



# Doubt Clearing Session

Comprehensive Course on Digital Logic Design 2023/2024

# Combinational Logic Circuit

# Combinational Logic Circuit

- The present output depends on present input only
- In combinational circuits feedback and clock is not present

- HA
- HS
- FA
- FS
- Parallel Adder
- Carry look ahead Adder
- Binary Multiplier
- Magnitude Comparators
- Multiplexer
- Demultiplexers
- Decoder
- Encoder
- Priority Encoder
- Code converters

# Half Adder

For the addition of two single bits



A	B	Sum	Carry

# Logic Circuit

# Half Adder using NAND Gates

# Half Adder using NOR Gates

# Half Subtractor (A-B )

For the subtraction of two single bits



A	B	Difference	Borrow

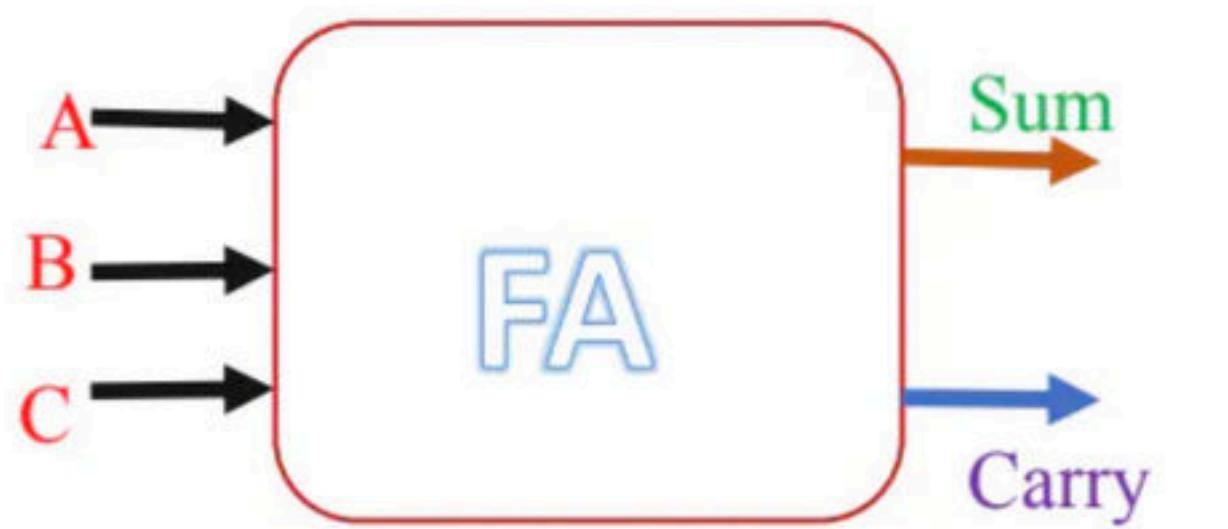
# Logic Circuit

# Half Adder / Half Subtractor

# Half Subtractor using NAND Gates

# Half Subtractor using NOR Gates

# Full Adder



# Full Adder with two Half Adders

# Full Adder with two Half Adders

# Full Adder using NAND Gates

# Full Adder using NOR Gates

## Full Subtractor(A - B-C )



# Logic Circuit

# Full Subtractor with two Half Subtractors

# Full Subtractor with two Half Subtractors

# Full Subtractor using NAND Gates

# Full Subtractor using NOR Gates

FS : A- B- C

FS : B- C- A

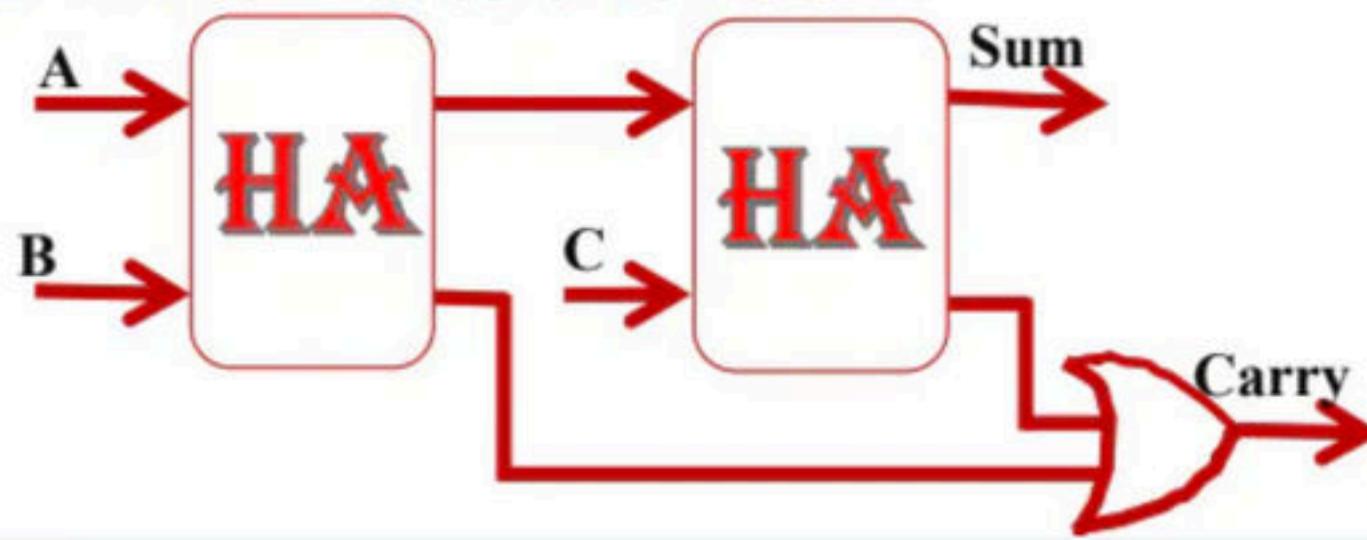
FS : C- A- B

## HA

1. Logical expression for Sum =
2. Logical expression for Carry =
3. Minimum number of NAND Gates =
4. Minimum number of NOR Gates =

## FA

1. Logical expression for Sum =
2. Logical expression for Carry =
3. Minimum number of NAND Gates =
4. Minimum number of NOR Gates =

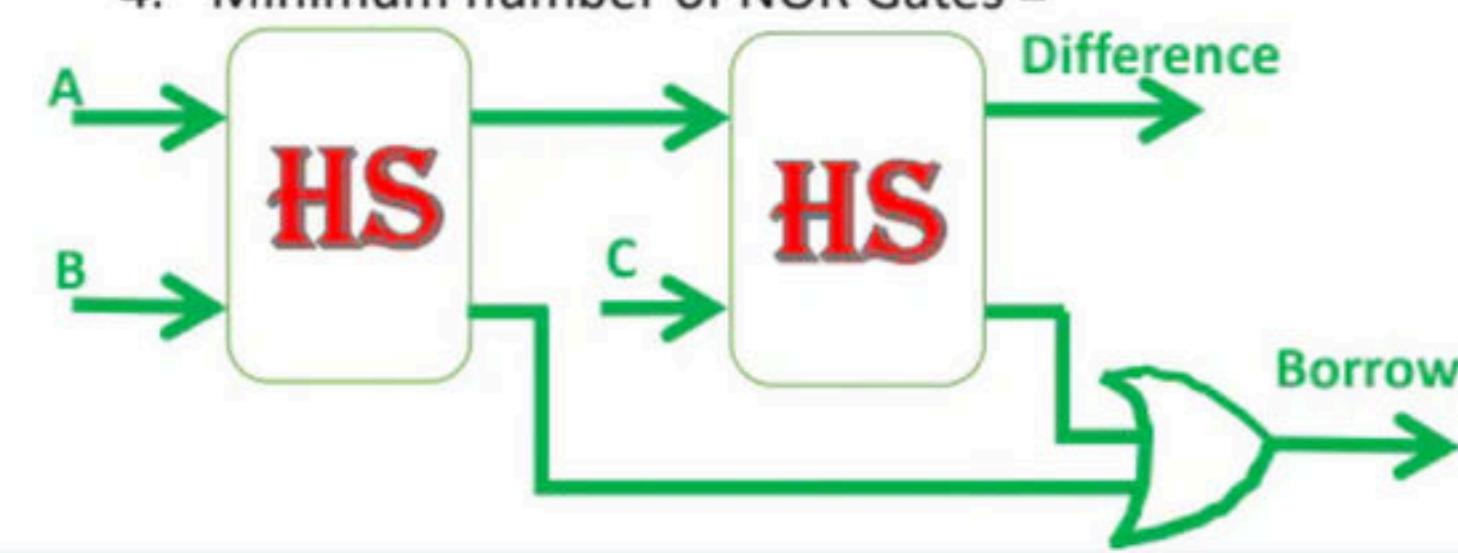


## HS

1. Logical expression for Difference =
2. Logical expression for Borrow=
3. Minimum number of NAND Gates =
4. Minimum number of NOR Gates =

## FS

1. Logical expression for Difference=
2. Logical expression for Borrow =
3. Minimum number of NAND Gates =
4. Minimum number of NOR Gates =

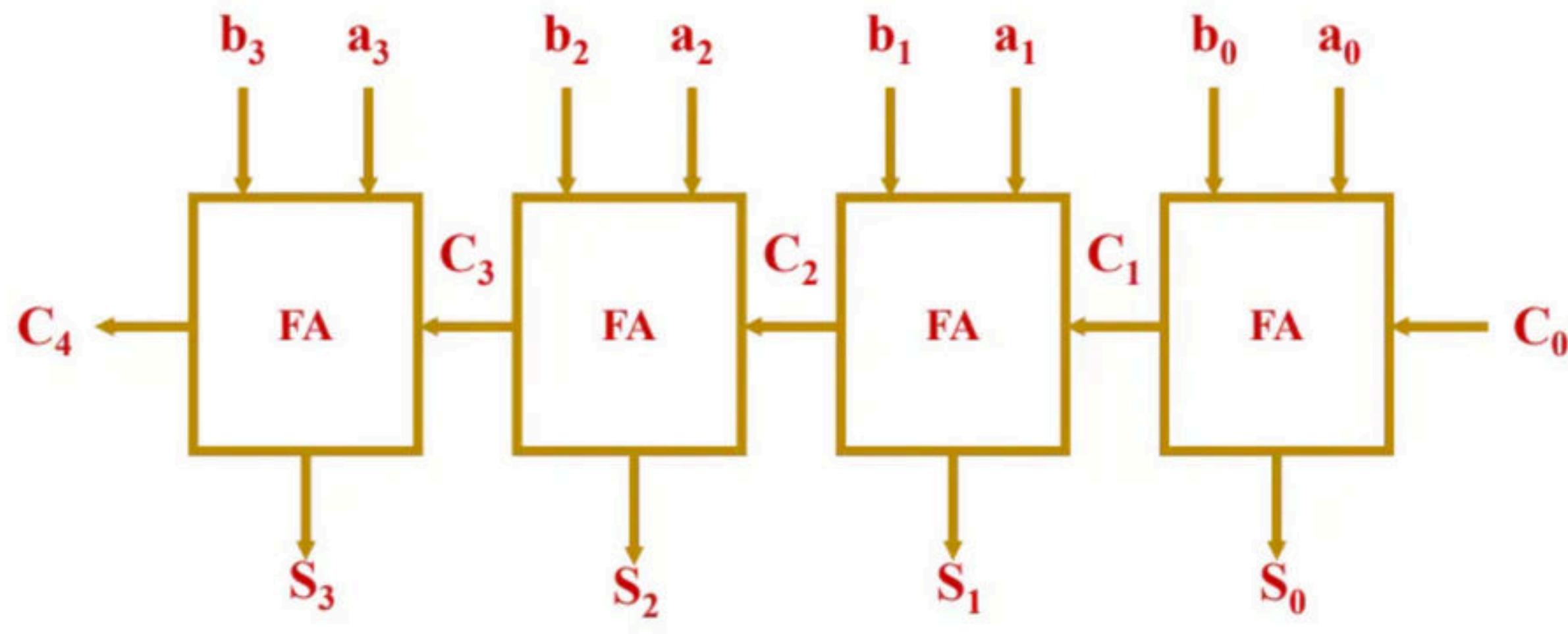


# Ripple Carry Adder (Parallel Adder)

A  $\longrightarrow a_3 \quad a_2 \quad a_1 \quad a_0$

B  $\longrightarrow b_3 \quad b_2 \quad b_1 \quad b_0$





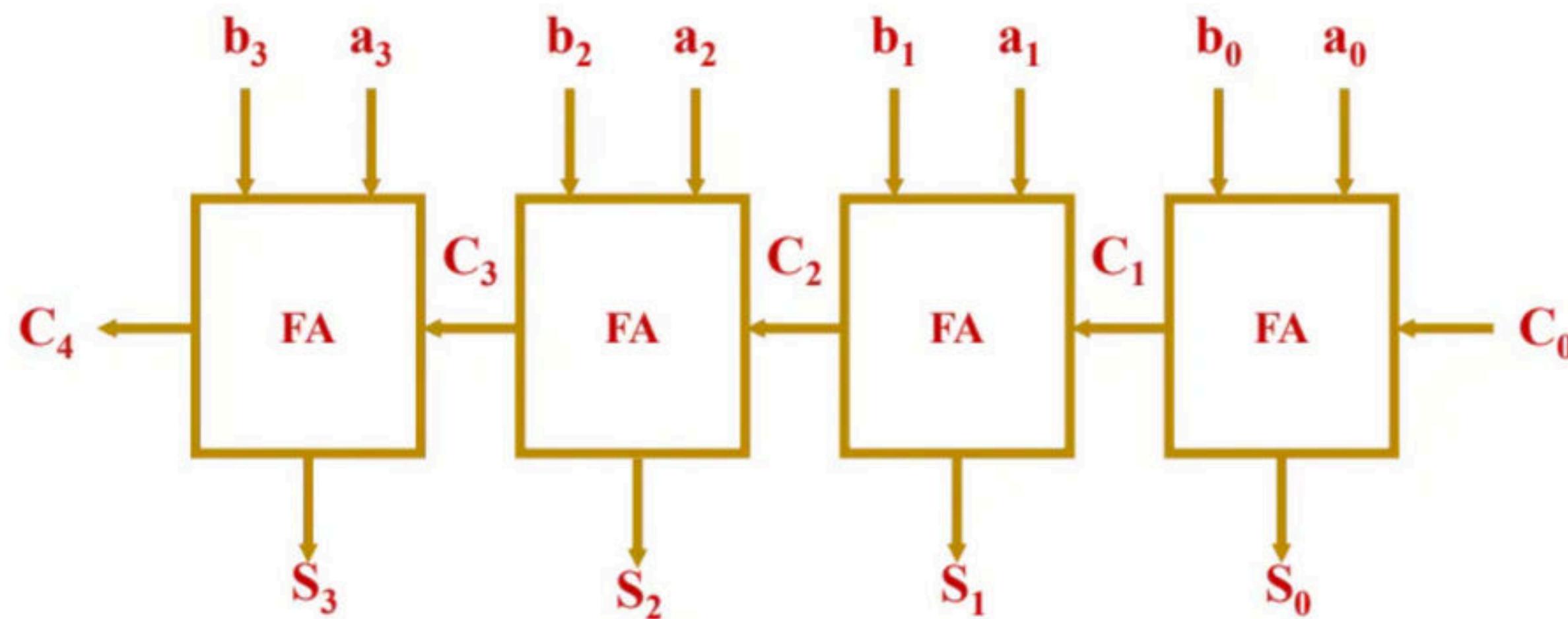
Note :

To implement 4-bit parallel adder

To implement n-bit parallel adder

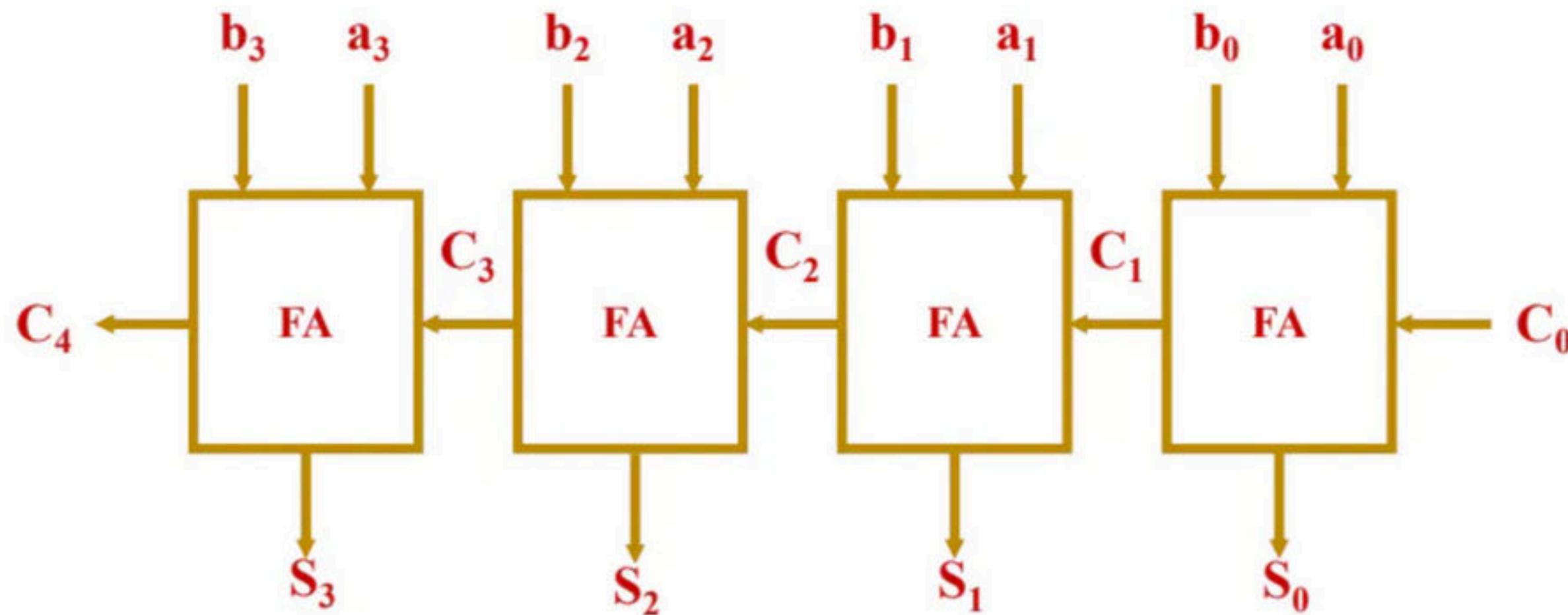
# Delay Analysis

Case : 1  $(tpd)_{sum} > (tpd)_{carry}$



# Delay Analysis

Case : 2  $(tpd)_{sum} < (tpd)_{carry}$



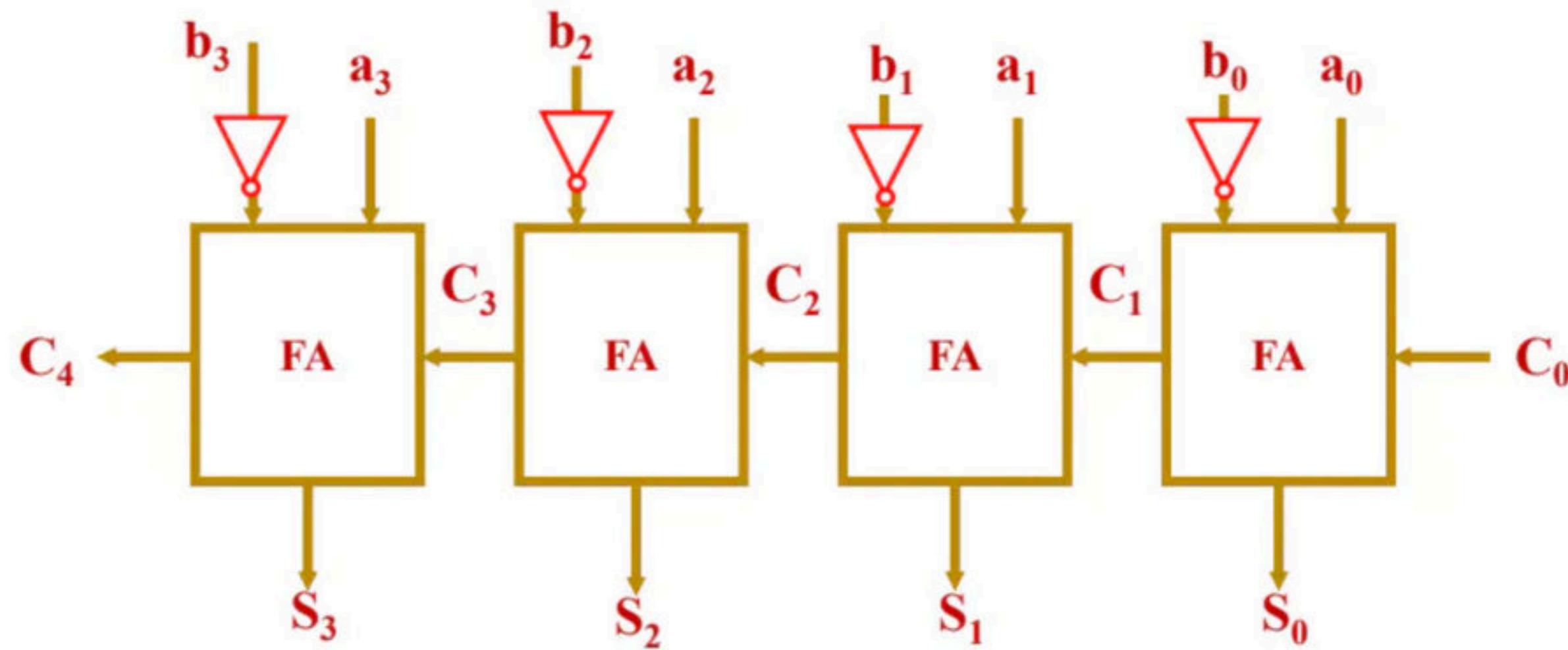
In general for n- bit Parallel Adder

Delay =

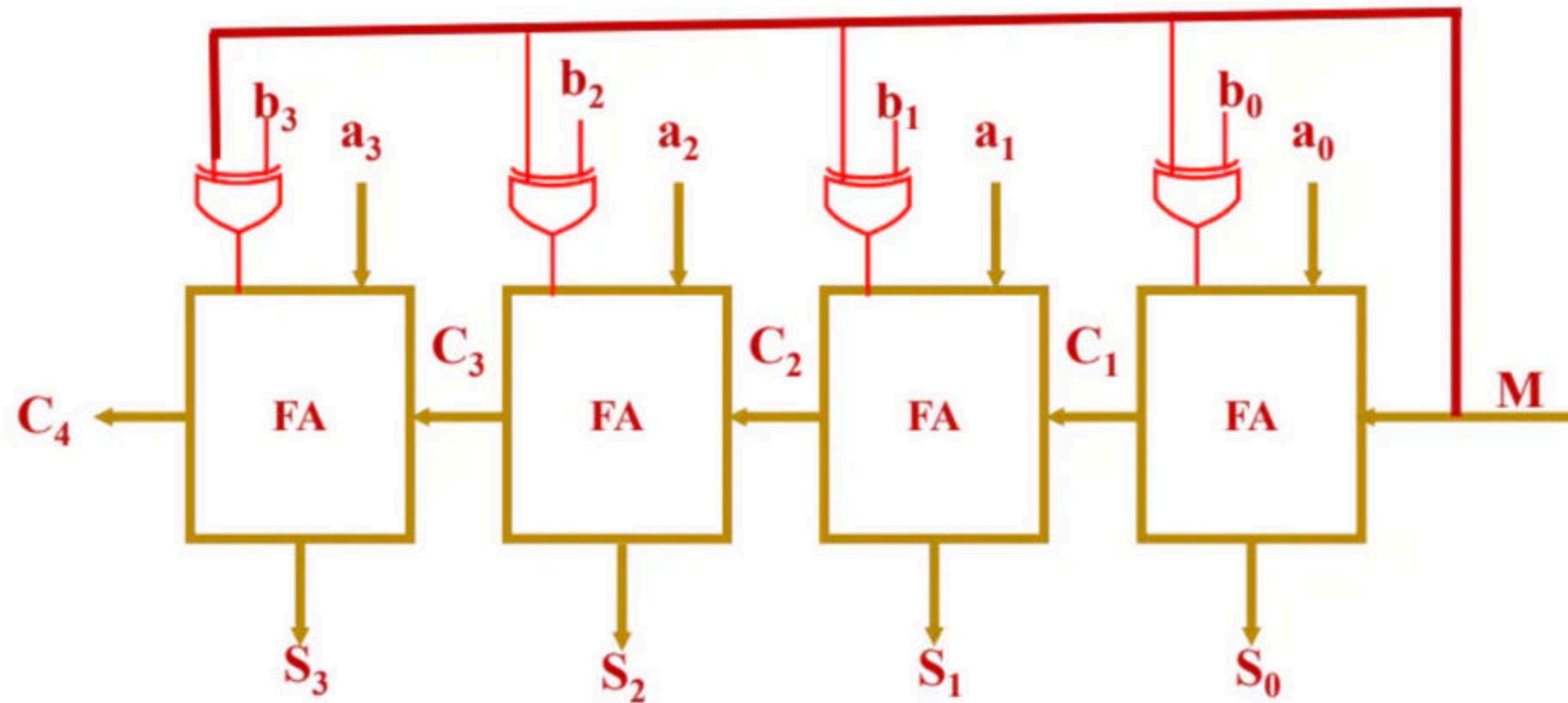
Q) A 16-bit RCA is realized using 16 identical FAs , if the ( tpd)carry = 12ns ,  
(tpd)sum= 15ns , then the overall delay is ----- ns

Q) A 4-bit RCA is implemented using 4-FAs , if the propagation delay of XOR – Gate is twice the delay of AND/OR Gate , then the overall delay of 4- bit RCA if the delay of AND/OR Gate is  $1.2 \mu\text{sec}$

# Parallel Subtractor

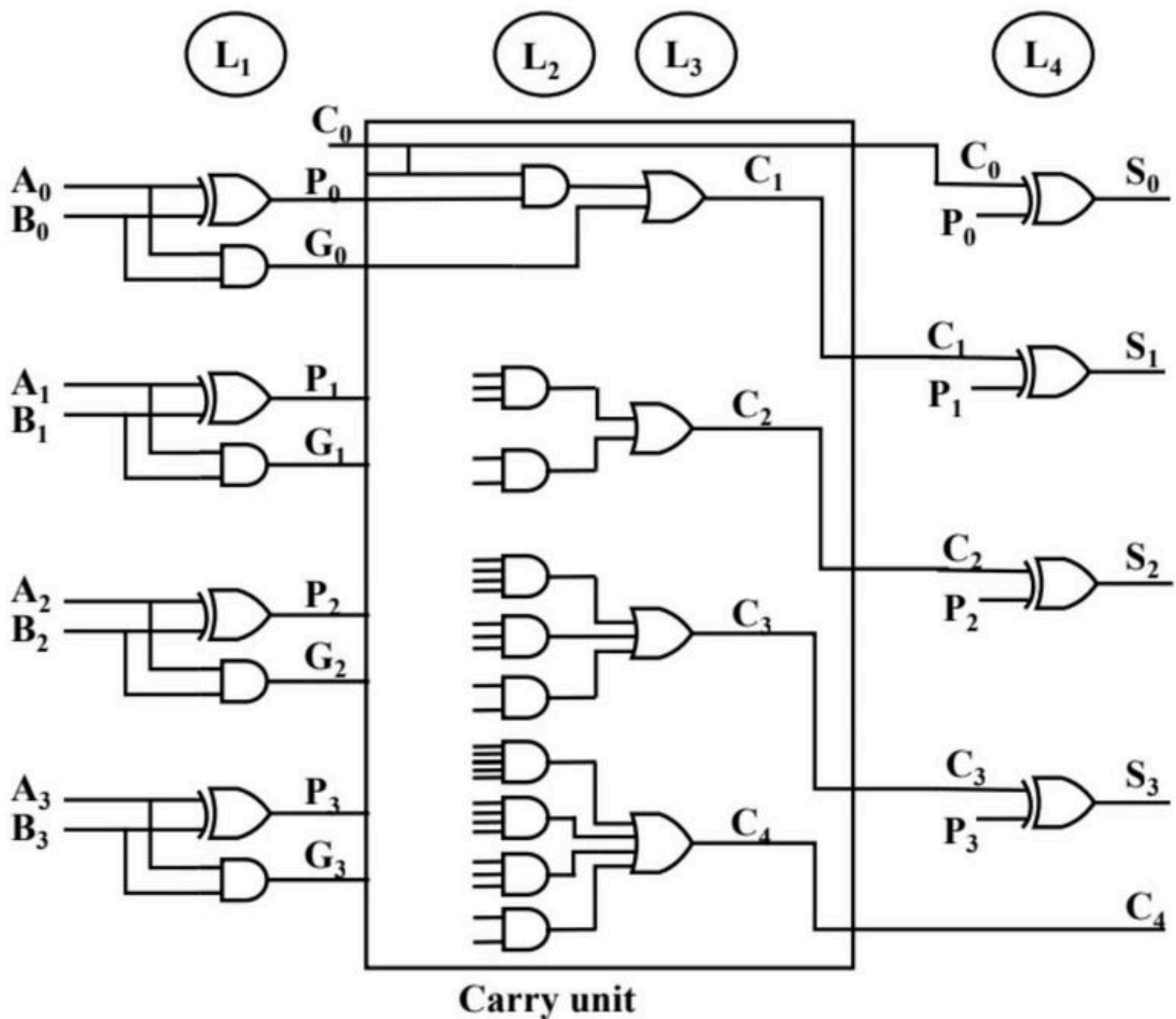


# Parallel Adder/ Subtractor



# **Look Ahead Carry Adder**

- In this adder ,the carry dependency of Ripple Carry Adder (RCA) is eliminated
- This is the fastest adder among all
- This adder have the maximum complexity



## Hardware Requirements

L 1 :

L 2 :

L 3 :

L 4 :

Total number of gates for carry =

Total number of gates for sum =

# Delay Analysis

Carry =

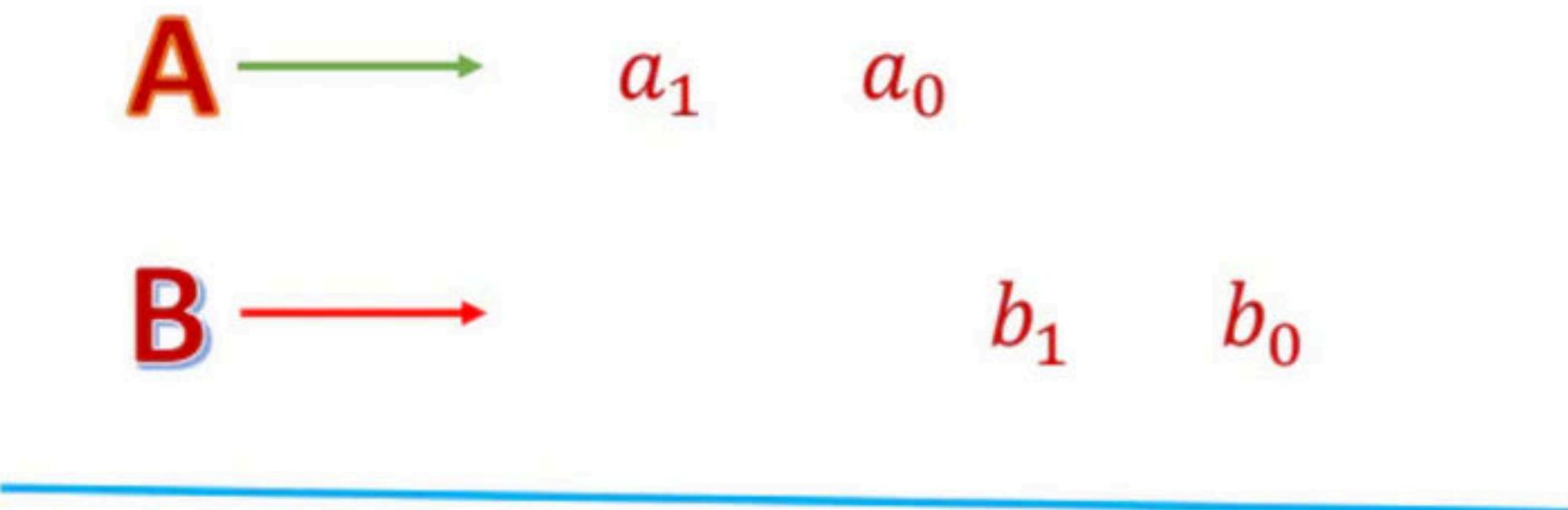
Sum =

Q) The minimum number of gates required for the implementation 4- bit look ahead carry adder are -----

Q) During the implementation of carry look ahead adder , if carry generator (  $G_i$  ) and carry propagator (  $P_i$  ) are available, then the minimum number of gates required are

Q) In 4- bit look ahead carry adder is implemented with the following gates NOT , AND, OR , NAND , NOR calculate the minimum time required to generate sum if each gate has 1 unit

# Binary Multiplier



**A** 

$a_2$   $a_1$   $a_0$

**B** 

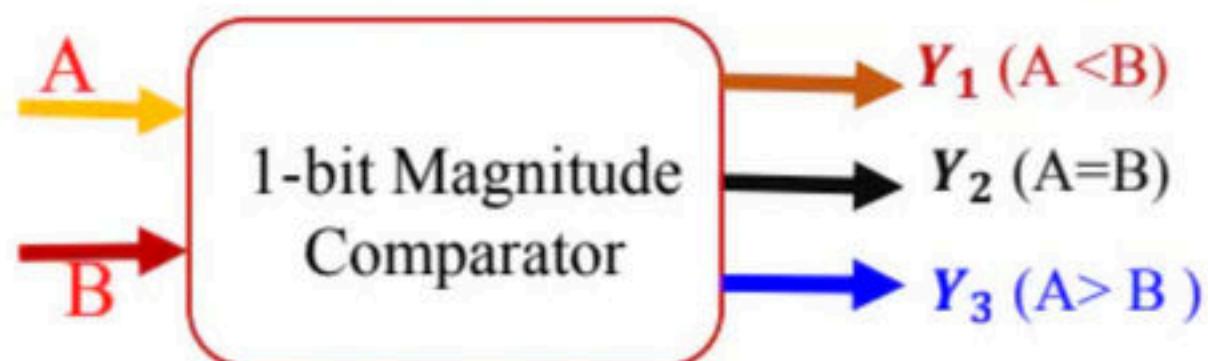
$b_1$   $b_0$



# Magnitude Comparator

To compare the magnitude of two binary numbers .

# 1-bit magnitude comparator



A	B	$Y_1$ (A < B)	$Y_2$ (A = B)	$Y_3$ (A > B)

$Y_1$  (A < B) =

$Y_2$  (A = B) =

$Y_3$  (A > B) =

# Logic circuit

# 1-bit Magnitude Comparator

Total number of input combinations =

Lesser than combinations =

Greater than combinations =

Equal combinations =

## 2-bit Magnitude Comparator



## For 2-bit Magnitude Comparator

Total number of input combinations =

Lesser than combinations =

Greater than combinations =

Equal combinations =

# For 2-bit Magnitude Comparator

$Y_1 (A < B) =$

$Y_2 (A = B) =$

$Y_3 (A > B) =$

For 3- bit Magnitude Comparator

$Y_1 (A < B) =$

$Y_2 (A = B) =$

$Y_3 (A > B) =$

# For 4-bit Magnitude Comparator

$Y_1 (A < B) =$

$Y_2 (A = B) =$

$Y_3 (A > B) =$

## For n-bit Magnitude Comparator

Total number of input combinations =

Lesser than combinations =

Greater than combinations =

Equal combinations =

Q ) Find the number of 1- bit comparators , AND gates and OR gates required to implement 2- bit Comparator

Q ) Find the number of 1- bit comparators , AND gates and OR gates required to implement 4- bit Comparator

# Multiplexer (MUX)

- Data selector
- Many to one
- Universal logic gate
- Parallel to serial converter

