Chapter 5: Sequential Circuits, Counters and Registers

Section 5-1- Introduction

Section 5-2 Flip-Flops

Section 5-3 Triggering of Flip-Flops

Section 5-4 Analysis of Clocked Sequential Circuits

Section 5-5 State Reduction and Assignment

Section 5-6 Flip-Flop Excitation Tables

Section 5-7 Design Procedure

Section 5-8 Design of Counters

Section 5-9 Registers

Section 5-10 Shift Registers

Section 5-11 Ripple Counters

Section 5-12 Synchronous Counters

Section 5-13 Asynchronous Counters

Section 5-14 Timing Sequences

Section 5-1: Introduction

•The digital circuits, like the half adder and the full adder, considered so far have been combinational, meaning the outputs at any instant of time are entirely dependent upon the inputs present at that time.

•Although every digital system is likely to have combinational circuits, most systems encountered in practice also include memory elements, which require that the system be described in terms of **sequential logic**.

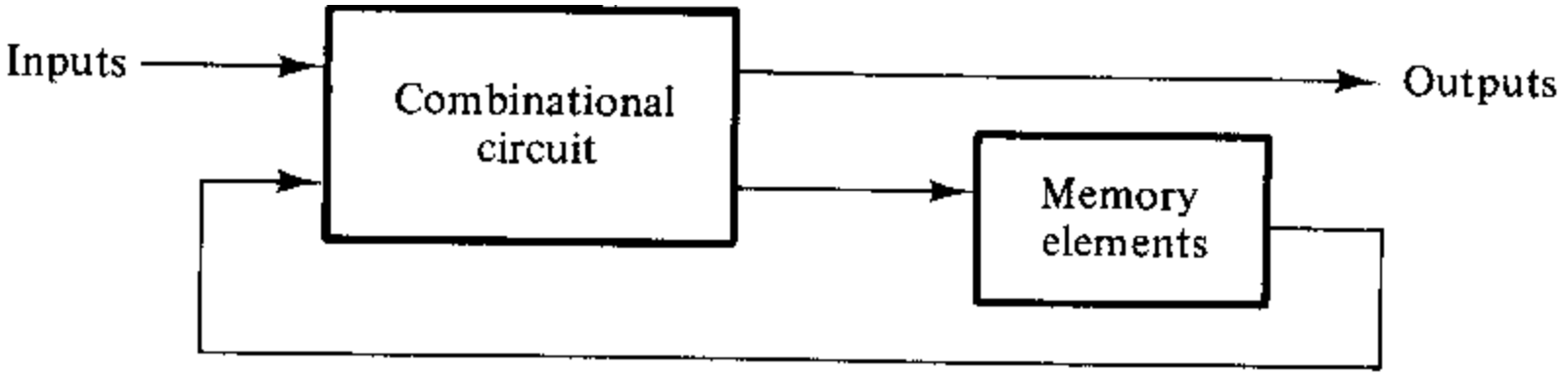


FIGURE 6-1
Block diagram of a sequential circuit

•Sequential circuits consist of a combinational circuit to which memory elements are connected to form a feedback path.

•The memory elements are devices capable of storing binary information within them.

•The binary information stored in the memory elements at any given time defines the state of the sequential circuit.

•The sequential circuit receives binary information from external inputs.

•These inputs, together with the present state of the memory elements, determine the binary value at the output terminals.

•They also determine the condition for changing the state in the memory elements.

•The block diagram demonstrates that the external outputs in a sequential circuit are a function not only of external inputs, but also of the present state of the memory elements.

•The next state of the memory elements is also a function of external inputs and the present state.

•Thus, a sequential circuit is specified by a time sequence of inputs, outputs, and internal states.

•There are two main types of sequential circuits. Their classification depends on the timing of their signals.

•A synchronous sequential circuit is a system whose behavior can be defined from the knowledge of its signals at discrete instants of time.

•The behavior of an asynchronous sequential circuit depends upon the order in which its input signals change and can be affected at any instant of time.

•A synchronous sequential logic system must employ signals that affect the memory of elements only at discrete instants of time.

