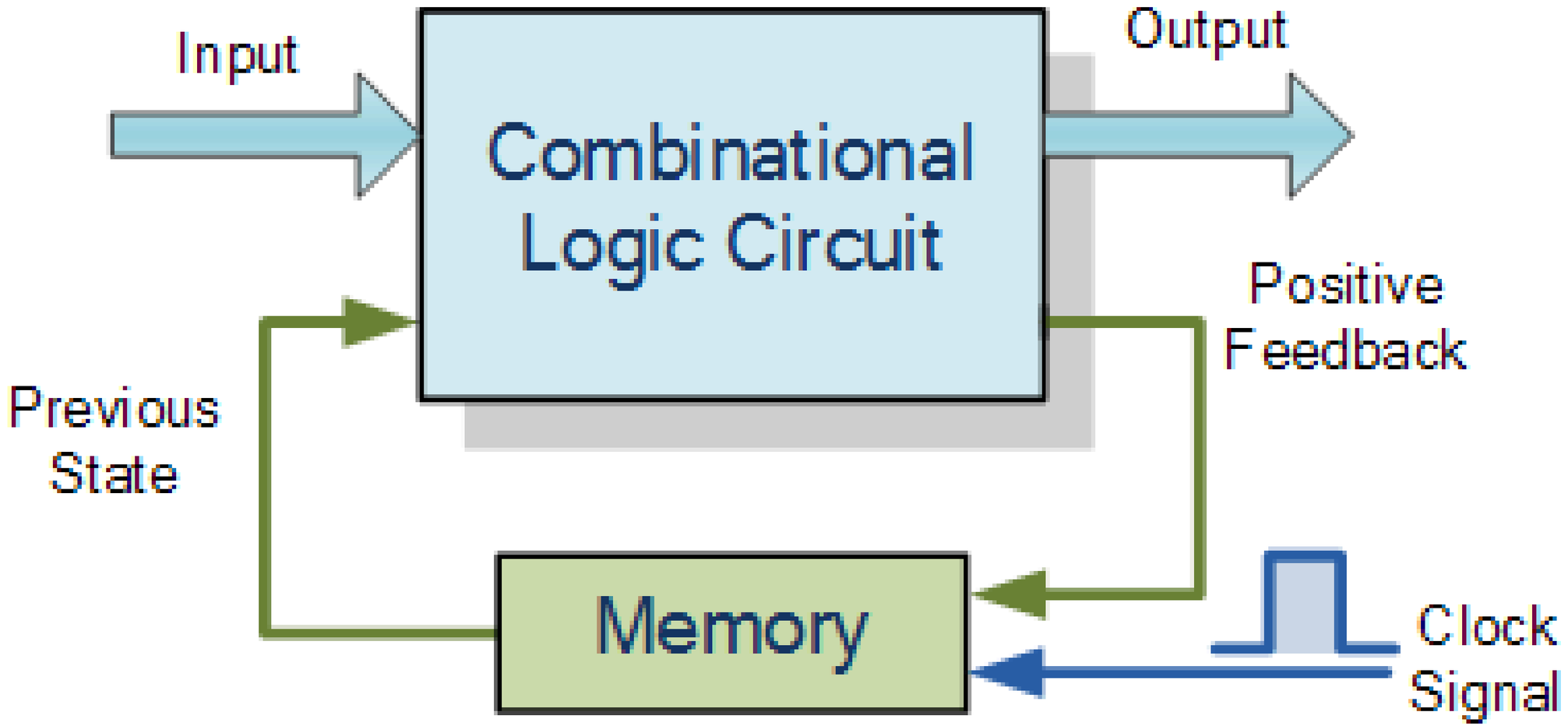
Sequential Circuit



- **FLIP FLOPS , LATCHES , COUNTERS**

- **U**nlike Combinational Logic circuits that change state depending upon the actual signals being applied to their inputs at that time, Sequential Logic circuits have some form of inherent “Memory” built in.

- This means that sequential logic circuits are able to take into account their previous input state as well as those actually present, a sort of “before” and “after” effect is involved with sequential circuits.

- the output state of a “sequential logic circuit” is a function of the following three states,

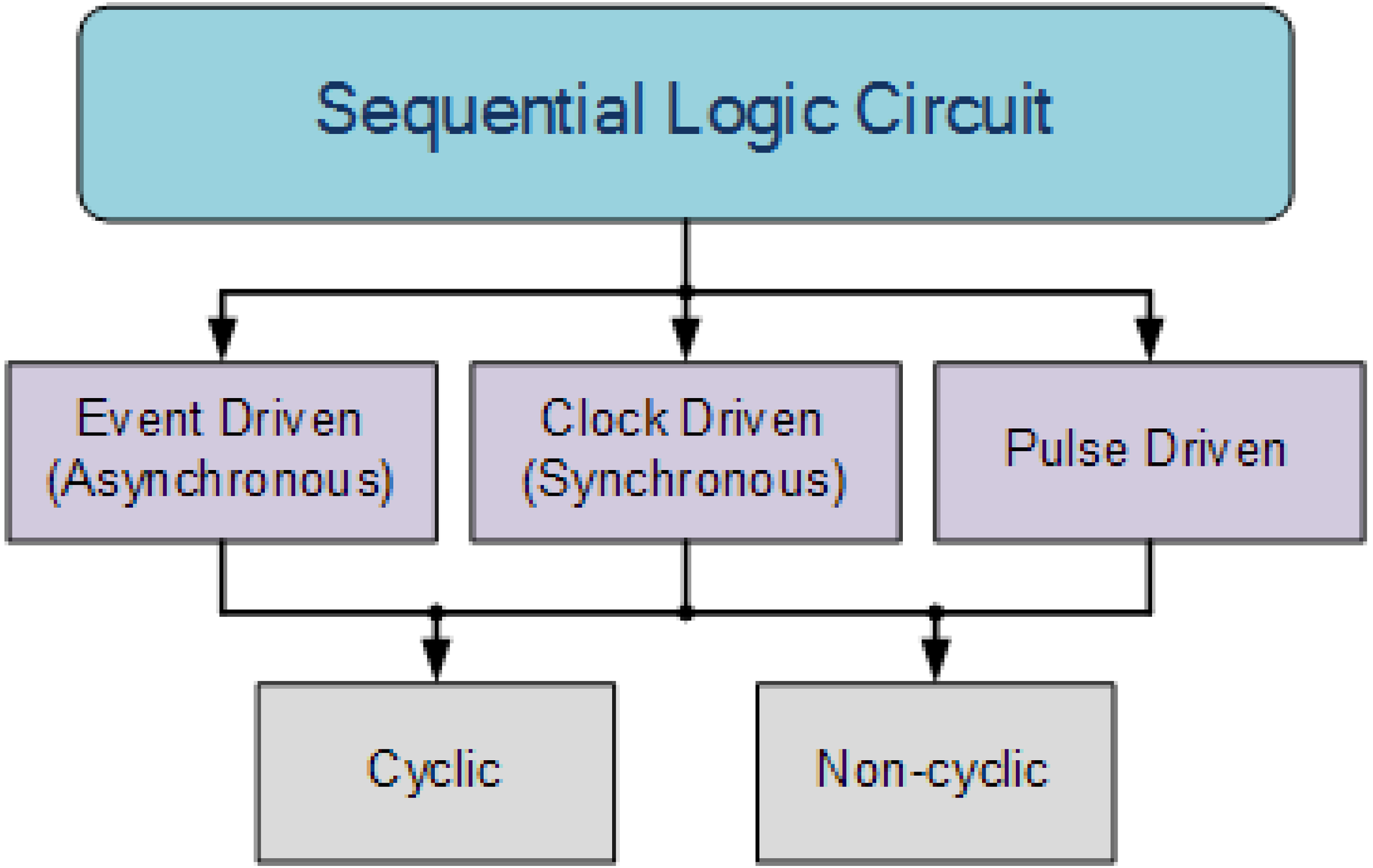
the “present input”,
the “past input” and/or
the “past output”.

- Sequential Logic circuits remember these conditions and stay fixed in their current state until the next clock signal changes one of the states, giving sequential logic circuits “Memory”.

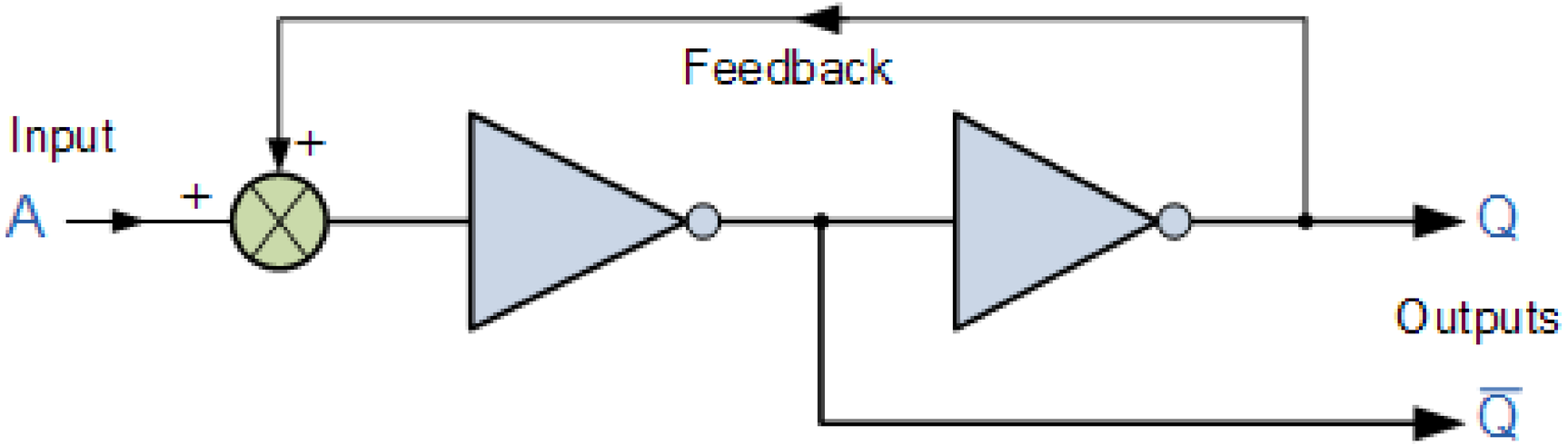
- Sequential logic circuits are generally termed as two state or Bistable devices which can have their output or outputs set in one of two basic states, a logic level “1” or a logic level “0” and

- will **remain “latched” (hence the name latch**) indefinitely in this current state or condition until some other input trigger pulse or signal is applied which will cause the bistable to change its state once again.

