CSE211

Computer Organization and Design

Lecture : 3 Tutorial: 1 Practical: 0 Credit: 4

Unit 1 : Basics of Digital Electronics

- Introduction
- Logic Gates
- Combinational / Sequential circuits
- **Flip Flops**
- **Registers**
- Decoder / Encoder
- Multiplexers / Demultiplexer
- Binary counters

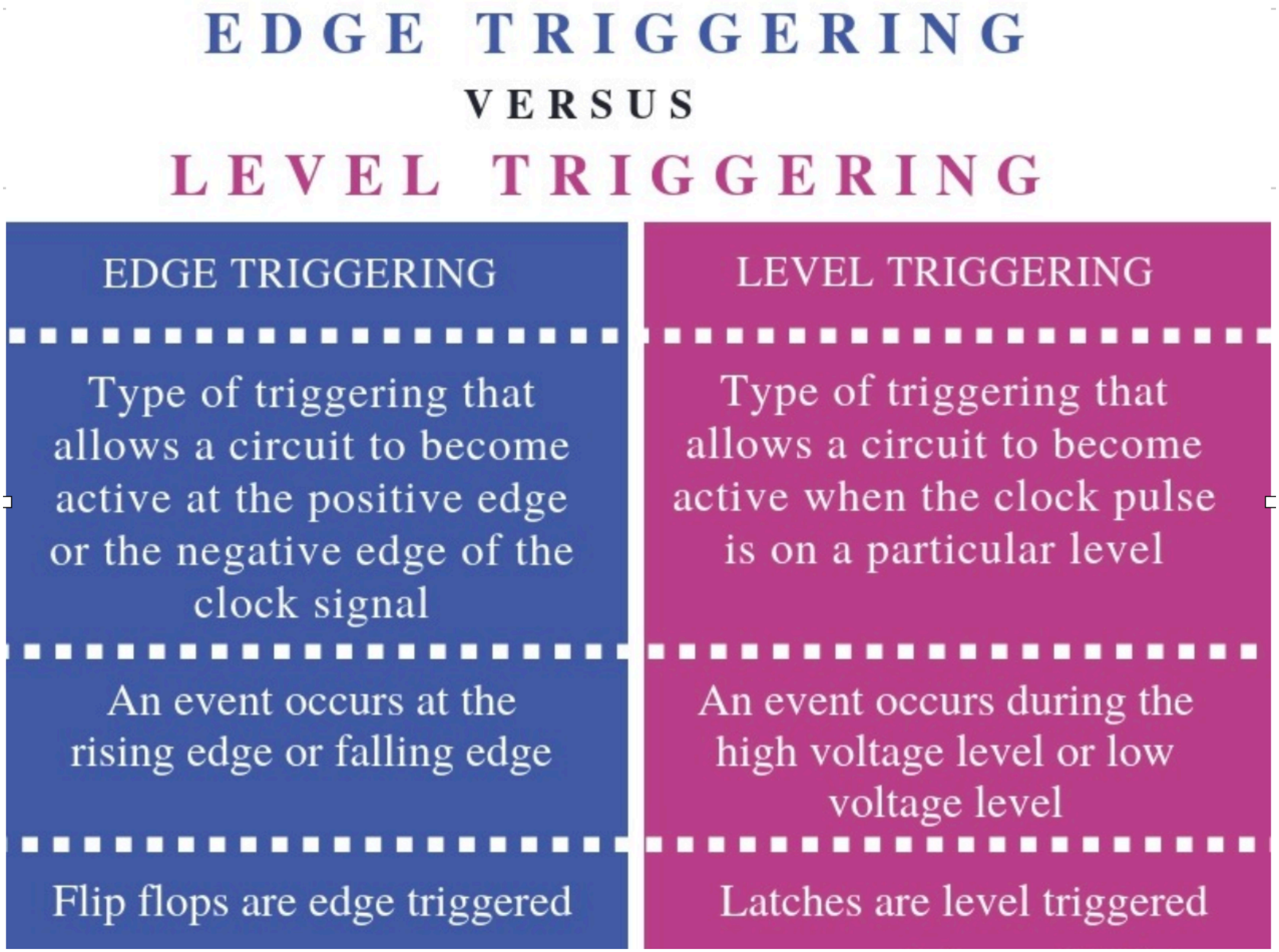
Whose operations are more faster among the following?
a) Combinational circuits
b) Sequential circuits
c) Latches
d) Flip-flops

- Answer: a
- Explanation: Combinational circuits are often faster than sequential circuits. Since, the combinational circuits do not require memory elements whereas the sequential circuits need memory devices to perform their operations in sequence. Latches and Flip-flops come under sequential circuits.