•One way of achieving this goal is to use pulses of limited duration throughout the system so that one pulse amplitude represents logic-1 and another pulse amplitude (or the absence of a pulse) represents logic-0.

•Practical synchronous sequential logic systems used fixed amplitudes such as voltage levels for the binary signals.

•Synchronization is achieved by a timing device called *master-clock generator,* which generates a periodic train of *clock pulses.*

•The clock pulses are distributed throughout the system in such a way that memory elements are affected only with the arrival of the synchronization pulse.

•In practice, the clock pulses are applied into AND gates together with the signals that specify the required change in memory elements.

•The AND-gate outputs can transmit signals only at instants that coincide with the arrival of clock pulses.

•Synchronous sequential circuits that use clock pulses in the inputs of memory elements are called clocked sequential circuits.

•The memory elements used in clocked sequential circuits are called *flip-flops.*
•These circuits are binary cells capable of storing one bit of information
.
•A flip-flop circuit has two outputs, one for the normal value and one for the complement value of the bit stored in it.

•Binary information can enter a flip-flop in a variety of ways, in fact that gives rise to different types of flip-flops.

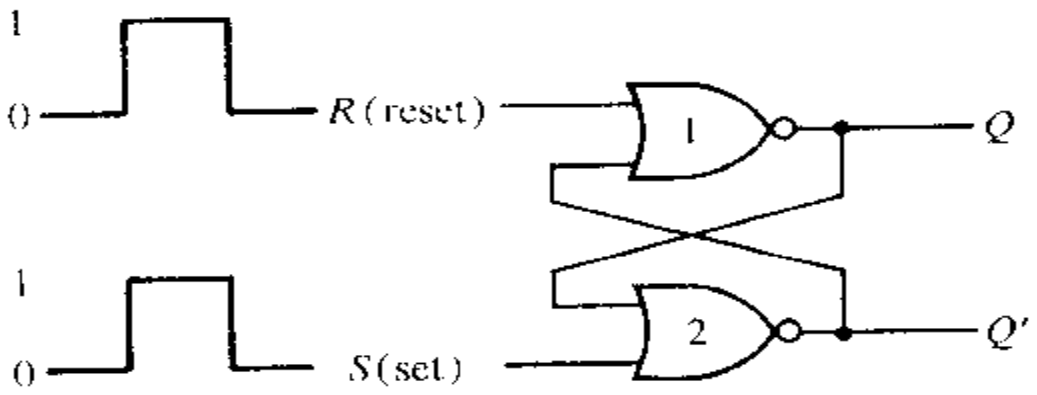
Section 5-2-Flip-Flops

•A flip-flop circuit can maintain a binary state indefinitely (as long as power is delivered to the circuit) until directed by an input signal to switch states.

•The major differences among various types of flip-flops are in the number of inputs they possess and in the manner in which the inputs affect the binary state.

Basic Flip-Flop Circuit

•A flip flop circuit can be constructed from two NAND gates or two NOR gates. These constructions are shown in the logic diagram of Figs. 6-2 and 6-3.

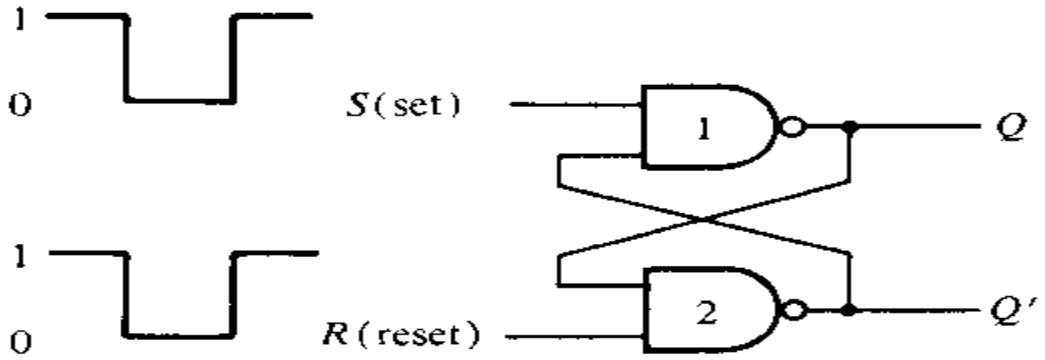


(a) Logic diagram

S	R	Q	Q'	
1	0	1	0	
0	0	1	0	(after S = 1, R = 0)
0	1	0	1	
0	0	0	1	(after S = 0, R = 1)
1	1	0	0	

