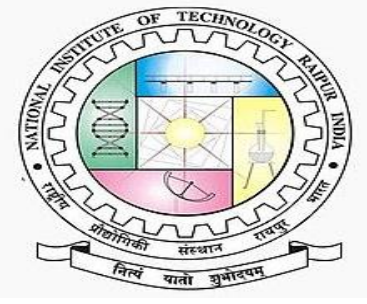
# Digital Logic and Design




Dr. Aakanksha Sharaff
Department of Computer Science and Engineering
National Institute of Technology Raipur

# Outlines

- Decoder
- Encoder
- Multiplexer
- De-multiplexer
- Modular Design using ICs

# Decoder

- A decoder is a combinational circuit.
- A decoder accepts a set of inputs that represents a binary number and activates only that output corresponding to the input number. All other outputs remain inactive.
- Fig. 1 shows the block diagram of decoder with ‘N’ inputs and ‘M’ outputs.
- There are $2^N$ possible input combinations, for each of these input combination only one output will be HIGH (active) all other outputs are LOW
- Some decoder have one or more ENABLE (E) inputs that are used to control the operation of decoder.

# Decoder

## BLOCK DIAGRAM OF DECODER

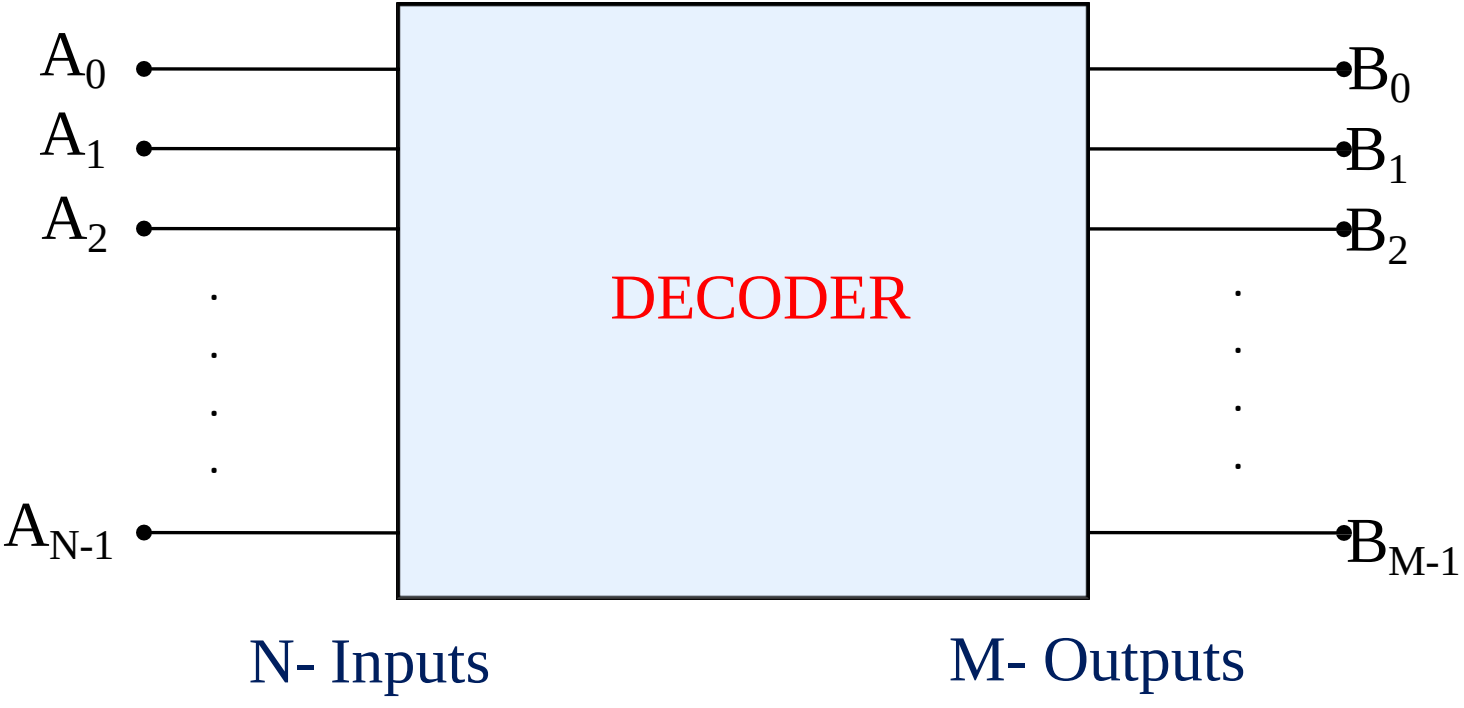



Fig. 1

*Only one output is High for each input*

# Decoder

2 to 4 Line Decoder:

- Block diagram of 2 to 4 decoder is shown in fig. 2
- A and B are the inputs. ( No. of inputs =2)
- No. of Outputs : $2^2$=4, they are indicated by $D_0$, $D_1$, $D_2$ and $D_3$
- From the Truth Table it is clear that each output is “1” for only specific combination of inputs.

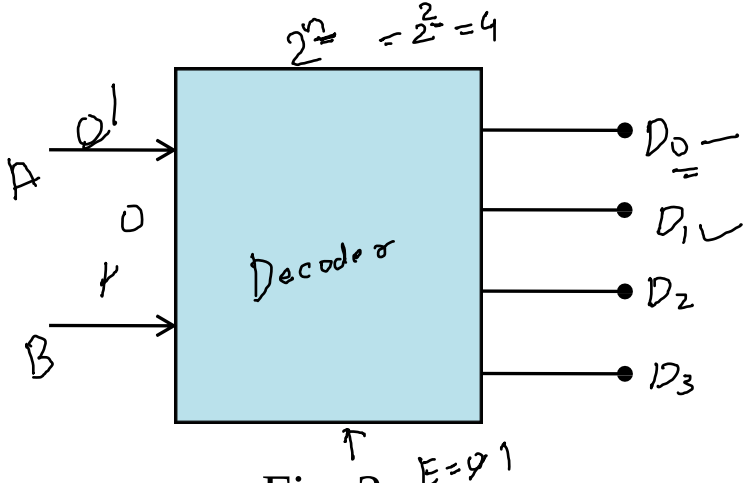



Fig. 2

| INPUTS | | OUTPUTS | | | |
|---|---|---|---|---|---|
| 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 |

TRUTH TABLE

# Decoder

BOOLEAN EXPRESSION:

From Truth Table

$$D_0 = \overline{A}\,\overline{B} \qquad D_1 = \overline{A}B$$

$$D_2 = A\,\overline{B} \qquad D_3 = AB$$

LOGIC DIAGRAM:

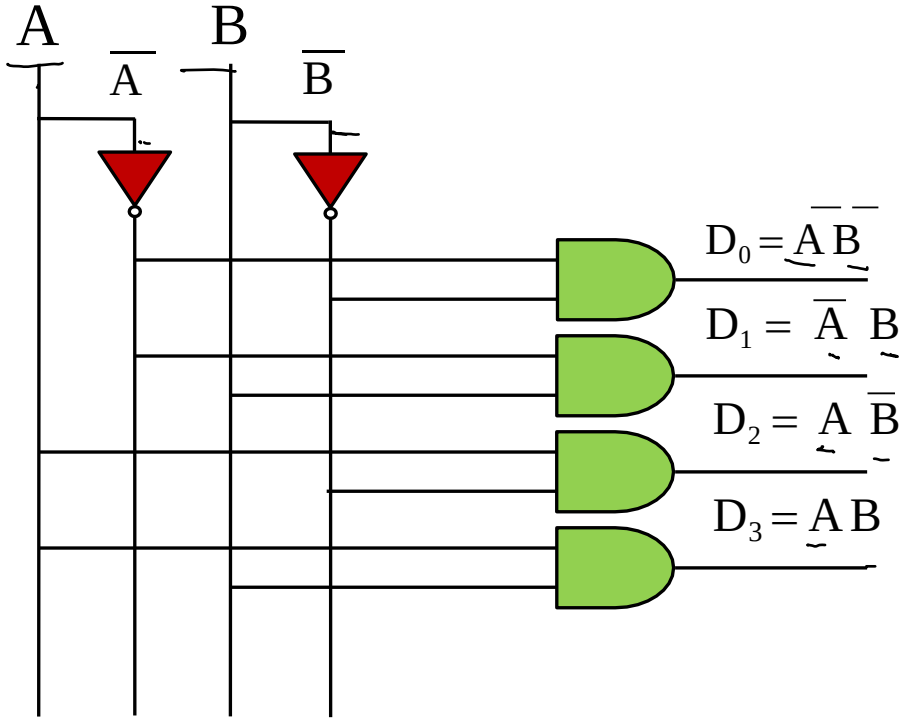



Fig. 3

# Decoder

- 3 to 8 Line Decoder/ 4 to 8 line Decoder
  - Block Diagram
  - Truth Table
  - Boolean Expression
  - Logic Circuit

# Encoder

- An Encoder is a combinational logic circuit.
- It performs the inverse operation of Decoder.
- The opposite process of decoding is known as Encoding.
- An Encoder converts an active input signal into a coded output signal.
- Block diagram of Encoder is shown in Fig.10. It has ‘M’ inputs and ‘N’ outputs.
- An Encoder has ‘M’ input lines, only one of which is activated at a given time, and produces an N-bit output code, depending on which input is activated.

