# Unit-5 : Combinational Circuit

Multiplexers-De-multiplexers
Decoder-Encoder

**Combinational circuits** are defined as the time independent circuits which do not depends upon previous inputs to generate any output are termed as combinational circuits.

**Sequential circuits** are those which are dependent on clock cycles and depends on present as well as past inputs to generate any output.

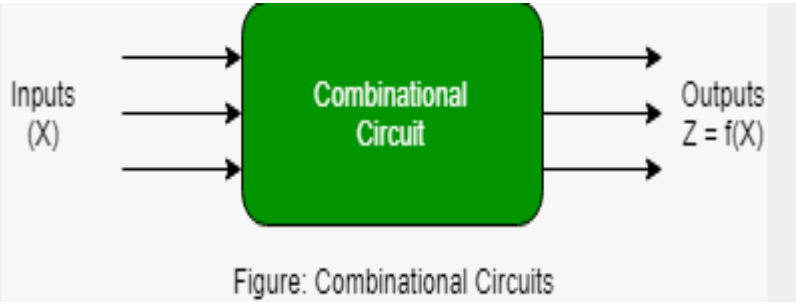
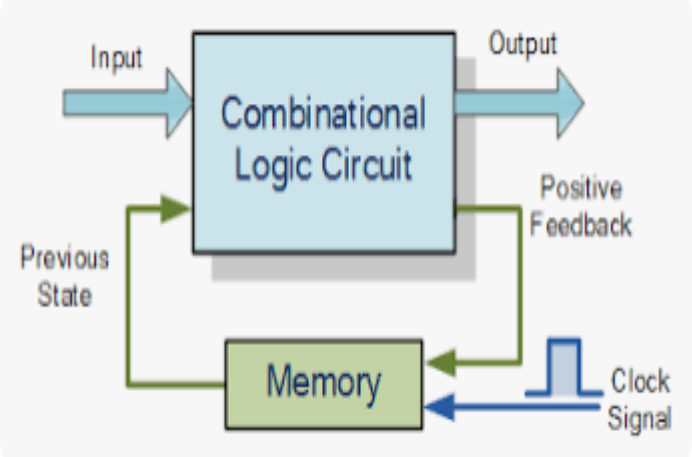


Figure: Combinational Circuits



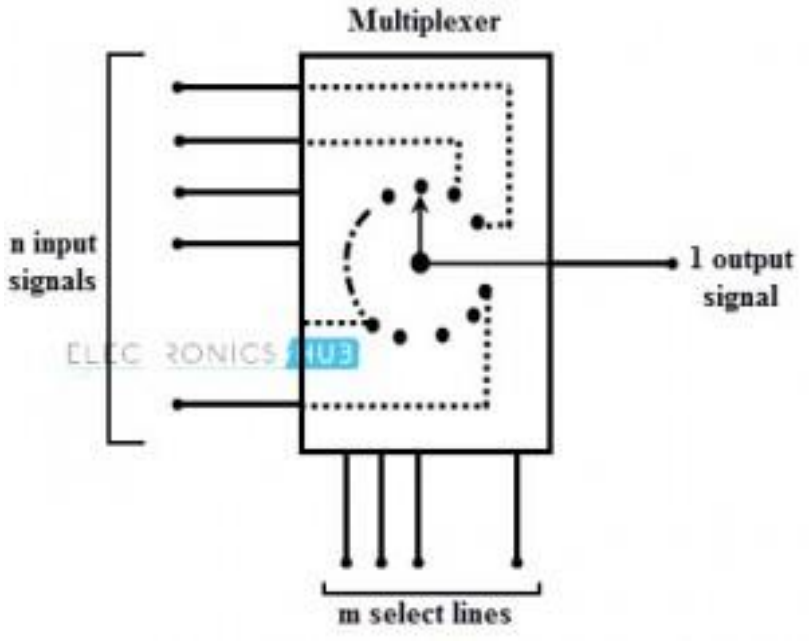
**Features of Combinational Circuit** –

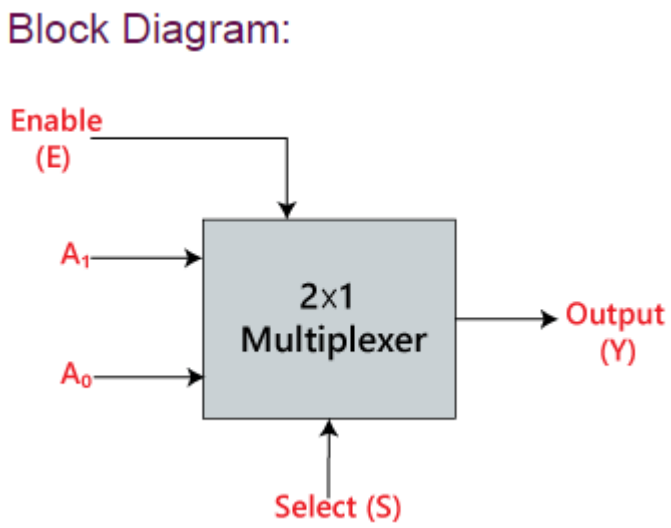
1. In this output depends only upon present input.
2. Simplest design and Speed is fast
3. There is no feedback between input and output.
4. Elementary building blocks: Logic gates
5. Used for arithmetic as well as boolean operations.
6. Combinational circuits don’t have capability to store any state.
7. As combinational circuits don’t have clock, they don’t require triggering.
8. These circuits do not have any memory element.