## The general structure of a Mux

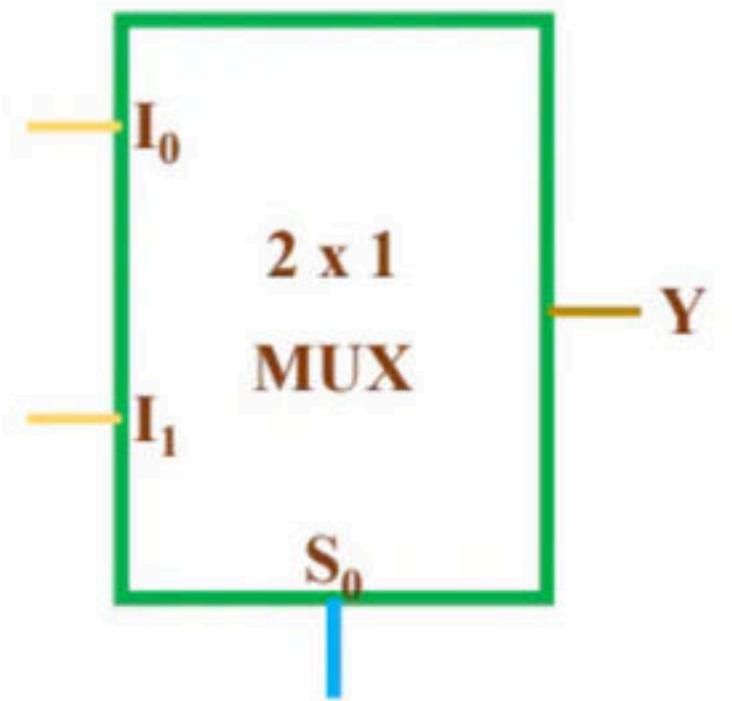
$2^n \times 1$

$2^n$  -----> number of data inputs

n -----> number of select inputs

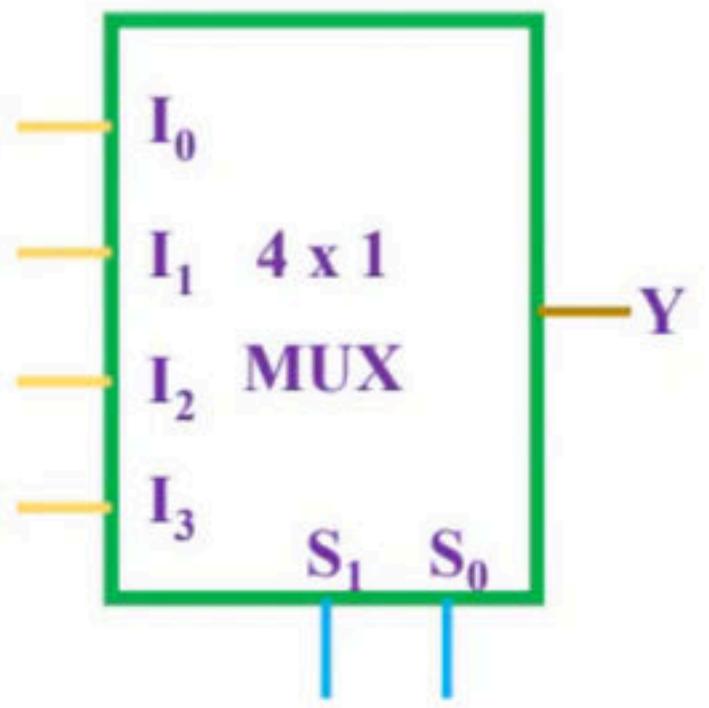
1 -----> number of outputs

# **$2 \times 1$ MUX**



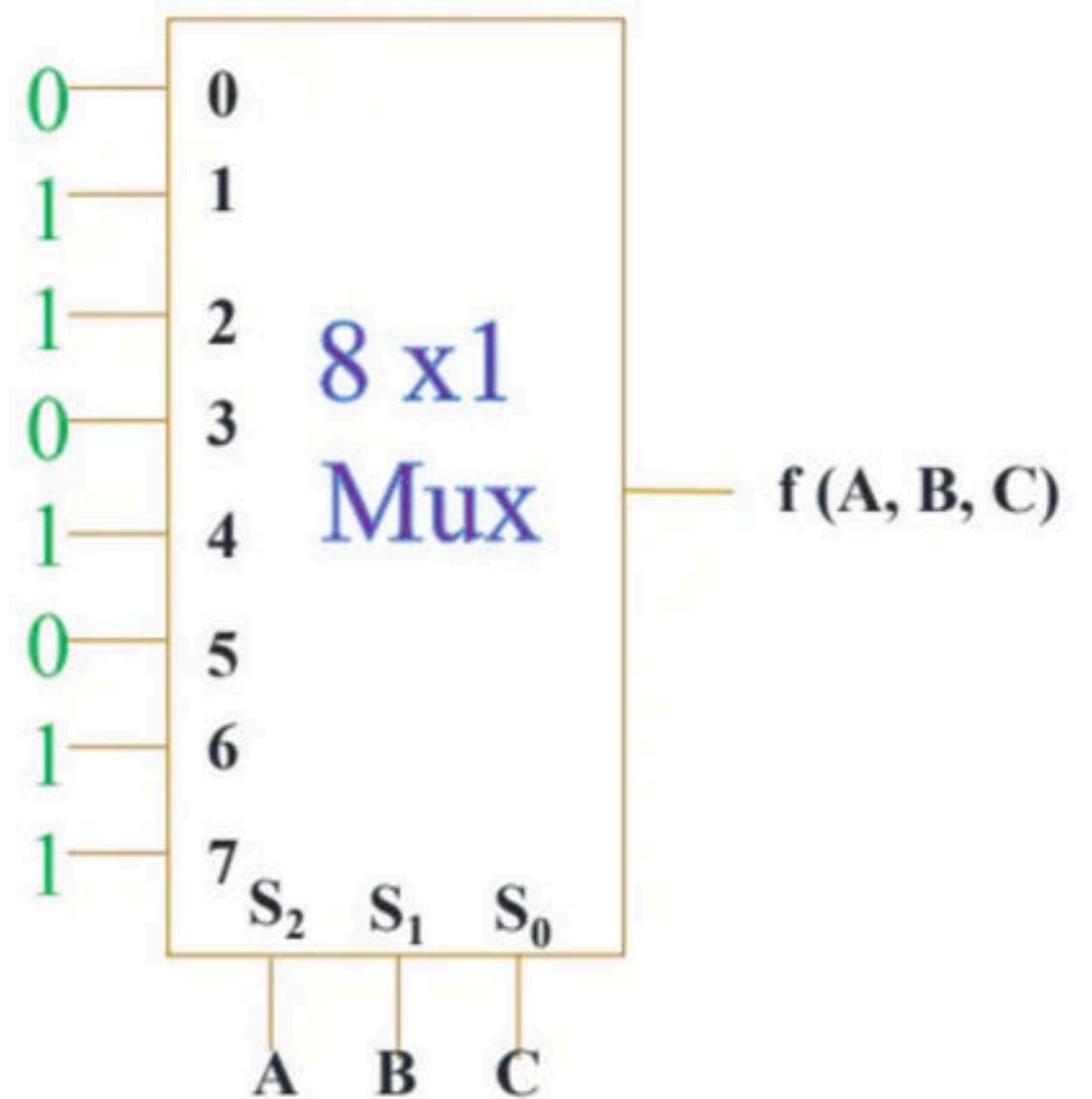
# Logic circuit

# **$4 \times 1$ MUX**



$S_1$	$S_0$	$Y$

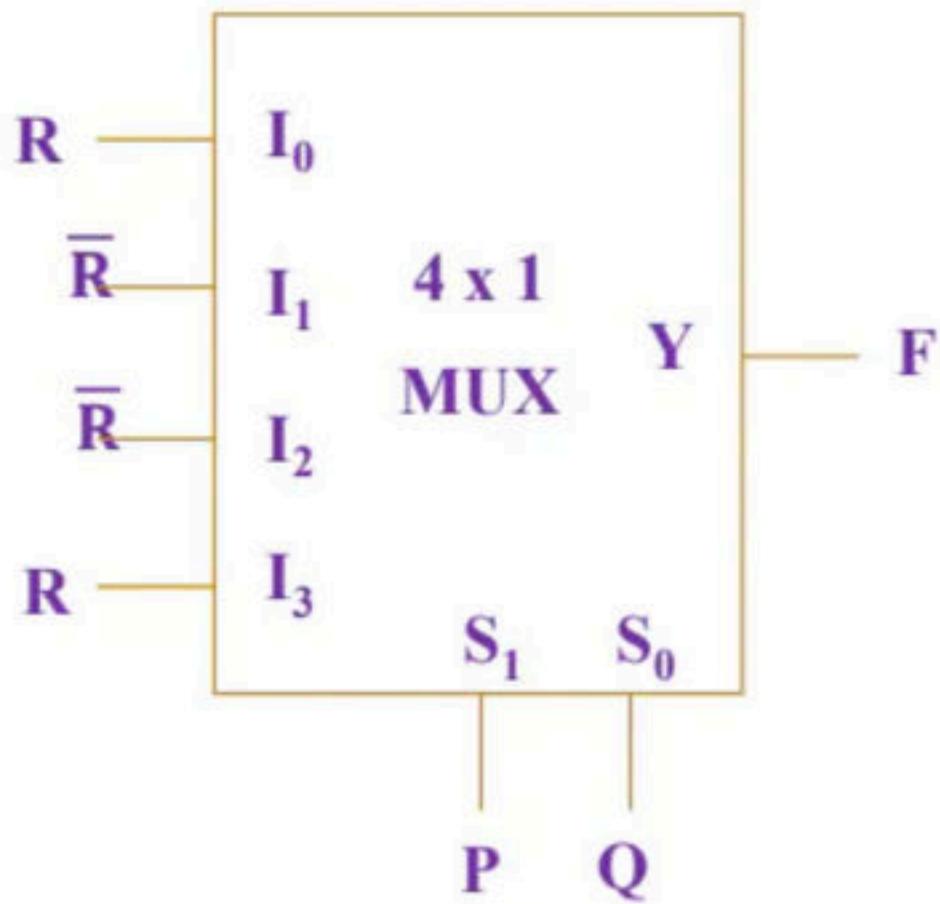
Q) Find the minterms



**Q.** The output F of the multiplexer circuit shown below expressed in terms of the inputs P, Q and R is

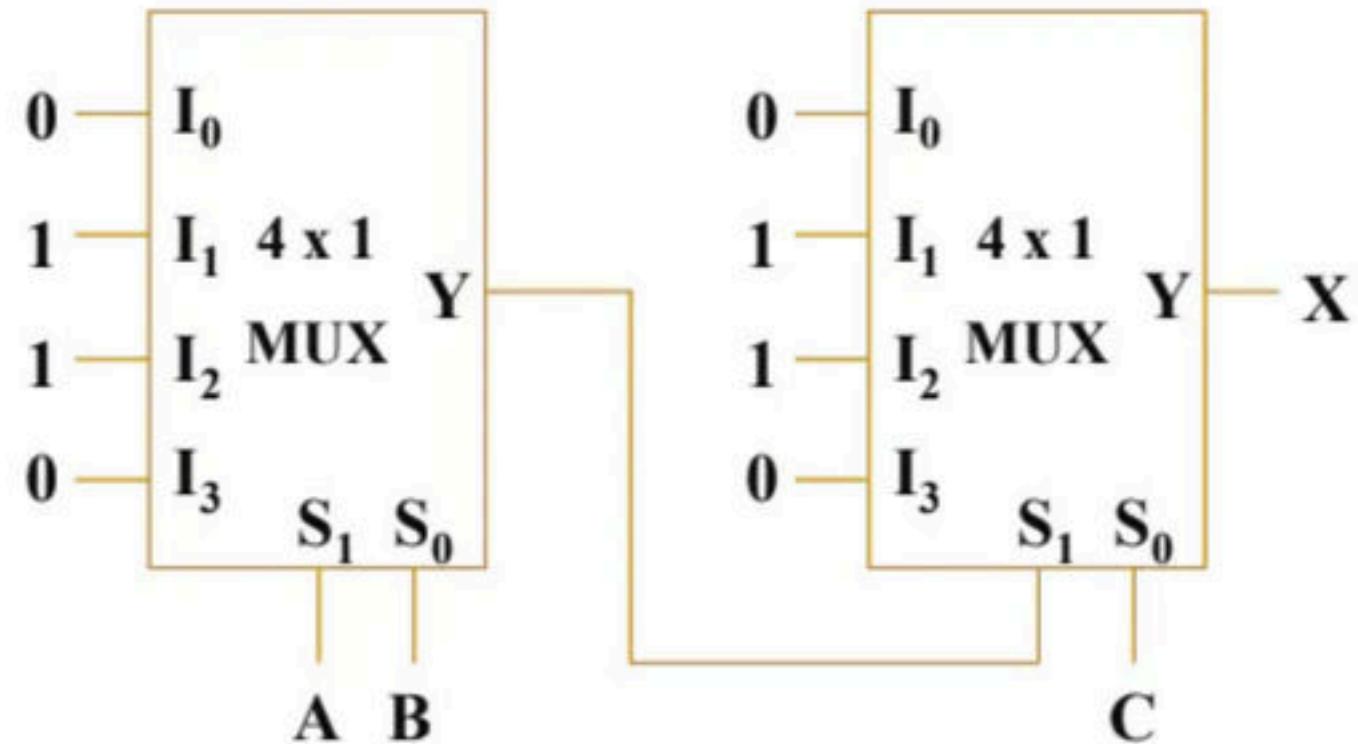
- (a)  $F = P \oplus Q \oplus R$
- (c)  $F = (P \oplus Q)R$

- (b)  $F = PQ + QR + RP$
- (d)  $F = (P \oplus Q)\bar{R}$



**Q.** in the following circuit, X is given by

- (a)  $X = A\bar{B}\bar{C} + \bar{A}B\bar{C} + \bar{A}\bar{B}C + ABC$
- (b)  $X = \bar{A}BC + A\bar{B}C + AB\bar{C} + \bar{A}\bar{B}\bar{C}$
- (c)  $X = AB + BC + AC$
- (d)  $X = \bar{A}\bar{B} + \bar{B}\bar{C} + \bar{A}\bar{C}$



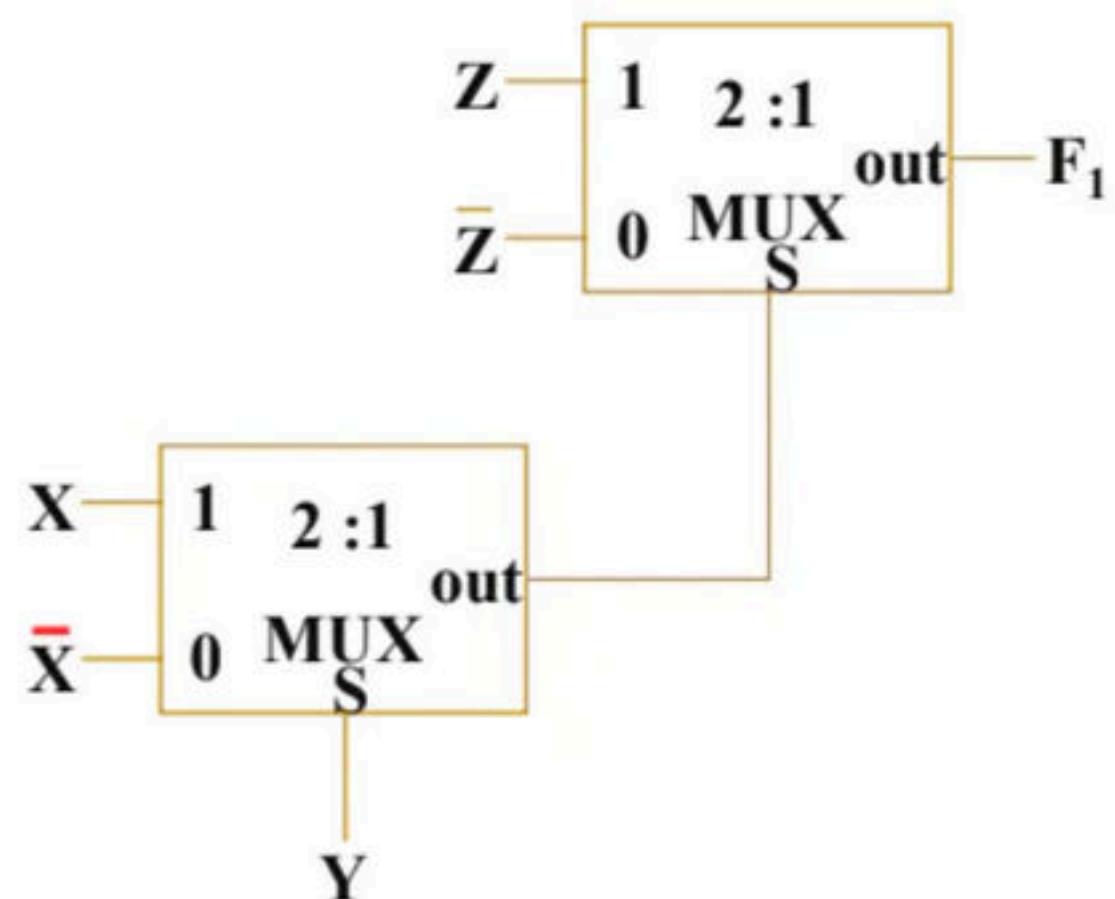
**Q.** A MUX circuit shown in the figure below implements a logic function  $F_1$ . The correct expression for  $F_1$  is.

(a)  $(\overline{X} \oplus \overline{Y}) \oplus Z$

(c)  $(X \oplus Y) \oplus \bar{Z}$

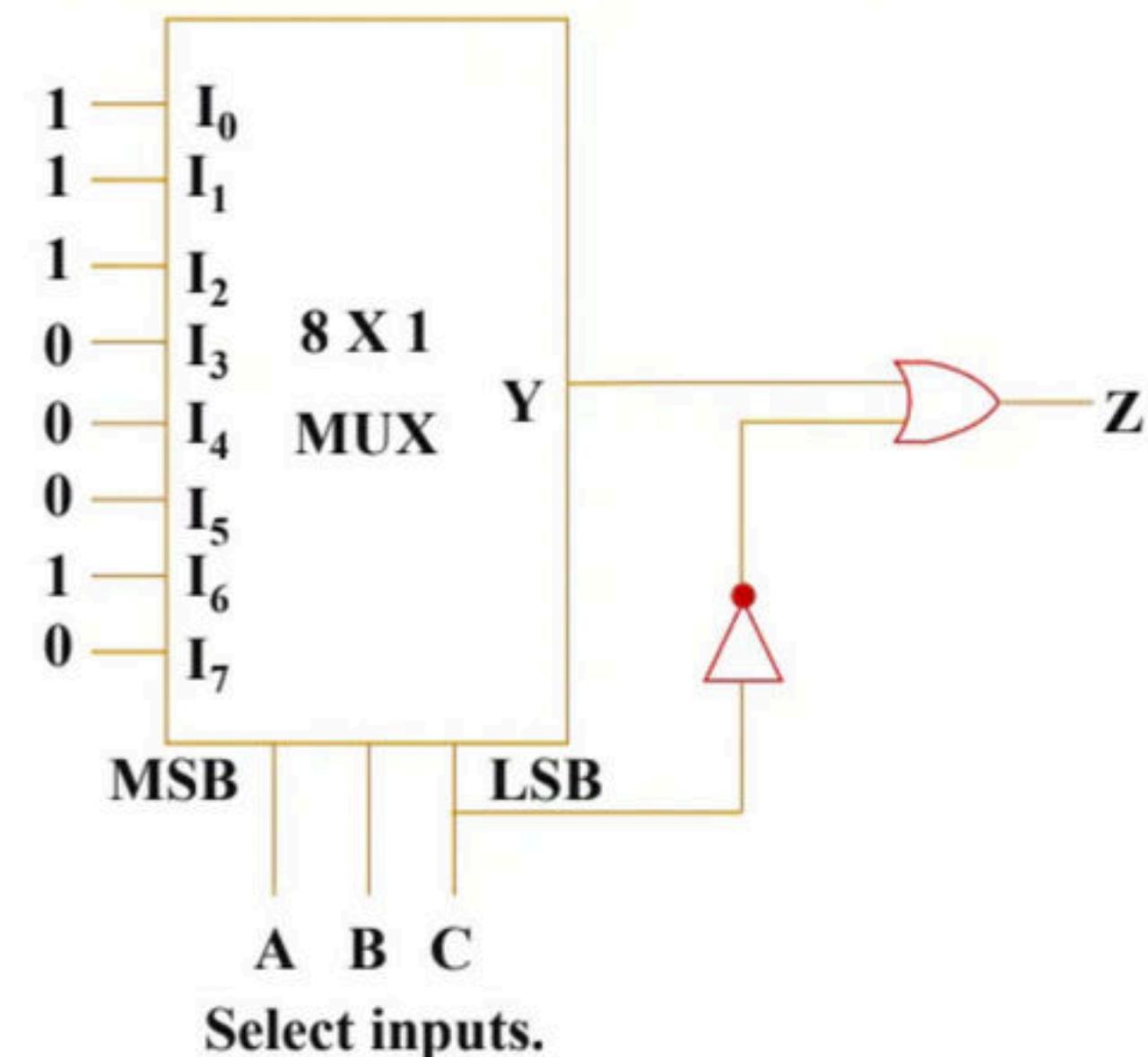
(b)  $\overline{(\overline{X} \oplus Y) \oplus Z}$

(d)  $(X \oplus Y) \oplus Z$



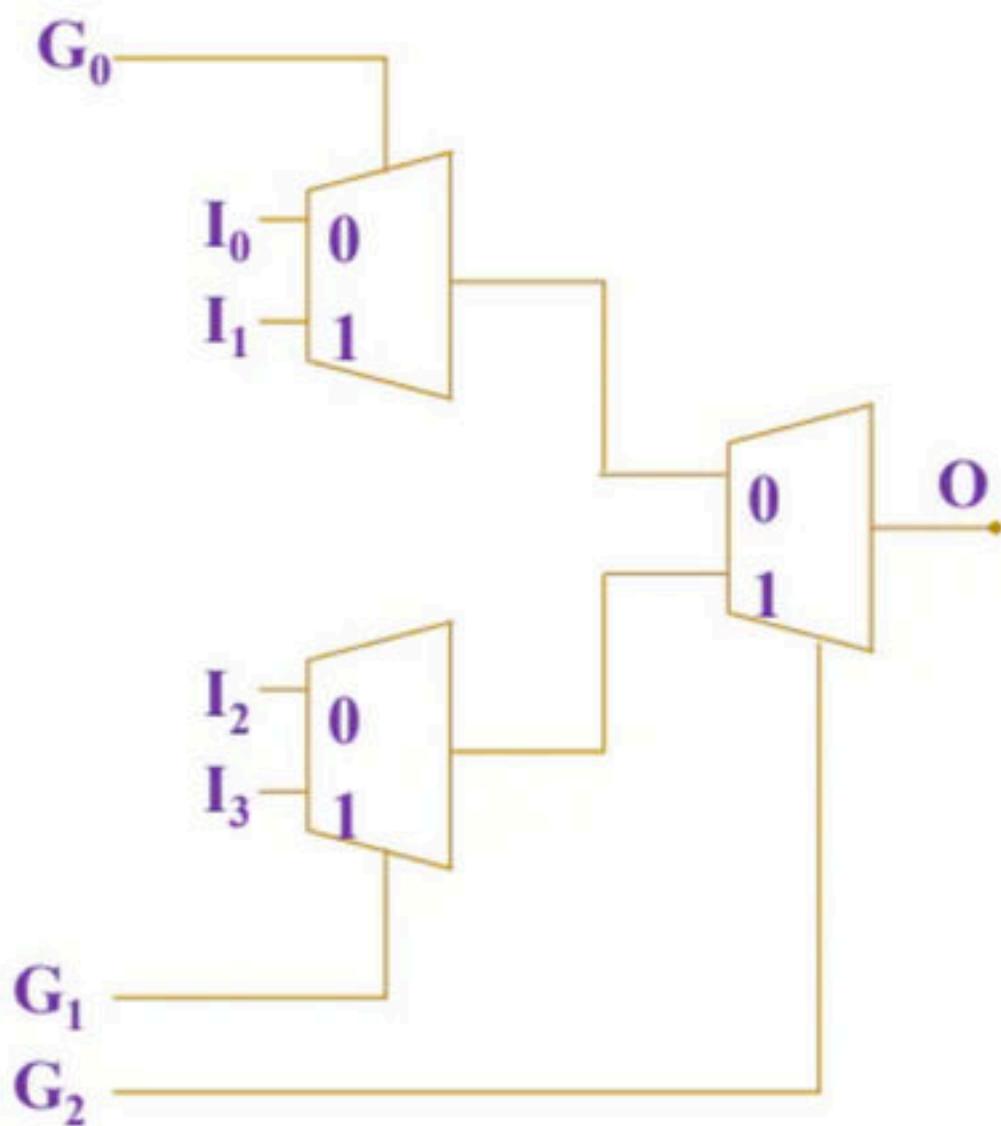
**Q.** A combinational circuit using an 8 x 1 multiplexer as shown in the figure. The minimized expression for the output (Z) is

- (a)  $C(\bar{A} + \bar{B})$
- (b)  $C(A + B)$
- (c)  $\bar{C} + \bar{A}\bar{B}$
- (d)  $\bar{C} + AB$



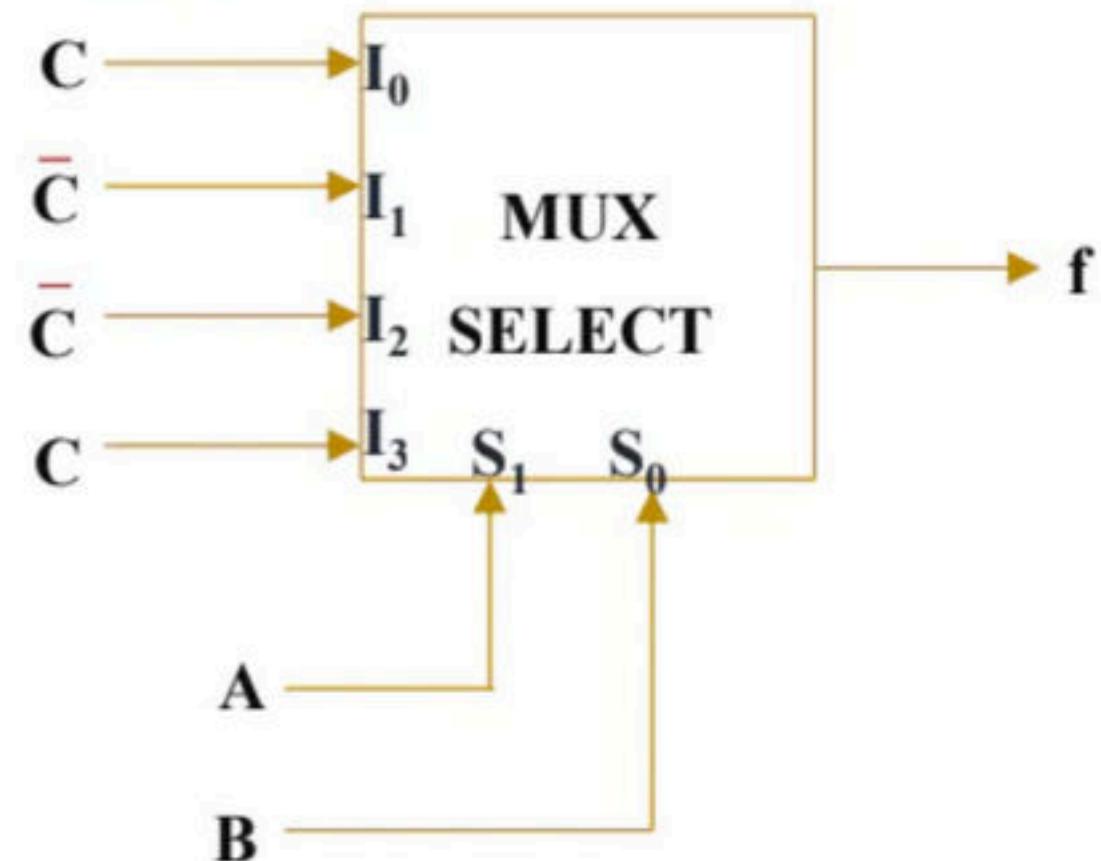
**Q.** The cell of a Field Programmable Gate Array is shown in the figure. It has three 2 to 1 multiplexers with their select lines  $G_0$ ,  $G_1$ ,  $G_2$  and 4 digital signal input lines  $I_0$ ,  $I_1$ ,  $I_2$  and  $I_3$ . The logical function that relates the output O to the select and signal input lines is.

- (a)  $\bar{G}_0\bar{G}_1I_2 + \bar{G}_0\bar{G}_1I_3 + \bar{G}_2\bar{G}_1I_0 + \bar{G}_2\bar{G}_1I_1$
- (b)  $\bar{G}_0I_2 + \bar{G}_0G_1 + \bar{G}_2I_0 + \bar{G}_2\bar{G}_1I_1 + G_0$
- (c)  $\bar{G}_0\bar{G}_2I_0 + G_0\bar{G}_2I_1 + G_2\bar{G}_1I_2 + G_2G_1I_3$
- (d)  $G_2G_1\bar{I}_2 + \bar{G}_2\bar{G}_1\bar{I}_3 + G_2\bar{G}_0I_0 + G_0\bar{G}_2I_1$



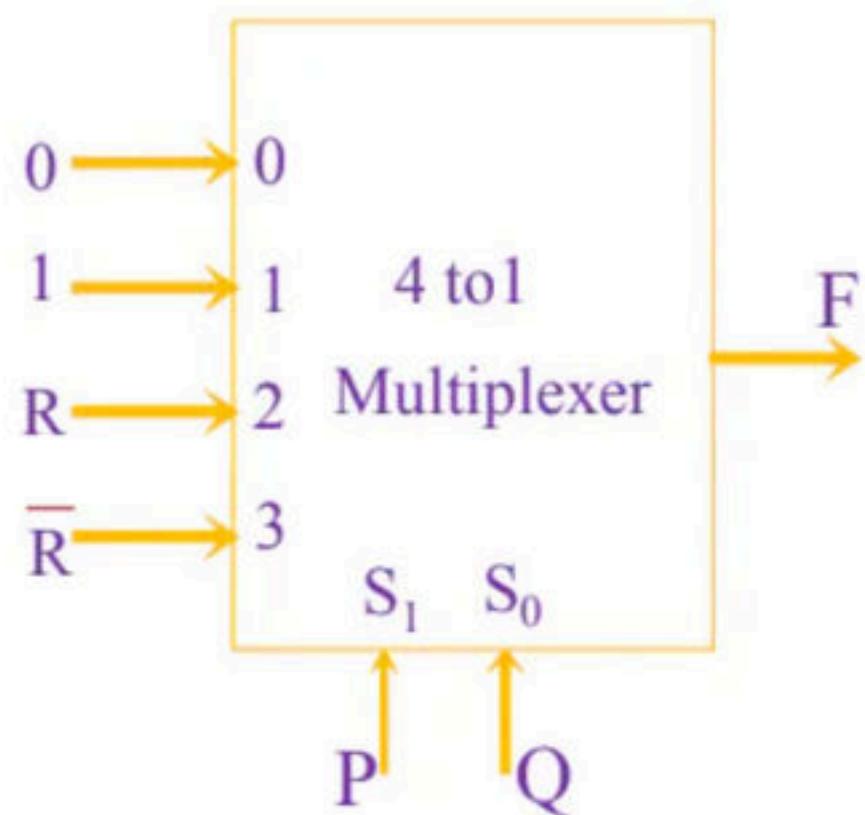
Q. The output 'F' of the multiplexer circuit shown in the figure will be

- (a)  $AB + B\bar{C} + \bar{C}A + \bar{B}\bar{C}$
- (b)  $A \oplus B \oplus C$
- (c)  $A \oplus B$
- (d)  $\overline{ABC} + \overline{ABC} + AB\bar{C} + ABC$



Q. Consider the 4-to-1 multiplexer with two lines  $S_1$  and  $S_0$  given below. The minimal sum of products form of the Boolean expression for the output F of the Multiplexer is

- (A)  $\bar{P}Q + Q\bar{R} + P\bar{Q}R$
- (B)  $\bar{P}Q + \bar{P}Q\bar{R} + PQ\bar{R} + P\bar{Q}R$
- (C)  $\bar{P}QR + \bar{P}Q\bar{R} + Q\bar{R} + P\bar{Q}R$
- (D)  $PQ\bar{R}$



**Q.** Consider the following combinational function block involving four Boolean variables  $x, y, a, b$  where  $x, a, b$  are inputs and  $y$  is the output.

$f(x, y, a, b)$

{

if ( $x$  is 1)  $y = a$ ;

else  $y = b$ ;

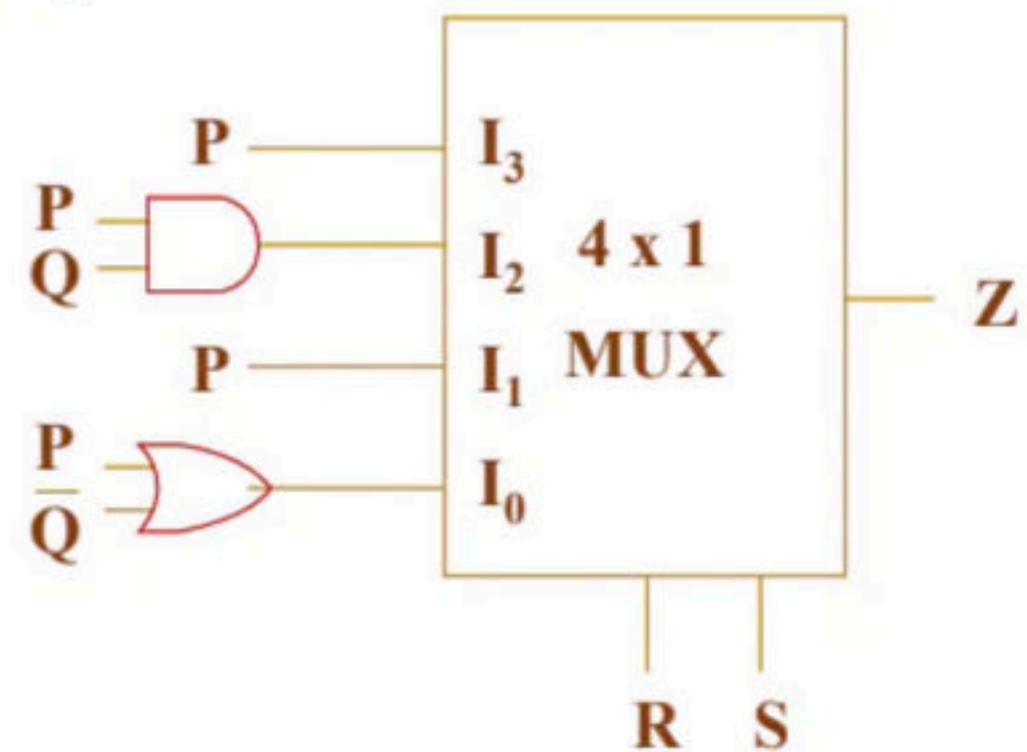
}

Which one of the following digital logic blocks is the most suitable for implementing this function?

- (A) Full adder
- (B) Priority encoder
- (C) Multiplexor
- (D) Flip-flop

**Q.** For the circuit shown in the following figure  $I_0 - I_3$  are inputs to the 4:1 multiplexer R(MSB) and S are control bits. The output Z can be represented by

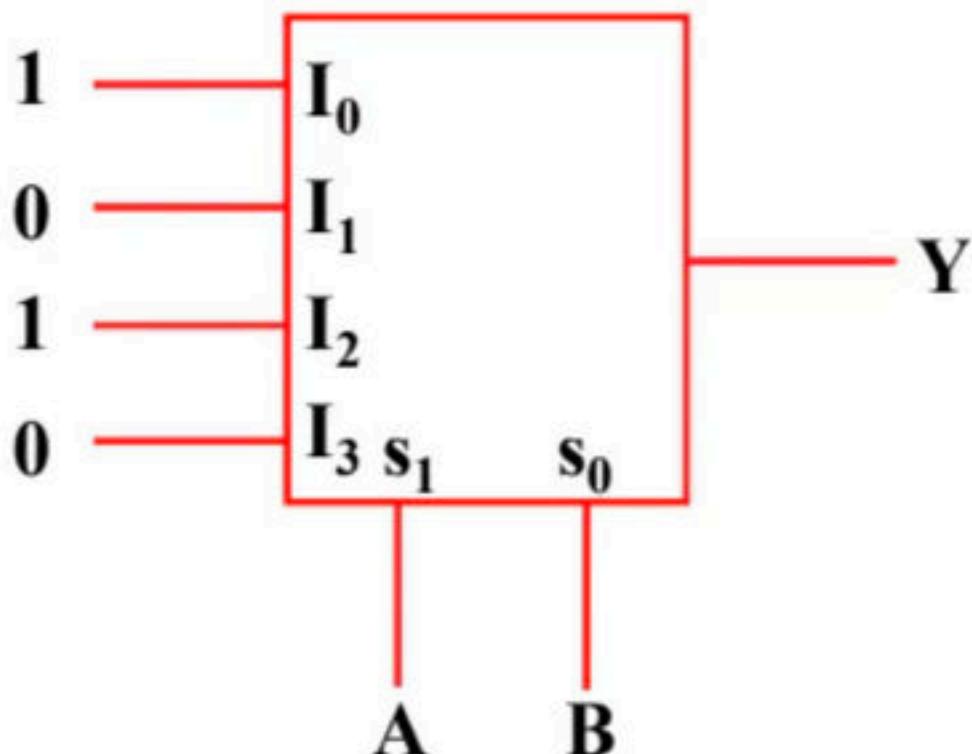
- (a)  $PQ + P\bar{Q}S + \bar{Q}\bar{R}\bar{S}$
- (b)  $P\bar{Q} + PQ\bar{R} + \bar{P}\bar{Q}\bar{S}$
- (c)  $P\bar{Q}\bar{R} + \bar{P}QR + PQRS + \bar{Q}\bar{R}\bar{S}$
- (d)  $PQR + PQRS + P\bar{Q}\bar{R}S + \bar{Q}\bar{R}\bar{S}$



**Q.** The logical expressions of the output of a  $4 \times 1$  multiplexer shown below is

- (a)  $A + \bar{B}$
- (c)  $\bar{A}$

- (b)  $\bar{B}$
- (d) B



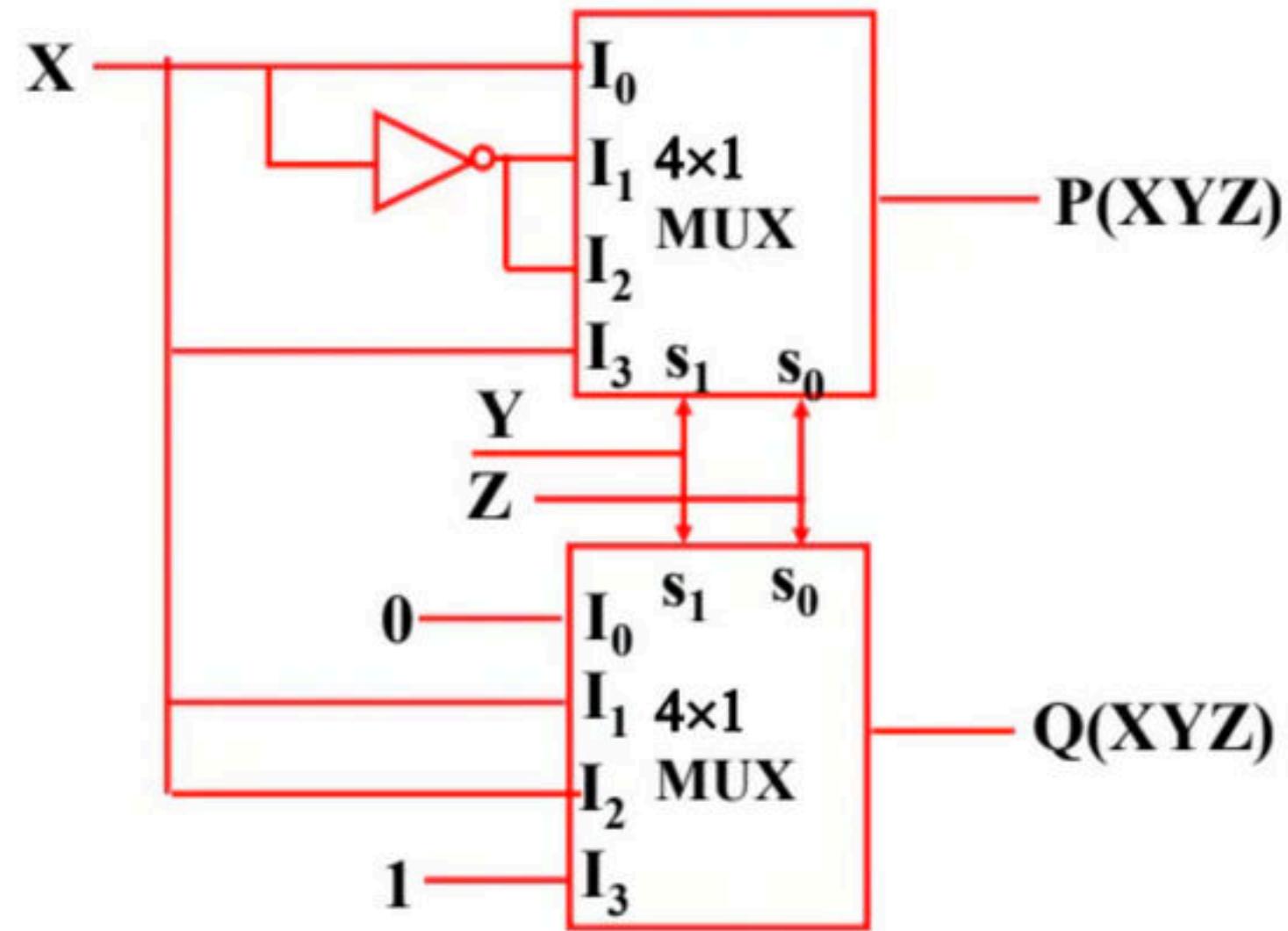
**Q.** The multiplexer circuit function as

(a) Full subtractor

(c) Two output comparator

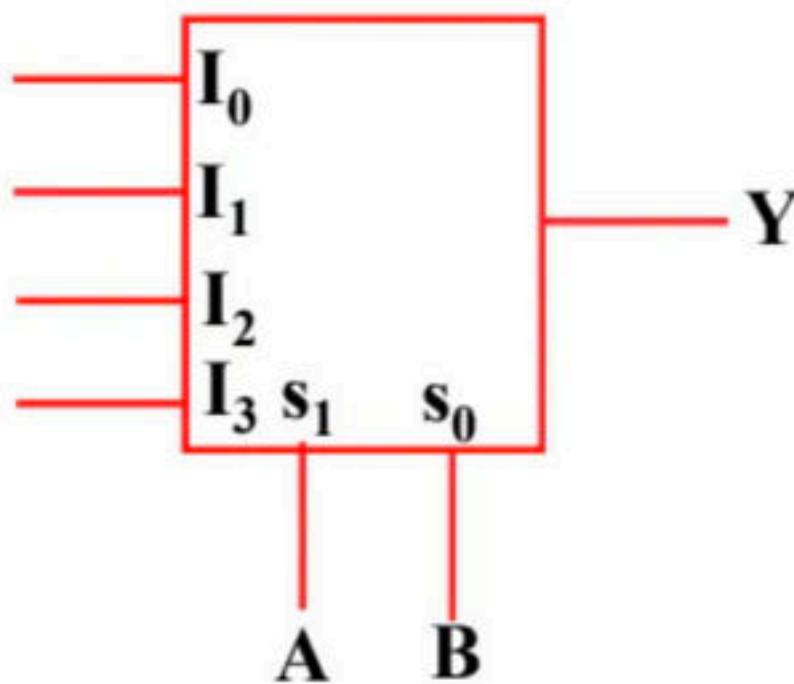
(b) Full adder

(d) Half adder

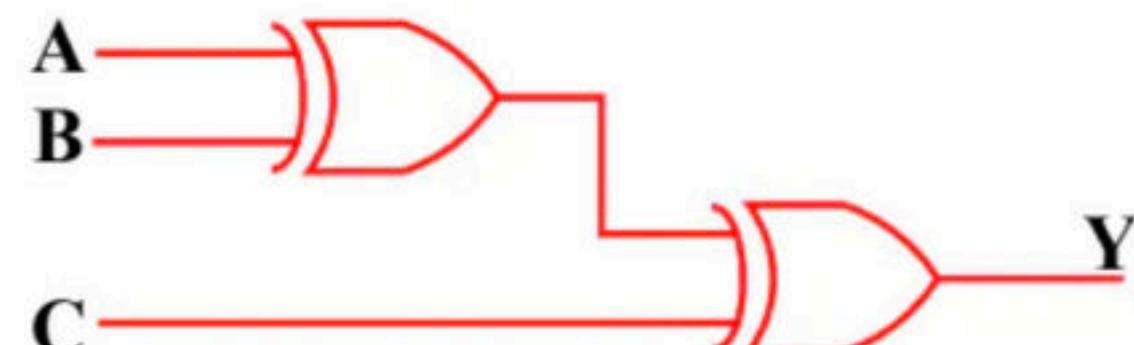


**Q.** If the logic expression of the outputs in the circuit shown in figure A and B are same, then select the correct combination of signals to be connected to the inputs of multiplexer

- |     |       |           |           |       |
|-----|-------|-----------|-----------|-------|
|     | $I_0$ | $I_1$     | $I_2$     | $I_3$ |
| (a) | C     | 0         | $\bar{C}$ | 1     |
| (b) | C     | C         | $\bar{C}$ | C     |
| (c) | C     | $\bar{C}$ | $\bar{C}$ | C     |
| (d) | 1     | C         | $\bar{C}$ | 1     |



