Sequential Circuits

Instructor: Saraju P. Mohanty

Part 1

- Definition of sequential circuit
- Latches
- Flip-Flops
- Flip-Flop symbols
- Few Terminology

NOTE: Mostly adapted from Dr. Valavanis lectures.

Sequential Circuit Definition

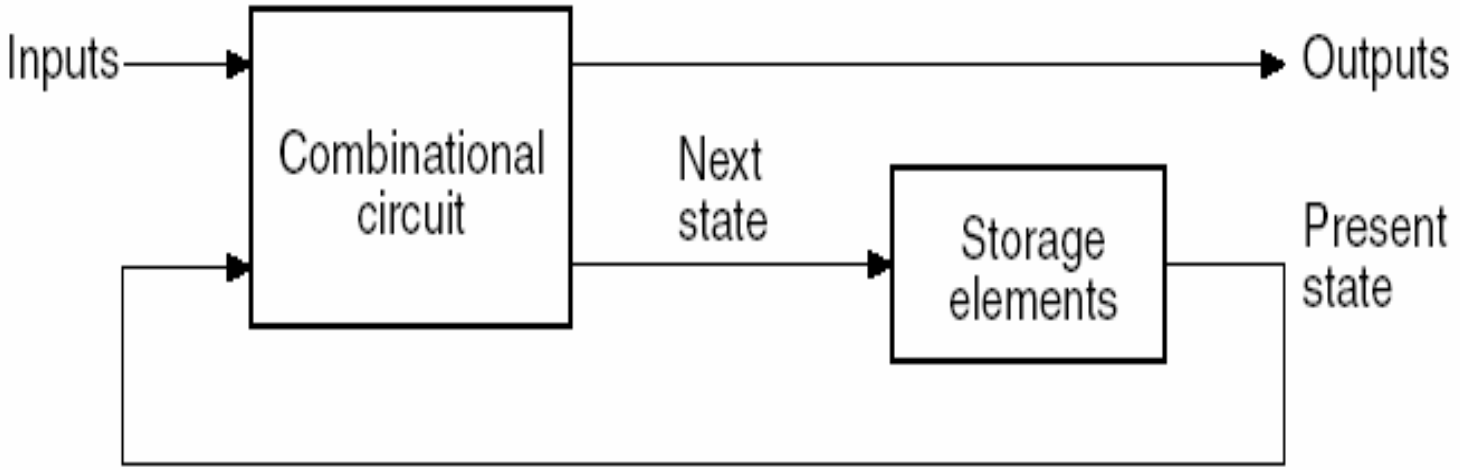
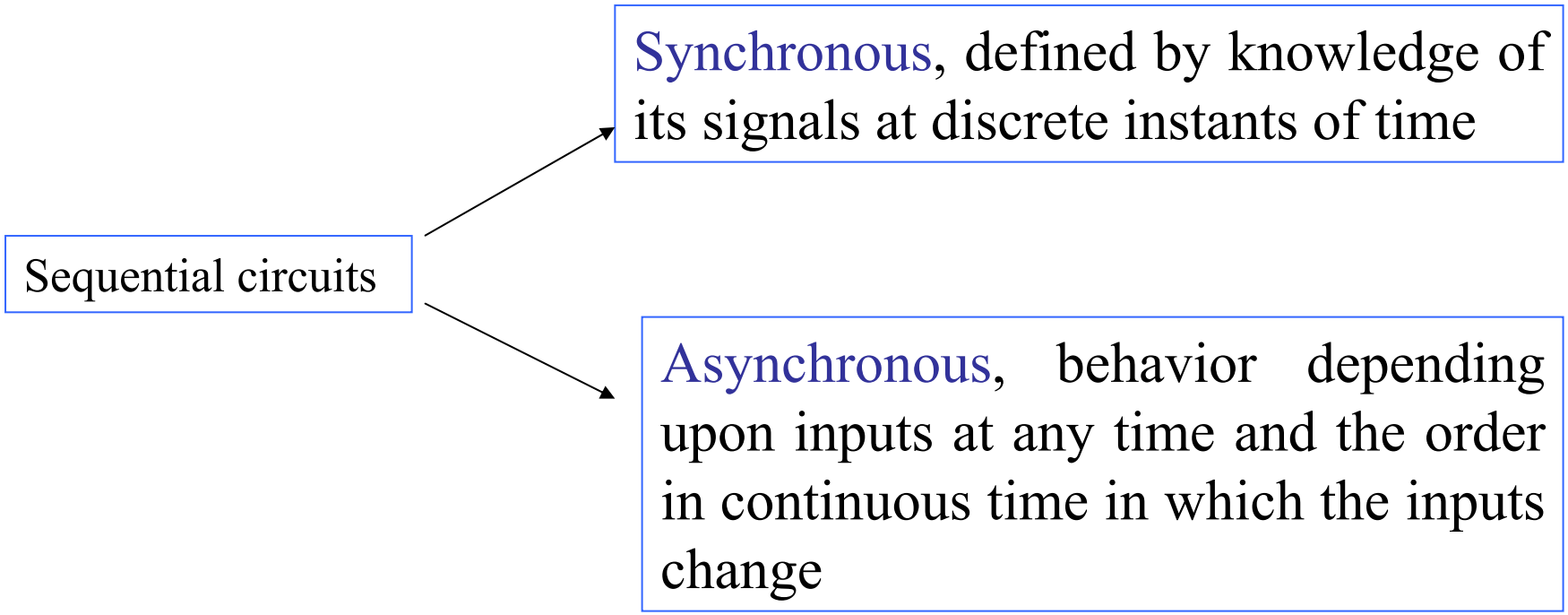


Fig. 4-1 Block Diagram of a Sequential Circuit

Combinatorial circuit + storage element = sequential circuit

Capable of storing binary information that defines at any given time the state of the sequential circuit at that time

Sequential Circuit Definition …



Also, next state of the storage element is a function of the inputs and the present state. Thus, a sequential circuit is specified by a time sequence of inputs, internal states and outputs.

