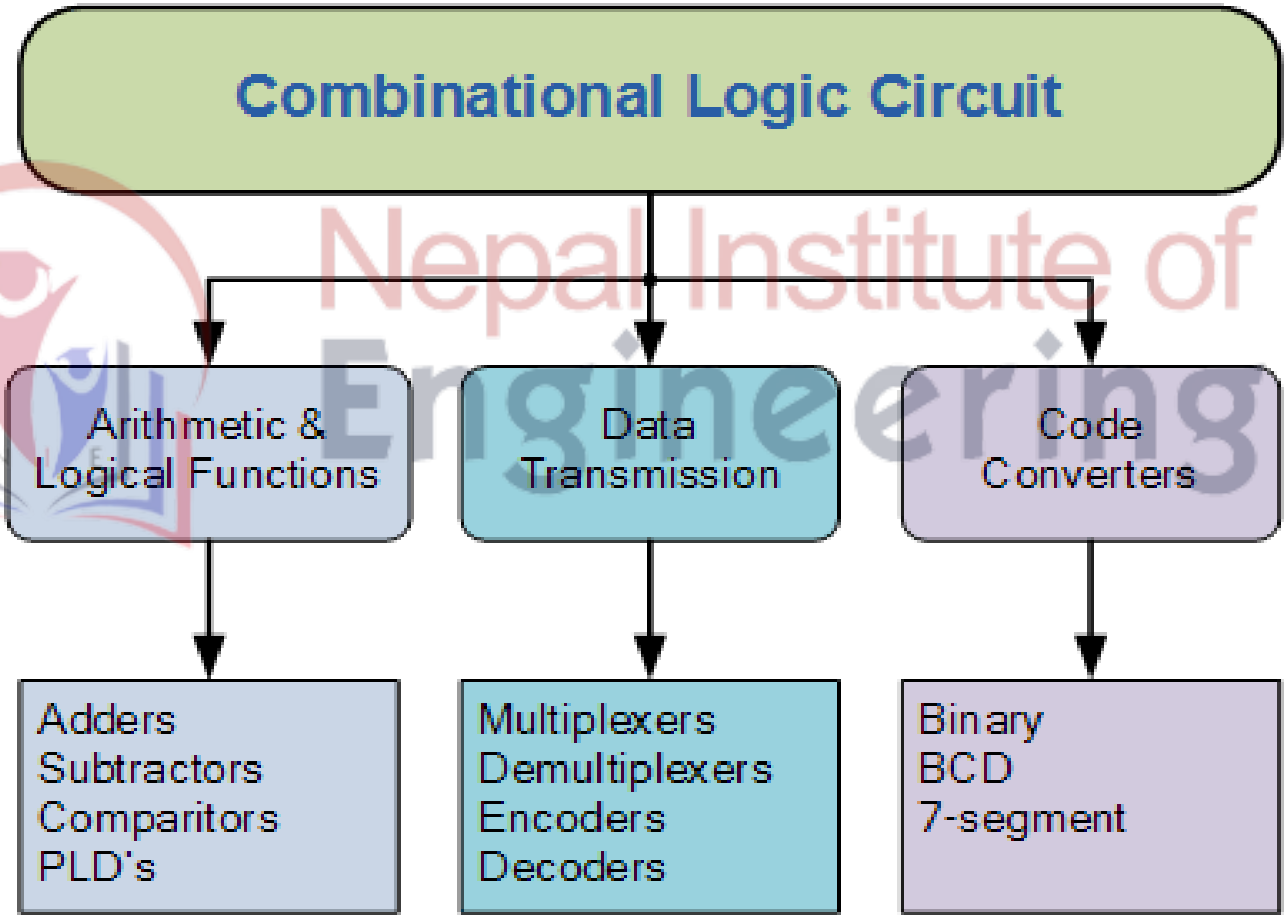
Combinational Circuits

Combinational circuits

- A combinational logic circuit is one in which the present state of the combination of the logic inputs decides the output .
- The term combination logic means combining of two or more logic gates to form a required function where the output at a given time depends only on the input.

- The required output data is obtained from this process by transforming the binary information given at the input.

Classification of Combinational Logic



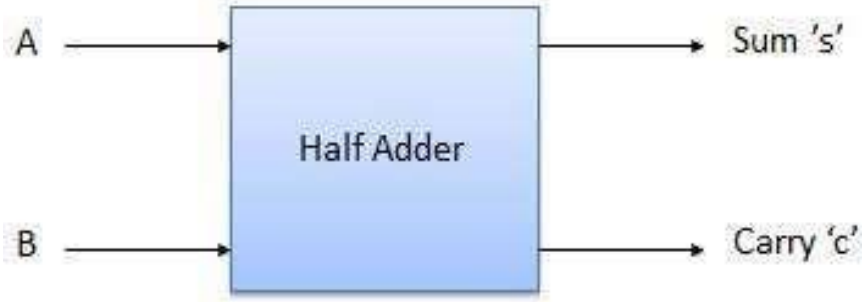
Characterstics:

- The output of combinational circuit at any instant of time, depends only on the levels present at input terminals.
- The combinational circuit do not use any memory. The previous state of input does not have any effect on the present state of the circuit.
- A combinational circuit can have an n number of inputs and m number of outputs.

Half Adder

- Half adder is a combinational logic circuit with two inputs and two outputs.
- The half adder circuit is designed to add two single bit binary number A and B.
- It is the basic building block for addition of two **single** bit numbers.
- This circuit has two outputs **carry** and **sum**.

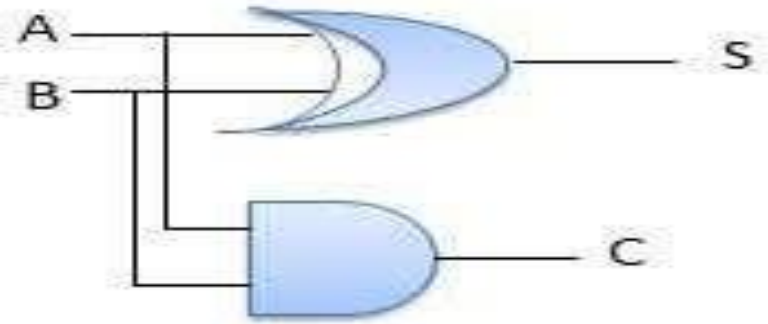
Block diagram:



Truth Table:

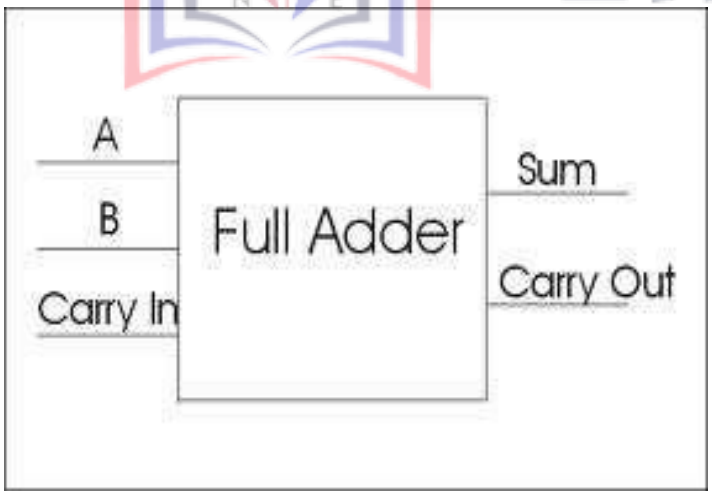
Inputs		Output	
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Circuit Diagram:



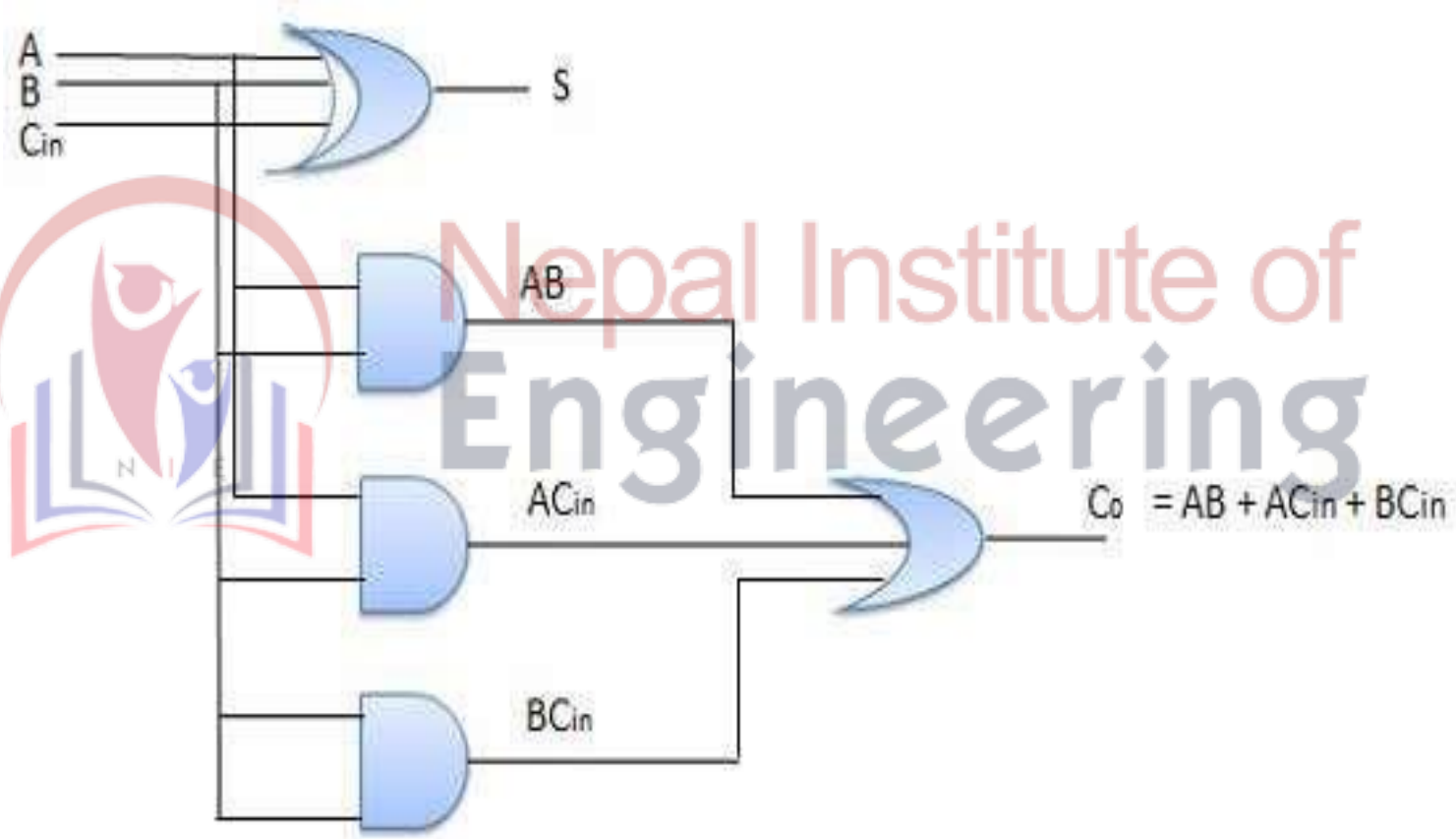
Full-Adder

- A combinational circuit that performs the addition of three bits at a time is called “ Full-Adder ”
- Block diagram & the truth table of Full-Adder .



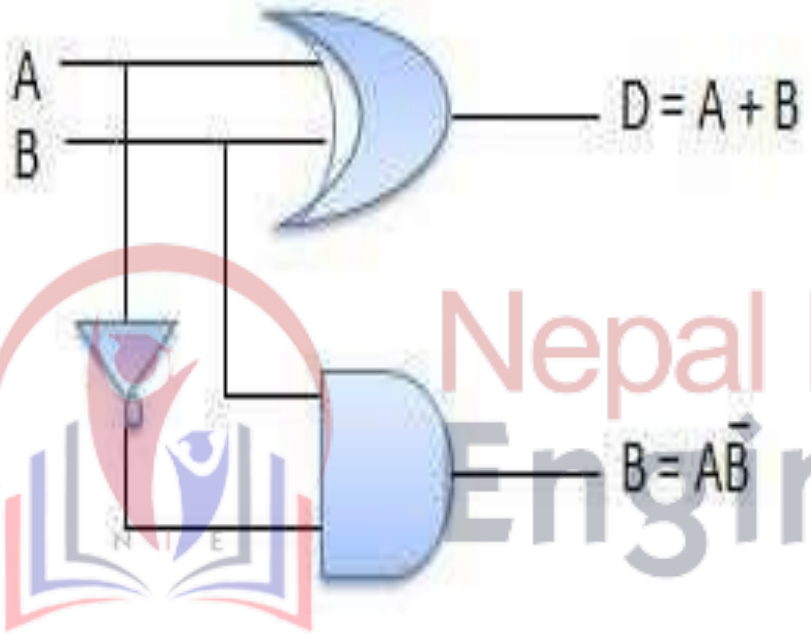
A	B	C_i	C_o	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Circuit diagrams of full-adder



Half Subtractor

- Half subtractor is a combination circuit with two inputs and two outputs (difference and borrow).
- It produces the difference between the two binary bits at the input and also produces a output (Borrow) to indicate if a 1 has been borrowed. In the subtraction (A-B), A is called as Minuend bit and B is called as Subtrahend bit.