**Figure A**



**Figure B**

Q. A combinational circuit using  $4 \times 1$  mux is shown in figure

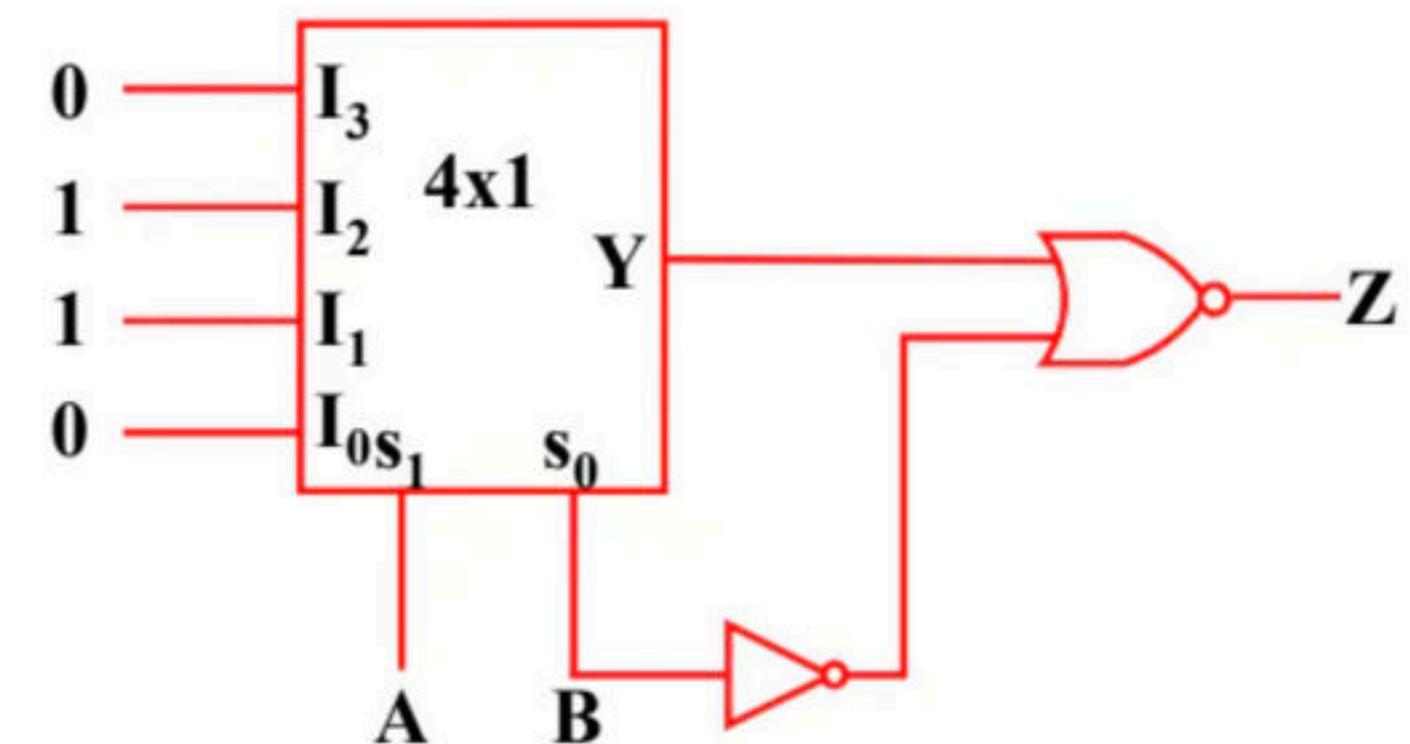
The output Z is

(a)  $A + B$

(b)  $\overline{A} \oplus B$

(c)  $AB$

(d)  $(\overline{A} + \overline{B})$



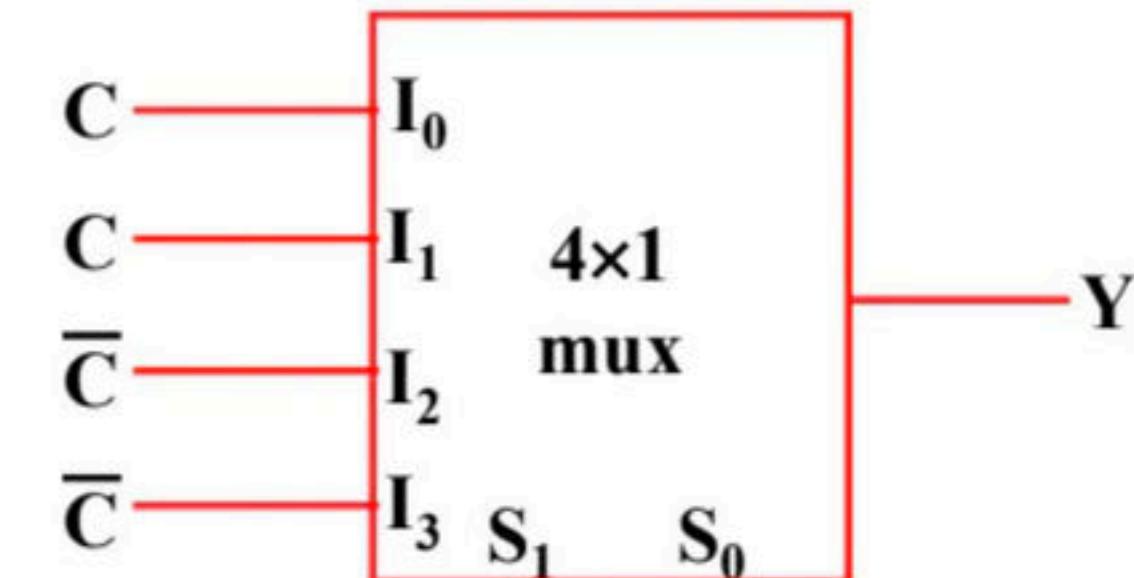
Q. The expression for Y is

(a)  $A \oplus B \oplus C$

(b)  $(A \oplus B)C + AB\bar{C}$

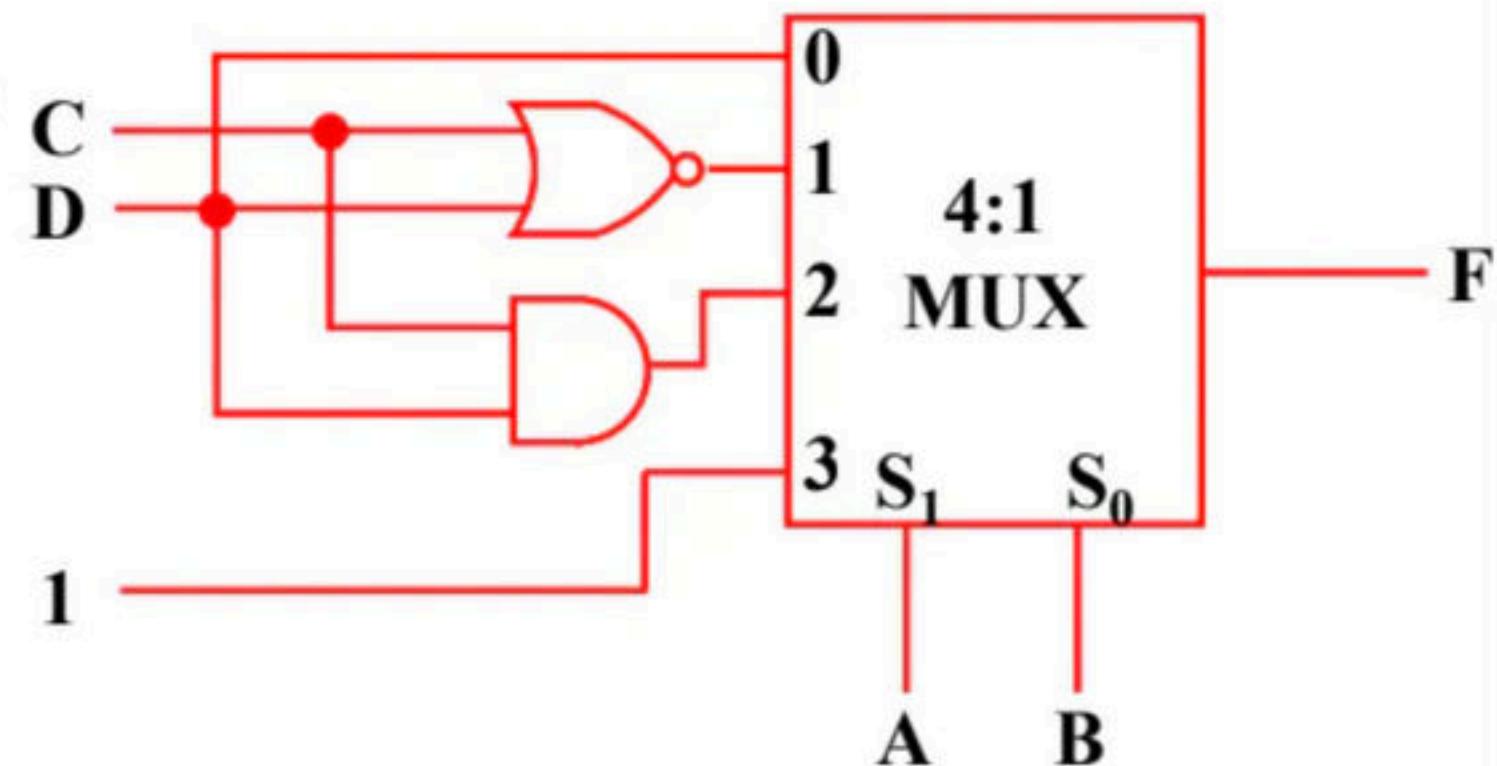
(c)  $AB + A \oplus B$

(d)  $ABC + (A \oplus B)\bar{C}$



Q. The Boolean function realized by the following circuit is

- (a)  $F(A, B, C, D) = \sum(1, 3, 4, 11, 12, 13, 14, 15)$
- (b)  $F(A, B, C, D) = \sum(0, 2, 4, 5, 9, 10, 11)$
- (c)  $F(A, B, C, D) = \sum(1, 8, 14, 15)$
- (d)  $F(A, B, C, D) = \sum(0, 2, 6, 8, 14, 15)$



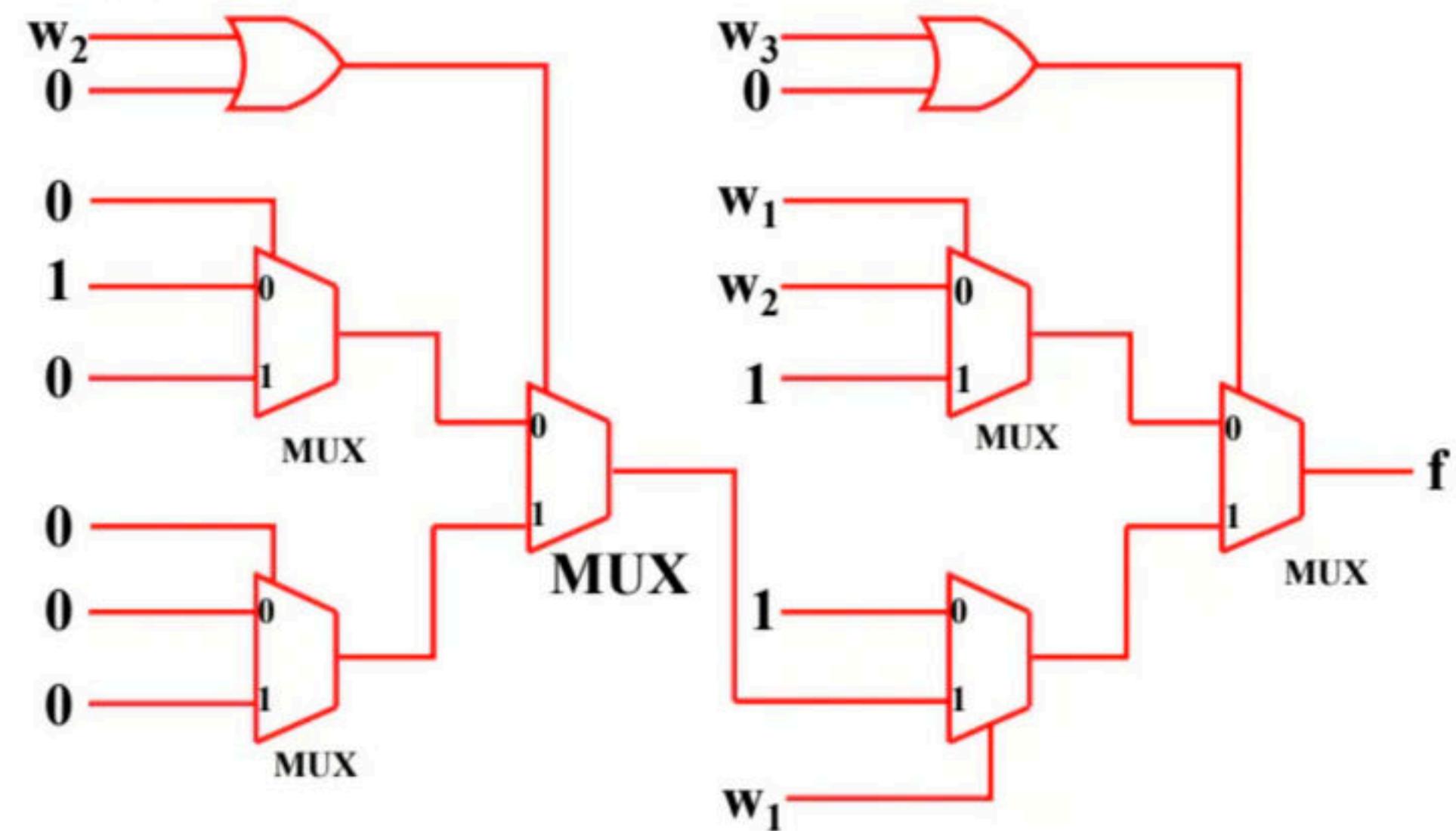
Q. The logic circuit shown below implements

(a)  $(w_1 \oplus w_2) + (w_2 \oplus w_3)$

(b)  $(w_2 \oplus w_1) + (w_1 \oplus w_3)$

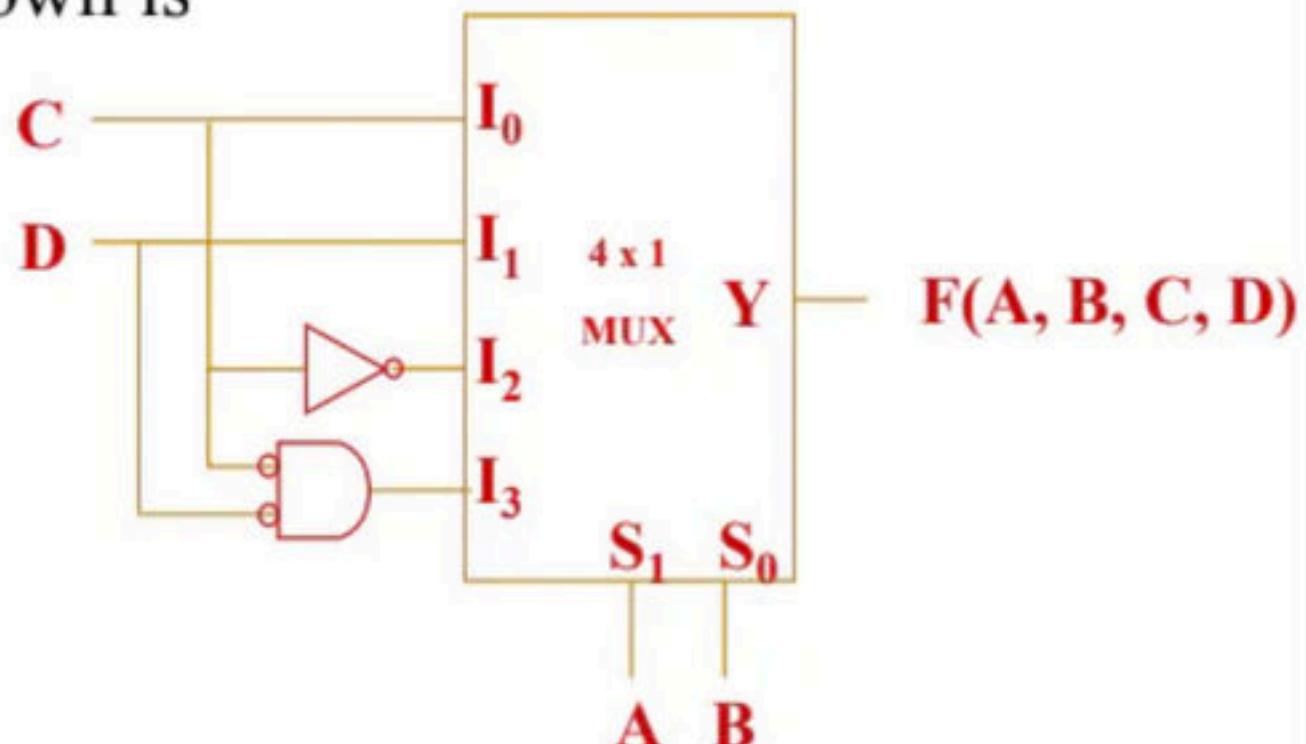
(c)  $(w_2 \oplus w_3) + (w_1 \oplus w_3)$

(d)  $(w_1 \oplus w_2 \oplus w_3)$



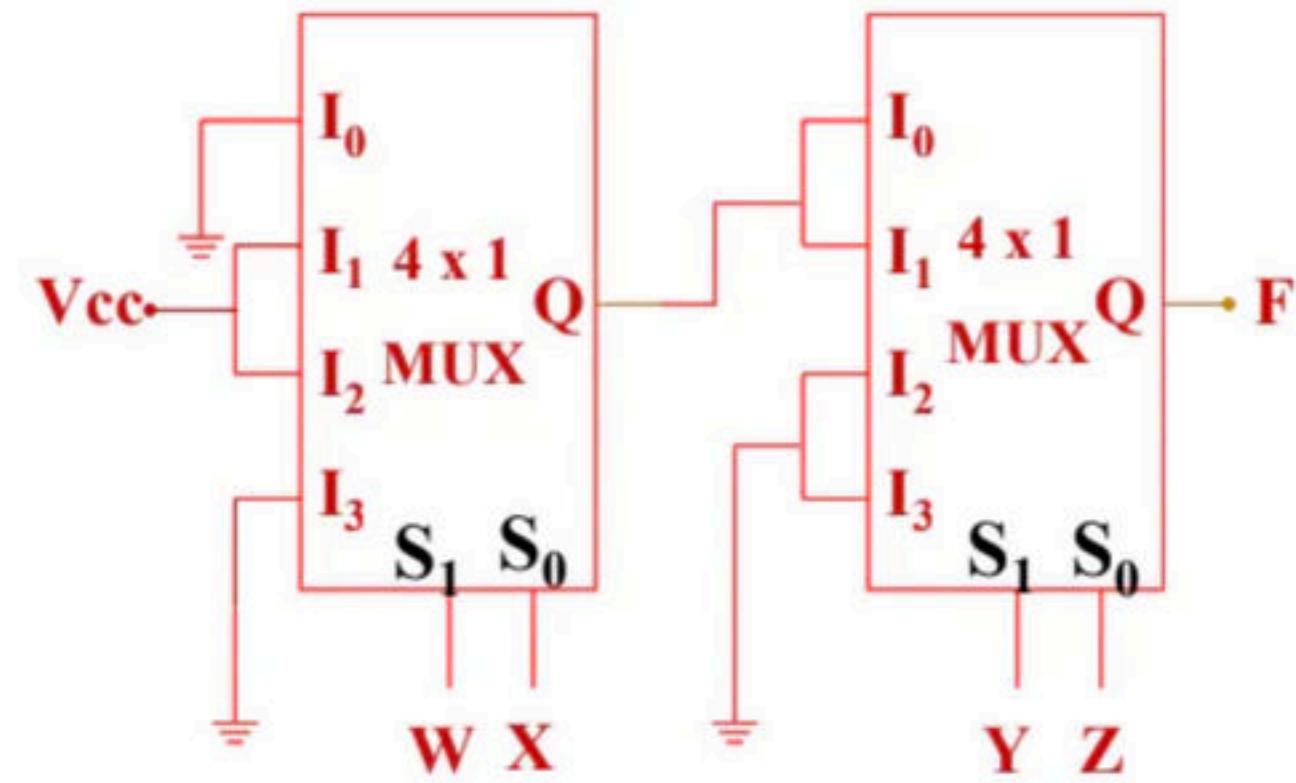
**Q.** The Boolean function realized by the logic circuit shown is

- (a)  $F = \sum m(0,1,3,5,9,10,14)$
- (b)  $F = \sum m(2,3,5,7,8,12,13)$
- (c)  $F = \sum m(1,2,4,5,11,14,15)$
- (d)  $F = \sum m(2,3,5,7,8,9,12)$

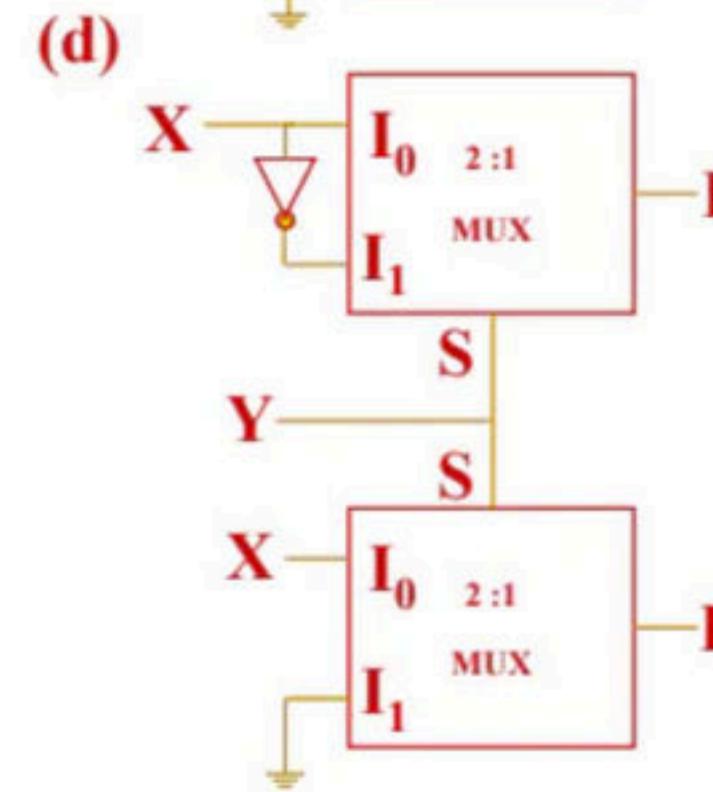
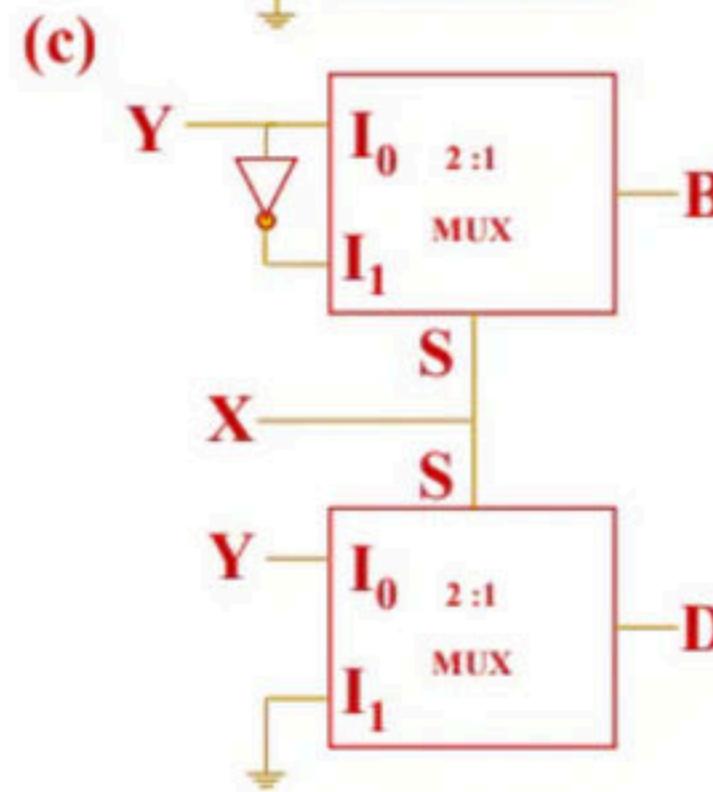
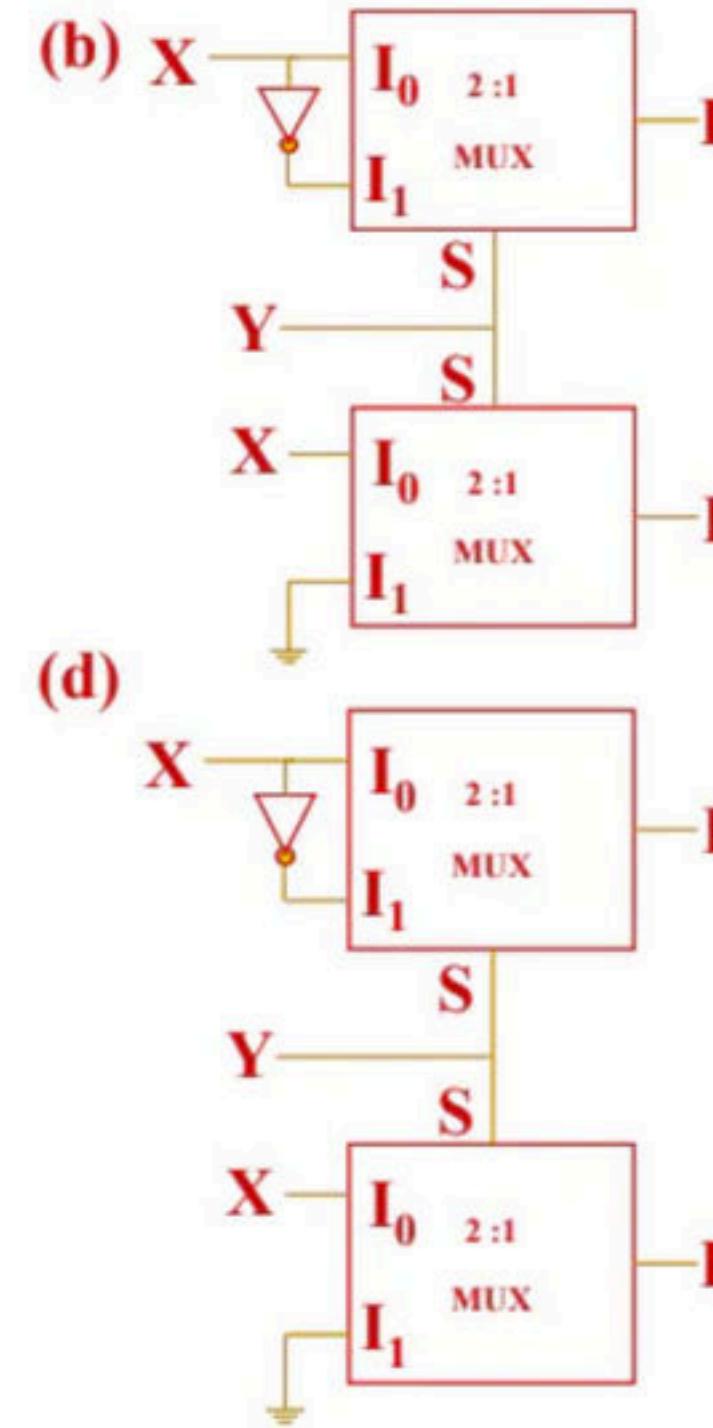
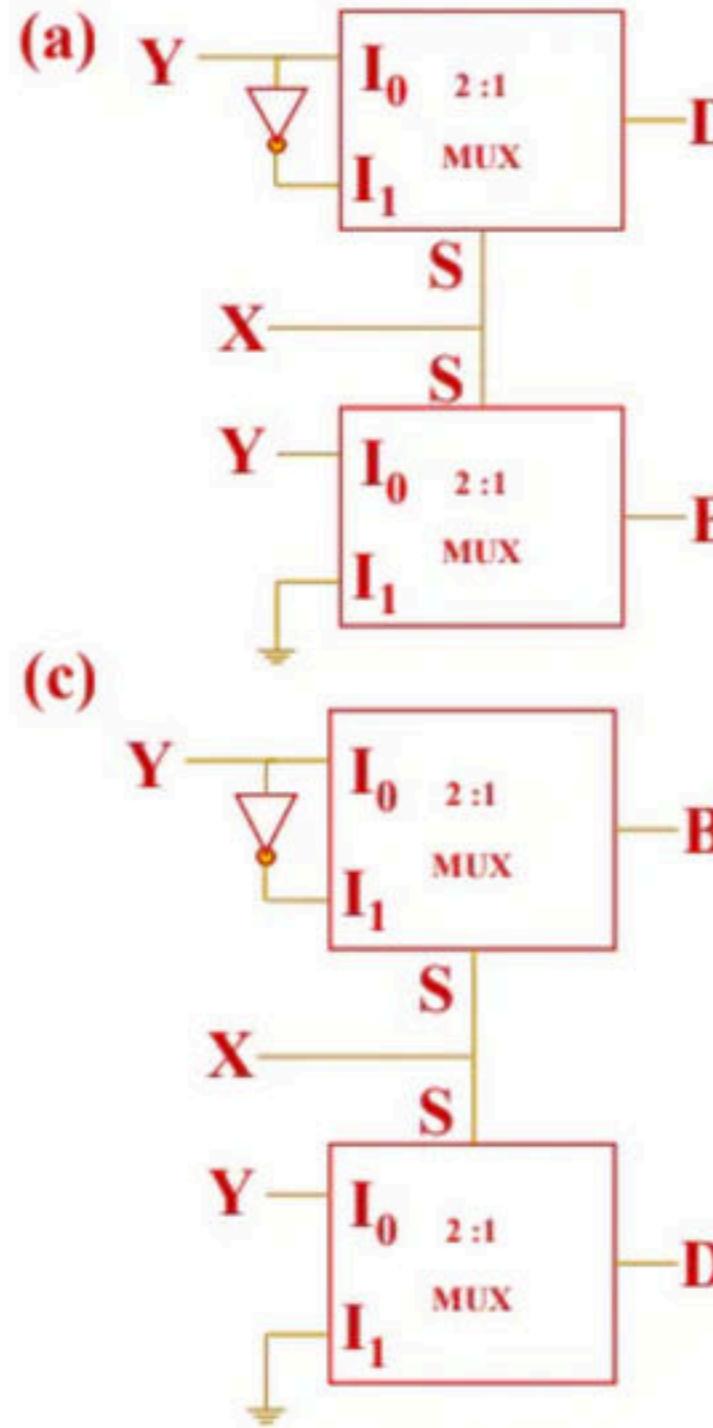


Q. In the circuit shown, W and Y are MSBs of the control inputs. The output F is given by

- (a)  $F = W\bar{X} + \bar{W}X + \bar{Y}\bar{Z}$
- (b)  $F = W\bar{X} + \bar{W}X + \bar{Y}Z$
- (c)  $F = W\bar{X}\bar{Y} + \bar{W}XY$
- (d)  $F = (\bar{W} + \bar{X}) + \bar{Y}Z$



Q. If X and Y are inputs and the Difference (D = X - Y) and the Borrow (B) are the outputs, which one of the following diagrams implements a half subtractor?



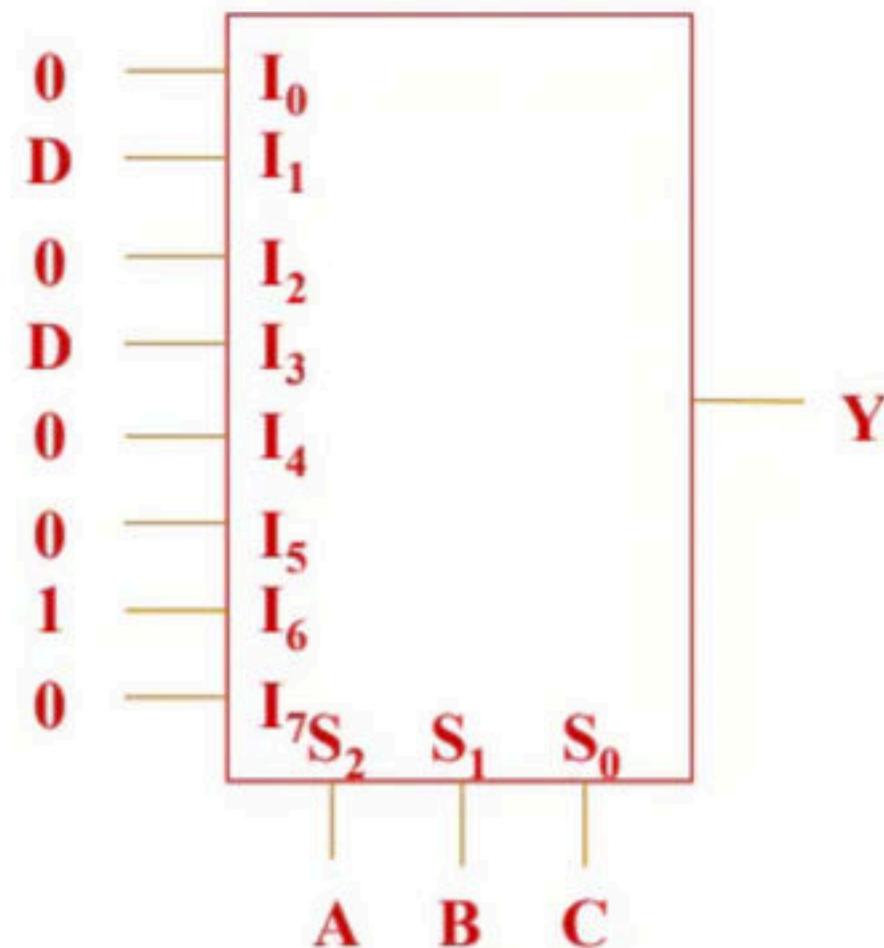
q. An 8-to-1 multiplexer is used to implement logical function Y as shown in the figure. The output Y is given by.

(a)  $Y = A\bar{B}C + A\bar{C}D$

(c)  $Y = AB\bar{C} + \bar{A}CD$

(b)  $Y = \bar{A}BC + A\bar{B}D$

(d)  $Y = \bar{A}\bar{B}D + A\bar{B}C$



Q) Design a logic circuit  $f(A, B, C) = \sum m(0, 1, 3, 6, 7)$  using suitable MUX

Q) Design a logic circuit  $f(A, B, C) = \sum m(0, 1, 3, 6, 7)$  using  $4 \times 1$  MUX

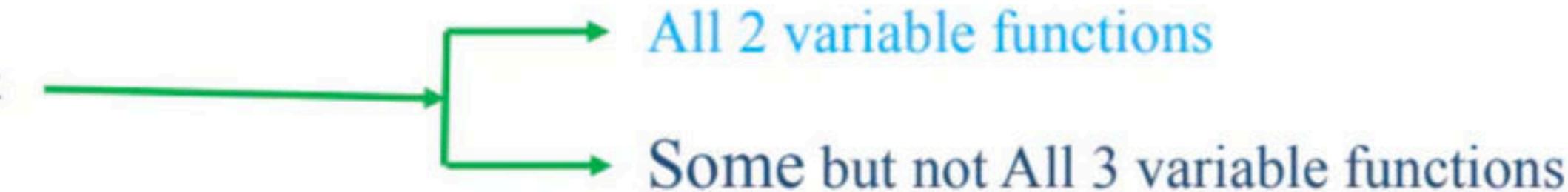
- a) AB as select lines
- b) BC as select lines
- c) AC as select lines

## **BC as select lines**

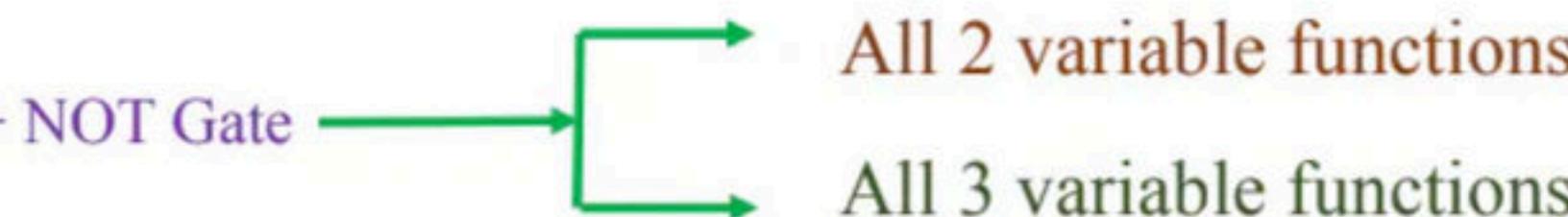
## **AC as select lines**

## Note :

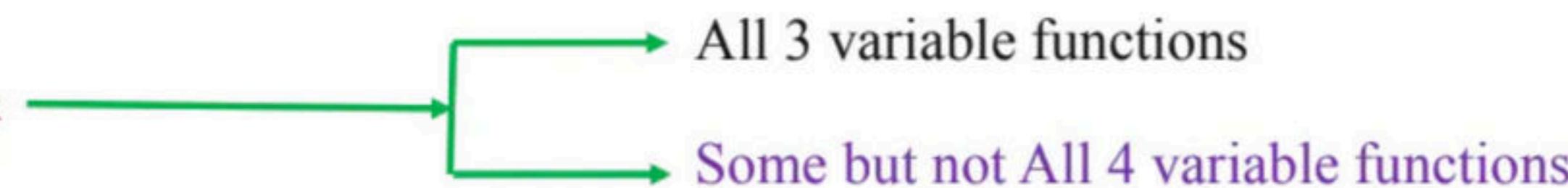
1. By using one  $4 \times 1$  Mux



2. By using one  $4 \times 1$  Mux + NOT Gate



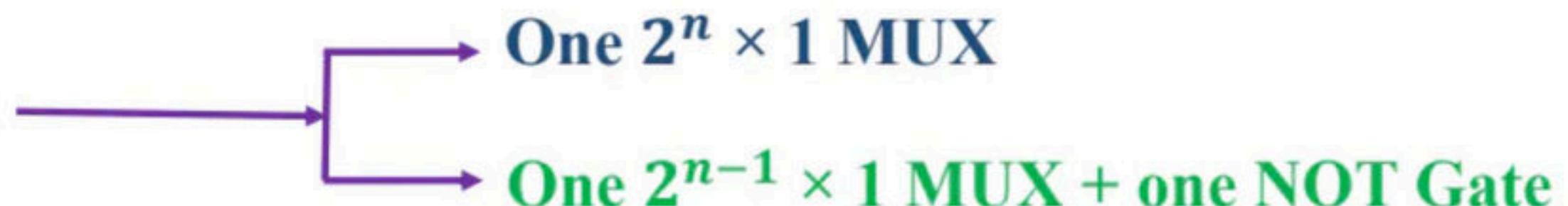
3. By using one  $8 \times 1$  Mux



4. By using one  $8 \times 1$  Mux + NOT Gate



5. n-variable function



Q) Suppose only one mux and one inverter are allowed to be used to implement Boolean function of n- variables , what is the minimum size of the mux needed

- a) $2^n \times 1$  MUX**
- b) $2^{n+1} \times 1$  MUX**
- c) $2^{n-1} \times 1$  MUX**
- d) $2^{n-2} \times 1$  MUX**

Q) Without using any additional circuitry an  $8 \times 1$  mux can be used to obtain

- a) Some but not all Boolean functions of 3 variables
- b) All functions of 3 variable & none of 4- variables
- c) All function's of 4 variables
- d) All functions of 3 variables and some but not all functions of 4 variables

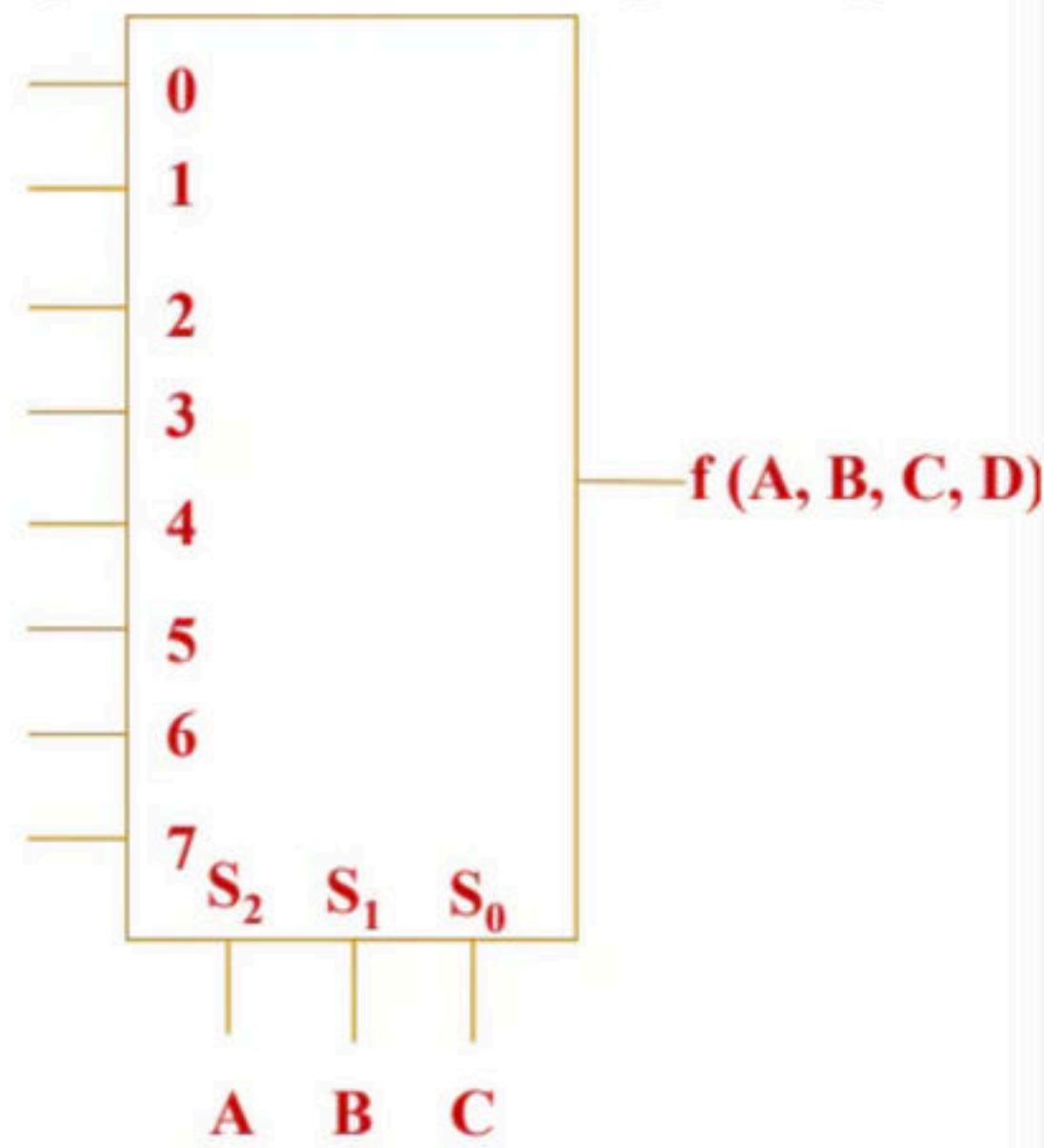
**Q.** A Boolean function  $f(A, B, C, D) = \pi(1, 5, 12, 15)$  is to be implemented using an  $8 \times 1$  multiplexer (A is MSB). The inputs ABC are connected to the select inputs  $S_2 S_1 S_0$  of the multiplexer, respectively. Which one of the following options gives the correct inputs to pins 0,1,2,3,4,5,6,7 in order?