(b) Truth table

FIGURE 6-2
Basic flip-flop circuit with NOR gates



(a) Logic diagram

S	R	Q	Q'	
1	0	0	1	
1	1	0	1	(after S = 1, R = 0)
0	1	1	0	
1	1	1	0	(after S = 0, R = 1)
0	0	1	1	

(b) Truth table

FIGURE 6-3
Basic flip-flop circuit with NAND gates

•Each circuit forms a basic flip-flop upon which other more complicated types can be built.
•The cross-coupled connection from the output of one gate to the input of the other gate constitutes a feedback path.

•For this reason, the circuits are classified as asynchronous sequential circuits.

•Each flip-flop has two outputs, Q and Q', and two inputs, *set* and *reset.*

•This type of flip-flop is sometimes called a *direct-coupled* RS flip-flop, or *SR latch.*
•The R and S are the first letters of the two input names.

•To analyze the operation of the circuit of Fig 2-2, we must remember that the output of a NOR gate is 0 if any input is 1, and that the output is 1 only when all inputs are 0.

•As a starting point, assume that the set input is 1 and the reset input is 0.

•Since gate 2 has an input of 1, its output Q' must be 0, which puts both inputs of gate 1 at 0, so that output Q is 1.

•When the set input is returned to 0, the outputs remain the same, because output Q remains at 1, leaving one input of gate 2 at 1.

•That causes output Q' to stay at 0, which leaves both inputs of gate number 1 at 0, so that Q is 1.

•In the same manner, it is possible to show that a 1 in the reset input changes output Q to 0 and Q' to 1.

•When the reset input return to 0, the outputs do not change.
•When a 1 is applied to both the set and reset inputs, both Q and Q' outputs go to 0.

•This condition violates the fact that outputs Q and Q' are the complements of each other.

•In normal operation, this condition must be avoided by making sure that 1's are not applied to both inputs simultaneously.

•A flip-flop has two useful states. When Q=1 and Q'=0, it is in the *set state* (or 1-state).
•When Q=0 and Q'=1, it is in the *clear state* (0-state).

•The outputs Q and Q' are complements of each other and are referred to as normal and complement outputs, respectively.

•The binary state of the flip flop is taken to be the value of the normal output.

•Under normal operation, both inputs remain at 0 unless the state of the flip-flop has to be changed.

•The application of a momentary 1 to the set input causes the flip-flop to go to the set state.

•The set input must go back to 0 before a 1 is applied to the reset input.

•A momentary 1 applied to the reset input causes the flip-flop to go to clear state.

•When both inputs are initially 0, a 1 applied to the set input while the flip-flop is in set state or a 1 applied to the reset input while the flip-flop is in the clear state leaves the outputs unchanged.

•When a 1 is applied to both the set and reset inputs, both outputs go to 0.

•This state is undefined and is usually avoided.

•If both inputs now go to 0, the state of the flip-flop is indeterminate and depends on which input remains a 1 longer before the transition to 0.

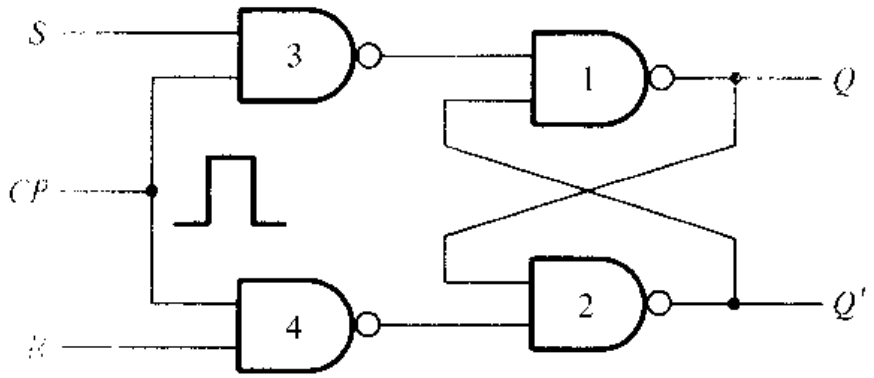
•The NAND basic flip-flop circuit of Fig 2-2 operates with both inputs normally at 1 unless the state of the flip-flop has to be changed.