- **word “Sequential” means that things happen in a “sequence”, one after another and in Sequential Logic circuits, the actual clock signal determines when things will happen next.**
- **sequential logic circuits can be constructed from standard Bistable circuits such as: Flip-flops, Latches and Counters and which themselves can be made by simply connecting together universal NAND Gates and/or NOR Gates in a particular combinational way to produce the required sequential circuit.**


Sequential Logic Circuit
Event Driven (Asynchronous)
Clock Driven (Synchronous)
Pulse Driven
Cyclic
Non-cyclic

Sequential Feedback Loop



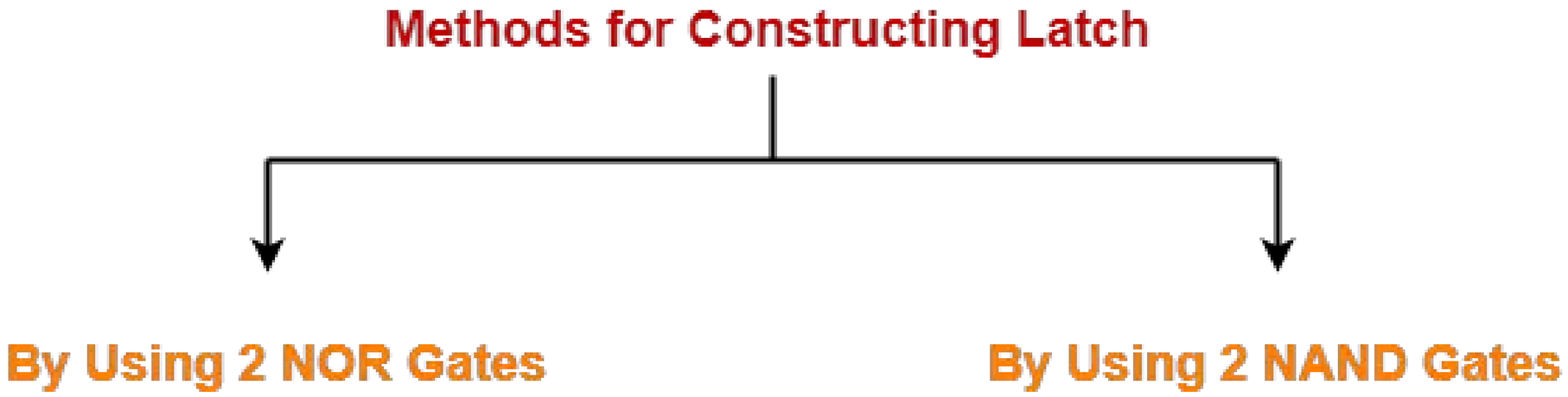
- The two inverters or NOT gates are connected in series with the output at Q fed back to the input. Unfortunately, this configuration never changes state because the output will always be the same, either a “1” or a “0”, it is permanently set.

Latch-

- A latch is basically an asynchronous level driven flip flop.
- A latch is the basic building block using which clocked flip flops are constructed.
- Latches are digital circuits that store a single bit of information and hold its value until it is updated by new input signals.
- They are used in digital systems as temporary storage elements to store binary information.
- Latches can be implemented using various digital logic gates, such as AND, OR, NOT, NAND, and NOR gates.
- Latches are useful for the design of the asynchronous sequential circuit

SR (Set-Reset) Latch –

- They are also known as preset and clear states.
- The SR latch forms the basic building blocks of all other types of flip-flops.



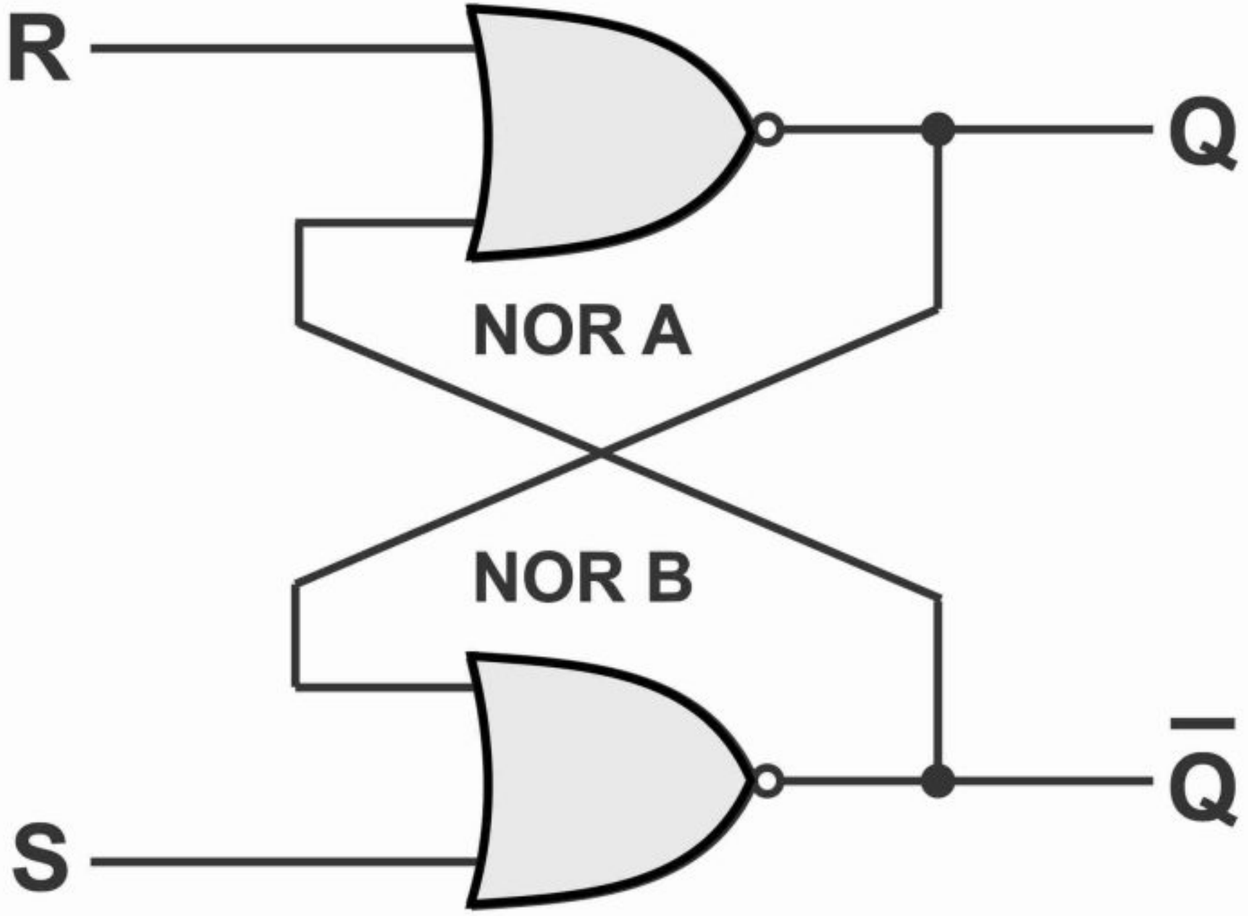
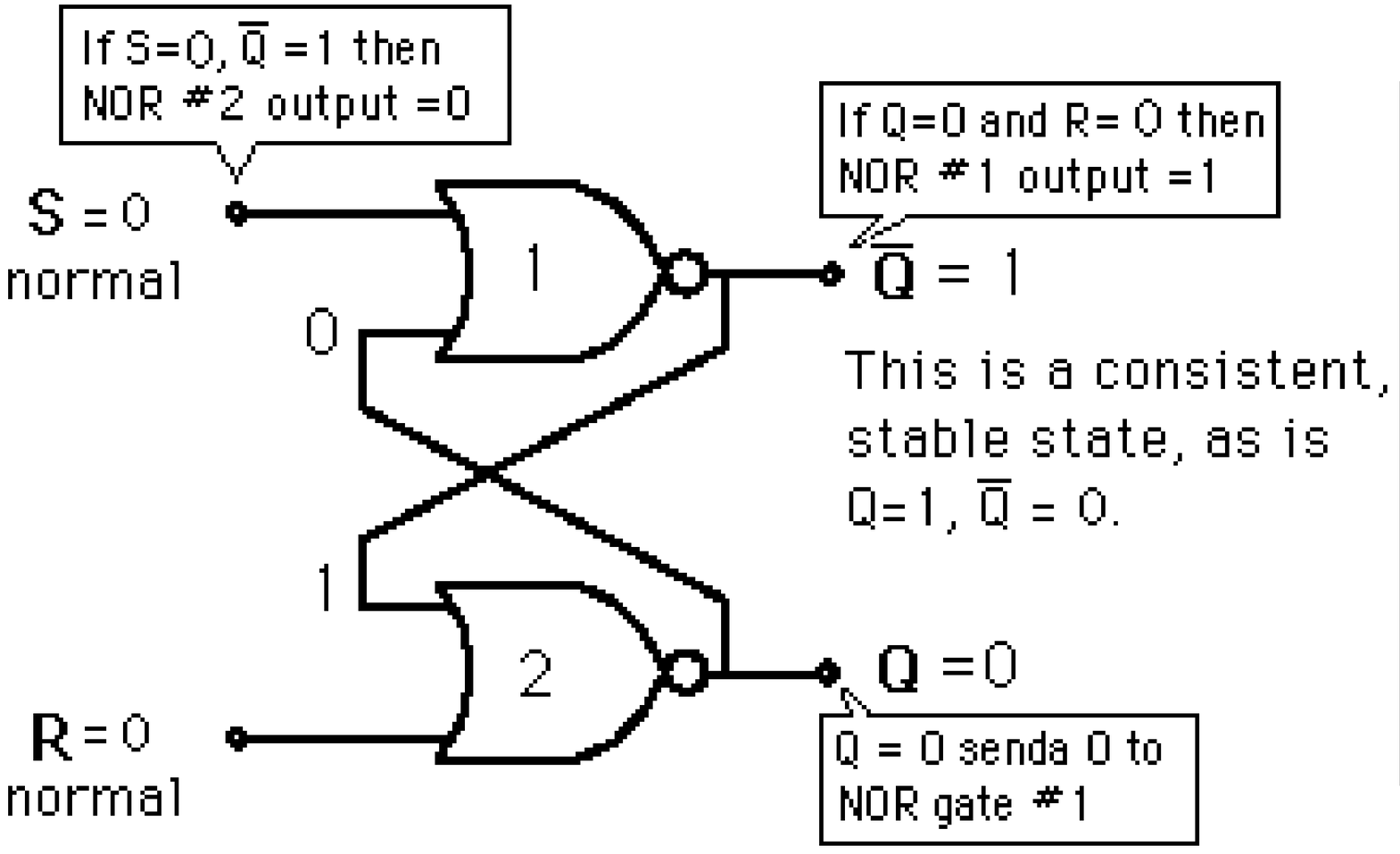


Fig 5.5 RS latch flip-flop using NOR gate

S	*R*	*Q*	*Q'*	
0	0	NC	NC	No change. Latch remained in present state.
1	0	1	0	Latch SET.
0	1	0	1	Latch RESET.
1	1	0	0	Invalid condition.



