DIGITAL ELECTRONICS

UNIT 05

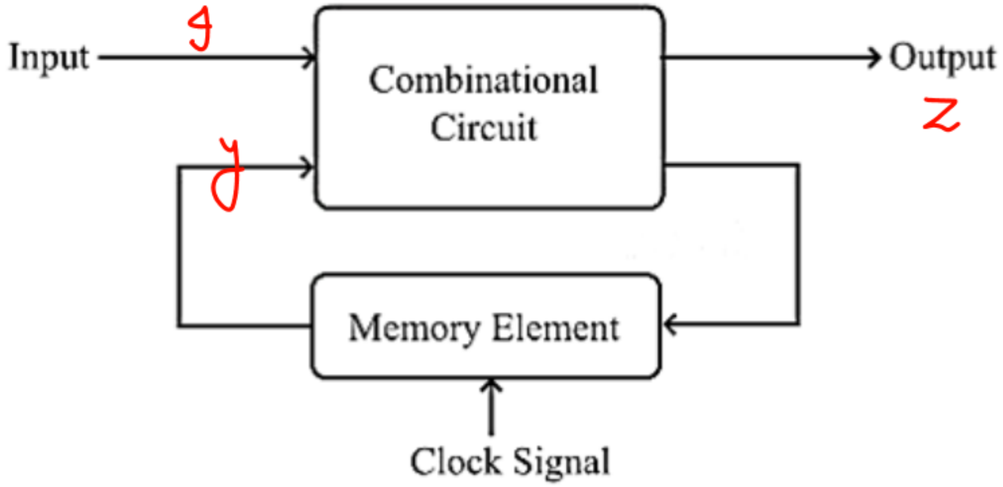
SEQUENTIAL CIRCUITS

SEQUENTIAL CIRCUITS

A sequential circuit is a circuit where the output depends on:- Current inputs AND Past inputs (history)

memory

This means it has the ability to store information, making it fundamentally different from a combinational circuit, where output is purely a function of the current inputs.



Sequence → Pattern

A.P → 2, 4, 6, 8, 10 ...

Current State = d

Past State = t

Next State = Current State + Past State

2 + 8 = 10

Sequential Circuit = Combinational Circuit + Memory Element

COMBINATIONAL VS SEQUENTIAL CIRCUITS

Aspect	Combinational Circuit	Sequential Circuit
Definition	Output depends **only on the present inputs**	Output depends on **present inputs and past inputs (stored state)**
Memory	❌ No memory — cannot store past inputs	☑ Has memory — can remember previous input history
Feedback Path	❌ No feedback loop — purely forward logic	☑ Feedback loop exists — output can affect next input
Basic Components	Logic gates (AND, OR, NOT, etc.)	Logic gates + Memory elements (latches, flip-flops)
Storage Capability	Cannot store data	Can store data using flip-flops or latches
Clock Signal	Not required	Often required (in synchronous sequential circuits)
Time Dependency	Output changes **immediately** with input	Output depends on **input + timing + past state**
Design Complexity	Relatively simple	More complex due to states and transitions
Examples	Adders, Subtractors, Encoders, Decoders, Multiplexers	Counters, Registers, Shift Registers, FSMs, Traffic Light Controllers
Representation	Represented using **Boolean expressions**	Represented using **state diagrams, tables, and Boolean expressions**

CLASSIFICATION OF SEQUENTIAL CIRCUITS

1. Synchronous Circuits → In synchronous circuit all variables changes at the same time when Clock arrives.
   - Clock → Signal
2. Asynchronous
   - ↓
   - Circuit does not need clock signal to operate
   - * As soon as Input changes the Changes in output get reflected.