# Encoder

## BLOCK DIAGRAM OF Encoder

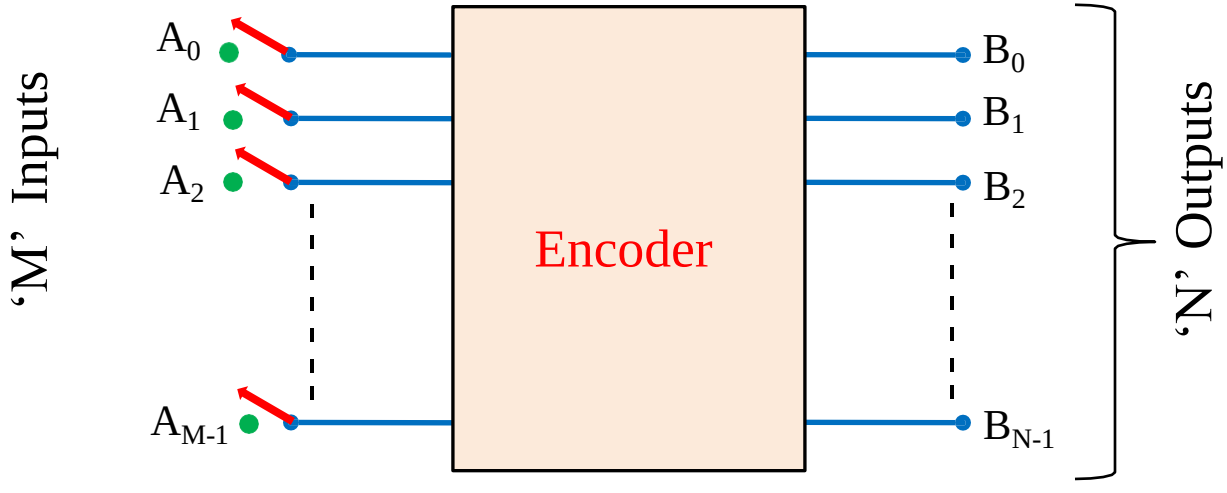



Fig. 4

# Encoder

4 to 2 Line encoder:

- Block diagram of 4 to 2 encoder is shown in fig. 2
- A ,B,C and D are the inputs. ( No. of inputs =4)
- No. of possible output combinations are $D_0$, and $D_1$ .
- No. of input : $2^2$=4, they are indicated by A,B,C and D.

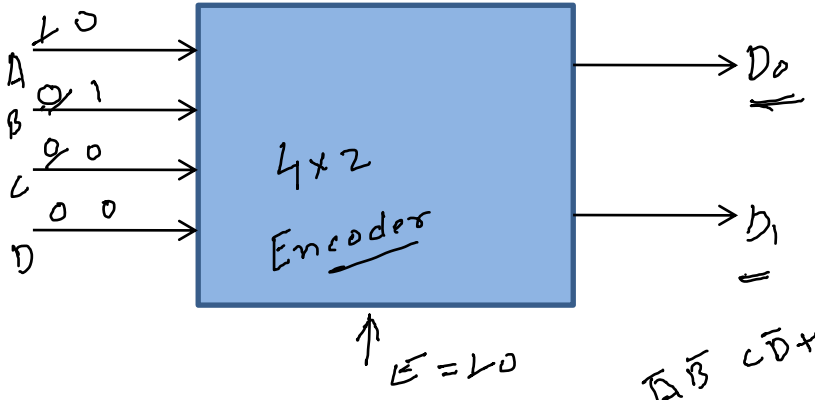



Fig. 5

TRUTH TABLE

| INPUTS | | | | OUTPUTS | |
|---|---|---|---|---|---|
| A | B | C | D | $D_0$ | $D_1$ |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 |

# Encoder

BOOLEAN EXPRESSION:

From Truth Table

$$D_0 = \overline{A}\,\overline{B}\,CD + A\overline{B}\overline{C}D$$

$$D_1 = \overline{A}\,B\,\overline{C}\,\overline{D} + A\overline{B}CD$$

LOGIC DIAGRAM:

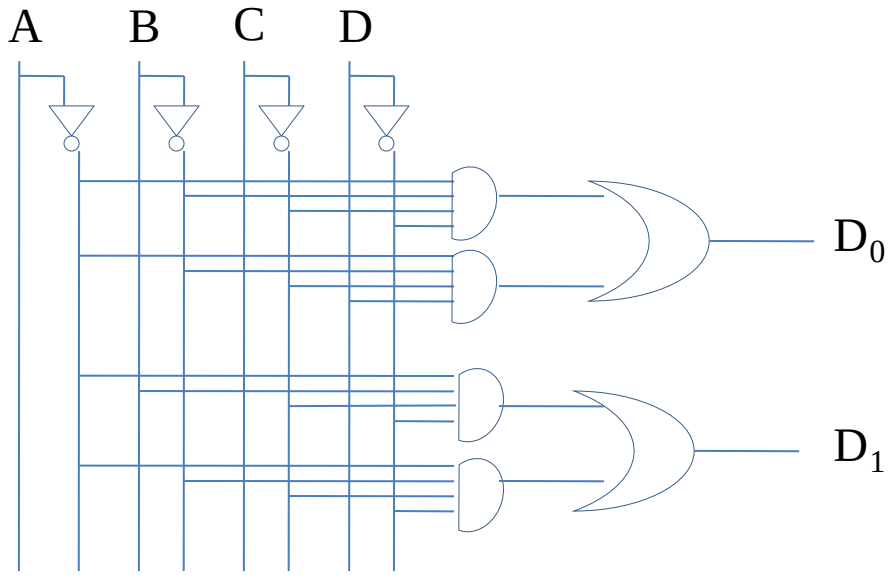



Fig. 6

# Encoder

- Encoders are used to translate the rotary or linear motion into a digital signal.
- The difference between Decoder and Encoder is that Decoder has Binary Code as an input while Encoder has Binary Code as an output.
- Encoder is an Electronics device that converts the analog signal to digital signal such as BCD Code.

- Types of Encoders

i. Priority Encoder
ii. Decimal to BCD Encoder
iii. Octal to Binary Encoder
iv. Hexadecimal to Binary Encoder

