## 2.5 MUTLIPLEXER

Multiplexer means many into one. A multiplexer is a circuit used to select and route any one of the several input signals to a single output. A simple example of an non-electronic circuit of a multiplexer is a single pole multi-position switch.

Multi-position switches are widely used in many electronics circuits. However, circuits that operate at high speed require the multiplexer to be automatically selected. A mechanical switch cannot perform this task efficiently. Therefore, multiplexer is used to perform high speed switching are constructed of electronic components.

Multiplexers can handle two type of data i.e., analog and digital. For analog application, multiplexer are built using relays and transistor switches. For digital application, they are built from standard logic gates.

The multiplexer used for digital applications, also called digital multiplexer, is a circuit with many input but only one output. By applying control signals (also known as Select Signals), we can steer any input to the output. Some of the common types of multiplexer are 2-to-1, 4-to-1, 8-to-1, 16-to-1 multiplexer.

Following figure shows the general idea of a multiplexer with n input signal, m control signals and one output signal.
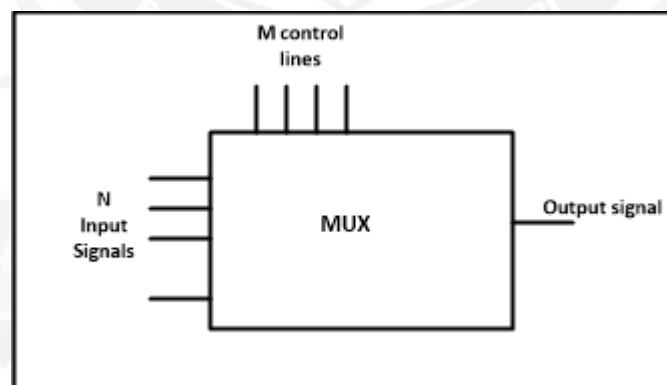
Figure 2.5.1 Multiplexer Block diagram

[Source: https://www.electronicshub.org/multiplexer-and-demultiplexer/]

The 4-to-1 multiplexer has 4 input bits, 2 control or select bits, and 1 output bit. The four input bits are D0,D1,D2 and D3. Only one of this is transmitted to the output Y. The output depends on the values of A and B, which are the control inputs. The control input determines which of the input data bit is transmitted to the output.

For instance, as shown in figure, when A B = 0 0 , the upper AND gate is enabled, while all other AND gates are disabled. Therefore, data bit D0 is transmitted to the output, giving Y = Do.
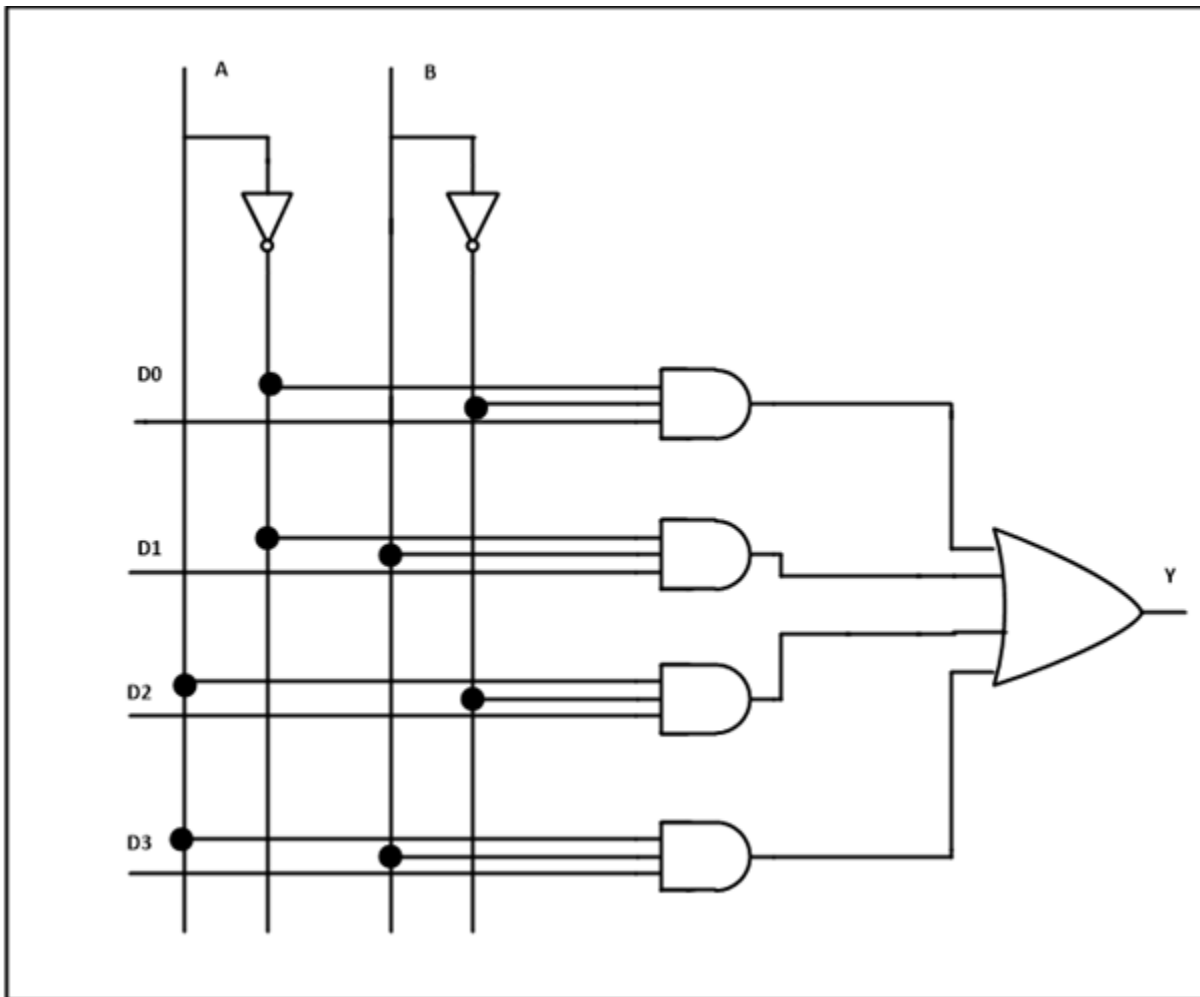
Figure 2.5.2 4-to-1 Multiplexer