Truth Table

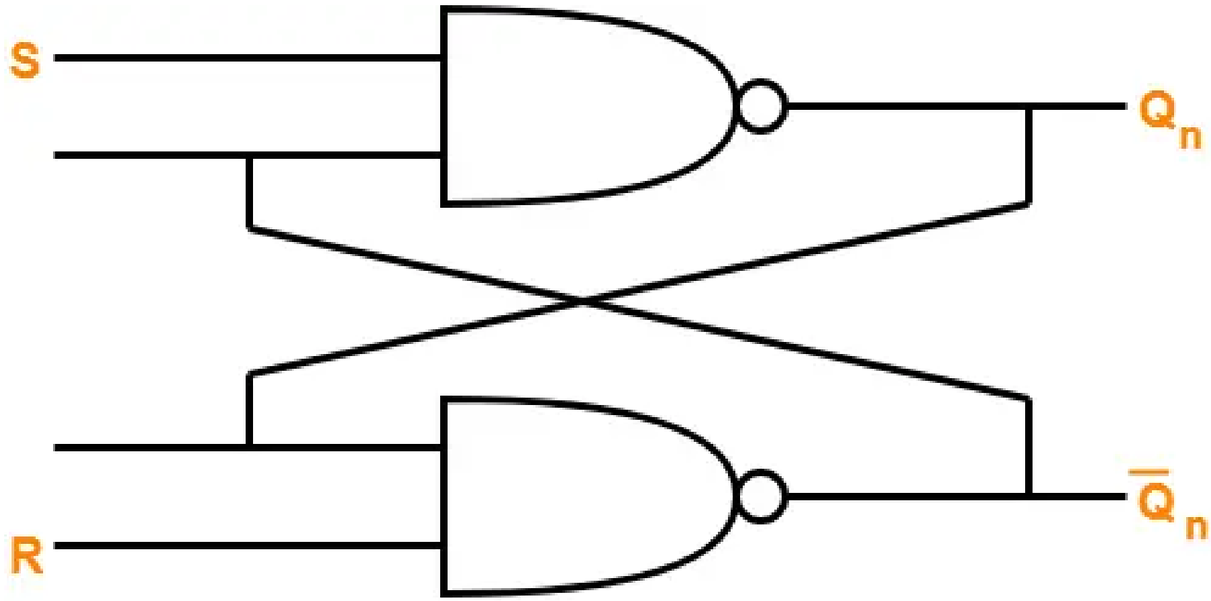
Set	Reset	Output
0	0	No change*
1	0	Q=1
0	1	Q=0
1	1	Invalid state

*can be used for data storage

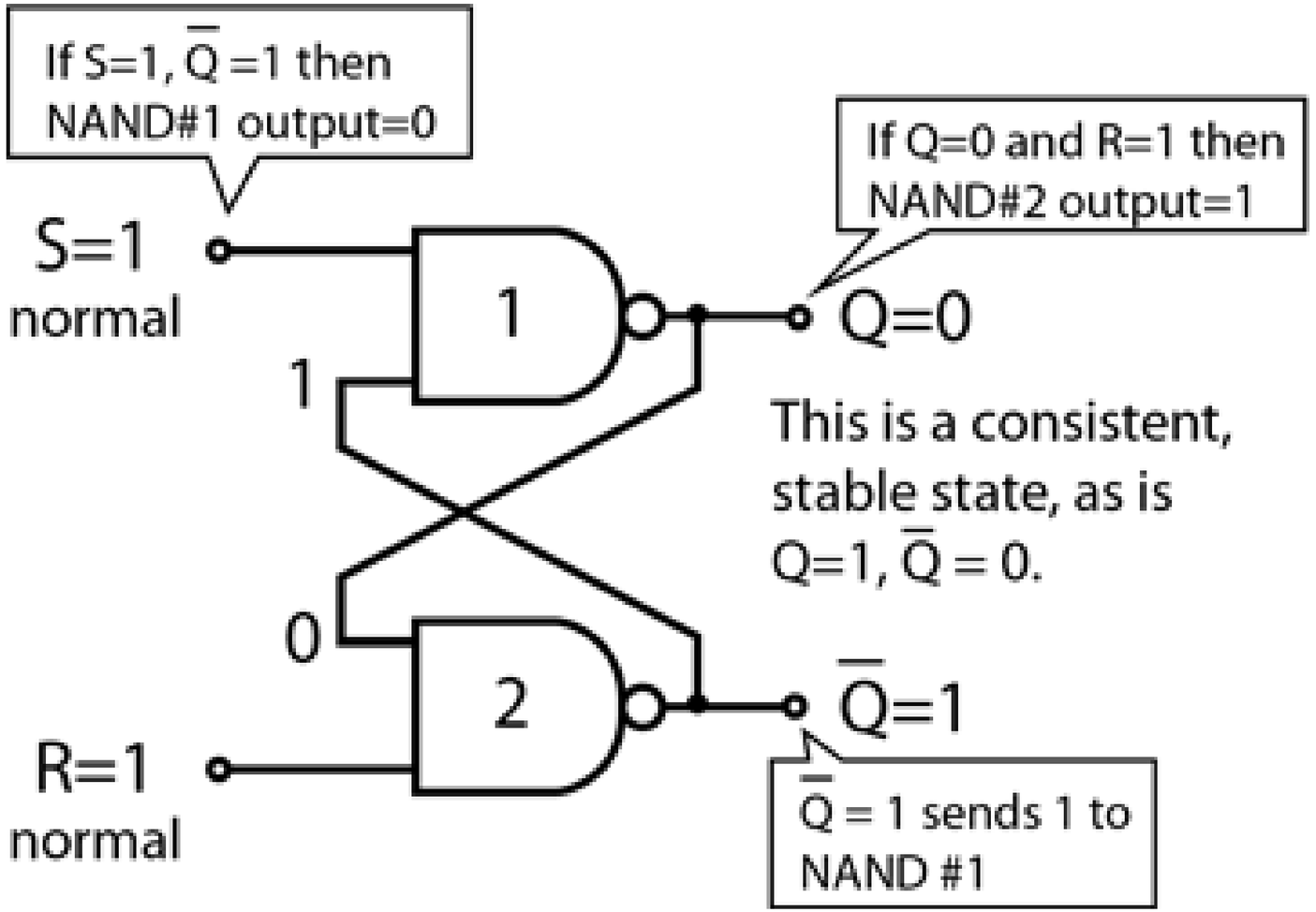
Logic Symbol-

Logic Symbol

Construction Of Latch By Using 2 NAND Gates-



Latch Construction Using NAND Gates



Truth Table

Set	Reset	Output
1	1	No change*
0	1	Q=1
1	0	Q=0
0	0	Invalid state

* can be used for data storage

Advantages of Latches:

- Easy to Implement: Latches are simple digital circuits that can be easily implemented using basic digital logic gates.
- Low Power Consumption: Latches consume less power compared to other sequential circuits such as flip-flops.
- High Speed: Latches can operate at high speeds, making them suitable for use in high-speed digital systems.
- Low Cost: Latches are inexpensive to manufacture and can be used in low-cost digital systems.
- Versatility: Latches can be used for various applications, such as data storage, control circuits, and flip-flop circuits.

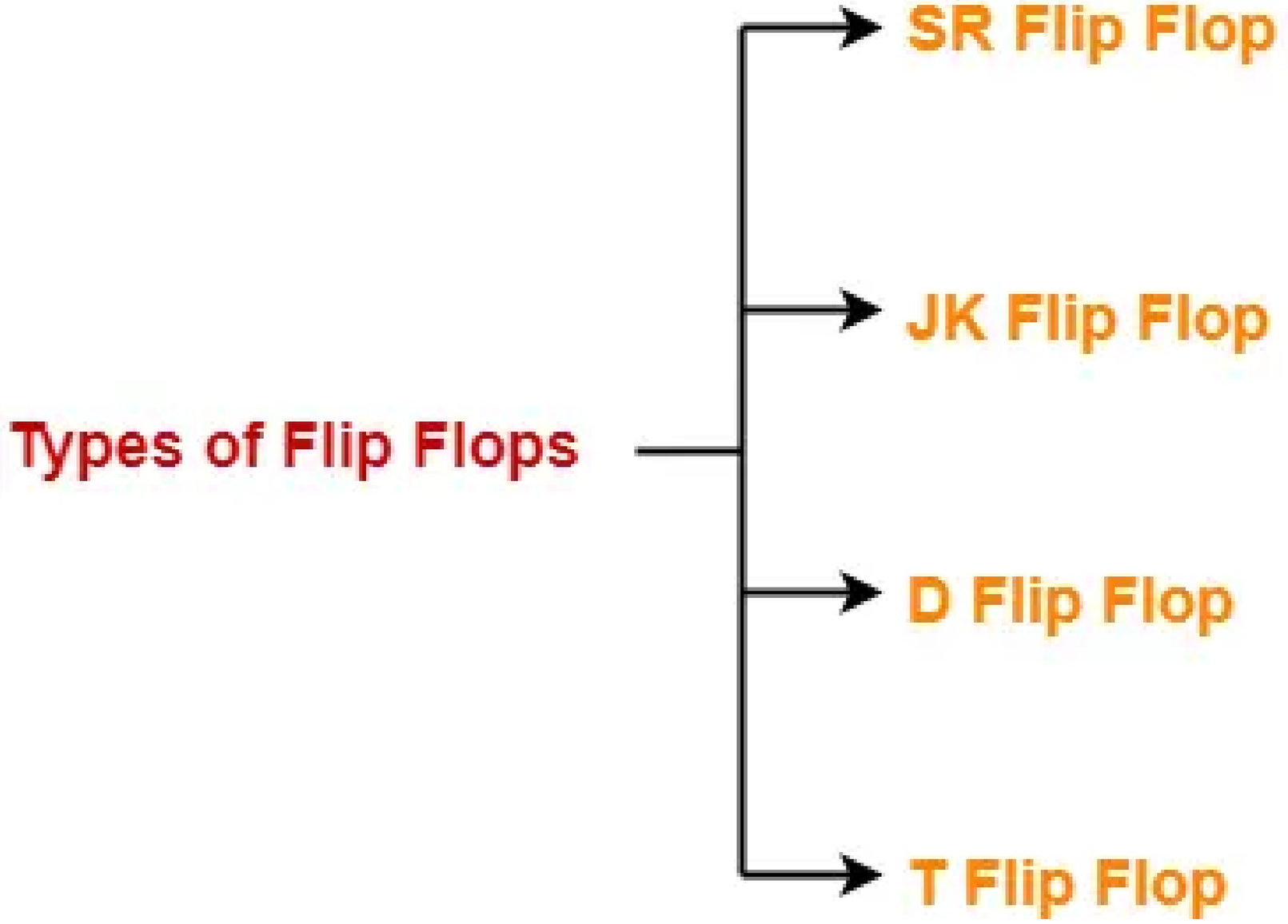
Disadvantages of Latches:

- No Clock: Latches do not have a clock signal to synchronize their operations, making their behavior unpredictable.
- Unstable State: Latches can sometimes enter into an unstable state when both inputs are at 1. This can result in unexpected behavior in the digital system.
- Complex Timing: The timing of latches can be complex and difficult to specify, making them less suitable for real-time control applications.

Flip Flop-

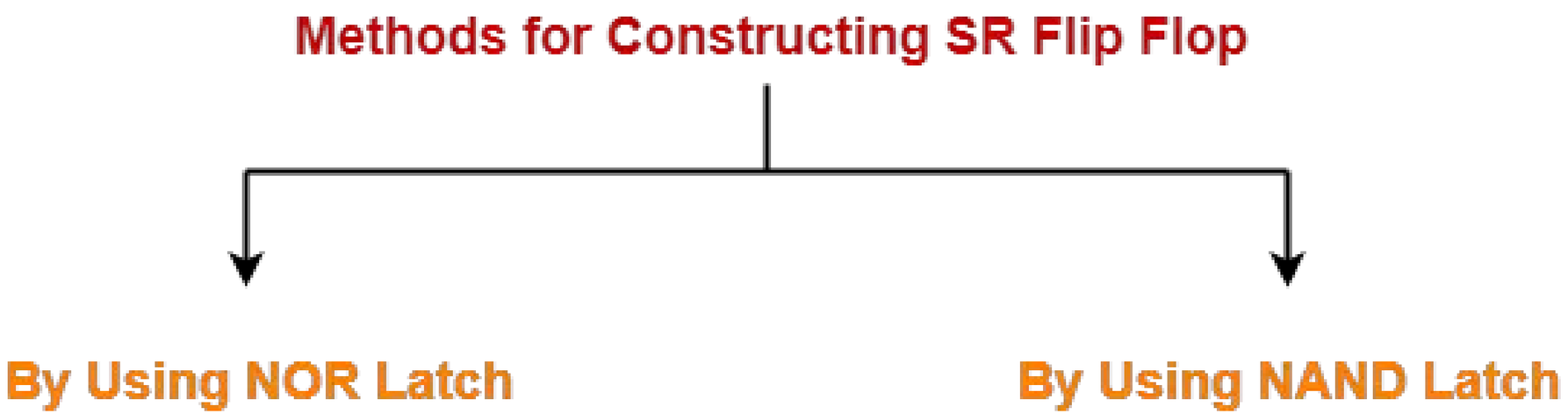
- A Flip Flop is a memory element that is capable of storing one bit of information.
  - flip flop is also called as Bistable Multivibrator because it has two stable states either 0 or 1.