# Encoder

M=4

$M=2^2$

$M=2^N$

'M' is the input and

'N' is the output

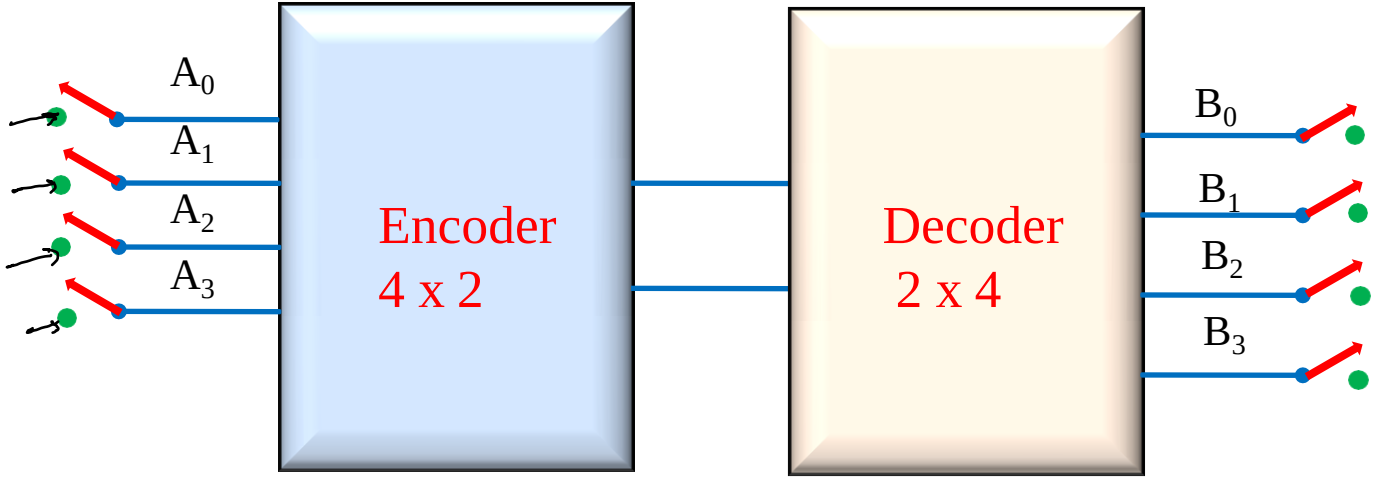



Fig.7

# Encoder

M=4

$M=2^2$

$M=2^N$

'M' is the input and

'N' is the output

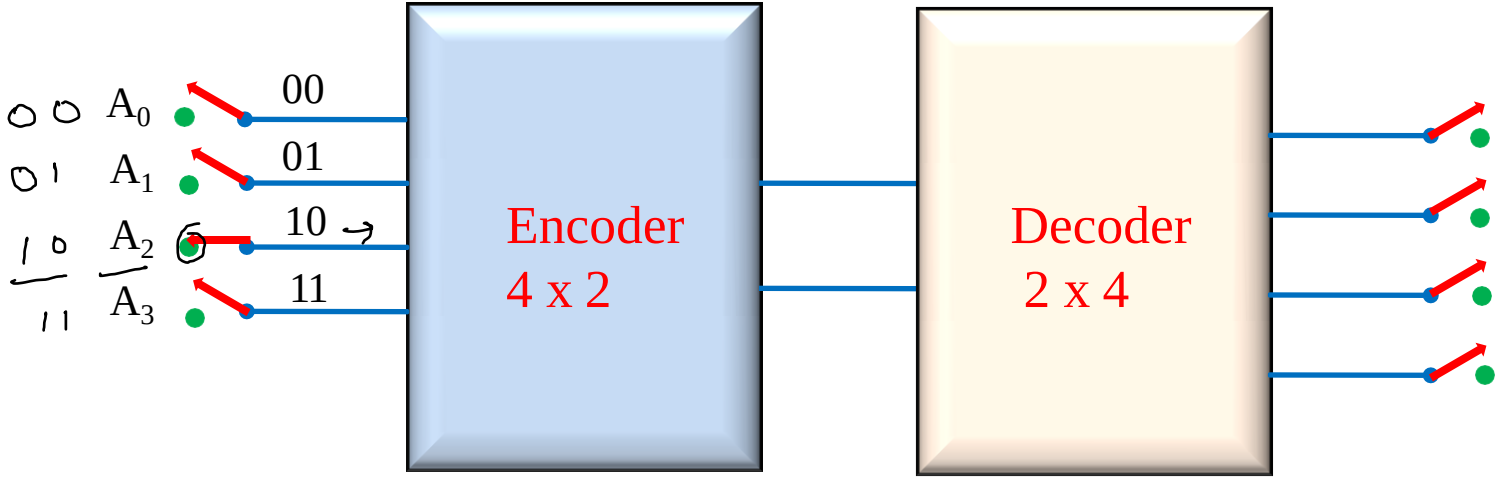



Fig. 8

# Encoder

M=4

$M=2^2$

$M=2^N$

'M' is the input and

'N' is the output

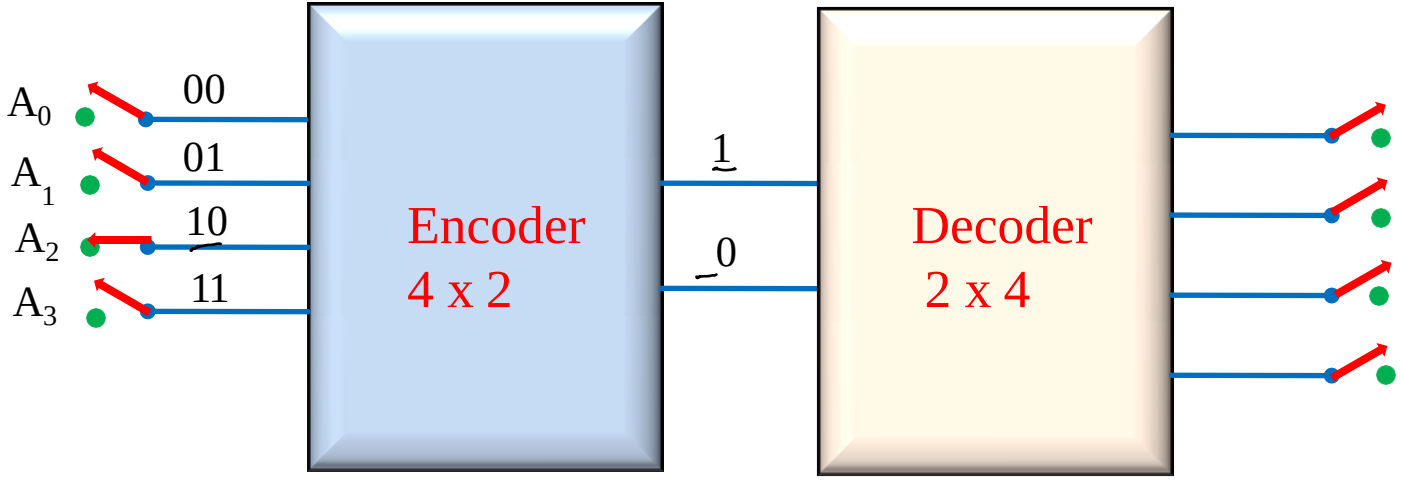



Fig. 9

# Encoder

M=4

$M=2^2$

$M=2^N$

'M' is the input and

'N' is the output

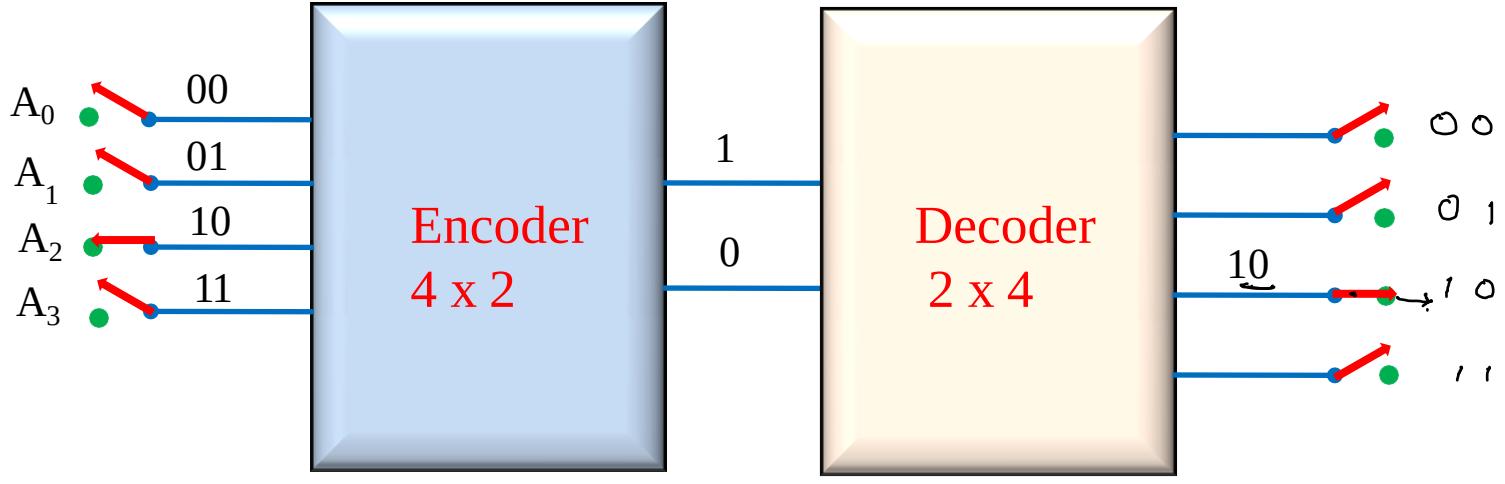
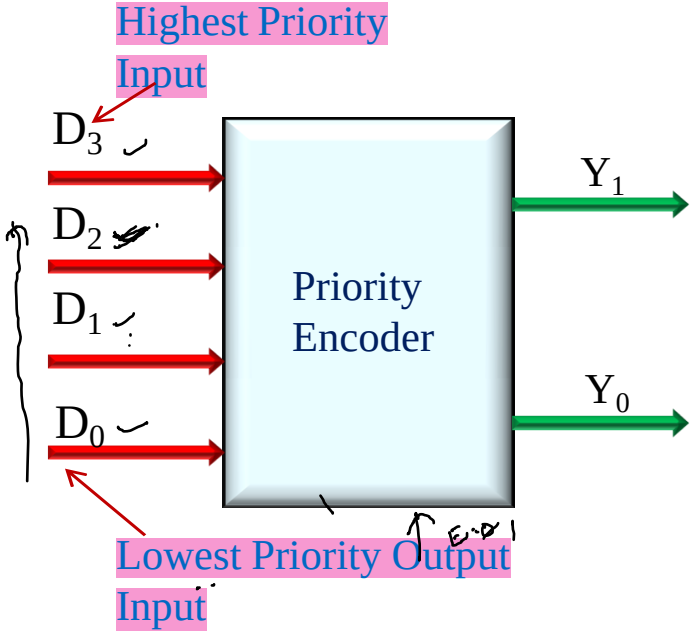



Fig. 10

# Encoder

## PRIORITY ENCODER:

- As the name indicates, the priority is given to inputs line.
- If two or more input lines are high at the same time i.e 1 at the same time, then the input line with high priority shall be considered.
- Block diagram and Truth table of Priority Encoder are shown in fig.11




Block Diagram of Priority Encoder

Fig.11

TRUTH TABLE:

| INPUTS | | | | OUTPUTS | | V |
|---|---|---|---|---|---|---|
| $D_3$ | $D_2$ | $D_1$ | $D_0$ | $Y_1$ | $Y_0$ | |
| 0 | 0 | 0 | 0 | x | x | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | x | 0 | 1 | 1 |
| 0 | 1 | x | x | 1 | 0 | 1 |
| 1 | x | x | x | 1 | 1 | 1 |

# Encoder

LOGIC DIAGRAM OF PRIORITY ENCODER:

$Y_1 = D_2 + \overline{D_3}$

$Y_0 = D_3 + \overline{D_2} D_1$

Logic ckt.

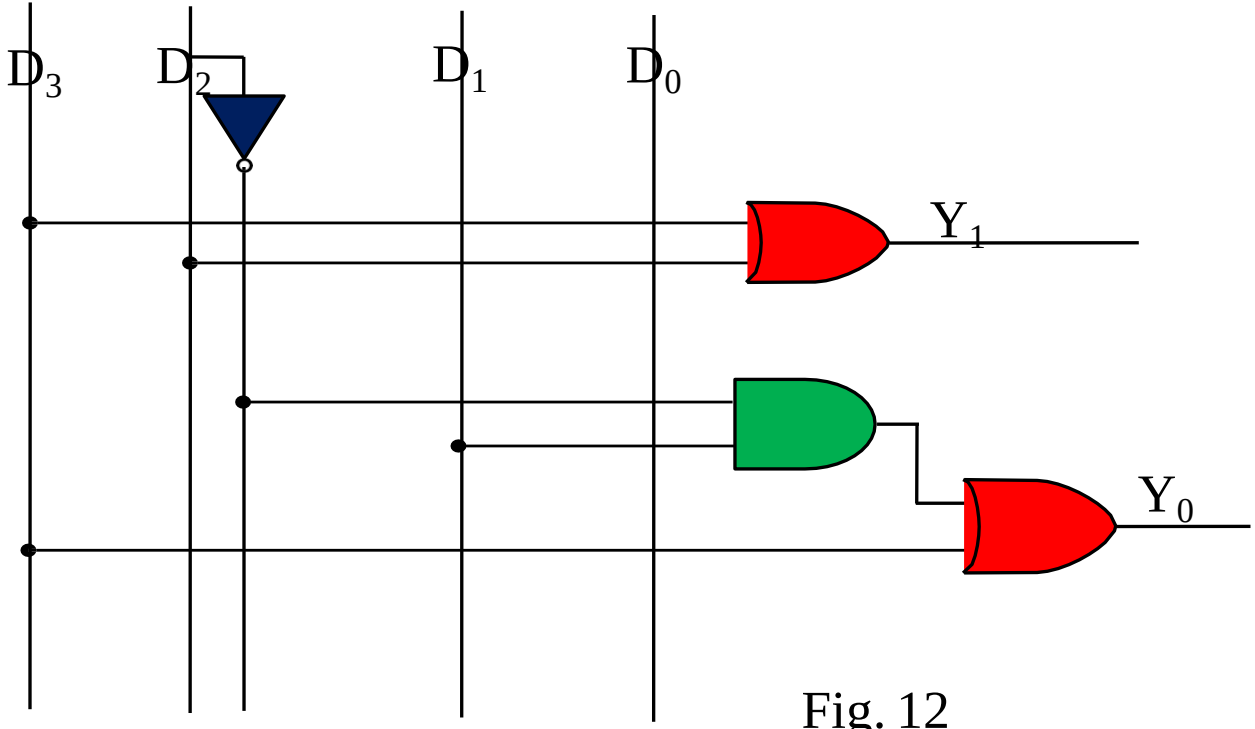



Fig. 12

# DECIMAL TO BCD ENCODER

- It has ten inputs corresponding to ten decimal digits (from 0 to 9) and four outputs (A,B,C,D) representing the BCD.
- The block diagram is shown in fig.13.