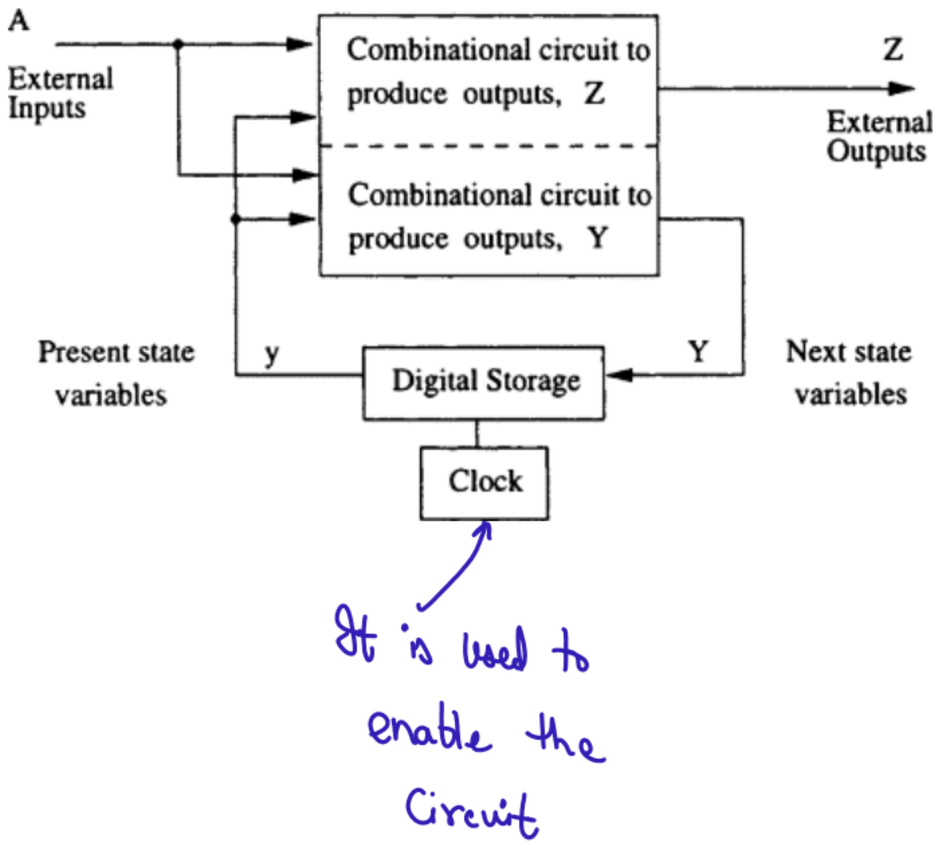
SYNCHRONOUS CIRCUITS

A Synchronous Sequential Circuit is a type of sequential circuit in which all memory elements (flip-flops) are triggered simultaneously by a common clock signal.

Clock decides, when to activate the Circuit

CLK = 1 : Enable Circuit

CLK = 0 : Disable the circuit

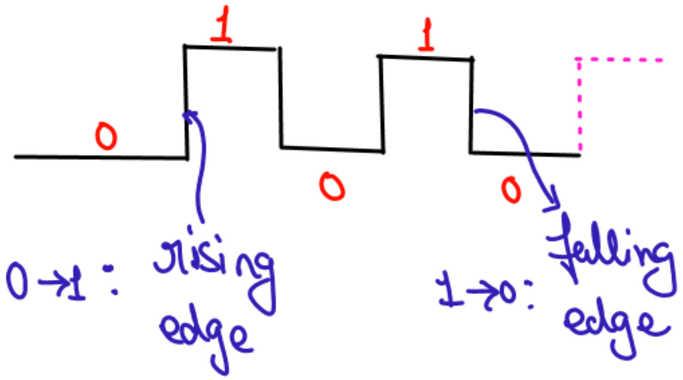


WHAT IS CLOCK?

A clock is a periodic signal (typically a square wave) used to synchronize the operations of various components in a digital system.

It provides the timing reference to tell the circuit when to perform certain actions — like updating memory, शिफ्टिंग डेटा, or changing state.

* Clock: A square wave generated by Crystal Oscilator that determines when transition take place.



Crystal Oscilator — used to design clock.

It is a material which vibrates at particular frequency when voltage applied.