Table 1

How		many		types	of	sequential	circuits	are?
a)	2							
b)	3							
c)	4							
d)	5							

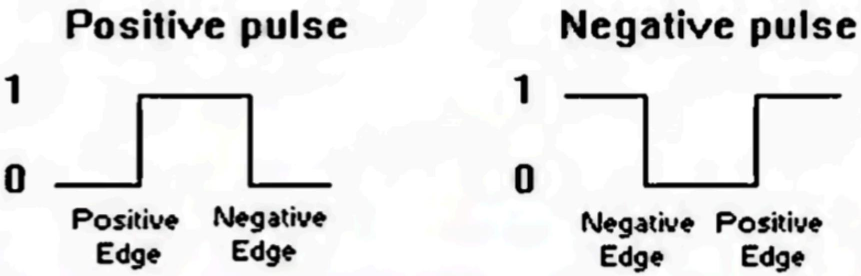
- Answer: a
- Explanation: There are two type of sequential circuits viz., (i) synchronous or clocked and (ii) asynchronous or unclocked. Synchronous Sequential Circuits are triggered in the presence of a clock signal, whereas, Asynchronous Sequential Circuits function in the absence of a clock signal.



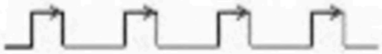
EDGE TRIGGERING VERSUS LEVEL TRIGGERING

EDGE TRIGGERING	LEVEL TRIGGERING
Type of triggering that allows a circuit to become active at the positive edge or the negative edge of the clock signal	Type of triggering that allows a circuit to become active when the clock pulse is on a particular level
An event occurs at the rising edge or falling edge	An event occurs during the high voltage level or low voltage level
Flip flops are edge triggered	Latches are level triggered

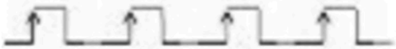
The movement of a trigger pulse is always from a 0 to 1 and then 1 to 0 of a signal. Thus it takes two transitions in a single signal. When it moves from 0 to 1 it is called a positive transition and when it moves from 1 to 0 it is called a negative transition. To understand more take a look at the images below.



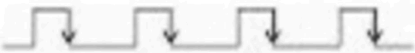
Definition of clock pulse transition



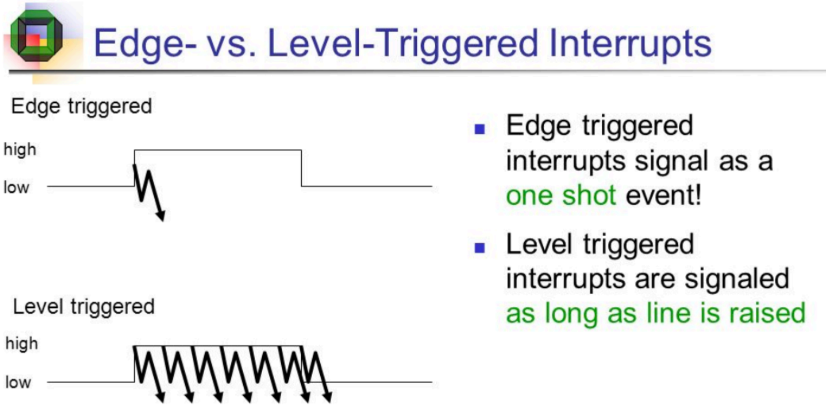
Level Triggering



Positive-Edge Triggering



Negative-Edge Triggering



Edge- vs. Level-Triggered Interrupts

- Edge triggered interrupts signal as a one shot event!
- Level triggered interrupts are signaled as long as line is raised

FLIP FLOPS

Flip-flop is a circuit that maintains a state until directed by input to change the state.

A basic flip-flop can be constructed using four-NAND or four-NOR gates.

Types of flip-flops:

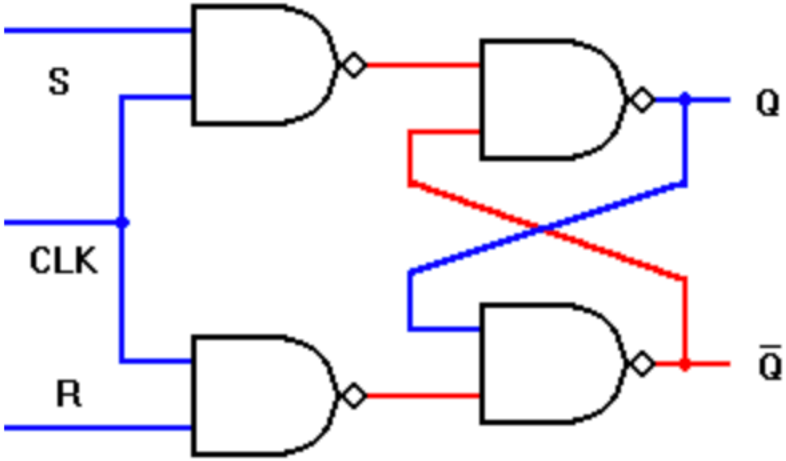
1. SR Flip Flop
2. JK Flip Flop
3. D Flip Flop
4. T Flip Flop