(a)  $D, 0, D, 0, 0, 0, \bar{D}, D$

(c)  $D, 1, D, 1, 1, 1, \bar{D}, D$

(b)  $\bar{D}, 1, \bar{D}, 1, 1, 1, D, \bar{D}$

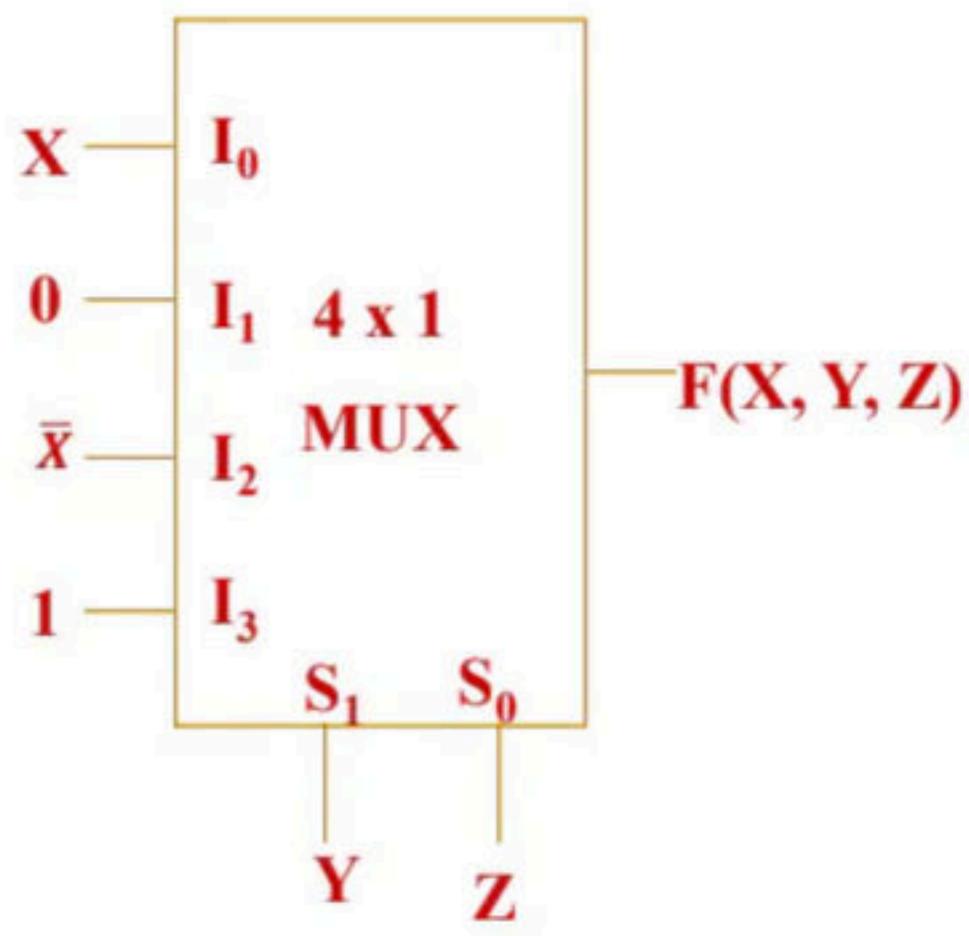
(d)  $\bar{D}, 0, \bar{D}, 0, 0, 0, D, \bar{D}$



**Q.** A 4 to 1 multiplexer to realize a Boolean function  $F(X, Y, Z)$  is shown in the figure below. The inputs  $Y$  and  $Z$  are connected to the selectors of the MUX ( $Y$  is more significant). The canonical sum-of-product expression for  $F(X, Y, Z)$  is

- (a)  $\sum m(2,3,4,7)$
- (c)  $\sum m(0,2,4,6)$

- (b)  $\sum m(1,3,5,7)$
- (d)  $\sum m(2,3,5,6)$

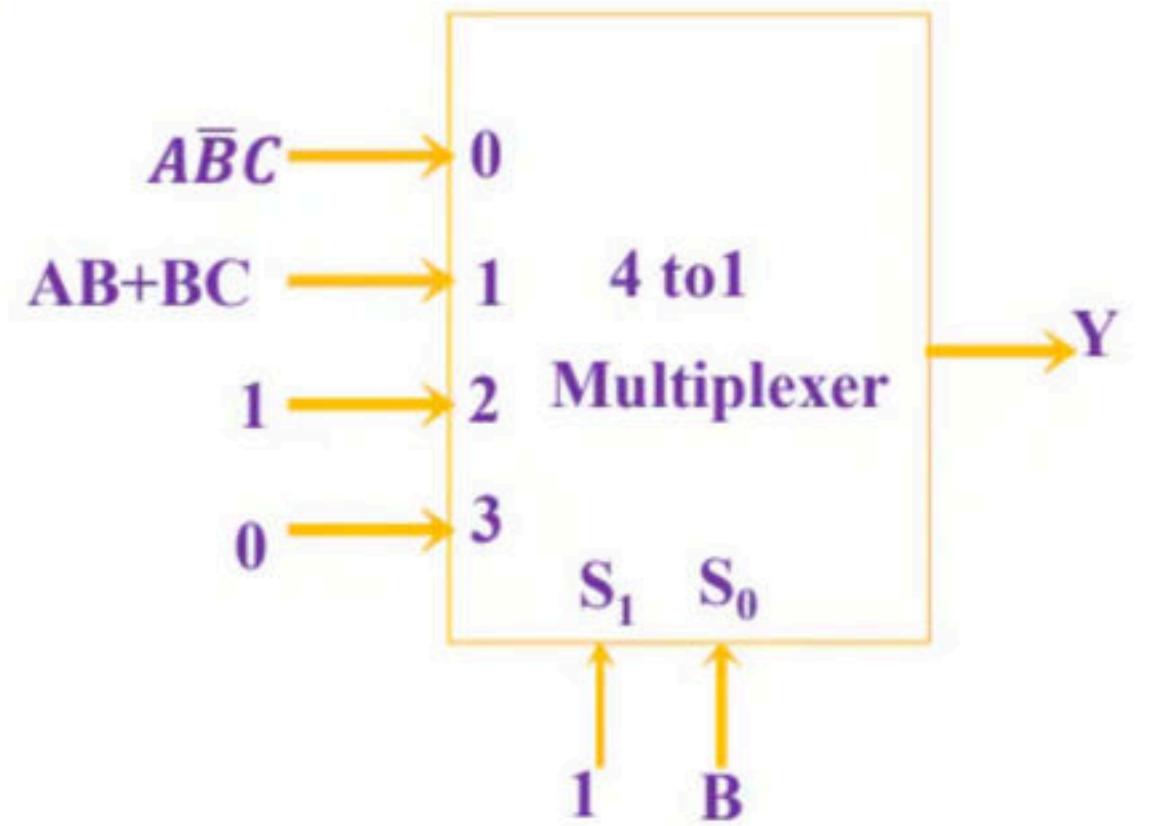


Q) Design a logic circuit  $F(A, B, C) = \sum m(0, 3, 6, 7)$  using  $2 \times 1$  MUX by using A as select line

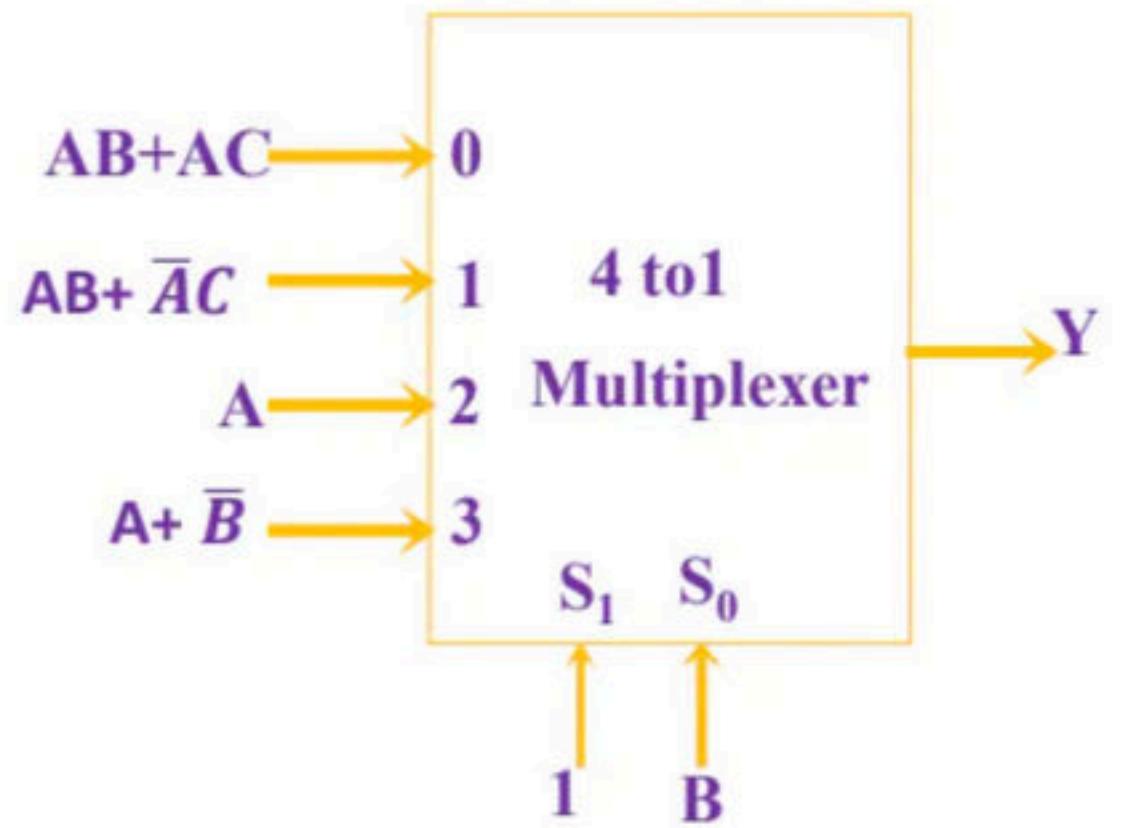
Q) Design a logic circuit  $F(A, B, C) = \sum m(0, 3, 6, 7)$  using  $2 \times 1$  MUX by using B as select line

# MUX as Universal Gate

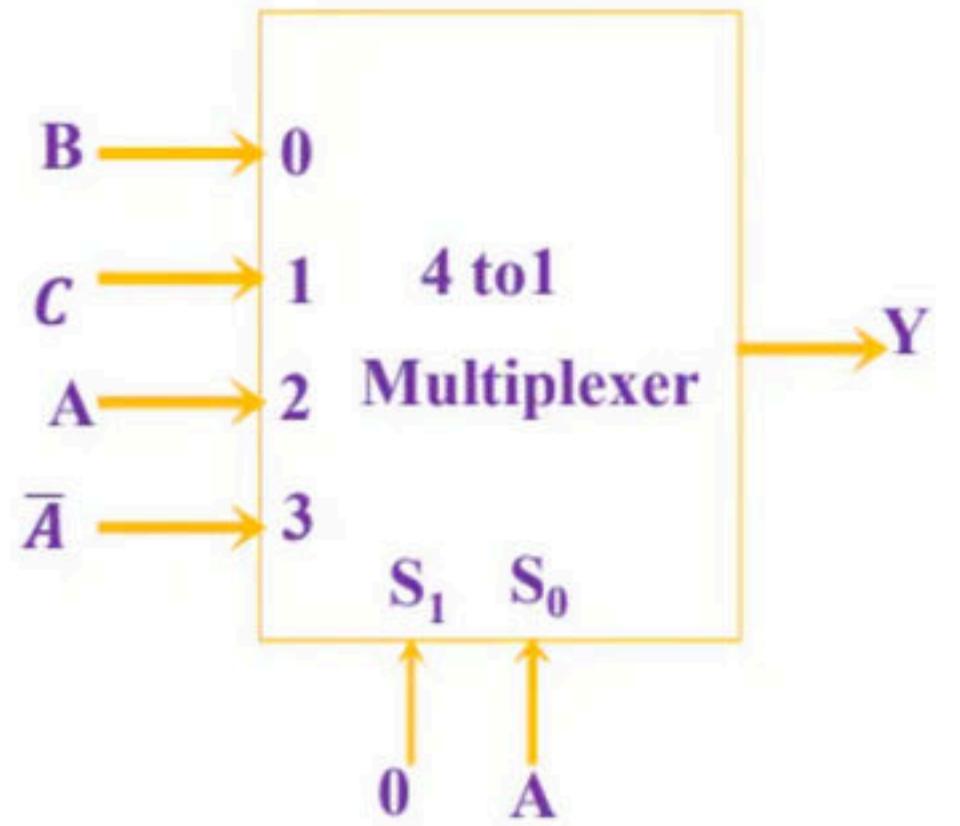
Q) Find the logic expression



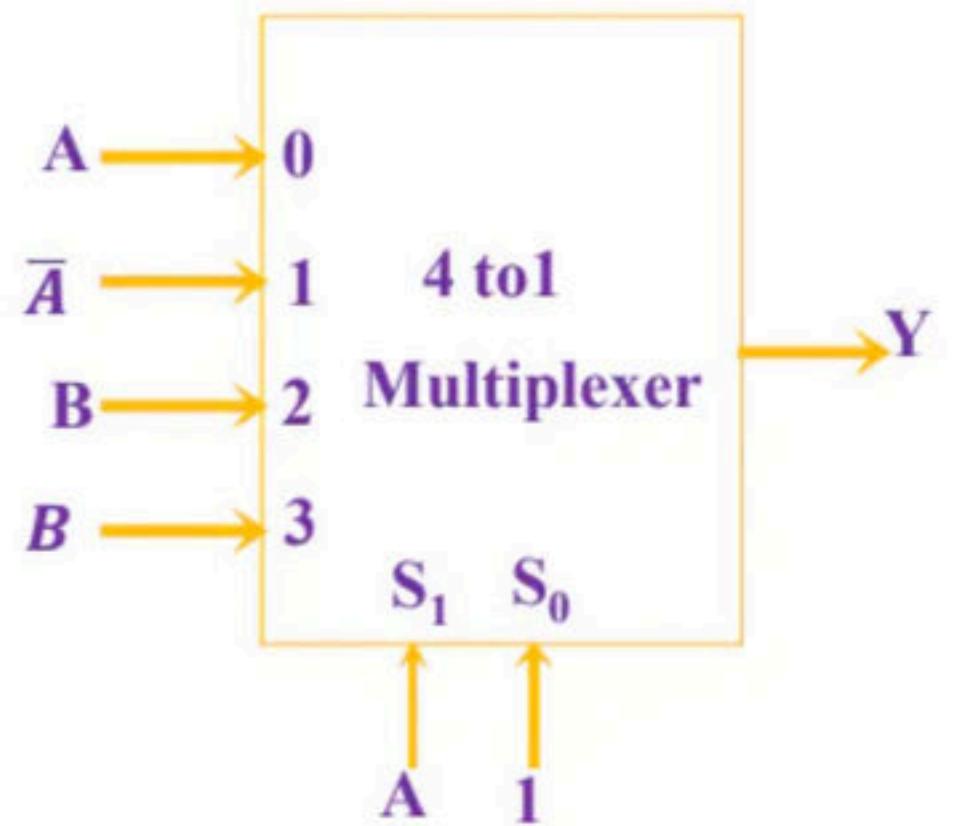
Q) Find the logic expression



Q) Find the logic expression



Q) Find the logic expression



# Implementation of Higher order MUX using lower order MUX

Q) Design  $4 \times 1$  MUX using  $2 \times 1$  MUX

Q) Design  $8 \times 1$  MUX using  $2 \times 1$  MUX

$$\frac{8}{2} = 4 \rightarrow b_1 \rightarrow C . f(A_1, B_1, C)$$

$$\frac{4}{2} = 2 \rightarrow b_2 \rightarrow B$$

$$\frac{2}{2} = 1. \rightarrow b_3 \rightarrow MSB(A)$$

$$\begin{array}{c} \cancel{f} \\ \hline \end{array}$$

Q) Design  $32 \times 1$  MUX using  $4 \times 1$  MUX

$$\frac{32}{4} = 8$$



$L_1(A, E)$

$$\frac{8}{4} = 2$$



$L_2(B, C)$   $f(\cancel{A}, \cancel{B}, \cancel{C}, D, E)$

$$\frac{2}{4} = 1$$

( $2 \times 1$  MUX)  $\rightarrow L_3 \rightarrow (\text{MSB})(A)$   
( $4 \times 1$  MUX)

Q) Design  $8 \times 1$  MUX using  $4 \times 1$  MUX

$$\frac{8}{4} = 2 \rightarrow L_1 \rightarrow B, C$$

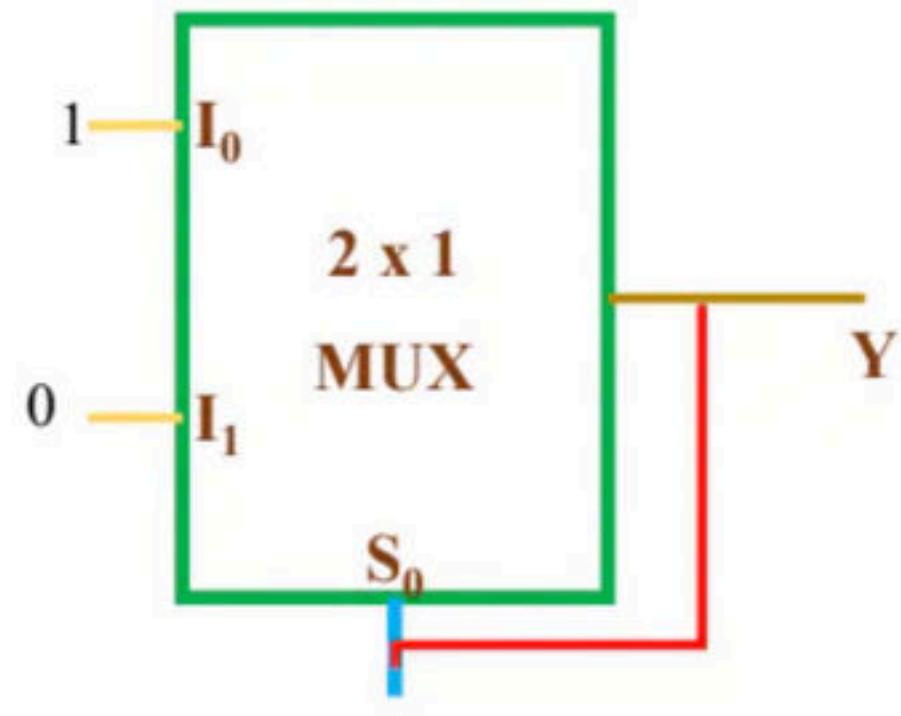
$$\frac{2}{4} = 1 \quad (2 \times 1 \text{ mux}) - L_2 \rightarrow A$$

3.

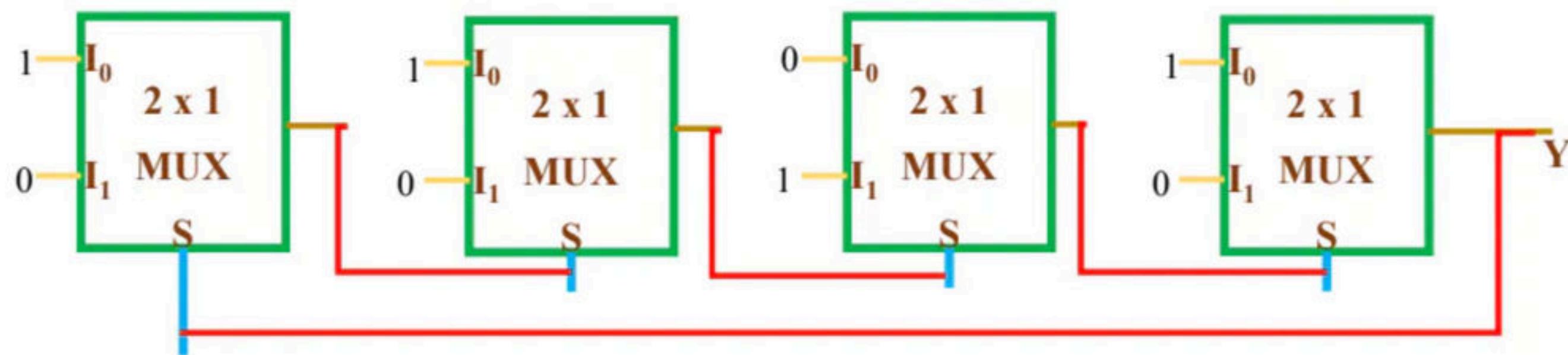
$$f(A_1, B_1, C)$$

# Delay Analysis of MUX

Q) Draw the output waveform of the circuit , if the delay of the MUX is tpd

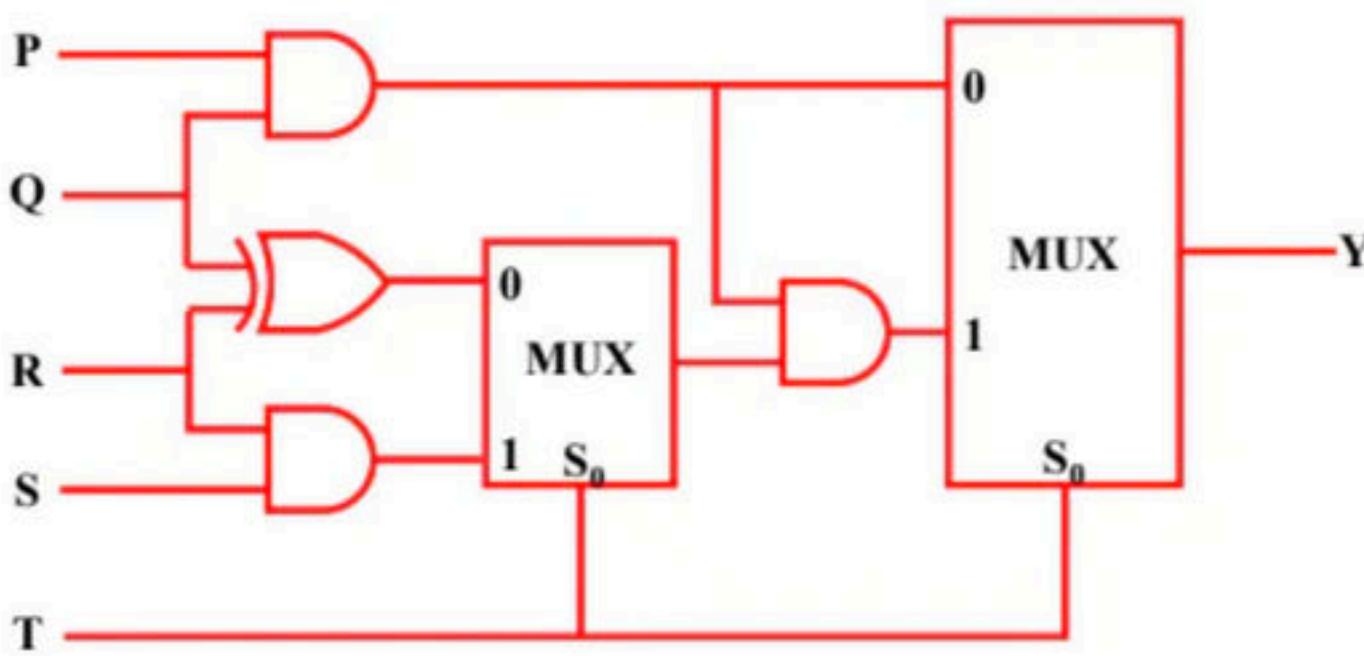


Q) Find the delay of the output Y , if the delay of each mux is 1ns



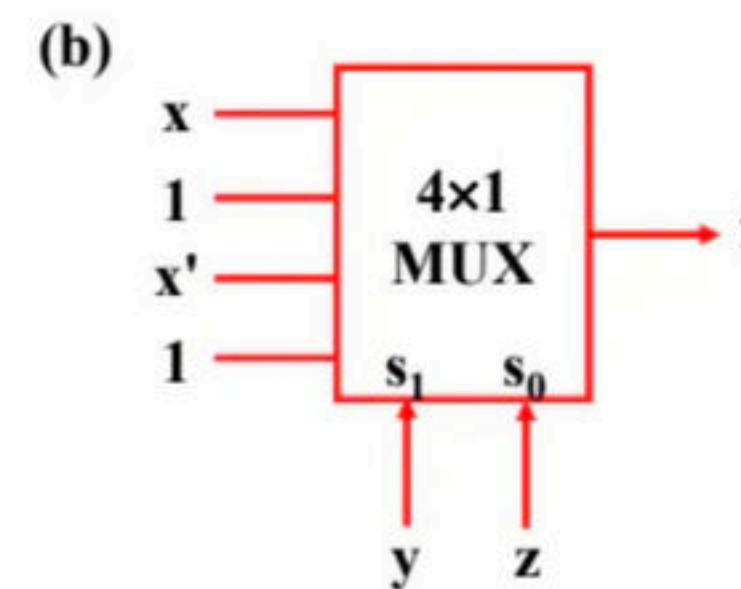
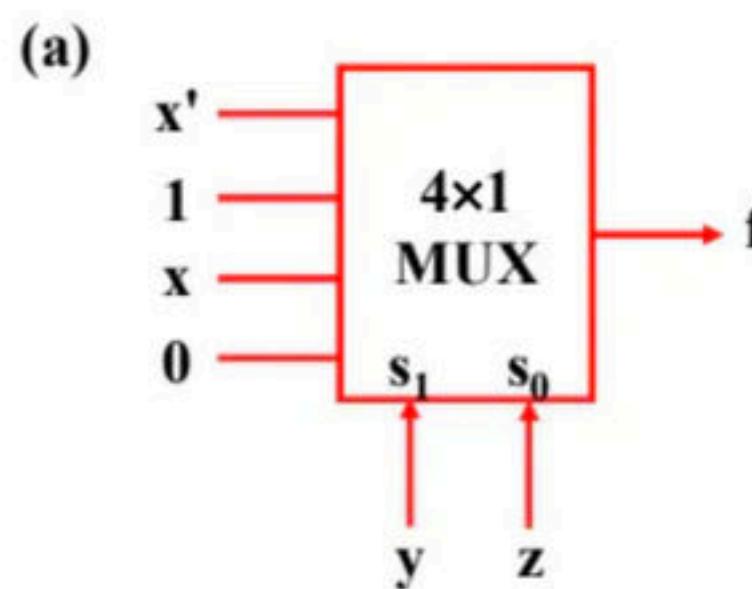
Q. The propagation delays of the XOR gate, AND gate and multiplexer (MUX) in the circuit shown in the figure are 4 ns, 2 ns and 1 ns, respectively. If all the inputs P, Q, R, S and T are applied simultaneously and held constant, the maximum propagation delay of the circuit is

- (a) 3 ns
- (b) 5 ns
- (c) 6 ns
- (d) 7 ns

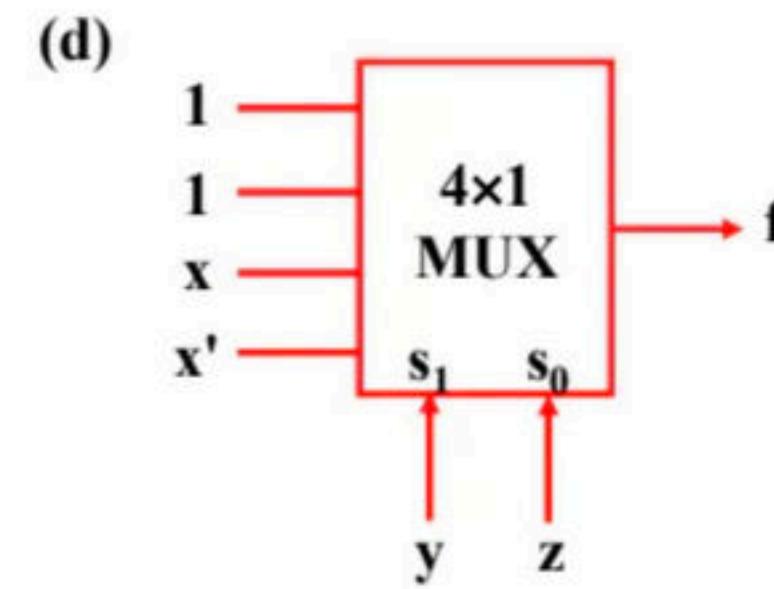
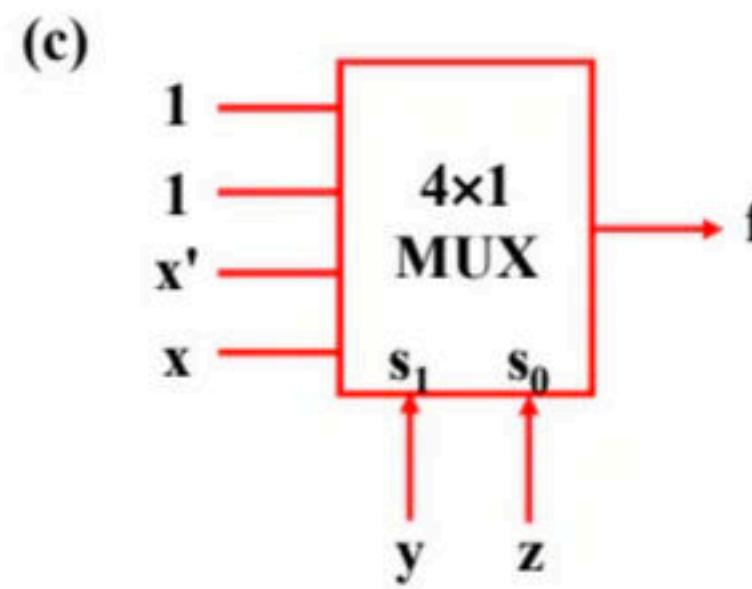


**Q.** Which one of the following circuits implements the Boolean function given below?

$$f(x, y, z) = m_0 + m_1 + m_3 + m_4 + m_5 + m_6, \text{ where } m_i \text{ is the } i^{\text{th}} \text{ minterm.}$$



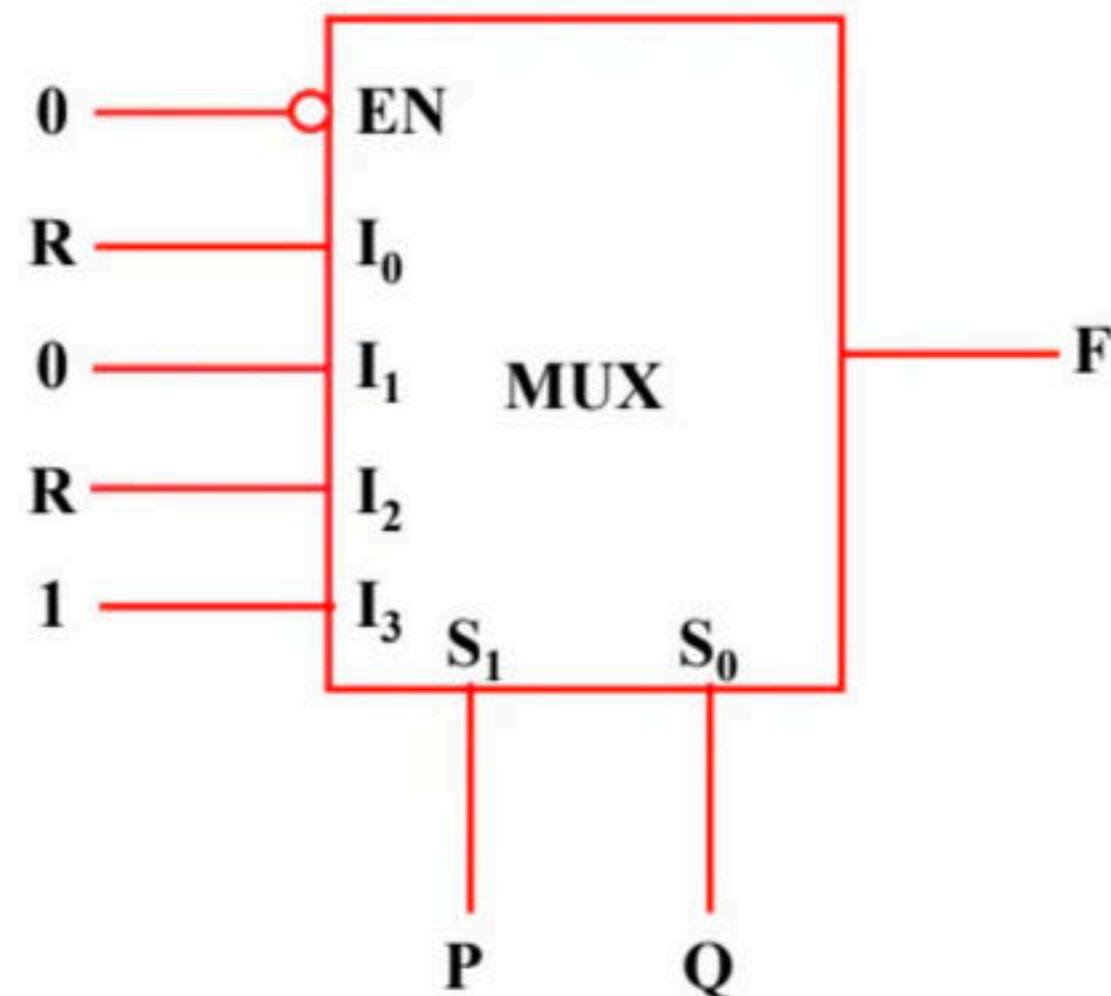
①



**Q.** The figure below shows a multiplexer where  $S_1$  and  $S_0$  are the select lines.  $I$  to  $I_0$  are the input data lines, EN is the enable line, and  $F(P, Q, R)$  is the output.  $F$  is

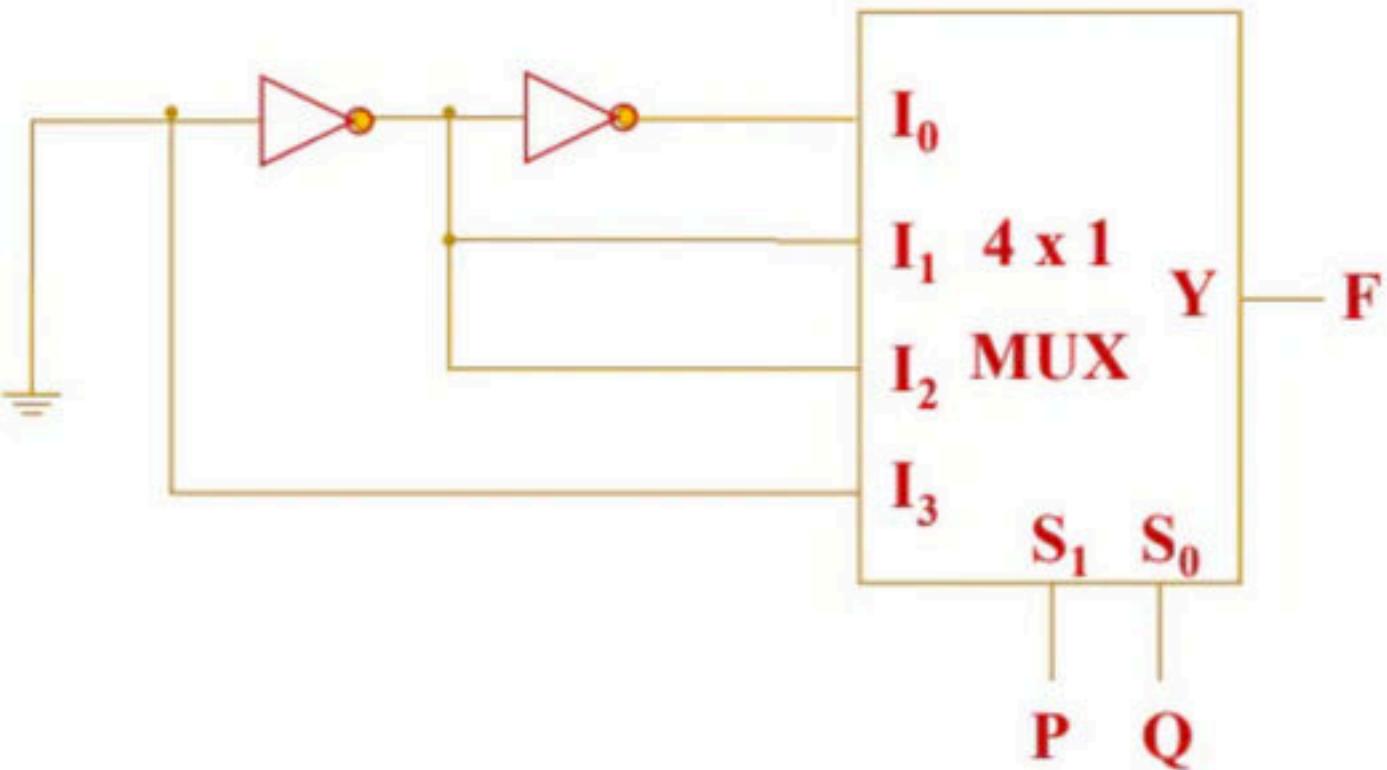
- (a)  $\bar{Q} + PR$ .
- (b)  $P + Q\bar{R}$ .
- (c)  $PQ + \bar{Q}R$ .
- (d)  $P\bar{Q}R + \bar{P}Q$ .

②



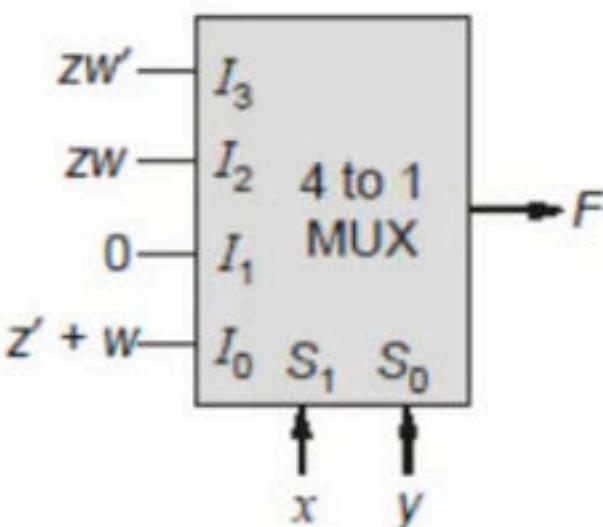
**Q.** The logic function implemented by the circuit below is (ground implies logic 0)

- (a)  $F = \text{AND}(P, Q)$
- (b)  $F = \text{XOR}(P, Q)$
- (c)  $F = \text{XNOR}(P, Q)$
- (d)  $F = \text{OR}(P, Q)$



③

A  $4 \times 1$  multiplexer with two selector lines is used to realize a Boolean function,  $F$  having four Boolean variables  $X, Y, Z$  and  $W$  as shown below.  $S_0$  and  $S_1$  denote the least significant bit (LSB) and most significant bit (MSB) of the selector lines of the multiplexer respectively.  $I_0, I_1, I_2, I_3$  are the input lines of the multiplexer.



The canonical sum of product representation of  $F$  is

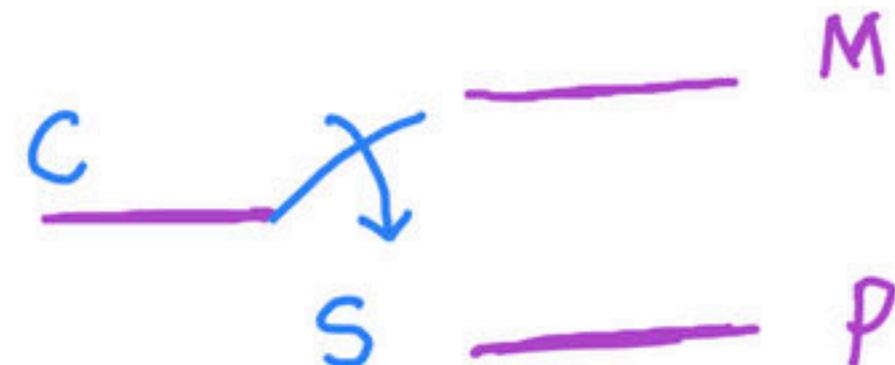
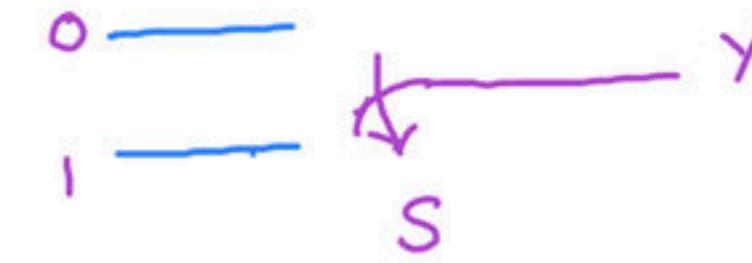
- (a)  $F(X, Y, Z, W) = \Sigma m(0, 1, 3, 14, 15)$
- (b)  $F(X, Y, Z, W) = \Sigma m(0, 1, 3, 11, 14)$
- (c)  $F(X, Y, Z, W) = \Sigma m(2, 5, 9, 11, 14)$
- (d)  $F(X, Y, Z, W) = \Sigma m(1, 3, 7, 9, 15)$

④

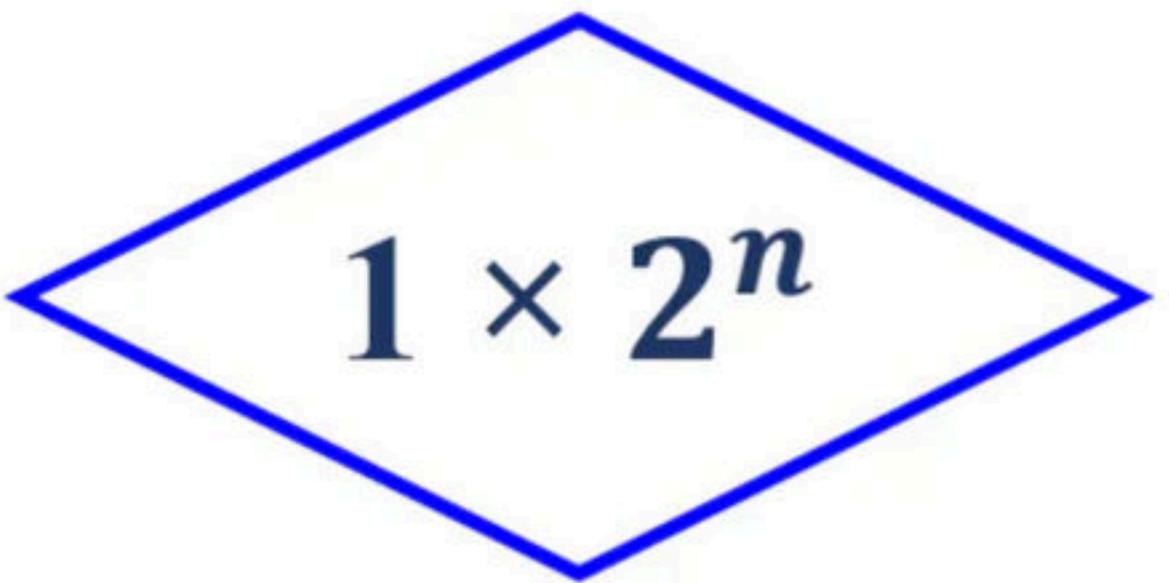
# Demultiplexer

A demultiplexer is a circuit that receives information on a single line and transmits to one of the  $2^n$  possible output lines , according to the selection lines .