"Quartz Crystal" used to make clock

"Silicon Dioxide" → Vibrates at 32.768 KHz = 32768 Hz → 2^15

⇒ We need 15 counters to get delay of Exact 1 second

ASYNCHRONOUS CIRCUITS

An asynchronous sequential circuit is a type of digital circuit where the state changes are not controlled by a global clock signal.
Instead, the output and state change immediately in response to input changes — using logic delays for timing.

- In simple words, circuit does not need a clock signal to operate. It does not wait for clock to arrive.

Feature	Description
No clock	State changes happen without a clock pulse
Depends on input change	Input changes directly affect the next state
Relies on propagation delay	Timing is controlled by gate delays (not synchronized)
More difficult to design	Prone to hazards and glitches

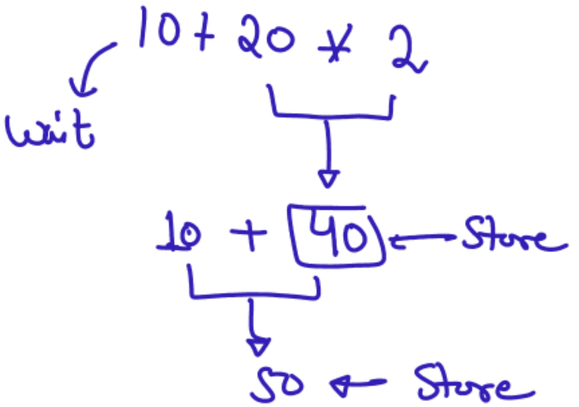
MEMORY ELEMENTS

The circuits which holds the data is called memory element.

A memory element can store one bit of information — either a 0 or a 1.

What Does a Memory Element Do?

Function	Example
Store current state	"The system is in State A"
Hold value	Keeps a 1 or 0 until changed
Respond to control inputs	Like Set/Reset or Clock
Feed back the value	Used in feedback loops to create memory



STATE AND STATE VARIABLES

STATE: The state is the stored information that represents the history of input conditions up to the current time.
in simple words, present stored value in the system is called state.

STATE VARIABLES: are binary variables used to represent the state of a sequential circuit.

Eg:

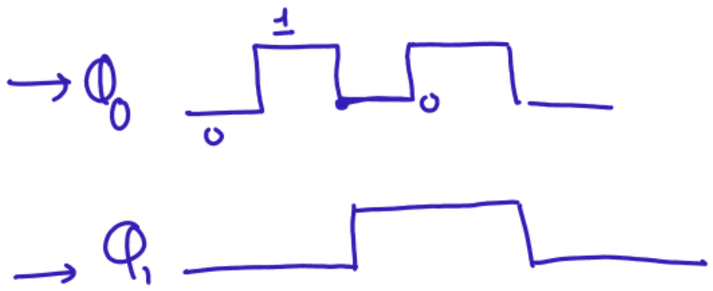
Counter (3 bit)

Q_2	Q_1	Q_0
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0

← State variable

← States (present)