Applications of Flip-Flops

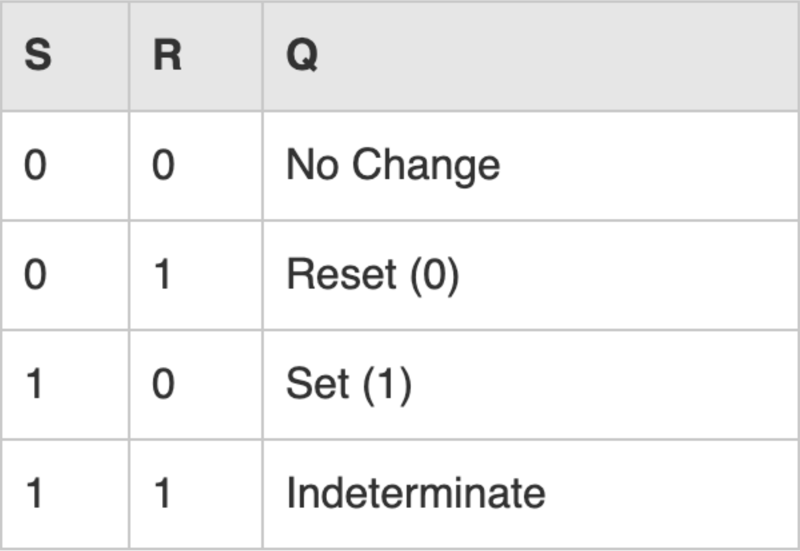
Various types of flip-flops are being used in digital electronic circuits and the applications of Flip-flops are as specified below.

- Counters
- Frequency Dividers
- Shift Registers
- Storage Registers
- Bounce elimination switch
- Data storage
- Data transfer
- Latch
- Registers
- Memory

S(set)/R(reset) Flip Flop

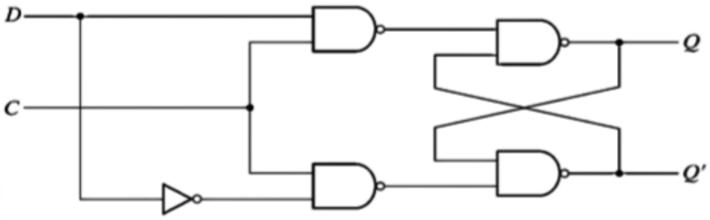


- Excitation table



S	R	Q
0	0	No Change
0	1	Reset (0)
1	0	Set (1)
1	1	Indeterminate

D-FLIP FLOP

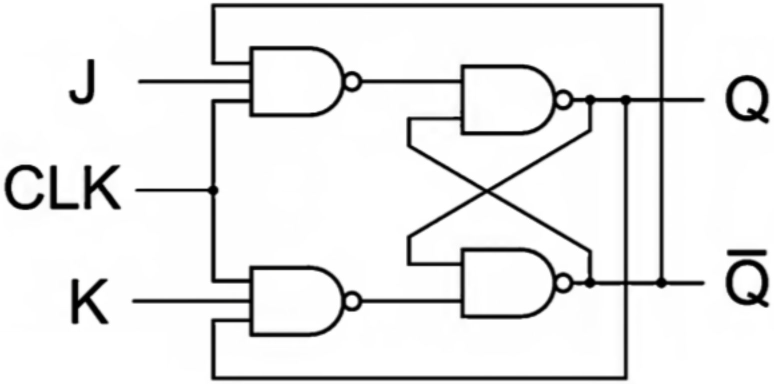


(a) Logic diagram

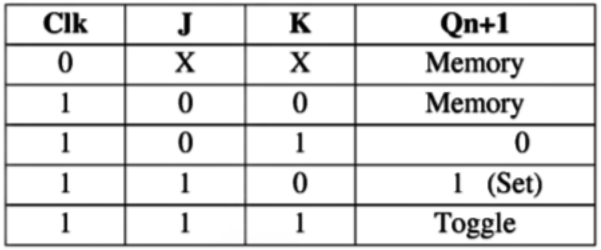
C	D	Next state of Q
0	X	No change
1	0	$Q = 0$; Reset state
1	1	$Q = 1$; Set state