[Source: https://www.electronicshub.org/multiplexer-and-demultiplexer/]

If the control input is changed to A B = 1 1 , all gates are disabled except the bottom AND gate. In this case, D3 is transmitted to the output and Y = D3.

- An example of 4-to-1 multiplexer is IC 74153 in which the output is same as the input.

- Another example of 4-to-1 multiplexer is 45352 in which the output is the compliment of the input.

- Example of 16-to-1 line multiplexer is IC 74150.

Applications of Multiplexer

Multiplexer are used in various fields where multiple data need to be transmitted using a single line. Following are some of the applications of multiplexers –

1. Communication System – Communication system is a set of system that enable communication like transmission system, relay and tributary station, and communication network. The efficiency of communication system can be increased considerably using multiplexer. Multiplexer allow the process of transmitting different type of data such as audio, video at the same time using a single transmission line.

2. Telephone Network – In telephone network, multiple audio signals are integrated on a single line for transmission with the help of multiplexers. In this way, multiple audio signals can be isolated and eventually, the desire audio signals reach the intended recipients.

3. Computer Memory – Multiplexers are used to implement huge amount of memory into the computer, at the same time reduces the number of copper lines required to connect the memory to other parts of the computer circuit.

4. Transmission from the Computer System of a Satellite – Multiplexer can be used for the transmission of data signals from the computer system of a satellite or spacecraft to the ground system using the GPS (Global Positioning System) satellites.

2.5.1 Demultiplexer

Demultiplexer means one to many. A demultiplexer is a circuit with one input and many outputs. By applying control signal, we can steer any input to the output. Few types of demultiplexer are 1-to 2, 1-to-4, 1-to-8 and 1-to 16 demultiplexer.