Sequential Circuit Definition: Storage elements

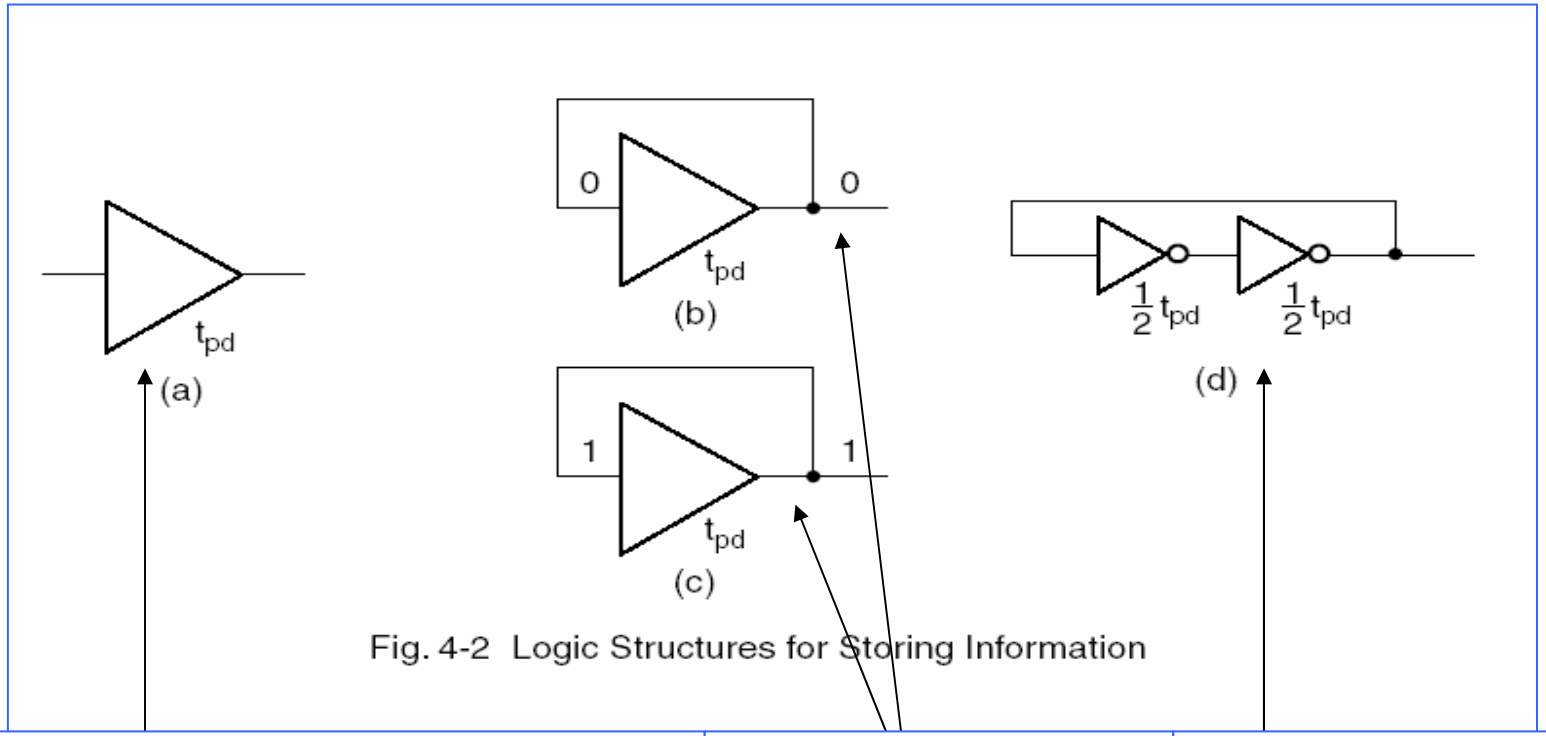


Fig. 4-2 Logic Structures for Storing Information

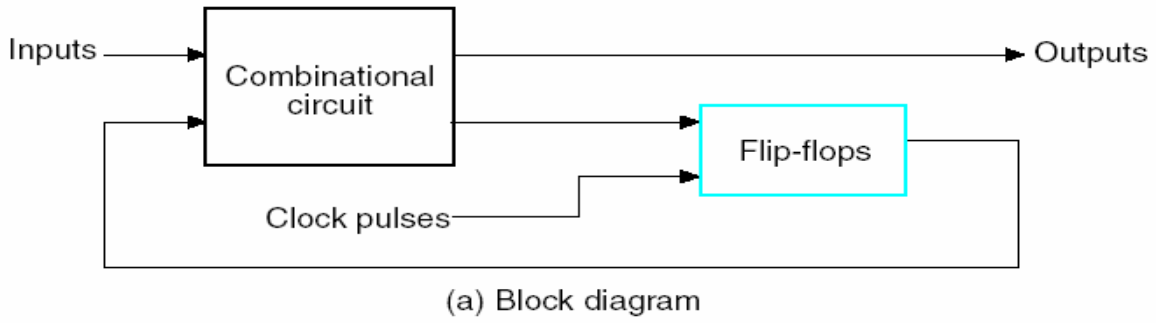
Buffer: info present in the input at time ***t***, appears in the output at $t+t_{pd}$ → information stored for time t_{pd}

Information is stored indefinitely

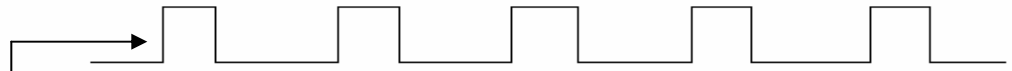
Buffer implementation – 2 inverters

Storage can be constructed from logic with delay connected in a closed loop. Any loop that produces such storage must also have a property possessed by the buffer, namely, that there must be no inversion of the signal around the loop.

Synchronous sequential circuits



(a) Block diagram



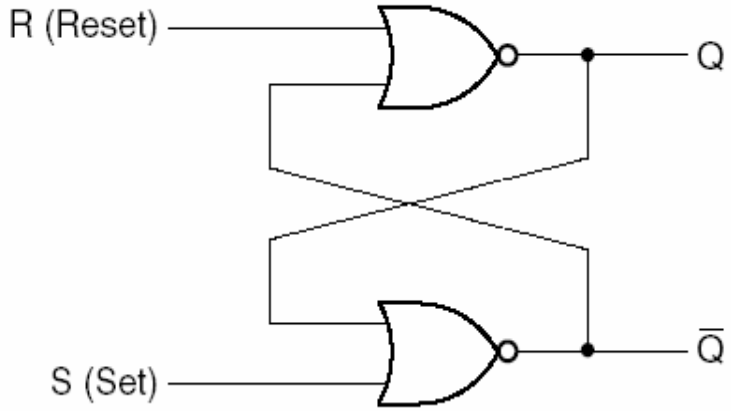
(b) Timing diagram of clock pulses

Fig. 4-3 Synchronous Clocked Sequential Circuit

Flip-Flops are binary storage device, capable of storing one bit of info following specific timing characteristics.

Synchronization is achieved by a timing device called a clock generator which produces a periodic train of clock-pulses. Pulses are distributed throughout the system in such a way that synchronous storage elements are affected only in some specified relationship to every pulse.

Latches: SR using NOR Gates



(a) Logic diagram

S	R	Q	$\overline{Q}$	
1	0	1	0	Set state
0	0	1	0	
0	1	0	1	Reset state
0	0	0	1	
1	1	0	0	Undefined

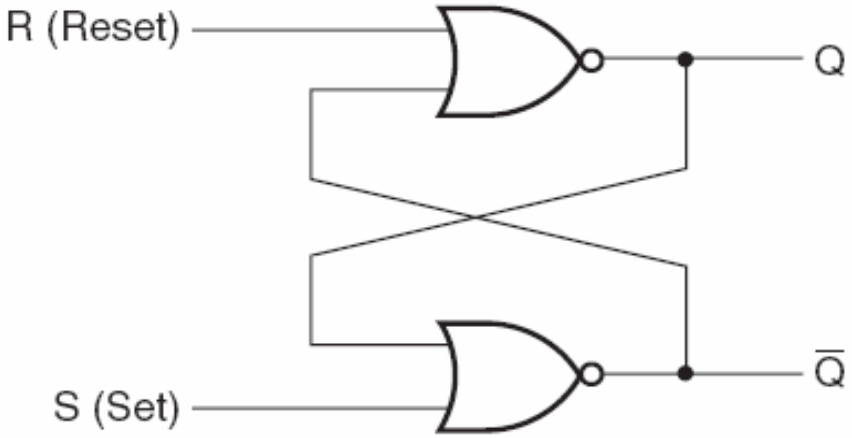
(b) Function table

Fig. 4-4 *SR* Latch with NOR Gates

Under normal conditions, both inputs remain at 0 unless the state is to be changed. **Initial condition** is S=1, R=0, bringing the system to the set state.

