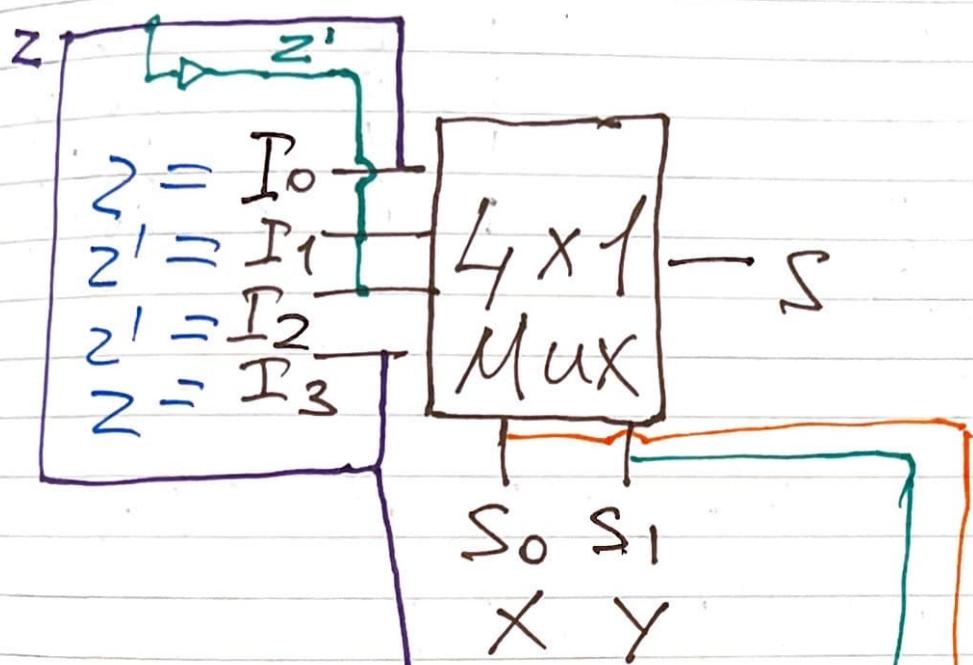


⑤ Implement a full adder with two  $4 \times 1$  multiplexers. Draw the truth table, diagrams and label inputs.

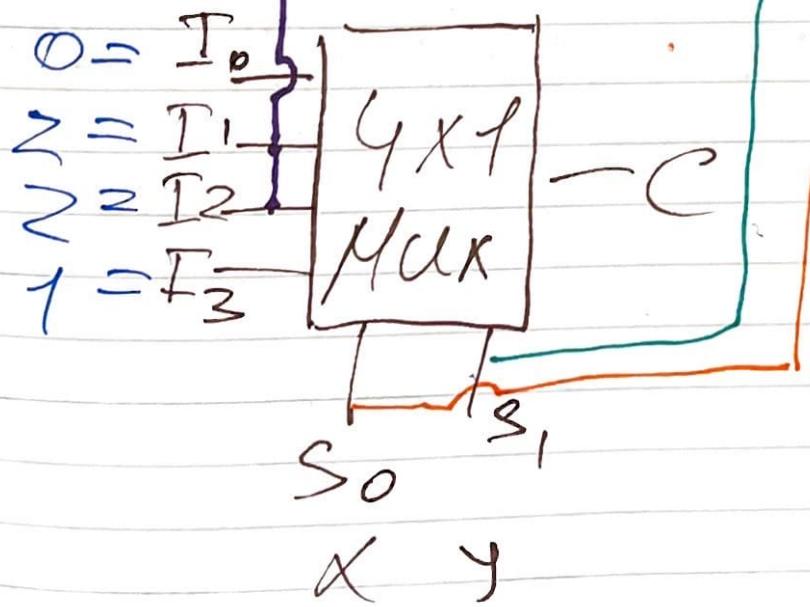
Full adder Truth Table is based on regular Truth table.

To get S we do summation  
C is carry from summation

	X	Y	Z	S	C		
$I_0$	0	0	0	0	$S=Z$	0	$S=0$
$I_1$	0	1	0	1	$S=Z'$	0	$S=1$
$I_2$	1	0	0	1	$S=Z'$	0	$C=Z$
$I_3$	1	1	0	0	$S=Z$	1	$C=1$



$S_0$	$S_1$	S
0	0	$I_0$
0	1	$I_1$
1	0	$I_2$
1	1	$I_3$



6.

Inputs

Output

	A	B	C	D	F	
$I_0 = D'$	0	0	0	0	1	$m_0$
$I_1 = D'$	0	0	0	1	0	
$I_2 = 0$	0	0	1	0	0	
$I_3 = 0$	0	1	0	0	0	
$I_4 = D$	0	0	1	1	1	$m_6$
$I_5 = D$	1	0	0	0	0	$m_4$
$I_6 = D$	1	0	1	0	1	$m_1$
$I_7 = D$	1	0	1	1	0	$m_{11}$
$I_8 = D$	1	1	0	0	1	$m_{12}$
$I_9 = D$	1	1	0	1	0	$m_{14}$
$I_{10} = D$	1	1	1	0	1	$m_{15}$

	CD		
	A	B	
	00	01	11
00	1	0	0
01	0	0	1
11	0	1	0
10	1	0	1

$$F = C'D'B' + ABD + A'BC + AB'C$$