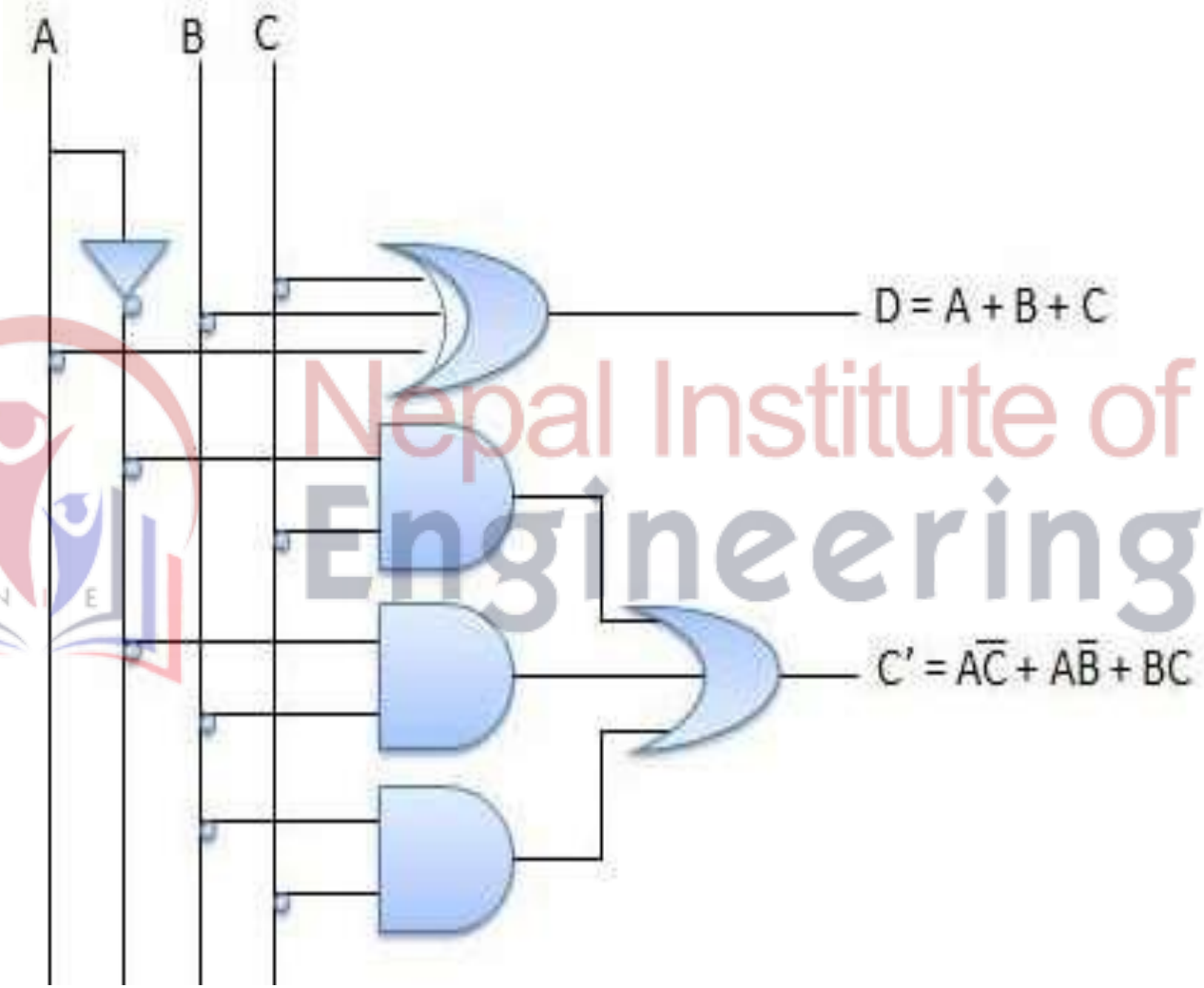


Inputs		Output	
A	B	(A – B)	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Full Subtractor

- The full subtractor is a combinational circuit with three inputs A,B,C and two output D and C'. A is the minuend, B is subtrahend, C is the borrow produced by the previous stage, D is the difference output and C' is the borrow output.

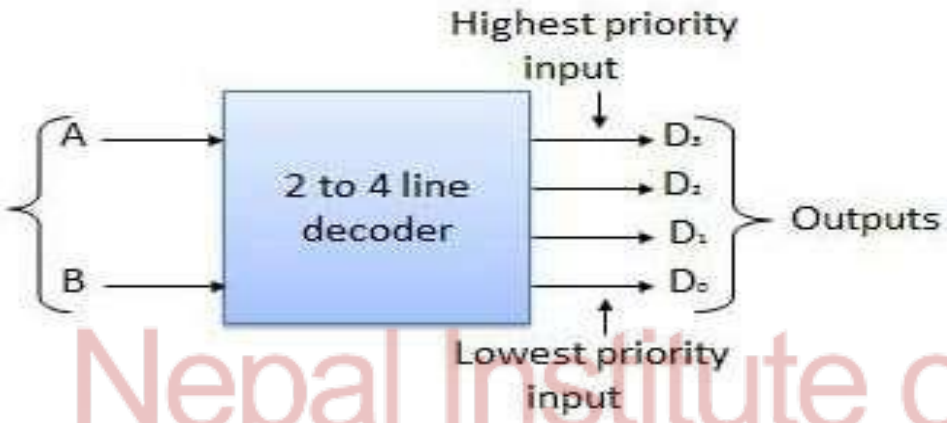
Inputs			Output	
A	B	C	(A-B-C)	C'
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1



Decoder

- A decoder is a combinational circuit that has n input and to a maximum m = 2^n outputs.
- Decoder is identical to a demultiplexer without any data input.
- It performs operations which are exactly opposite to those of an encoder.

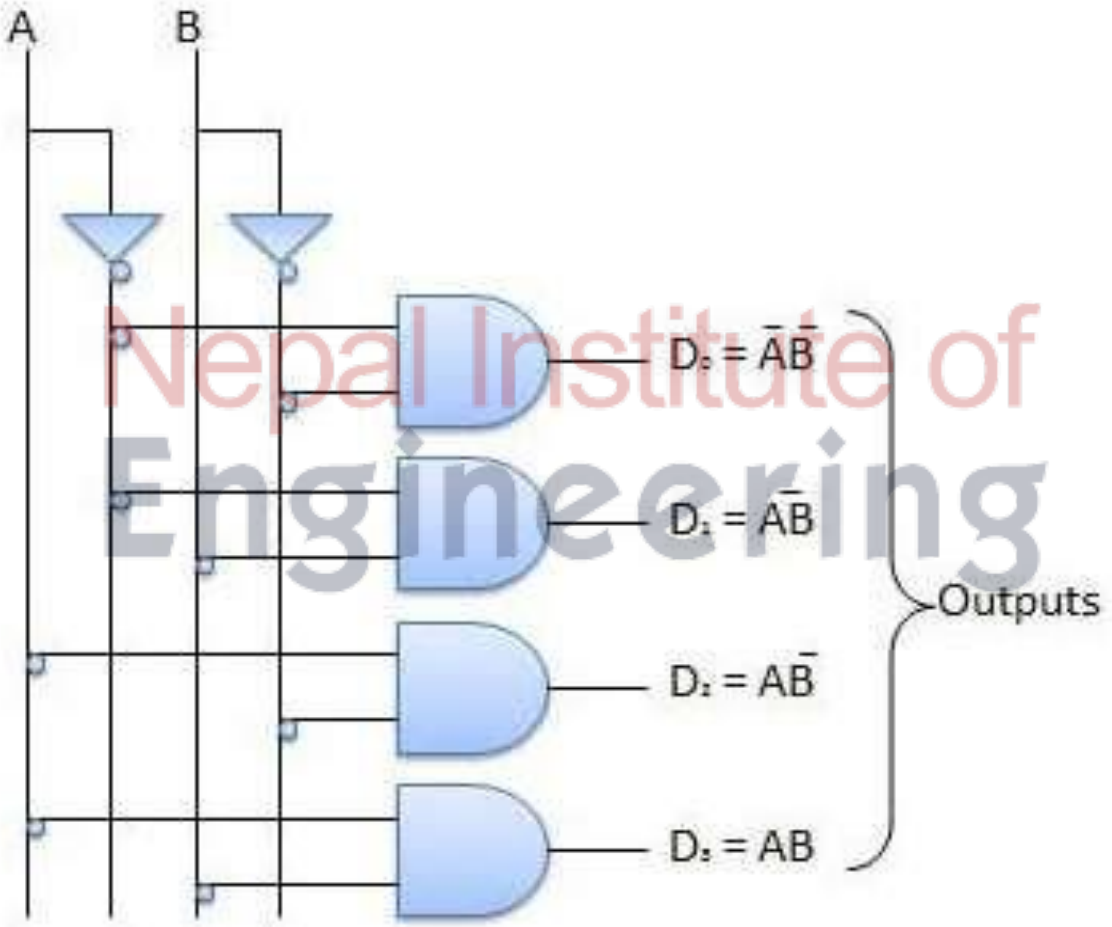
BLOCK DIAGRAM



TRUTH TABLE

Inputs		Output			
A	B	D_0	D_1	D_2	D_3
0	0	1	0	0	0
0	1	0	1	0	0
0	1	0	0	1	0
1	1	0	0	0	1

LOGIC CIRCUIT



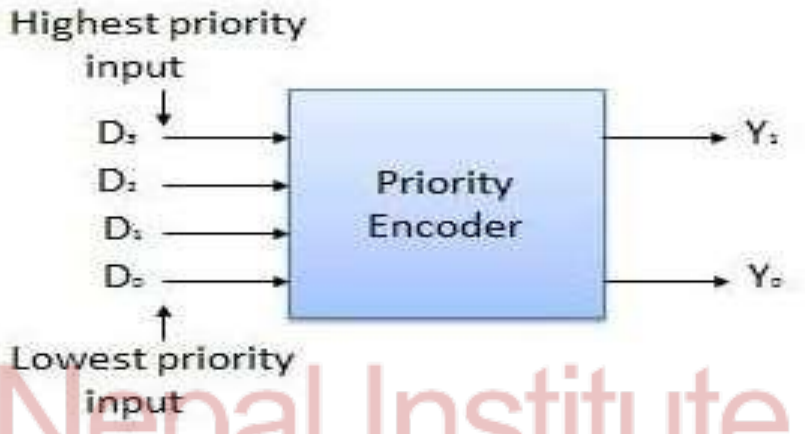
Encoder

- Encoder is a combinational circuit which is designed to perform the inverse operation of the decoder.
- An encoder has n number of input lines and m number of output lines.
- An encoder produces an m bit binary code corresponding to the digital input number.
- The encoder accepts an n input digital word and converts it into an m bit another digital word.

Priority Encoder

- This is a special type of encoder. Priority is given to the input lines.
- If two or more input line are 1 at the same time, then the input line with highest priority will be considered.
- There are four input D_0, D_1, D_2, D_3 and two output Y_0, Y_1. Out of the four input D_3 has the highest priority and D_0 has the lowest priority.
- If $D_3 = 1$ then Y_1 $Y_1 = 11$ irrespective of the other inputs. Similarly if $D_3 = 0$ and $D_2 = 1$ then Y_1 $Y_0 = 10$ irrespective of the other inputs.

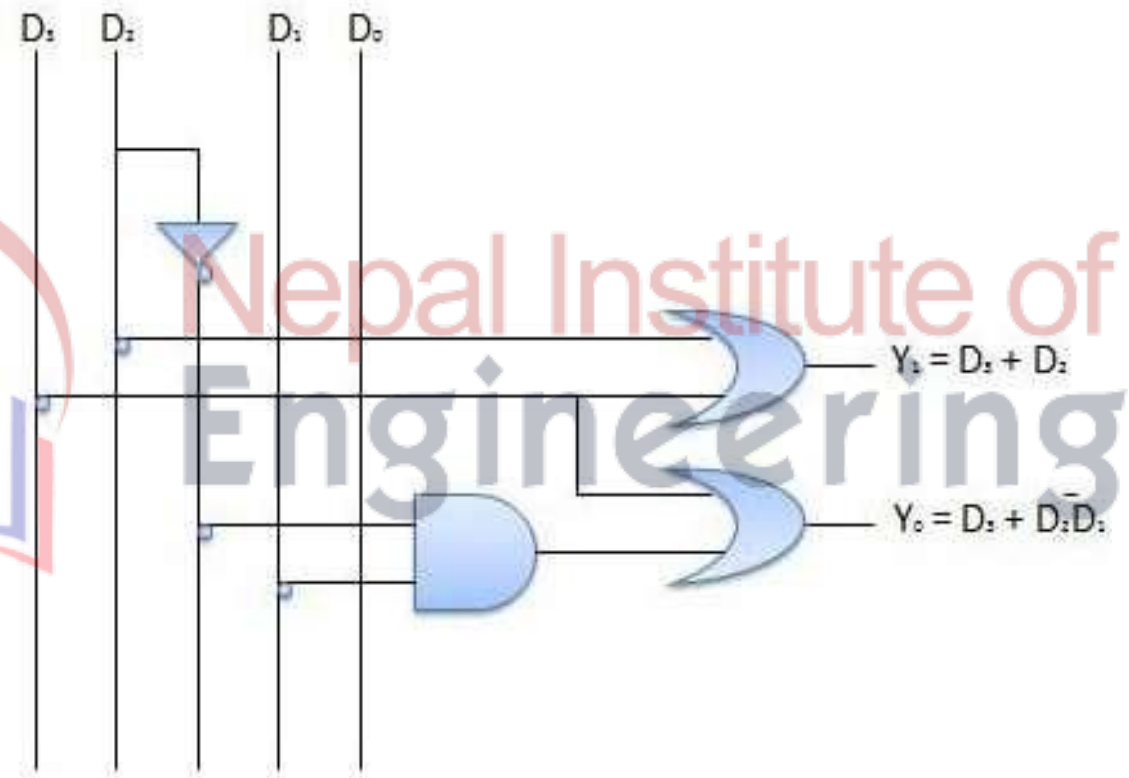
BLOCK DIAGRAM



TRUTH TABLE

Highest	Inputs		Lowest	Outputs	
D_3	D_2	D_1	D_0	Y_0	Y_1
0	0	0	0	x	x
0	0	0	1	0	0
0	0	1	x	0	1
0	1	x	x	1	0
1	x	x	x	1	1

LOGIC CIRCUIT



Multiplexer
(DATA SELECTOR)

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.