- One input to many output
- Data distributor
- One to many circuit



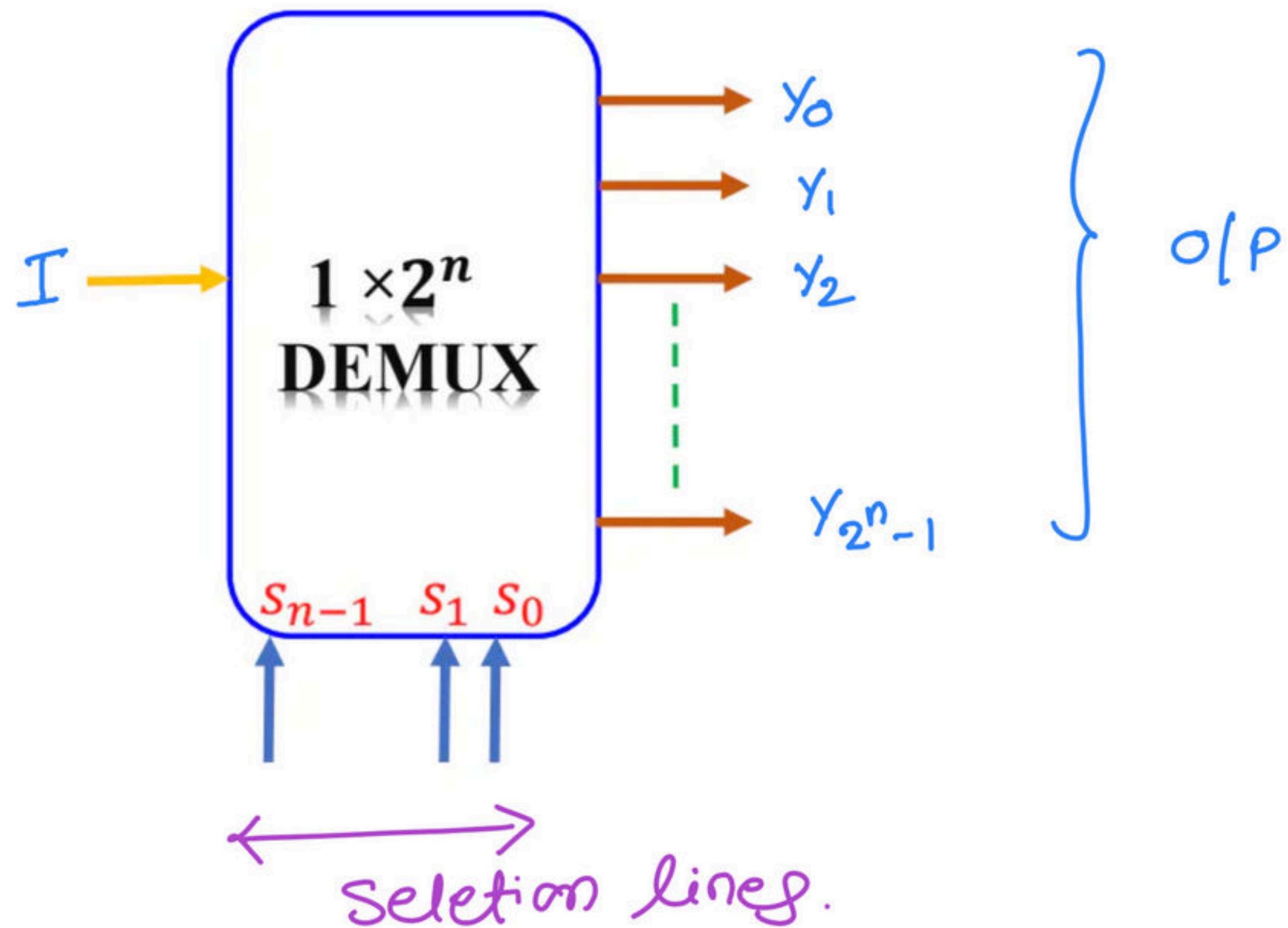
## General structure



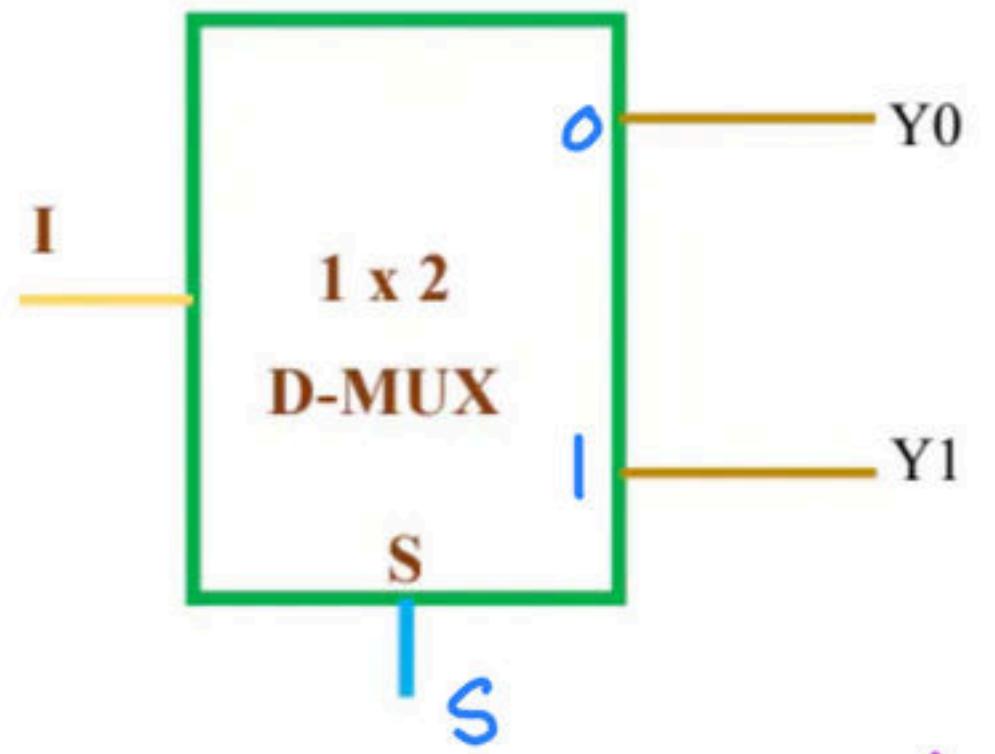
$n$  -----> number of select lines

$2^n$  -----> number of output lines

1 -----> number of inputs



# **$1 \times 2$ DEMUX**



$$y_0 = \bar{S} I$$

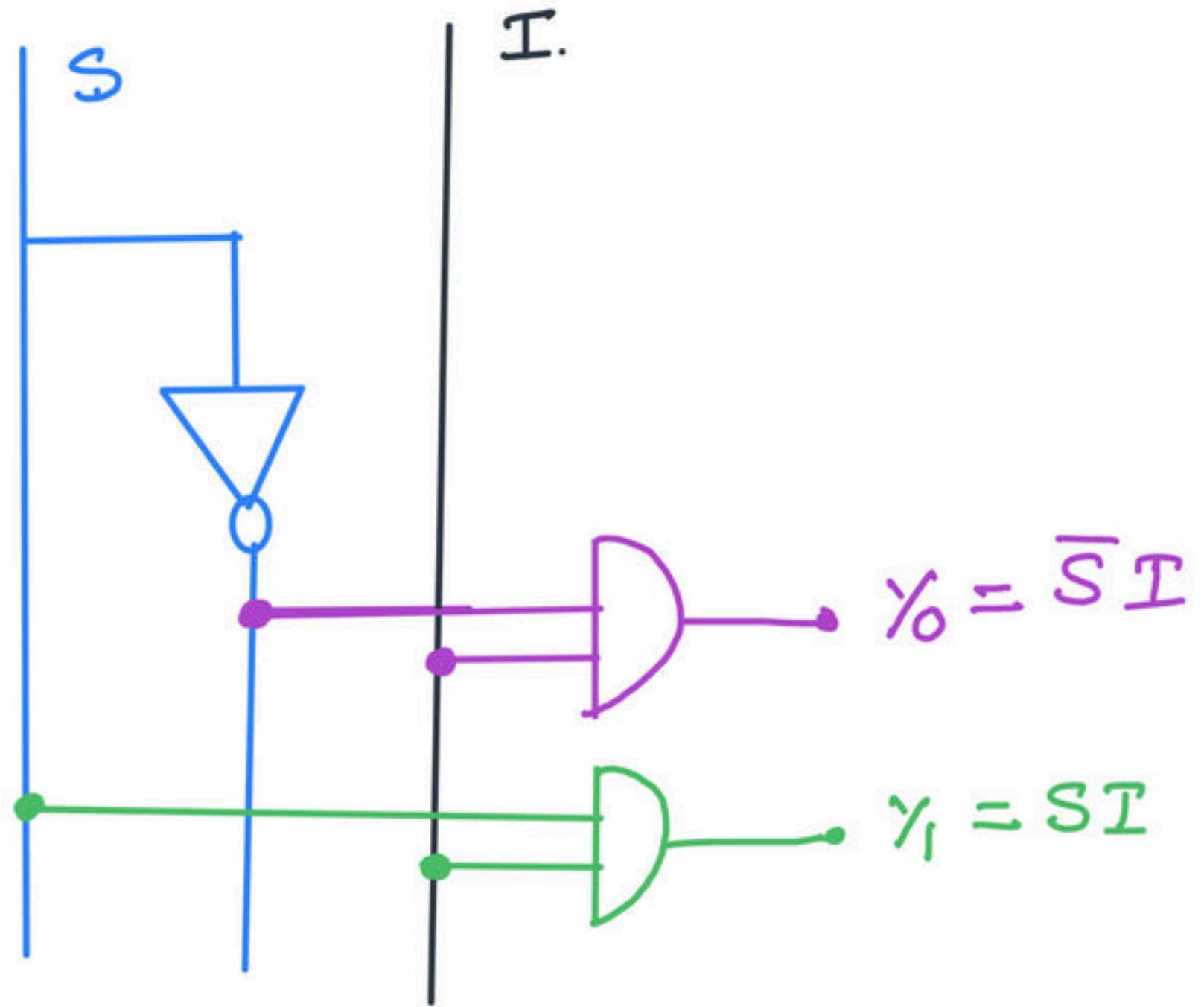
$$y_1 = S I$$

S	$Y_0$	$Y_1$
0	I	0
1	0	I

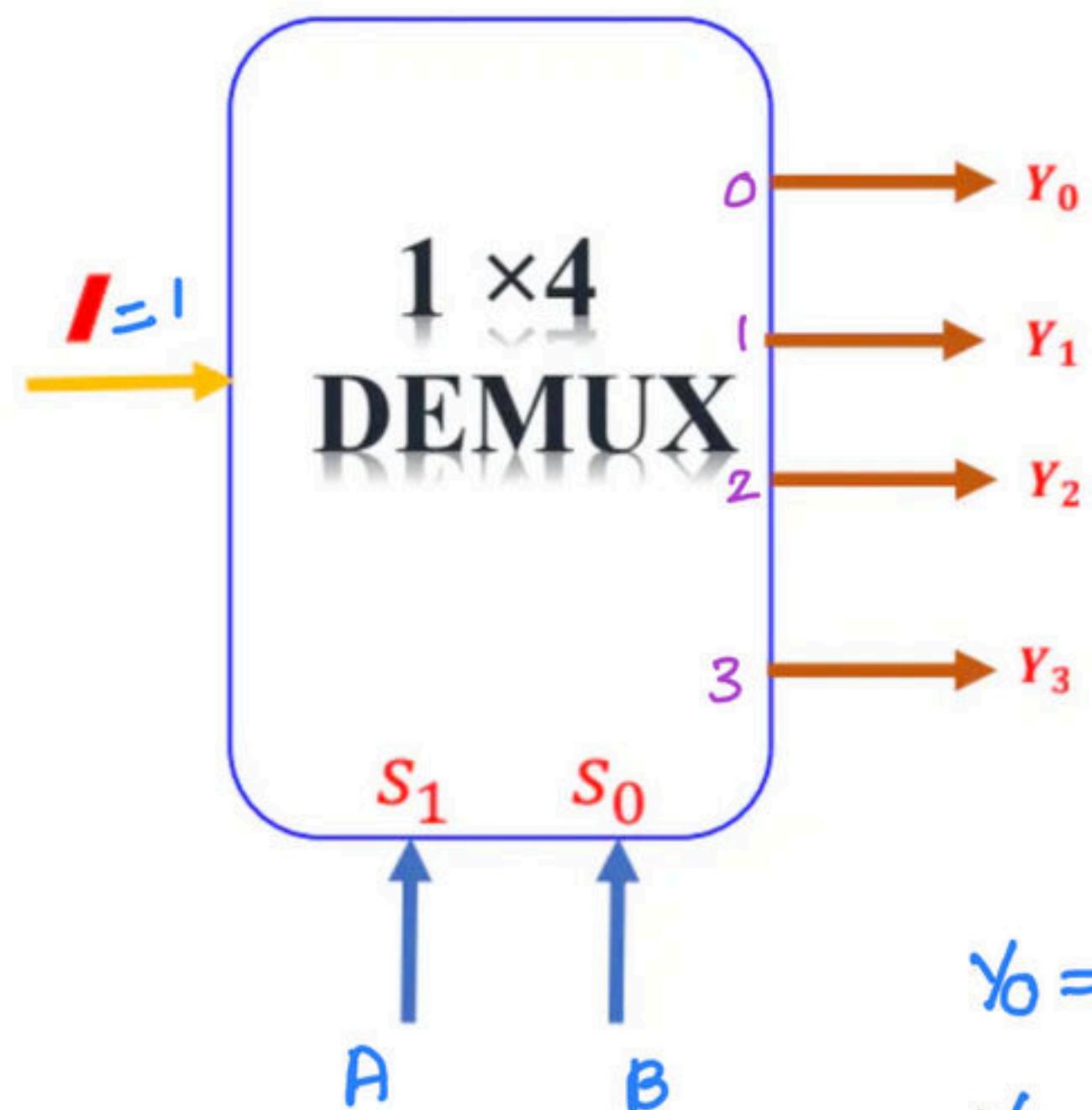
# Logic circuit

$$\gamma_0 = \overline{S} T$$

$$\gamma_1 = S T$$



# $1 \times 4$ Demultiplexer

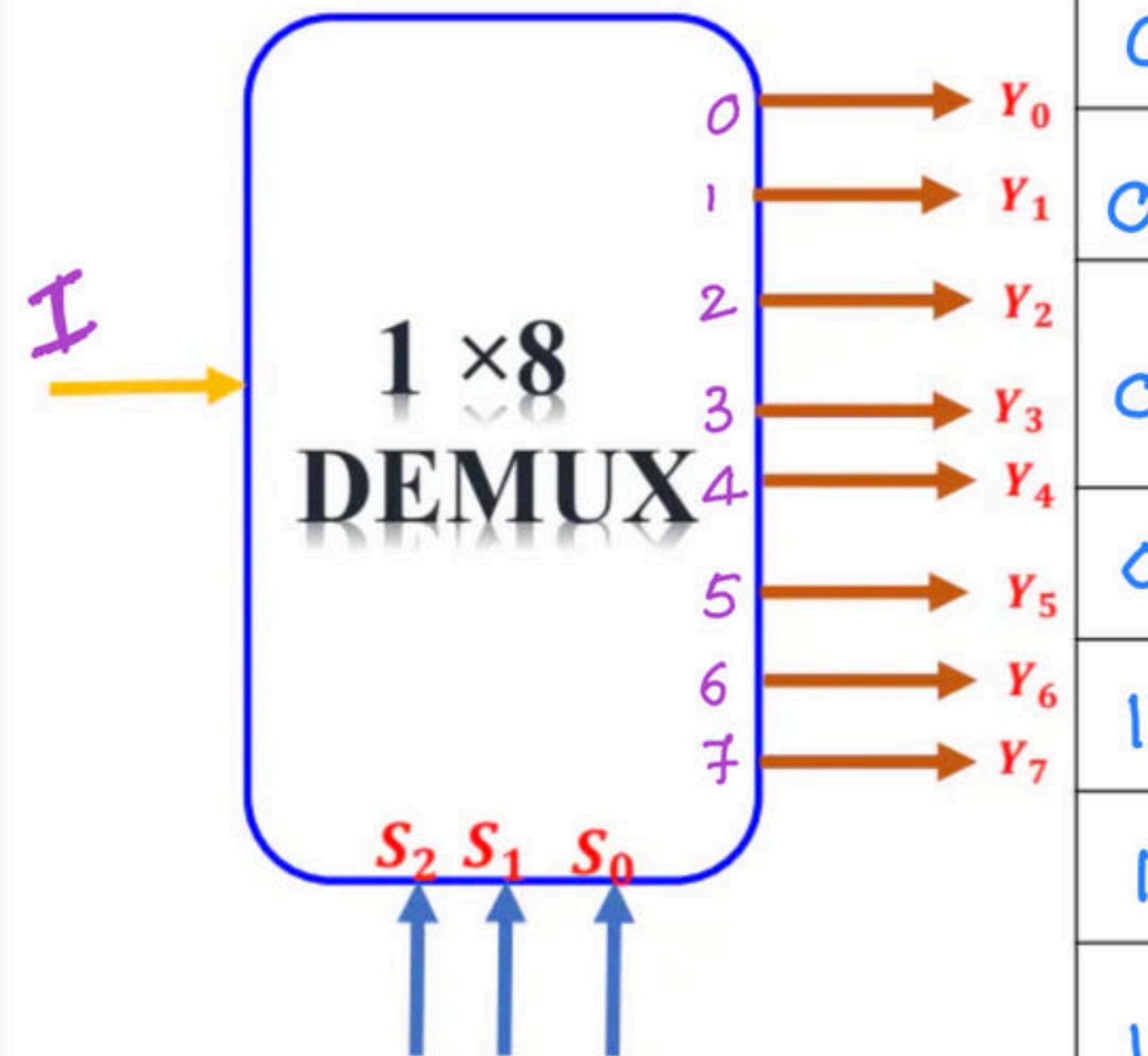


$S_1$	$S_0$	$Y_3$	$Y_2$	$Y_1$	$Y_0$
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

$$Y_0 = \bar{S}_1 \bar{S}_0 1$$

$$Y_0 = \bar{A} \bar{B} (1)$$

# $1 \times 8$ DEMUX

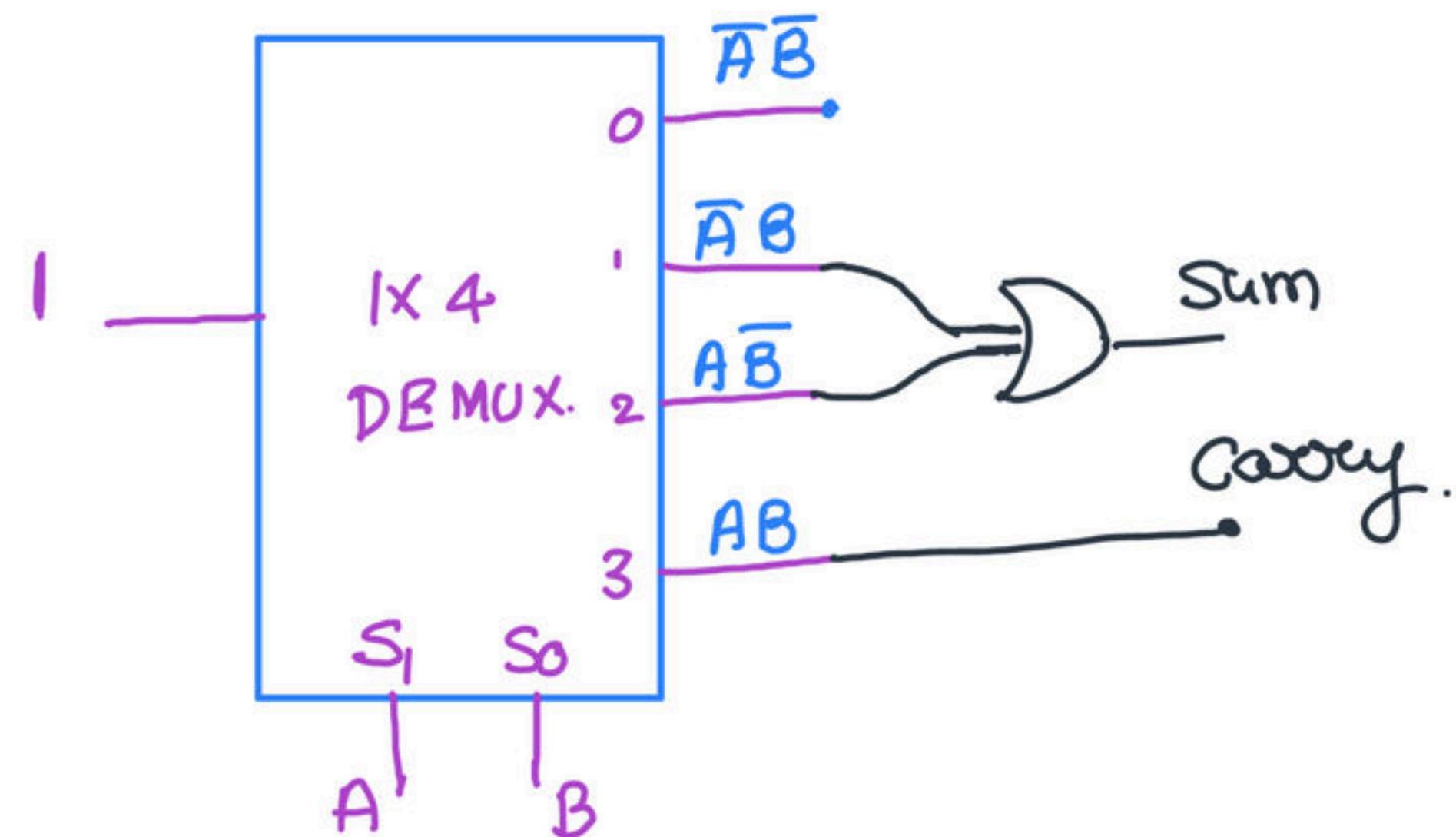


S2	S1	S0	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
0	0	0	0	0	0	0	0	0	0	$I$
0	0	1	0	0	0	0	0	0	$I$	0
0	1	0	0	0	0	0	0	$I$	0	0
0	1	1	0	0	0	0	$I$	0	0	0
1	0	0	0	0	0	$I$	0	0	0	0
1	0	1	0	0	$I$	0	0	0	0	0
1	1	0	0	$I$	0	0	0	0	0	0
1	1	1	$I$	0	0	0	0	0	0	0

Q) Implement HA using  $1 \times 4$  DEMUX

$$\text{Sum} = A \oplus B = \overline{A}B + A\overline{B}$$

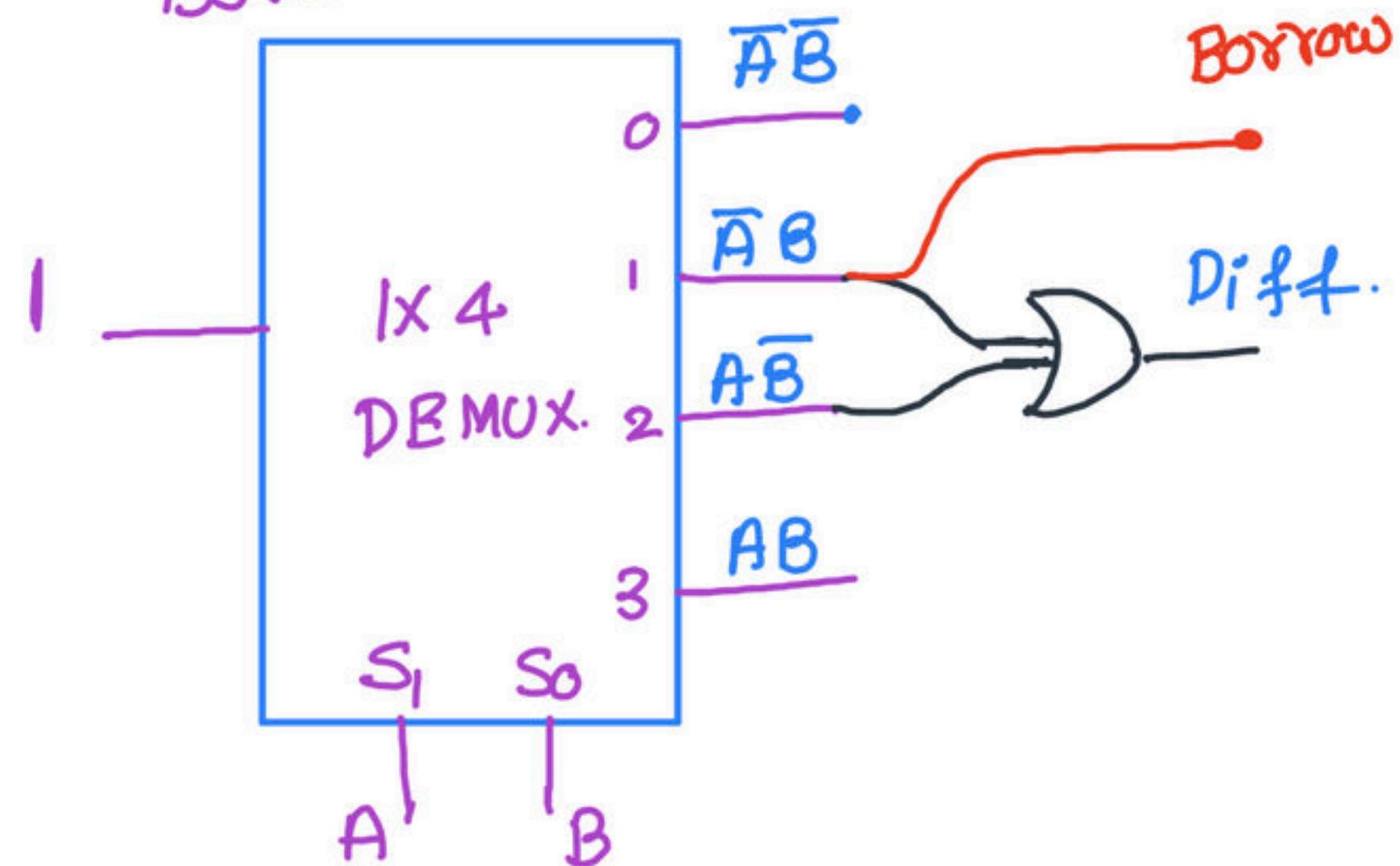
$$\text{Carry} = AB.$$



Q) Implement HS using  $1 \times 4$  DEMUX

$$\text{Diff.} = \overline{A}B + A\overline{B}$$

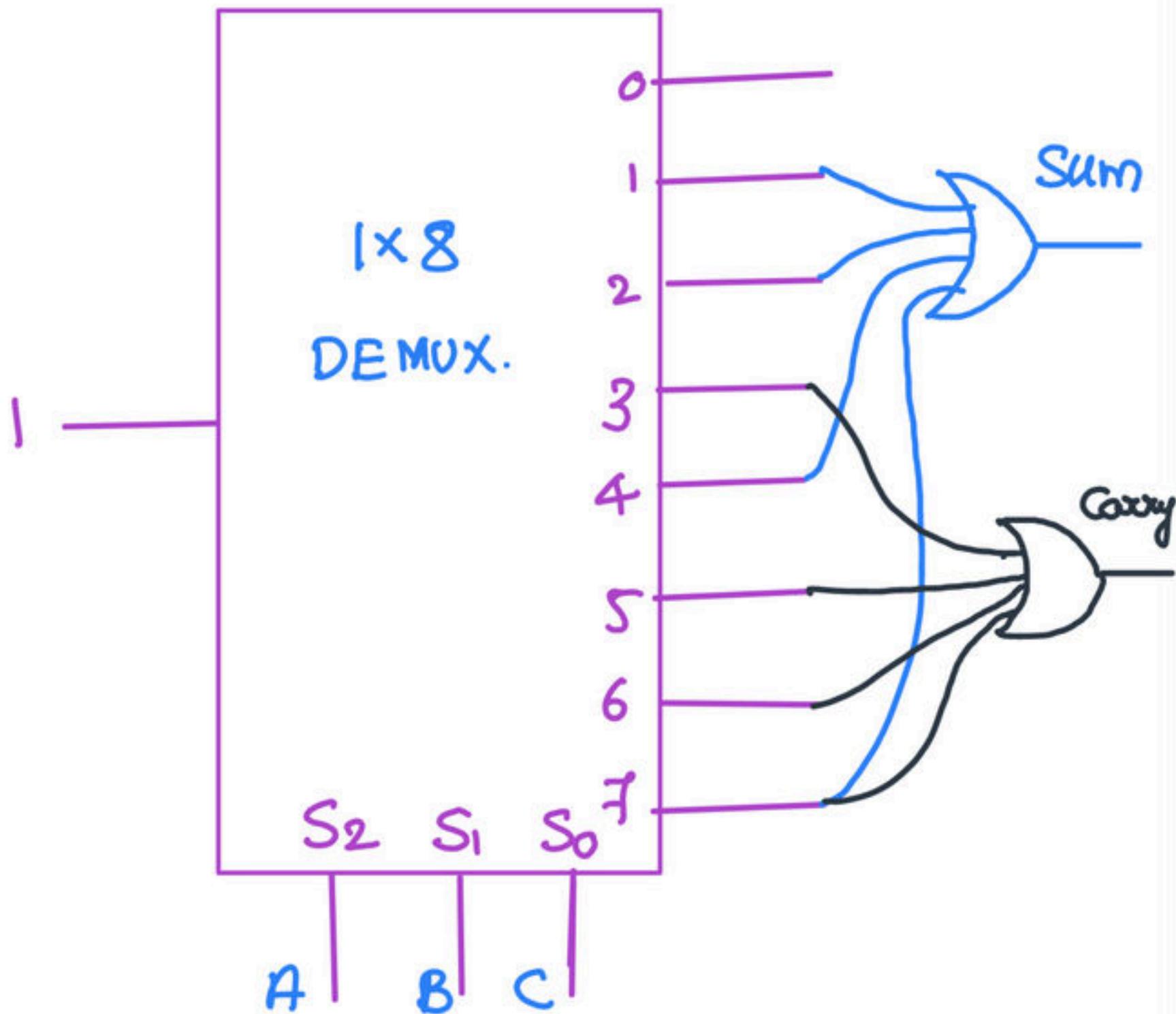
$$\text{Borrow} = \overline{A}B$$



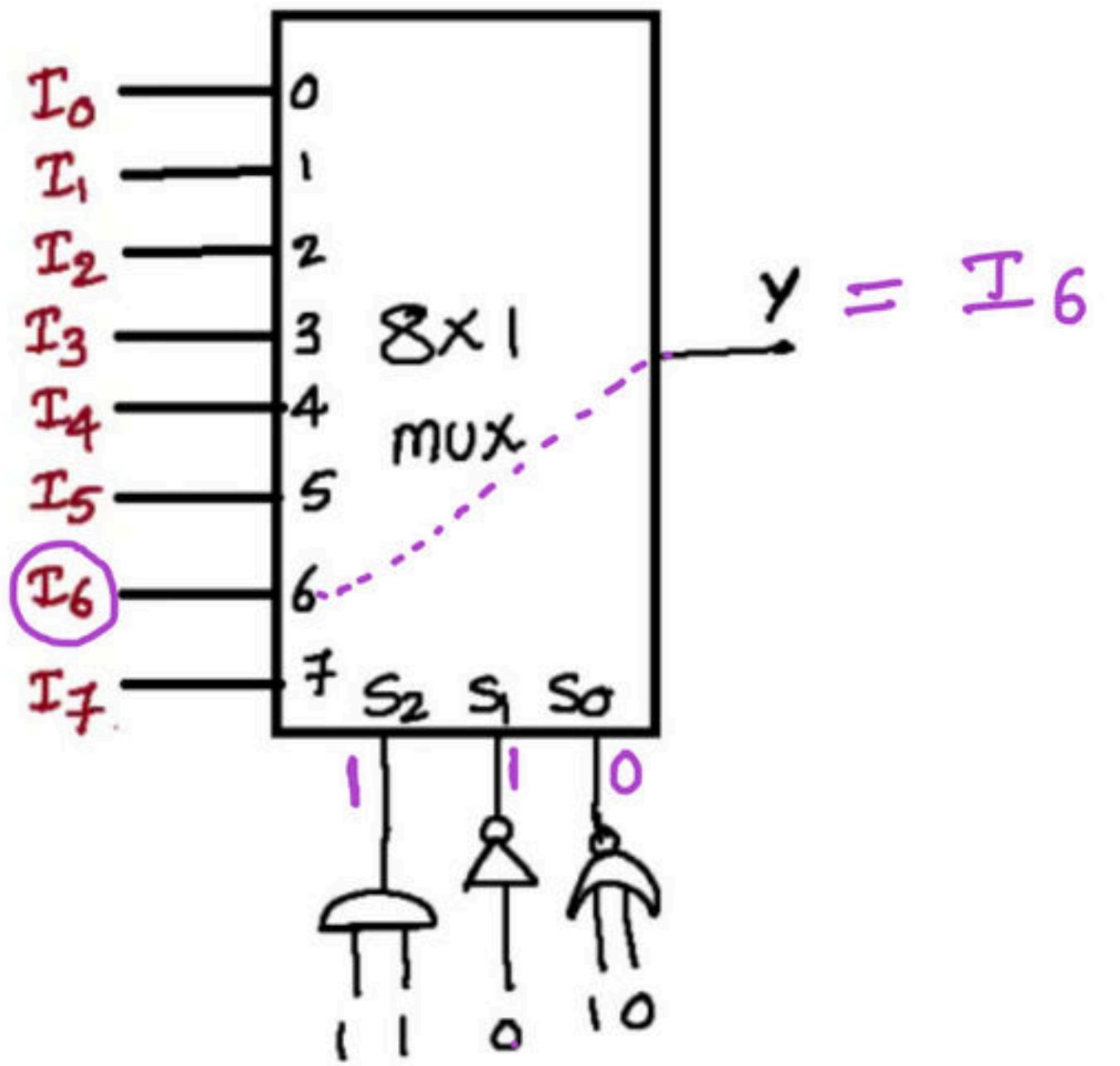
Q) Implement FA using  $1 \times 8$  DEMUX

$$\text{Sum} = \sum m(1, 2, 4, 7)$$

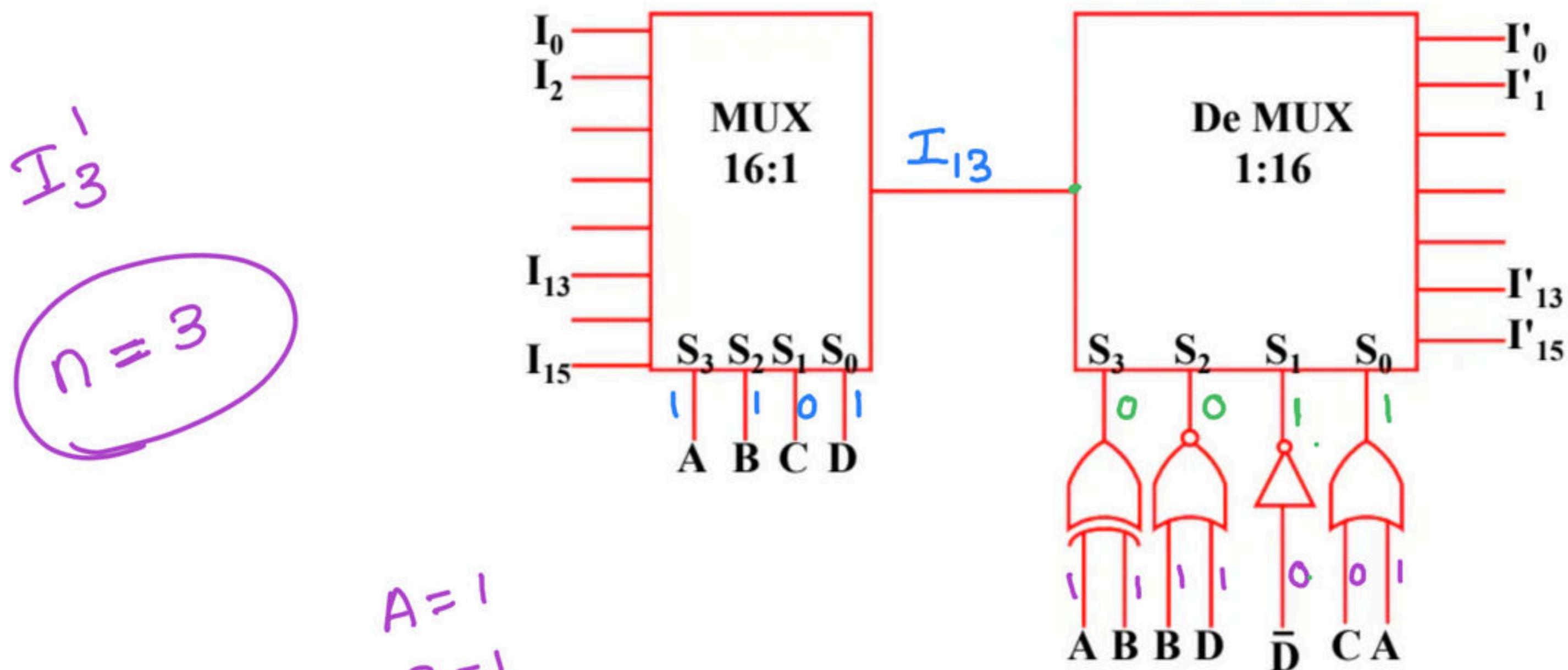
$$\text{Carry} = \sum m(3, 5, 6, 7)$$



Q) The output of the mux (Y) is



Q. Consider the logical circuit given below , Input at line  $I_{13}$  in  $16 \times 1$  MUX corresponds to output at line  $I_n$  of  $1 \times 16$  De-MUX. The value of 'n' is \_\_\_\_\_.



$$A = 1$$

$$B = 1$$

$$C = 0$$

$$D = 1$$

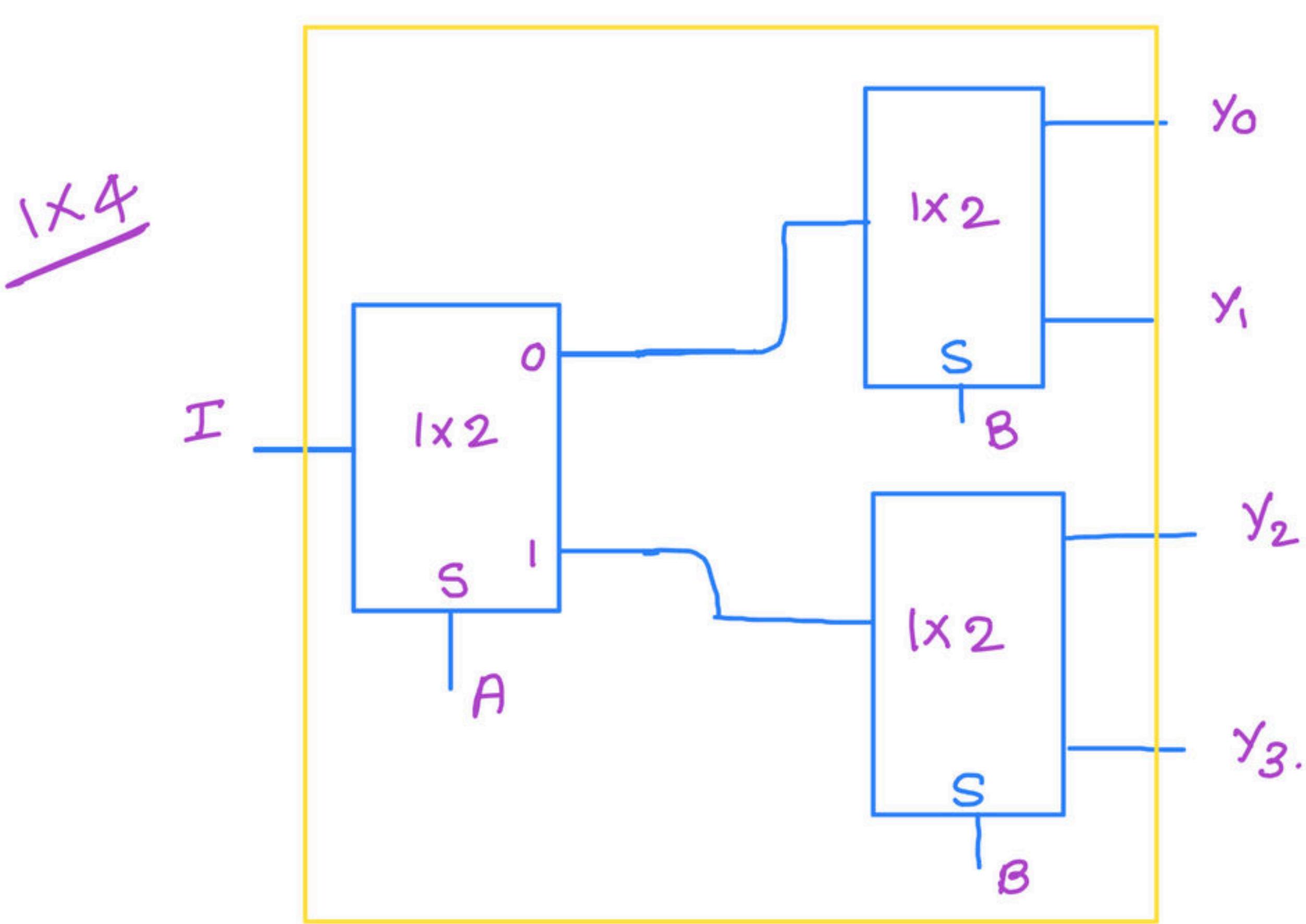
Implementation of higher order Demux using lower order  
Demux

Q) Implement  $1 \times 4$  Demux using  $1 \times 2$  Demux

$$\begin{array}{c} \frac{4}{2} \\ \hline \end{array} = 2 \rightarrow L_2 \quad (B)$$
$$\begin{array}{c} \frac{2}{2} \\ \hline \end{array} = 1 \rightarrow L_1 \quad (\text{MSB}) \quad (A)$$

3.

$f(A, B)$



Q) Implement  $1 \times 16$  Demux using  $1 \times 2$  Demux

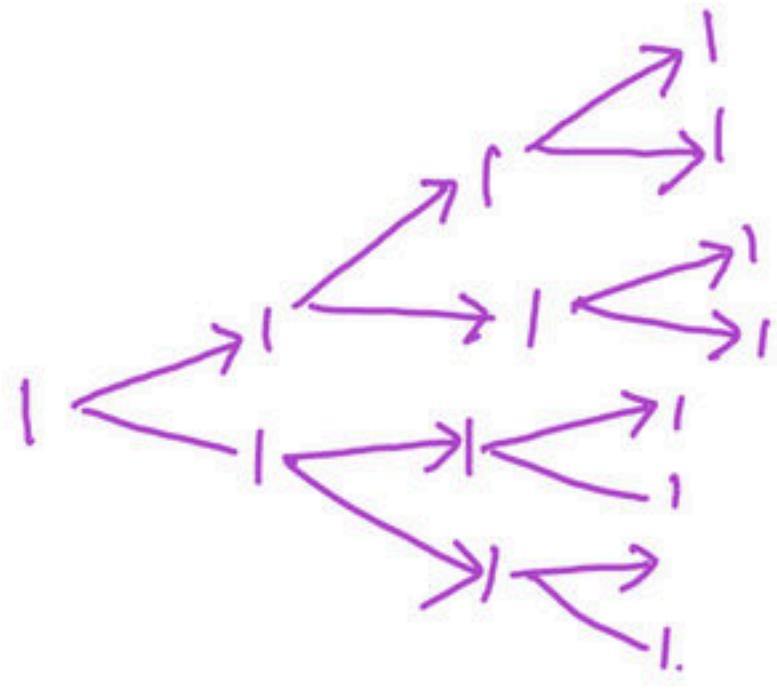
$$\frac{16}{2} = 8 \rightarrow b_4(D)$$

$$\frac{8}{2} = 4 \rightarrow b_3(C)$$

$$\frac{4}{2} = 2 \rightarrow b_2(B)$$

$$\frac{2}{2} = 1 \rightarrow b_1(A)$$

15



A B C D

Q) Implement  $1 \times 8$  Demux using  $1 \times 4$  Demux

$$\frac{8}{4} = 2 \rightarrow h_2, (B, C)$$

$$\frac{2}{4} = 1 \quad (1 \times 2 \text{ Demux}) \rightarrow h_1 \underline{(A)}$$

$f(A_1, B, C)$

# Decoder

Decoder is a multi input ,multi output logic circuit which converts coded input into coded output , where the input and output codes are different .