•The application of a momentary 0 to the set input causes output Q to go to 1 and Q' to go to 0.

RS Flip-flop

- The operation of the basic flip-flop can be modified by providing an additional control input that determines when the state of the circuit is to be changed.

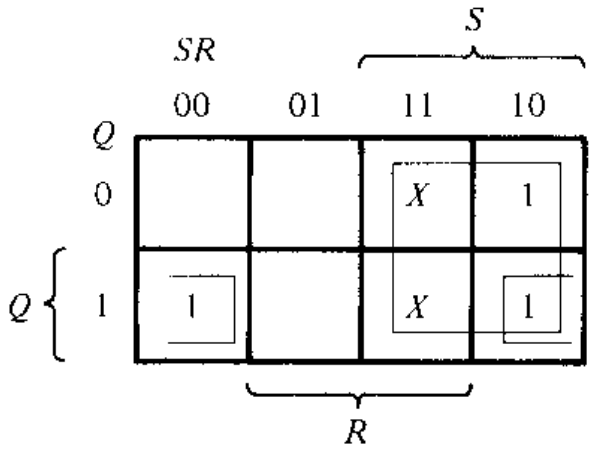
- An RS flip-flop with a clock pulse (CP) input is shown in Fig 6-4(a).



(a) Logic diagram

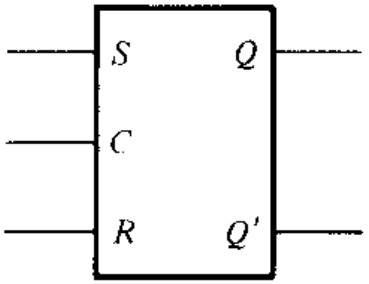
Q	S	R	Q(t + 1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	Indeterminate
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	Indeterminate

(b) Characteristic table



$$Q(t+1) = S + R'Q$$
$$SR = 0$$

(c) Characteristic equation



(d) Graphic symbol

Fig 6-4: RS Flip-Flop

•It consists of a basic flip-flop circuit and two additional NAND gates.

•The pulse input acts as an enable signal for the other two inputs.
•The outputs of NAND gates 3 and 4 stay at logic 1 level as long as CP input remain at 0.

•This is the basic condition for the basic flip-flop.

•When the pulse input goes to 1, information from the S or R input is allowed to reach the output.

•The set state is reached when S=1, R=0 and CP =1. This causes the output of gate 3 to go to 0, the output of gate 4 to remain at 1, and the output of the flip-flop at Q to go to 1.

•To change to the reset sate, the inputs must be S=0, R=1, and CP=1.
•In either case, when CP return to 0, the circuit remains in its previous state.

•When CP =1 and both the S and R inputs are equal to 0, the state of the circuit does not change.

•An indeterminate condition occurs when CP=1 and both S and R are equal to 1.

•This condition places 0's in the outputs of gates 3 and 4 and 1's in both outputs Q and Q'.

•When the CP input goes back to 0 (while S and R are maintained at 1), it is not possible to determine the next state, as it depends on whether the output of gate 3 or gate 4 goes to 1 first.

•This indeterminate condition makes the circuit of Fig 2-4(a) difficult to manage and it is seldom used in practice.

•Nevertheless, it is an important circuit because all other flip-flops are constructed from it.

•The characteristic table of the flip-flop is shown in Fig 2-4(b).

•This table shows the operation of the flip-flop in tabular form.

•Q is an abbreviation of Q(t) and stands for the binary state of the flip-flop before the application of a clock pulse, referred to as the *present state.*

•The S and R columns give the possible values of the inputs, and Q(t +1) is the state of the flip-flop after the application of a single pulse, referred to as the *next state.*

•Note that the CP input is not included in the characteristic table.