**Examples –** Encoder, Decoder, Multiplexer, Demultiplexer

# Multiplexer

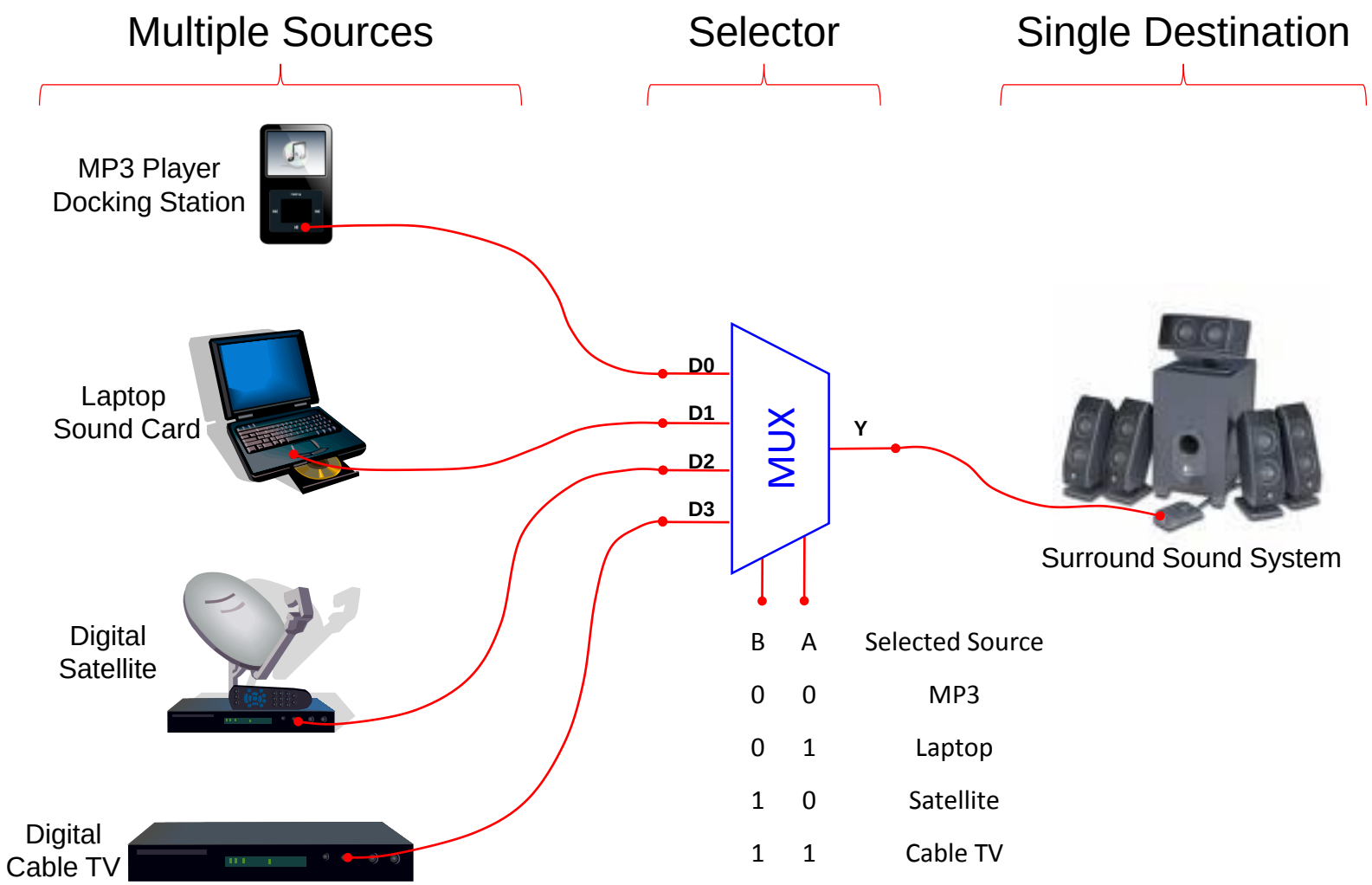
- Also called data selectors.
- Basic function: select one of its $2^n$ data input lines and place the corresponding information onto a single output line.
- $n$ input bits needed to specify which input line is to be selected.
  - Place binary code for a desired data input line onto its $n$ select input lines.



# Typical Application of a MUX

- One of the primary applications of multiplexers is to provide for the transmission of information from several sources over a single path. This process is known as multiplexing




| B | A | Selected Source |
|---|---|---|
| 0 | 0 | MP3 |
| 0 | 1 | Laptop |
| 1 | 0 | Satellite |
| 1 | 1 | Cable TV |

# MCQ

Q. The two input MUX would have _____

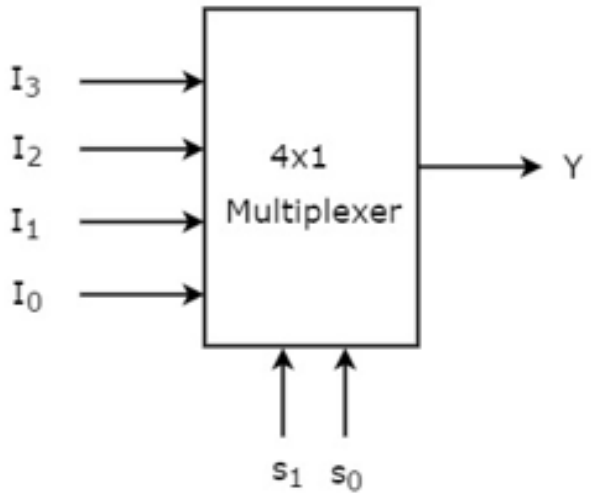
a) 1 select line
b) 2 select lines
c) 4 select lines
d) 3 select lines

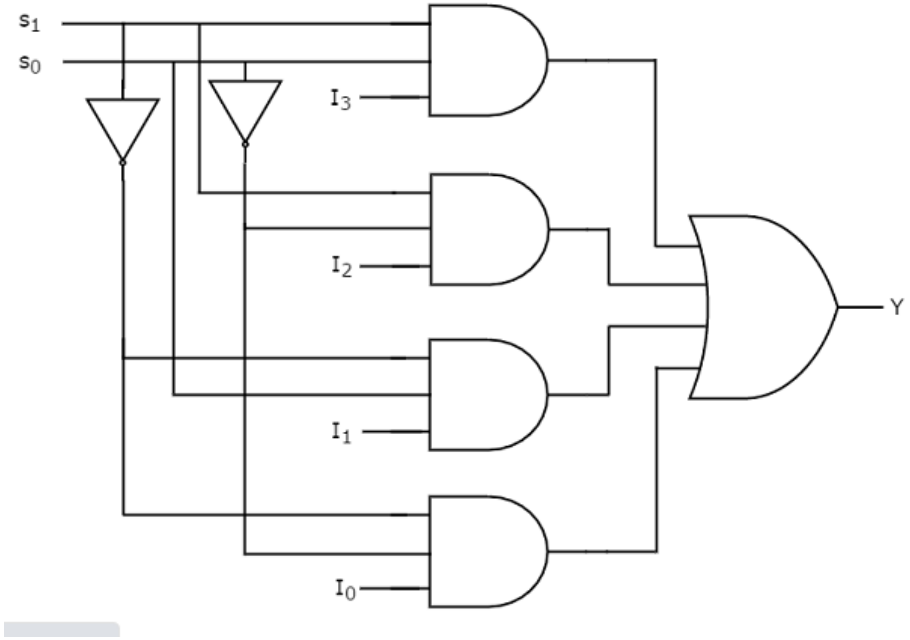
# MCQ

Q. 4 to 1 MUX would have _____

a) 2 inputs
b) 3 inputs
c) 4 inputs
d) 5 inputs

# 4:1 Multiplexer




| Selection Lines | | Output |
|---|---|---|
| **$S_1$** | **$S_0$** | **Y** |
| 0 | 0 | $I_0$ |
| 0 | 1 | $I_1$ |
| 1 | 0 | $I_2$ |
| 1 | 1 | $I_3$ |

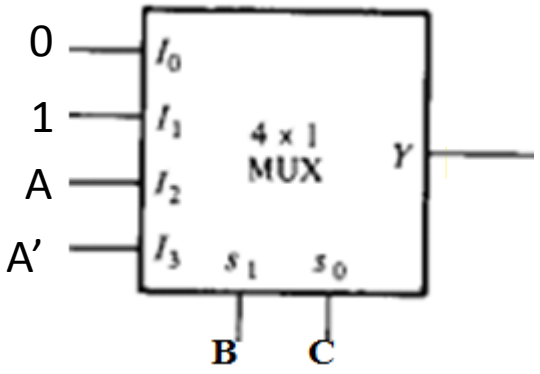
$$Y = S_1'S_0'I_0 + S_1'S_0I_1 + S_1S_0'I_2 + S_1S_0I_3$$