Flip Flop


Types of Flip Flops
SR Flip Flop
JK Flip Flop
D Flip Flop
T Flip Flop

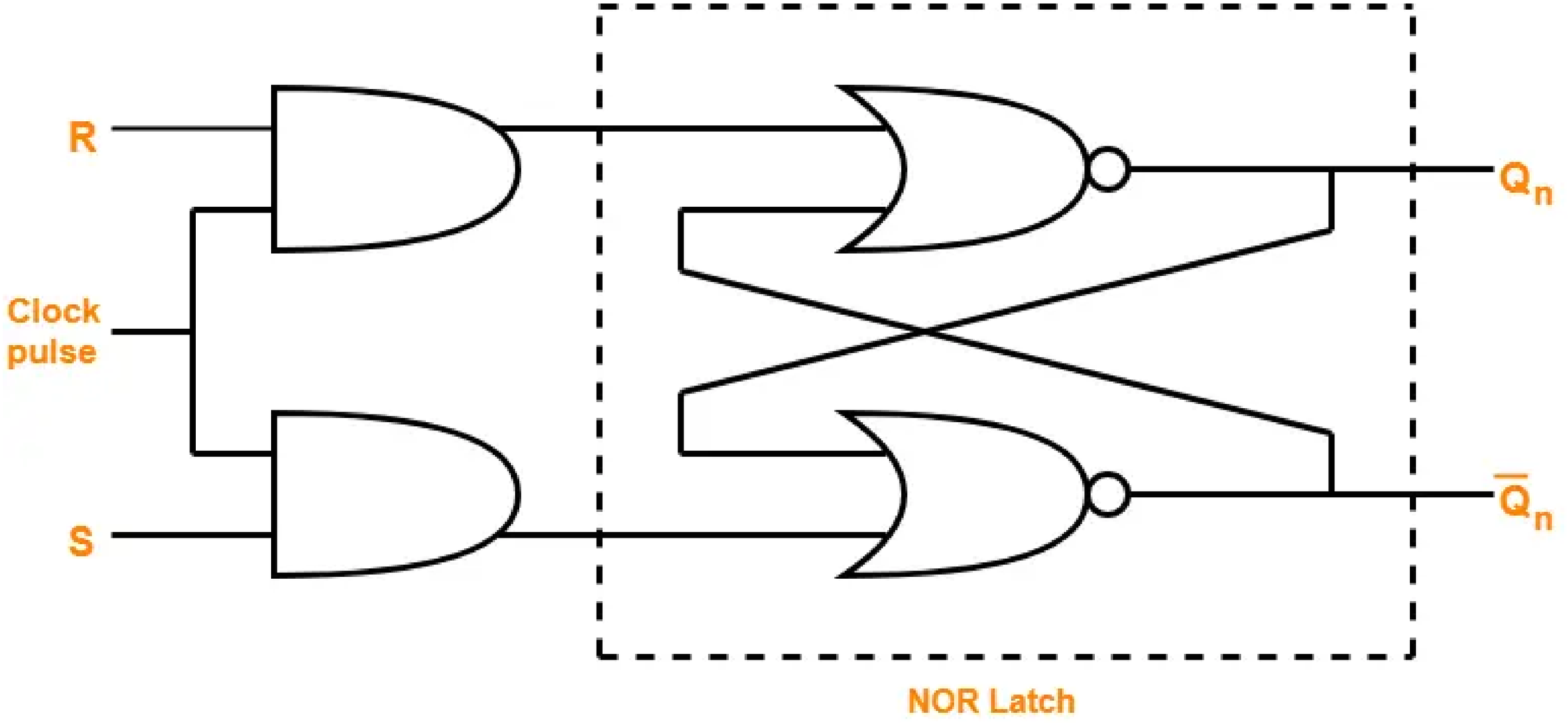
SR Flip Flop-

- SR flip flop is the simplest type of flip flops.
- It stands for Set Reset flip flop.
- It is a clocked flip flop.

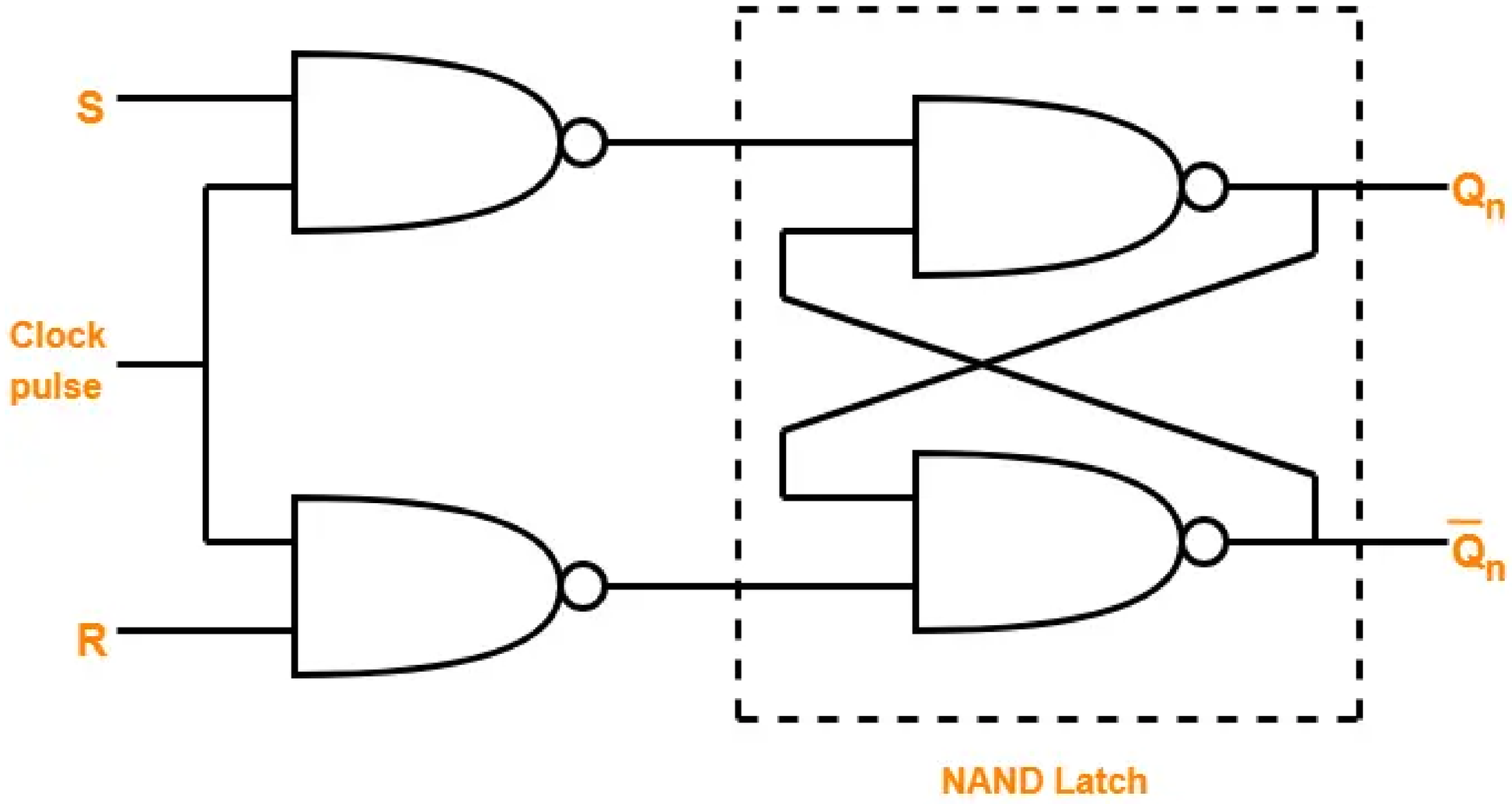


- A basic NAND gate SR flip-flop circuit provides feedback from both of its outputs back to its opposing inputs and is commonly used in memory circuits to store a single data bit.
- Then the SR flip-flop actually has three inputs, Set, Reset and its current output Q relating to it’s current state or history.

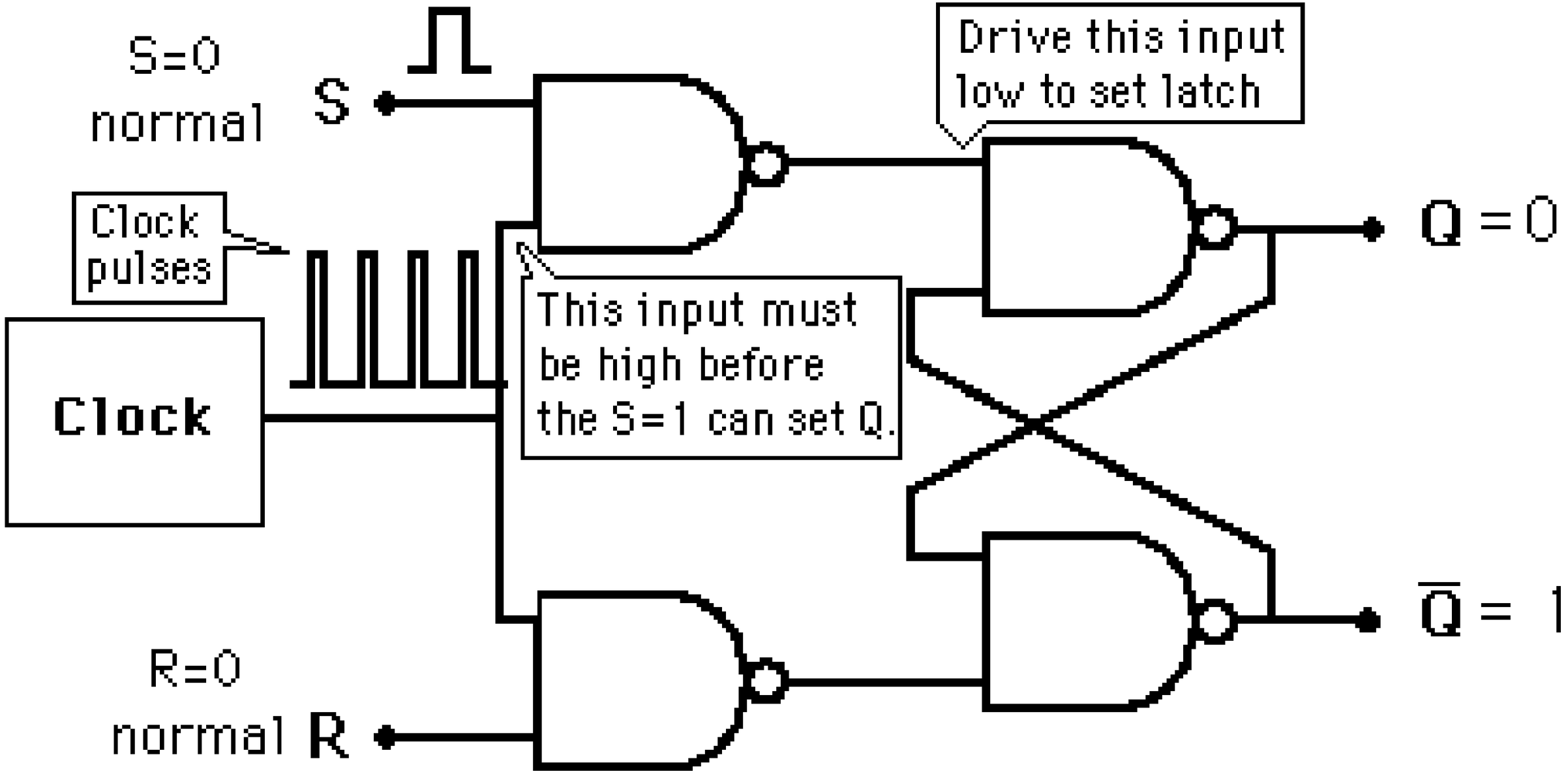
- The term “Flip-flop” relates to the actual operation of the device, as it can be “flipped” into one logic Set state or “flopped” back into the opposing logic Reset state.