Following figure illustrate the general idea of a demultiplexer with 1 input signal, m control signals, and n output signals.
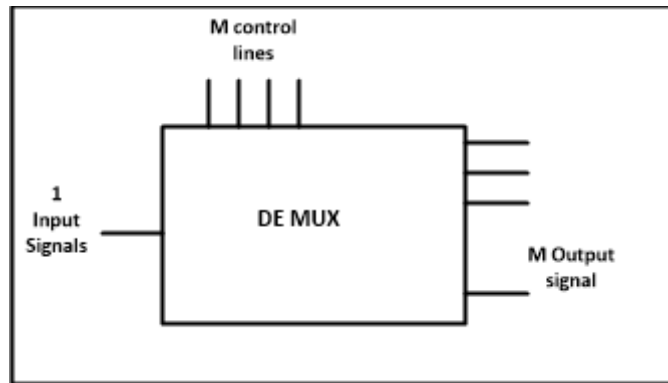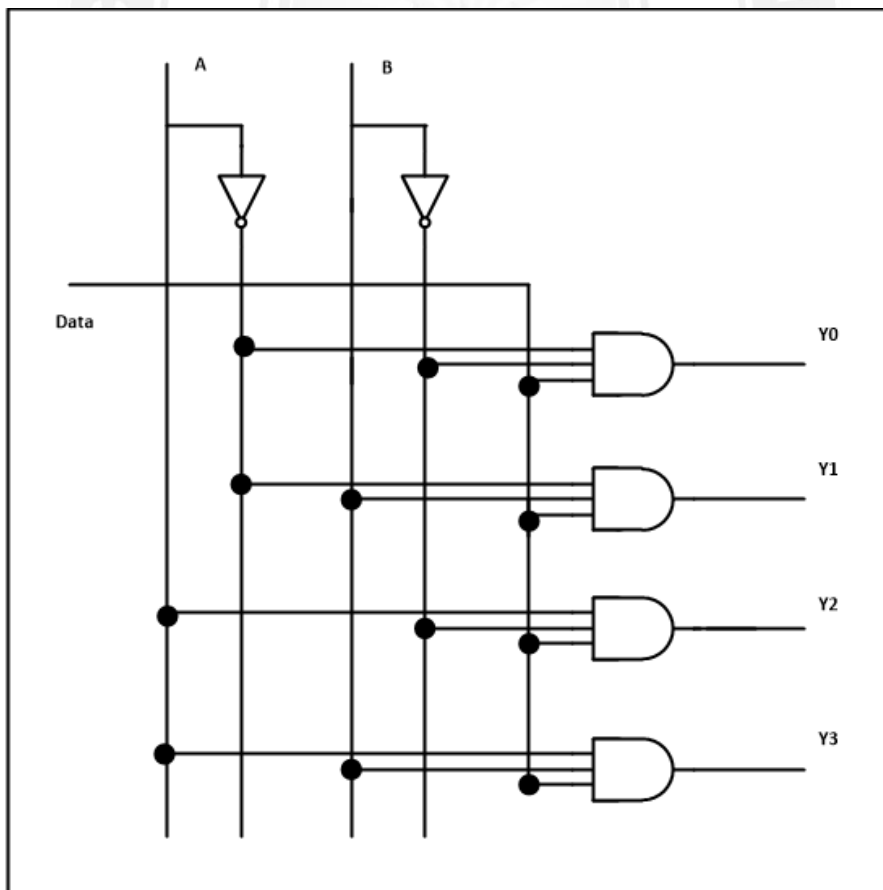
Figure 2.5.3 De-Multiplexer Block diagram

[Source: https://www.electronicshub.org/multiplexer-and-demultiplexer/]

1-to-4 Demultiplexer

The 1-to-4 demultiplexer has 1 input bit, 2 control or select bits, and 4 output bits. An example of 1-to-4 demultiplexer is IC 74155. The 1-to-4 demultiplexer is shown in figure below-

Figure 2.5.4 1-to-4 De-Multiplexer

[Source: https://www.electronicshub.org/multiplexer-and-demultiplexer/]

The input bit is labelled as Data D. This data bit is transmitted to the selected output lines, which depends on the values of A and B, the control or Select Inputs.

When A B = 0 1 , the second AND gate from the top is enabled while other AND gates are disabled. Therefore, data bit D is transmitted to the output Y1, giving Y1 = Data.

If D is LOW, Y1 is LOW. If D is HIGH, Y1 is HIGH. The value of Y1 depends upon the value of D. All other outputs are in low state.

If the control input is changed to A B = 1 0 , all the gates are disabled except the third AND gate from the top. Then, D is transmitted only to the Y2 output, and Y2 = Data.

Example of 1-to-16 demultiplexer is IC 74154. It has 1 input bit, 4 control / select bits and 16 output bit.

Applications of Demultiplexer

1. Demultiplexer is used to connect a single source to multiple destinations. The main application area of demultiplexer is communication system, where multiplexers are used. Most of the communication system are bidirectional i.e., they function in both ways (transmitting and receiving signals). Hence, for most of the applications, the multiplexer and demultiplexer work in sync. Demultiplexer are also used for reconstruction of parallel data and ALU circuits.

2. Communication System – Communication system use multiplexer to carry multiple data like audio, video and other form of data using a single line for transmission. This process make the transmission easier. The demultiplexer receive the output signals of the multiplexer and converts them back to the original form of the data at the receiving end. The multiplexer and demultiplexer work together to carry out the process of transmission and reception of data in communication system.

3. ALU (Arithmetic Logic Unit) – In an ALU circuit, the output of ALU can be stored in multiple registers or storage units with the help of demultiplexer. The output of ALU is fed as the data input to the demultiplexer. Each output of demultiplexer is connected to multiple register which can be stored in the registers.

4. Serial to Parallel Converter – A serial to parallel converter is used for reconstructing parallel data from incoming serial data stream. In this technique, serial data from the incoming serial data stream is given as data input to the

demultiplexer at the regular intervals. A counter is attach to the control input of the demultiplexer. This counter directs the data signal to the output of the demultiplexer where these data signals are stored. When all data signals have been stored, the output of the demultiplexer can be retrieved and read out in parallel.