•The table must be interpreted as follows: Given the present state of Q and the inputs S and R, the application of a single pulse in the CP input causes the flip-flop to go to the next sate Q(t+1).

•The characteristic equation of the flip flop is derived in the map of Fig. 2-4(c).

•This equation specifies the value of the next state as a function of the present state and the inputs.

•The characteristic equation is an algebraic expression for the binary information of the characteristic table.

•The two indeterminate states are marked with don't-care X's in the map, since they may result in either 1 or 0.

•However, the relation SR=0 must be included as part of the characteristic equation to specify that both S and R cannot equal to 1 simultaneously.
•The graphic symbol of the RS flip-flop is shown in Fig 2-4(d).

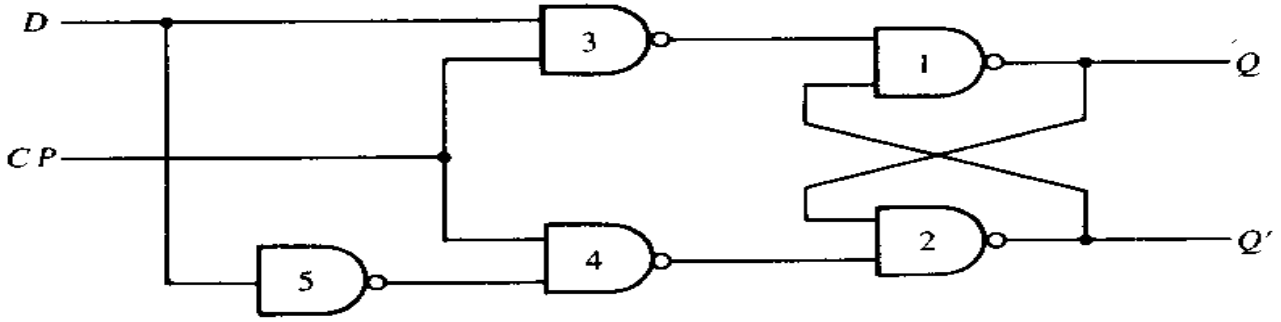
•It consists of a rectangular-shape block with inputs S,R, and C.

•The outputs are Q and Q', where Q' is the complement of Q (except in the indeterminate state).

D Flip-Flop

•One way to eliminate the undesirable condition of the indeterminate state in the RS flip-flop is to ensure that the inputs S and R are never equal to 1 at the same time.

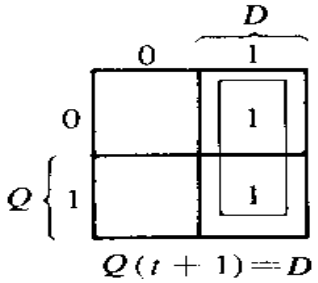
•This is done in the D flip-flop shown in Fig 2-5(a).



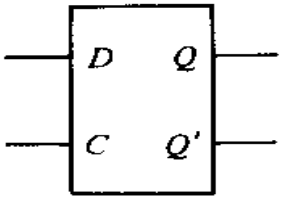
(a) Logic diagram

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

(b) Characteristic table



(c) Characteristic equation



(d) Graphic symbol

FIGURE 6-5
D flip-flop

•The D flip-flop has only two inputs: D and CP.

•The D input goes directly to the S input and its complement is applied to the R input.

•As long as the pulse input is at 0, the outputs of gates 3 and 4 are at the 1 level and the circuit cannot change sate regardless of the value of D.

•The D input is sampled when CP=1, the Q output goes to 1, placing the circuit in the set sate. If D is 0, the output Q goes to 0 and the circuit switches to the clear state.

•The D flip-flop receives the designation from its ability to hold data into its internal storage.
•This type of flip-flop is sometimes called a D-latch gate.

•The CP input is often given the designation G(for gate) to indicate that this input enables the gated latch to make possible data entry into the circuit.

•The binary information present at the data input of the D flip-flop is transferred to the Q output when the CP is enabled.

•The output follows the data input as long as the pulse remains in its 1 state.

•When the pulse goes to 0, the binary information that was present at the data input at the time the pulse transition occurred is retained at the Q output until the pulse is enabled again.