S must go to 0 before R is changed to 1 to avoid occurrence of undefined state.

Latches: SR using NOR Gates ….



(a) Logic diagram

S	R	Q	$\overline{Q}$	
1	0	1	0	Set state
0	0	1	0	
0	1	0	1	Reset state
0	0	0	1	
1	1	0	0	Undefined

(b) Function table

Fig. 4-4 *SR* Latch with NOR Gates

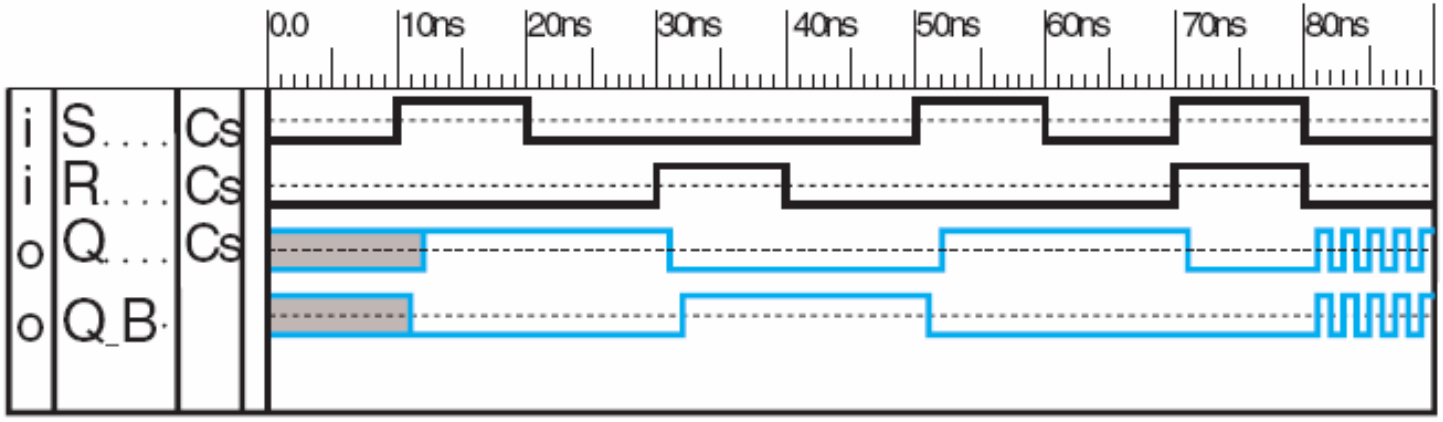
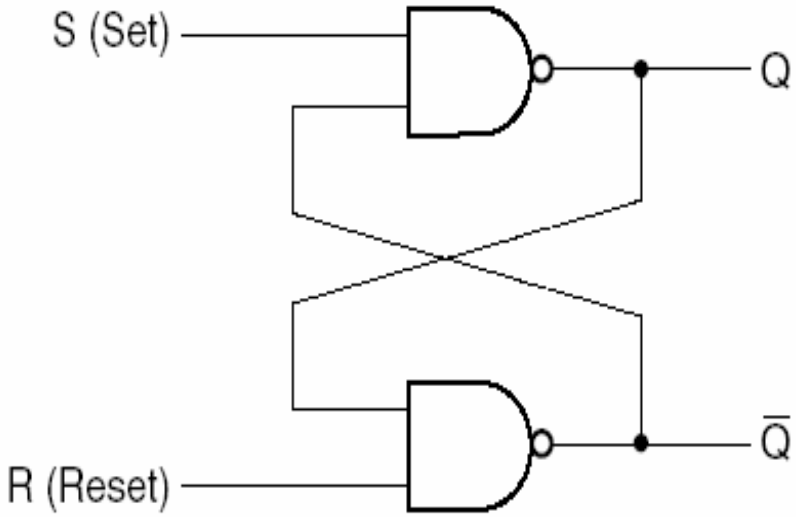


Fig. 4-5 Logic Simulation of *SR* Latch Behavior

Latches: SR using NAND Gates



(a) Logic diagram

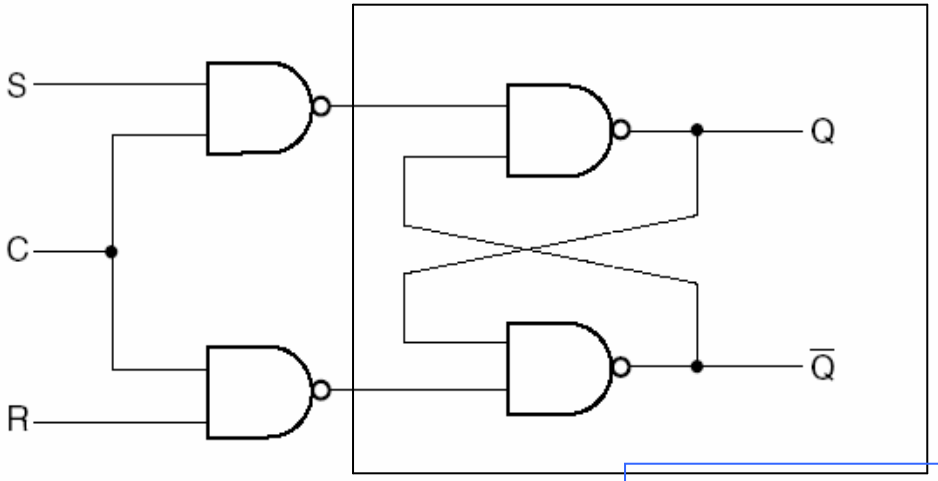
S	R	Q	$\overline{Q}$	
0	1	1	0	Set state
1	1	1	0	
1	0	0	1	Reset state
1	1	0	1	
0	0	1	1	Undefined

(b) Function table

Fig. 4-6 $\overline{S}\overline{R}$ Latch with NAND Gates

Input signals require the complements of the latch with NOR gates.

Latches: SR using NAND Gates with a Control Input



(a) Logic diagram

C	S	R	Next state of Q
0	X	X	No change
1	0	0	No change
1	0	1	Q = 0; Reset state
1	1	0	Q = 1; Set state
1	1	1	Undefined

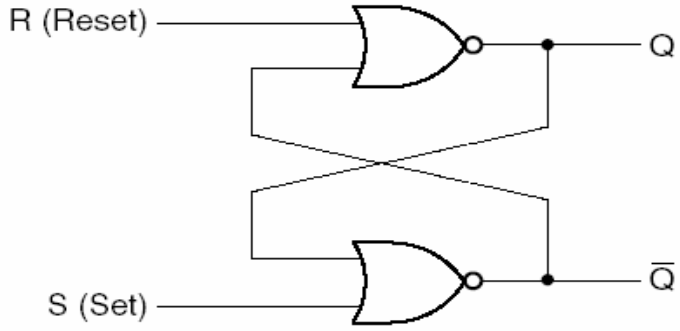
(b) Function table

Basic S'R' latch

Fig. 4-7 *SR* Latch with Control Input

Control input C acts as an enable signal for the other two inputs. When C=1, info from the inputs S, R is allowed to affect the S'R' latch. C=0 disables the circuit regardless of S, R.