SR Flip Flop Using NOR Latch



SR Flip Flop Using NAND Latch


S=0
normal
S
Drive this input
low to set latch
Clock
pulses
Clock
This input must
be high before
the S=1 can set Q.
Q = 0
$\overline{Q}$ = 1
R=0
normal R

Truth Table

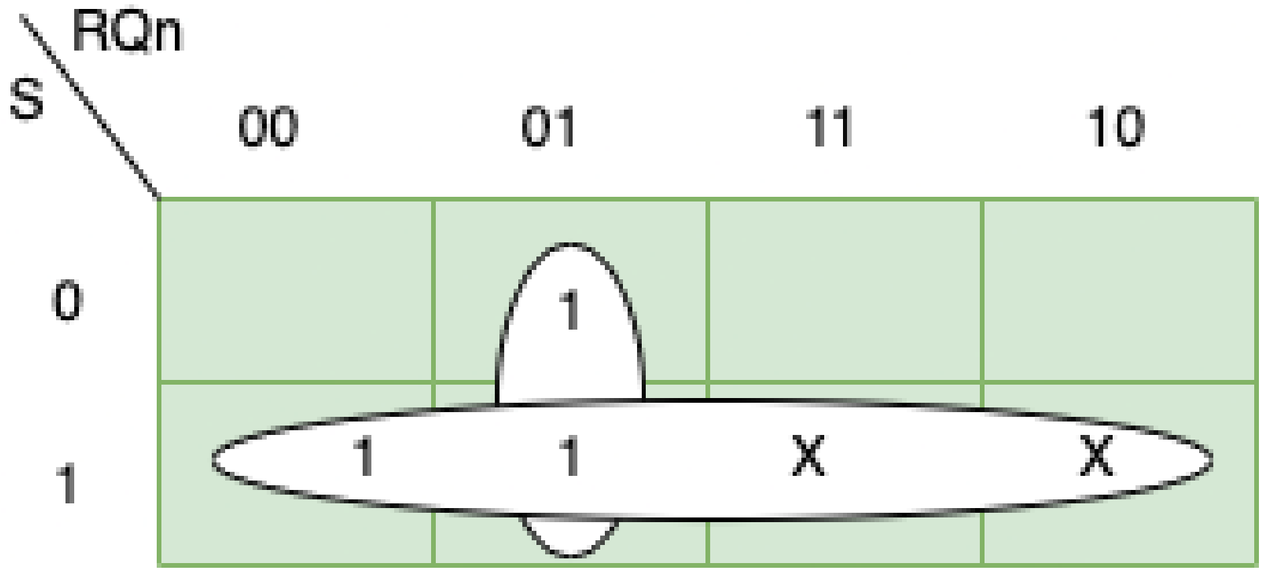
S	R	Qn+1	State
0	0	Qn	Hold
0	1	0	Reset
1	0	1	Set
1	1	X	Invalid

Characteristic Table

S	R	Qn	Qn+1
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	Invalid
1	1	1	Invalid

Characteristic Equation

- tells us about what will be the next state of flip flop in terms of present state.
- In order to get the characteristic equation, K-Map is constructed which will be shown as below:



Qn+1 = S + QnR’

Excitation Table

- Excitation Table basically tells about the excitation which is required by flip flop to go from current state to next state.

Qn	Qn+1	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

Applications of SR Flip Flop

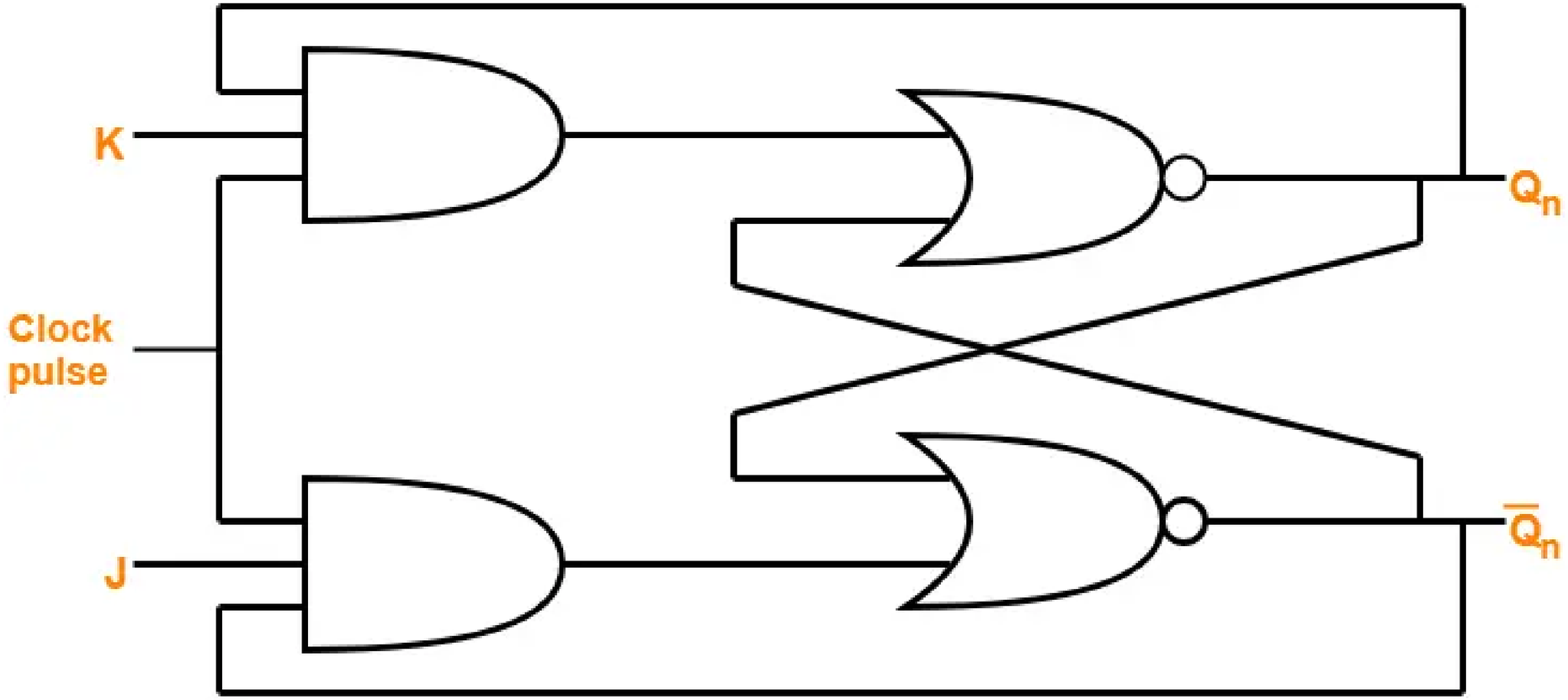
- Register: SR Flip Flop used to create register. Designer can create any size of register by combining SR Flip Flops.
- Counters: SR Flip Flops used in counters. Counters counts the number of events that occurs in a digital system.
- Memory: SR Flip Flops used to create memory which are used to store data, when the power is turned off.
- Synchronous System: SR Flip Flop are used in synchronous system which are used to synchronise the operation of different component.

JK Flip-Flop?

- JK flip flop is a refined & improved version of SR Flip Flop that has been introduced to solve the problem of indeterminate state that occurs in SR flip flop when both the inputs are 1.

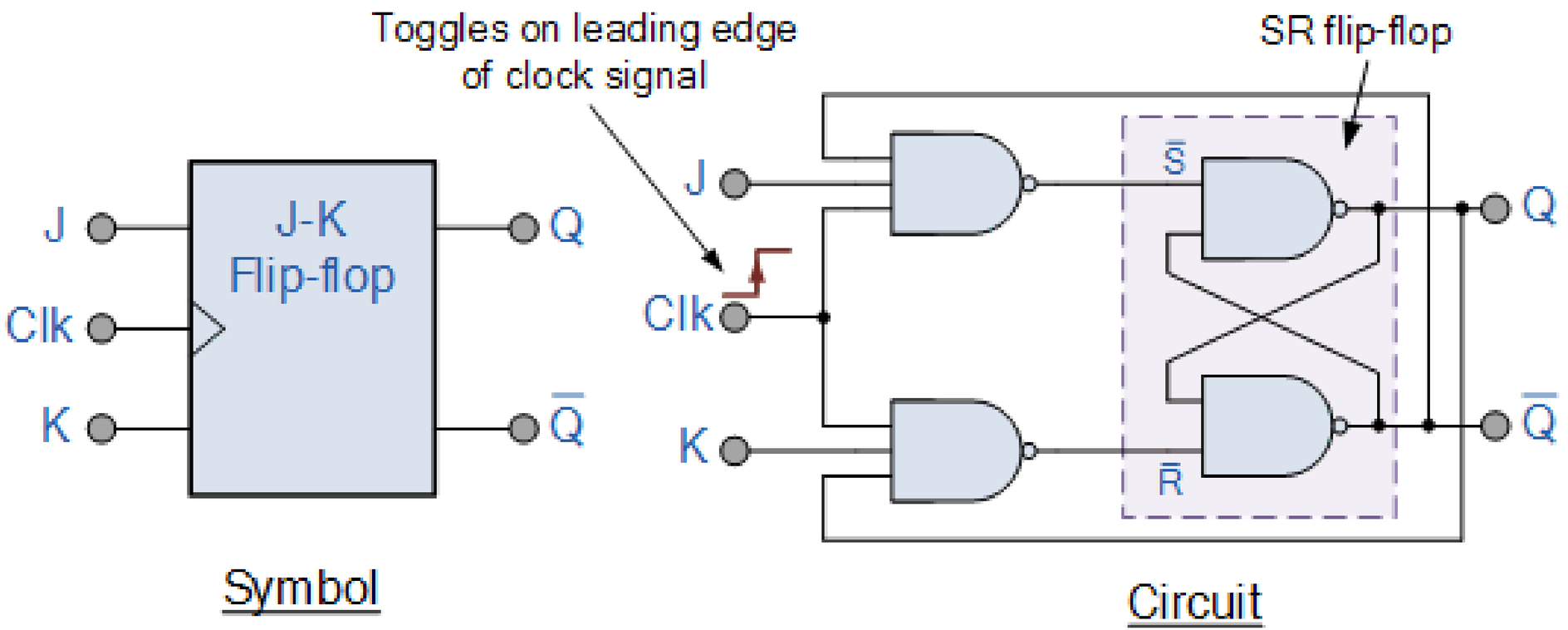
- Input J behaves like input S of SR flip flop which was meant to set the flip flop.