•The characteristic table for the D flip-flop is shown in Fig 2-5(b).

•It shows that the next sate of the flip-flop is independent of the present state since Q(t+1) is equal to input D whether Q is equal to 0 or 1.

•This means that an input pulse will transfer the value of the input D into the output of the flip-flop independent of the value of the output before the pulse is applied.

•The characteristic equation shows clearly that Q(t+1) is equal to D.

•The graphic symbol for the D flip-flop is shown in Fig 6-5(d).

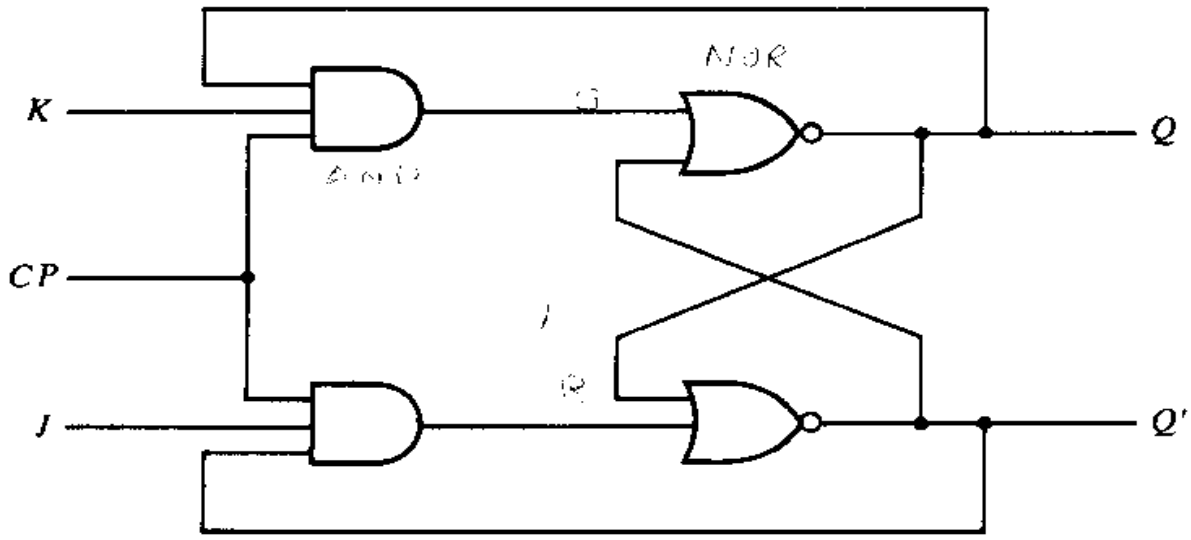
JK and T Flip-flops

•A JK flip-flop is a refinement of the RS flip-flop in that the indeterminate state of the RS type is defined in the JK type.

•Inputs J and K behave like inputs S and R to set and clear the flip-flop, respectively.

•The input marked J is for *set* and the input marked K is for *reset*.
•When both inputs J and K are equal to1, the flip-flop switches to its complement state, that is if Q =1, it switches to Q =0, and vice versa.

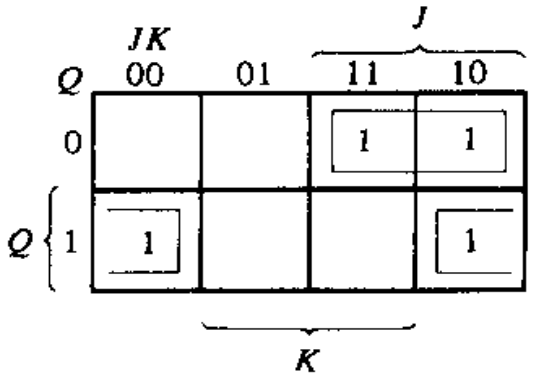
•A JK flip-flop constructed with two cross-coupled NOR gates and two AND gates is shown in Fig 6-6(a).



(a) Logic diagram

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

(b) Characteristic table



$$Q(t+1) = JQ' + K'Q$$

(c) Characteristic equation

•Output Q is ANDed with K and CP inputs so that the flip-flop is cleared during a clock pulse only if Q was previously 1.