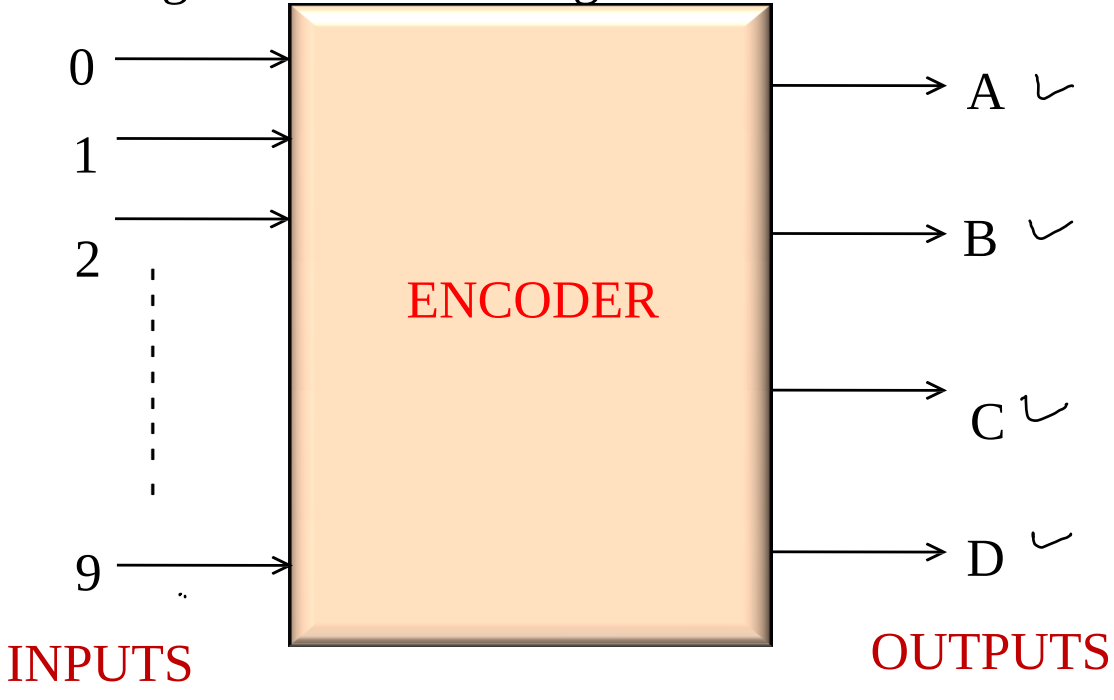



Fig. 13

# DECIMAL TO BCD ENCODER

## Truth Table

| INPUTS | | | | | | | | | | BCD OUTPUTS | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |

- From Truth Table it is clear that the output Ais HIGH when input is 8 OR 9 is HIGH

  Therefore A=8+9
- The output B is HIGH when 4 OR 5 OR 6 OR 7 is HIGH

  Therefore B=4+5+6+7
- The output C is HIGH when 2 OR 3 OR 6 OR 7 is HIGH

  Therefore C=2+3+6+7
- Similarly D=1+3+5+7+9

  Logic Diagram is shown in fig.20

DECIMAL TO BCD ENCODER

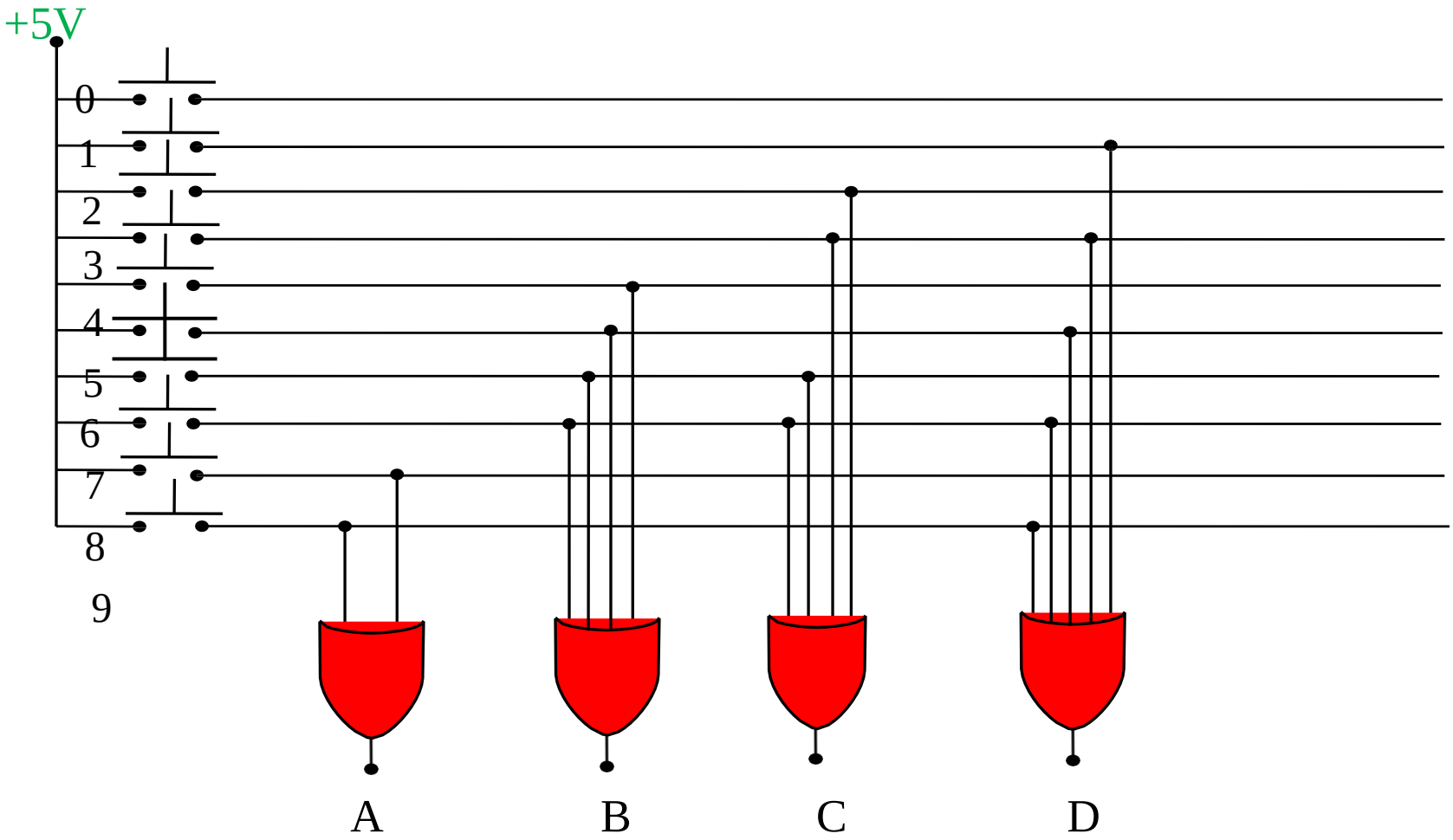



Fig. 14

# OCTAL TO BINARY ENCODER:

- Block Diagram of Octal to Binary Encoder is shown in Fig. 21
- It has eight inputs and three outputs.
- Only one input has one value at any given time.
- Each input corresponds to each octal digit and output generates corresponding Binary Code.

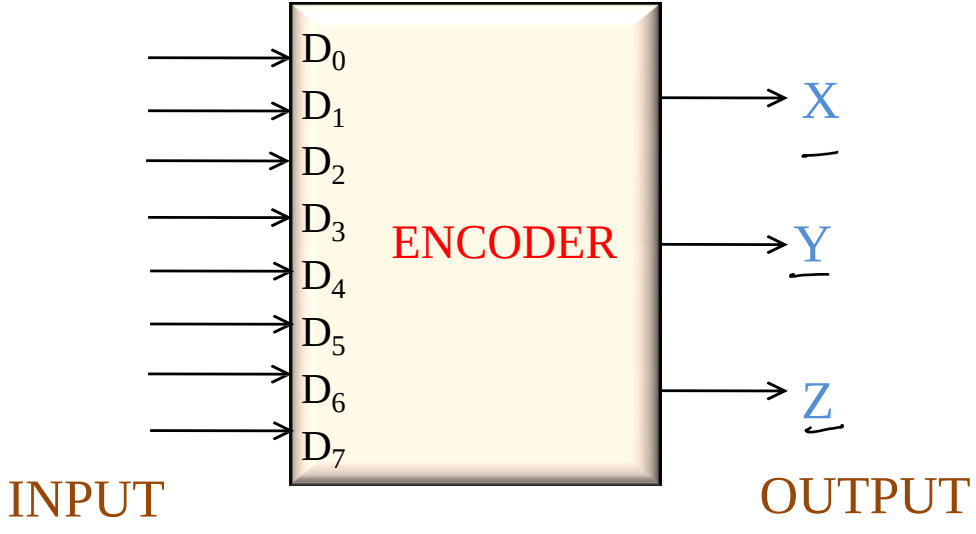



Fig. 15

TRUTH TABLE:

| INPUT | | | | | | | | OUTPUT | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $D_0$ | $D_1$ | $D_2$ | $D_3$ | $D_4$ | $D_5$ | $D_6$ | $D_7$ | X | Y | Z |
| 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 |