General structure

$n \times 2^n$

$1 \times 2$

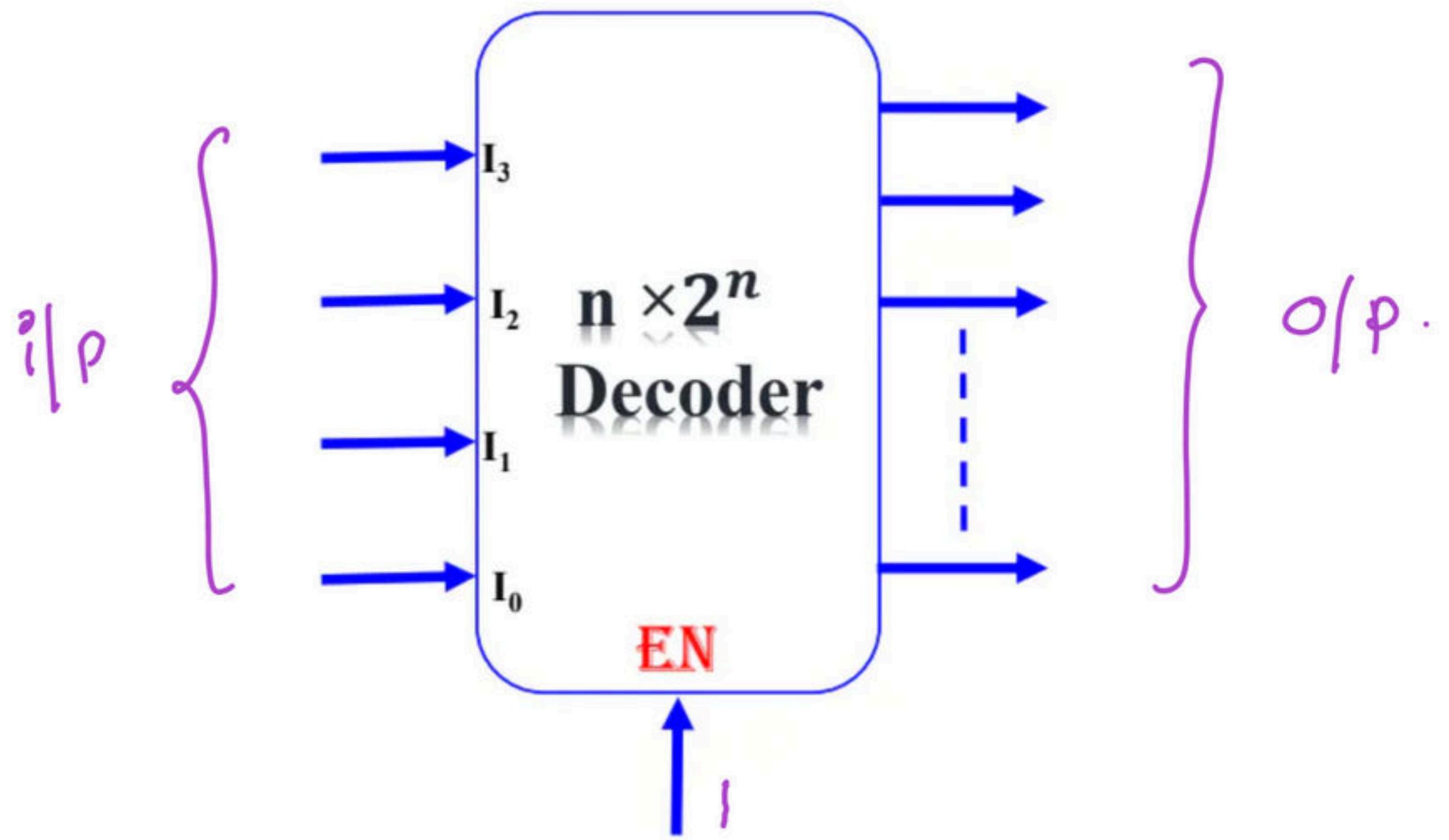
$2 \times 4$

$3 \times 8$

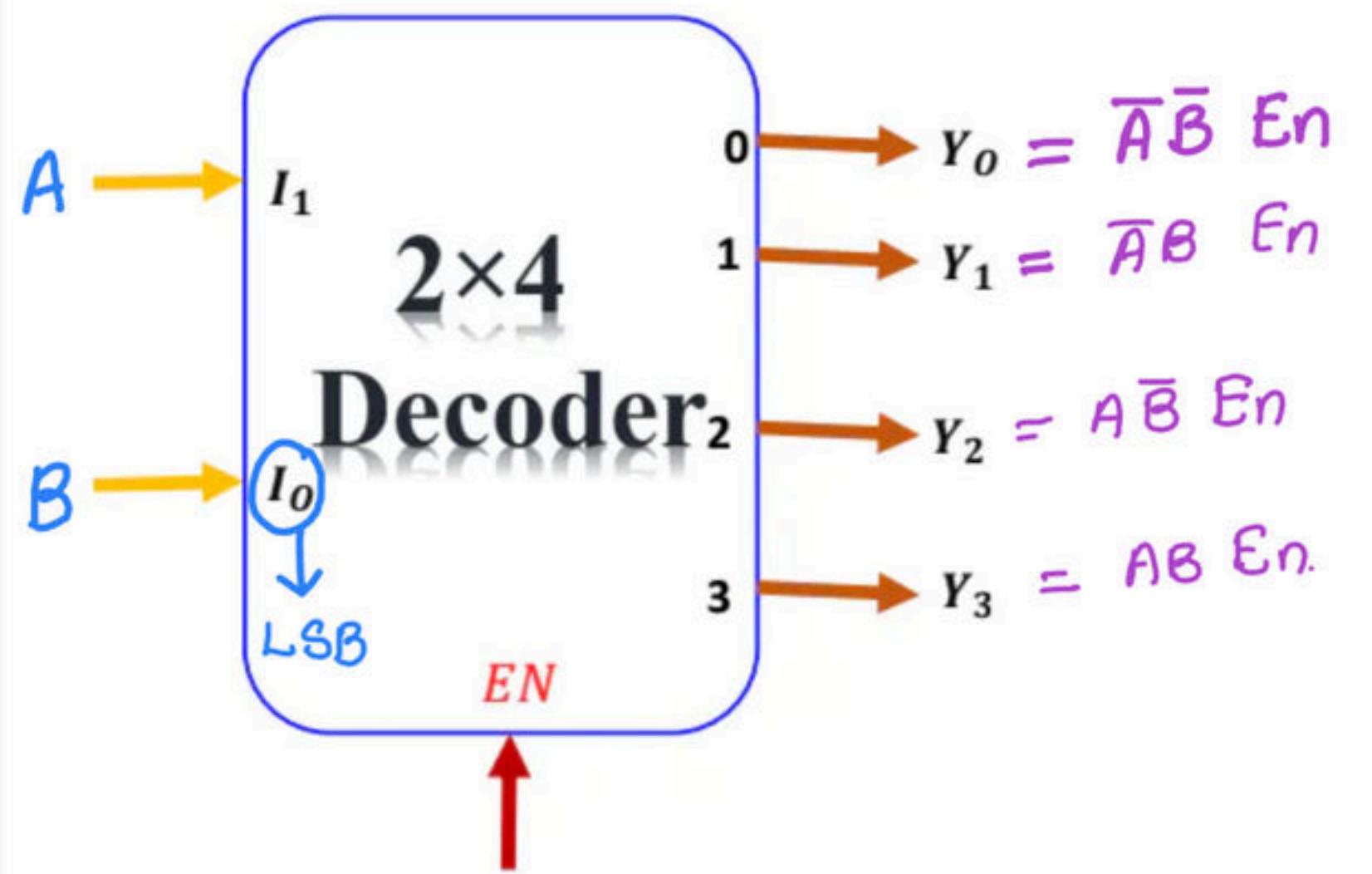
$4 \times 16$

$n$  -----> number of inputs

$2^n$  -----> number of outputs

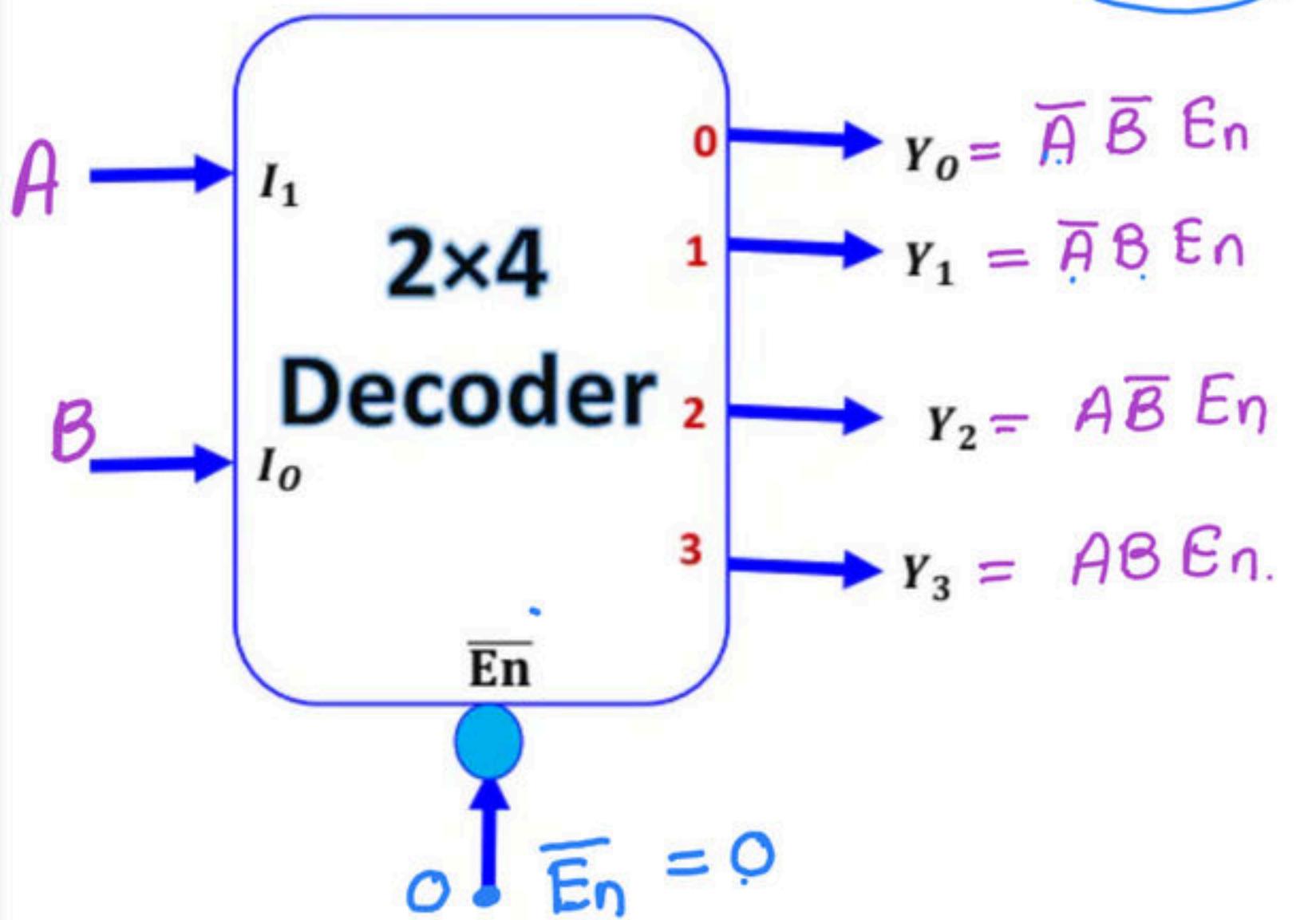


# Active High Decoder



<b>En</b>	<b>A</b>	<b>B</b>	<b>Y<sub>3</sub></b>	<b>Y<sub>2</sub></b>	<b>Y<sub>1</sub></b>	<b>Y<sub>0</sub></b>
0	X	X	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0

# Active High Decoder



$$Y_0 = 0$$

$$Y_1 = 0$$

$$Y_2 = 1$$

$$Y_3 = 0$$

$$\bar{E}_n = 1$$

$$E_n = 0$$

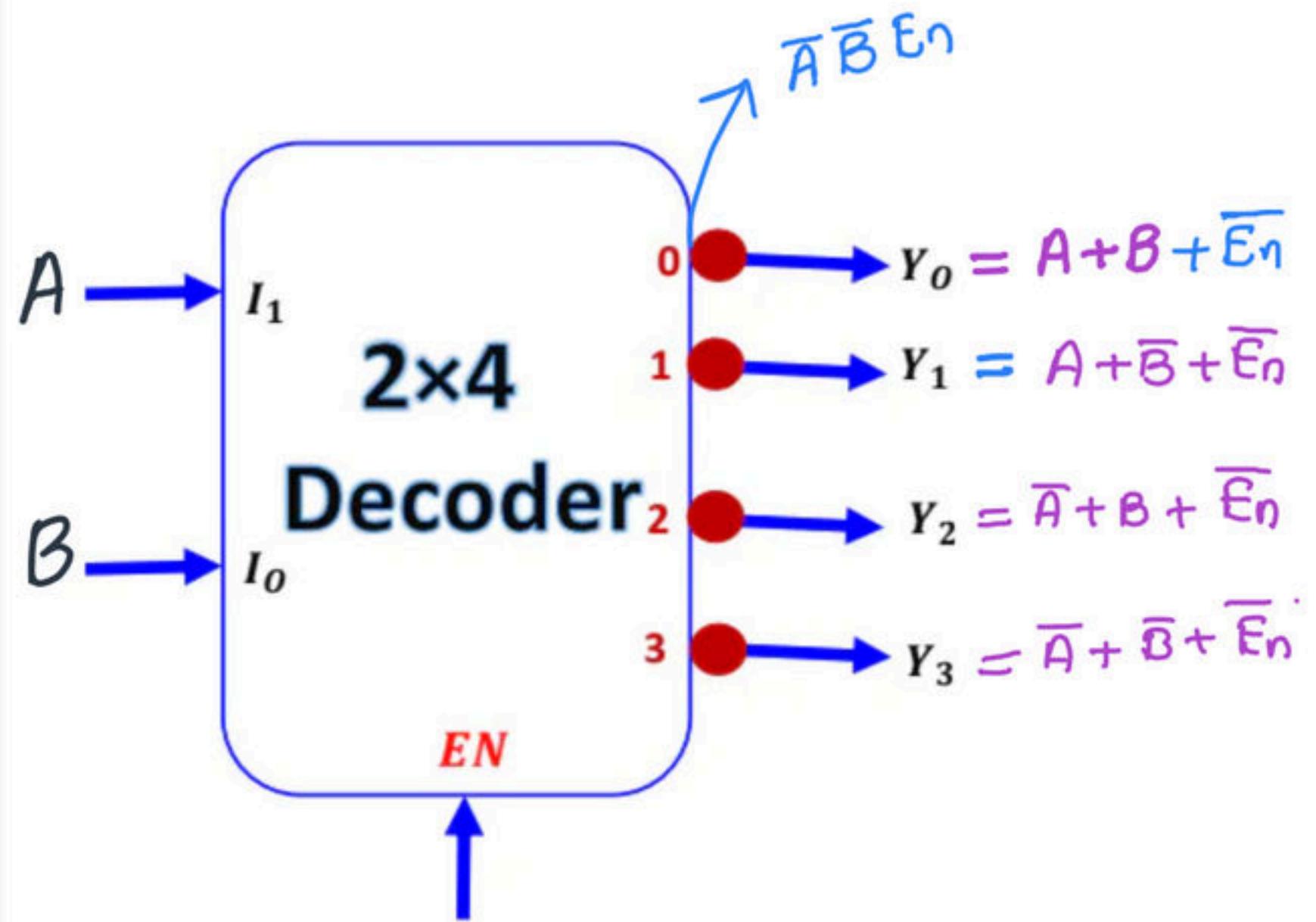
$$\bar{E}_n = 0 \quad | \quad E_n = 1$$

$$A = 1 \quad B = 0$$

$\bar{E}_n$	A	B	$Y_3$	$Y_2$	$Y_1$	$Y_0$
1	X	X	0	0	0	0
0	0	0	0	0	0	1
0	0	1	0	0	1	0
0	1	0	0	1	0	0
0	1	1	1	0	0	0

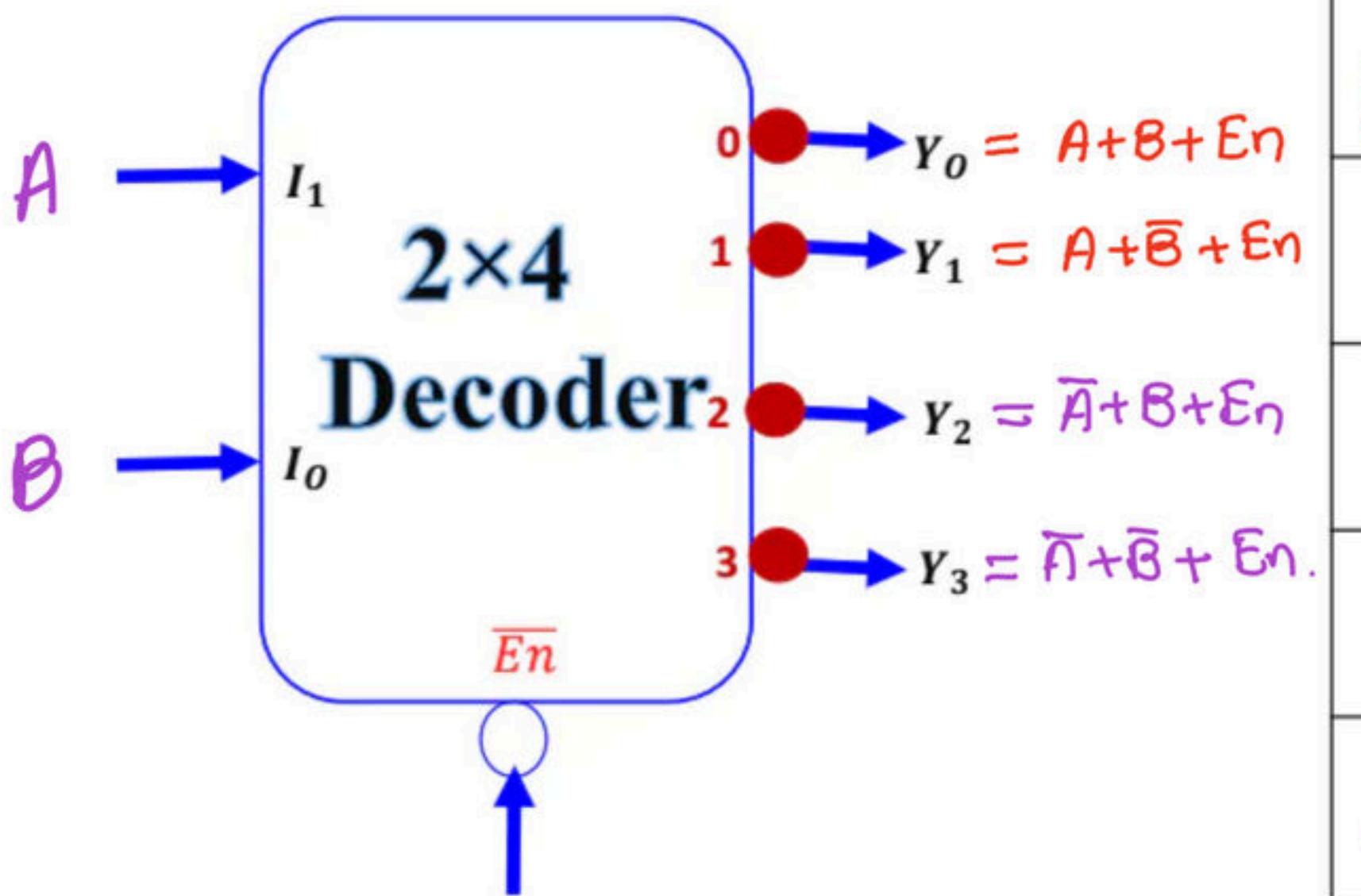
# Active Low Decoder

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



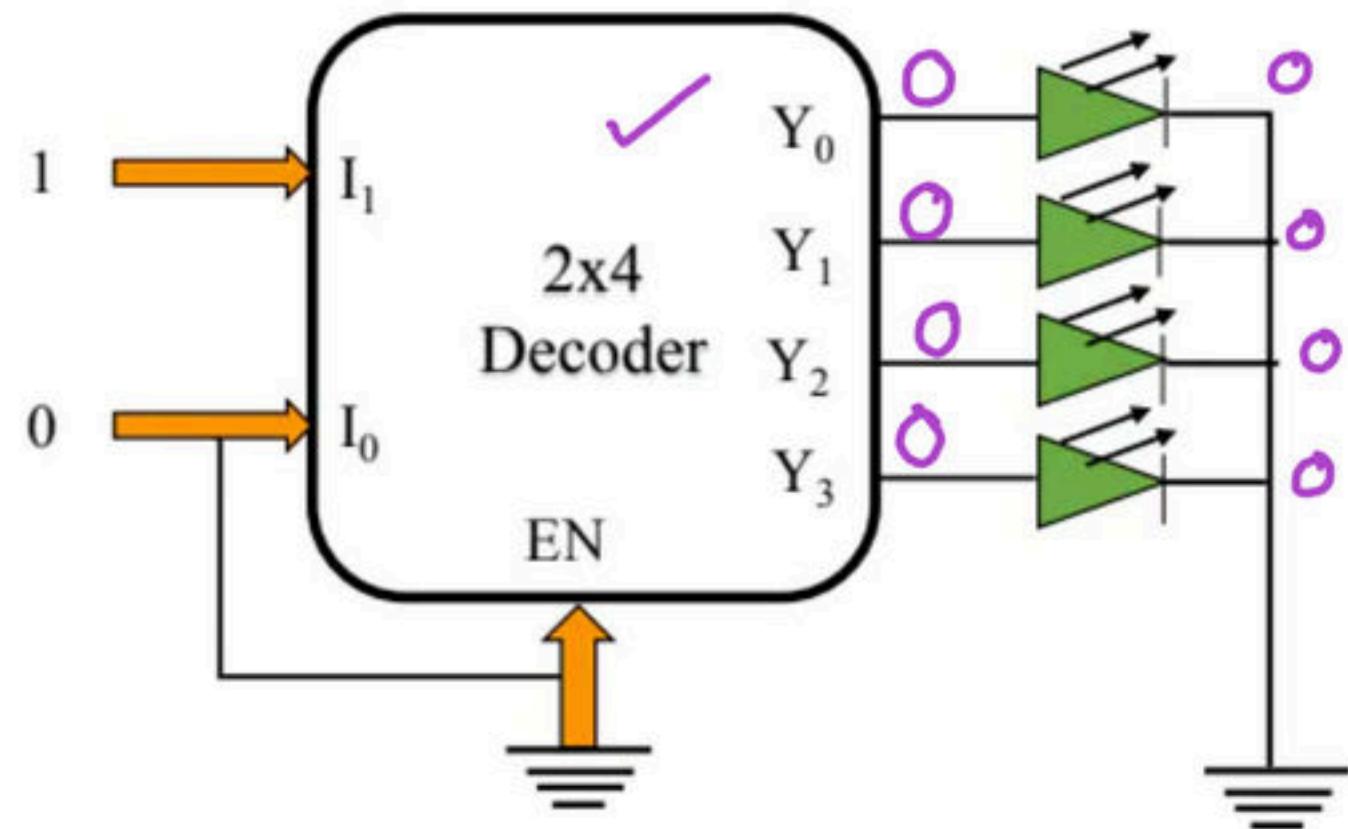
$EN$	$A$	$B$	$Y_3$	$Y_2$	$Y_1$	$Y_0$
0	X	X	1	1	1	1
1	0	0	1	1	1	0
1	0	1	1	1	0	1
1	1	0	1	0	1	1
1	1	1	0	1	1	1

# Active Low Decoder



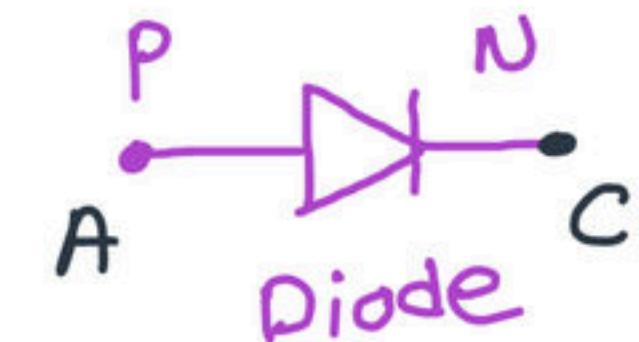
En	A	B	Y <sub>3</sub>	Y <sub>2</sub>	Y <sub>1</sub>	Y <sub>0</sub>
1	X	X	1	1	1	1
0	0	0	1	1	1	0
0	0	1	1	1	0	1
0	1	0	1	0	1	1
0	1	1	0	1	1	1

Q) Which of the following LED will glow



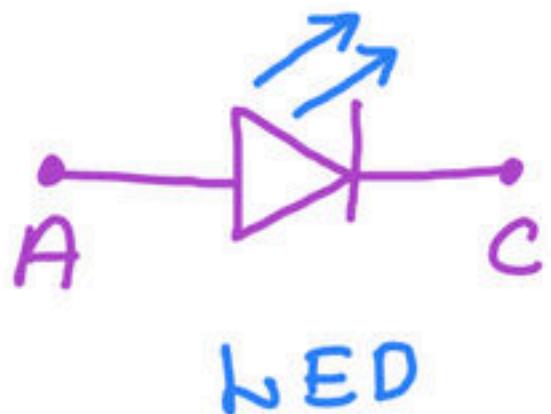
$$E_n = 0$$

NO LED glowing



$$V_A > V_C \rightarrow FB$$

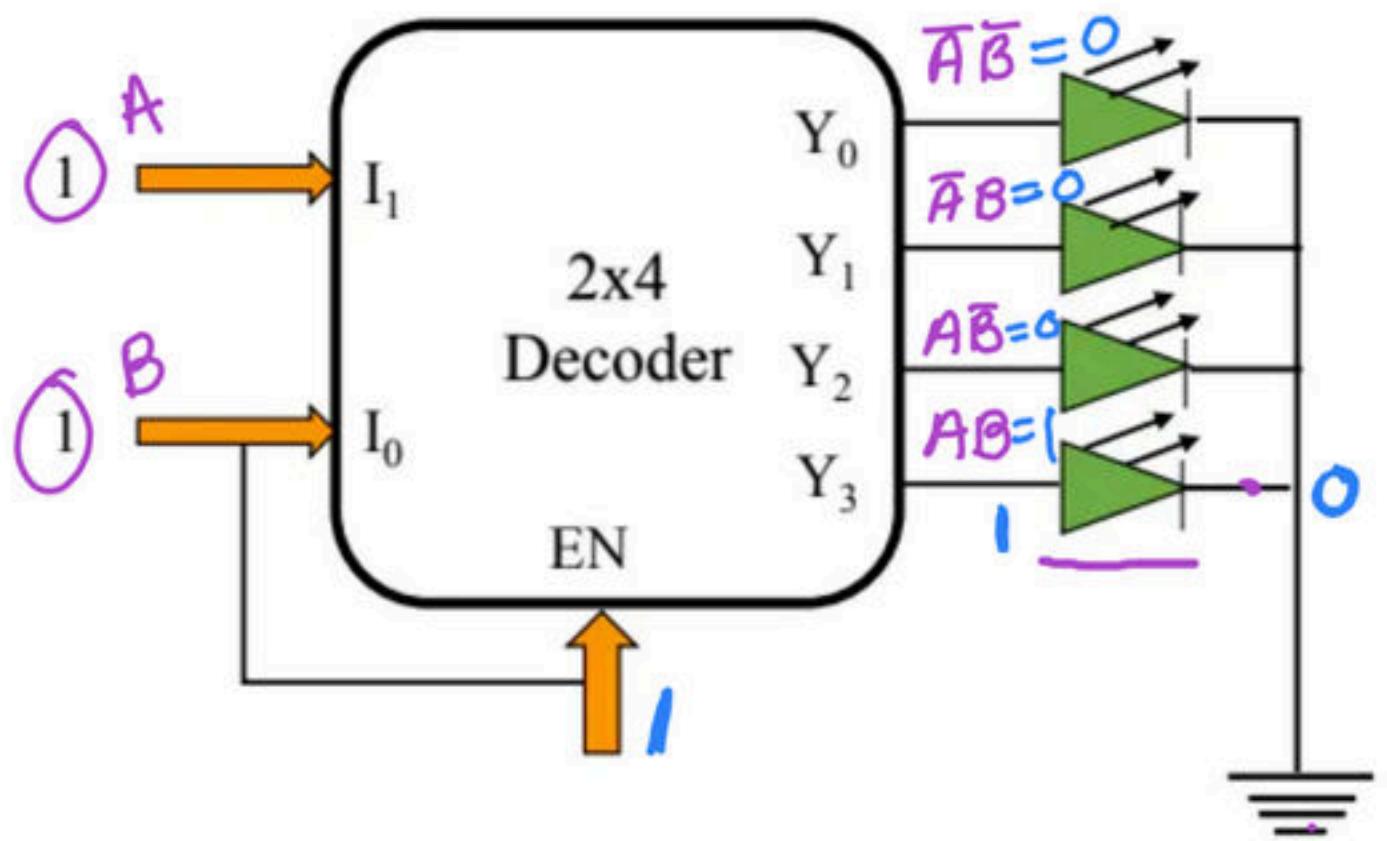
$$V_A < V_C \rightarrow RB$$



$$V_A > V_C \rightarrow LED \text{ on}$$

$$V_A \leq V_C \rightarrow LED \text{ off}$$

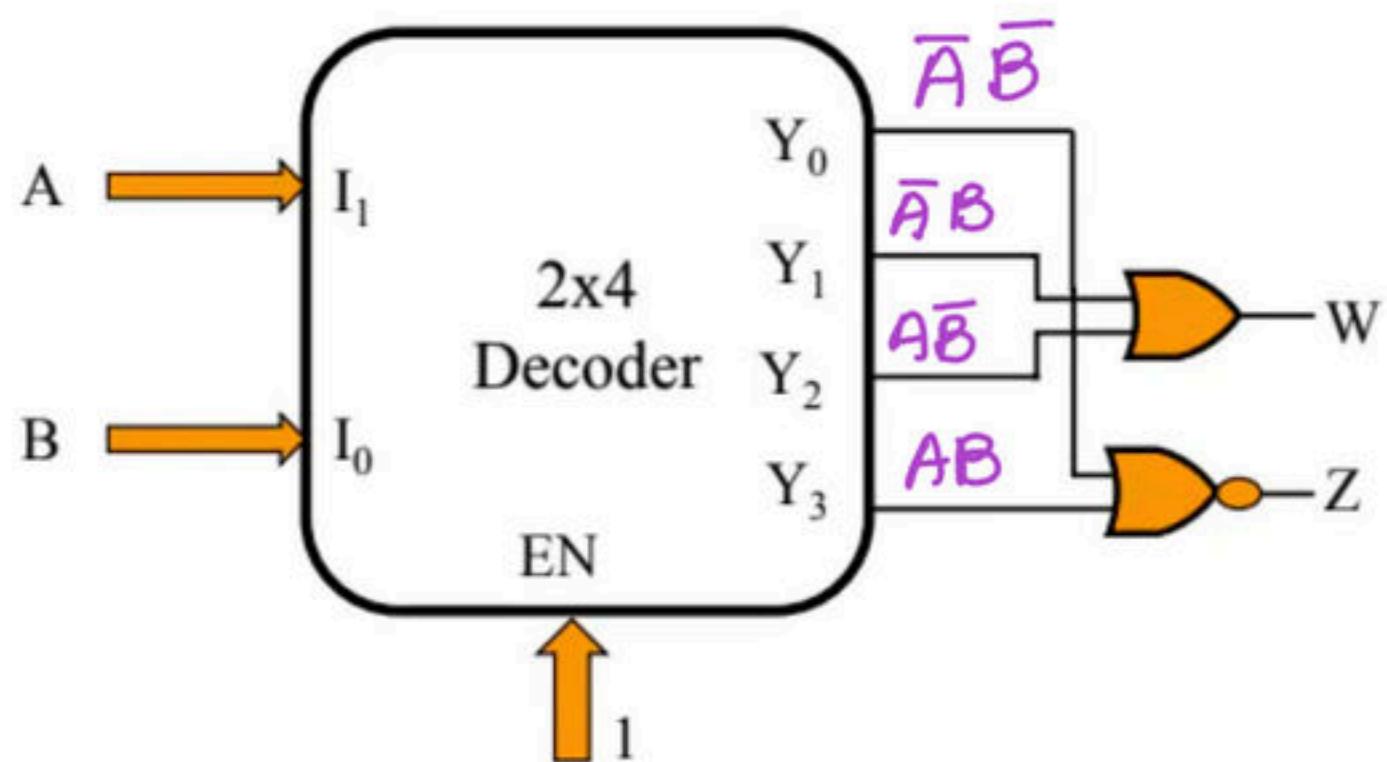
Q) Which of the following LED will glows



$$\begin{array}{l} AB \\ \downarrow \\ 11 \end{array}$$

$y_3 \rightarrow$  LED on  
glows.

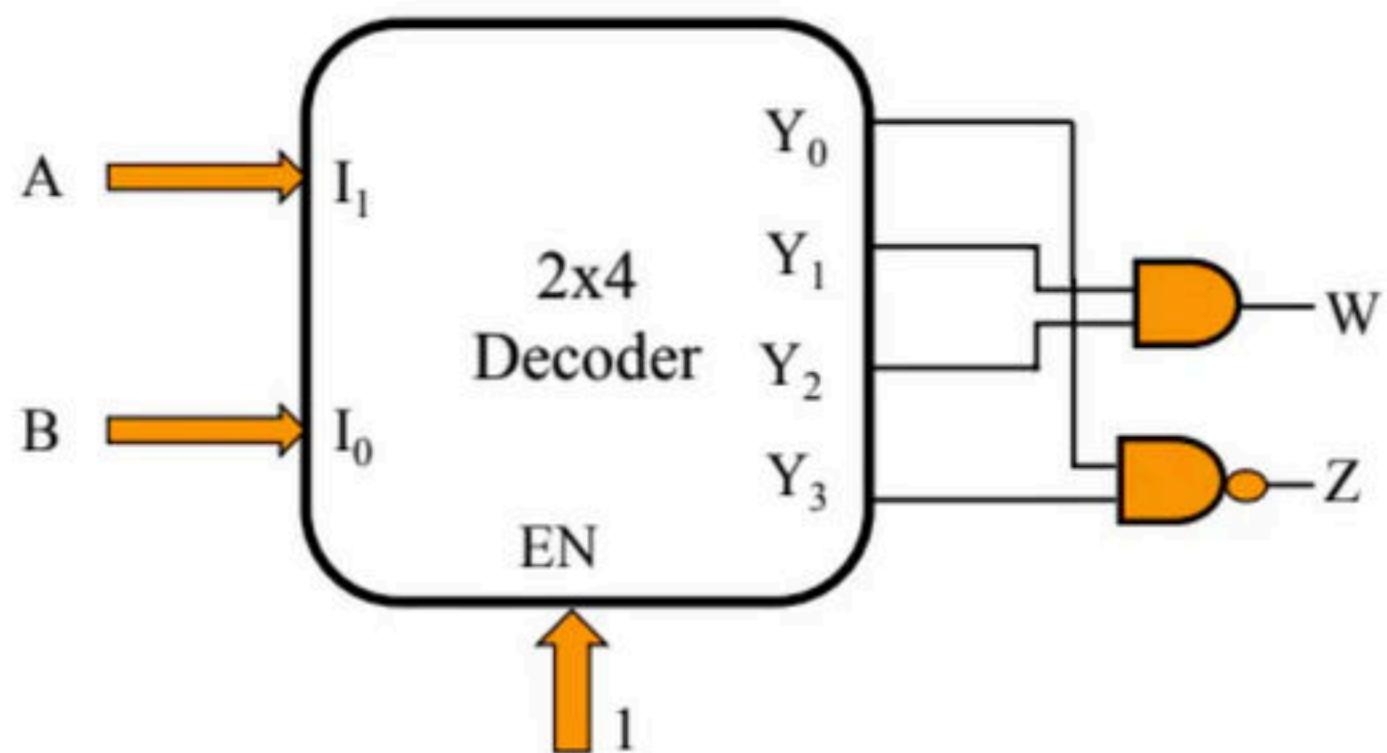
Q) Find the logic expression of W and Z



$$W = \bar{A}\bar{B} + A\bar{B} = A \oplus B$$

$$Z = \overline{\bar{A}\bar{B} + AB} = A \oplus B.$$

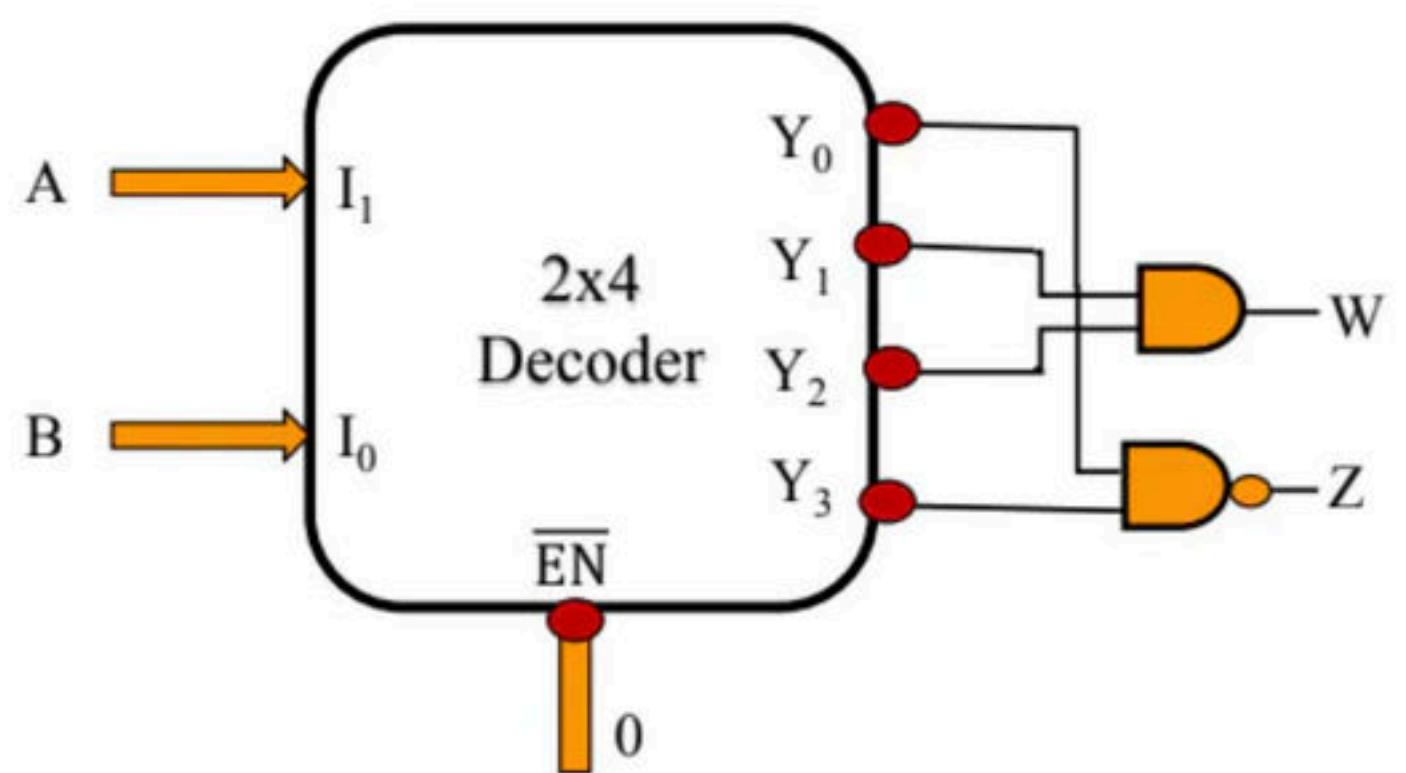
Q) Find the logic expression of W and Z



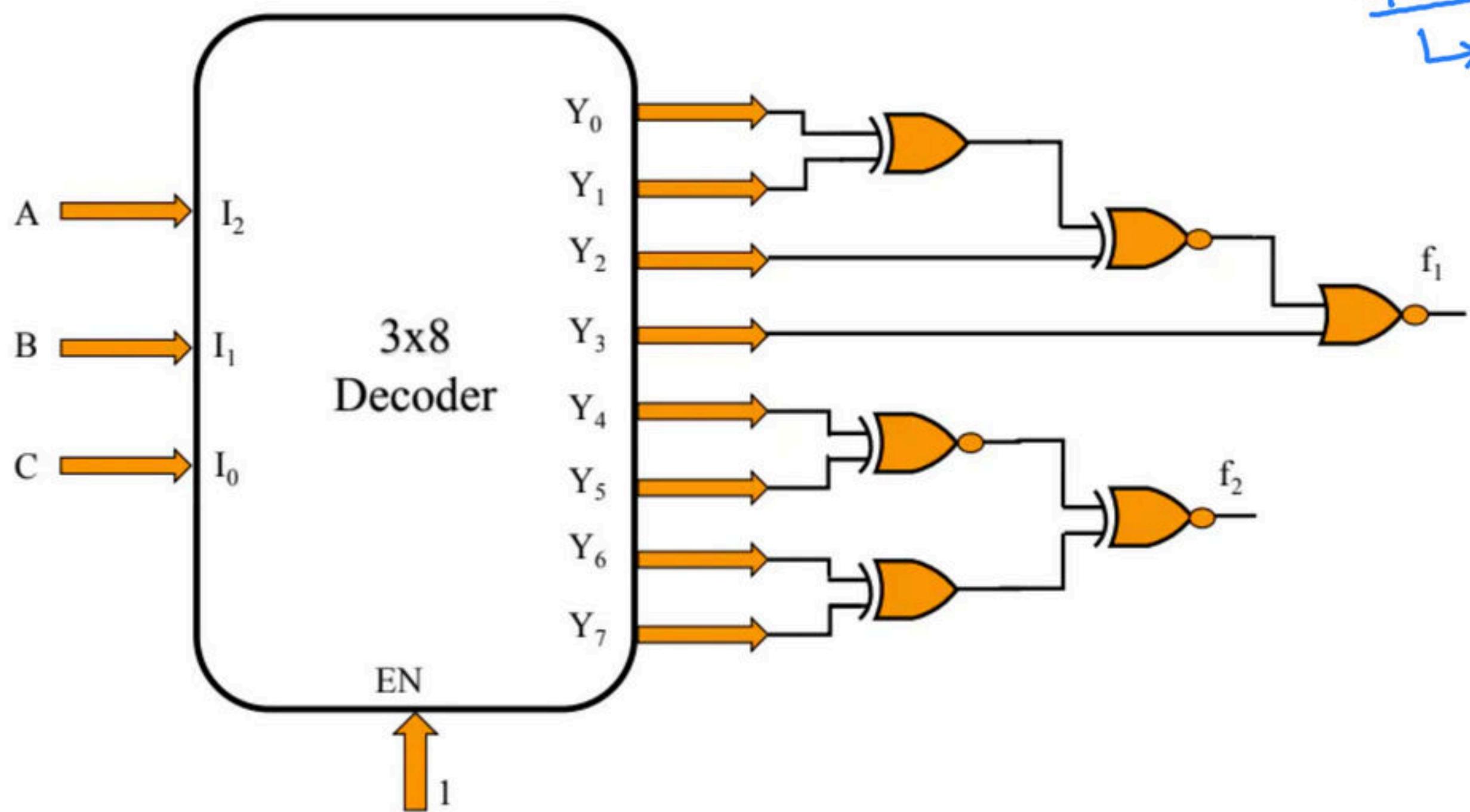
$$W = (\bar{A}B)(A\bar{B}) = 0$$

$$Z = \overline{\bar{A}\bar{B} \cdot AB} = 1.$$

Q) Find the logic expression of W and Z



### Q) The logic expression of F1 and F2



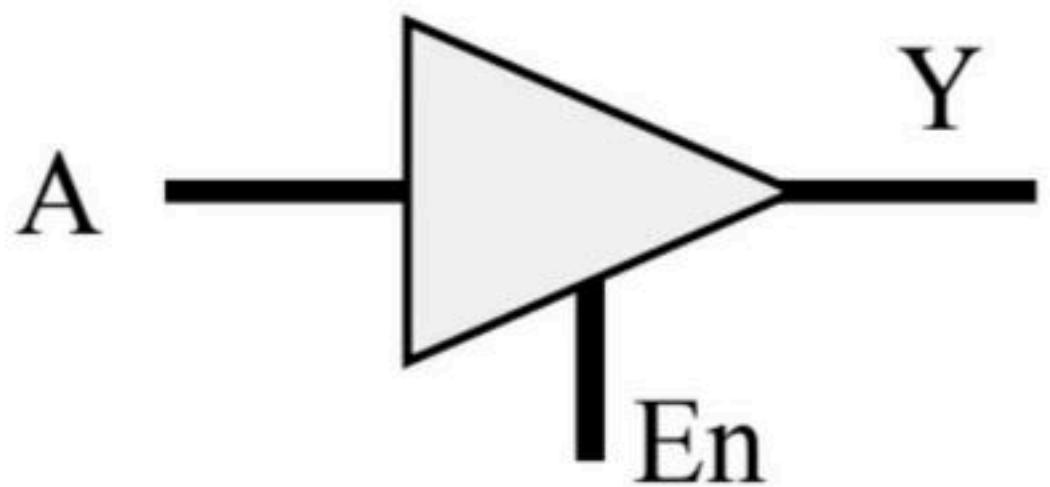
Q am  
Digital Revision  
Special classy.