# Expression implementation with Multiplexer

Q. Y(A, B,C) = ∑m (1, 3, 5, 6) implement with 4:1 multiplexer

$$Y = \overline{A}\,\overline{B}C + \overline{A}BC + A\overline{B}C + AB\overline{C}$$

Any 2 input (AB/BC/AC) select line



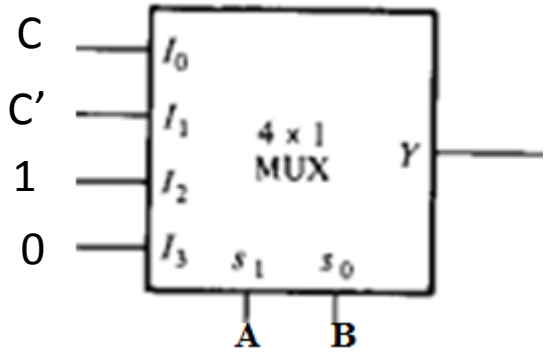

Rearrange equation with selection line

$$Y = \underbrace{0.(\overline{B}\,\overline{C})}_{I0} + \underbrace{\overline{B}C(\overline{A}+A)}_{I1} + \underbrace{AB\overline{C}}_{I2} + \underbrace{\overline{A}BC}_{I3}$$

Q. Y(A, B,C) = ∑m (1, 2, 4, 5) implement with 4:1 multiplexer

$$Y = \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\,\overline{C} + A\overline{B}C$$

Any 2 input (AB/BC/AC) select line




Rearrange equation with selection line

$$Y = \underbrace{\overline{A}\,\overline{B}C}_{I0} + \underbrace{\overline{A}B\overline{C}}_{I1} + \underbrace{A\overline{B}(\overline{C}+C)}_{I2} + \underbrace{AB.0}_{I3}$$