Latches: SR NAND Vs SR NOR Gates



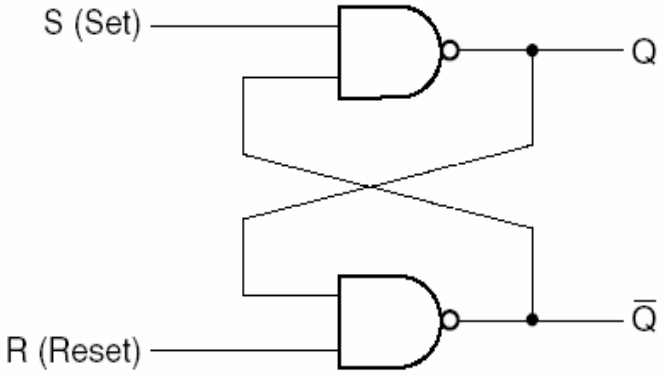
(a) Logic diagram

S	R	Q	$\overline{Q}$	
1	0	1	0	Set state
0	0	1	0	
0	1	0	1	Reset state
0	0	0	1	
1	1	0	0	Undefined

(b) Function table

Fig. 4-4 *SR* Latch with NOR Gates

Why S'R' ??



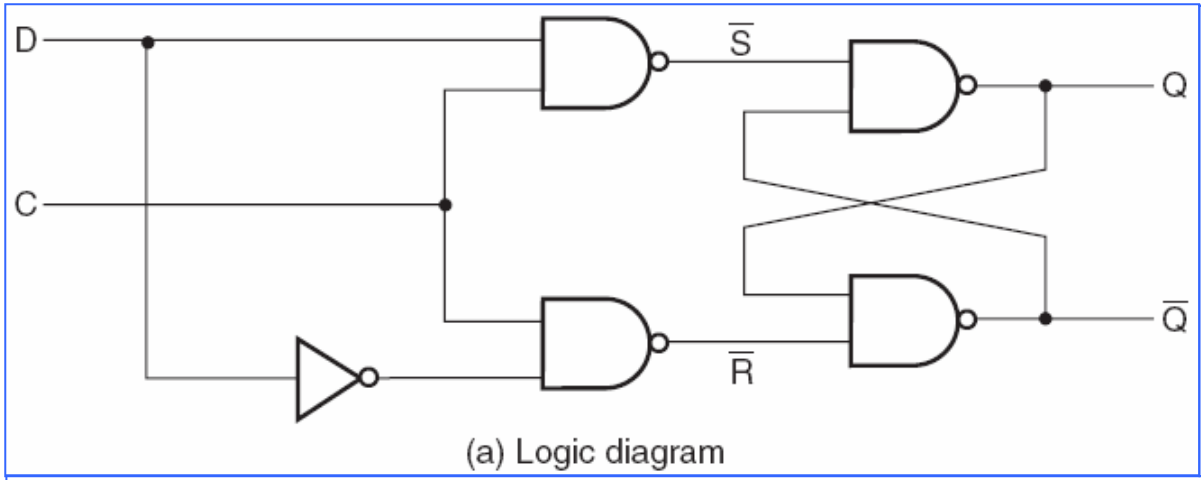
(a) Logic diagram

S	R	Q	$\overline{Q}$	
0	1	1	0	Set state
1	1	1	0	
1	0	0	1	Reset state
1	1	0	1	
0	0	1	1	Undefined

(b) Function table

Fig. 4-6 $\overline{S}\,\overline{R}$ Latch with NAND Gates

Latches: The D- Latch using NAND gates



(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

Objective: Eliminate the undesirable undefined state in the SR latch by ensuring that the inputs are never equal to 1 at the same time. As long as C=0, circuit cannot change state. D input is sampled when C=1.

Latches: D- Latch using Transmission Gates

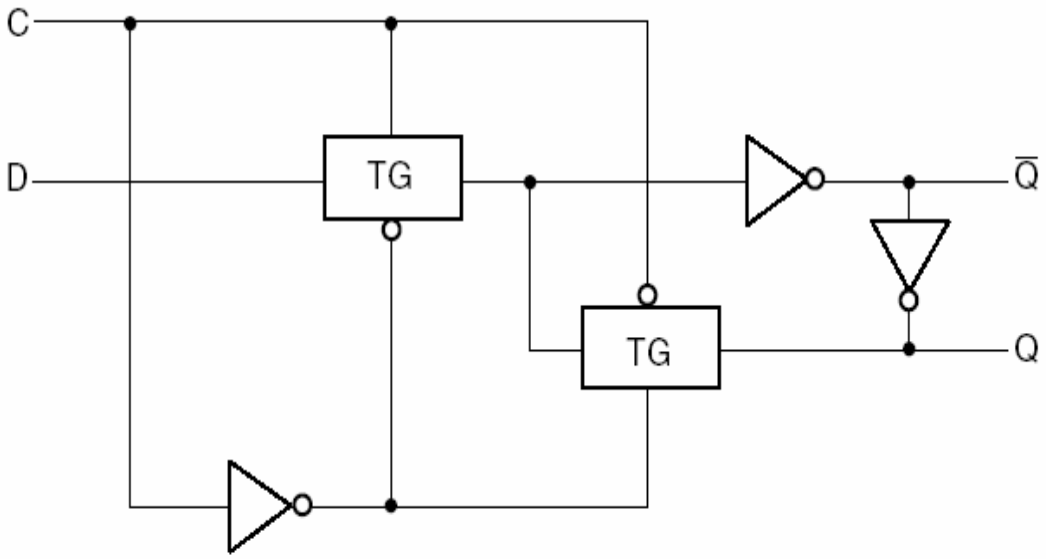


Fig. 4-9 *D* Latch with Transmission Gates

C=1, the TG connected to input D conducts, other TG disconnected, thus Q is D.

C=0, TG connected to output conducts, other TG disconnected two inverters loop and retains the value of D.

Latches Vs Flip-Flops

•Trigger: state at which switching is allowed by momentary change in value of control input.

•Transparent: The D latch with clock pulses on its control input is triggered every time a pulse to the logic-1 level occurs. As long as the pulse remains at the active 1 level, any changes in the data input will change the state of the latch.

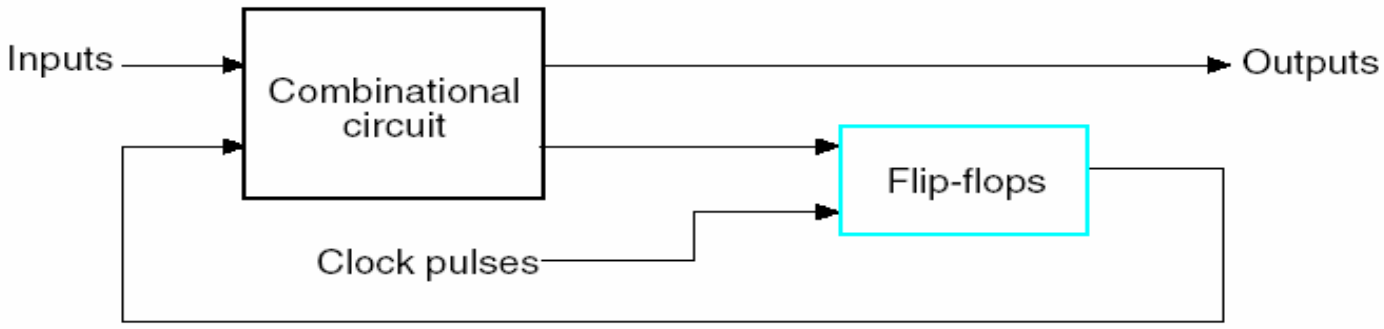
•Difficulty in a latch: A latch output cannot be applied directly or through combinatorial logic to the input of the same or another latch when all latches are triggered by a single clock signal. If the inputs applied to the latches change while the clock pulse is still in the logic-1 level, the latches will respond to new state values of other latches instead of the original state values, and a succession of changes of state instead of a single one may occur, which may lead to unpredictable situation.