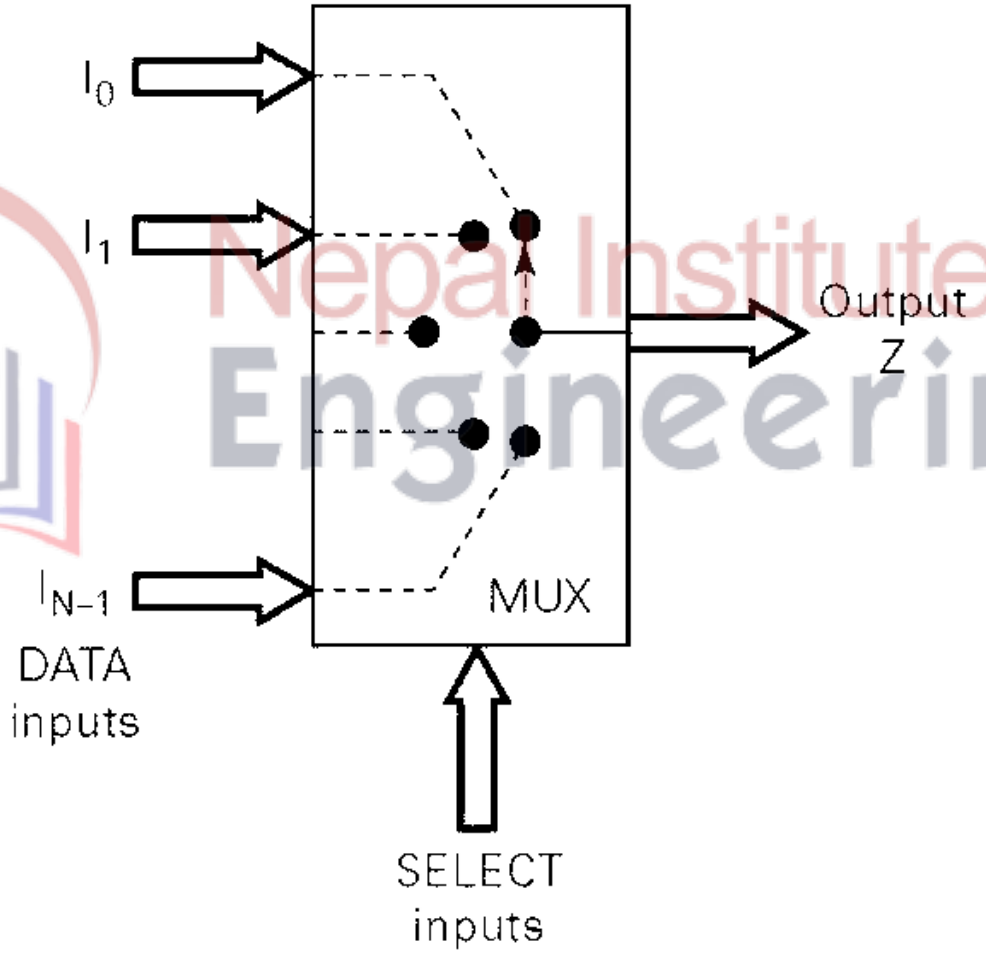
MUX Types

→ 2-to-1 (1 select line)
→ 4-to-1 (2 select lines)
→ 8-to-1 (3 select lines)
→ 16-to-1 (4 select lines)

Multiplexers

- A multiplexer has
  - N control / select inputs
  - 2^N data inputs
  - 1 output
- Selection input (N) determines the input that should be connected to the output

Functional Diagram Of a Multiplexer



2 : 1 Multiplexer

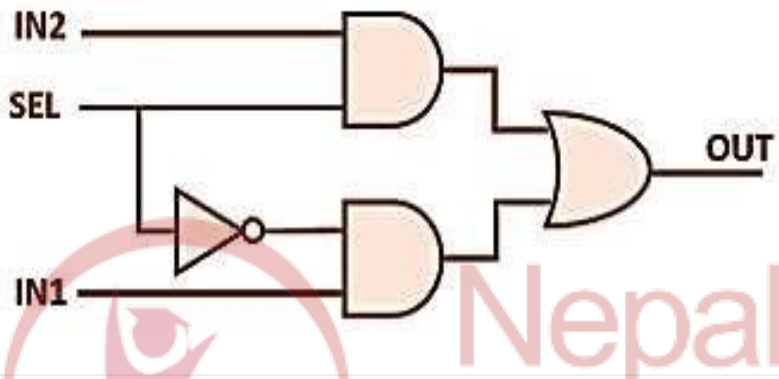


Figure : Logic diagram of 2x1 mux

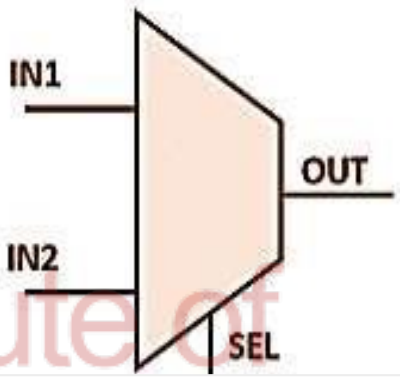
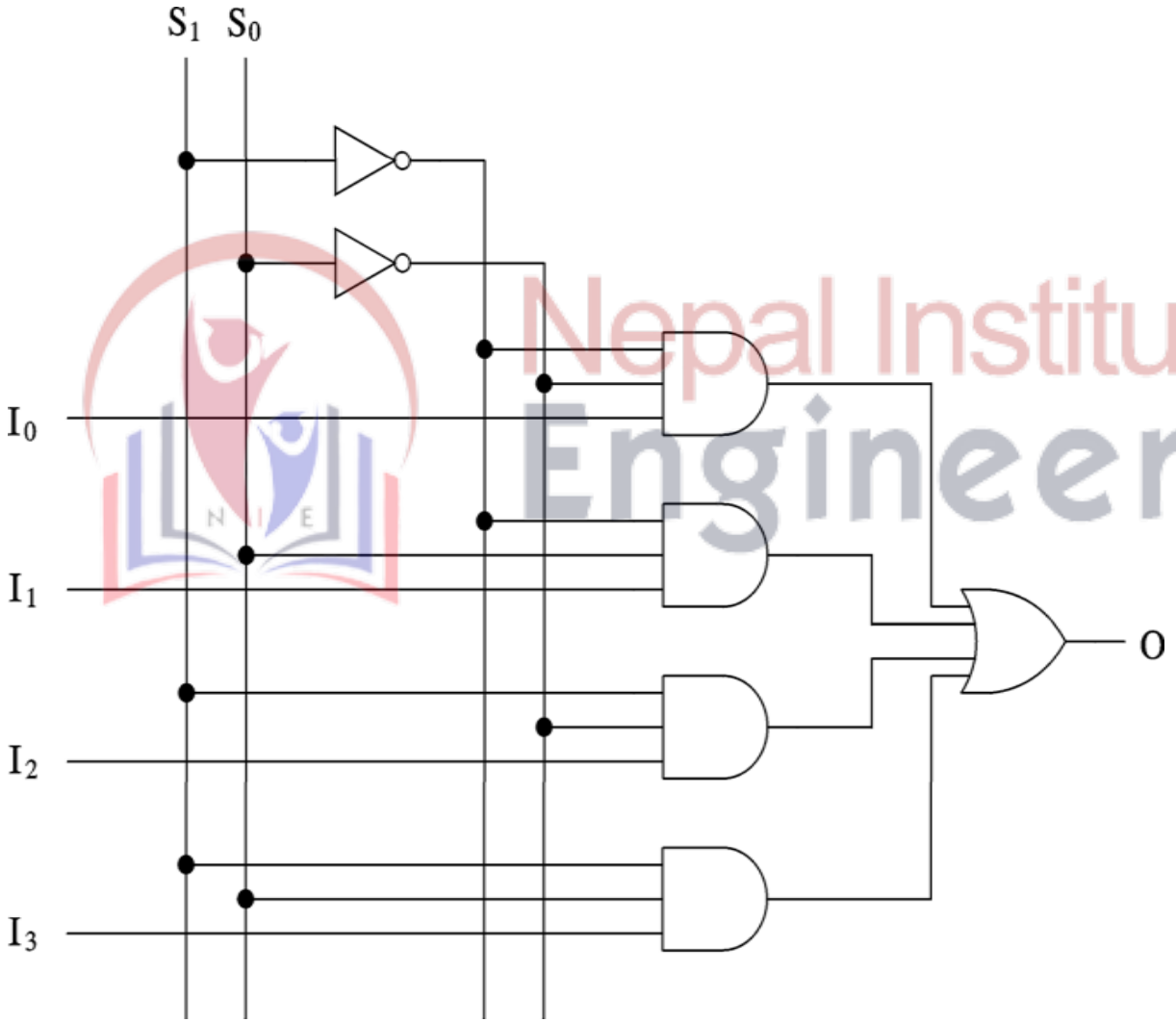
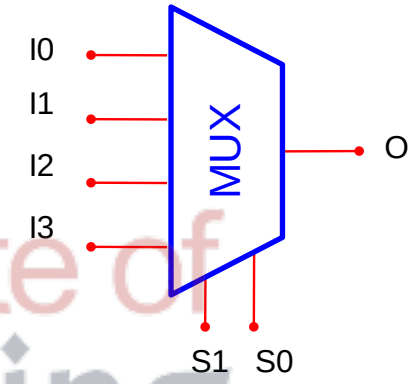


Figure : Schematic symbol of 2x1 mux

S	Z
0	I_0
1	I_1

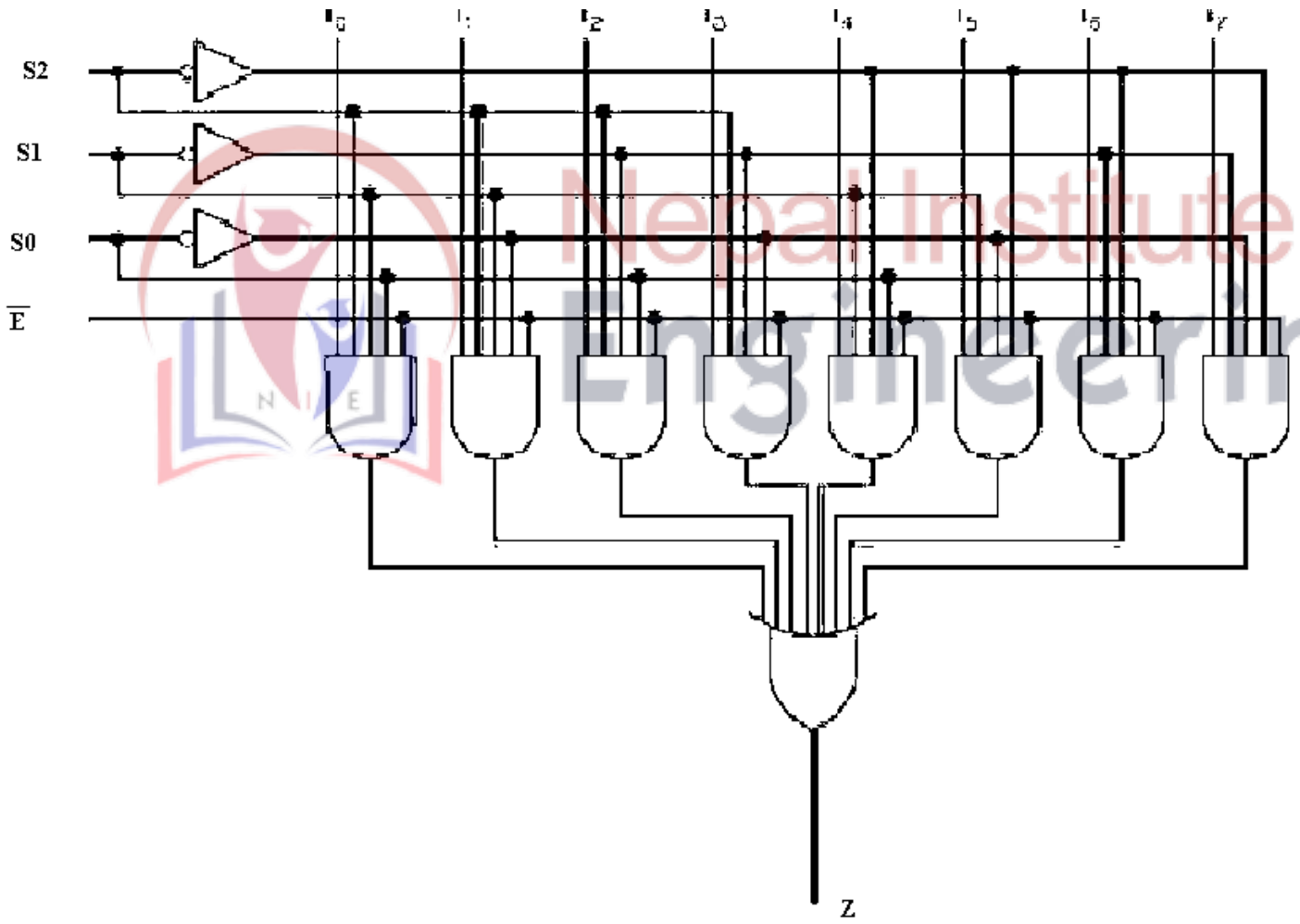
4-to-1 Multiplexer (MUX)