(b) Function table

J-K FLIP FLOP

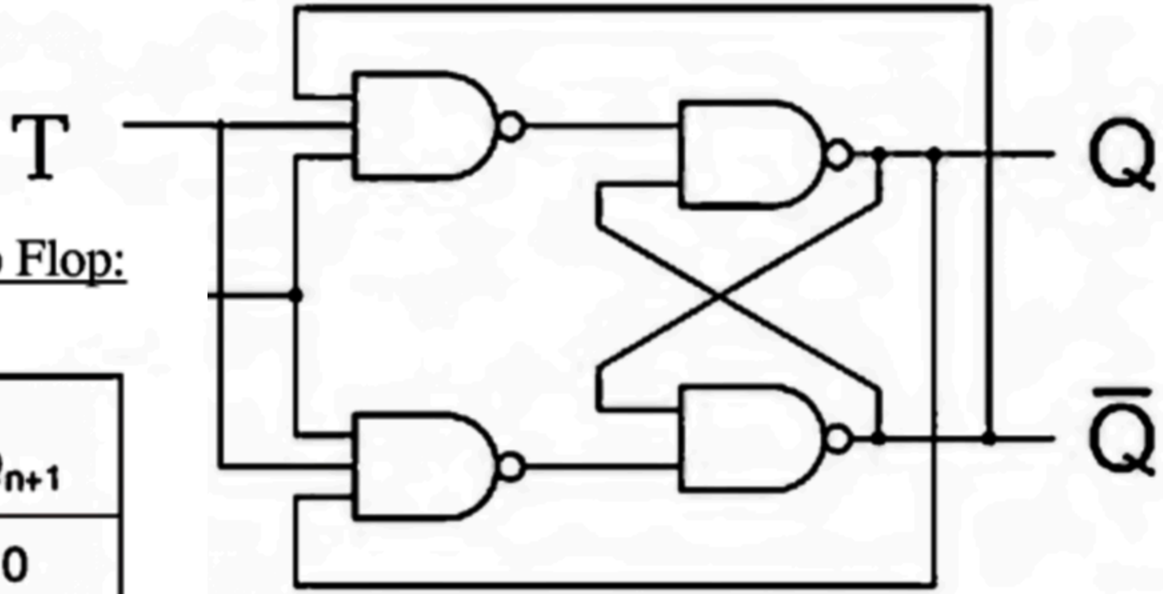


Characteristic table:



Clk	J	K	Qn+1
0	X	X	Memory
1	0	0	Memory
1	0	1	0
1	1	0	1 (Set)
1	1	1	Toggle

T-FLIP FLOP



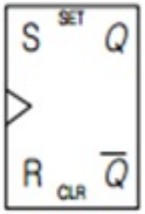
,Excitation Table for T Flip Flop:

Q_n	T	Q_{n+1}
0	0	0
0	1	1
1	0	1
1	1	0

1-6 Flip-Flops

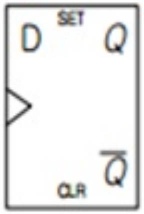
Combinational Circuit = Gate
Sequential Circuit = Gate + F/F

- Flip-Flop
  - The *storage elements* employed in clocked *sequential circuit*
  - A binary cell capable of storing one bit of information
- SR*(Set/Reset)* F/F



S	R	Q(t+1)	
0	0	Q(t)	no change
0	1	0	clear to 0
1	0	1	set to 1
1	1	?	Indeterminate

- D*(Data)* F/F

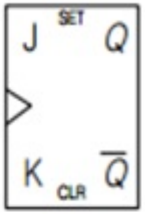


D	Q(t+1)	
0	0	clear to 0
1	1	set to 1

  - "no change" condition

1) Disable Clock
2) Feedback output into input

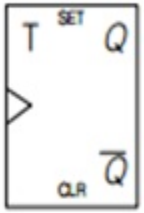
- JK*(Jack/King)* F/F



J	K	Q(t+1)	
0	0	Q(t)	no change
0	1	0	clear to 0
1	0	1	set to 1
1	1	Q(t)'	Complement

  - JK F/F is a refinement of the SR F/F
  - The indeterminate condition of the SR type is defined in complement

- T*(Toggle)* F/F

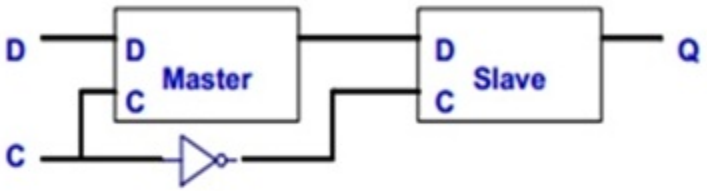


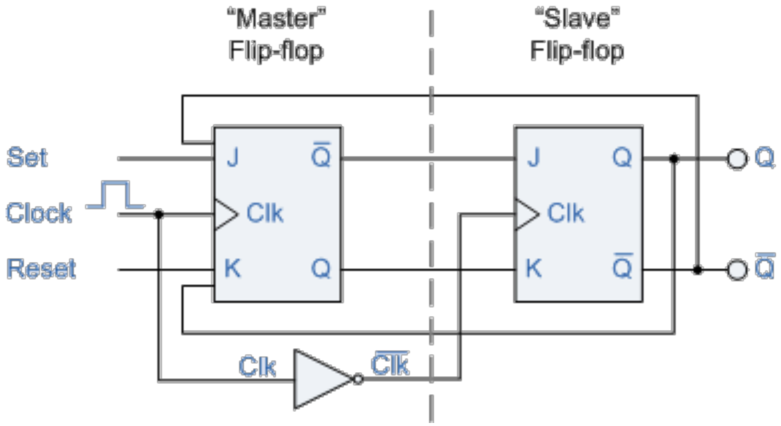
T	Q(t+1)	
0	Q(t)	no change
1	Q'(t)	Complement