# Logic Gate implementation with 2:1 Multiplexer

## AND GATE

| A | B | OUT |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

OUT = 0 when A = 0

OUT = B when A = 1


## OR GATE

| A | B | OUT |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

OUT = B when A = 0

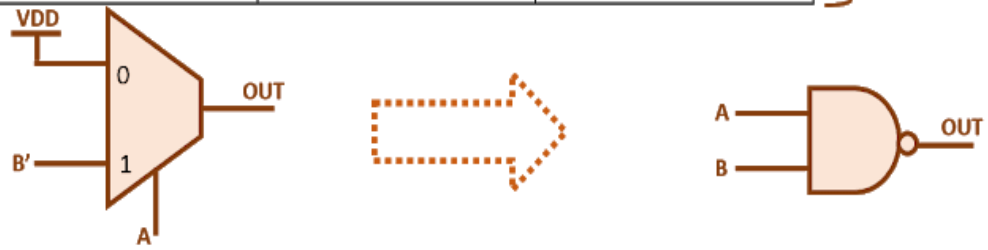
OUT = 1 when A = 1


## NAND GATE

| A | B | OUT |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

OUT = 1 when A = 0

OUT = B' when A = 1




## NOR GATE

| A | B | OUT |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

OUT = B' when A = 0

OUT = 0 when A = 1

# Logic Gate implementation with 2:1 Multiplexer

## XOR GATE

| A | B | OUT |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

OUT = B when A = 0

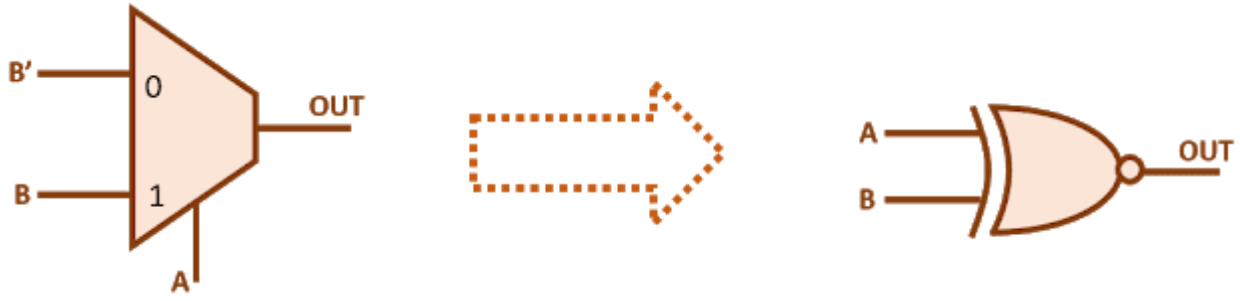
OUT = B' when A = 1


## XNOR GATE

| A | B | OUT |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

OUT = B' when A = 0

OUT = B when A = 1

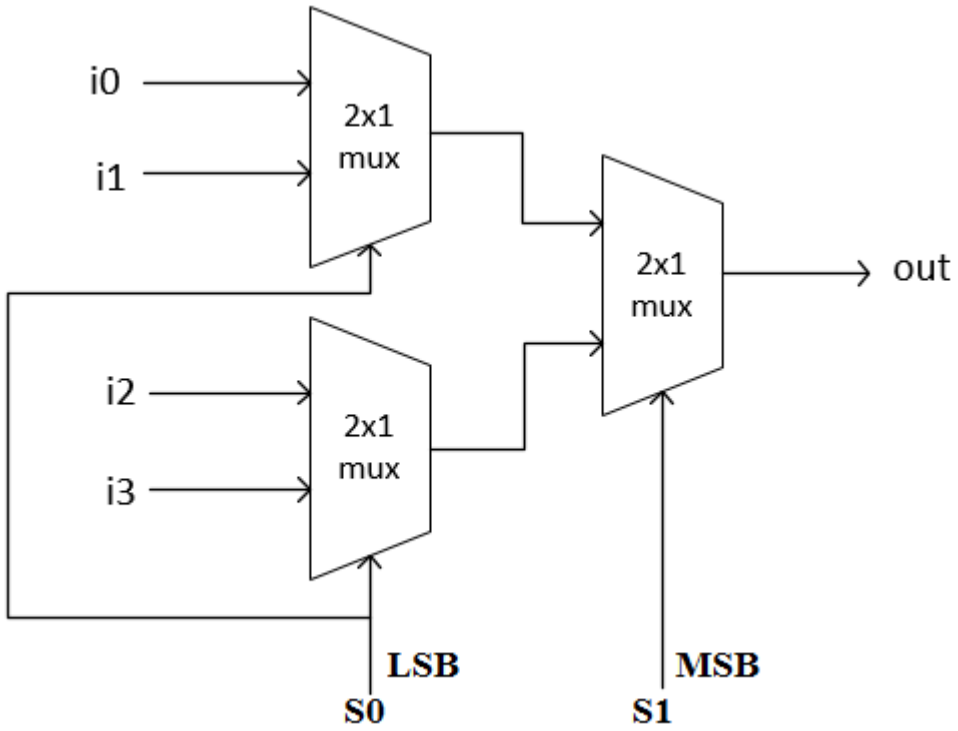



## NOT GATE

| A | OUT |
|---|---|
| 0 | 1 |
| 1 | 0 |

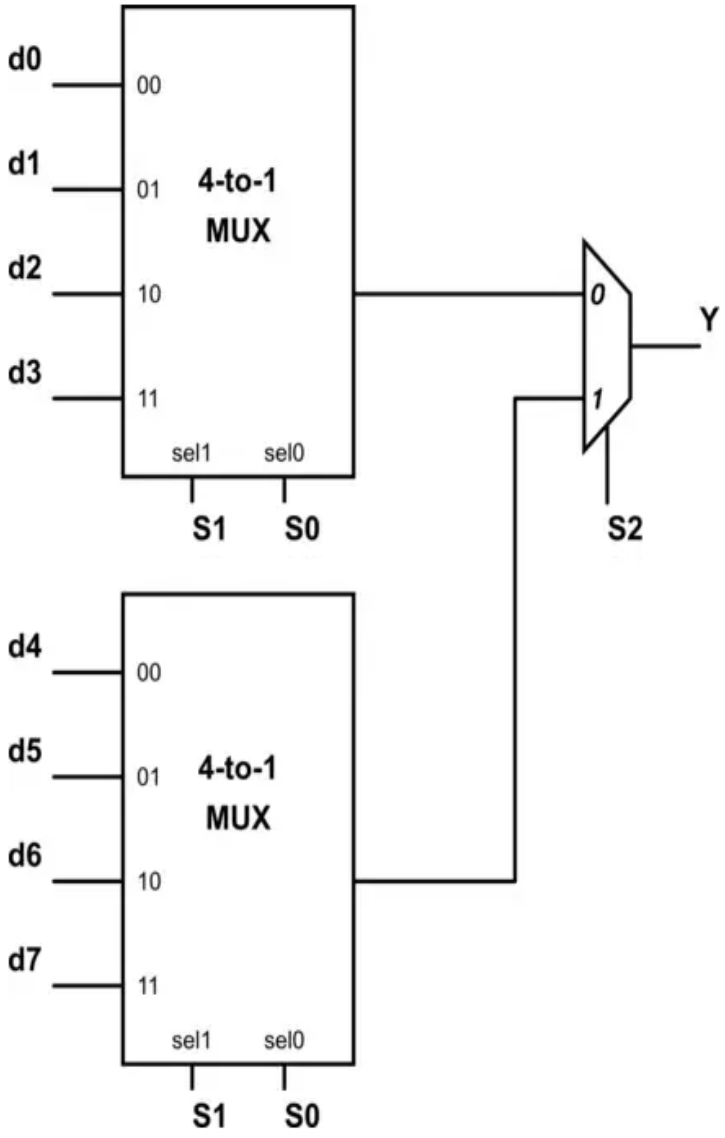
# 4:1 Mux using 2:1 Mux

| S1 (MSB) | S0(LSB) | Out |
|---|---|---|
| 0 | 0 | I0 |
| 0 | 1 | I1 |
| 1 | 0 | 12 |
| 1 | 1 | I3 |




# 8:1 Mux using 4:1 and 2:1 Mux

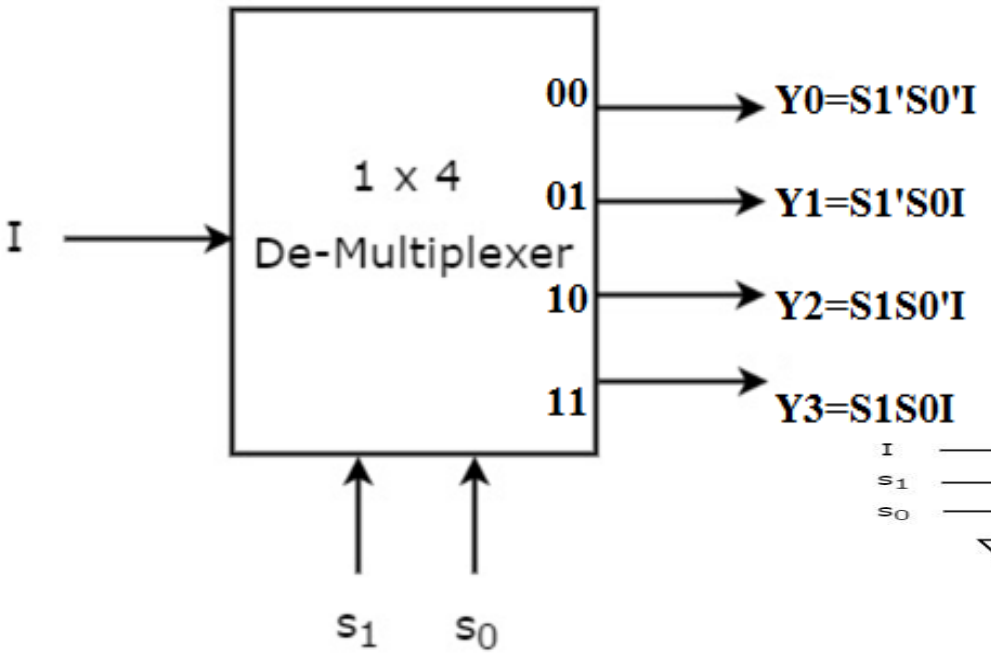
| Select Data Inputs | | | Output |
|---|---|---|---|
| $S_2$ | $S_1$ | $S_0$ | Y |
| 0 | 0 | 0 | $D_0$ |
| 0 | 0 | 1 | $D_1$ |
| 0 | 1 | 0 | $D_2$ |
| 0 | 1 | 1 | $D_3$ |
| 1 | 0 | 0 | $D_4$ |
| 1 | 0 | 1 | $D_5$ |
| 1 | 1 | 0 | $D_6$ |
| 1 | 1 | 1 | $D_7$ |