State change



2 Stable state

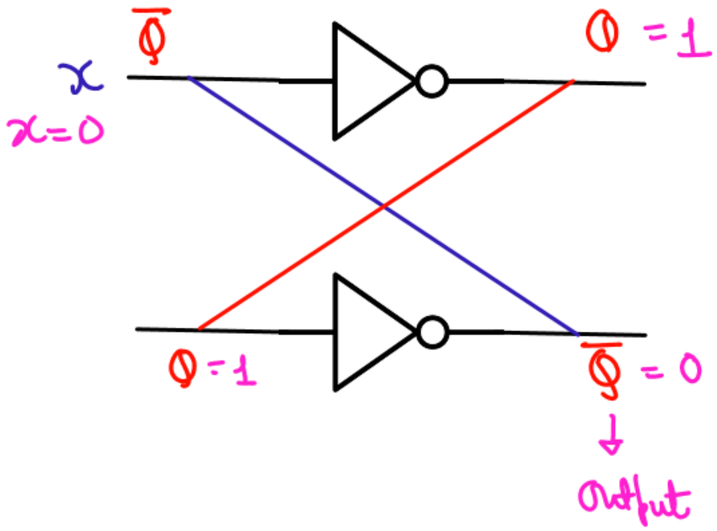
BISTABLE MULTIVIBRATOR

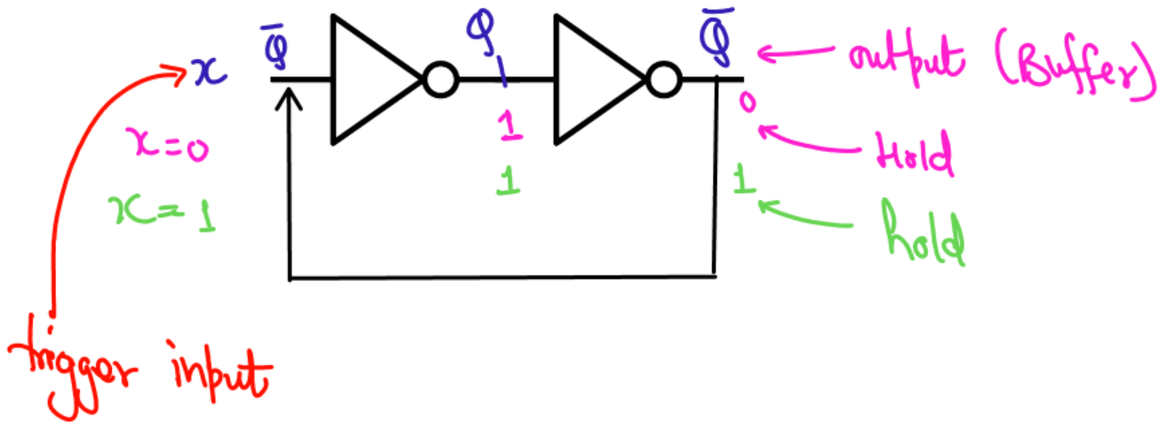
A bistable multivibrator is a digital circuit that has two stable states, and it remains in either state indefinitely until it is triggered to switch to the other.

It's called "bi-stable" because it has two stable outputs:

- State 0 (Q = 0)
- State 1 (Q = 1)

It is used as a basic memory element — capable of storing one bit (0 or 1).





* In such a system, a particular value {0, 1} gets stuck / Stored / Latched in circuit

LATCHES

A latch is a basic memory element that can store 1 bit of information. It is level-sensitive, meaning its output depends on the input as long as a control signal (Enable) is active.

level triggered

* Used to create memory elements like registers.

Types

- SR Latch → Active High
  → Active Low
- D Latch

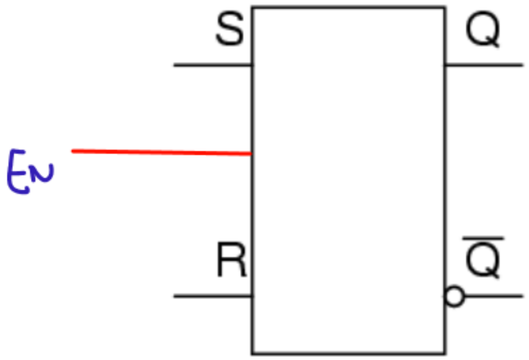
ACTIVE HIGH SR LATCH

In an Active High SR Latch, the Set (S) and Reset (R) inputs perform their operations when they are logic HIGH (1).

