

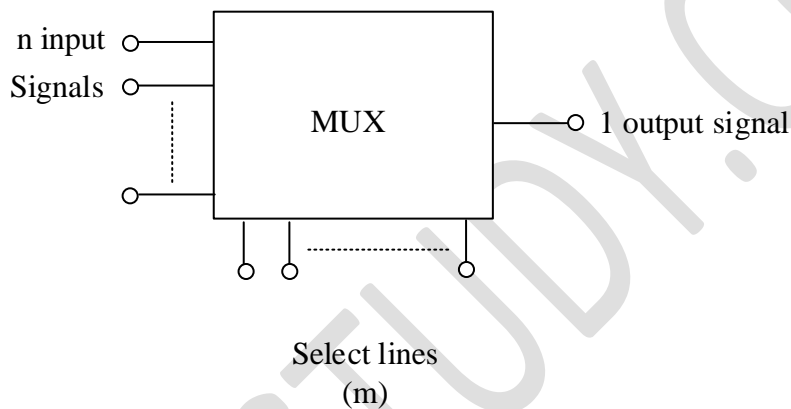
# Combinational Circuits (Part II) Notes

This part of combinational circuits consists of the class of circuits based on data transmission and code converters. These circuits are multiplexers, de multiplexers, decoders, encoders, code converters etc.

**Multiplexer Or Data Selector:** A digital multiplexer is a combinational circuit that selects binary information from one of many input lines and directs it to a single output line.

There are set of select lines ( $2^m$  input lines and  $m$  select lines)

We have known the term multiplexing it means transmitting large number of information unit over a smaller number of channels or lines.



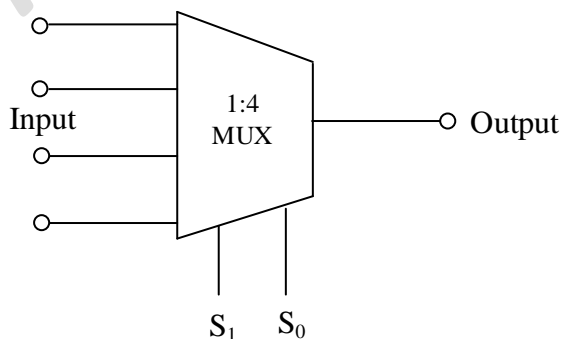
Multiplex means many into one.

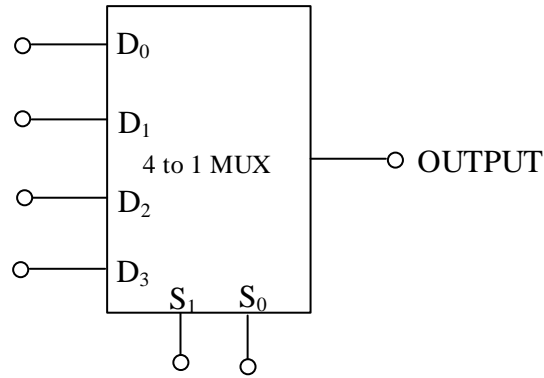
Some available MUXs are:

4:1 MUX                      4 input so  $2^2$  or 2 select lines

8:1 MUX                      8 input so  $2^3$  or 3 select lines

16:1 MUX                     16 input so  $2^4$  or 4 select lines





Block Diagram

Truth table

Data select input		Output
S <sub>1</sub>	S <sub>0</sub>	Y
0	0	D <sub>0</sub>
0	1	D <sub>1</sub>
1	0	D <sub>2</sub>
1	1	D <sub>3</sub>

#### For 4:1 MUX

Data output = Data input D<sub>0</sub> if and only if S<sub>1</sub> = 0 & S<sub>0</sub> = 0

Similarly, other outputs.

Output expression

$$Y = D_0 \bar{S}_1 \bar{S}_0 + D_1 \bar{S}_1 S_0 + D_2 S_1 \bar{S}_0 + D_3 S_1 S_0$$

Note that 4:1 MUX can be implemented using 2 NOT gates, four 3 input AND gates and 4 input OR gate.