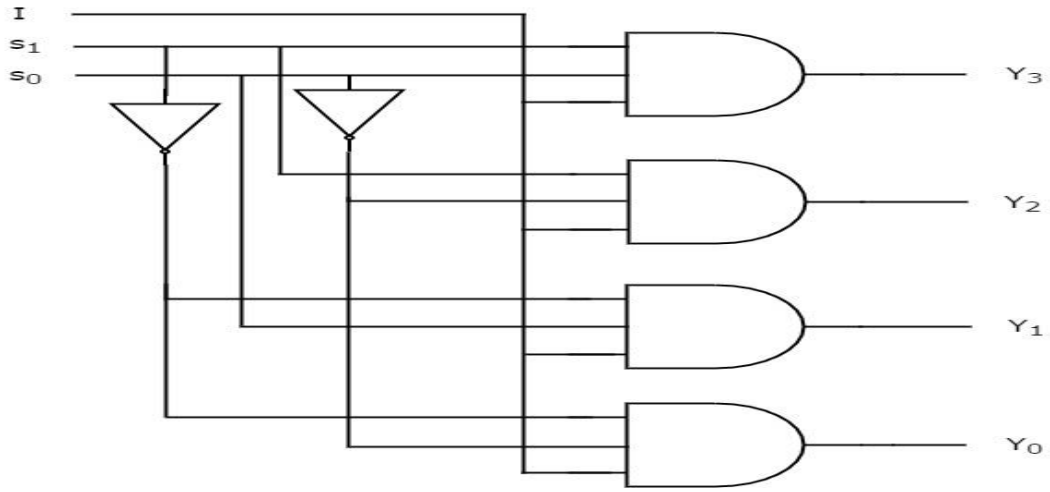


# De-multiplexer

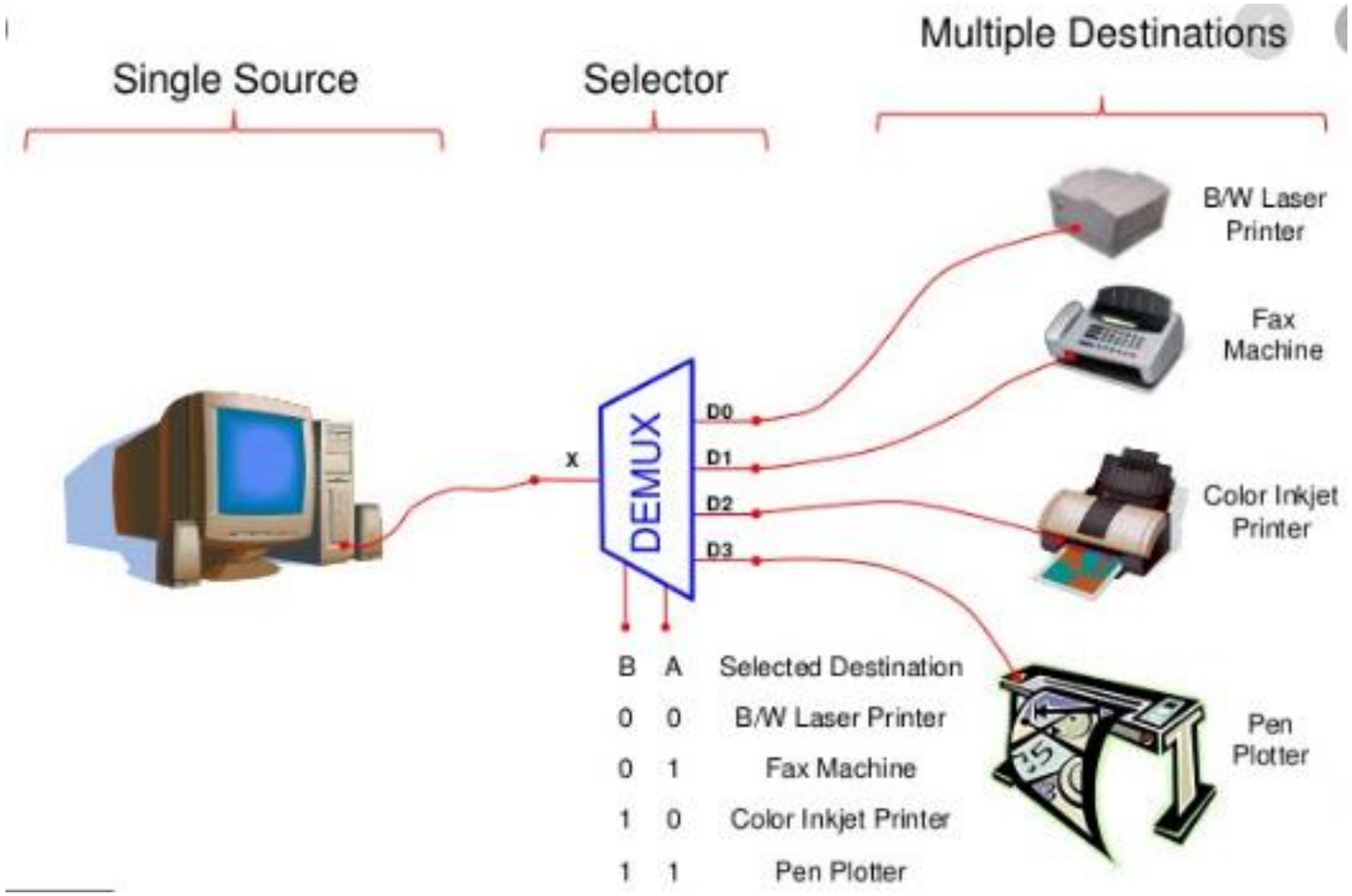
- Switch one common input line to one of several output line based on select input.
- It is data distributor.
- Size of demux $1{:}2^n$ Example 1:4, 1:8, 1:16….. Demultiplexer

| Selection Inputs | | Outputs | | | |
|---|---|---|---|---|---|
| $S_1$ | $S_0$ | $Y_3$ | $Y_2$ | $Y_1$ | $Y_0$ |
| 0 | 0 | 0 | 0 | 0 | I |
| 0 | 1 | 0 | 0 | I | 0 |
| 1 | 0 | 0 | I | 0 | 0 |
| 1 | 1 | I | 0 | 0 | 0 |



# Application

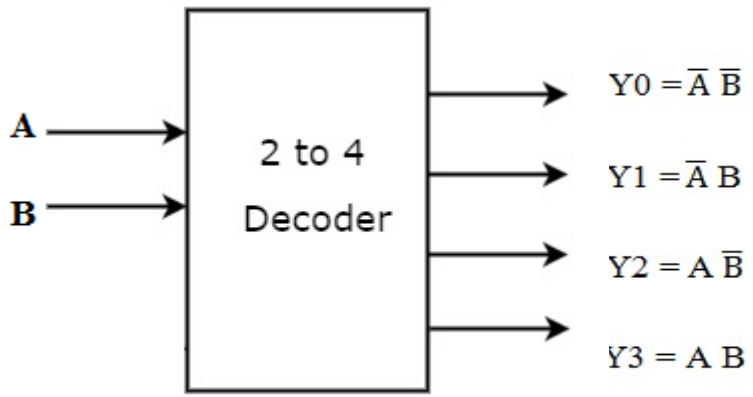


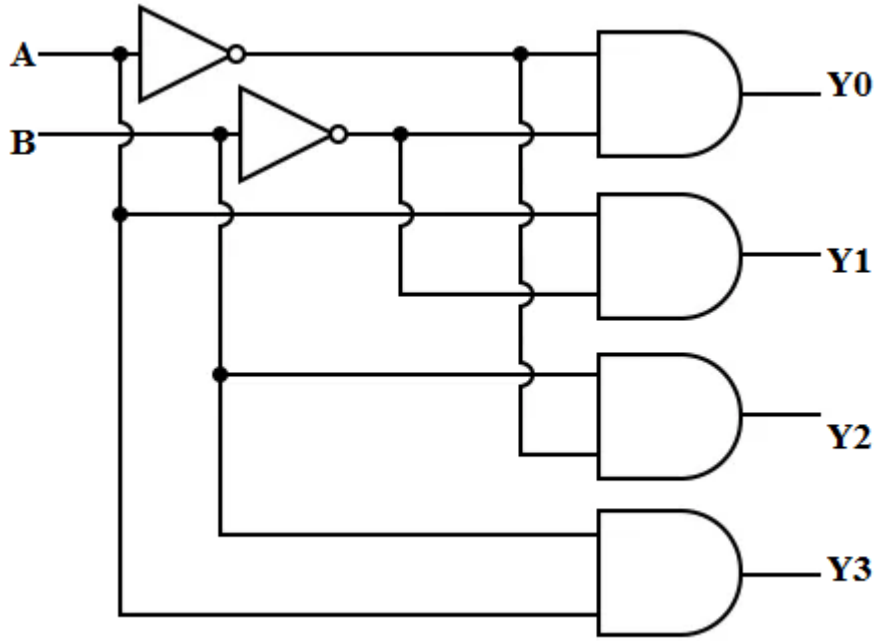
# Decoder

- A combinational circuit that has ‘n’ input lines and maximum of $2^n$ output lines.
- One of these outputs will be active High based on the combination of inputs present, when the decoder is enabled.