  - T=1(J=K=1), T=0(J=K=0)

1-6 Flip-Flops

- Master – Slave D*(Data)* F/F

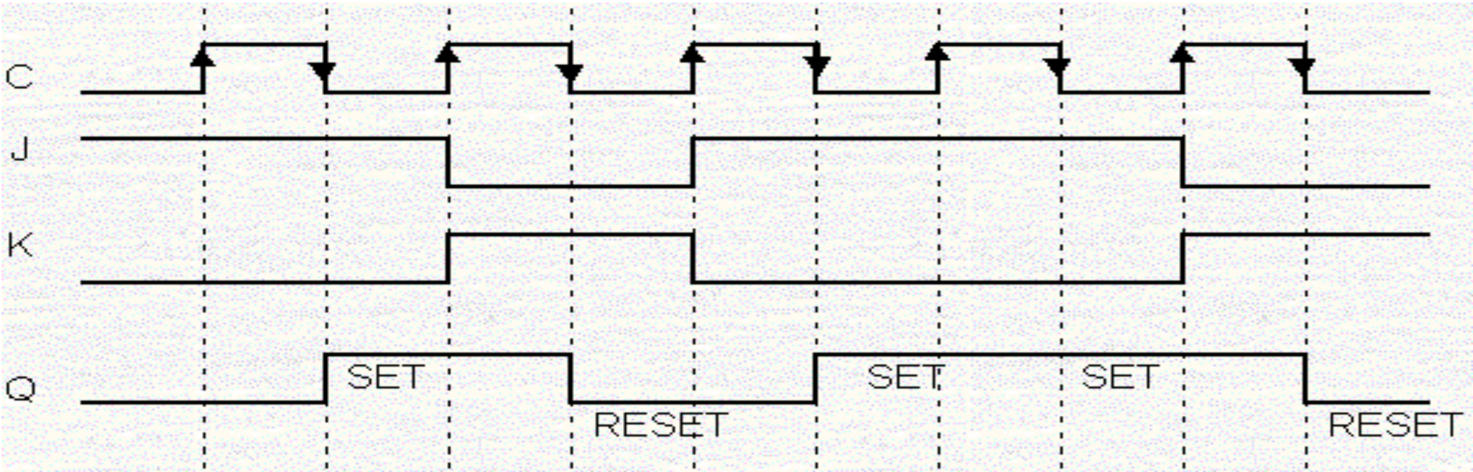




Working

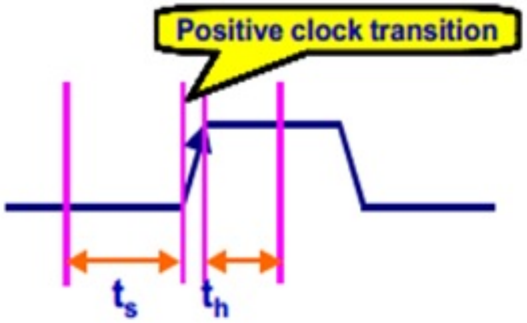
When Clock=1, the master J-K flip flop gets disabled. The Clock input of the master input will be the opposite of the slave input. So the master flip flop output will be recognized by the slave flip flop only when the Clock value becomes 0. Thus, when the clock pulse males a transition from 1 to 0, the locked outputs of the master flip flop are fed through to the inputs of the slave flip-flop making this flip flop edge or pulse-triggered. To understand better take a look at the timing diagram illustrated below.

Thus, the circuit accepts the value in the input when the clock is HIGH, and passes the data to the output on the falling-edge of the clock signal. This makes the Master-Slave J-K flip flop a Synchronous device as it only passes data with the timing of the clock signal.



1-6 Flip-Flops

- Edge-Triggered F/F
  - State Change : *Clock Pulse*
    - Rising Edge(positive-edge transition)
    - Falling Edge(negative-edge transition)
  - Setup time(20ns)
    - minimum time that D input must remain at constant value before the transition.
  - Hold time(5ns)
    - minimum time that D input must not change after the positive transition.
  - Propagation delay(max 50ns)
    - time between the clock input and the response in Q
  - Master-Slave F/F



What is a multiplexer?