MUX ICs have enable input. Enable input (also called strobe) can be used to cascade two more MUXs. *This signal is active low.*

**Implementing High Order MUXs:** Higher order MUXs can be implemented from lower order MUXs.

For example 16:1 MUX can be implemented by two 8:1 MUXs or Four 4:1 MUXs

**Example** Implement 16 to 1 MUX from two 8:1 MUX

**Solution** Input are 16 so select lines needed are  $2^4$ .i.e.4 .

The least significant three lines  $S_2, S_1$  &  $S_0$  are connected to other MUX. Most significant line  $S_3$  is connected to  $\bar{E}$  (enable) of MUX-1 while same with inverter is connected to MUX-2

When  $S_3 = 0$ , MUX-1 is selected

$S_3 = 1$ , MUX-2 is selected

Outputs are connected to OR Gate to get total output.

One can also use 2:1 MUX instead of OR gate.

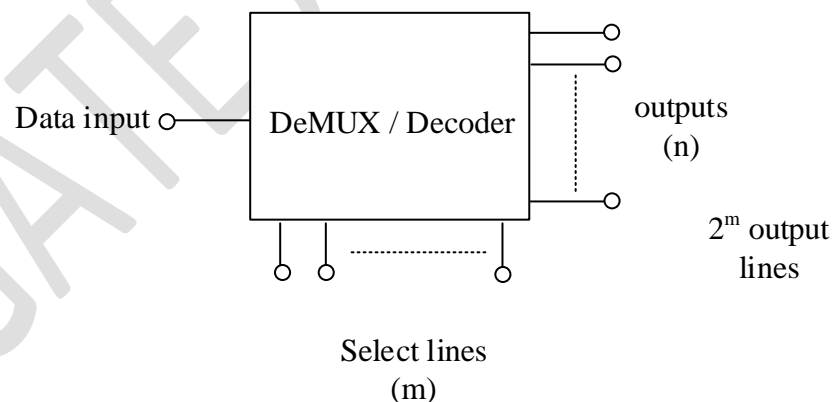
**Implementing Boolean Function Using MUXs:** We have seen MUX consists of set of AND gates feeding a single output OR gate. We know that any Boolean function can be realized using AND, OR and NOT functions. Each AND gate can generate min term when number of variables is equal to number of select lines. 8:1 MUX can generate  $2^3$  min- terms.

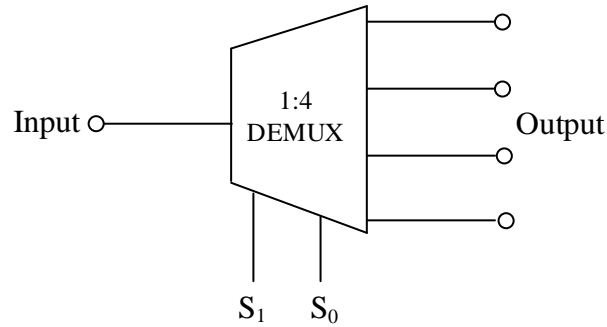
Number of variables that given MUX can realize can be increased by connecting MUX data input to 0, 1, variable or complemented variable.

For example a four variable Boolean function can be realized using 8:1 MUX.

Three variable used as select inputs and fourth to MUX data inputs.

**De Multiplexers (Data Distributors):** De Multiplexers means one into many. It is the process of taking information from one input and transmitting the same over one of several outputs.





Truth table for 1 to 4 Demux

Data input	Selected inputs		Outputs			
D	S <sub>1</sub>	S <sub>0</sub>	Y <sub>3</sub>	Y <sub>2</sub>	Y <sub>1</sub>	Y <sub>0</sub>
D	0	0	0	0	0	D
D	0	1	0	0	D	0
D	1	0	0	D	0	0
D	1	1	D	0	0	0