The outputs of the decoder are **min terms** of ‘n’ input variables lines when it is enabled

Size of Decoder are 2:4, 3:8, 4:16….




| $A_1$ | $A_0$ | $Y_3$ | $Y_2$ | $Y_1$ | $Y_0$ |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 |

# MCQ

Which one is decoder ?

(a) 4:1

(b) 1:4

(c) 3:8

(d) 8:3

# MCQ

How many inputs and outputs for four variable function in a decoder.

(a) 4,8

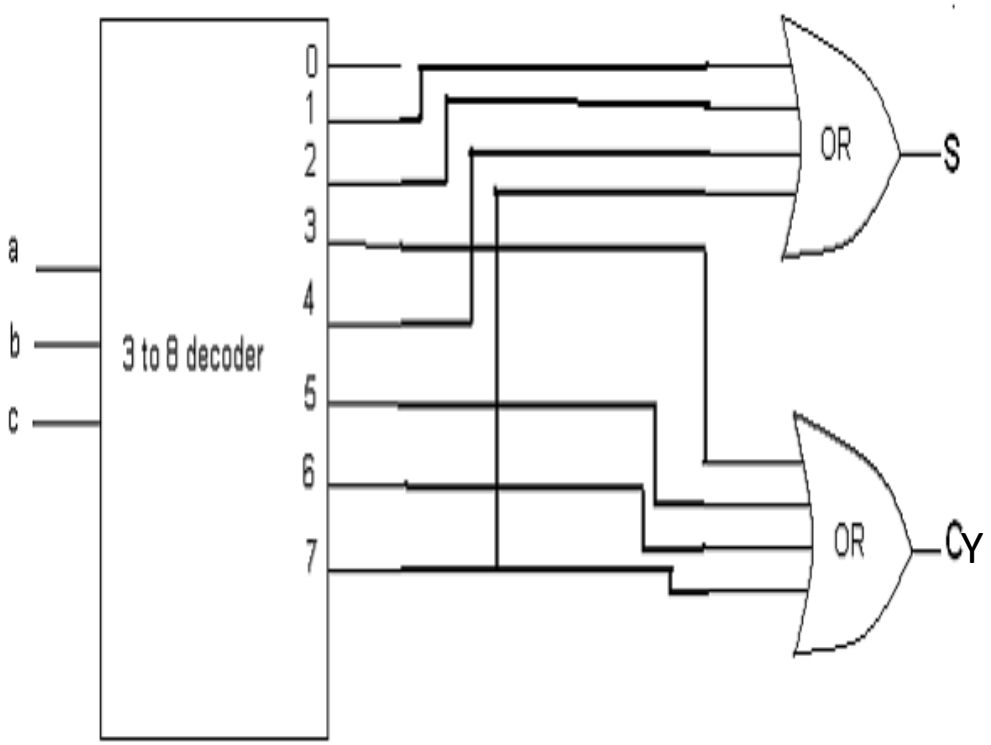
(b) 4,16

(c) 3,8

(d) 16, 1

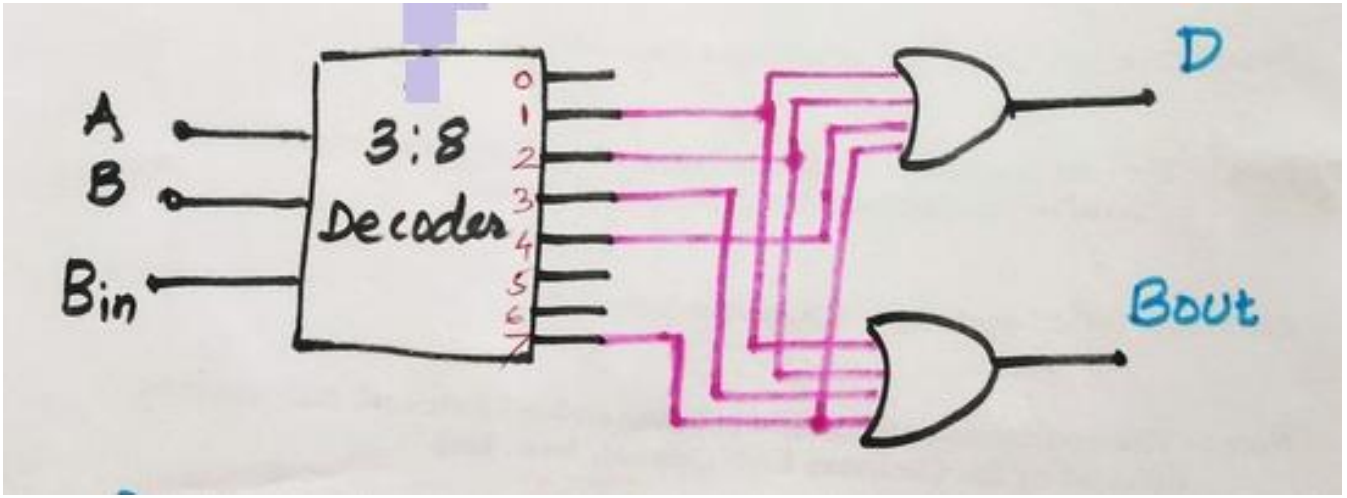
# Full Adder using 3:8 Decoder

S = A'B'C + A'BC' + AB'C' + ABC = Σ(1,2,4,7)
CY = A'BC +AB'C +ABC'+ABC= Σ (3, 5, 6, 7)