From Truth table:

$$X = D_4 + D_5 + D_6 + D_7$$

$$Y = D_2 + D_3 + D_6 + D_7$$

$$Z = D_1 + D_3 + D_5 + D_7$$

- It is assume that only one input is HIGH at any given time. If two outputs are HIGH then undefined output will produced. For example $D_3$ and $D_6$ are HIGH, then output of Encoder will be 111. This output neither equivalent code corresponding to $D_3$ nor to $D_6$.
- To overcome this problem, priorities should be assigned to each input.
- Form the truth table it is clear that the output X becomes 1 if any of the digit $D_4$ or $D_5$ or $D_6$ or $D_7$ is 1.
- $D_0$ is considered as don't care because it is not shown in expression.
- If inputs are zero then output will be zero. Similarly if $D_0$ is one, the output will be zero.

$$X = D_4 + D_5 + D_6 + D_7$$
$$Y = D_2 + D_3 + D_6 + D_7$$
$$Z = D_1 + D_3 + D_5 + D_7$$

Logic expression

## LOGIC DIAGRAM:

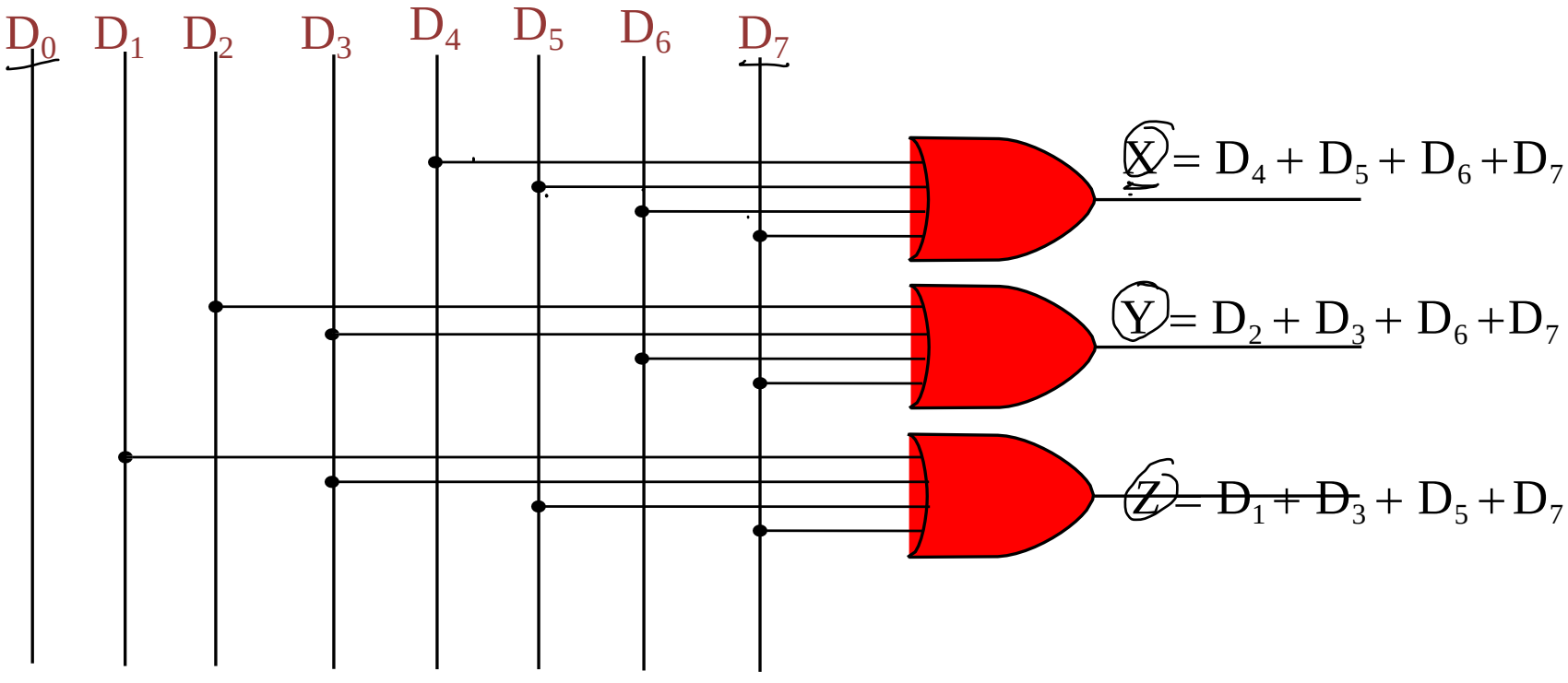



Fig. 16

# Encoder

- 8 x 3 Line Encoder/ 16 to 4 line ~~Decoder~~ Encoder
  - Block Diagram
  - Truth Table
  - Boolean Expression
  - Logic Circuit

# Multiplexer

- Definition : A multiplexers (MUX) is a device that allows digital information from several sources to be routed onto a single line for transmission over that line to a common destination.
- Several data input lines Some select line (less than the no. of input lines) Single output line.
- If there are n data input lines and m select lines, then $2^m = n$

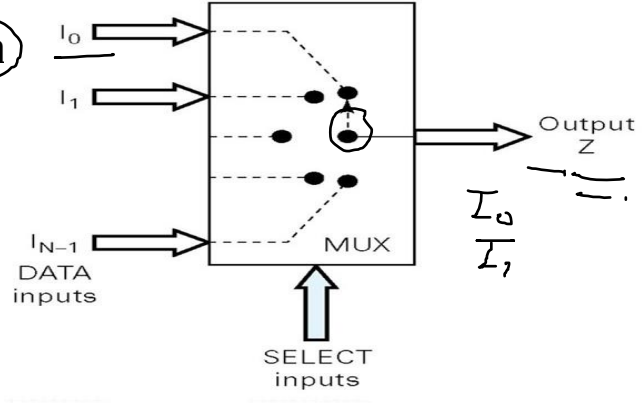



Fig. 17

# Multiplexer

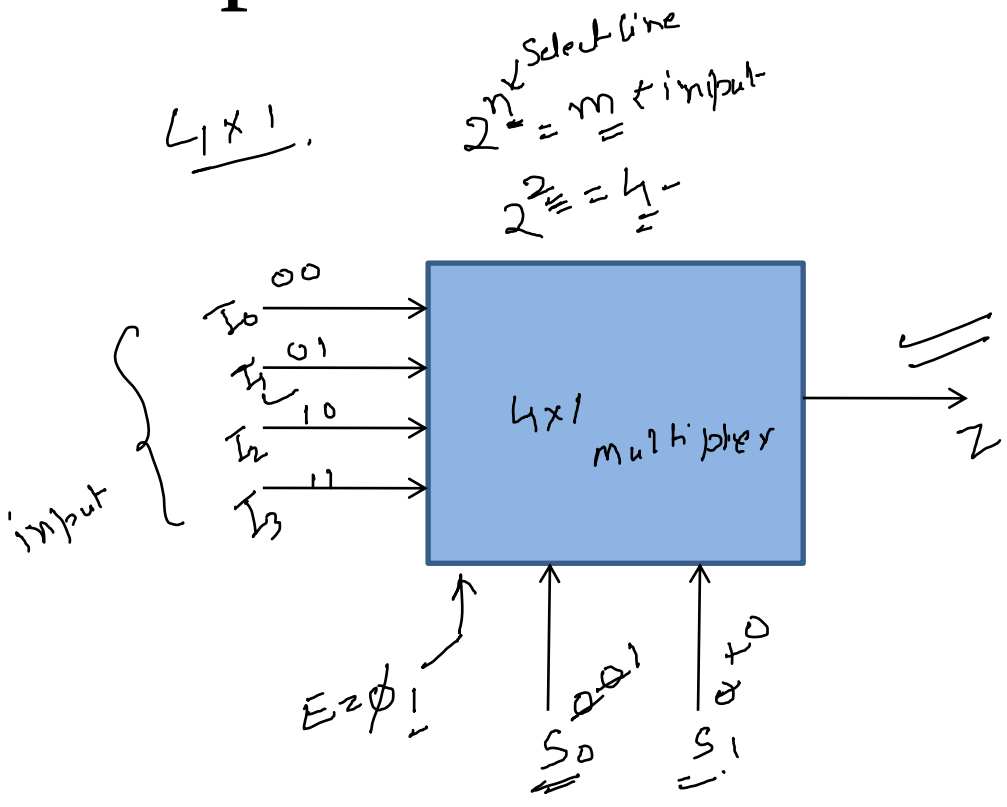



Fig. 18

TRUTH TABLE:

| $S_0$ | $S_1$ | Z |
|---|---|---|
| 0 | 0 | $I_0$ |
| 0 | 1 | $I_1$ |
| 1 | 0 | $I_2$ |
| 1 | 1 | $I_3$ |

# Multiplexer

4x1 mux

LOGIC DIAGRAM

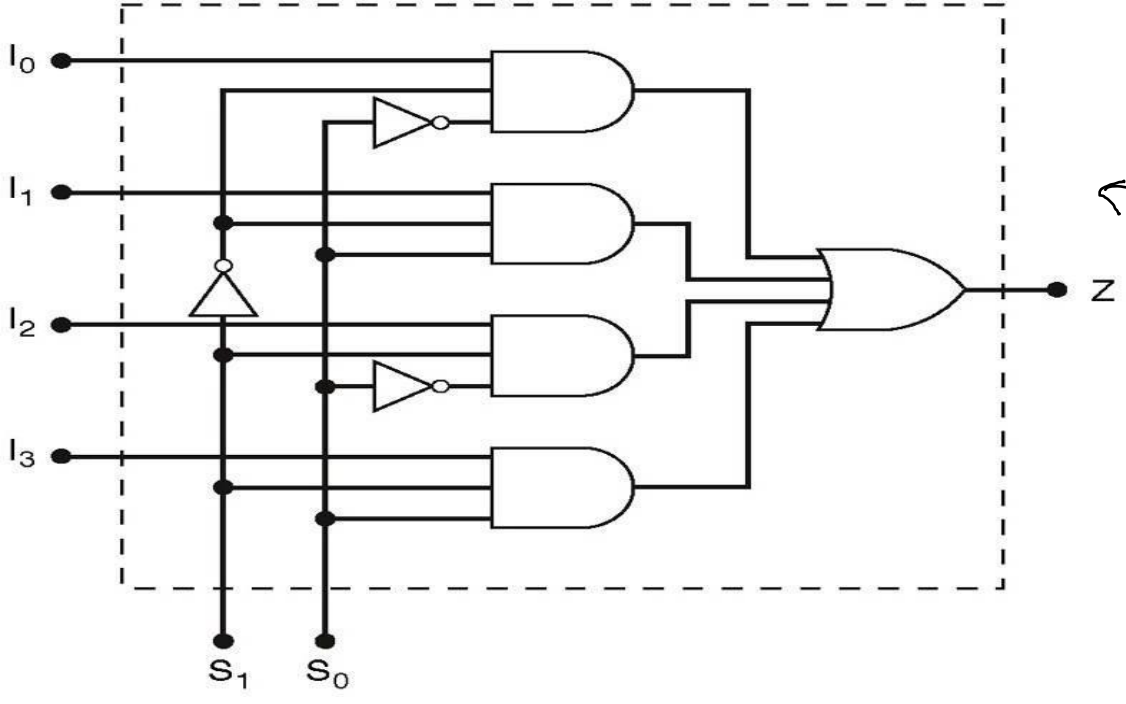



Fig. 19

TRUTH TABLE:

| $S_0$ | $S_1$ | Z |
|---|---|---|
| 0 | 0 | $I_0$ |
| 0 | 1 | $I_1$ |
| 1 | 0 | $I_2$ |
| 1 | 1 | $I_3$ |

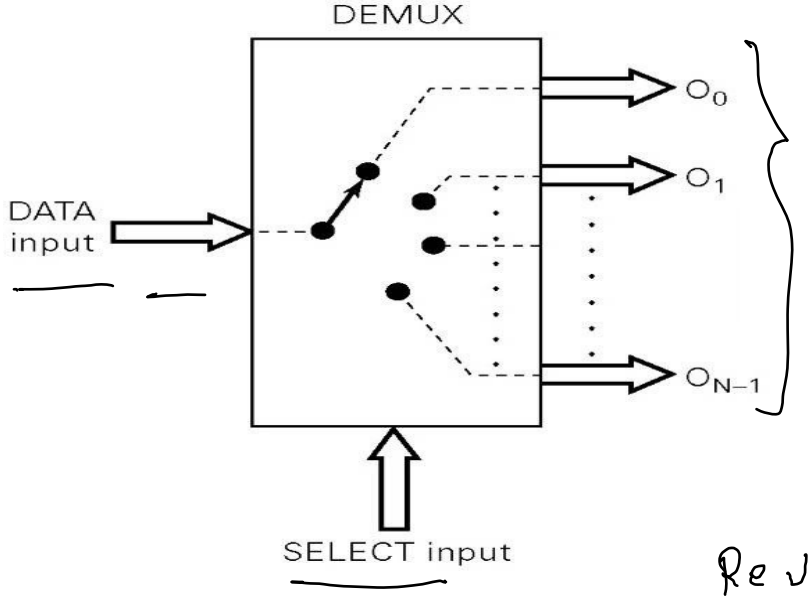
# Multiplexer

- 8 x 1 multiplexer/ 16 x 1 line multiplexer
  - Block Diagram
  - Truth Table
  - Boolean Expression
  - Logic Circuit

# De-multiplexer

- Definition: A DEMULTIPLEXER (DEMUX) basically reverses the multiplexing function. It take data from one line and distributes them to a given number of output lines. For this reason, the demultiplexers is also known as a data distributor.
- Single data input lines Some select line (less than the no. of output lines) several output line
- If there are n data output lines and m select lines, then $2^m = n$

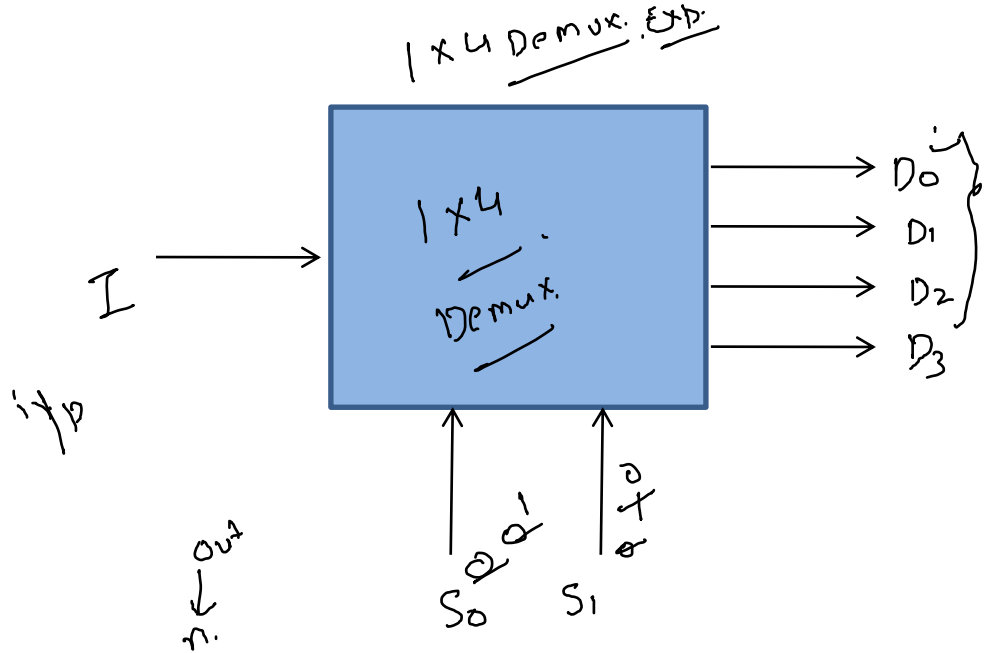
# De-multiplexer




Reverse process of mux.

Fig. 20

# De-multiplexer




TRUTH TABLE:

| I | $S_0$ | $S_1$ | Output | $D_1$ | $D_2$ | $D_3$ |
|---|---|---|---|---|---|---|
| 1 | 0 | 0 | $D_0$ | | | |
| 1 | 0 | 1 | $D_1$ | | | |
| 1 | 1 | 0 | $D_2$ | | | |
| 1 | 1 | 1 | $D_3$ | | | |

Fig. 21

# De-multiplexer

## 1x4 Demultiplexer

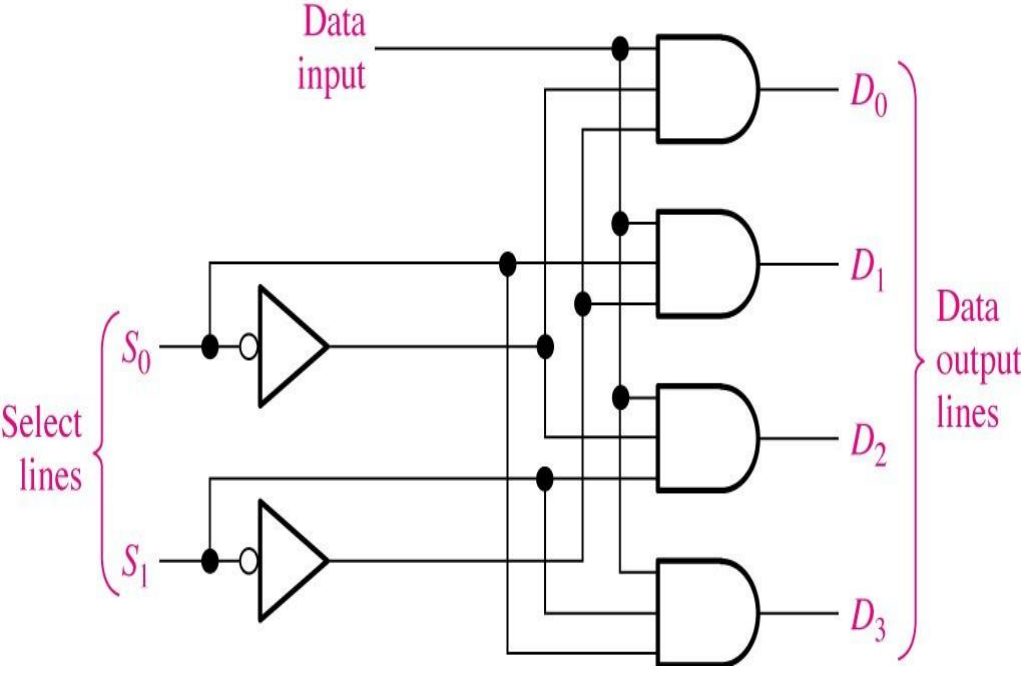



Fig. 22

TRUTH TABLE:

| $S_0$ | $S_1$ | $D_0$ | $D_1$ | $D_2$ | $D_3$ |
|---|---|---|---|---|---|

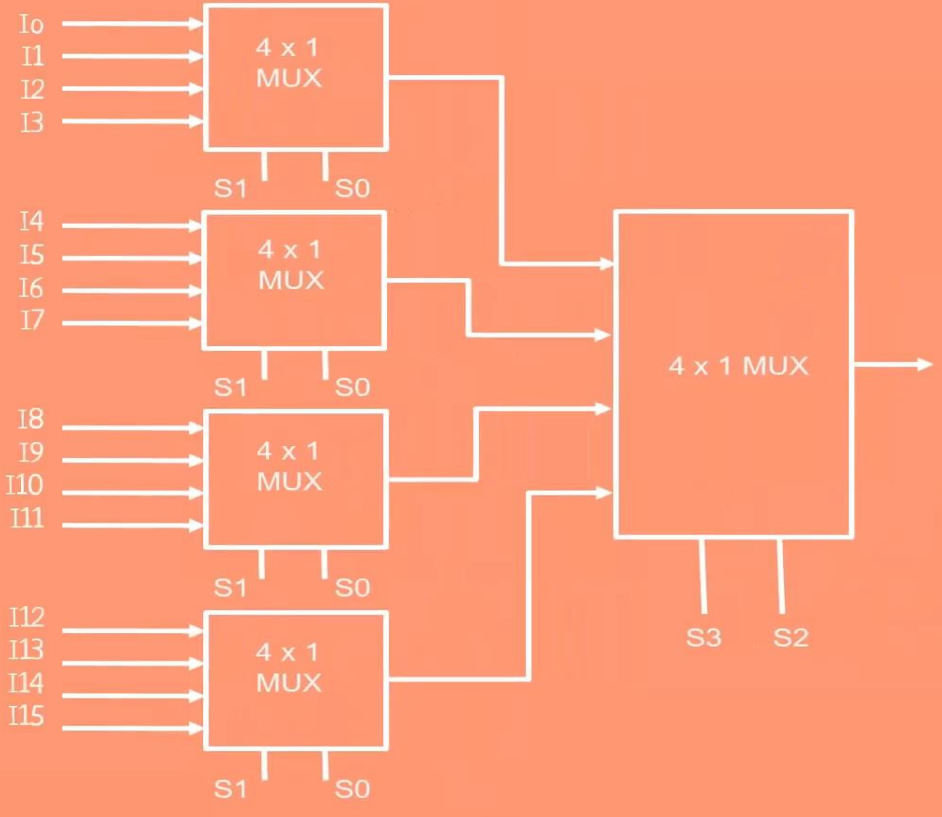
# Demultiplexer

- 1 x 8  demultiplexer/ 1 x 16 line multiplexer
  - Block Diagram
  - Truth Table
  - Boolean Expression
  - Logic Circuit

# Modular Design using ICs

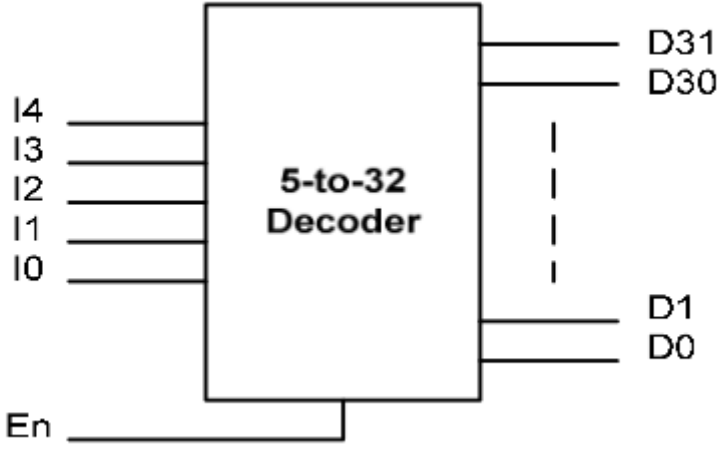
- **Positional Encoders**
- Another more common application is in magnetic positional control as used on ships navigation or for robotic arm positioning etc. Here for example, the angular or rotary position of a compass is converted into a digital code by a 74LS148 8-to-3 line priority encoder and input to the systems computer to provide navigational data and an example of a simple 8 position to 3-bit output compass encoder is shown below. Magnets and reed switches could be used at each compass point to indicate the needles angular position.

# Modular Design using Ics (16x1 mux using 4 x1 mux)

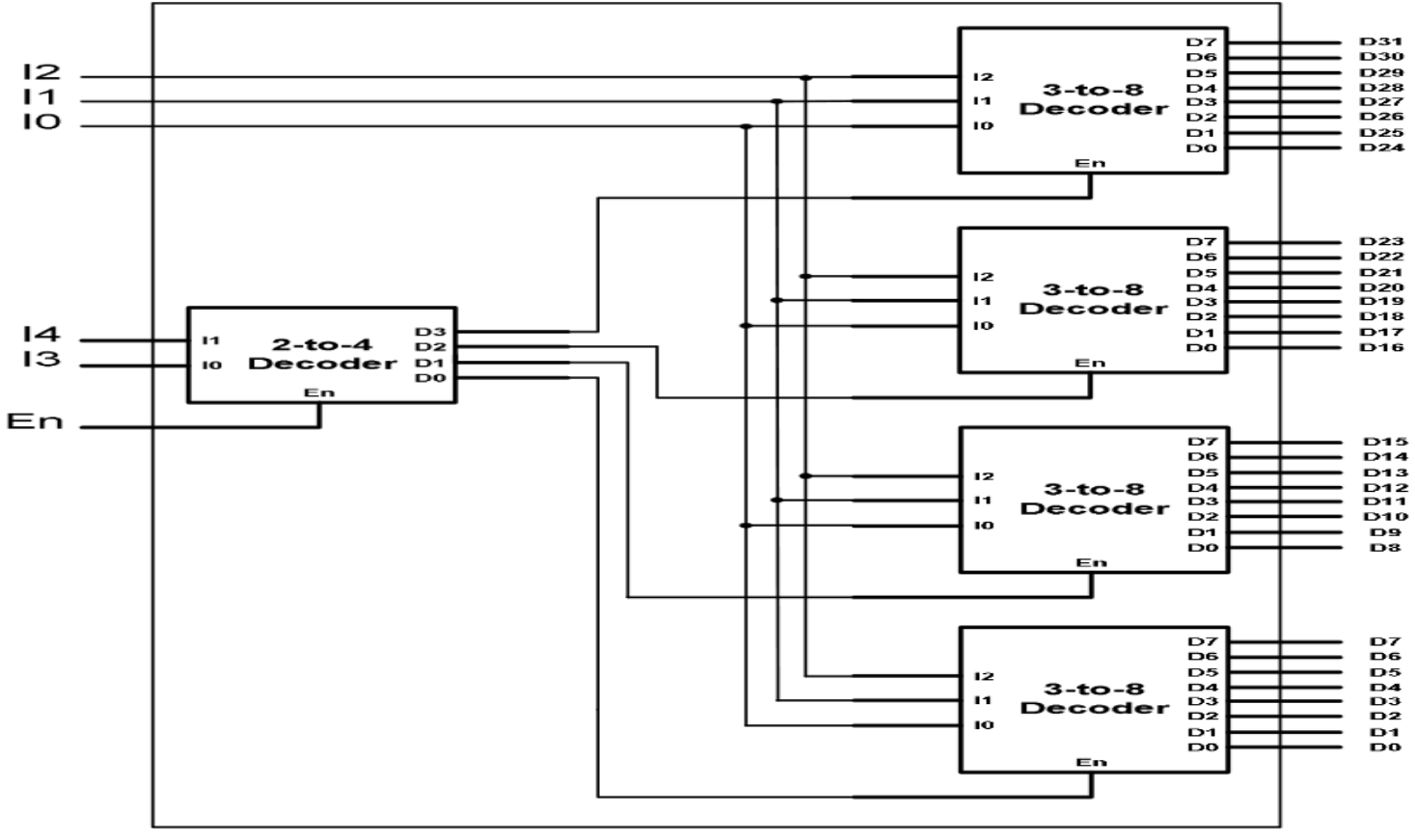



| S3 | S2 | S1 | S0 | Out |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | I0 |
| 0 | 0 | 0 | 1 | I1 |
| 0 | 0 | 1 | 0 | I2 |
| 0 | 0 | 1 | 1 | I3 |
| 0 | 1 | 0 | 0 | I4 |
| 0 | 1 | 0 | 1 | I5 |
| 0 | 1 | 1 | 0 | I6 |
| 0 | 1 | 1 | 1 | I7 |
| 1 | 0 | 0 | 0 | I8 |
| 1 | 0 | 0 | 1 | I9 |
| 1 | 0 | 1 | 0 | I10 |
| 1 | 0 | 1 | 1 | I11 |
| 1 | 1 | 0 | 0 | I12 |
| 1 | 1 | 0 | 1 | I13 |
| 1 | 1 | 1 | 0 | I14 |
| 1 | 1 | 1 | 1 | I15 |

# Modular Design using Ics (5x32 Decoder using 3 x 8 Decoder)



# Modular Design using Ics (5x32 Decoder using 3 x 8 Decoder)



# Modular Design using ICs

- **Priority Encoder Navigation**

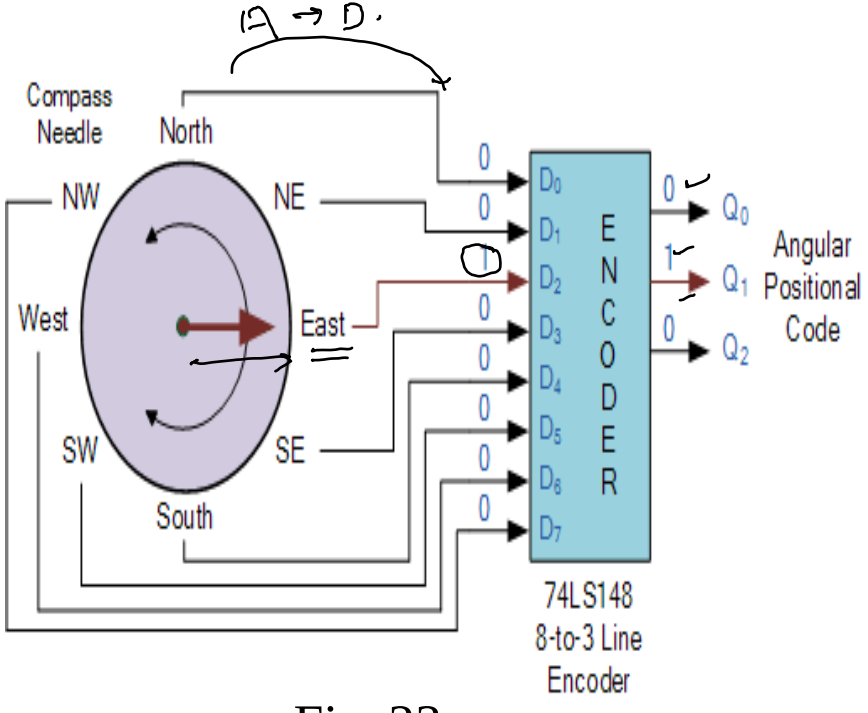



Fig. 23

| Compass Direction | Binary Output | | |
|---|---|---|---|
| | $Q_0$ | $Q_1$ | $Q_2$ |
| North | 0 | 0 | 0 |
| North-East | 0 | 0 | 1 |
| East | 0 | 1 | 0 |
| South-East | 0 | 1 | 1 |
| South | 1 | 0 | 0 |
| South-West | 1 | 0 | 1 |
| West | 1 | 1 | 0 |
| North-West | 1 | 1 | 1 |