Problem 1

Implement the boolean expression $F(A, B, C) = \sum m(0, 2, 5, 6)$ using 4 : 1 multiplexer.
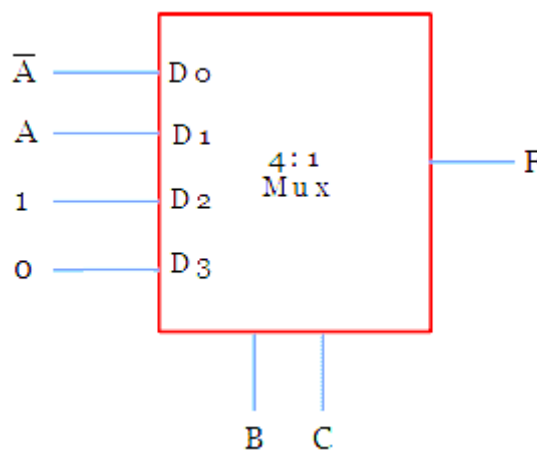
Solution:

In the given boolean expression, there are 3 variables. We should use $2^3 : 1 = 8 : 1$ multiplexer. But as per the question, it is to be implemented with 4 : 1 mux.

For 4 : 1 multiplexer, there should be 2 selection lines. So from the given 3 variables, the 2 least significant variables(B, C) are used as selection line inputs.

4 : 1 multiplexer implementation table.

The minterms given in the boolean expression is circled and analyzed. After analyzing, the input values of 4 : 1 mux is obtained as , A, 1, 0. Thus the circuit can be drawn as below.

Figure 2.5.5 4 : 1 multiplexer implementation

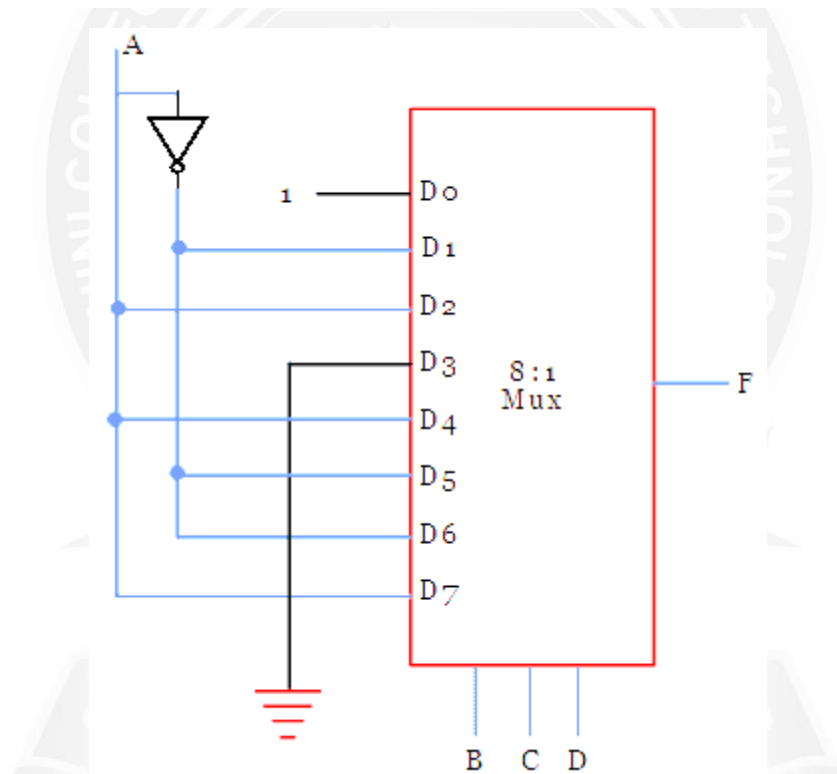*[Source: https://www.electrically4u.com/solved-problems-on-multiplexer/]*

Problem 2

Implement $F(A, B, C, D) = \sum m(0, 1, 5, 6, 8, 10, 12, 15)$ using 8 : 1 multiplexer.

Solution:

In the given boolean expression, there are 4 variables. We should use $2^4 : 1 = 16 : 1$ multiplexer.

The 8 inputs are derived using the implementation table shown below

	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
$\overline{A}$	⓪	①	2	3	4	⑤	⑥	7
A	⑧	9	⑩	11	⑫	13	14	⑮
	1	$\overline{A}$	A	0	A	$\overline{A}$	$\overline{A}$	A

Figure 2.5.6 8 : 1 multiplexer implementation

[Source: https://www.electrically4u.com/solved-problems-on-multiplexer/]