- Input K behaves like input R of SR flip flop which was meant to reset the flip flop.



Logic Circuit For JK Flip Flop Using SR Flip Flop

(Constructed From NOR Latch)


Toggles on leading edge
of clock signal
SR flip-flop
J
Clk
K
J-K
Flip-flop
Q
$\overline{Q}$
$\overline{S}$
$\overline{R}$
Symbol
Circuit

Truth Table

J	K	CLK	Q
0	0	↓	Q_0 (no change)
1	0	↓	1
0	1	↓	0
1	1	↓	$\overline{Q}_0$ (toggles)

CLK	J	K	Q_n	Q_{n+1}	Q_{n+1}
0	X	X	0/1	0/1	Q_n
↑	0 0	0 0	0 1	0 1	Q_n
↑	0 0	1 1	0 1	0 0	0
↑	1 1	0 0	0 1	1 1	1
↑	1 1	1 1	0 1	1 0	Q_n'

Characteristic table

Characteristic Equation-

K Map

$$Qn+1 = Q'nJ + QnK'$$

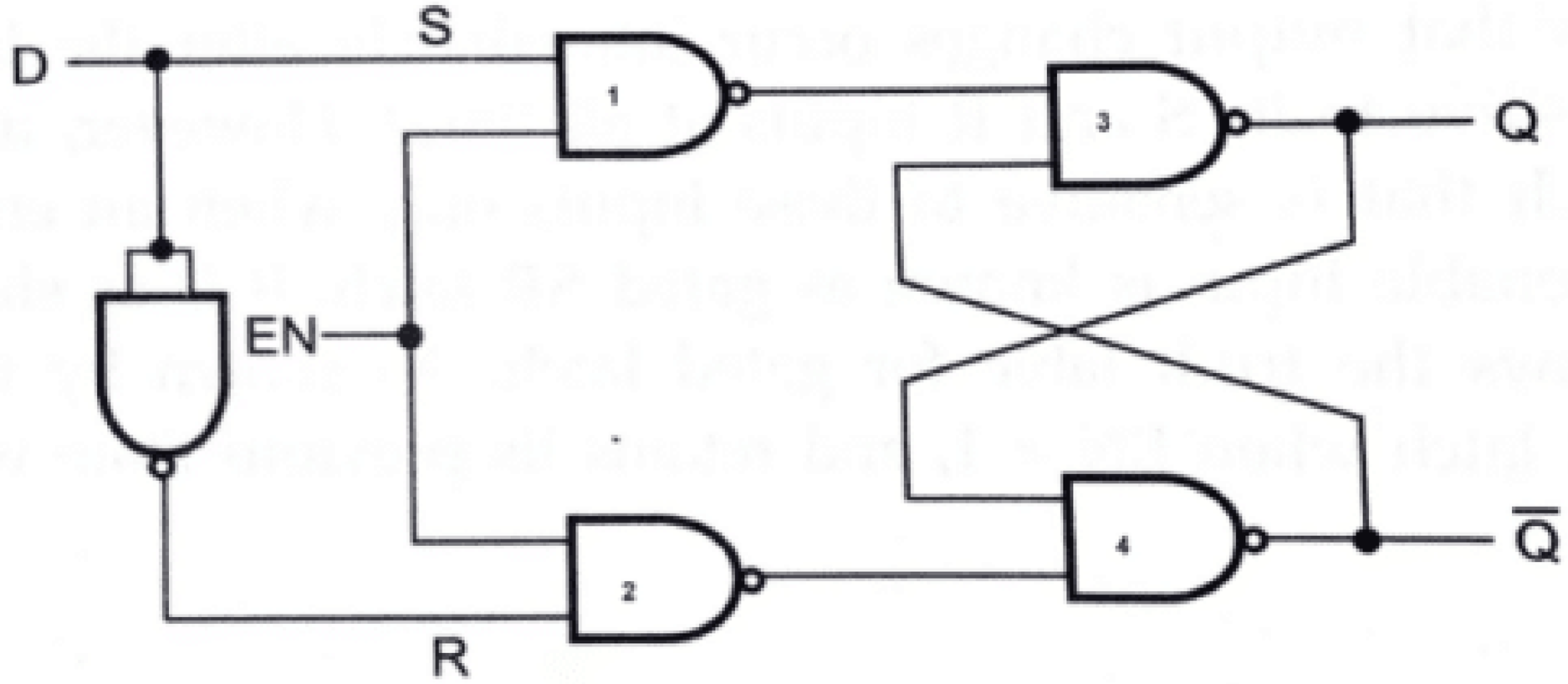
Q Output		Inputs	
Present State	Next State	J_n	K_n
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

EXCITATION TABLE

D Flip Flop

- D flip flop is an electronic devices that is known as “delay flip flop” or “data flip flop” which is used to store single bit of data.D flip flops are synchronous or asynchronous.
- The clock single required for the synchronous version of D flip flops but not for the asynchronous one.
- The D flip flop has two inputs, data and clock input which controls the flip flop.

- when clock input is high, the data is transferred to the output of the flip flop and when the clock input is low, the output of the flip flop is held in its previous state.


D
S
EN
R
1
2
3
4
Q
$\overline{Q}$

D Flip-Flop Truth Table

CLK	D	Q(n+1)	State
–	0	0	RESET
–	1	1	SET

Characteristic Table of D Flip Flop

D	Q(Current)	Q(n+1) (Next)
0	0	0
0	1	0
1	0	1
1	1	1

D is the input, and Q is current state, Qn + 1 is the next state outputs.

Qn+1 will always be zero when D is 0, irrespective of current state of flip flop.

When the input of the flip flop is 1, next state of flip flop will always be 1, regardless of the current state of flip flop.

Characteristic Equation of D Flip Flop

$$Q(n+1) = D(n)$$

- Qn+1 represents the output of flip flop at the next clock cycle.
- Dn is the input to the flip flop at the current clock cycle, and n represents the current clock cycle.
- This characteristic equation of D flip flop states “that the output of the flip flop at the next clock cycle will be equal to the input at the current clock cycle“.

D Flip Flop Excitation Table

Q(n)	D(n)	Q(n+1) (Next)
0	0	0
0	1	1
1	0	0
1	1	1

- When the Qn is 0 and the Dn is also 0, then the Qn+1 becomes 0. This situation explains the condition of “hold” state.
- When the Qn is 0 but Dn is 1, then the Qn+1 becomes 1. This situation explains the condition of “reset” state.
- When the Qn is 1 but Dn is 0, then the Qn+1 becomes 0. This situation explains the condition of “hold” state.
- When the Qn is 1 and the Dn is also 1, then the Qn+1 becomes 1. This situation explains the condition of “reset” state.

Advantages of D Flip Flop

- D flip flop is very simple to design.
- The computation speed of D flip flop is very fast compared to other flip flops.
- D flip flop requires very few components to design which makes it simple to understand.
- Disadvantages of D Flip Flop
- D flip flops are glitch prone. When input varies fast, flip flop output may glitch. Digital circuit glitches are hard to identify and fix.

CHARACTERISTIC EQUATIONS

Summary of the characteristic equations

Flip-flop	Characteristic Equation
D	$Q(t+1) = D$
T	$Q(t+1) = T \oplus Q(t)$
SR	$Q(t+1) = S + R' Q(t)$
JK	$Q(t+1) = J Q(t)' + K' Q(t)$