$$Y_0 = \bar{S}_1 \bar{S}_0 \cdot D$$

$$Y_1 = \bar{S}_1 S_0 \cdot D$$

$$Y_2 = S_1 \bar{S}_0 \cdot D$$

$$Y_3 = S_1 S_0 \cdot D$$

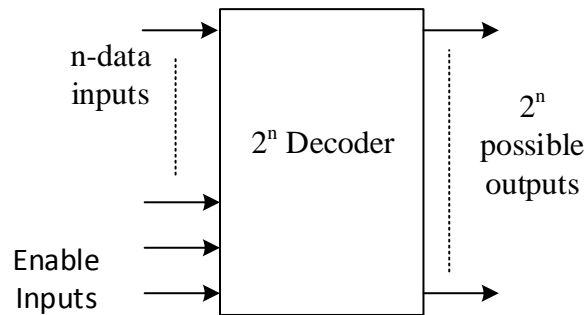
Circuit can be implemented using 3 input AND gates and two NOT gates

## Decoders:

Decoder is like DEMUX but without any data input.

Decoder finds applications in digital display, digital to analog converters and memory addressing.

A decoder is a combinational circuit that converts binary information from n input lines to a maximum of 2<sup>n</sup> output lines. Some decoders have enable inputs they are active high.



Note that the numbers of outputs are greater than number of inputs.

If the number of inputs of inputs and outputs are equal then it is called converter.

For example: BCD to excels 3 code binary to gray etc.

As an example we consider 3 to 8 decoder.

**3 To 8 Decoder:** It has 3 inputs (A,B and C) and eight outputs ( $D_0$  to  $D_7$ ) based on 3 inputs one of the eight outputs is selected. Truth table is given.

Truth table of 3 to 8 decoder

Inputs			Outputs							
A	B	C	$D_0$	$D_1$	$D_2$	$D_3$	$D_4$	$D_5$	$D_6$	$D_7$
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

From the truth logic expression for output can be written.

$$D_0 = \bar{A}\bar{B}\bar{C}, D_1 = \bar{A}\bar{B}C, D_2 = \bar{A}B\bar{C}, D_3 = \bar{A}BC, D_4 = A\bar{B}\bar{C}, D_5 = A\bar{B}C, D_6 = AB\bar{C}$$

$$D_7 = ABC$$

This decoder can be implemented using 3 NOT gates and eight 3 input AND gates.

TTL IC 7445 is a BCD to decimal decoder/driver. The term driver means that it can operate at a higher current and voltage limits than normal TTL outputs.

Another important decoder is BCD to seven segment decoder/driver.

IC 7446 or 7447 used to drive common anode seven segment LED displayed.

Each segment of a typical LED display is rated to operate at 10mA and 2.8V for normal brightness.

**Encoders:** A combinational logic circuit that accepts  $2^n$  binary inputs and produces n encoded output values.

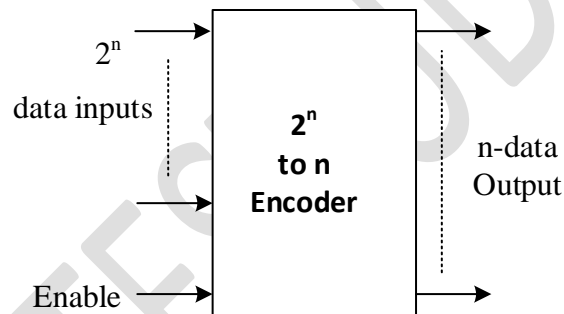
They perform a function that is inverse of decoders. Encoders have more inputs than output variables.

We have already seen binary to octal decoder(3to 8 decoder) accepts 3 bit input code and activates one of the 8 output lines corresponding to that code.

An octal to binary encoder performs opposite function.It accepts 8 inputs and produces 3 bit output code corresponding to activated input.

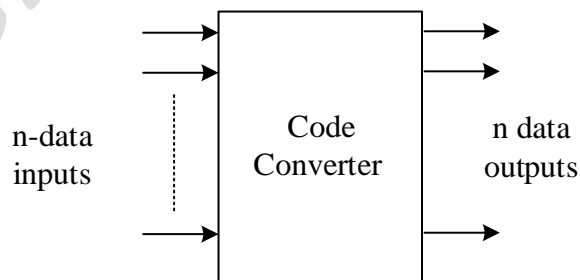
Decimal to BCD decoder is another example.