# Modular Design using ICs

- **Binary Coded Decimal**

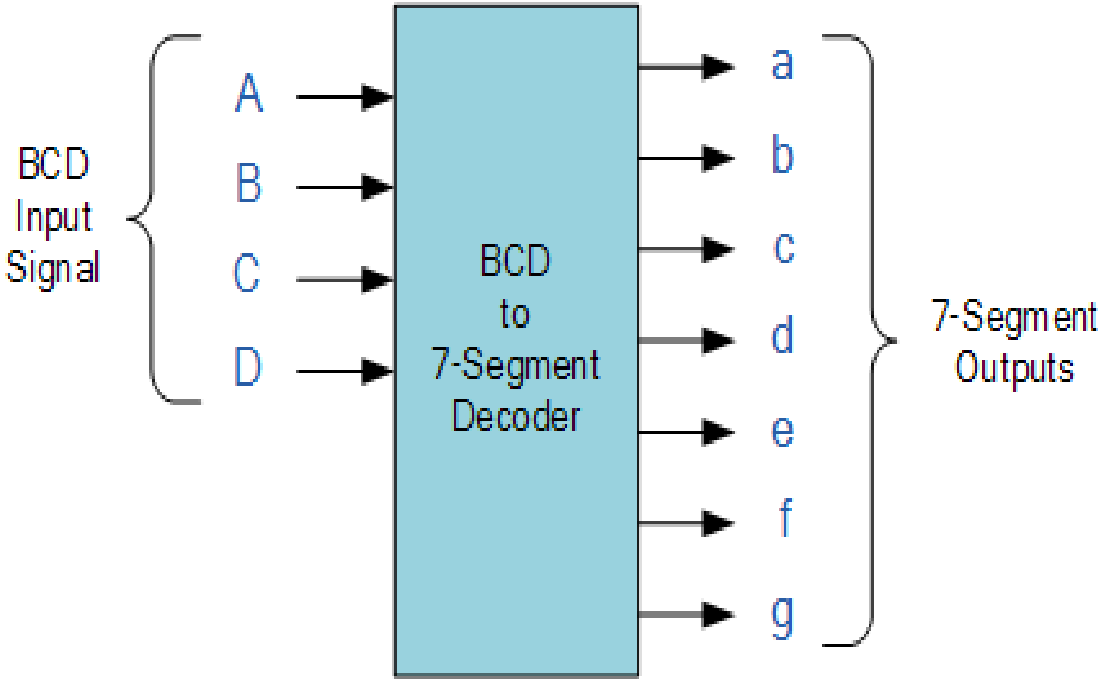



Fig. 24

# Modular Design using ICs

- **Binary Coded Decimal**

### Display Decoder Example No1

In practice current limiting resistors of about 150Ω to 220Ω would be connected in series between the decoder/driver chip and each LED display segment to limit the maximum current flow. There are different display decoders and drivers available for the different types of available displays, either LED or LCD. For example, the 74LS48 for common-cathode LED types, the 74LS47 for common-anode LED types, or the CMOS CD4543 for liquid crystal display (LCD) types.

Liquid crystal displays (LCD´s) have one major advantage over similar LED types in that they consume much less power and nowadays, both LCD and LED displays are combined together to form larger Dot-Matrix Alphanumeric type displays which can show letters and characters as well as numbers in standard Red or Tri-colour outputs.

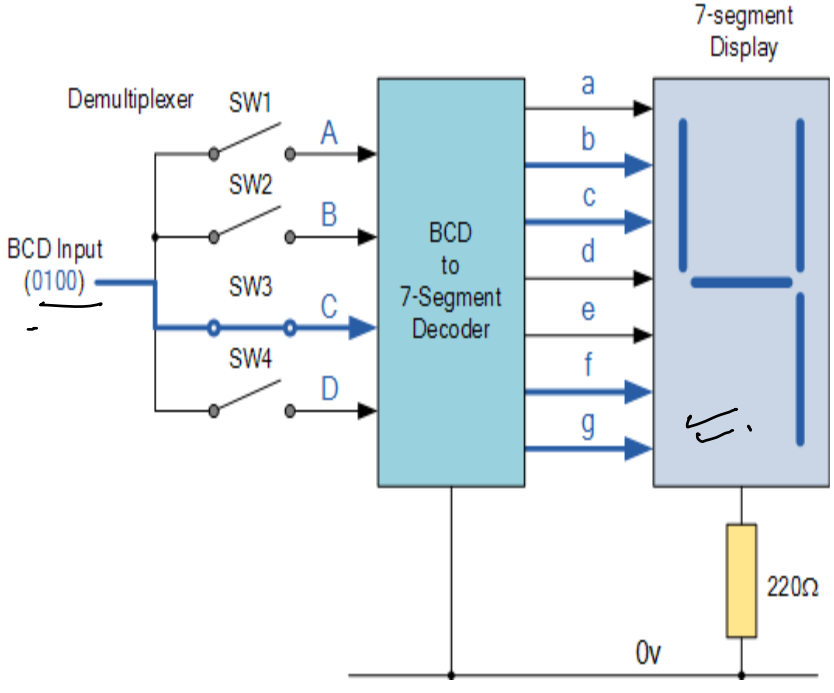



Fig. 25

# Modular Design using ICs

- **Binary Coded Decimal**

**BCD to 7-Segment Display Decoders**

A binary coded decimal (BCD) to 7-segment display decoder such as the TTL 74LS47 or 74LS48, have 4 BCD inputs and 7 output lines, one for each LED segment. This allows a smaller 4-bit binary number (half a byte) to be used to display all the denary numbers from 0 to 9 and by adding two displays together, a full range of numbers from 00 to 99 can be displayed with just a single byte of eight data bits.

The use of **packed** BCD allows two BCD digits to be stored within a single byte (8-bits) of data, allowing a single data byte to hold a BCD number in the range of 00 to 99.
An example of the 4-bit BCD input ( 0100 ) representing the number “4” is given below.

# Modular Design using ICs

**Adjustable Amplifier Gain**

As well as sending parallel data in a serial format down a single transmission line or connection, another possible use of multi-channel multiplexers is in digital audio applications as mixers or where the gain of an analogue amplifier can be controlled digitally, for example.

**Digitally Adjustable Amplifier Gain**

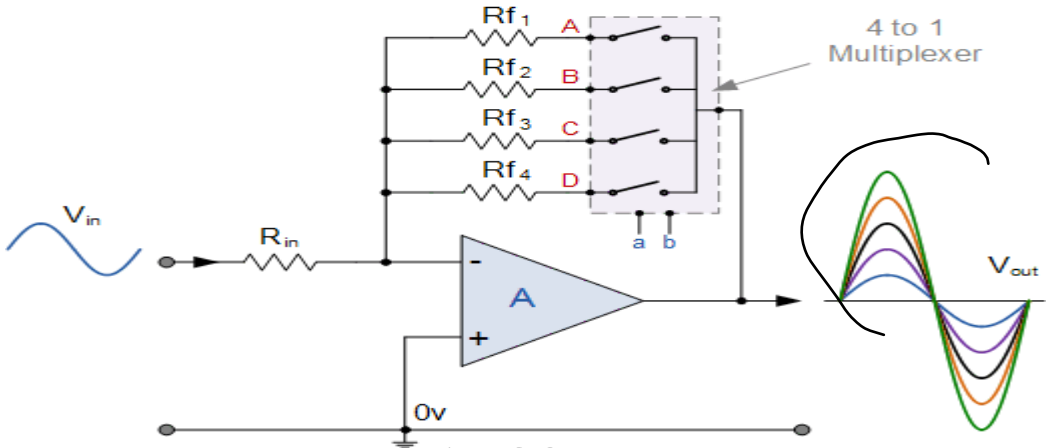



Fig. 26

# Modular Design using ICs

**Digitally Adjustable Amplifier Gain**

The circuit above illustrates how to provide digitally controlled adjustable/variable op-amp gain using a demultiplexer. The voltage gain of the inverting operational amplifier is dependent upon the ratio between the input resistor, $R_{IN}$ and its feedback resistor, R*f* as determined in the Op-amp tutorials.

The digitally controlled analogue switches of the demultiplexer select an input resistor to vary the value of Rin. The combination of these resistors will determine the overall voltage gain of the amplifier, (Av). Then the voltage gain of the inverting operational amplifier can be adjusted digitally simply by selecting the appropriate input resistor combination.

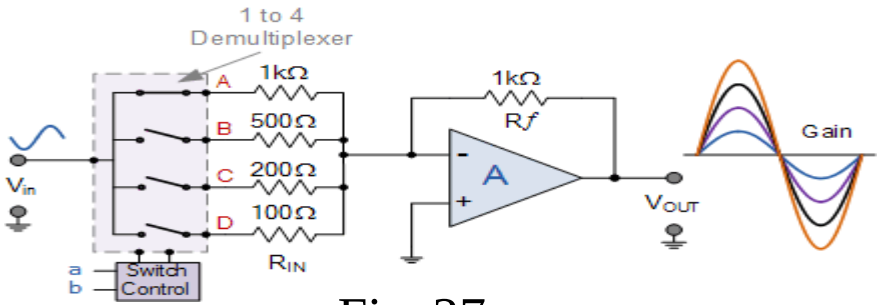



Fig. 27

Thanking You… ☺

Any Question…?