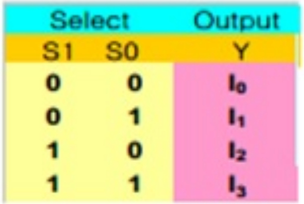
a)It is a type of decoder which decodes several inputs and gives one output
b) A multiplexer is a device which converts many signals into one
c) It takes one input and results into many output
d) It is a type of encoder which decodes several inputs and gives one output

○answer-b
○A multiplexer (or MUX) is a device that selects one of several analog or digital input signals and forwards the selected input into a single line, depending on the active select lines

2-3 Multiplexers

- Multiplexer(Mux)
  - A combinational circuit that receives binary information from one of 2^n input data lines and directs it to a single output line
  - A 2^n -to 1 multiplexer has *2^n input data lines* and *n input selection lines*(Data Selector)
  - 4-to-1 multiplexer Diagram : *Fig. 2-4*
  - 4-to-1 multiplexer Function Table : *Tab. 2-3*

Tab. 2-3 Function Table for 4-to-1 line Multiplexter



Select		Output
S1	S0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

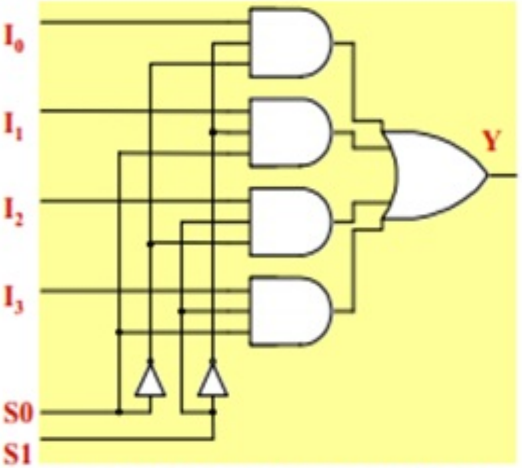
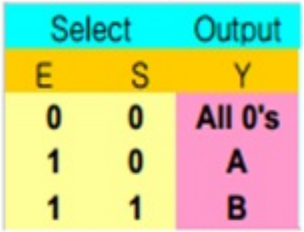


Fig. 2-4 4-to-1 Line Multiplexer

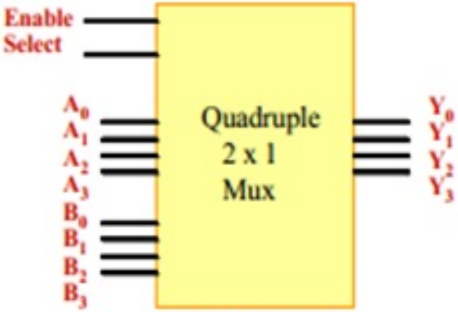
- Quadruple 2-to-1 Multiplexer
  - Quadruple 2-to-1 Multiplexer : *Fig. 2-5*

Fig. 2-5 Quadruple 2-to-1 line Multiplexter



Select		Output
E	S	Y
0	0	All 0's
1	0	A
1	1	B

(a) Function Table



(b) Block Diagram

2-3 Multiplexers

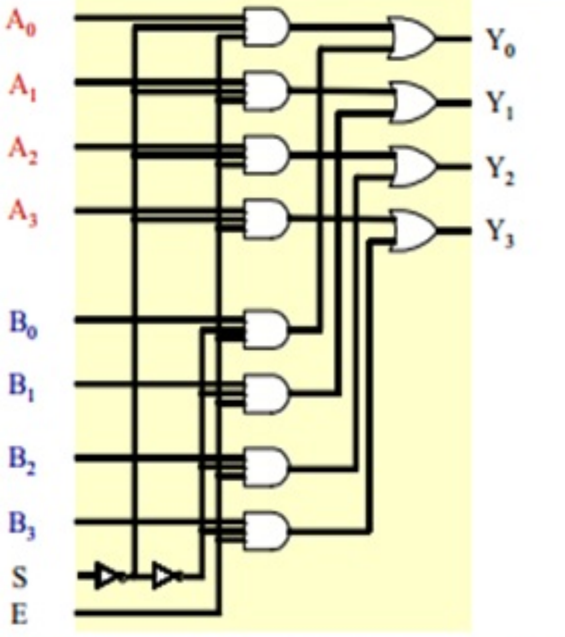
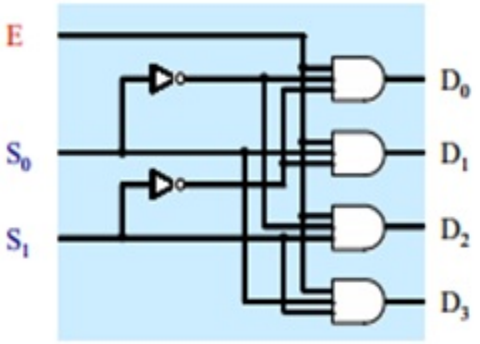