In normal encoder only one input line is high at a time. In Priority Encoder more than one input line can be high but only the one with high priority will be available.



An 8-to-3 encoder has 8 input lines and 3 output lines.

**Code Converters** Code converter is a logic circuit that changes data from one type of binary code to another type.



Most commonly used code converters are

BCD to binary

Binary to BCD

Binary to Gray code

Gray code to binary.

To convert from binary code A to binary code B, the input lines must supply the bit combination of elements as per code A and the output lines generate the corresponding bit combination of code B. It is illustrated through an example.

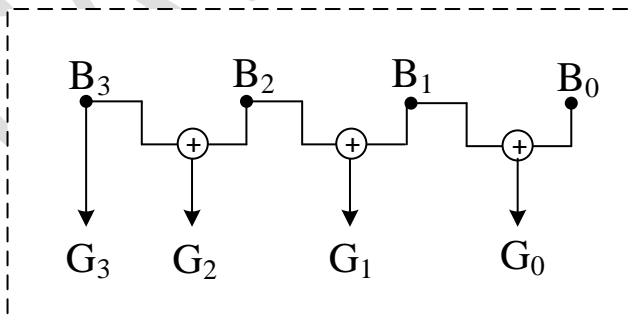
BCD to excess 3 code conversion

Inputs BCD				Outputs Excess-3 code			
A	B	C	D	w	x	y	z
0	0	0	0	00	11		
0	0	0	1	0	1	0	0
0	0	1	0	0	101		
0	0	1	1	0	110		
0	1	0	0	0	111		
0	1	0	1	10	0	0	
0	1	1	0	10	0	1	
0	1	1	1	10	10		
1	00	0		1	0	1	1
1	00	1		1	1	0	0

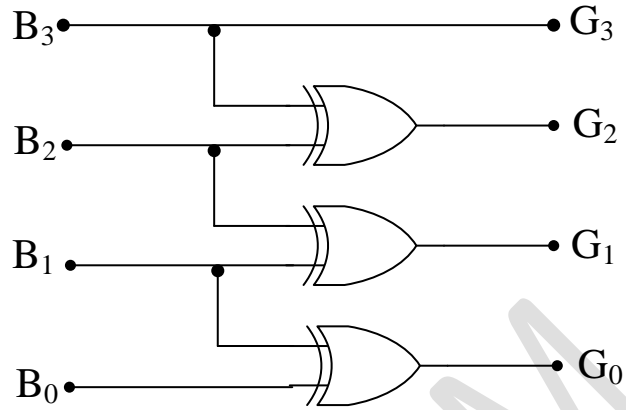
Note that Excess-3 code is generated by  $BCD + 0011(3_{10}) = \text{Excess 3-code}$ .

**Binary to Gray code converter:** Figure gives the governing Boolean Equations and logical Gate diagram for such conversion. Also a diagram indicating bit by bit conversion is also given. This will be helpful in Problem solving.

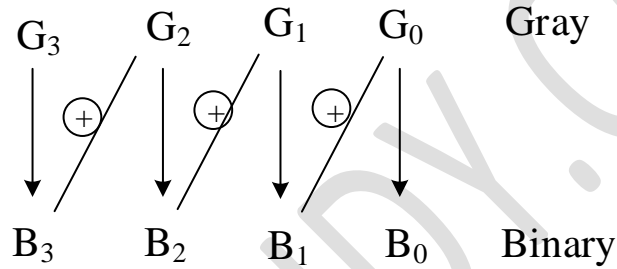
Binary to Gray code converter



$$\begin{aligned}
 G_3 &= B_3 \\
 G_2 &= B_3 \oplus B_2 \\
 G_1 &= B_2 \oplus B_1 \\
 G_0 &= B_1 \oplus B_0
 \end{aligned}$$



**Gray to Binary Code Converter:** This figure gives the conversion equations and logical gate diagram. Bit to bit conversion diagram is also given.



$$\begin{aligned}
 B_3 &= G_3 \\
 B_2 &= B_3 \oplus G_2 \\
 B_1 &= B_2 \oplus G_1 \\
 B_0 &= B_1 \oplus G_0
 \end{aligned}$$