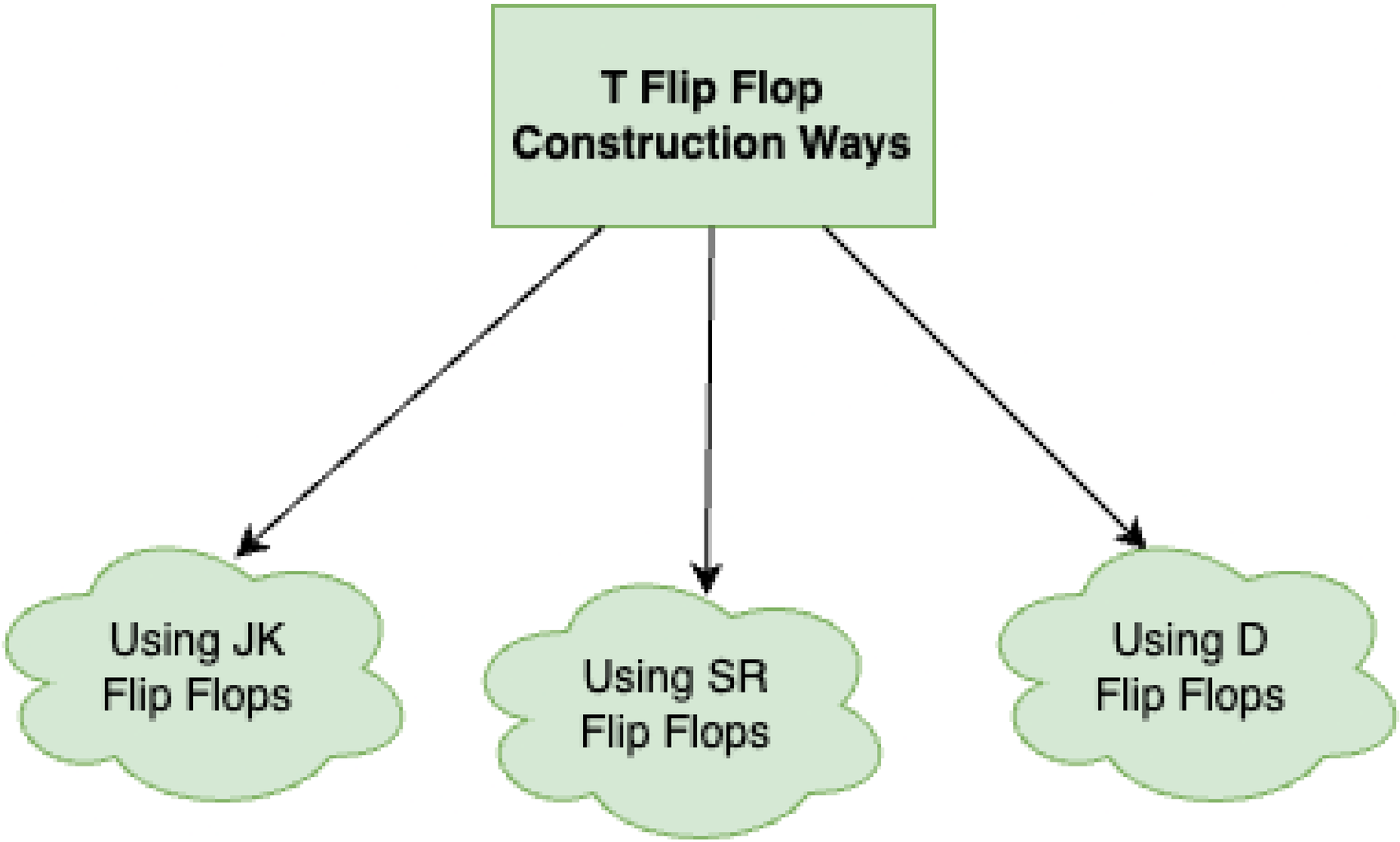
Excitation tables

- Summary of the excitation tables

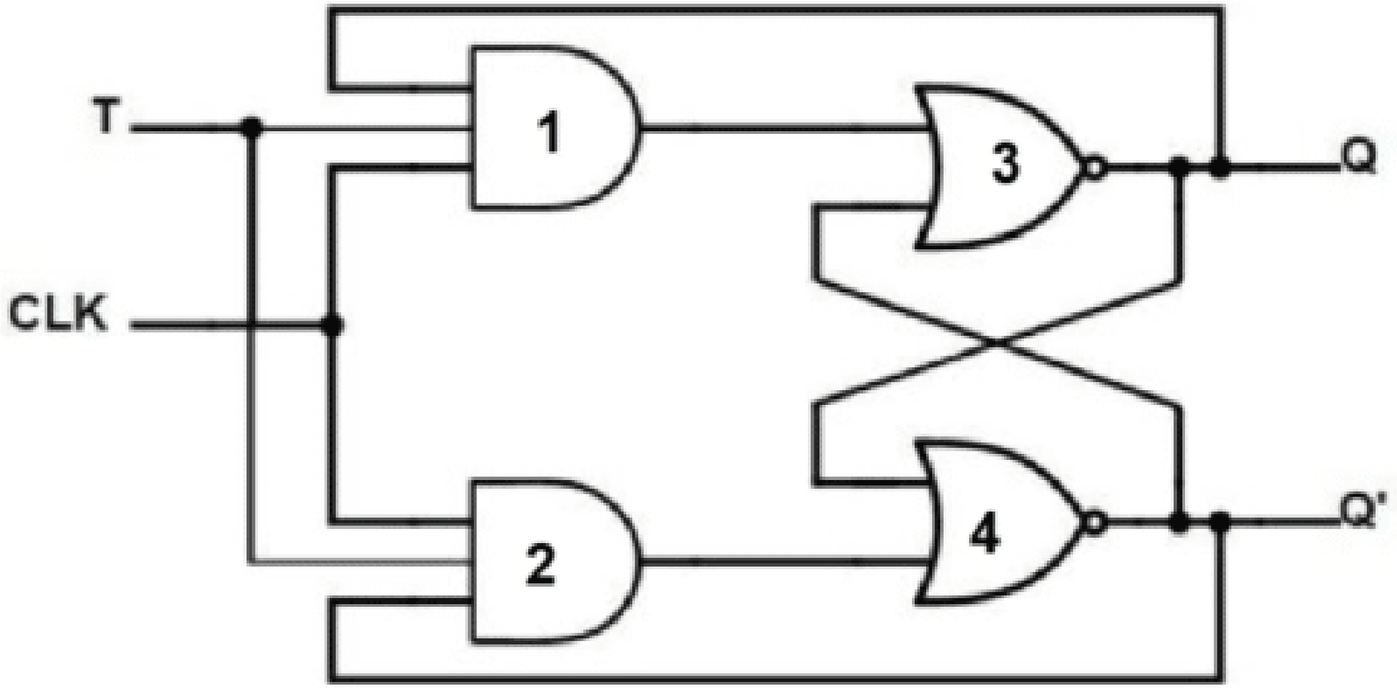
For each state transition Q(t) → Q(t+1)

Q(t)	Q(t+1)	D	T	SR	JK
0	0	0	0	0x	0x
0	1	1	1	10	1x
1	0	0	1	01	x1
1	1	1	0	x0	x0

T Flip Flop


T Flip Flop
Construction Ways
Using JK
Flip Flops
Using SR
Flip Flops
Using D
Flip Flops

T Flip Flop

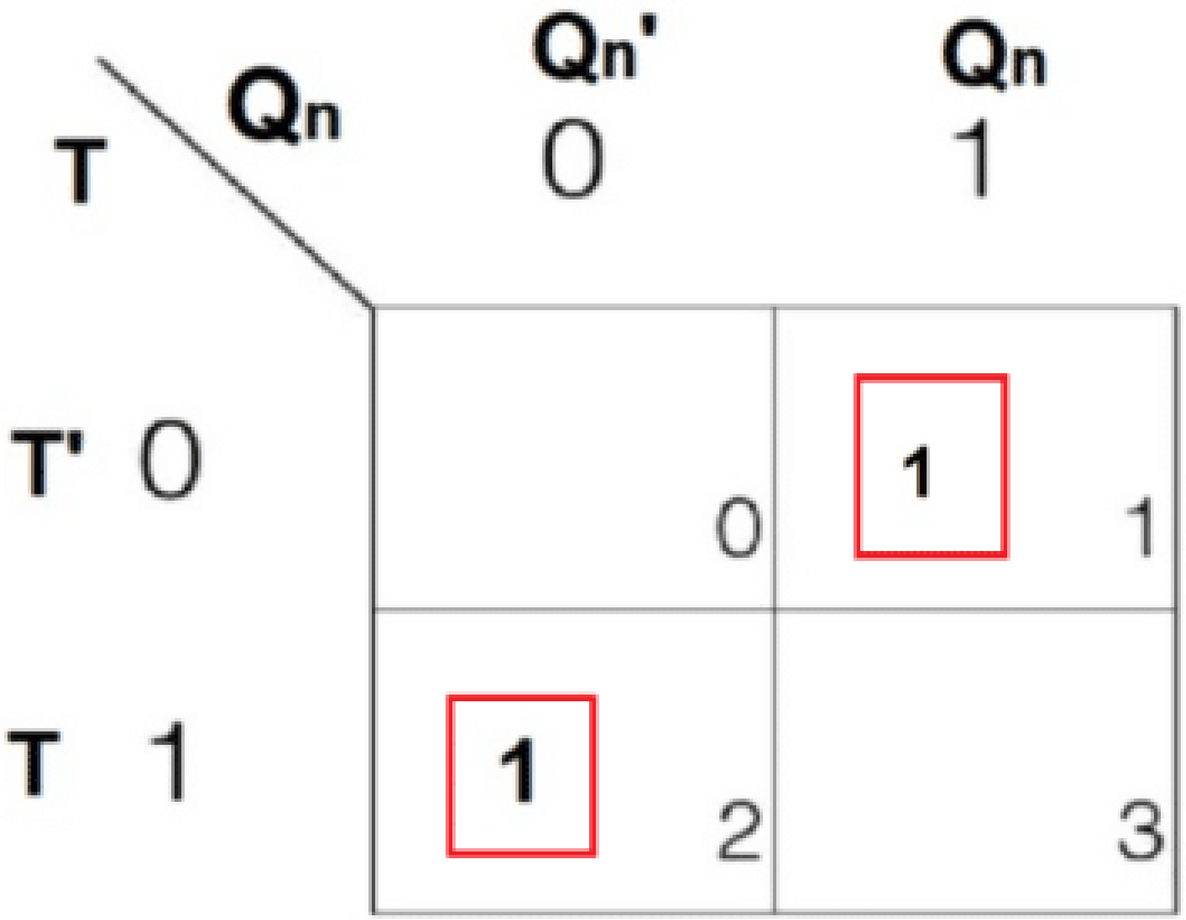


CLK	T	Q(n+1)	State
	0	Q	NO CHANGE
	1	Q’	TOGGLE

T Flip-Flop Truth Table

characteristic table,

T	Qn	Q(n+1)
0	0	0
0	1	1
1	0	1
1	1	0



Q(n+1) = TQn’ + T’Qn = T XOR Qn

CHARACTERISTIC EQUATION

TABLE 1-3 Excitation Table for Four Flip-Flops

SR flip-flop

$Q(t)$	$Q(t+1)$	S	R
0	0	0	×
0	1	1	0
1	0	0	1
1	1	×	0

D flip-flop

$Q(t)$	$Q(t+1)$	D
0	0	0
0	1	1
1	0	0
1	1	1

JK flip-flop

$Q(t)$	$Q(t+1)$	J	K
0	0	0	×
0	1	1	×
1	0	×	1
1	1	×	0

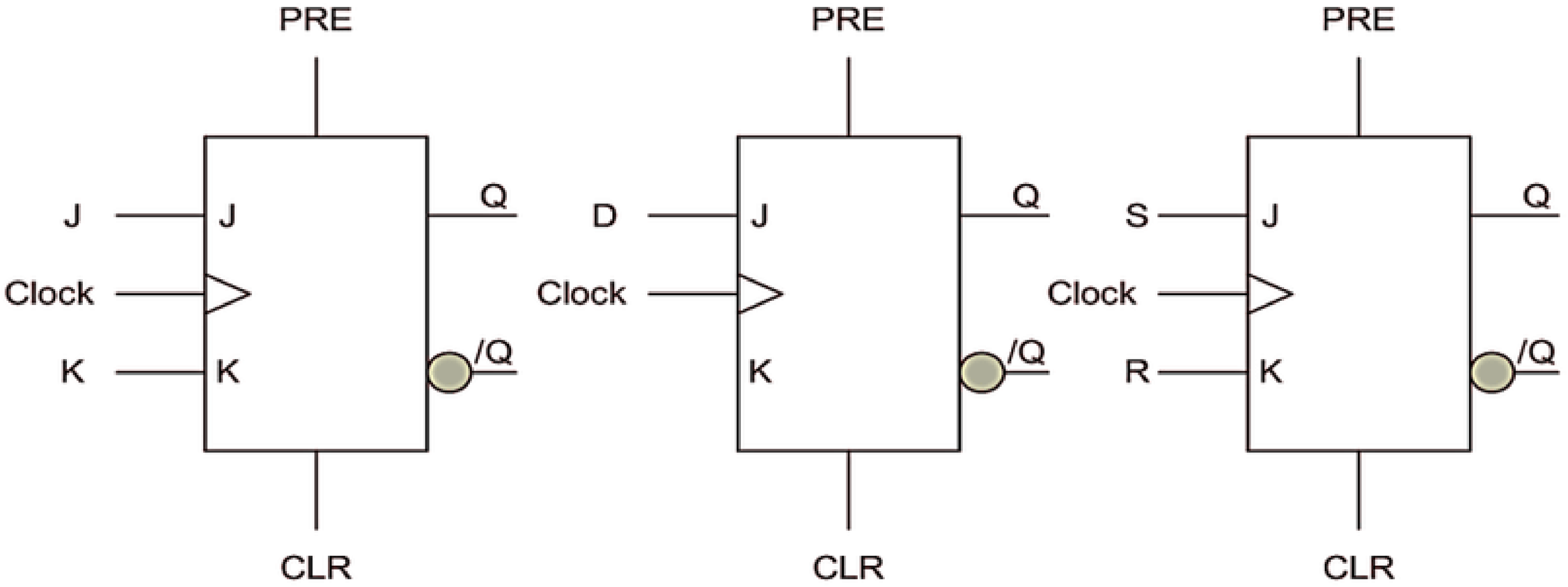
T flip-flop

$Q(t)$	$Q(t+1)$	T
0	0	0
0	1	1
1	0	1
1	1	0

Edge Triggered Flip Flops with Asynchronous Set and Reset

Asynchronous inputs on a flip-flop have control over the outputs (Q and not-Q) regardless of clock input status. These inputs are called the preset (PRE) and clear (CLR). The preset input drives the flip-flop to a set state while the clear input drives it to a reset state.

Asynchronous inputs, just like synchronous inputs, can be engineered to be active-high or active-low. If they’re active-low, there will be an inverting bubble at that input lead on the block symbol, just like the negative edge-trigger clock inputs.