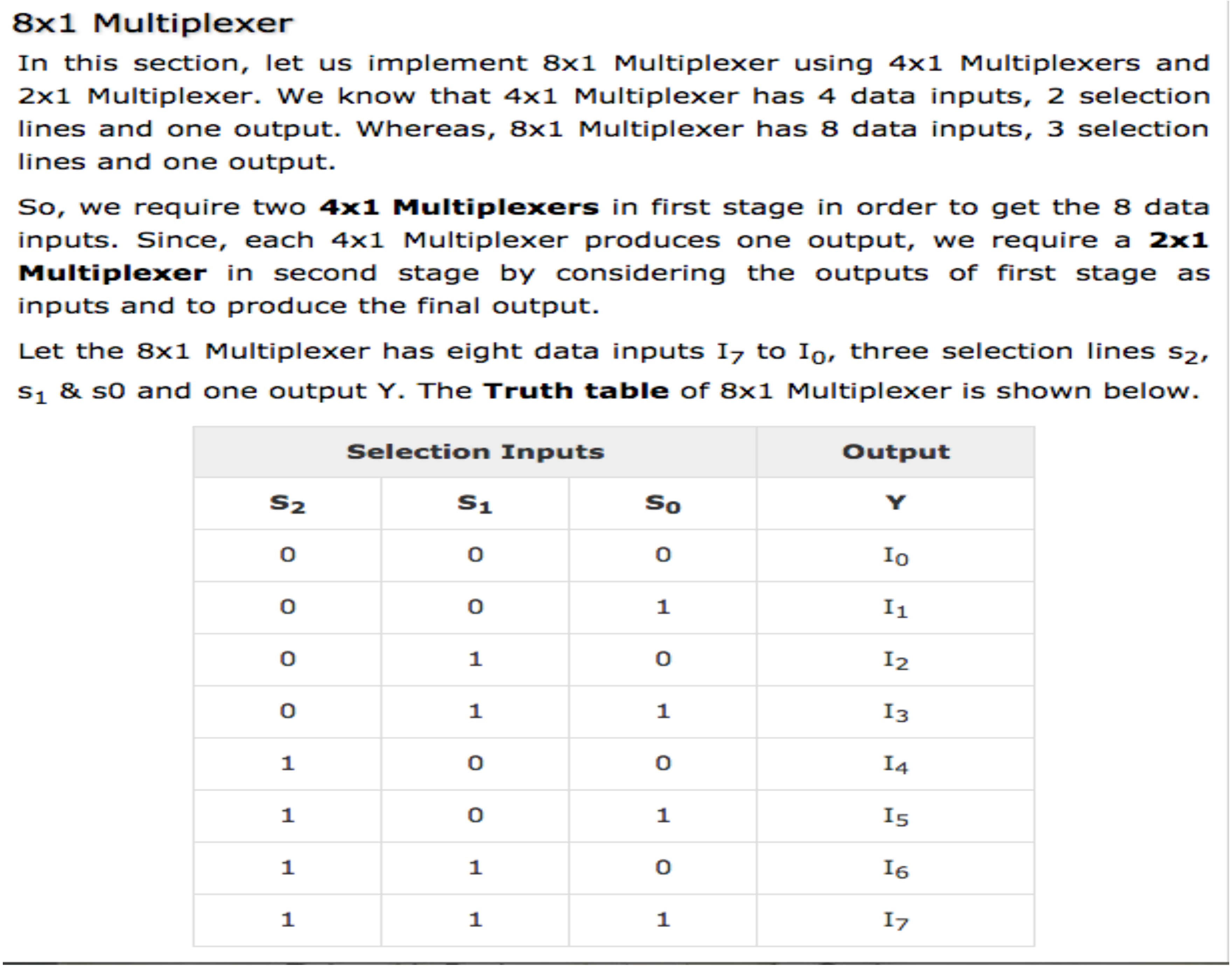
Fig A. Combinational logic diagram with four 2×1 multiplexer



E → Data input
S_0, S_1 → Select Data

Fig B. Demultiplexer

A **Demultiplexer,** sometimes abbreviated **DMUX** is a circuit that has one input and more than one output. It is used when a circuit wishes to send a signal to one of many devices