Flip-flops

- Storage elements used in clocked sequential circuits are called flip-flops.
- A flip-flop (FF) is a binary storage device that is capable of storing one bit of information, possessing certain timing characteristics.
- A flip-flop has one or two outputs, one for the normal value of the bit stored, and one optional one for the complemented stored bit value.
- Binary information can enter a flip-flop in a variety of ways, thus giving rise to different types of flip-flops.
- Flip-flops are constructed from latches.

Flip-flops ….

- Latch state in a flip-flop is allowed to switch by a momentary change in value of the control unit.
- This change is called a trigger and it enables, or triggers, the flip-flop.



(a) Block diagram

(b) Timing diagram of clock pulses

Fig. 4-3 Synchronous Clocked Sequential Circuit

FF data inputs are derived in part from the outputs of the same and other ff.

Flip-flops …

•The key to FF proper operation is to prevent them from being transparent. In a FF, before an output can change, the path from its inputs to its outputs is broken. So a FF "cannot see" the change of its output or of the outputs of other, at its input during the same clock pulse.

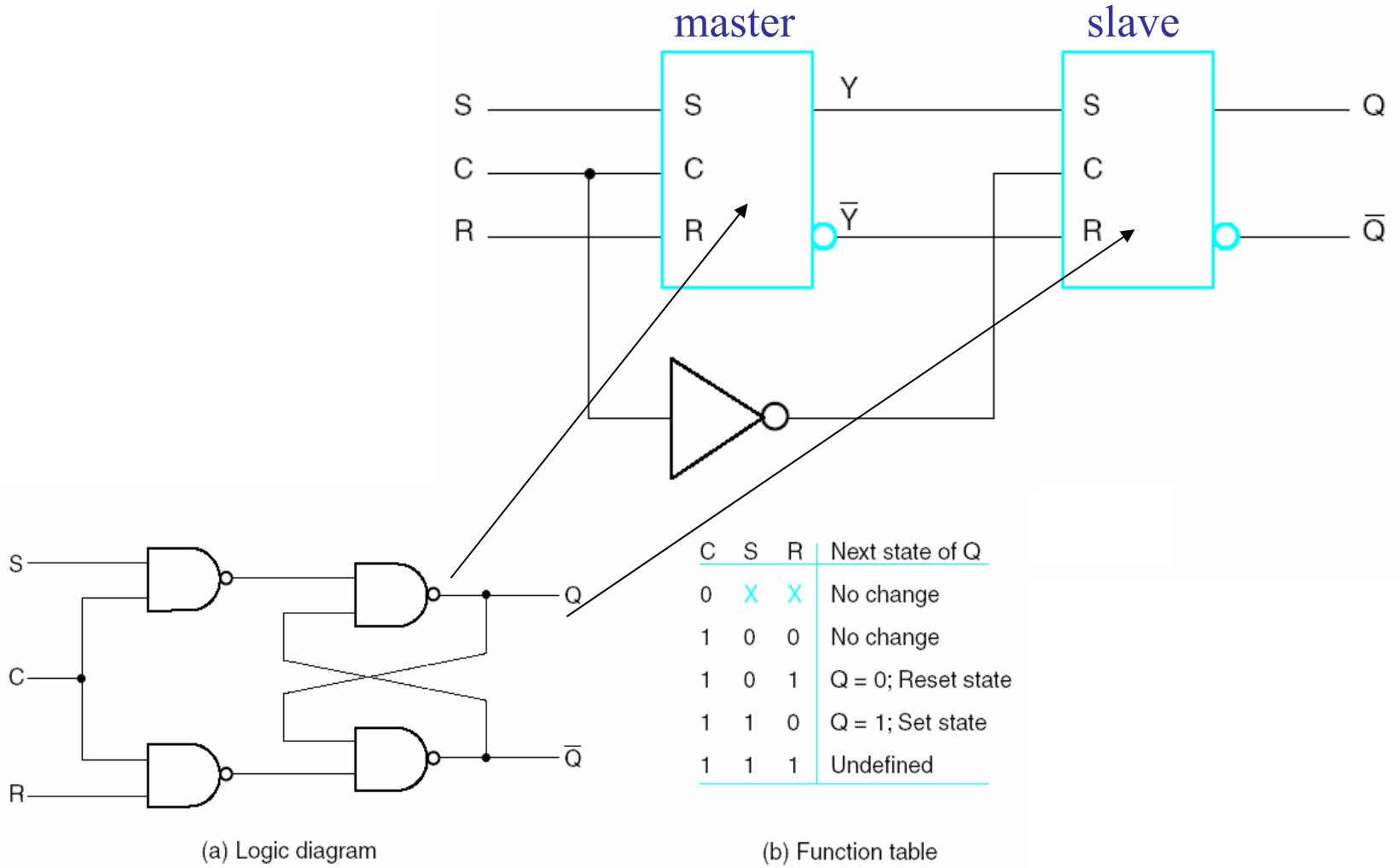
•Thus, the new state of a flip-flop depends only on the immediately preceding state, and the flip-flops do not go through multiple changes of state.

There are two ways latches are combined to form a flip-flop.

•Combine two latches such that: 1) inputs presented to the FF when a clock pulse is present control its state and 2) the ff state changes only when a clock pulse is not present. This is the **master-slave** flip-flop.

•Produce a flip-flop that triggers only during a signal transition from 0 to 1 (or from 1 to 0) on the clock and that it is disabled at all other times, including for the duration of the clock pulse. This is the **edge-triggered** flip-flop.

Flip-Flop: SR master-slave



(a) Logic diagram

C	S	R	Next state of Q
0	X	X	No change
1	0	0	No change
1	0	1	Q = 0; Reset state
1	1	0	Q = 1; Set state
1	1	1	Undefined