Q) Implement HA using  $2 \times 4$  decoder

Q) Implement HS using  $2 \times 4$  decoder

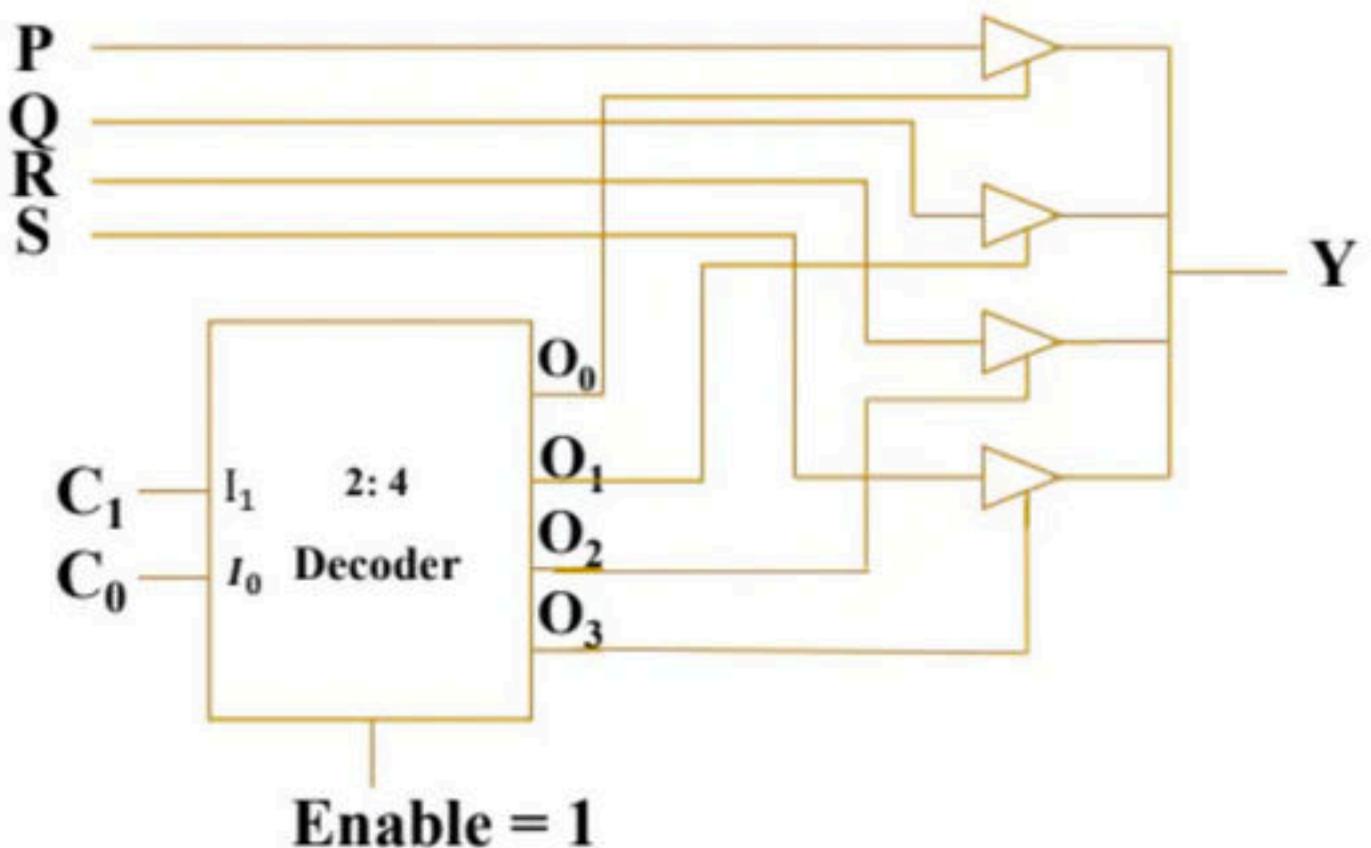
Q ) implement  $F(A, B, C) = A + BC$  , using decoder

## Tri-state Buffer



**Q.** The functionality implemented by the circuit below is.

- (a) 2-to-1 multiplexer
- (b) 4-to-1 multiplexer
- (c) 7-to-1 multiplexer
- (d) 6-to-1 multiplexer



Tristate buffer

**Q.** A logic circuit consists of two  $2 \times 4$  decoder as shown below, The output of decoder are given below

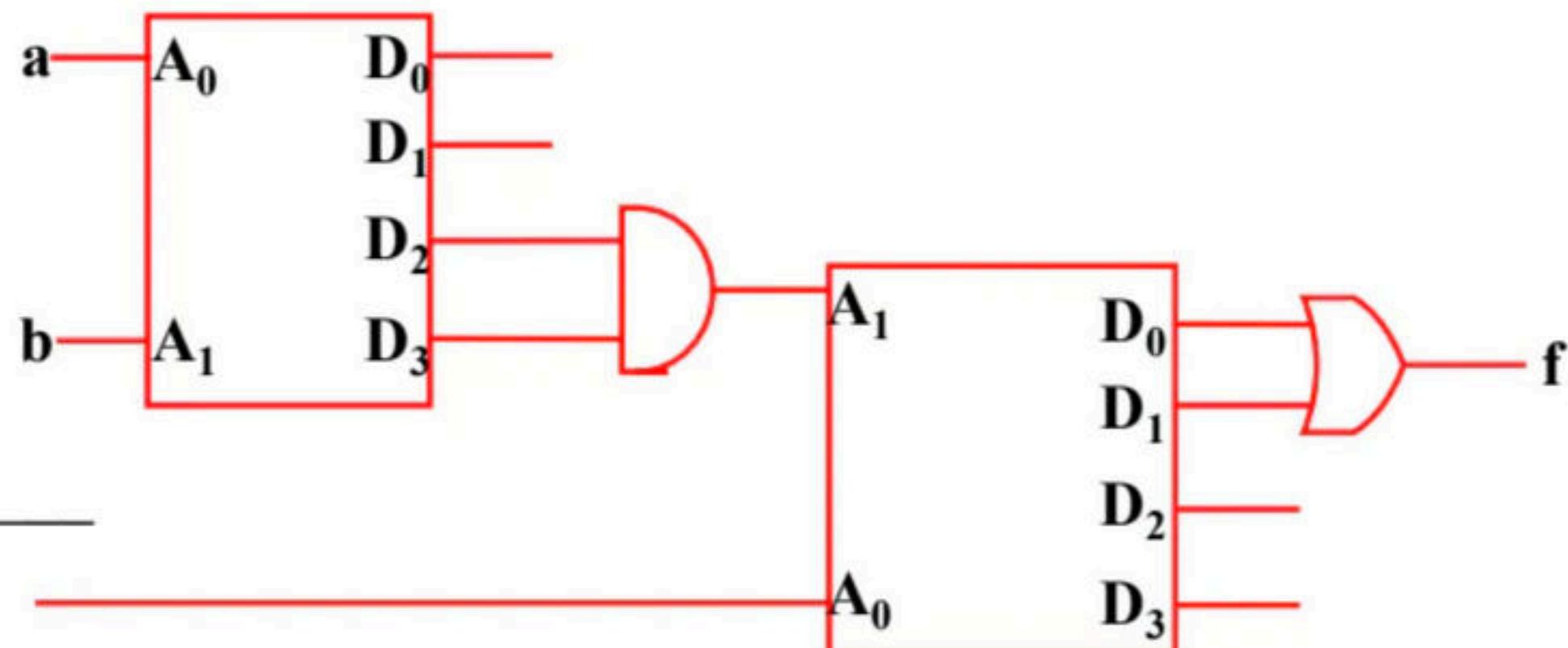
$$D_0 = 1 \text{ when } A_0 = 0, A_1 = 0$$

$$D_1 = 1 \text{ when } A_0 = 1, A_1 = 0$$

$$D_2 = 1 \text{ when } A_0 = 0, A_1 = 1$$

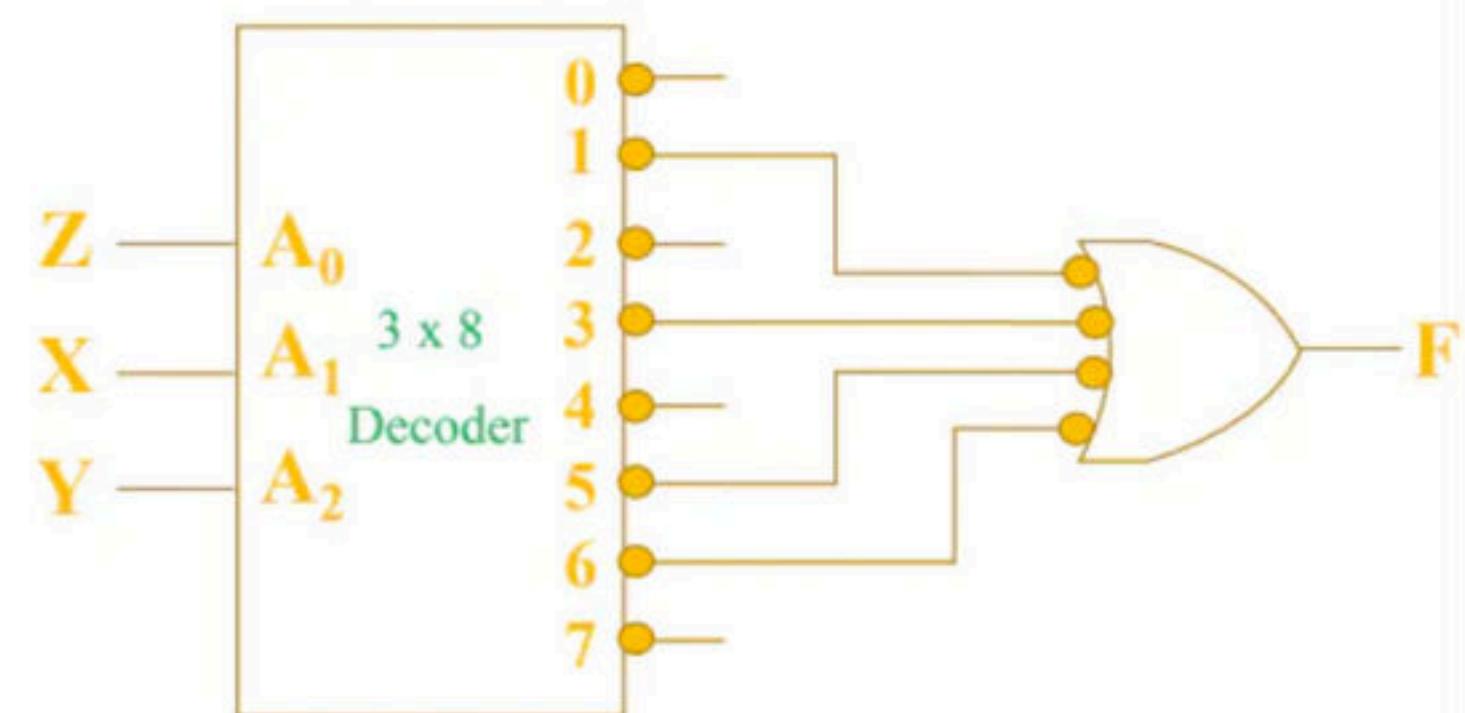
$$D_3 = 1 \text{ when } A_0 = 1, A_1 = 1$$

The value of  $f(a, b, c)$  will be \_\_\_\_\_



**Q.** A 3 line to 8-line decoder, with active low outputs, is used to implement a 3-variable Boolean function as shown in the figure. The simplified form of Boolean function F (X,Y,Z) implemented in ‘Product of Sum’ form will be.

- (a)  $(X + Z) \cdot (\bar{X} + \bar{Y} + \bar{Z}) \cdot (Y + Z)$
- (b)  $(\bar{X} + \bar{Y}) \cdot (X + Y + Z) \cdot (\bar{Y} + \bar{Z})$
- (c)  $(\bar{X} + \bar{Y} + Z) \cdot (\bar{X} + Y + Z) \cdot (X + \bar{Y} + Z) \cdot (X + Y + \bar{Z})$
- (d)  $(\bar{X} + \bar{Y} + \bar{Z}) \cdot (\bar{X} + Y + \bar{Z}) \cdot (X + Y + Z) \cdot (X + \bar{Y} + \bar{Z})$



Conversation of  
Demultiplexer <-----> Decoder



Inputs  $\longleftrightarrow$  Enable

Select lines  $\longleftrightarrow$  Inputs

Decoder is a special case of Demux , in which the select lines of Demux are treated as input's to the decoder and input of Demux is treated as Enable input of the Decoder

# **Implementation of higher order Decoders using lower order Decoders**

Q) Implement  $4 \times 16$  decoder using  $2 \times 4$  decoder

Q) Implement  $3 \times 8$  decoder using  $2 \times 4$  decoder

Q) Implement  $4 \times 16$  decoder using  $3 \times 8$  decoder

# Encoder

Encoder is a combinational circuit , which is used to convert

1. Octal to binary (  $8 \times 3$  encoder )
2. Decimal to Binary (  $10 \times 4$  encoder )
3. Hexadecimal to Binary (  $16 \times 4$  encoder )

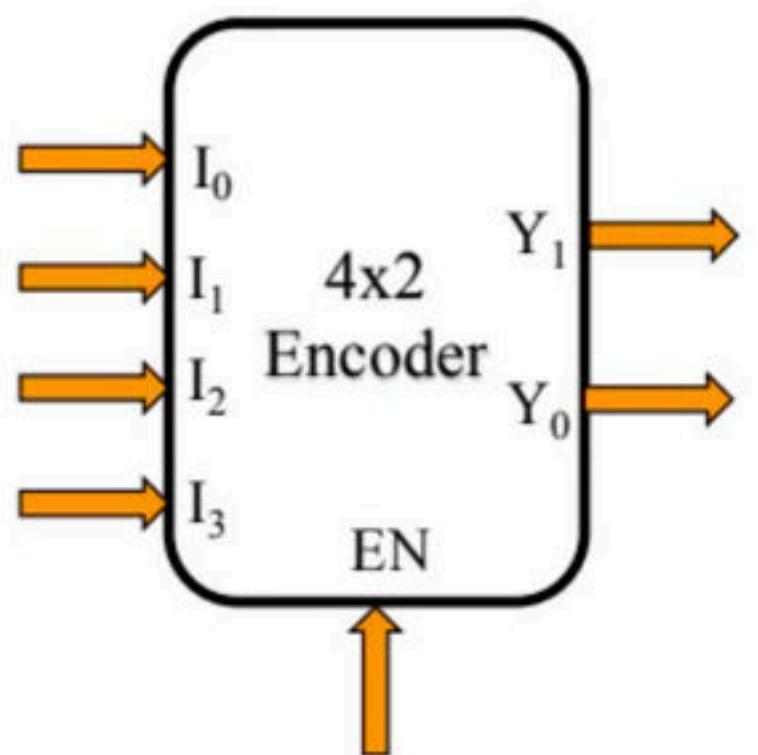
## General structure

$2^n \times n$

$n$  -----> number of outputs

$2^n$  -----> number of inputs

# 4 X 2 Encoder

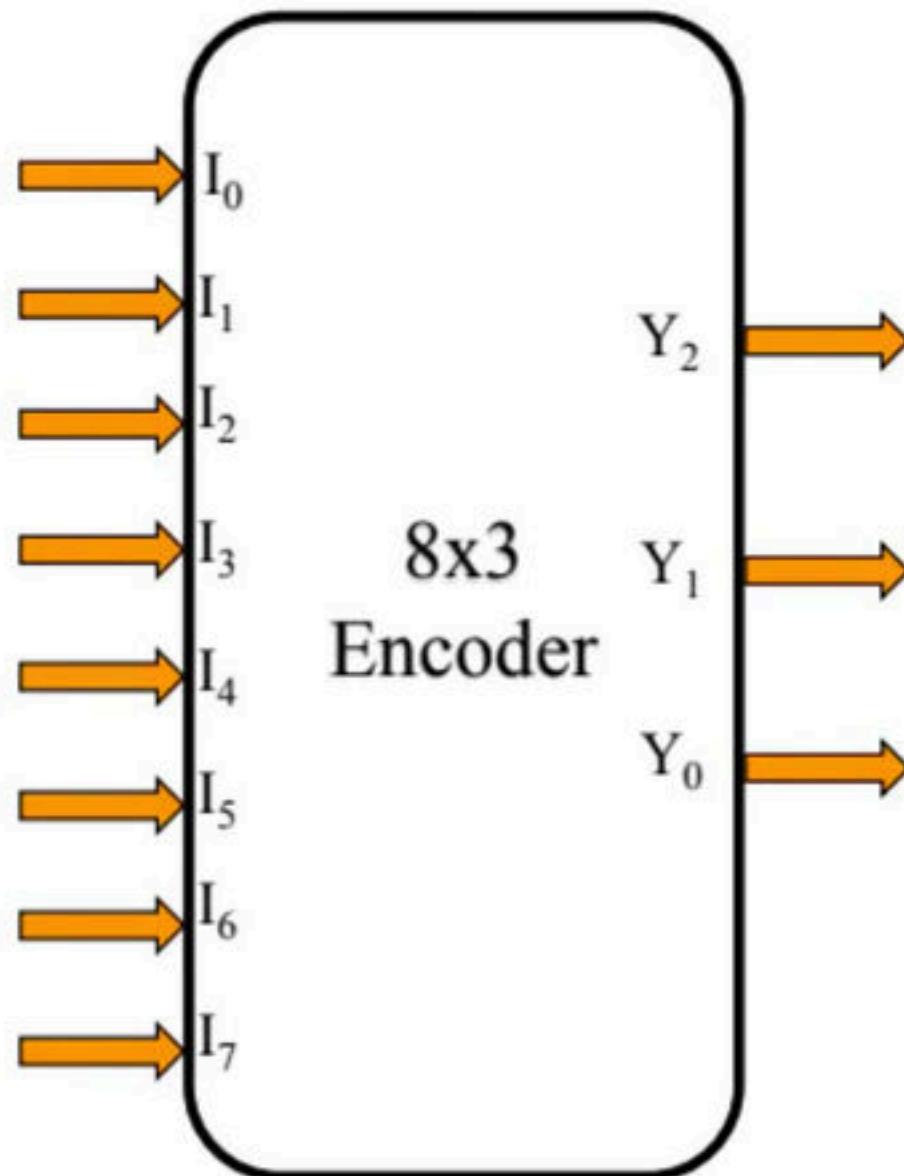


$I_3$	$I_2$	$I_1$	$I_0$	$Y_1$	$Y_0$	Valid

## Drawbacks of Encoder

- For an Encoder at a time only one among the all inputs is high , remaining inputs should be zero
- If multiple inputs are simultaneously high, then the output is not valid, to avoid this restriction we will go for priority encoder.

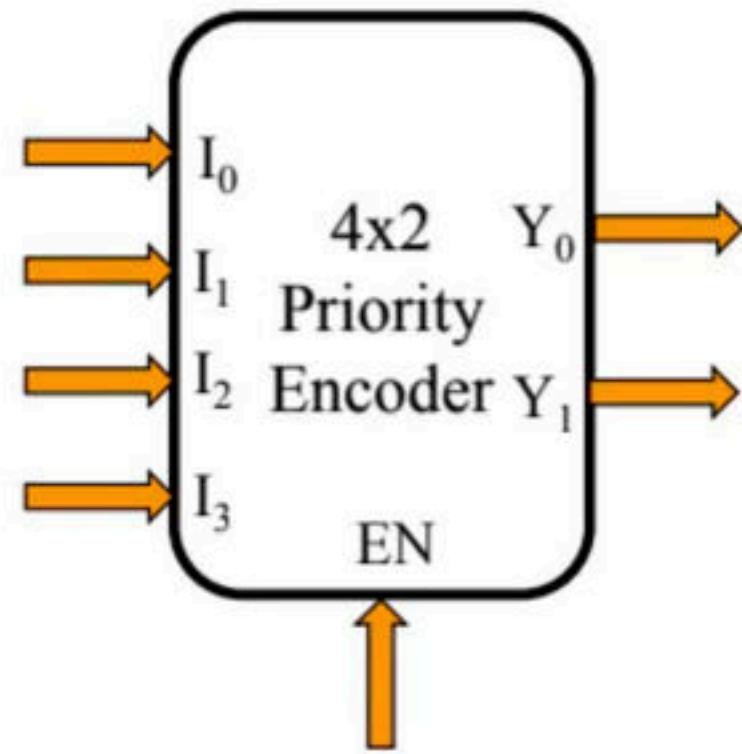
# 8 X 3 Encoder



# Priority Encoder

Priority encoder assign priority to every input and whenever higher priority input is one , then other inputs are not consider

# Priority Encoder



I3	I2	I1	I0	$Y_1$	$Y_0$	Valid

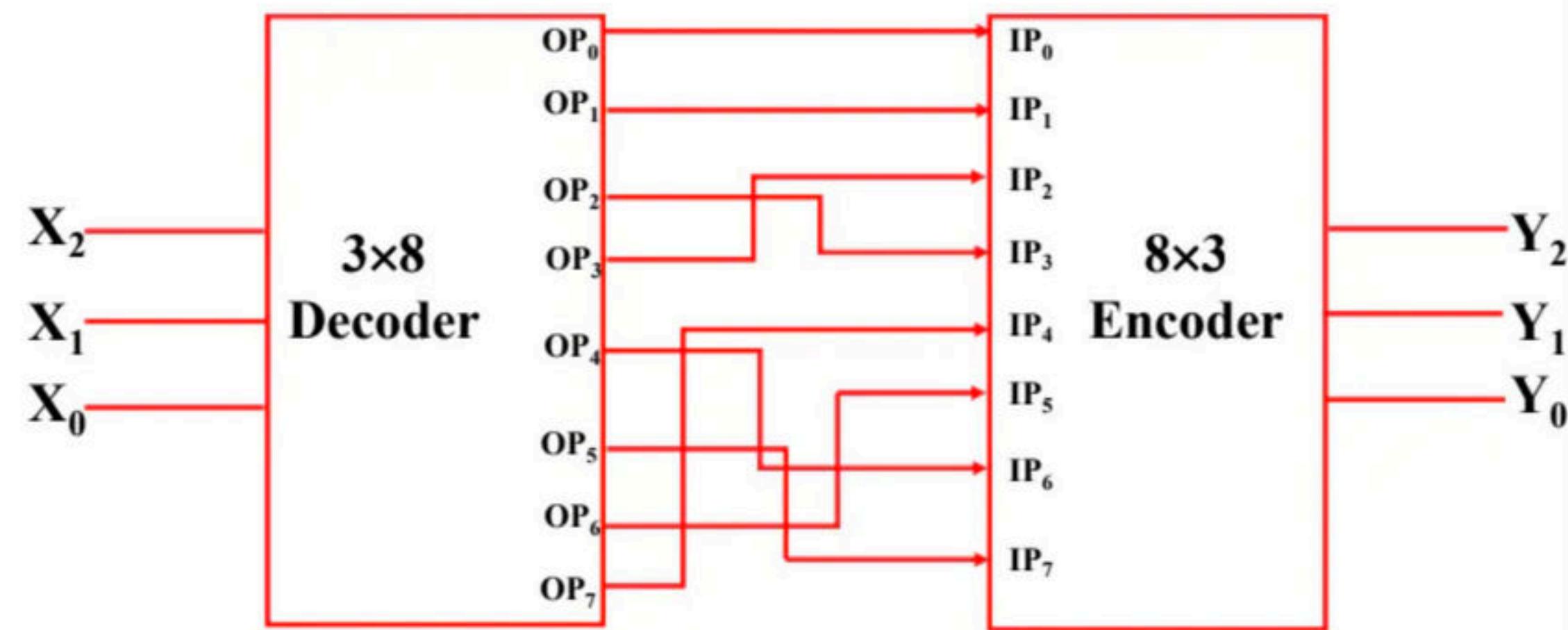
**Q.** Identify the circuit below

(a) Gray to binary converter

(b) Binary to excess 3 converter

(c) Binary to gray converter

(d) Excess-3 to binary converter



# Code Converter

Q) Design a circuit for BCD to 7– segment display decoder

Q) Design a circuit for Binary to BCD

Q) Design a circuit for BCD to EX-3 code

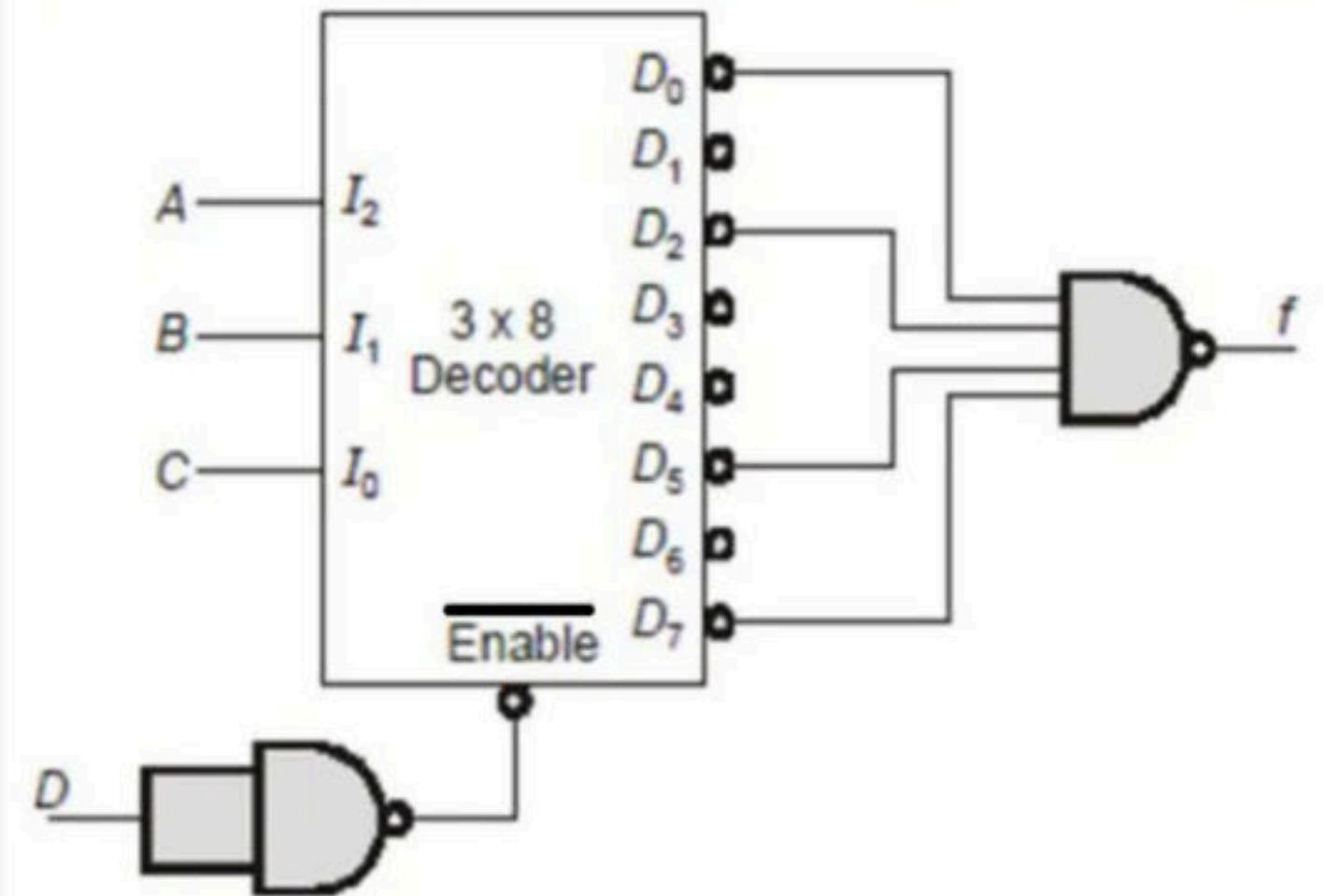
An  $n$ -bit carry look ahead adder is designed using only Ex-OR, AND, OR gates. The propagation delay of each Ex-OR gate is 20 ns and that of each AND, OR gates is  $t_0$  ns. If the total propagation delay of the adder circuit is 60 ns, then the value of  $t_0$  will be

(given that  $t_0 \leq 20$  ns)

- a. 10
- b. 15
- c. 20
- d. depends on ' $n$ ' value

A one bit full adder takes 75 nsec to produce sum and 50 nsec to produce carry. A 4 bit parallel adder is designed using this type of full adder. The maximum rate of additions per second can be provided by 4 bit parallel adder is  $A \times 10^6$  additions/sec. The value of A is \_\_\_\_\_

The logic function  $f(A, B, C, D)$  implemented by the circuit shown below is



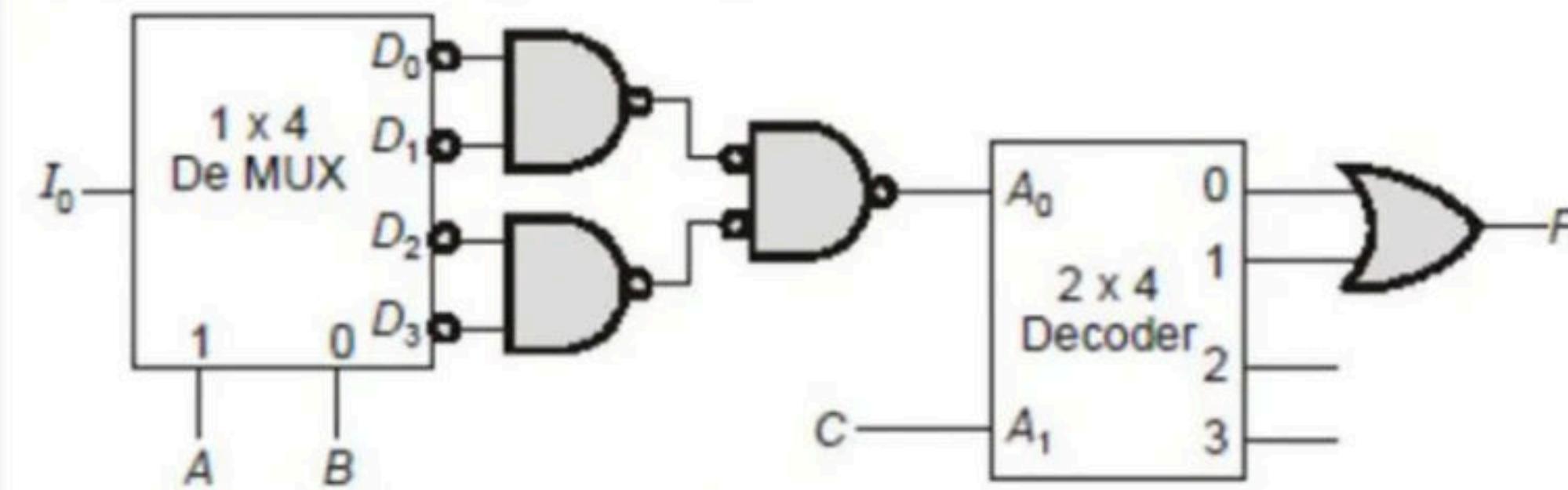
a.  $\bar{D}(A \oplus C)$

b.  $\bar{D}(A \odot C)$

c.  $\bar{D}(A \oplus B)$

d.  $D(A \odot C)$

Consider the logic circuit given below



The minimized expression for  $F$  is

- a.  $\bar{C}$
- b.  $I_0$
- c.  $C$
- d.  $\bar{I}_0$

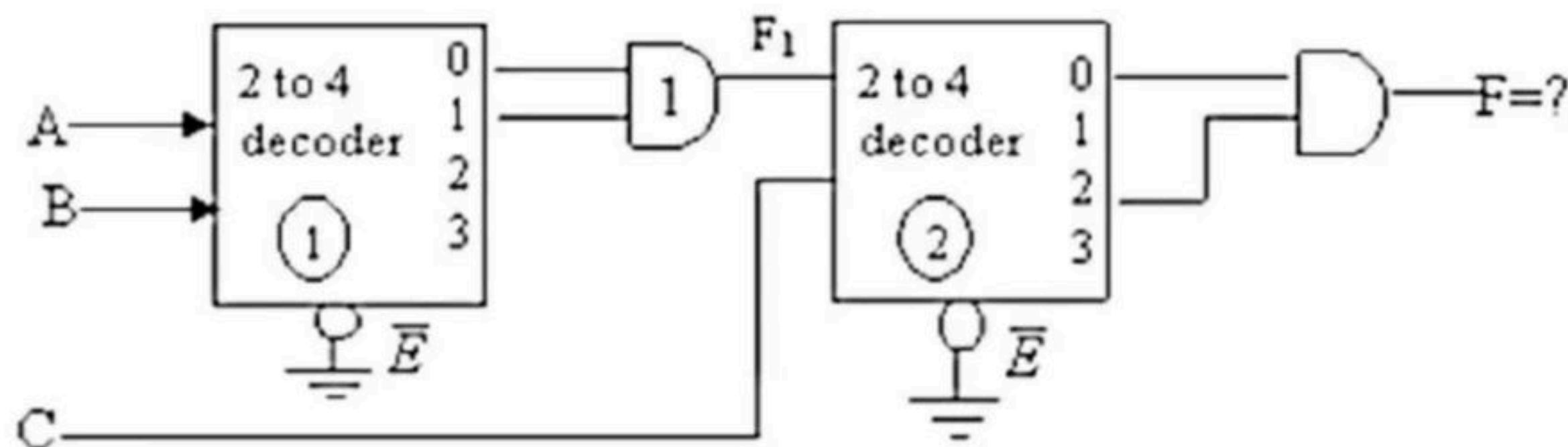
Find the output of the following circuit

(a) 1

(b) C

(c)  $\bar{C}$

(d) 0



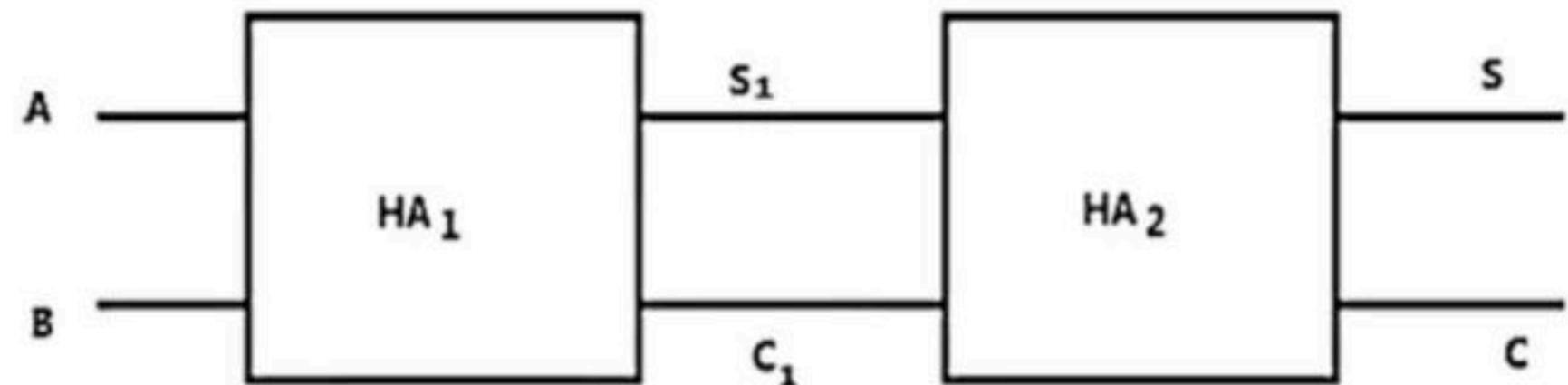
Two Half Adders are connected in cascade as shown in figure below. The output "S" and "C" are

(a)  $S = A \oplus B, C = AB$

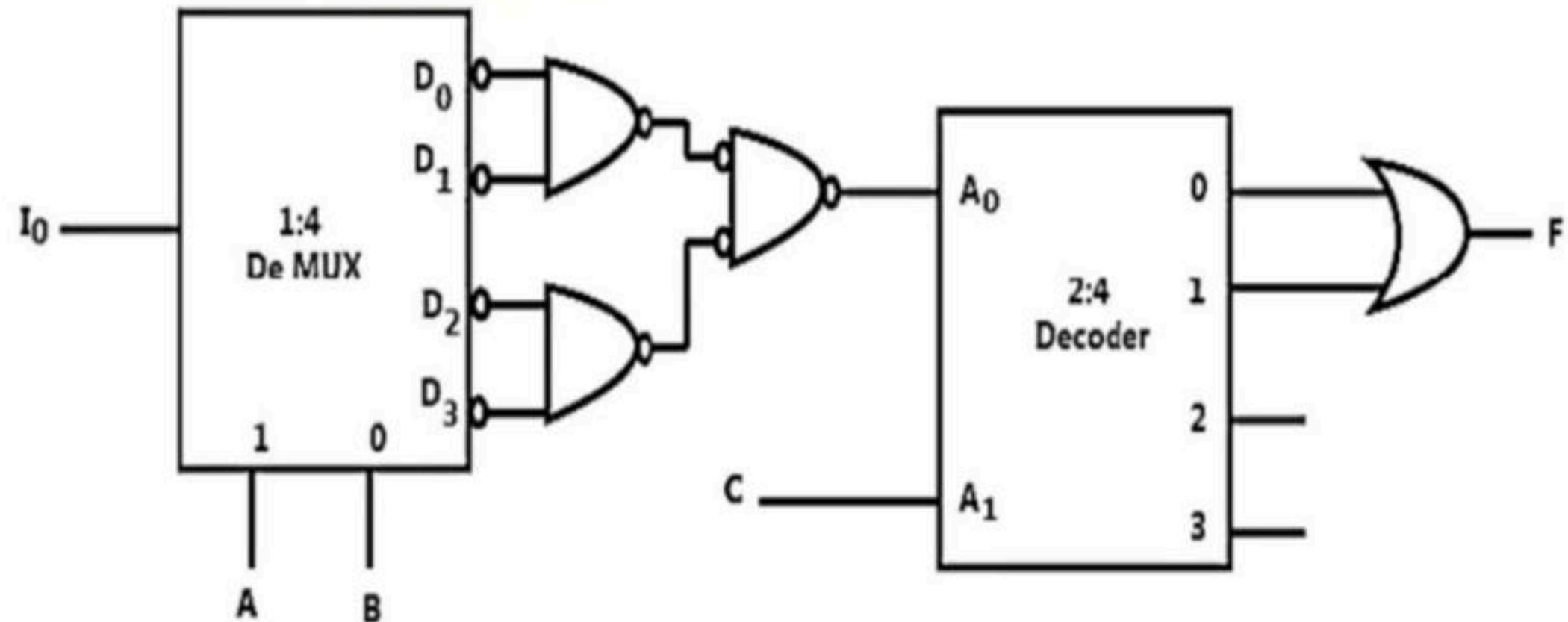
(b)  $S = A \Theta B, C = 0$

(c)  $S = A + B, C = 0$

(d)  $S = AB, C = 0$



Consider the logic circuit given below

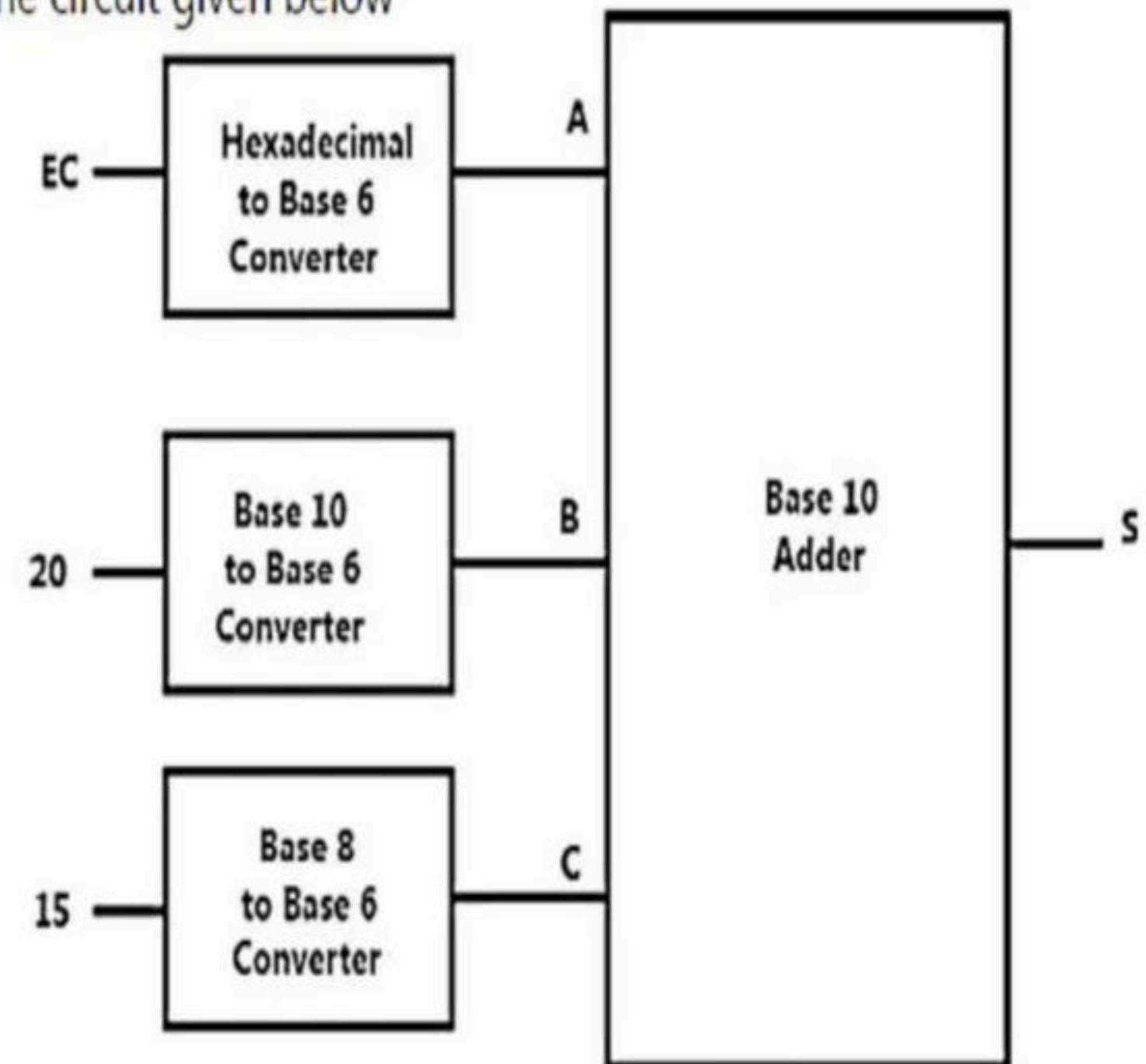


The minimized expression for  $F$  is

- (a)  $\bar{C}$
- (c)  $C$

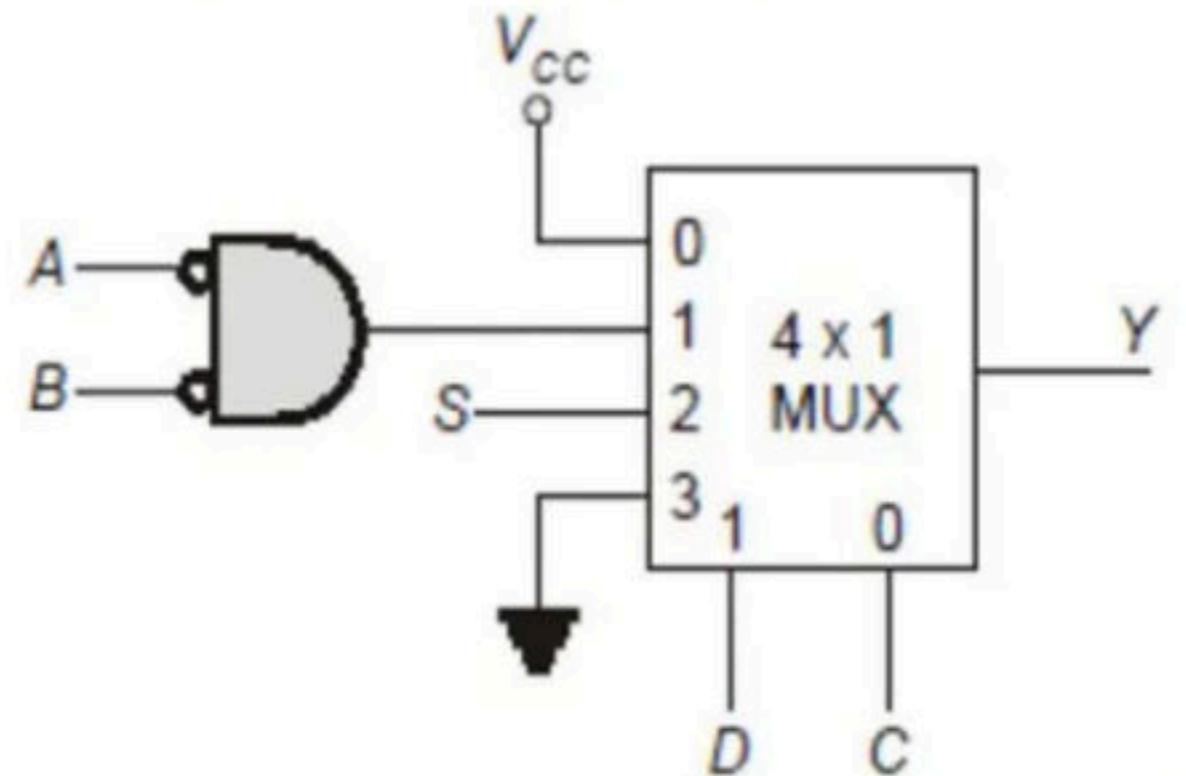
- (b)  $I_0$
- (d)  $\bar{I}_0$

Consider the circuit given below



The output of each converter is given to adder which adds them considering decimal number. The output of adder is S. The value of S is \_\_\_\_.