8x1 Multiplexer

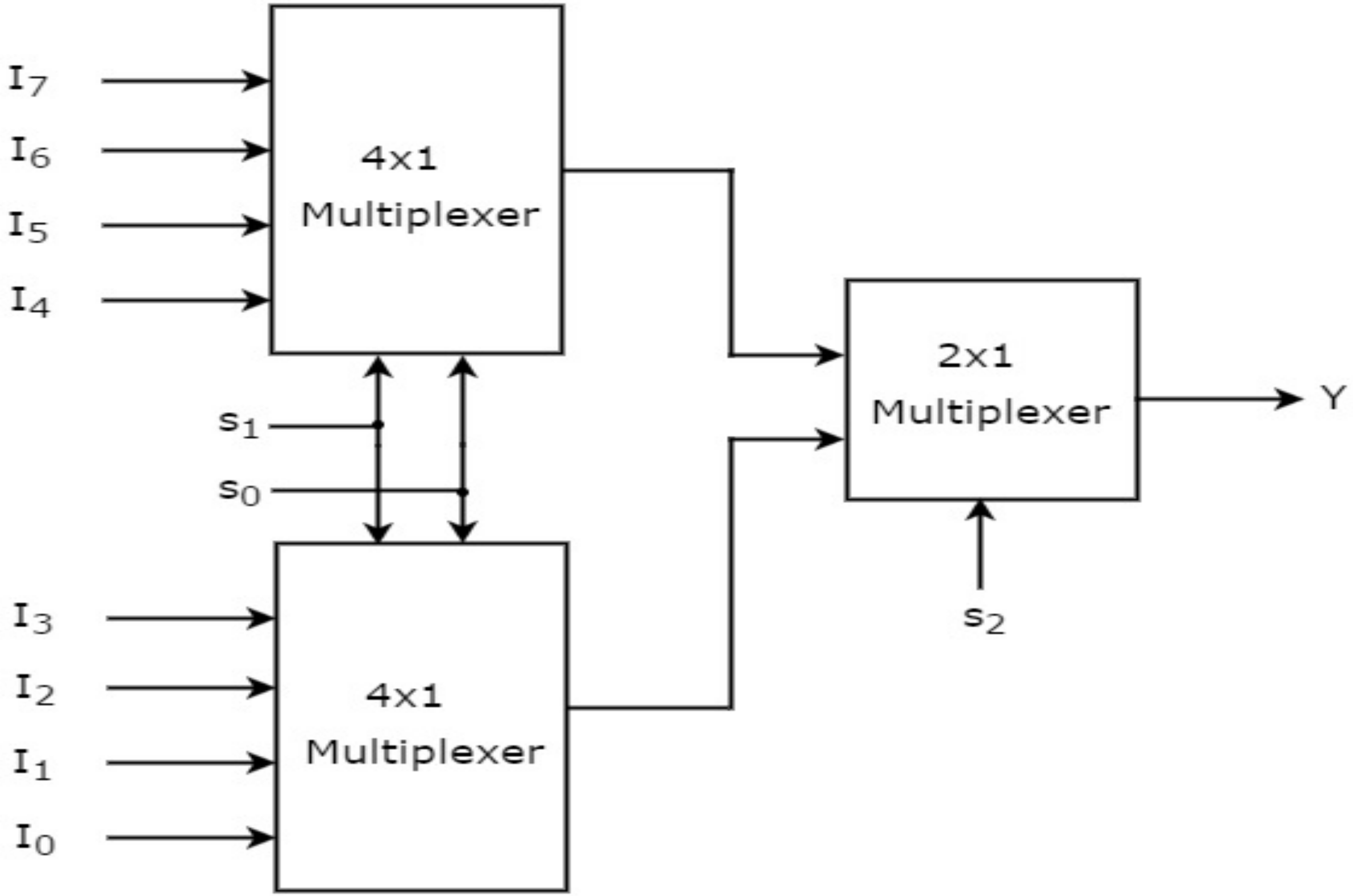
In this section, let us implement 8x1 Multiplexer using 4x1 Multiplexers and 2x1 Multiplexer. We know that 4x1 Multiplexer has 4 data inputs, 2 selection lines and one output. Whereas, 8x1 Multiplexer has 8 data inputs, 3 selection lines and one output.

So, we require two **4x1 Multiplexers** in first stage in order to get the 8 data inputs. Since, each 4x1 Multiplexer produces one output, we require a **2x1 Multiplexer** in second stage by considering the outputs of first stage as inputs and to produce the final output.

Let the 8x1 Multiplexer has eight data inputs I_7 to I_0, three selection lines s_2, s_1 & s0 and one output Y. The **Truth table** of 8x1 Multiplexer is shown below.

Selection Inputs			Output
S_2	S_1	S_0	Y
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

We can implement 8x1 Multiplexer using lower order Multiplexers easily by considering the above Truth table. The **block diagram** of 8x1 Multiplexer is shown in the following figure.


I_7
I_6
I_5
I_4
4x1
Multiplexer
s_1
s_0
I_3
I_2
I_1
I_0
4x1
Multiplexer
2x1
Multiplexer
Y
s_2