•Similarly, output Q' is ANDed with J and CP inputs so that the flip-flop is set with clock pulse only when Q' was previously 1.

•When both J and K are 1, the input pulse is transmitted through one AND gate only: the one whose input is connected to the flip-flop output that is presently equal to 1.

•Thus, if Q=1, the output of the upper AND gate becomes 1 upon application of the clock pulse, and the flip flop is cleared.

•If Q'=1, the output of the lower AND gate becomes 1 and flip-flop is set.

•In either case, the output sate of the flip-flop is complemented.

•The behavior of the JK flip-flop is demonstrated in the characteristic table of Fig 6-6(b).

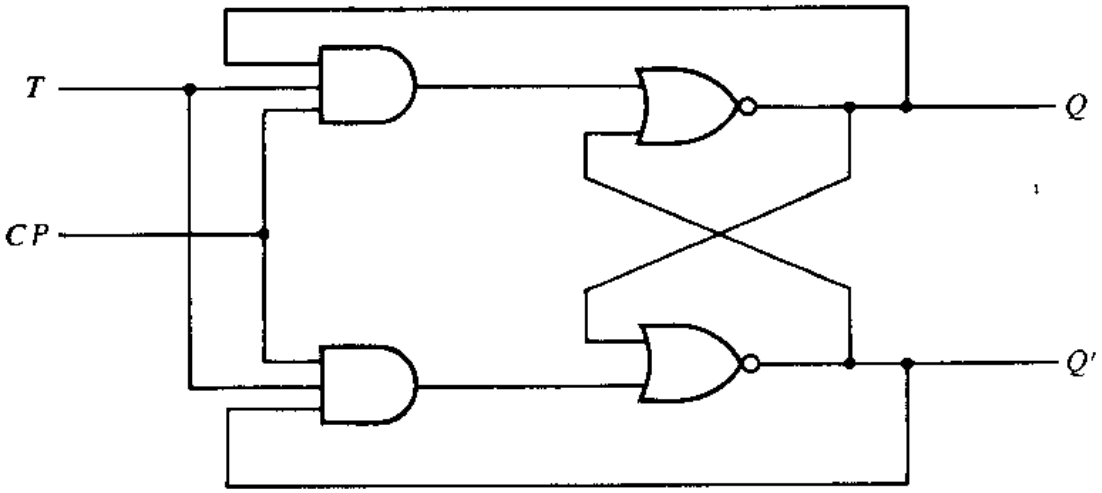
•It is very important to realize that because of the feedback connection in the JK flip-flop, a CP pulse that remains in the 1 state while both J and K are equal to 1 will cause the output to complement again and repeat complementing until the pulse goes back to 0.

•To avoid this undesirable operation, the clock pulse must have a time duration that is shorter than the propagation delay time of the flip-flop.

•This is a restrictive requirement, since the operation of the circuit depends on the width of the pulse.

•For this reason, JK flip-flops are never constructed as shown in Fig. 2-6(a).
•The restriction on the pulse width can be eliminated with an edge-triggered construction, as discussed in next section.

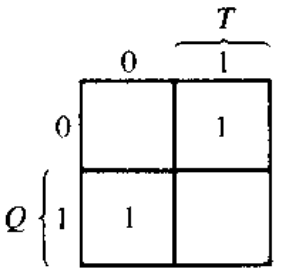
The T flip-flop is a single-input version of the JK flip-flop.



(a) Logic diagram

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

(b) Characteristic table



$Q(t+1) = TQ' + T'Q$

(c) Characteristic equation

FIGURE 6-7
T flip-flop

•As shown in Fig 2-7(a), the T flip-flop is obtained from the JK flip-flop when both inputs are tied together.

•The designation T comes from the ability of the flip-flop to "toggle", or complement, its state.
•Regardless of the present state, the flip-flop complements its output when the clock pulse occurs while input T is 1.

•The characteristic equation shows when T=0, Q(t+1)=Q.

•Hence, the next state is the same as the present state and no change occurs.
•When T=1, then Q(t +1)=Q', and the state of the flip-flop is complemented.