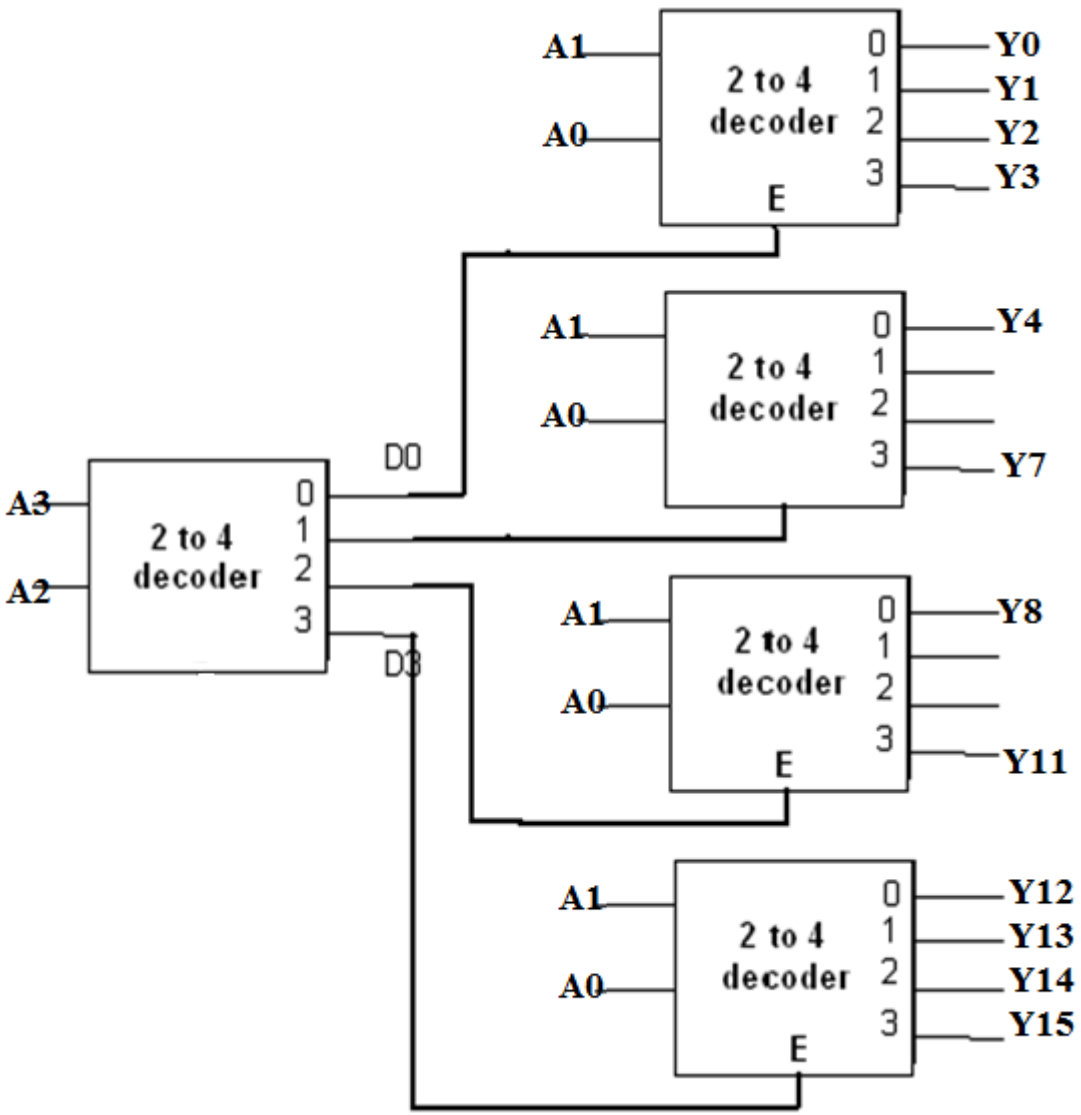

# Full Subtractor using 3:8 Decoder

DIFFERENCE = A'B'C + A'BC' + AB'C' + ABC = Σ(1,2,4,7)
BORROW = A'B'C +A'BC' +A'BC+ABC= Σ (1, 2, 3, 7)

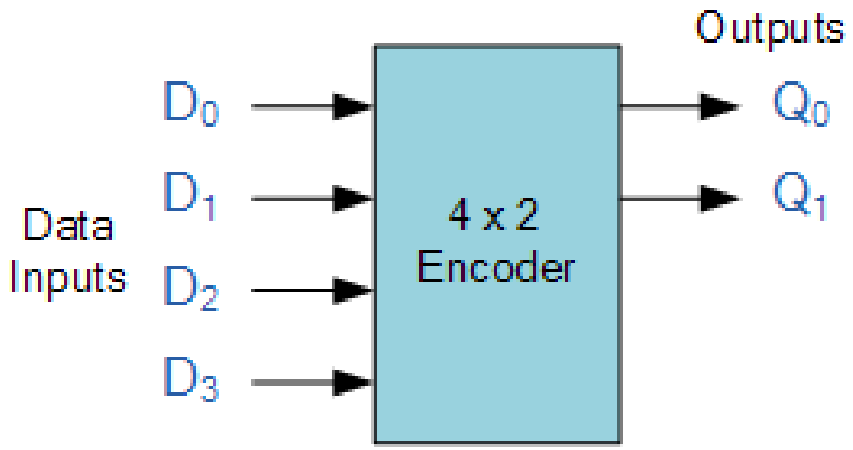


# 4:16 decoder using 2:4 decoder

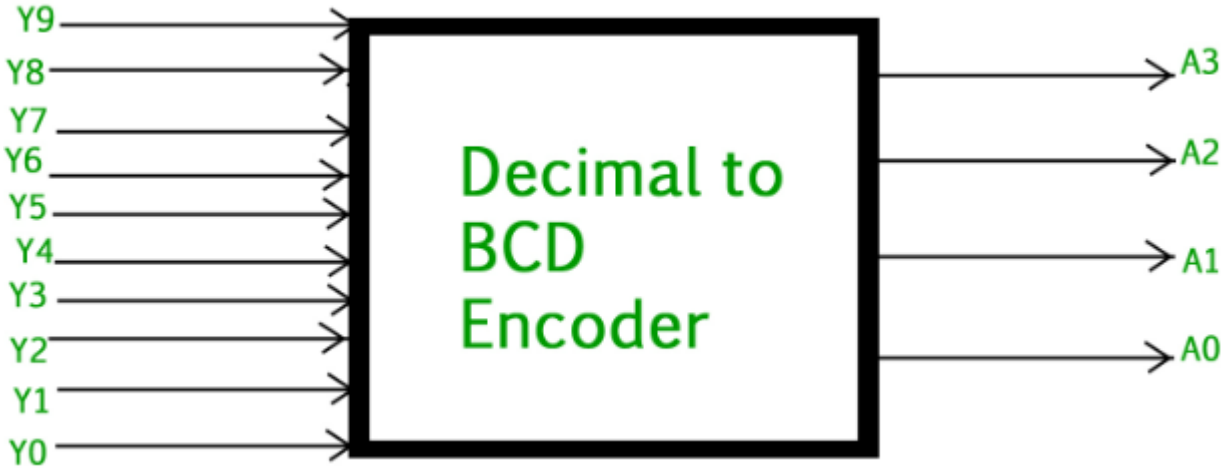
| A3 | A2 | A1 | A0 | HIGH OUTPUT |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | Y0 |
| 0 | 0 | 0 | 1 | Y1 |
| 0 | 0 | 1 | 0 | Y2 |
| 0 | 0 | 1 | 1 | Y3 |
| 0 | 1 | 0 | 0 | Y4 |
| 0 | 1 | 0 | 1 | Y5 |
| 0 | 1 | 1 | 0 | Y6 |
| 0 | 1 | 1 | 1 | Y7 |
| 1 | 0 | 0 | 0 | Y8 |
| 1 | 0 | 0 | 1 | Y9 |
| 1 | 0 | 1 | 0 | Y10 |
| 1 | 0 | 1 | 1 | Y11 |
| 1 | 1 | 0 | 0 | Y12 |
| 1 | 1 | 0 | 1 | Y13 |
| 1 | 1 | 1 | 0 | Y14 |
| 1 | 1 | 1 | 1 | Y15 |