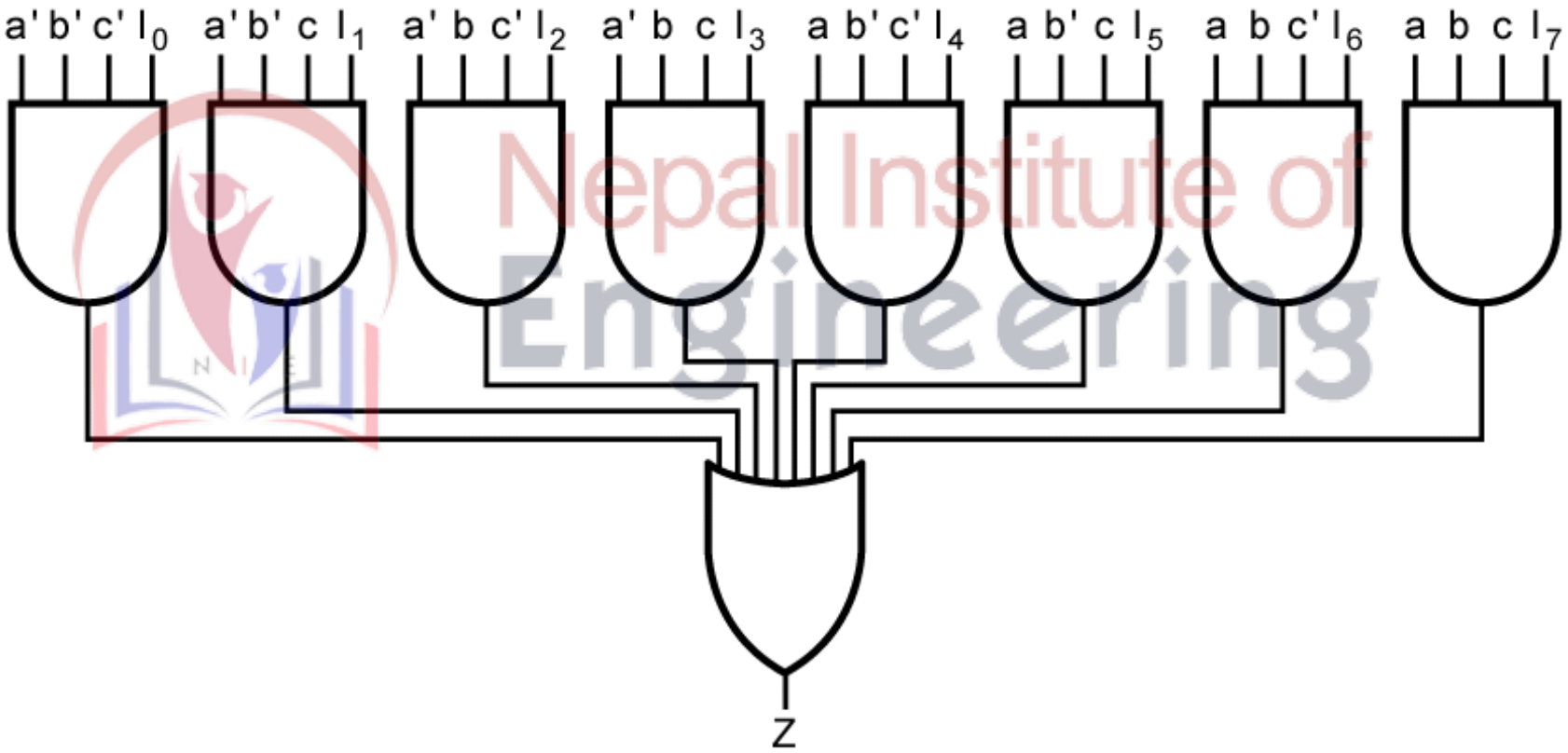
S1	S0	O
0	0	I0
0	1	I1
1	0	I2
1	1	I3

8 : 1 Multiplexer



S_0	S_1	S_3	Z
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3
1	0	0	I_4
1	0	1	I_5
1	1	0	I_6
1	1	1	I_7

8 : 1 Multiplexer



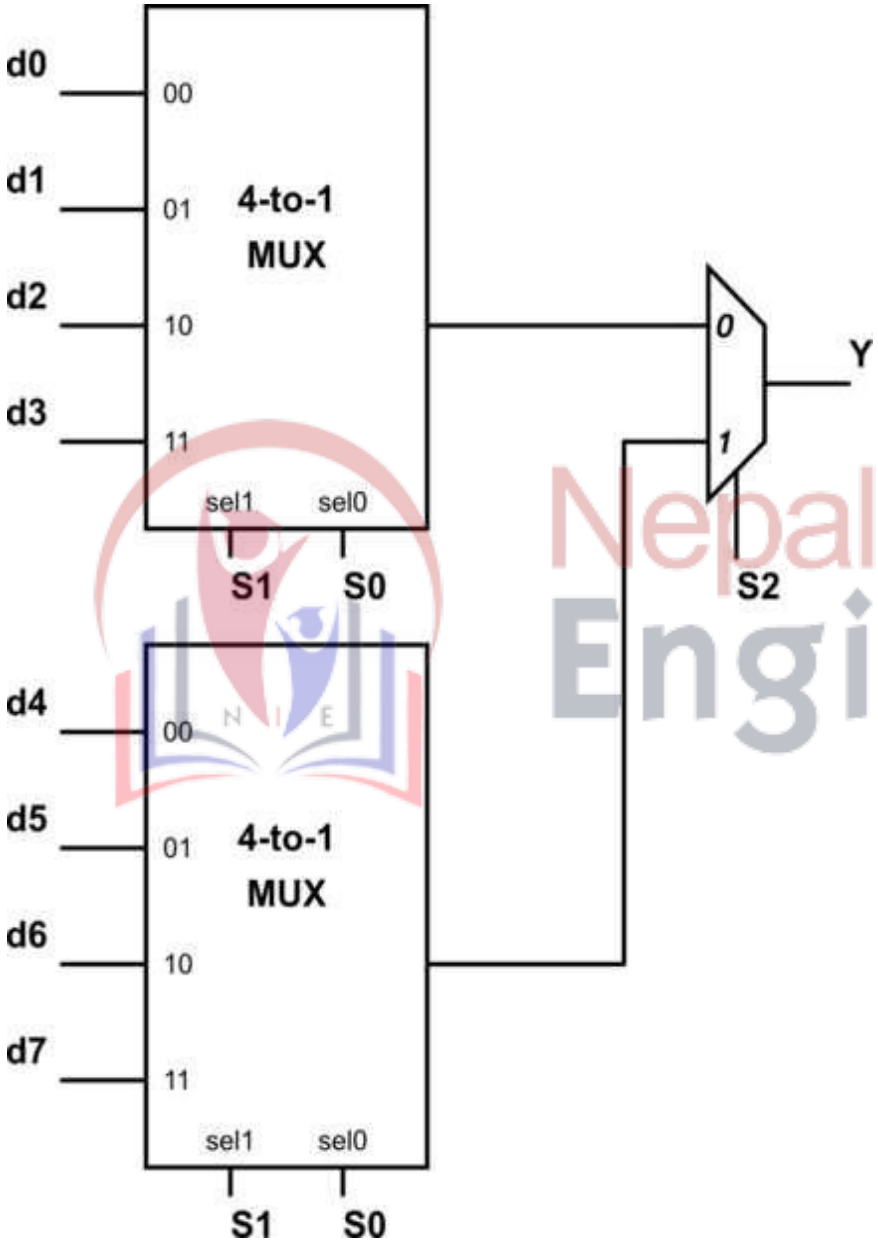
Multiplexers

Exercise:

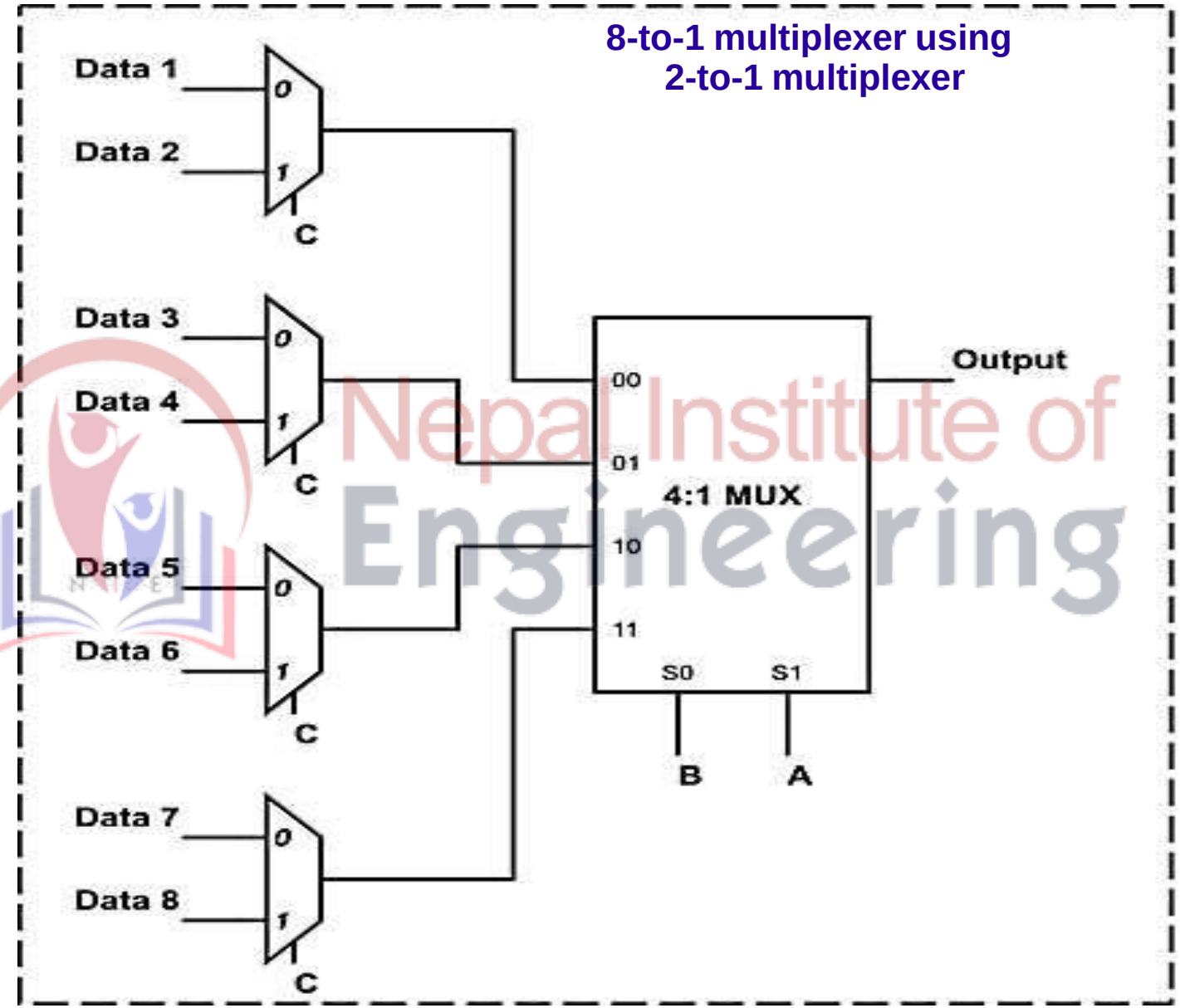
Design an 8-to-1 multiplexer using

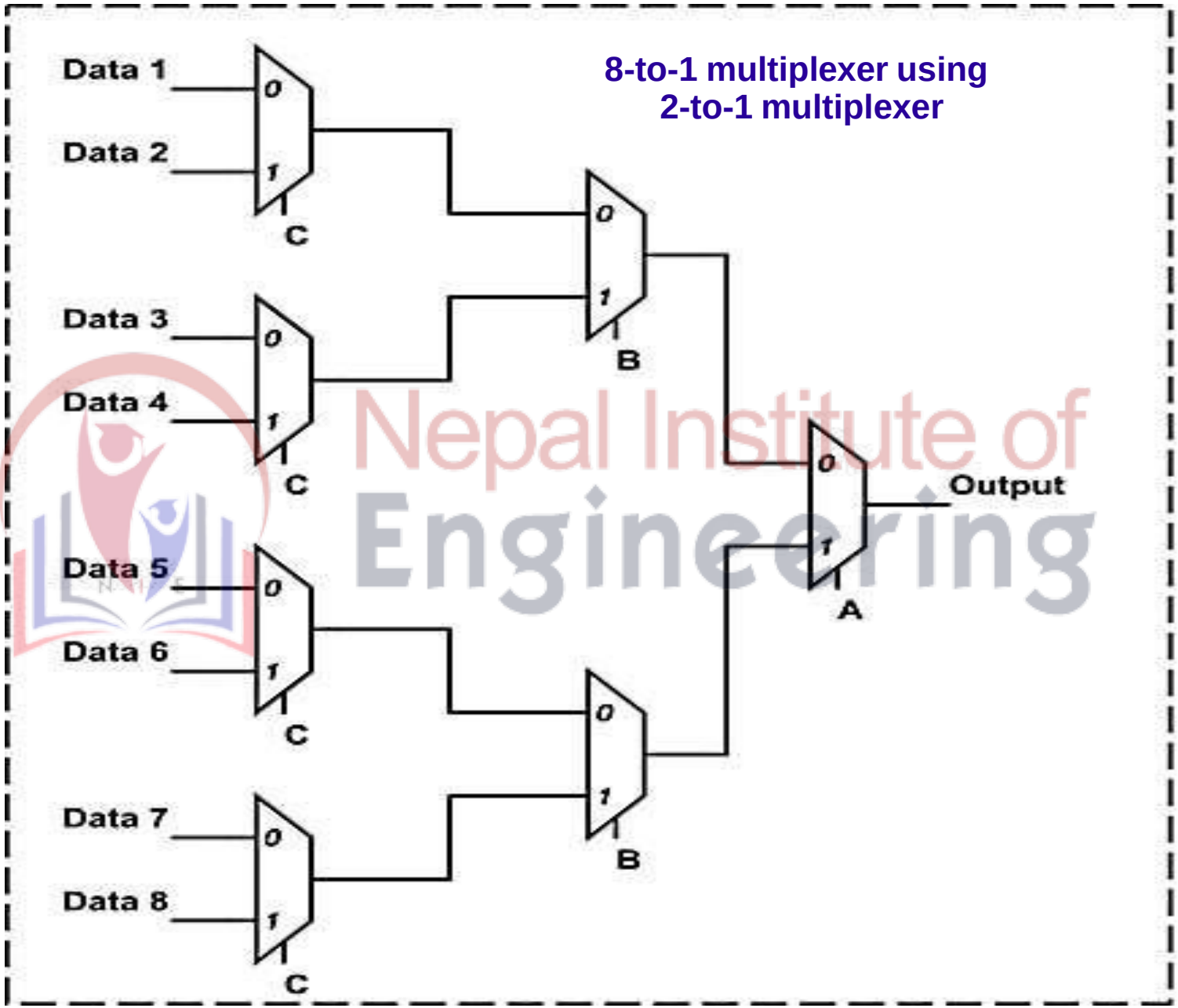
- 4-to-1 multiplexer
- 2-to-1 multiplexer

8-to-1 multiplexer using 4-to-1 multiplexer



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



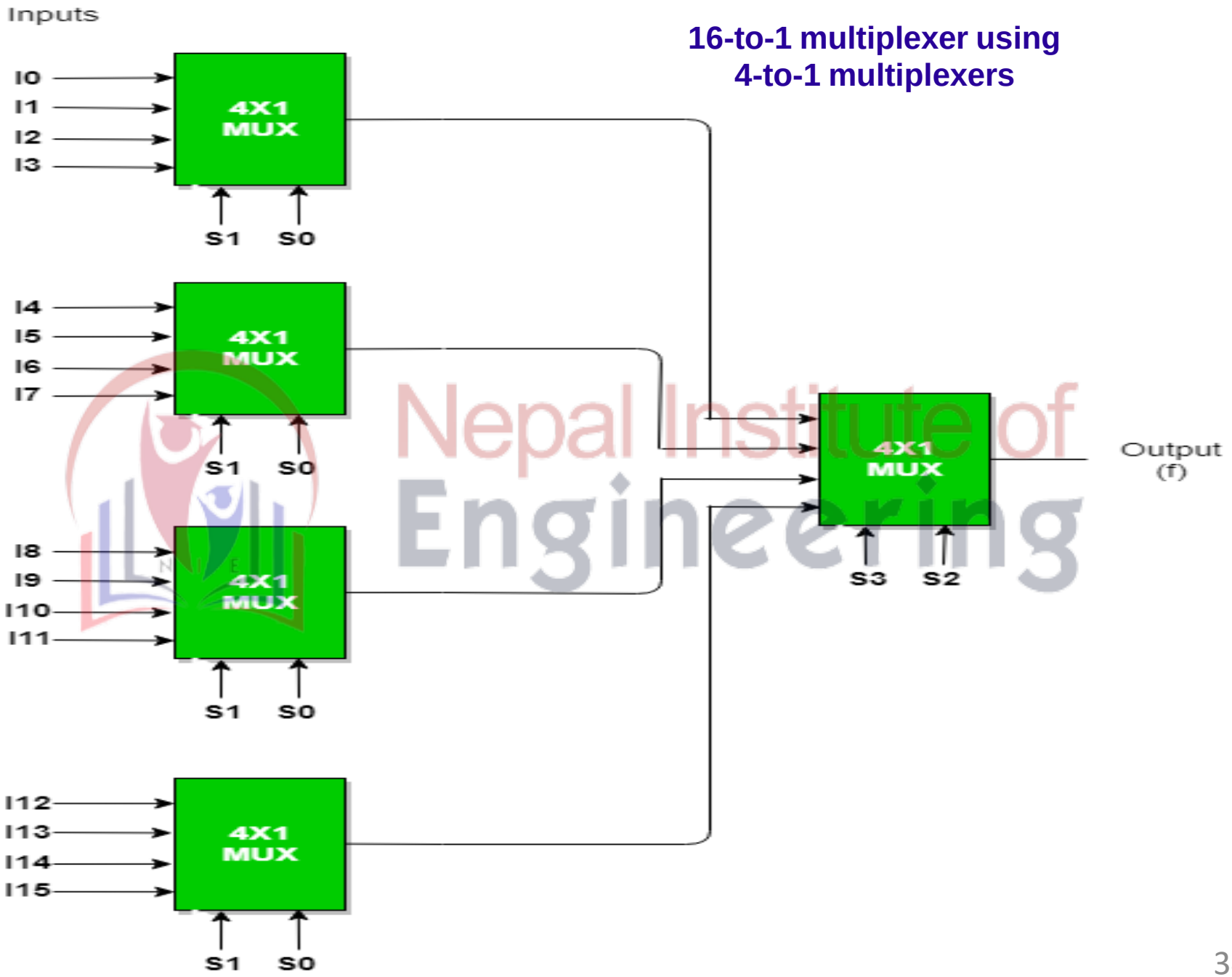


Multiplexers

Exercise:

Design a 16-to-1 multiplexer using 4-to-1 multiplexers only.

16-to-1 multiplexer using 4-to-1 multiplexers



Enable Input

- It's often desirable to add an enable input *EN* to a multiplexer. An enable input makes the multiplexer operate.
- It is like an on switch of your computer, when you connect the power supply to cpu it has provision to start functioning.
- A digital logic circuit may have more than 1 enable pins.
- When *EN = 0*, the output LOW (depending on the specific device).
- When *EN = 1*, the multiplexer performs its operation depending on the selection line.

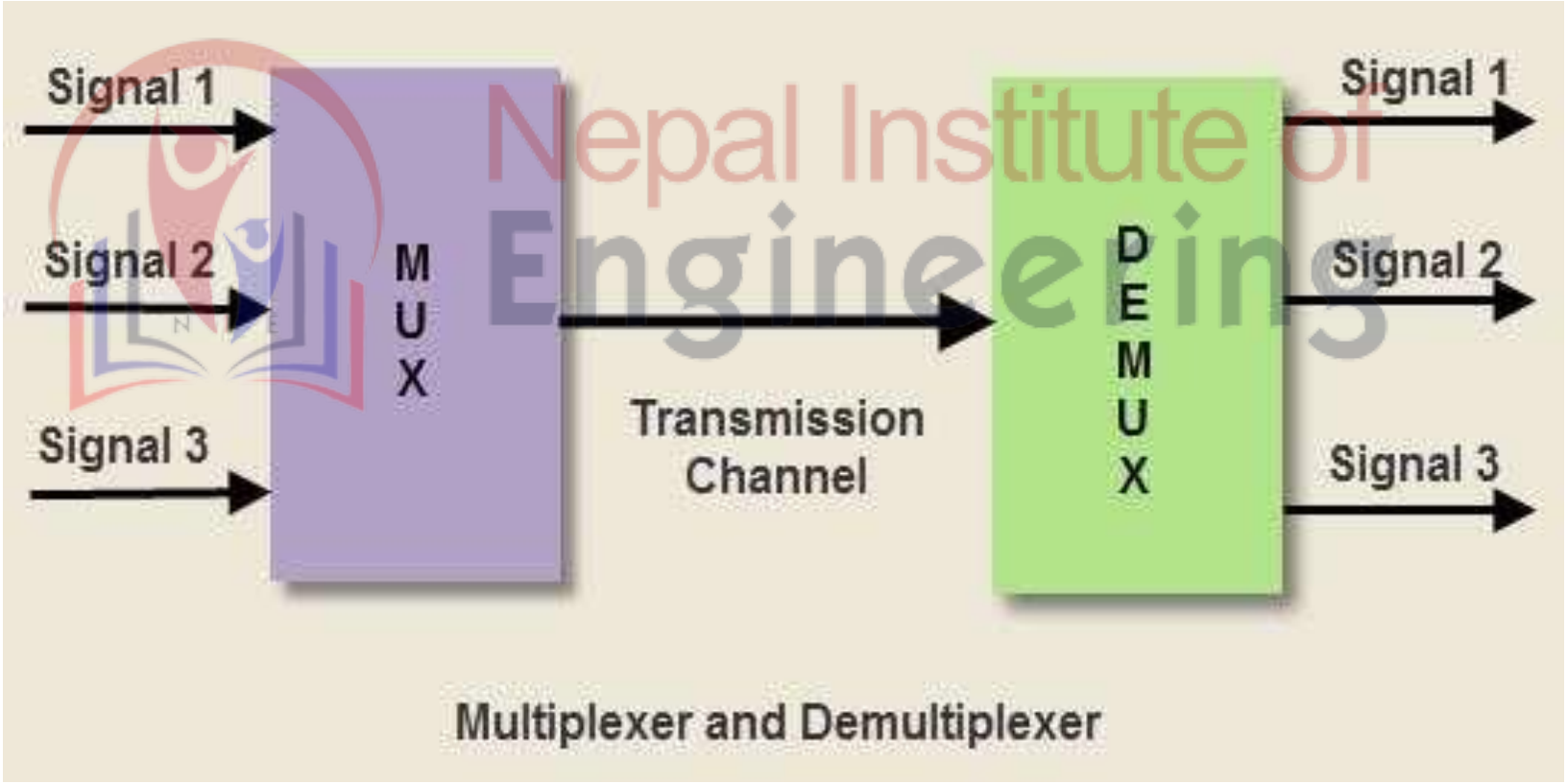
Demultiplexers
(Data Distributor)

A DEMULTIPLEXER (DEMUX) basically reverses the multiplexing function. It takes data from one line and distributes them to a given number of output lines.