- Also called as NOR latch

Store

Make the value 0

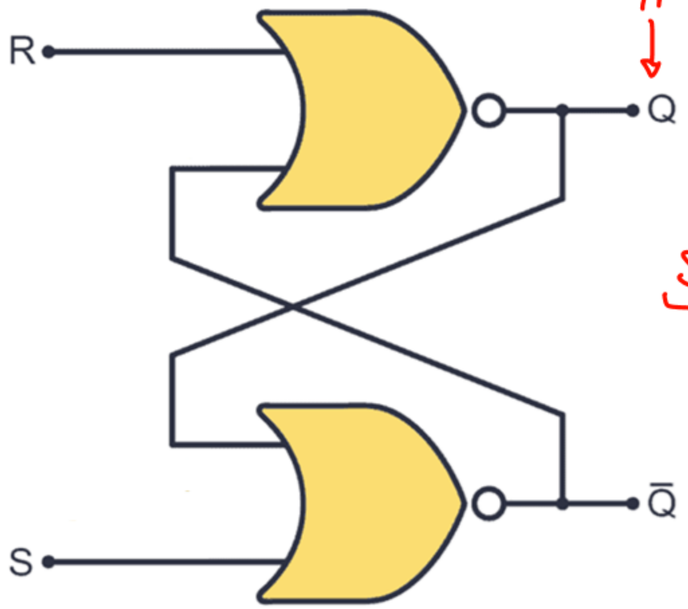


- When SET is active, then stored value in the latch becomes 1.
- When RESET is active, then stored value in the latch become 0.

En = 1, then latch work

En = 0, Latch will not work

ACTIVE HIGH SR LATCH



O/p

S=1, Q=1 (fixed)

R=1, Q=0 (fixed)

S=R=1

Invalid

Present State Q_n, $\bar{Q}_n$

Next State Q_{n+1}, $\bar{Q}_{n+1}$

S=1, R=0

⇒ $\bar{Q}=0$ Next $\bar{Q}_{n+1}=0$ ...

$Q=1$ (Current) $Q_{n+1}=1$ } latch

When Set is active Q=1, & Reset is active; Q=0

S=0, R=1

$Q_n=0$ Next $Q_{n+1}=0$

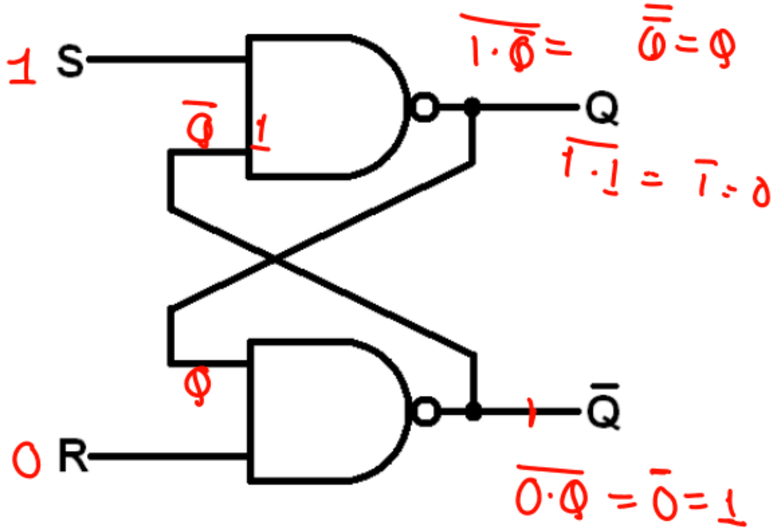
$\bar{Q}_n=1$ (Current) $\bar{Q}_{n+1}=1$ } latch

S	R	Q (Next State)	$\bar{Q}$	Description
0	0	No change	No change	Memory (hold)
1	0	1	0	Set (Q = 1)
0	1	0	1	Reset (Q = 0)
1	1	✗ Invalid	✗	Forbidden state

(Invalid State)

ACTIVE LOW SR LATCH

- Also called as NOR latch



Set = 0,
then Q = 1

Reset = 0,
then Q = 0

DYS
Understand
If S=R=0
&
S=R=1

If Set is low that means write the circuit by 1, & hold

If Reset = 0 that means write the circuit by 0 & hold

S̄	R̄	Q (Next State)	Q̄	Description
1	1	No change	No change	Memory (hold)
0	1	1	0	Set (Q = 1)
1	0	0	1	Reset (Q = 0)
0	0	✗ Invalid	✗	Forbidden state

← Invalid