(b) Function table

Fig. 4-7 *SR* Latch with Control Input

Flip-Flop: SR master-slave …

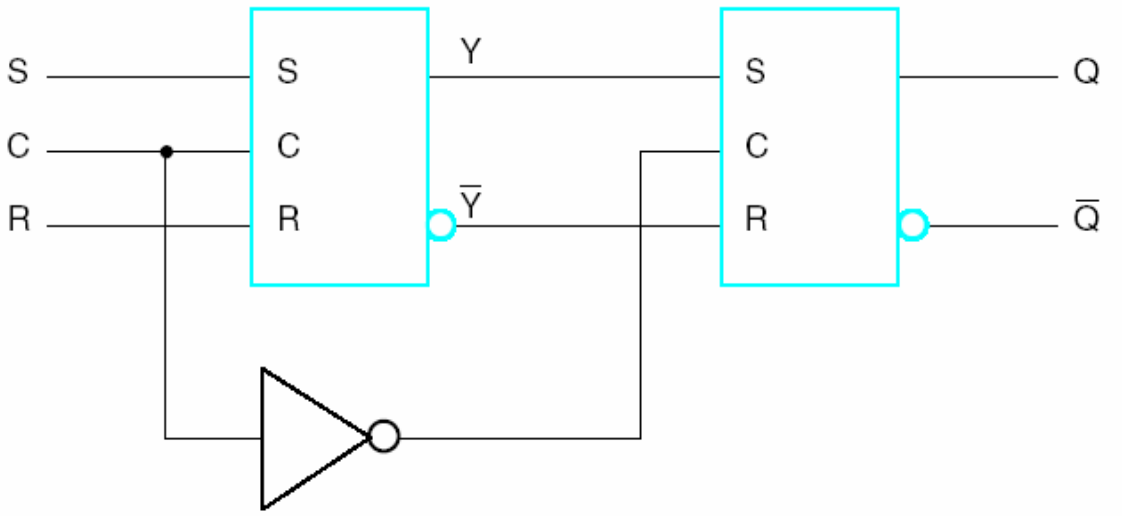
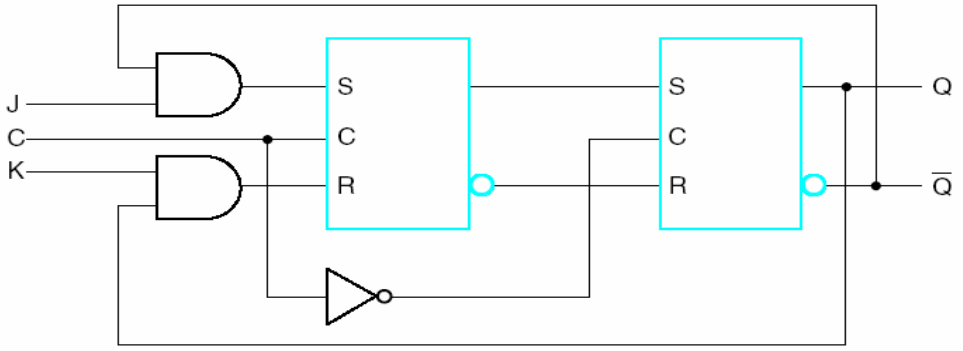


Fig. 4-10 *SR* Master-Slave Flip-Flop

When C=0, output of inverter is 1, slave latch is enabled and its output Q is equal to the master output Y. Master latch is disabled.

When C=1 the values on S, R control the value stored in the master latch Y. Slave is disabled. Any changes in external inputs affect Y but not the slave output Q.

Flip-Flop: SR Vs JK master-slave



(a)

J	K	Next State of Q
0	0	Q
0	1	0
1	0	1
1	1	$\overline{Q}$

(b)

Fig. 4-12 Master-Slave *JK* Flip-Flop

SR master-slave

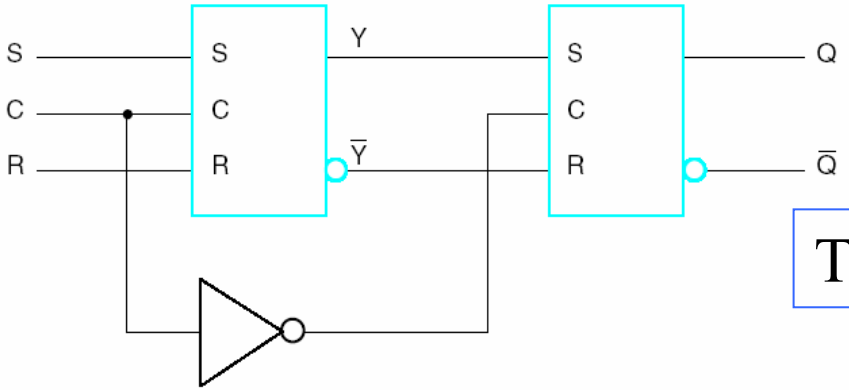
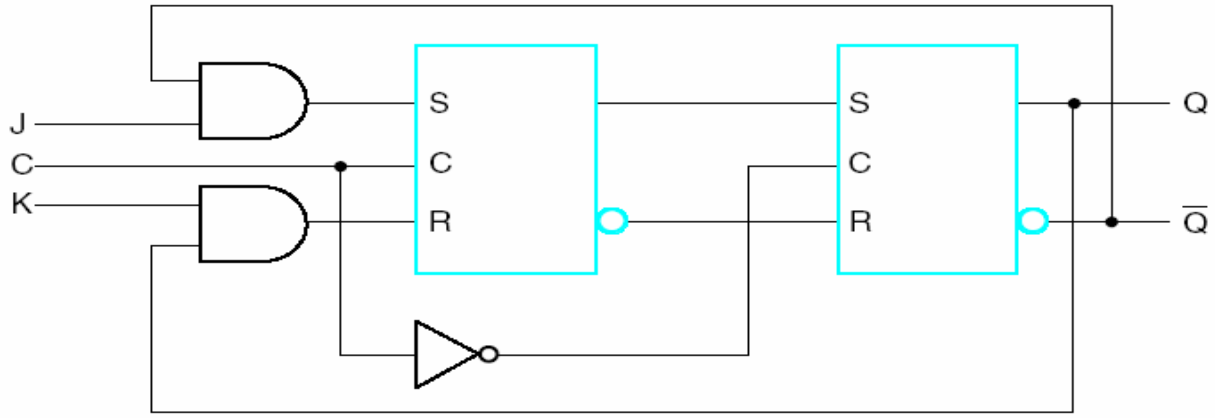


Fig. 4-10 *SR* Master-Slave Flip-Flop

The difference between the two ??

Flip-Flop: JK master-slave



(a)

J	K	Next State of Q
0	0	Q
0	1	0
1	0	1
1	1	$\overline{Q}$

(b)

The condition of both inputs J, K being 1, causes the output to complement its value. K input behaves like the S to set the ff. The K is similar to R for resetting the ff. The only difference is the response to the 11 input.

Flip-Flop: master-slave ….

- For reliable sequential circuit operation, all signals must propagate from the outputs of FF, through the combinatorial circuit, and back to inputs of FF, while the clock pulse remains at the logic-0 level. Any changes occurring at the inputs of FF after the clock pulse goes to logic-1 level are recognized by the outputs. Due to this there may be erroneous transfer of values from the master to the slave.

- Two consequences of this behavior:
  - Master-slave ff is also referred to as pulse-triggered ff since it responds to input values that cause a state change and occur anytime during its clock pulse
  - Design circuit so that combinatorial circuit delays are very short to prevent S, R changing during the clock pulse.