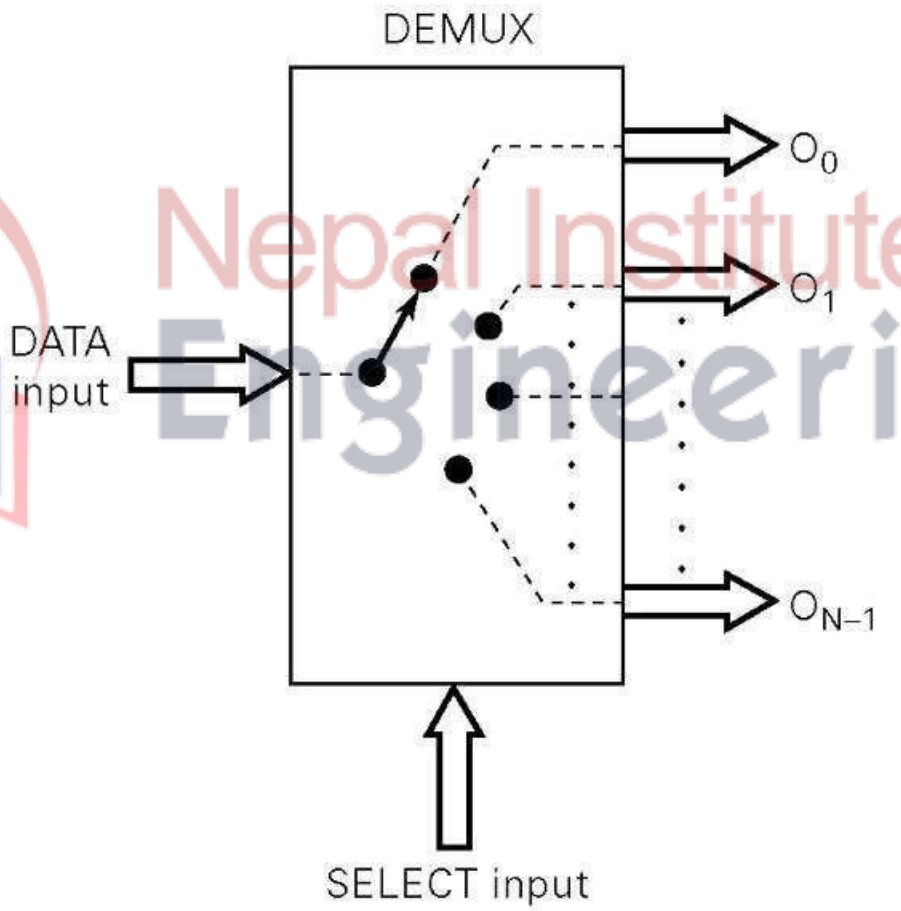
Demultiplexers

- A demultiplexer has
  - N control inputs
  - 1 data input
  - 2N outputs
- A demultiplexer routes (or connects) the data input to the selected output.
  - The value of the control inputs determines the output that is selected.
- A demultiplexer performs the opposite function of a multiplexer.

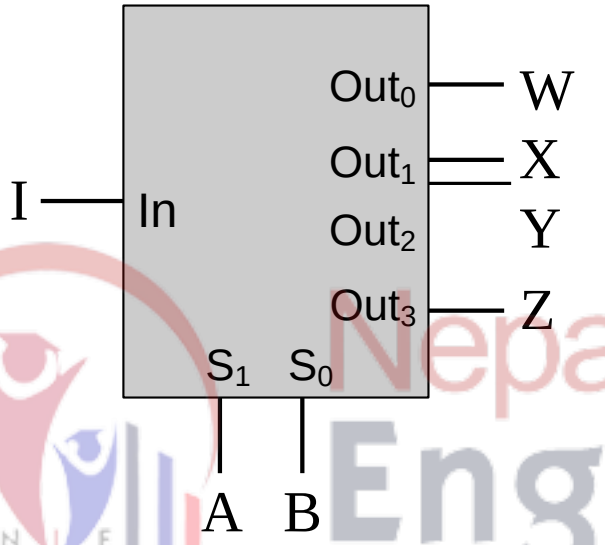
Multiplexer/ Demultiplexer for information transmission



Functional Diagram Of a Demultiplexer



Demultiplexers



$W = A'.B'$

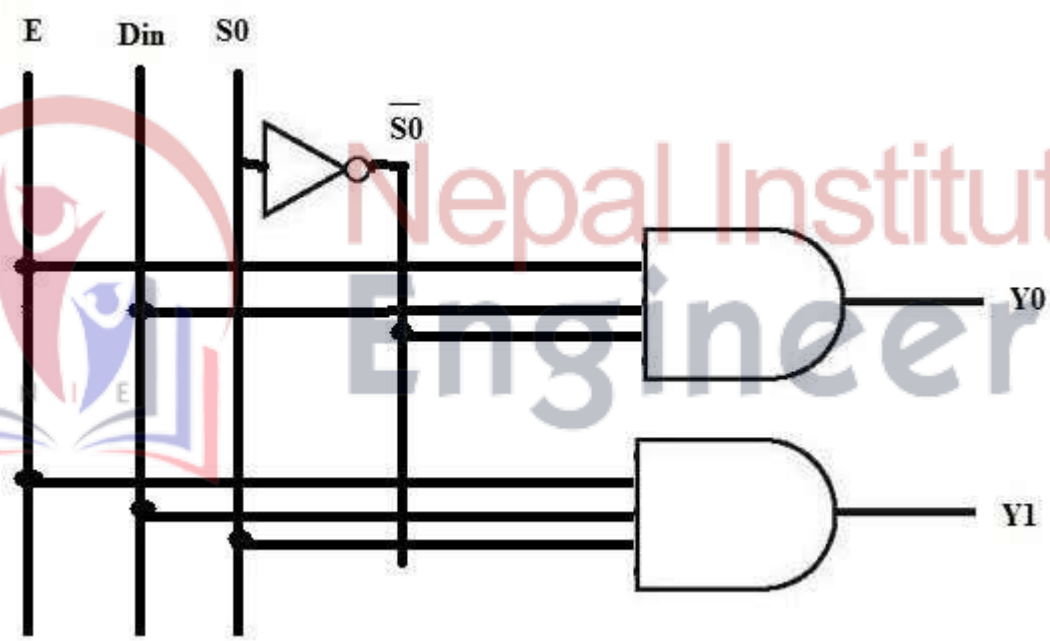
$X = A.B'$

$Y = A'.B$

$Z = A.B$

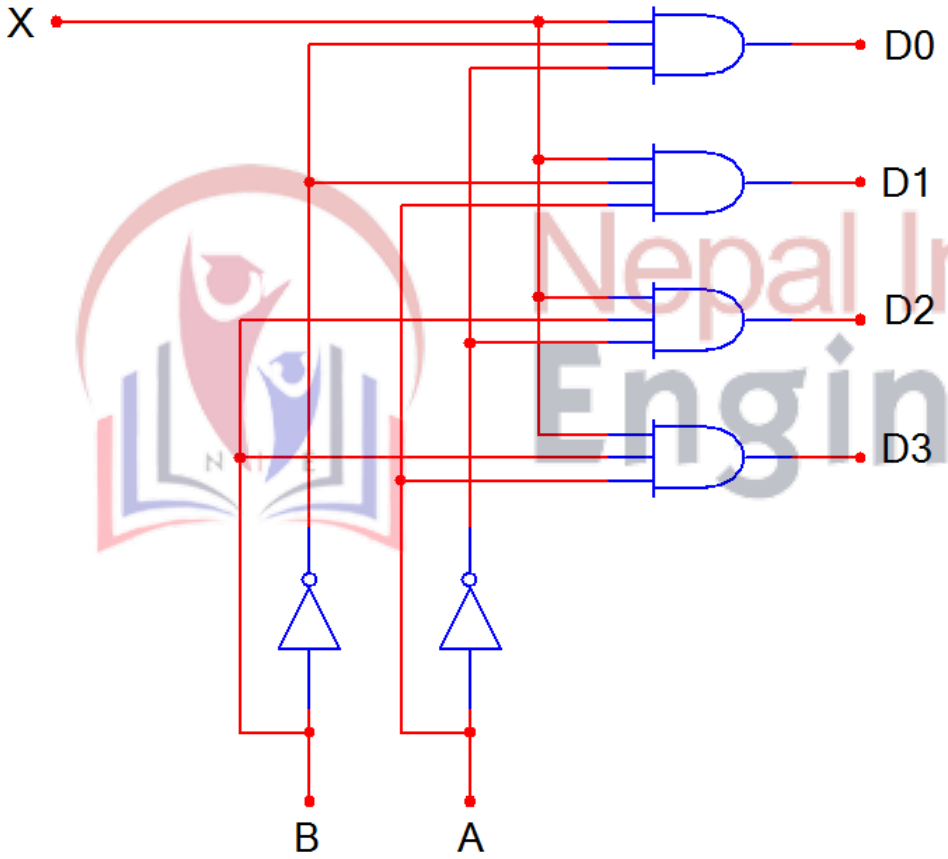
A	B	W	X	Y	Z
0	0	I	0	0	0
0	1	0	I	0	0
1	0	0	0	I	0
1	1	0	0	0	I

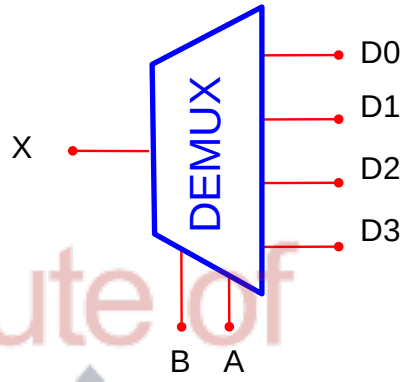
1 : 2 Demultiplexer



S_0	Y_0	Y_1
0	D	0
1	0	D

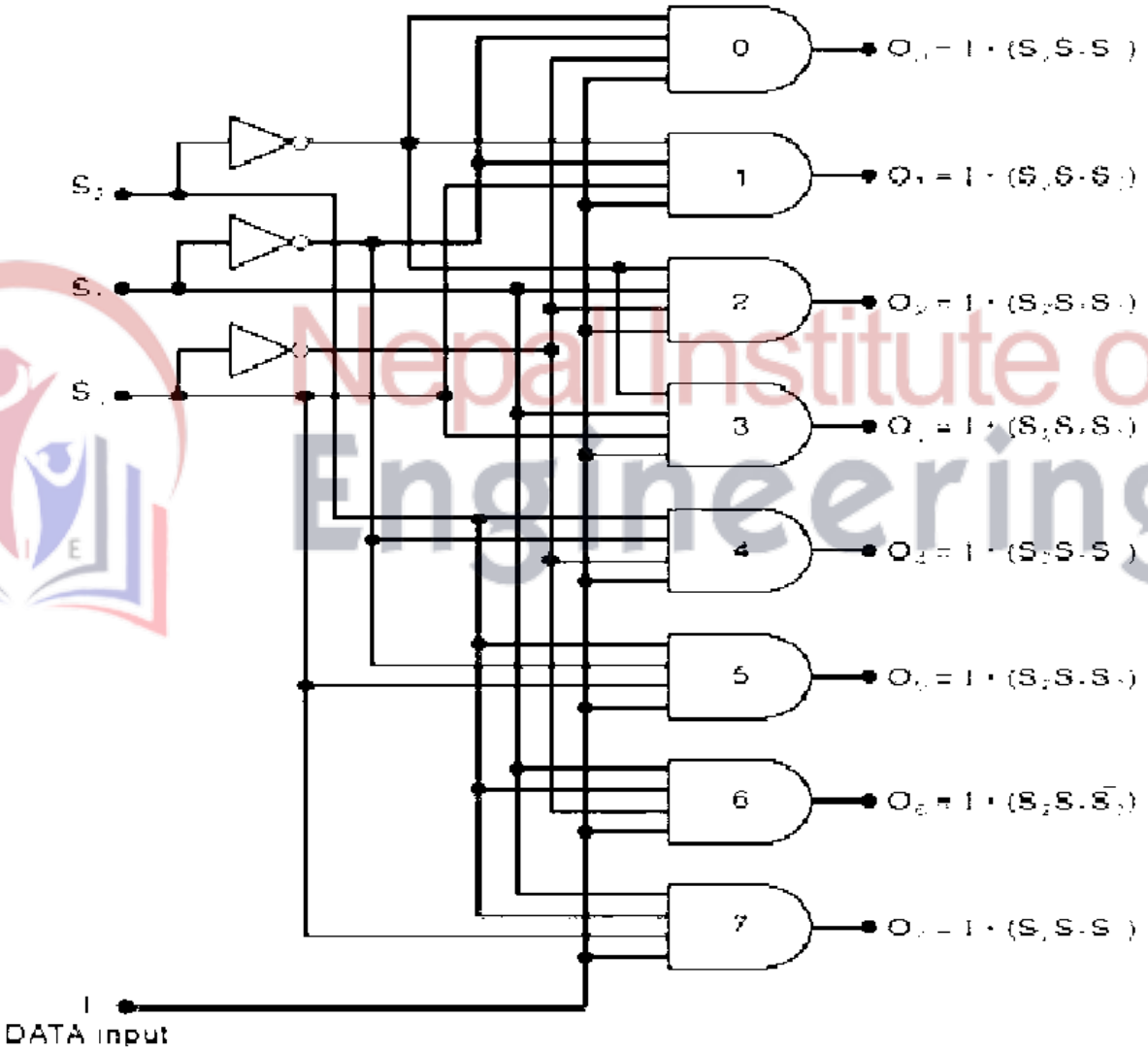
1-to-4 De-Multiplexer (DEMUX)