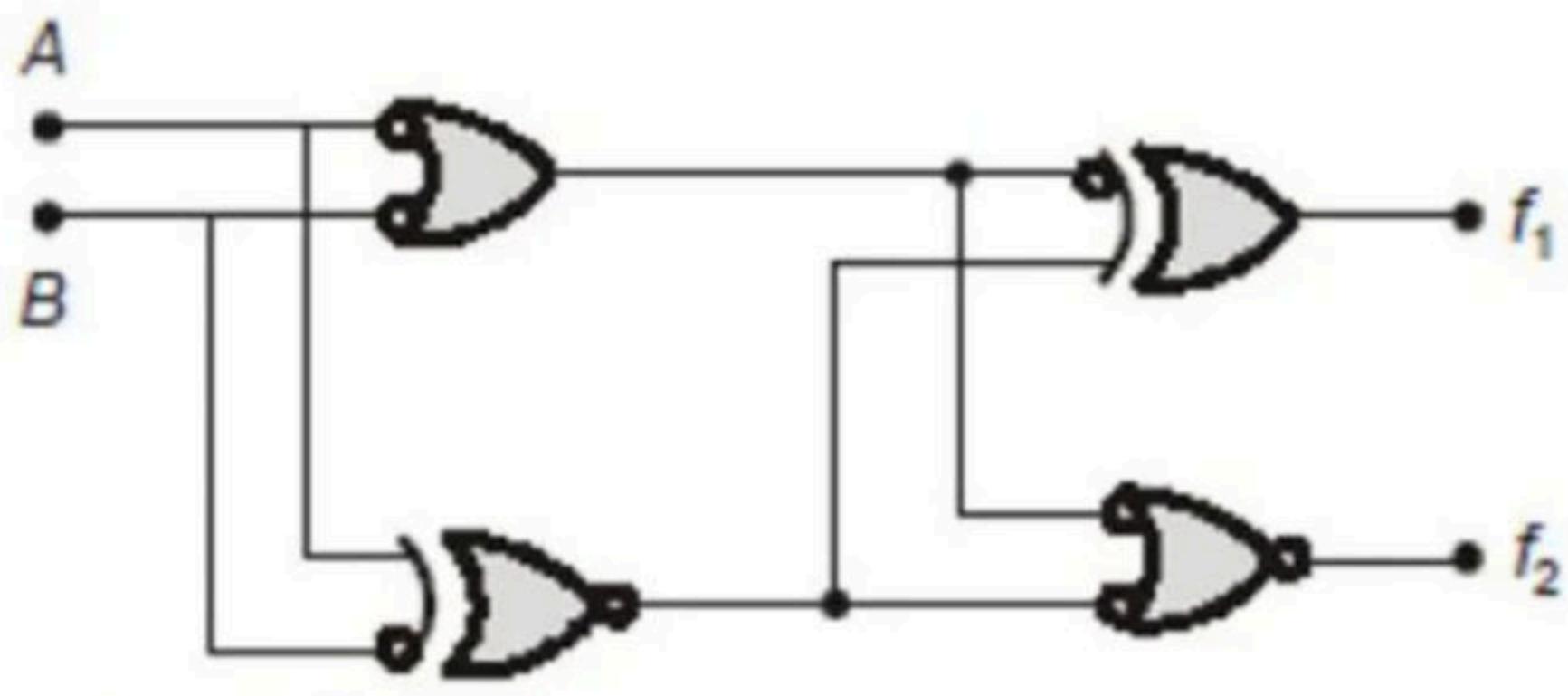
Consider the circuit given below



Which of the following statements is true for Y.

- a.  $Y = \bar{C}\bar{D} + \bar{D}C(\bar{A} + \bar{B}) + \bar{C}DS$
- b.  $Y = CD + D\bar{C}(\bar{A} + \bar{B}) + CDS$
- c.  $Y = \bar{C}\bar{D} + (\bar{D} + C)(\bar{A} + \bar{B}) + \bar{C} + \bar{D} + \bar{S}$
- d.  $Y = \bar{C}\bar{D} + (D + \bar{C})(\bar{A} + \bar{B}) + \bar{C} + \bar{D} + \bar{S}$

**Consider the digital circuit shown below**



**It represents**

- a. Half adder followed by half subtractor
- b. Half subtractor followed by half adder
- c. Half adder followed by a half adder
- d. A full adder

The minimum number of NOR gates required to realize the half adder circuit is

-----.

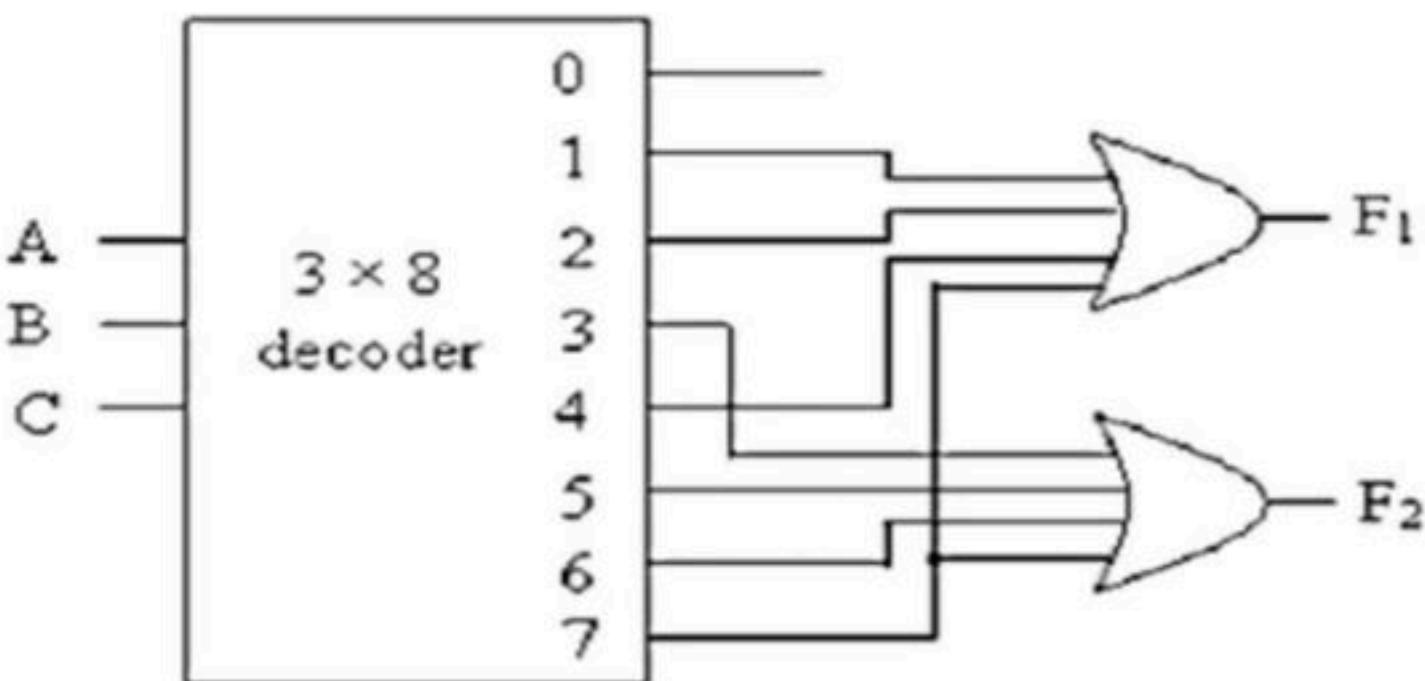
The output  $Y$  of a 2 bit comparator is logic 1 whenever the 2 bit input  $A$  is greater than the 2-bit input  $B$ . The number of combinations for which the output is logic 1 is \_\_\_\_\_.

How many 1-bit comparators, 2-input AND gates, 2-input OR gates required to design a 2-bit comparator.

- (a) 2, 3, 2
- (b) 2, 2, 3
- (c) 2, 3, 3
- (d) 2, 2, 2

What is the name of given circuit?

- (a) Full Subtractor
- (b) Full Adder
- (c) 3-bit even parity generator
- (d) 3-bit odd parity generator



In a 2-bit magnitude comparator circuit ( $A = A_1A_0$ ,  $B = B_1B_0$ ), the expression for  $A > B$  &  $A < B$  is

(a)  $A > B = \bar{A}_1 \bar{B}_1 + (A_1 \oplus B_1) A_0 \bar{B}_0$

$A < B = \bar{A}_1 \bar{B}_1 + (A_1 \oplus B_1) \bar{A}_0 B_0$

(c)  $A > B = A_1 B_1 + (A_1 \oplus B_1) A_0 \bar{B}_0$

$A < B = A_1 B_1 + (A_1 \oplus B_1) \bar{A}_0 B_0$

(b)  $A > B = A_1 \bar{B}_1 + (A_1 \oplus B_1) A_0 \bar{B}_0$

$A < B = \bar{A}_1 B_1 + (A_1 \oplus B_1) \bar{A}_0 B_0$

(d)  $A > B = A_1 B_1 + (A_1 \oplus B_1) A_0 \bar{B}_0$

$A < B = A_1 B_1 + (A_1 \oplus B_1) \bar{A}_0 B_0$

An  $8 \times 1$  multiplexer has inputs A, B and C connected to the selection input  $S_2$ ,  $S_1$ , and  $S_0$ , respectively. The data inputs  $I_0$  through  $I_7$  are as follows:

$$I_1 = I_2 = I_7 = 0; I_3 = I_5 = 1; I_0 = I_4 = D; \text{ and } I_6 = \bar{D};$$

The Boolean function that the multiplexer implements is

(a)  $Y = \sum m(1, 6, 7, 9, 10, 11, 13)$

(c)  $Y = \sum m(4, 5, 7, 8, 9, 11, 15)$

(b)  $Y = \sum m(1, 6, 7, 9, 10, 11, 12)$

(d)  $Y = \sum m(0, 1, 3, 4, 5, 7, 9, 11)$

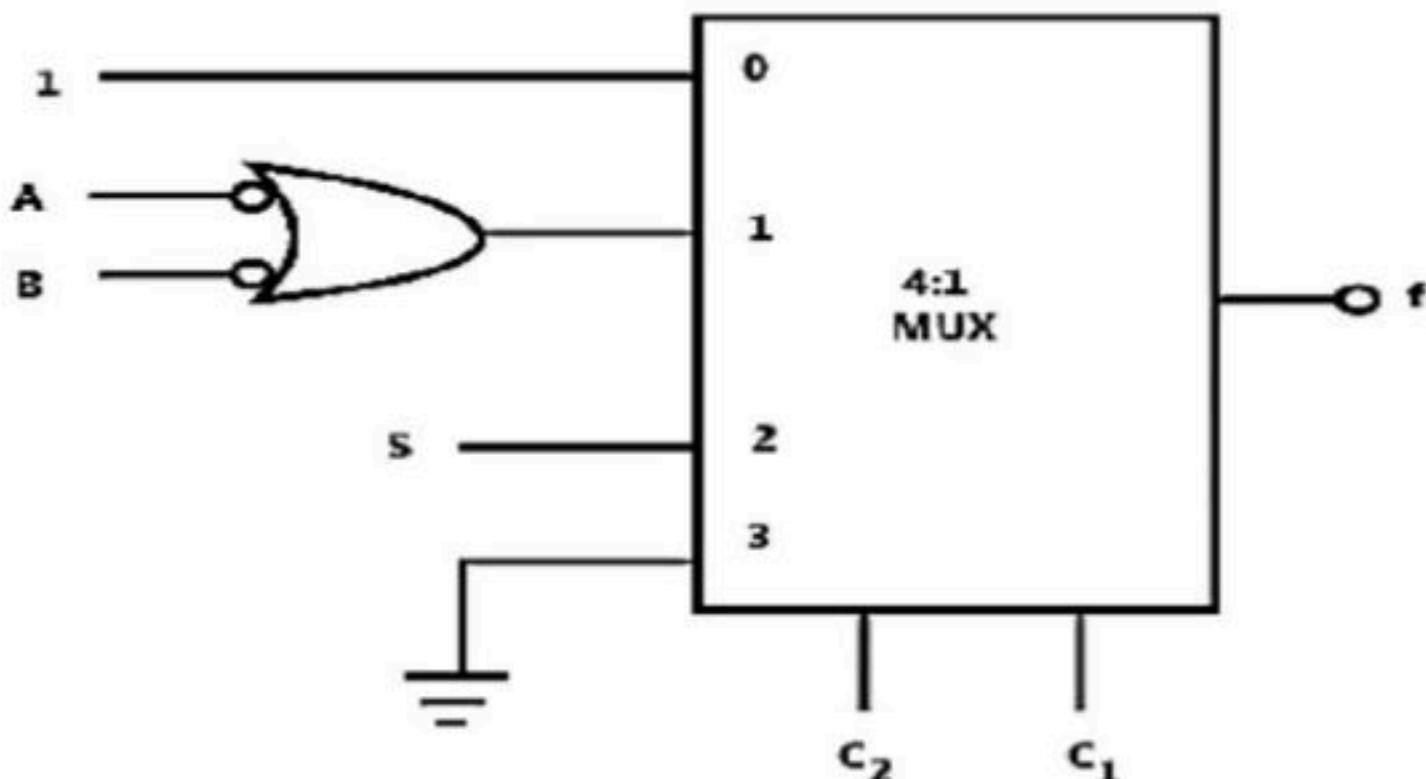
In the following MUX, find the output f.

(a)  $C_2 \cdot \bar{C}_1 S + \bar{C}_2 C_1 (\bar{A} + \bar{B})$

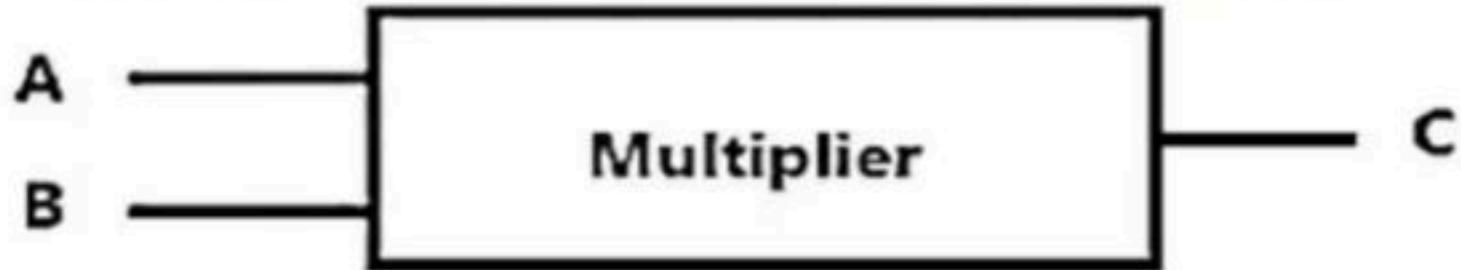
(b)  $\bar{C}_2 \bar{C}_1 + C_2 C_1 + \bar{C}_2 \bar{C}_1 S + \bar{C}_2 C_1 \bar{A} \bar{B}$

(c)  $\bar{A} \bar{B} + S$

(d)  $\bar{C}_2 \bar{C}_1 + C_2 \bar{C}_1 S + \bar{C}_2 C_1 (\bar{A} \bar{B})$



Consider a 3-bit number A and 2 bit number B are given to a multiplier. The output of multiplier is realized using AND gate and one bit full adders. If minimum number of AND gates required are X and one bit full adders required are Y, then  $X + Y = \underline{\hspace{2cm}}$ .



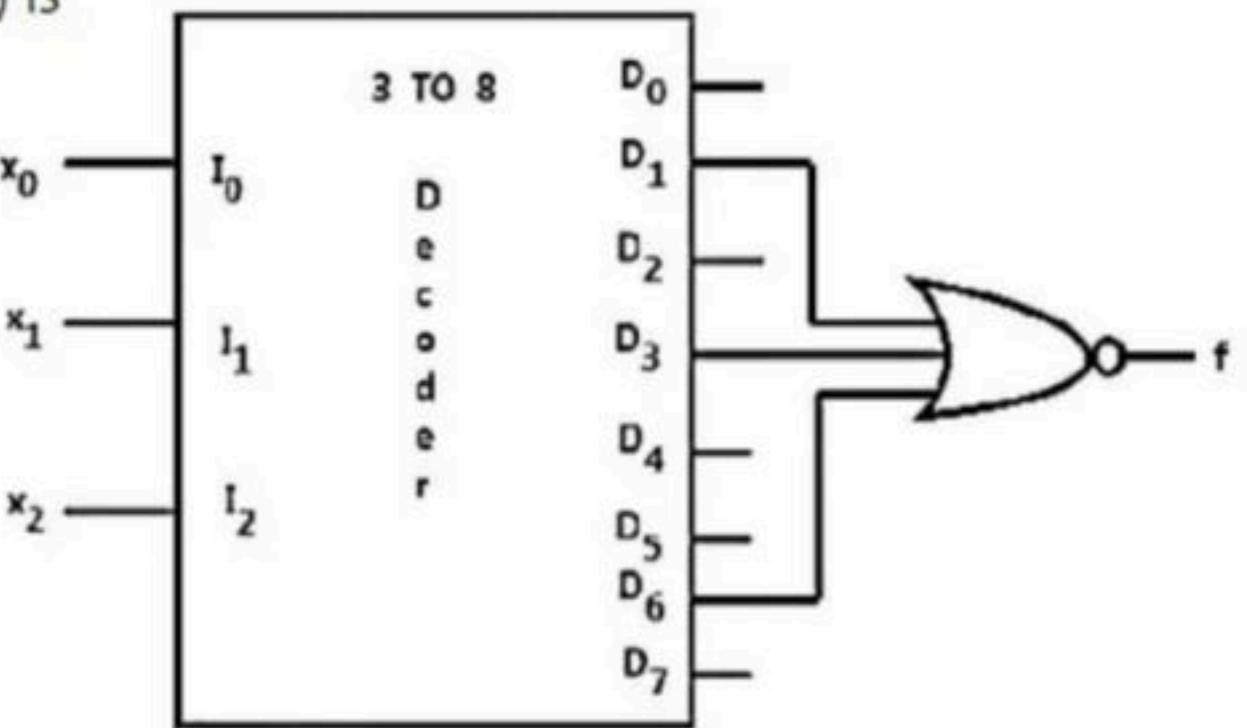
In the following circuit the function  $f(x_2, x_1, x_0)$  is

(a)  $\prod M(0, 2, 4, 5)$

(b)  $\sum m(0, 2, 4, 5, 7)$

(c)  $\sum m(1, 3, 6)$

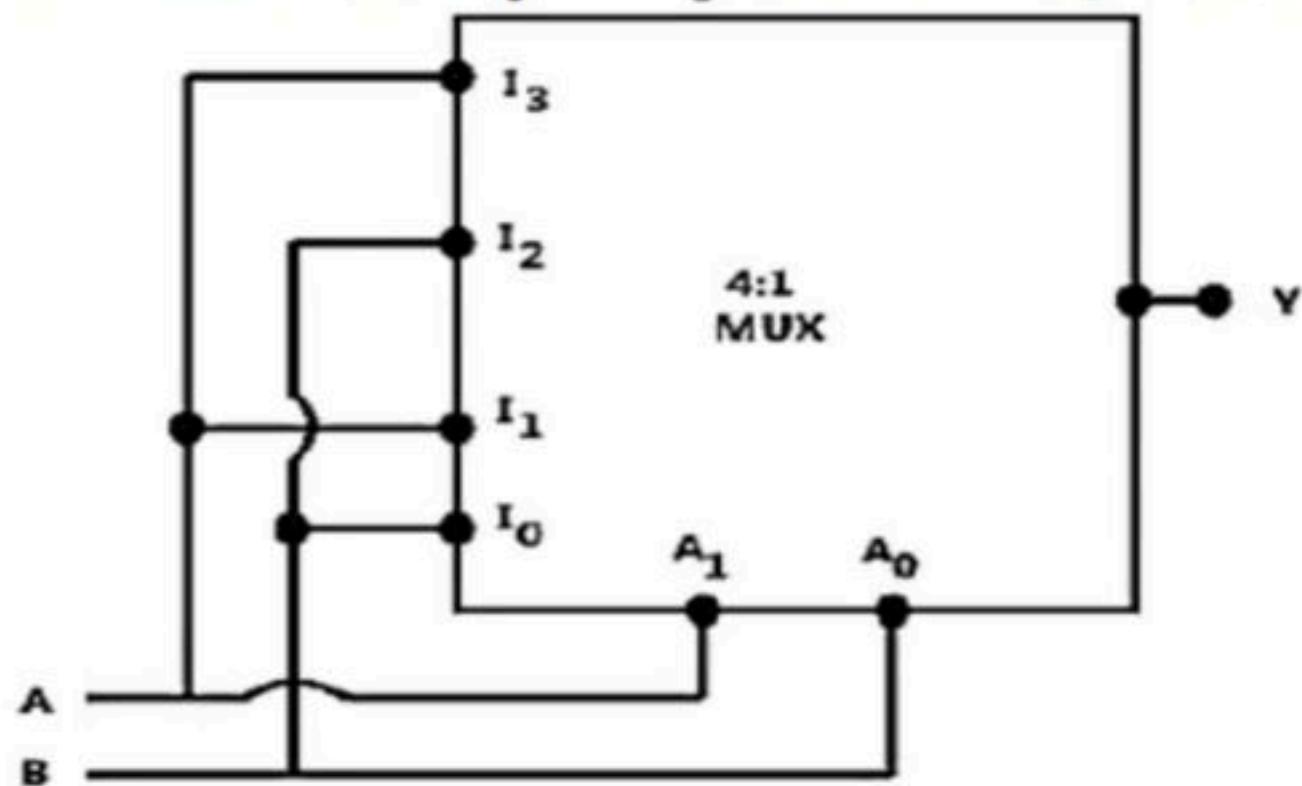
(d)  $\sum M(1, 3, 6)$



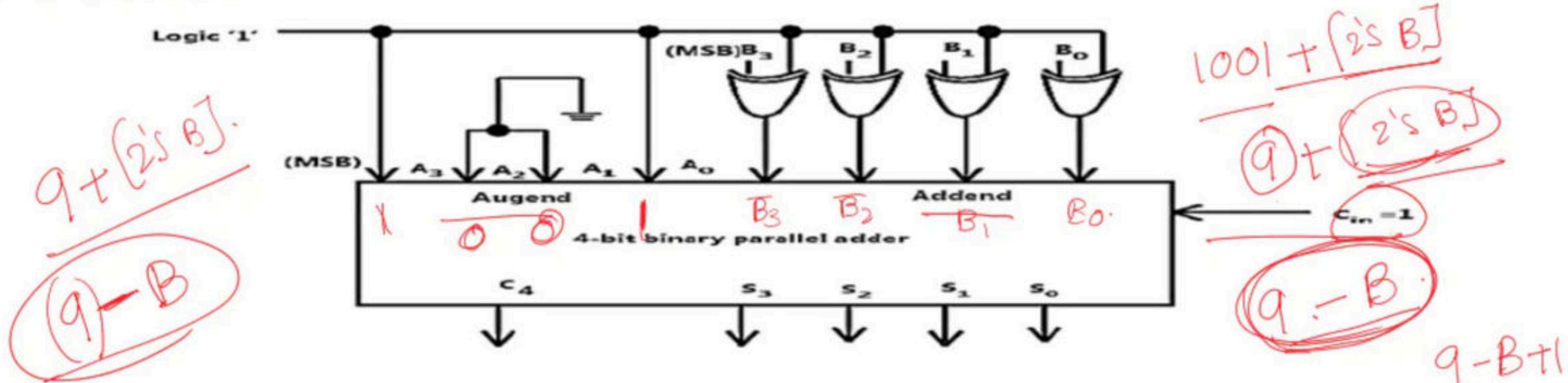
A combinational logic circuit has three inputs A, B and C and one output Y. The output  $Y = 1$  when at least two inputs are 1. Otherwise,  $Y = 0$ . In its minimized SOP realization, the maximum number of two input terms is \_\_\_\_\_.

A gate having two inputs ( $A$ ,  $B$ ) and one output ( $Y$ ) is implemented using 4 : 1 MUX as shown in figure below.  $A_1$  (MSB) and  $A_0$  are the control bits and  $I_0$  to  $I_3$  are the inputs to the MUX. The gate is

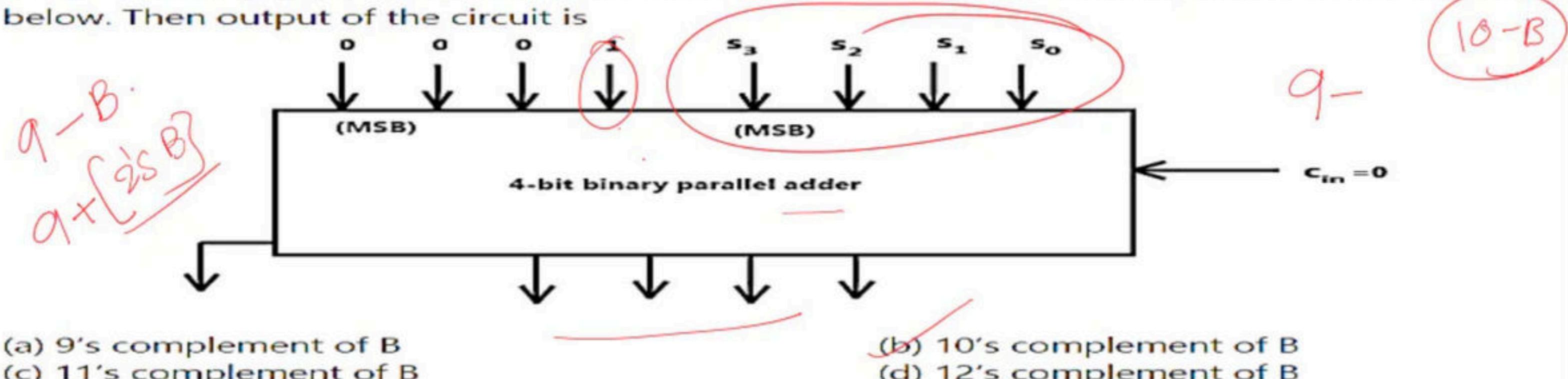
- (a) AND
- (b) NOR
- (c) OR
- (d) EX-OR



Consider the digital circuit shown below. A single digit decimal number(B) is converted into its 4 bit binary equivalent( $B_3B_2B_1B_0$ ) and then applied to the addend bits of the adder as shown below:



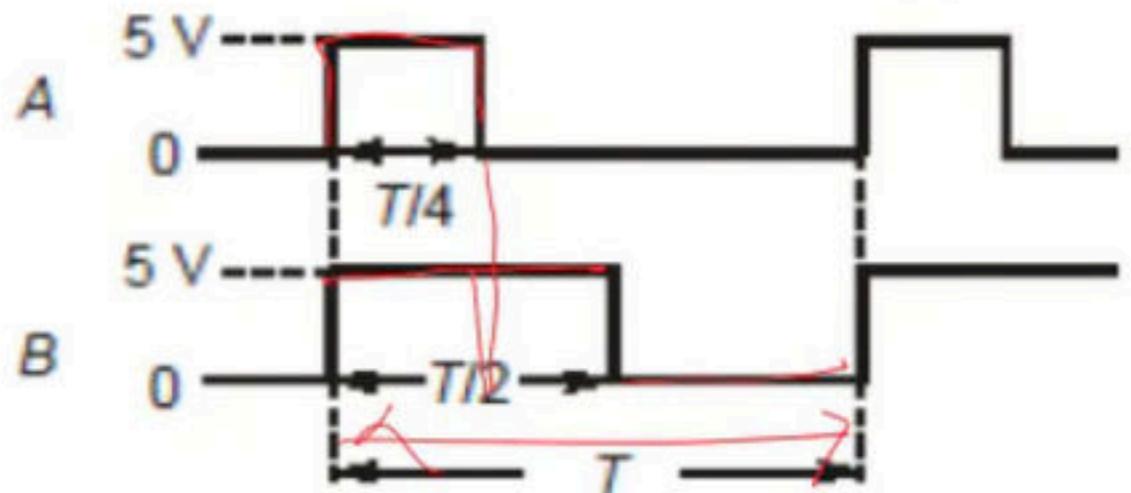
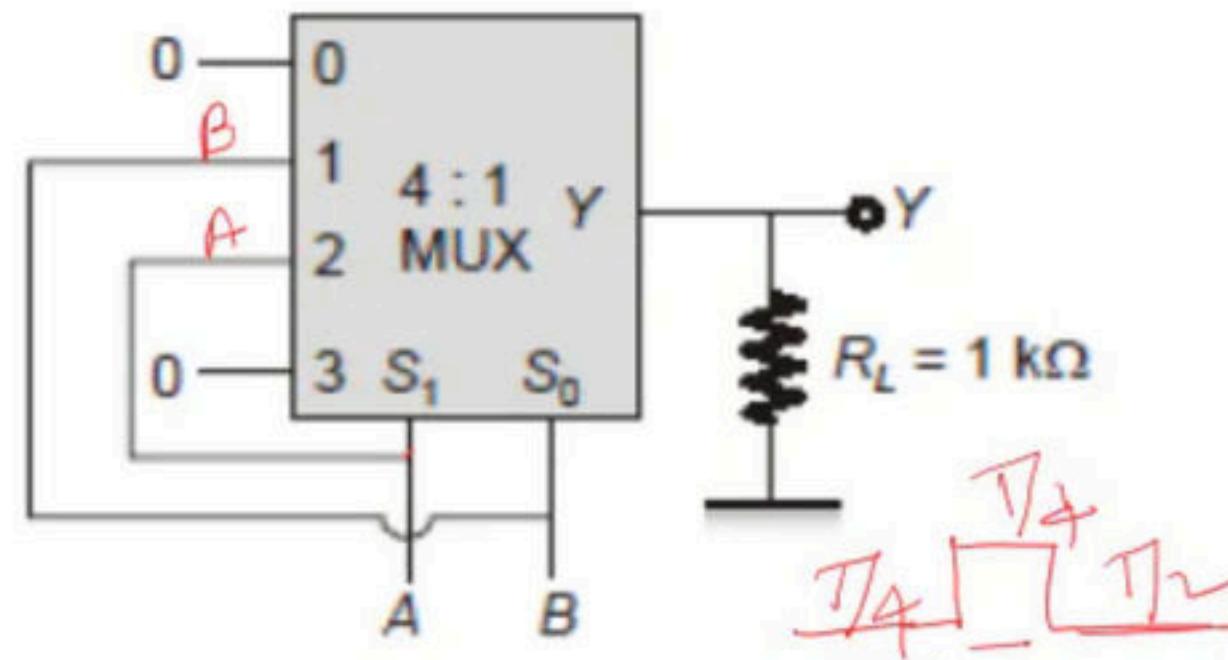
If  $C_4 = 1$  and  $S_3S_2S_1S_0$  are given to the addend bits of the 4-bit binary parallel adder as shown below. Then output of the circuit is



- (a) 9's complement of B
- (c) 11's complement of B

- (b) 10's complement of B
- (d) 12's complement of B

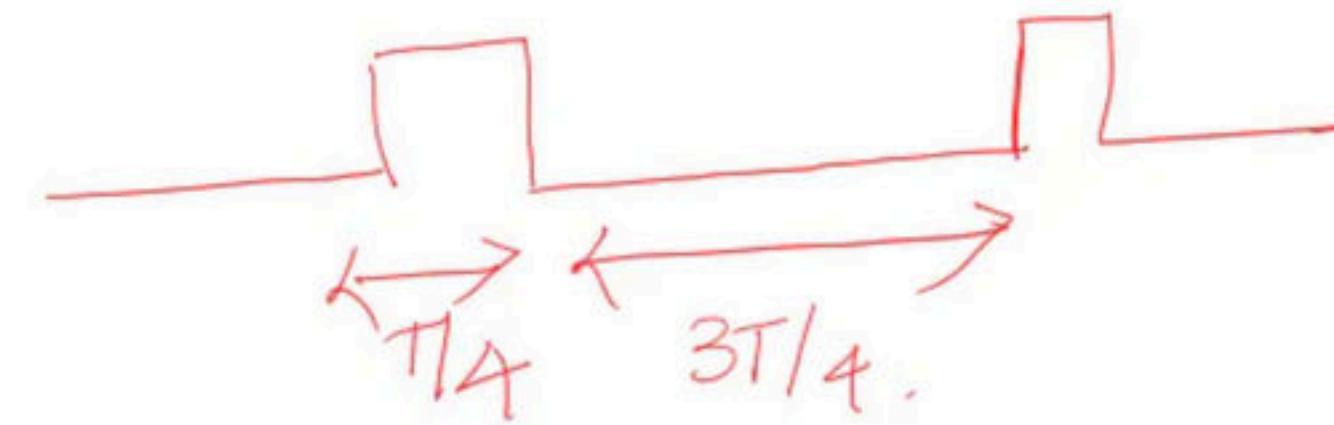
Consider the 4 : 1 MUX based circuit as shown in the figure. A and B are two periodic signals with duty cycles 25% and 50% respectively as given in the figure. If +5 V and 0 V are used to represent logic-1 and logic-0 respectively, then the average power dissipated by the resistor  $R_L$  will be \_\_\_\_\_ mW.



$$Y = \overline{AB}B + \overline{A}\overline{B}A$$

$$Y = \overline{AB} + \overline{A}\overline{B}$$

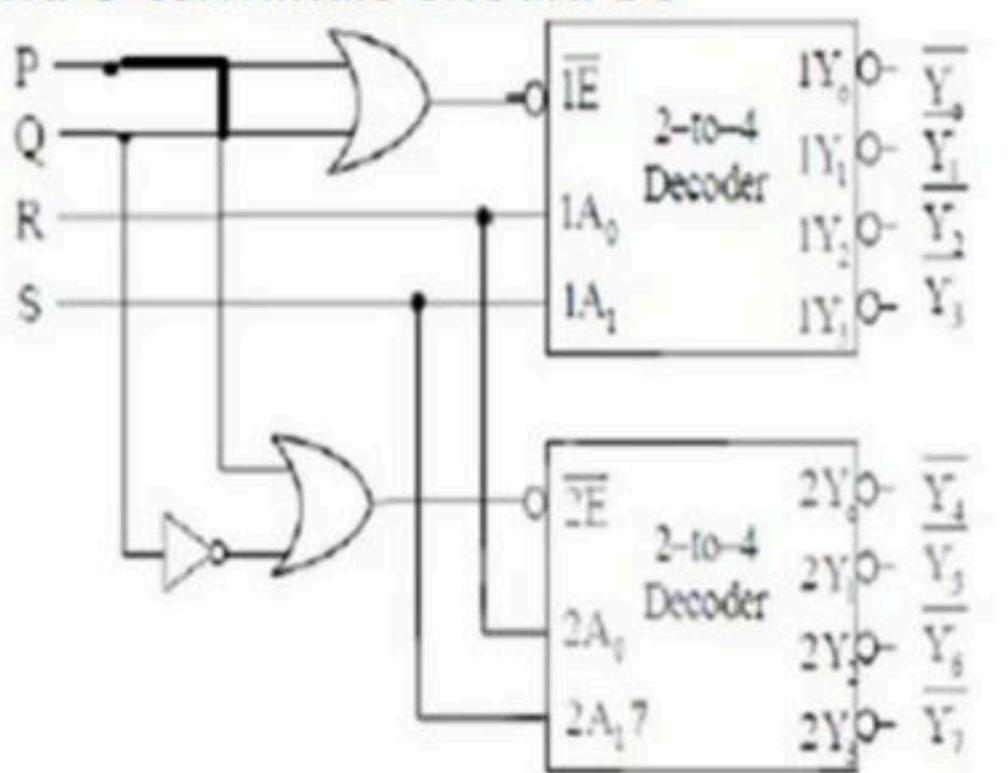
$$Y = A \oplus B =$$



$$D = \frac{T/4}{T} = \frac{1}{4}$$

$$D = \frac{T/4}{T}$$

A 1-to-8 demultiplexer with data input  $D_{in}$ , address inputs  $S_0, S_1, S_2$  (with  $S_0$  as the LSB) and  $\bar{Y}_0$  to  $\bar{Y}_7$  as the eight demultiplexed output, is to be designed using two 2-to-4 decoders (with enable input  $\bar{E}$  and address input  $A_0$  and  $A_1$ ) as shown in the figure.  $D_{in}, S_0, S_1$  and  $S_2$  are to be connected to P, Q, R and S, but not necessarily in this order. The respective input connections to P, Q, R and S terminals should be



- A.  $\underline{S_2}, \bar{D}_{in}, S_0, S_1$
- B.  $S_1, \bar{D}_{in}, S_0, S_2$
- C.  $\bar{D}_{in}, S_0, S_1, S_2$
- D.  $\bar{D}_{in}, S_2, S_1, S_0$