Flip-Flop: Edge-triggered

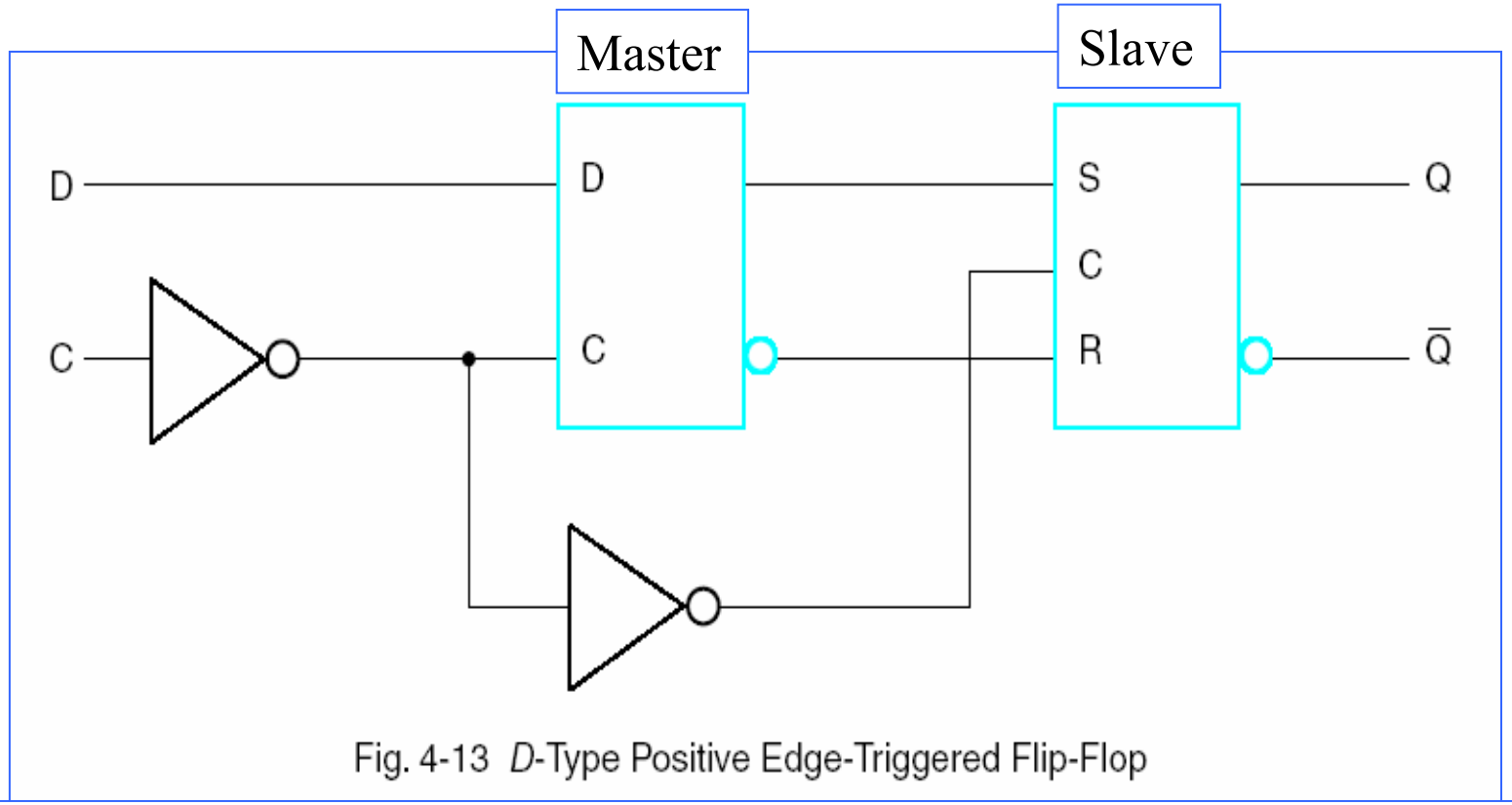


Fig. 4-13 *D*-Type Positive Edge-Triggered Flip-Flop

- Edge-triggered FF ignores the pulse while it is at a constant level and triggers only during a transition of the clock signal.
- Slave, either SR latch or a D latch

Flip-Flop: Edge-triggered D-Type

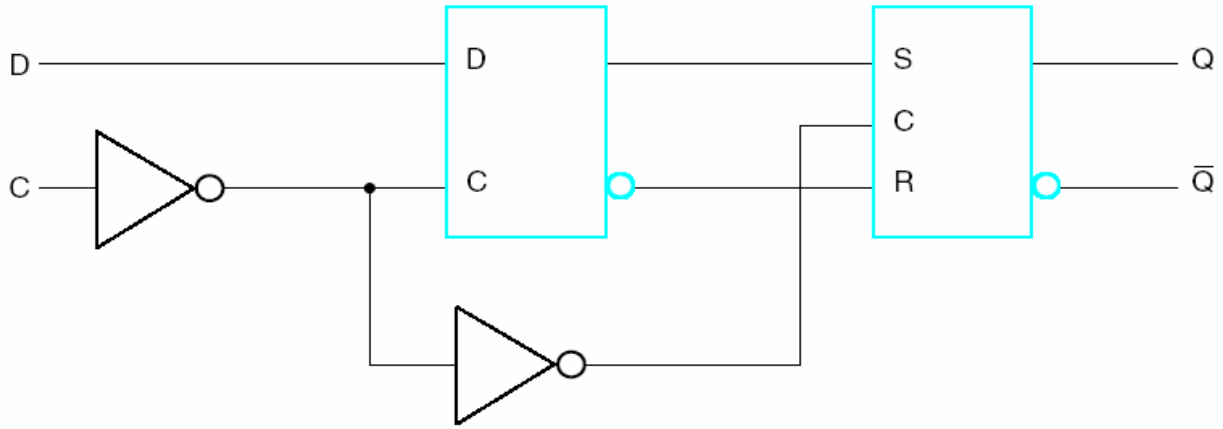


Fig. 4-13 *D*-Type Positive Edge-Triggered Flip-Flop

Master latch is a D latch, FF exhibits edge-triggered rather than master-slave behavior.

For the clock input 0, master latch is enabled and transparent and follows the D input value. Slave latch is disabled and holds the FF state fixed. When positive edge occurs, clock input changes to 1, disabling master so that its value is fixed, and enabling slave latch so that it copies the state of the master latch.