B	A	D0	D1	D2	D3
0	0	X	0	0	0
0	1	0	X	0	0
1	0	0	0	X	0
1	1	0	0	0	X

1 : 8 Demultiplexer

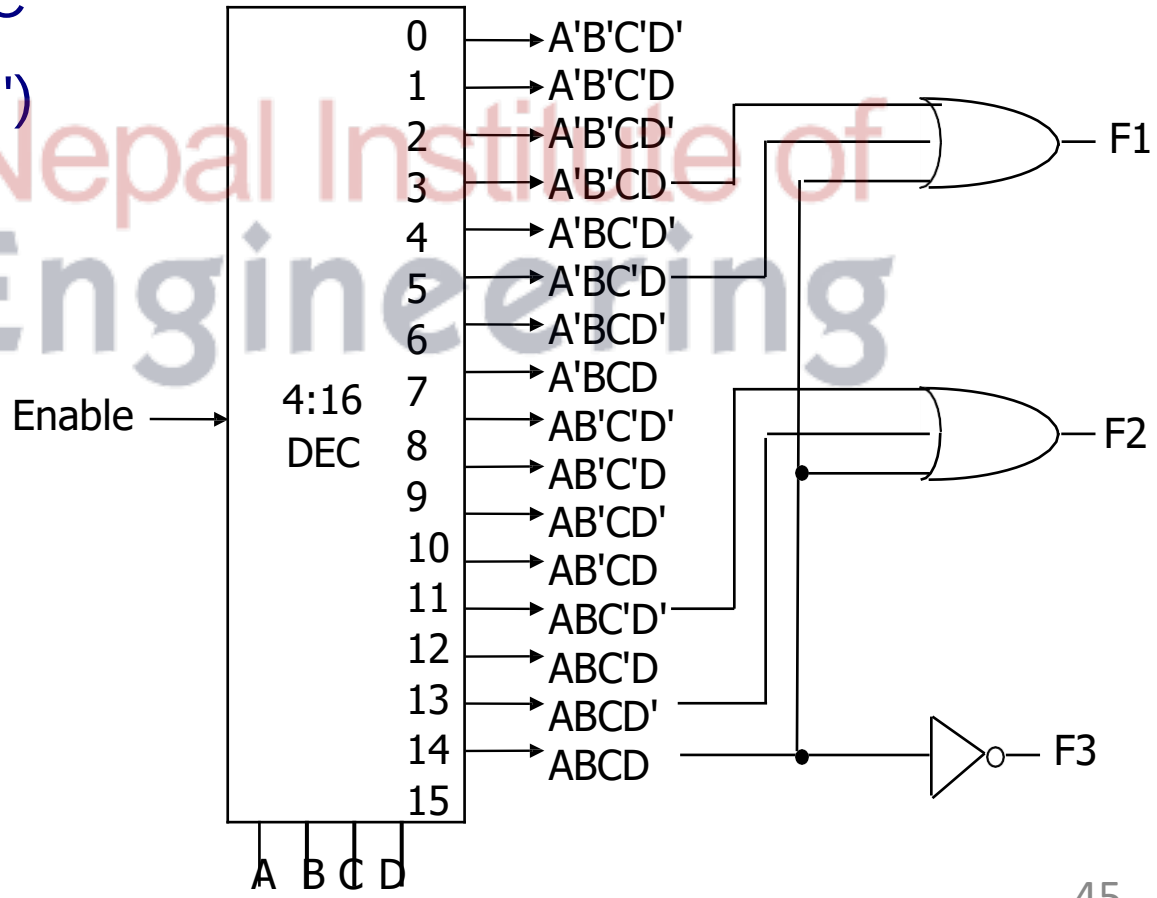


1 : 8 Demultiplexer (Truth Table)

S_0	S_1	S_3	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0	0	0	D	0	0	0	0	0	0	0
0	0	1	0	D	0	0	0	0	0	0
0	1	0	0	0	D	0	0	0	0	0
0	1	1	0	0	0	D	0	0	0	0
1	0	0	0	0	0	0	D	0	0	0
1	0	1	0	0	0	0	0	D	0	0
1	1	0	0	0	0	0	0	0	D	0
1	1	1	0	0	0	0	0	0	0	D

Demultiplexers as General-purpose Logic

- F1 = A' B C' D + A' B' C D + A B C D
- F2 = A B C' D’ + A B C
- F3 = (A' + B' + C' + D')



Sequential Circuit

- A **sequential circuit** is a type of digital logic circuit whose output not only relies on the current input but also depends on the previous output. Hence, a sequential circuit is basically a set of combinational circuits and memory elements connected in a feedback path.
- Memory elements are devices that can save binary data within them. The binary data saved in the memory elements at any given time denotes the state of the sequential circuit. Therefore, the output of a sequential circuit is determined by a time sequence of inputs and previous outputs.
- Sequential circuits are used as the central unit of digital circuits. A common example of a sequential circuit is the finite state machine. The sequential circuits can be further classified into two types namely, Synchronous Sequential Circuits and Asynchronous Sequential Circuits.

Combinational Circuit	Sequential Circuit
A Combinational Circuit is a type of circuit in which the output is independent of time and only relies on the input present at that particular instant.	A Sequential circuit is a type of circuit where output not only relies on the current input but also depends on the previous output.
Since output does not depend on the time instant, no feedback is required for its next output generation.	The output relies on its previous feedback so output of previous input is being transferred as feedback used with input for next output generation.
As the input of current instant is only required in case of Combinational circuit, it is faster and better in performance as compared to that of Sequential circuit.	Sequential circuits are comparatively slower and has low performance as compared to that of Combinational circuit.
No implementation of feedback makes the combinational circuit less complex as compared to sequential circuit.	The implementation of feedback makes sequential circuit more complex as compared to combinational circuit.
The elementary building blocks of a combinational circuit are its logic gates.	The building blocks of a sequential circuit are the logic gates along with flip flops.

Flip Flop

- Flip flop is formed using logic gates, which are in turn made of transistors. Flip flop are basic building blocks in the memory of electronic devices. Each flip flop can store one bit of data.
- These are also called as sequential logic circuits.
- Sequential Logic circuits
- Latches
- Flip – flops have two stable states and hence they are bistable multivibrators. The two stable states are High (logic 1) and Low (logic 0).
- The term flip – flop is used as they can switch between the states under the influence of a control signal (clock or enable) i.e. they can 'flip' to one state and 'flop' back to other state.
- Flip – flops are a binary storage device because they can store binary data (0 or 1).
- Flip – flops are edge sensitive or edge triggered devices i.e. they are sensitive to the transition rather than the duration or width of the clock signal.
- They are also known as signal change sensitive devices which mean that the change in the level of clock signal will bring change in output of the flip flop.
- A Flip – flop works depending on clock pulses.
- Flip flops are also used to control the digital circuit's functionality. They can change the operation of a digital circuit depending on the state.

- Some of the most common flip – flops are SR Flip – flop (Set – Reset), D Flip – flop (Data or Delay), JK Flip – flop and T Flip – flop.

LATCH vs FLIP-

LATCH	FLIP – FLOP
Latches do not require clock signal.	Flip – flops have clock signals
A latch is an asynchronous device.	A flip – flop is a synchronous device.
Latches are transparent devices i.e. when they are enabled, the output changes immediately if the input changes.	A transition from low to high or high to low of the clock signal will cause the flip – flop to either change its output or retain it depending on the input signal.
A latch is a Level Sensitive device (Level Triggering is involved).	A flip – flop is an edge sensitive device (Edge Triggering is involved).
Latches are simpler to design as there is no clock signal (no careful routing of clock signal is required).	When compare to latches, flip – flops are more complex to design as they have clock signal and it has to be carefully routed. This is because all the flip – flops in a design should have a clock signal and the delay in the clock reaching each flip – flop must be minimum or negligible.
The operation of a latch is faster as they do not have to wait for any clock signal.	Flip - flops are comparatively slower than latches due to clock signal.
The power requirement of a latch is less.	Power requirement of a flip – flop is more.
A latch works based on the enable signal.	A flip – flop works based on the clock signal.

Types of flip flops

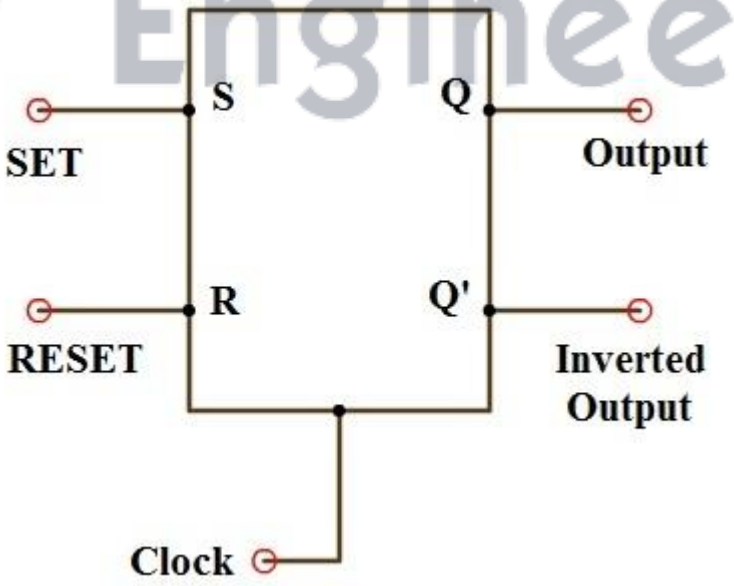
Based on their operations, flip flops are basically 4 types.

They are

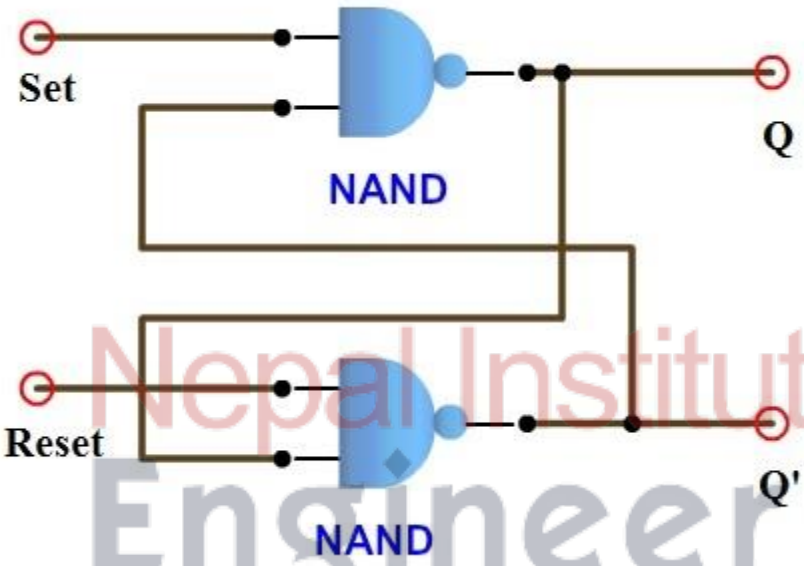
1. R-S flip flop
2. D flip flop
3. J-K flip flop
4. T flip flop

S-R Flip Flop

The S-R flip-flop is basic flip-flop among all the flip-flops. All the other flip flops are developed after SR-flip-flop.
S-R stands for SET and RESET. This can also be called RS flip-flop. Difference is RS is inverted SR flip-flop.
Any flip flop can be build using logic gates. NAND and NOR gates were used as they are universal gates.



SR flip-flop using NAND gate

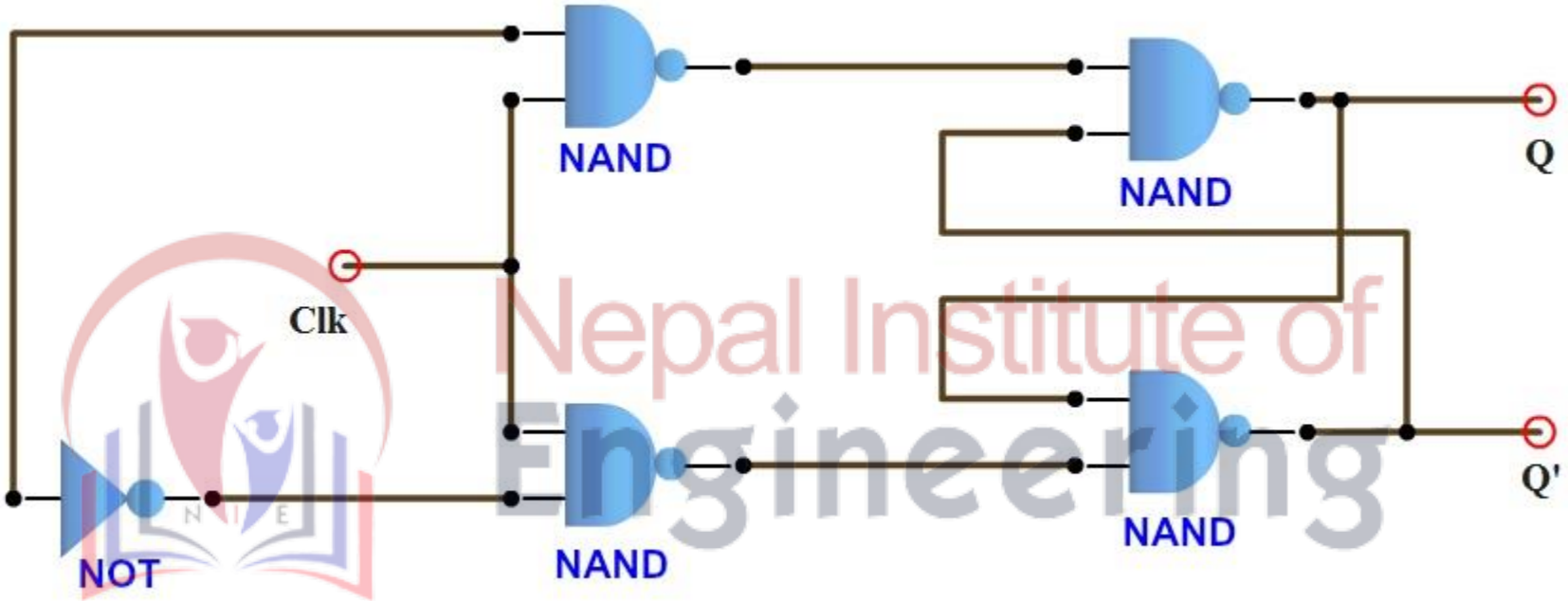


Sno	S	R	Q	Q'	State
1	1	0	1	0	Q is set to 1
2	1	1	1	0	No change
3	0	1	0	1	Q' is set to 1
4	1	1	0	1	No change
5	0	0	1	1	Invalid

D flip flop

In the SR flip flop an uncertain state occurred. This can be avoided by using D flip flop. Here D stands for “Data”.
It is constructed from SR flip flop. The two inputs (S &R) of the clocked SR flip flop are connected to an inverter.
It is one of the most widely used flip – flops. It has a clock signal (Clk) as one input and Data (D) as other. There are two outputs and these outputs are complement to each other.