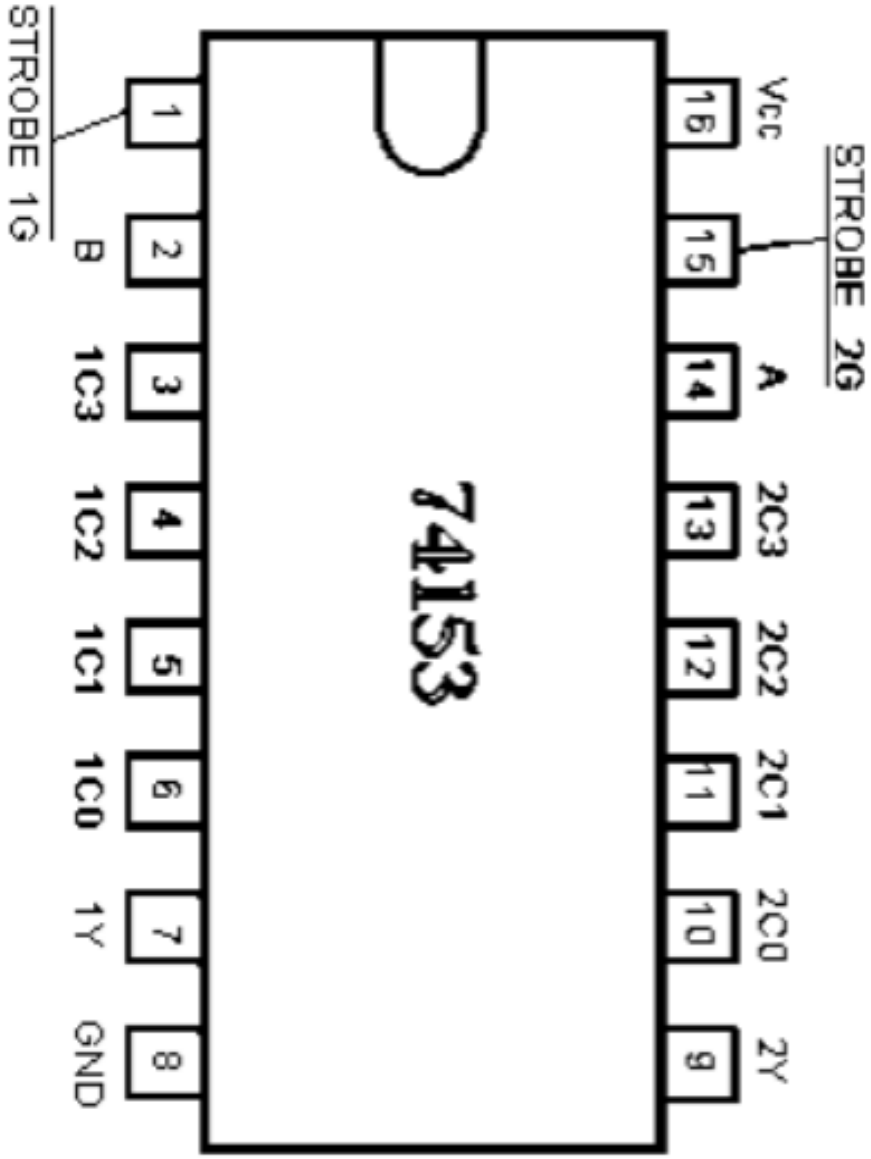
# Encoder



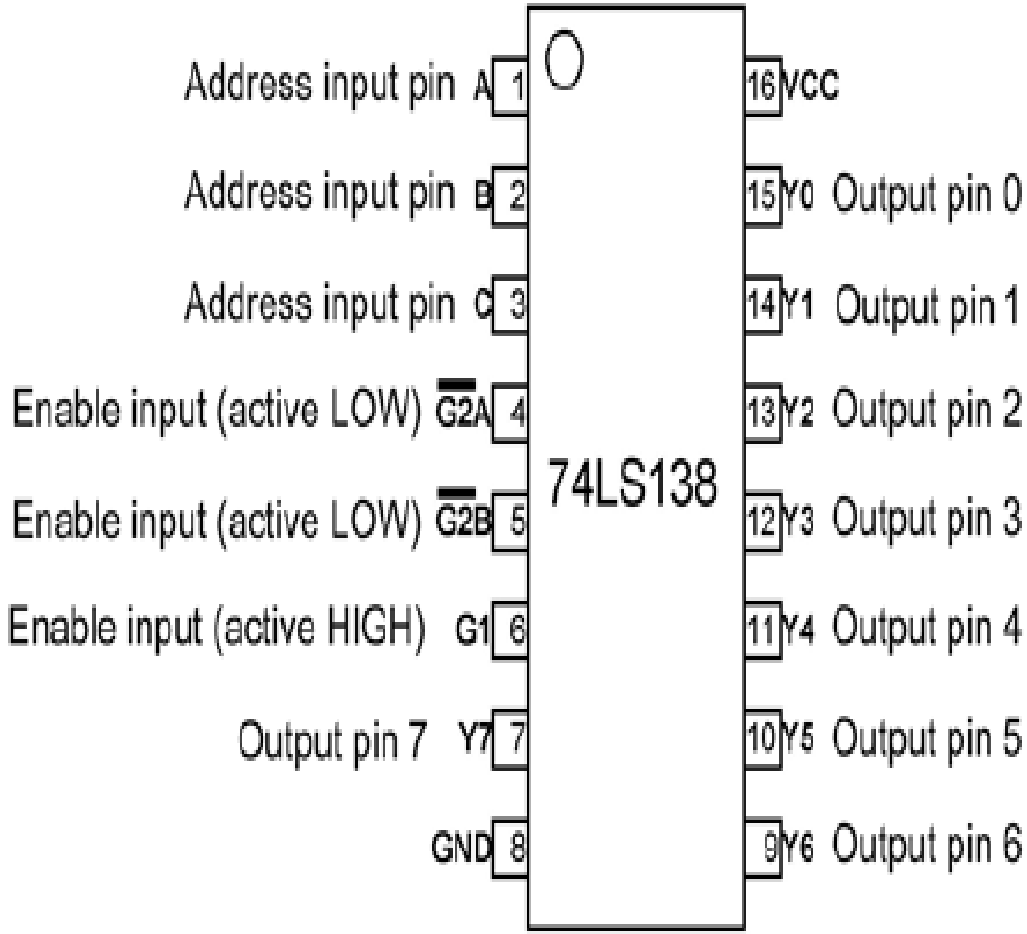

| Inputs | | | | Outputs | |
|---|---|---|---|---|---|
| $D_3$ | $D_2$ | $D_1$ | $D_0$ | $Q_1$ | $Q_0$ |
| 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | x | x |



## multiplexer




## Decoder



# MCQ

**Which of the following is not valid encoder?**

**(A)**
8 X 3

**(B)**
5 X 32

**(C)**
2 X 1

**(D)**
All are valid