Flip-Flop: Edge-triggered JK Type

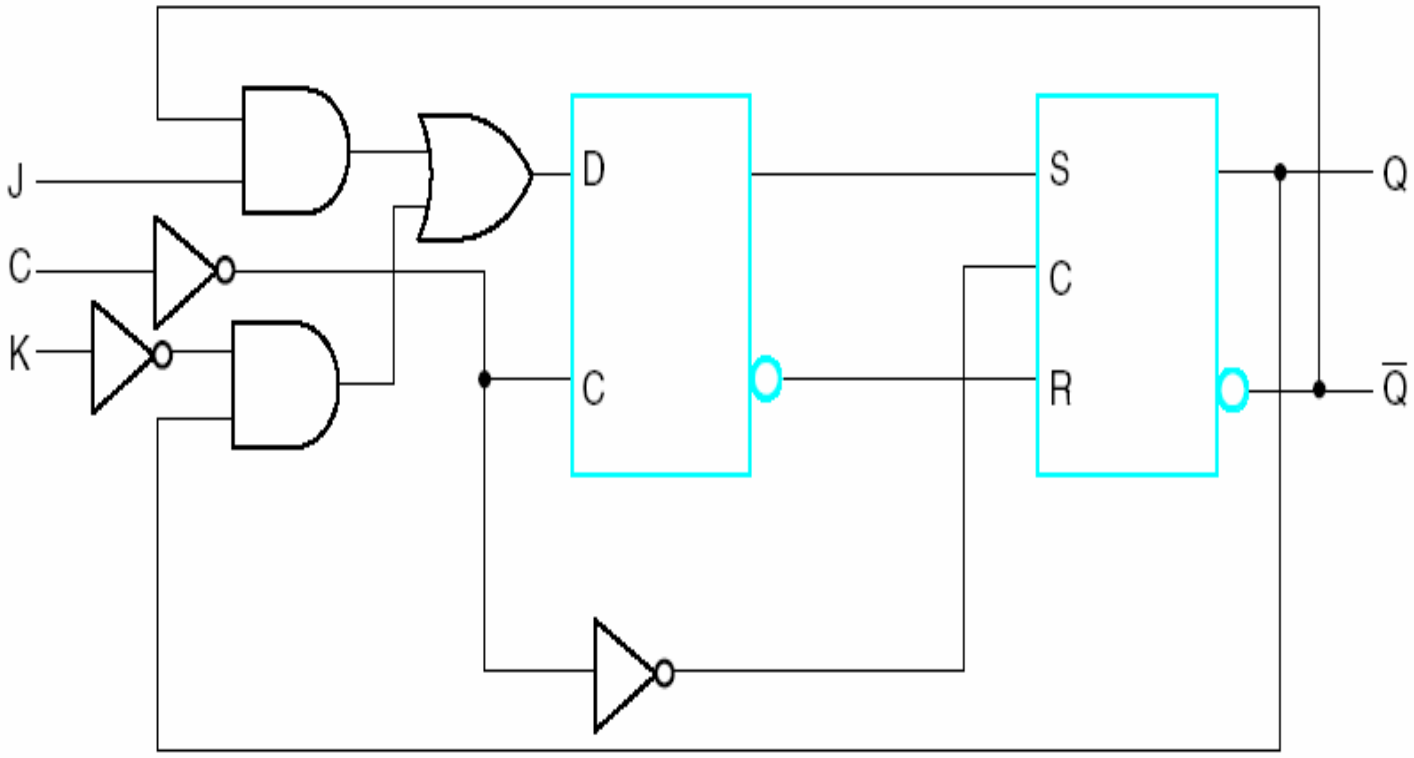


Fig. 4-14 Positive Edge-Triggered *JK* Flip-Flop

Flip-Flops: Standard Graphics Symbols

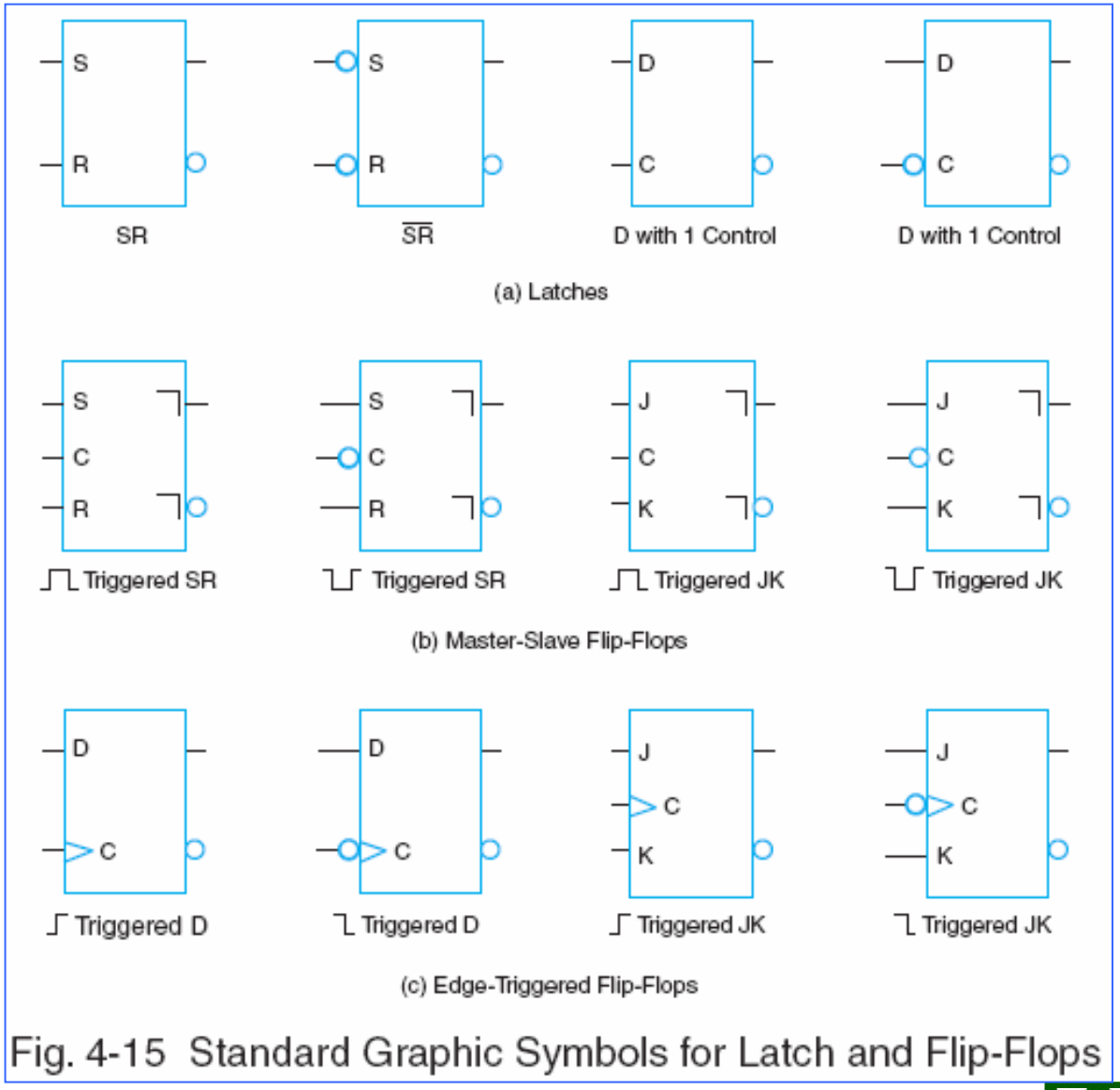


Fig. 4-15 Standard Graphic Symbols for Latch and Flip-Flops

Flip-Flops: Characteristic Tables

TABLE 4-1
Flip-Flop Characteristic Tables

(a) *JK* Flip-Flop

J	K	$Q(t+1)$	Operation
0	0	$Q(t)$	No change
0	1	0	Reset
1	0	1	Set
1	1	$\overline{Q}(t)$	Complement

(b) *SR* Flip-Flop

S	R	$Q(t+1)$	Operation
0	0	$Q(t)$	No change
0	1	0	Reset
1	0	1	Set
1	1	?	Undefined

(c) *D* Flip-Flop

D	$Q(t+1)$	Operation
0	0	Reset
1	1	Set

(d) *T* Flip-Flop

T	$Q(t+1)$	Operation
0	$Q(t)$	No change
1	$\overline{Q}(t)$	Complement

Table 4-1 Flip-Flop Characteristic Tables

Flip-Flops: Few Terminology

- Setup time: the time for which input must be maintained in an edge-triggered flip-flop prior to the occurrence of clock transition.
- Hold time: time for which the input must not change after the application of positive transition of the pulse.
- Propagation delay time: the interval between the trigger edge and the stabilization of the output to the new state.
- Clock gating: technique that disables clock pulses used to design low power circuits.
- Clock skew: the delay introduced due to clock gating circuit (usually, a multiplexer) so that gated clock FFs and non-gated clock FFs change at different time. May lead to unreliable circuit operations.