PRE
J
J
Q
Clock
K
K
/Q
CLR
PRE
D
J
Q
Clock
K
/Q
CLR
PRE
S
J
Q
Clock
R
K
/Q
CLR

TRUTH TABLE

INPUTS				OUTPUTS	
$\overline{PR}$	$\overline{CLR}$	CLK	D	Q	$\overline{Q}$
0	1	X	X	1	0
1	0	X	X	0	1
0	0	X	X	X	X
1	1	↑	1	1	0
1	1	↑	0	0	1
1	1	0	X	Q_0	$\overline{Q}_0$

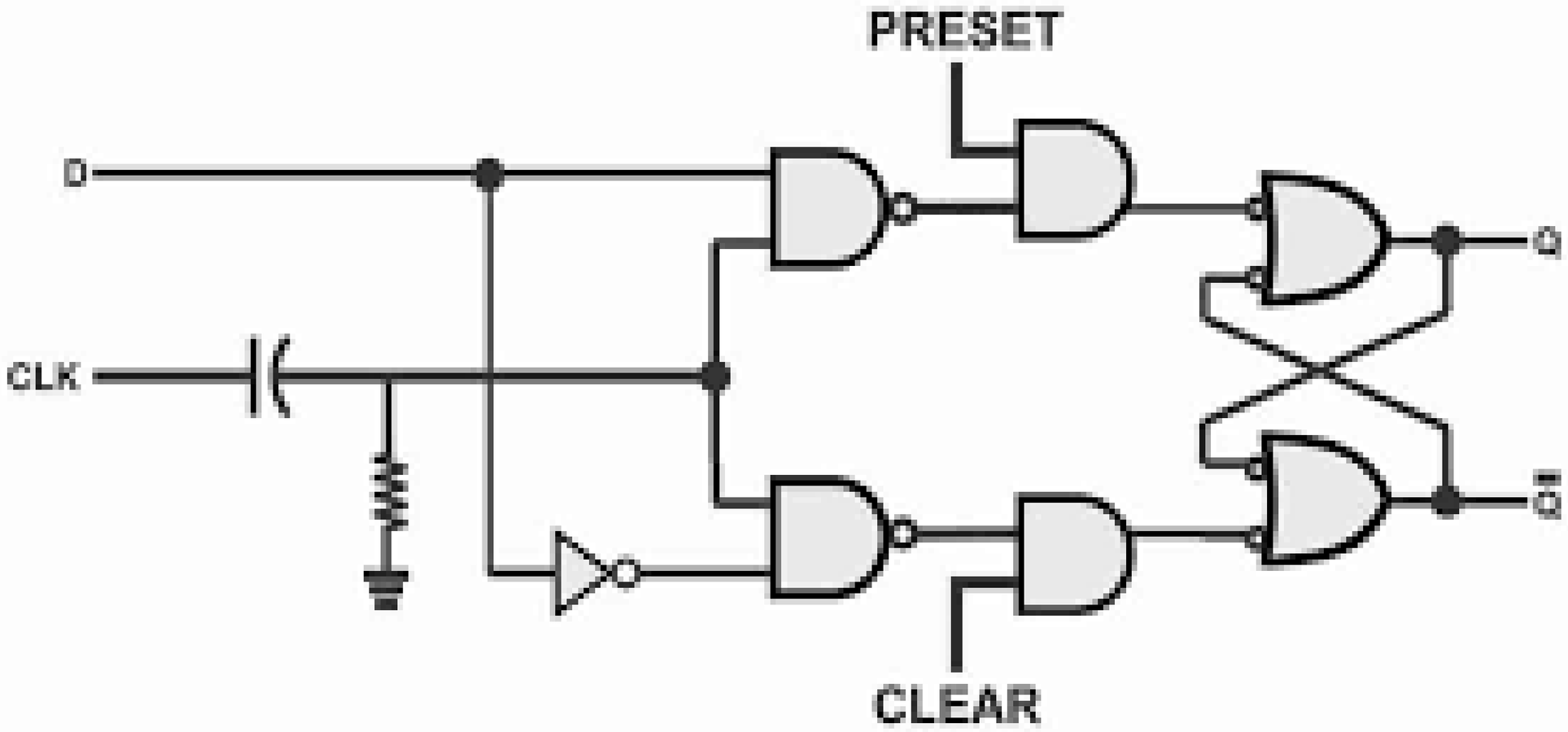
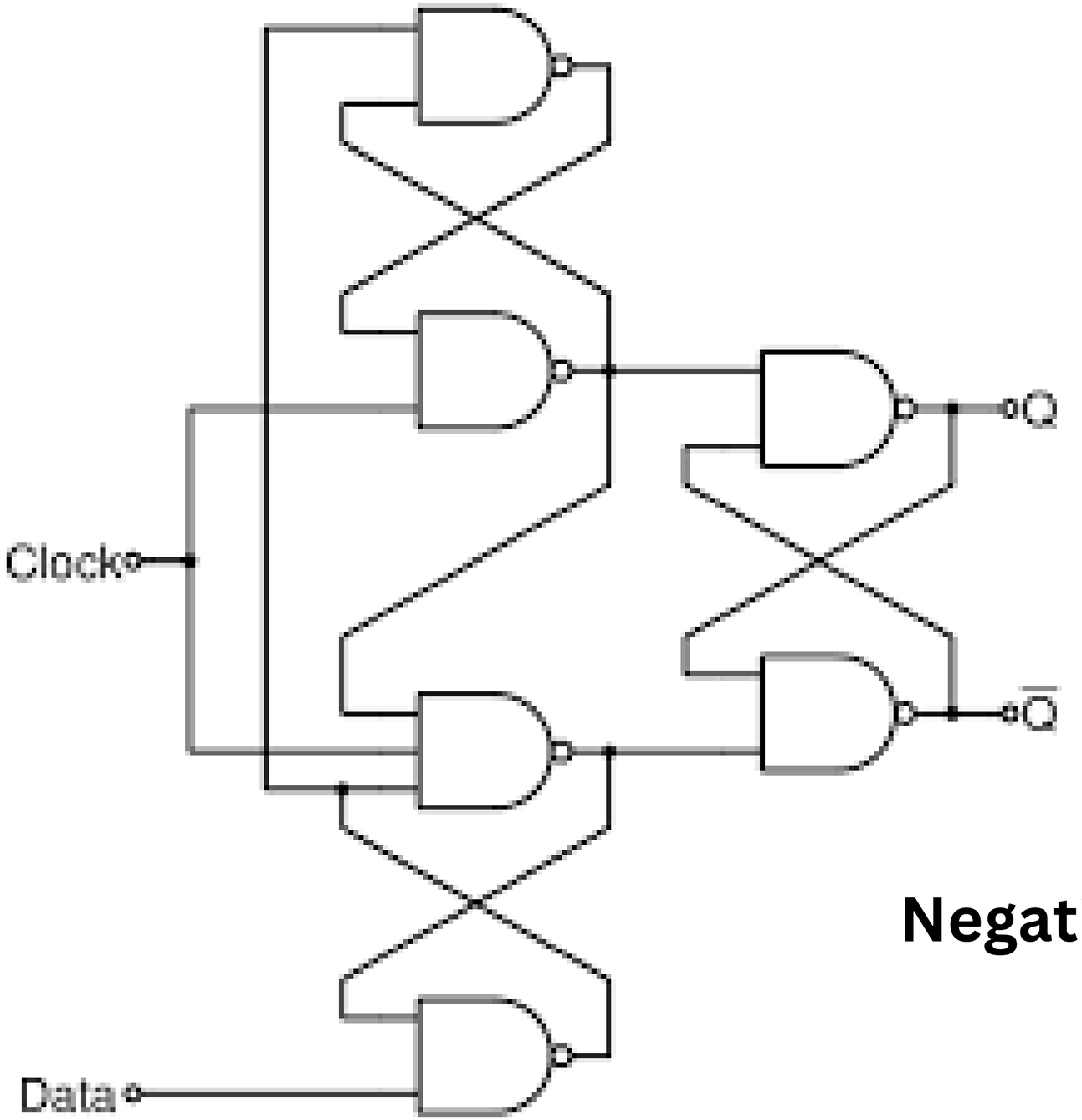
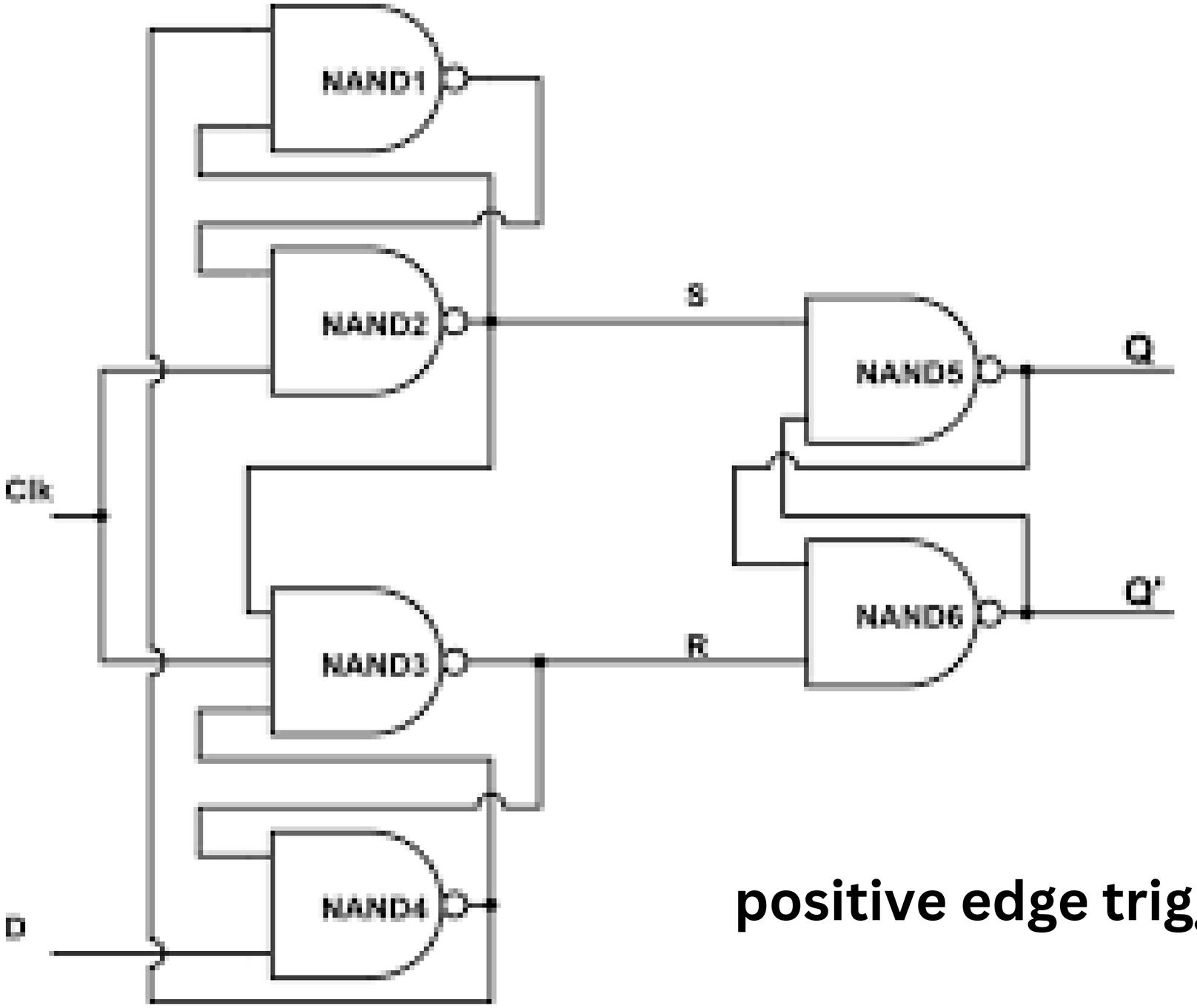


Fig 5.21 Edge-triggered D flip-flop with preset and clear.

D flip flop with synchronous inputs preset and clear



Negative edge triggered d flip flop



positive edge triggered d flip flop