D flip fop



Clk	D	Q	Q'	State
0	0	Q	Q'	No change in state
1	0	0	1	Resets Q to 0
1	1	1	1	Sets Q to 1

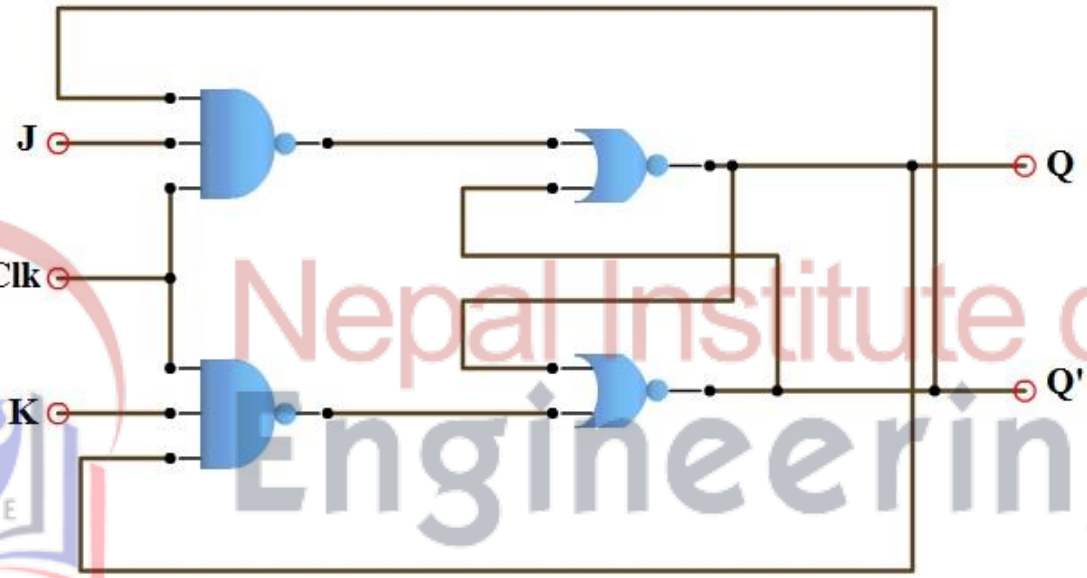
Working

- D flip flop will work depending on the clock signal.
- When the clock is low there will be no change in the output of the flip flop i.e. it remembers the previous state.
- When the clock signal is high and if it receives any data on its data pin, it Changes the state of output.
- When data is high Q reset to 0,while Q is set to 0 if data is low.

J-K Flip Flop

- A JK flip – flop is a modification of SR flip – flop. In this the J input is similar to the set input of SR flip – flop and the K input is similar to the reset input of SR flip – flop.
- The condition J = K = 1 which is not allowed in SR flip – flop (S = R = 1) is interpreted as a toggle command.
- The JK flip flop has
- Two data inputs J and K.
- One clock signal input (CLK).
- Two outputs Q and Q'.
- Working
- When J is low and K is low, then Q returns its previous state value i.e. it holds the current state.
- When J is low and K is high, then flip – flop will be in reset state i.e. Q = 0, Q' =1.
- When J is high and K is low then flip – flop will be in set state i.e. Q = 1, Q' =0.
- When J is high and K is high then flip – flop will be in Toggle state or flip state. This means that the output will complement to the previous state value.

J-K Flip Flop

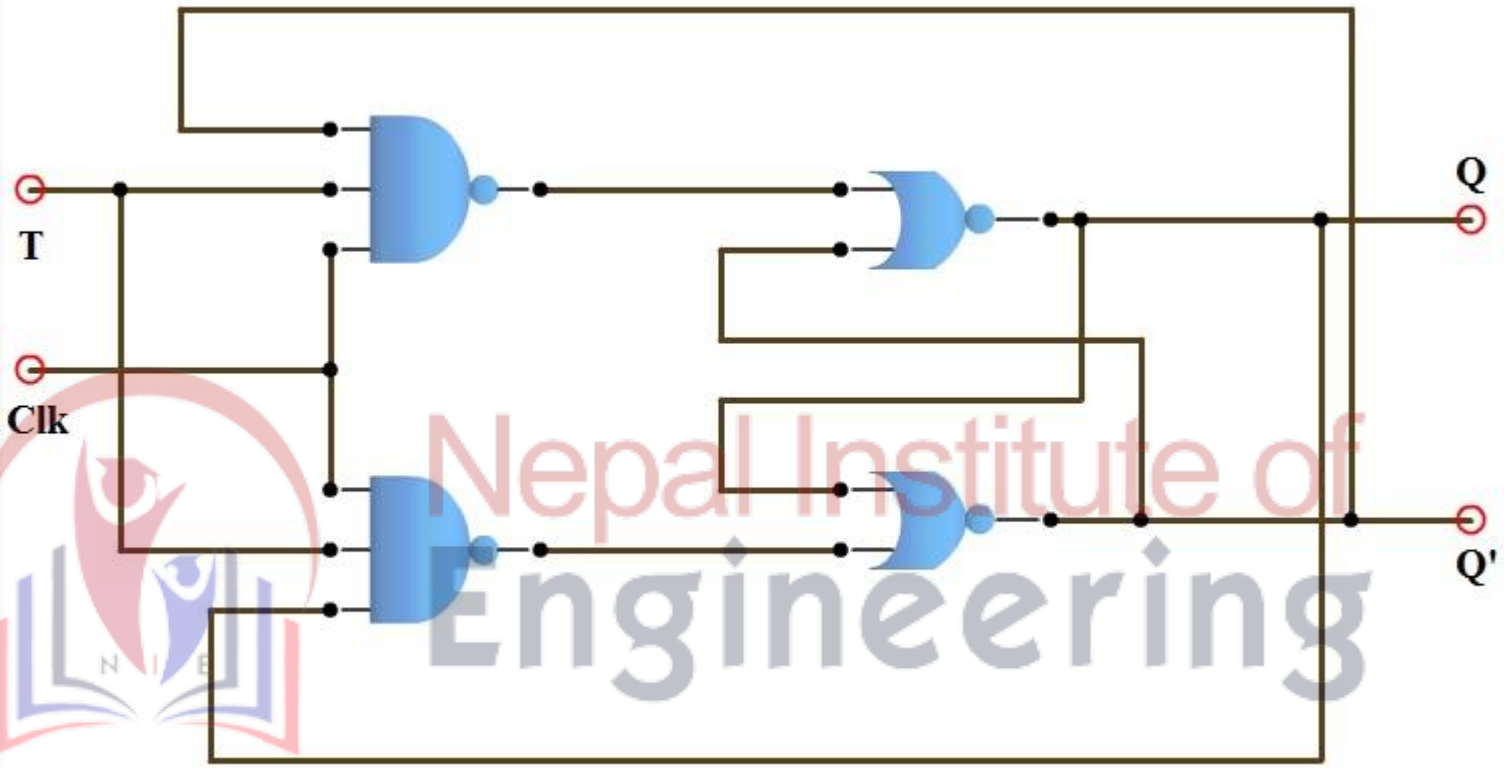


Clk	J	K	Q	Q'	State
1	0	0	Q	Q'	No change in state
1	0	1	0	1	Resets Q to 0
1	1	0	1	0	Sets Q to 1
1	1	1	-	-	Toggles

T Flip Flop

T flip flop is also known as “Toggle Flip – flop”. Toggle is to change the output to complement of the previous state in the presence of clock input signal.
The T flip flop has
•T input.
•One clock signal input (CLK).
•Two outputs Q and Q’.

Working
The operation of the T flip flop is explained below.
When the T input is low, then the next sate of the T flip – flop is same as the present state i.e. it holds the current state.
•T = 0 and present state = 0 then the next state = 0.
•T = 0 and present state = 1 then the next state = 1.
When the T input is high, then the next sate of the T flip – flop is toggled i.e. it is same as the complement of present state on clock transition.
•T = 1 and present state = 0 then the next state = 1.
•T = 1 and present state = 1 then the next state = 0.



T	Q	Q'
0	0	0
1	0	1
0	1	0
1	1	0

Master-Slave Flip-Flop

- A **JK flip flop** is a type of 1-bit memory element having inputs namely J and K, one clock input, and two output specified by Q and Q'.
- The JK flip flop is an improved version of SR flip flop which does not have forbidden state. To avoid the forbidden or indeterminate state, the outputs of the JK flip flop are fed back to its inputs.
- However, due to these feedback paths, a new problem is raised in the circuit, which is called race around condition. Race around condition in the JK flip is a major problem in which the outputs of flip flop are toggled continuously till the end of applied clock signal.
- To avoid the problem of race around condition in JK flip flop, we use the JK flip flop in the **Master and Slave Mode.** Hence, the JK flip flop is called **Master-Slave Flip Flop**.

Master-Slave JK Flip-Flop

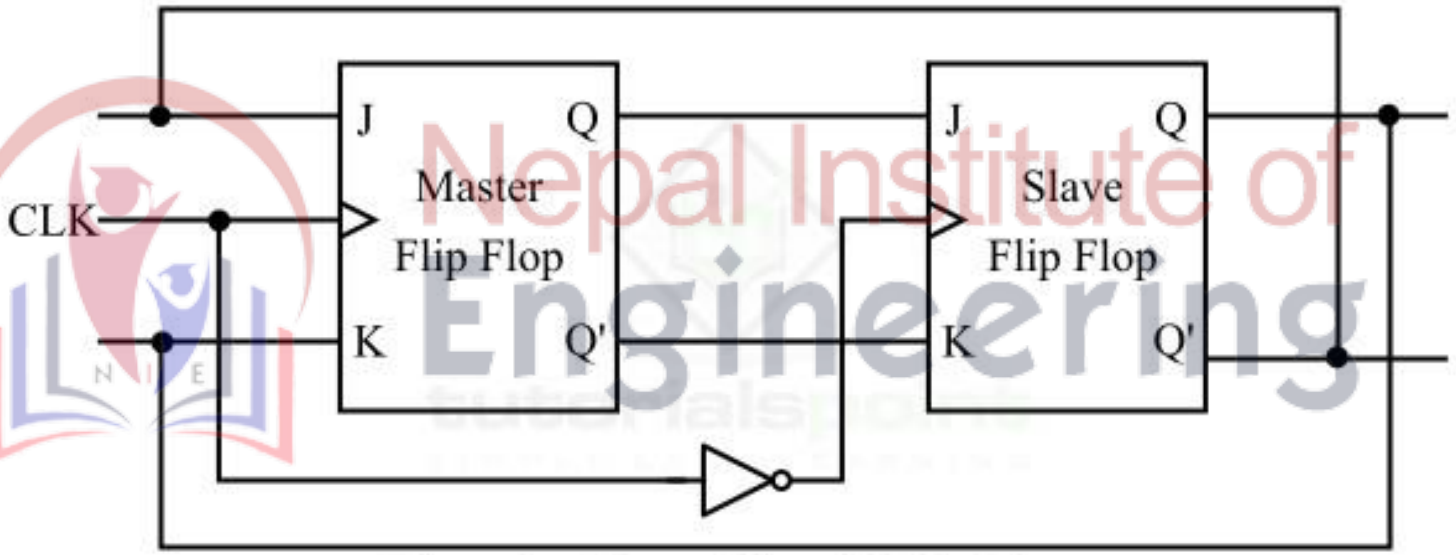


Figure 1 - Master-Slave JK Flip Flop

Master-Slave Flip-Flop

- **Master Slave JK Flip Flop** is a combination of two JK flip flops which are connected in the cascaded manner.
- In this combination of two JK flip flop, one acts as a **master flip flop** and the other acts as a **slave flip flop.**
- In this master-slave flip flop, the outputs of the master JK flip flop are connected to the inputs of the slave JK flip flop. The outputs of the slave flip flop are fed back to the inputs of the master JK flip flop.
- In the master-slave JK flip flop, a NOT gate (Inverter) is also used which is connected to clock signal in a manner that the inverted clock signal is applied to the slave flip flop.
- Therefore, when clock signal to master flip flop is 0, then for slave flip flop the clock signal is 1, and if the clock signal to master flip flop is 1, then for the slave flip flop it 0.