•5.2 Triggering of Flip Flops

The state of a flip-flop is switched by a momentary change in the input signal.

This momentary change is called a *trigger* and the transition it causes is said to trigger the flip-flop.

Asynchronous flip-flops, such as the basic circuits of Figs. 5-2 and 5-3, require an input trigger defined by a change of signal level.
This level must be returned to its initial value (0 in the NOR and 1 in the NAND flip-flop) before a second trigger is applied.

Clocked flip-flops are triggered by pulses.

A pulse starts from an initial value of 0, goes momentarily to 1, and after a short time, return to its initial 0 value.

•The time interval from the application of the pulse until the output transition occurs is a critical factor that needs further investigation.

•As seen from the block diagram of 5-1, a sequential circuit has a feedback path between the combinational circuit and the memory elements.

•This path can produce instability if the outputs of memory elements (flip-flops) are changing while the outputs of the combinational circuit that go to flip-flop inputs are being sampled by the clock pulse.

•This timing problem can be prevented if the outputs of flip-flops do not start changing until the pulse input has returned to 0.

•To ensure such an operation, a flip-flop must have a signal-propagation delay from input to output in excess of the pulse duration.

•This delay is usually very difficult to control if the designer depends entirely on the propagation delay of logic gates.

•One way of ensuring the proper delay is to include within the flip-flop circuit a physical delay unit having a delay equal to or greater than the pulse duration.

•A better way to solve the feedback timing problem is to make the flip-flop sensitive to the pulse transition rather than the pulse duration.

•A clock pulse may be either positive or negative.

•A positive clock source remains at 0 during the interval between pulses and goes to 1 during the occurrence of a pulse.

•The pulse goes through two signal transitions: from 0 to 1 and return from 1 to 0.

•As shown in Fig. 5-8, the positive transition is defined as the *positive edge* and the negative transition as the *negative edge*.

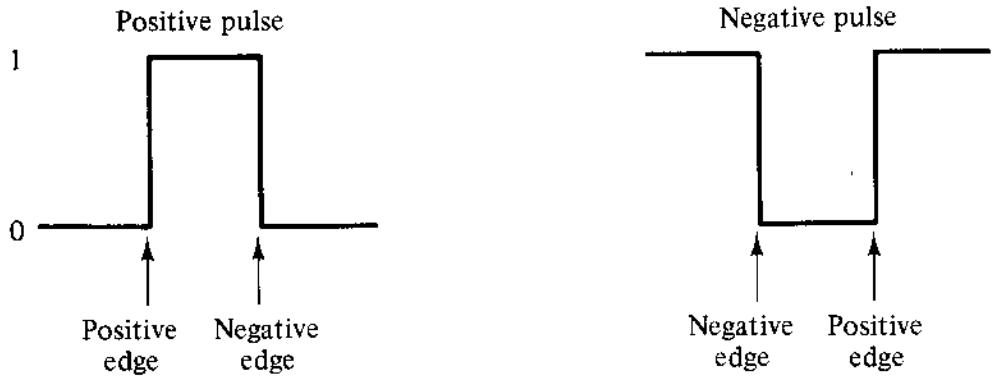


Fig 5-8 Definition of clock-pulse transition

- The clocked flip-flops introduced in Section 5-1 are triggered during the positive edge of the pulse, and the state transition starts as soon as the pulse reaches the logic-1 level.

- The new state of the flip-flop may appear at the output terminals while the input pulse is still 1.

- If the other inputs of the flip-flop change while the clock is still 1, the flip-flop will start responding to these new values and a new output state may occur.

•When this happens, the output of one flip-flop cannot be applied to the inputs of another flip-flop when both are triggered by the same clock pulse.

•However, if we can make the flip-flop respond to the positive-(or negative-) edge transition only, instead of the entire pulse duration, then the multiple-transition problem can be eliminated.

One way to make the flip-flop respond only to a pulse transition is to use capacitive coupling.

In this configuration, an RC (resistor-capacitor) circuit is inserted in the clock input of the flip-flop.
This circuit generates a spike in response to a momentary change of input signal.

•A positive edge emerges from such a circuit with a positive spike, and a negative edge emerges with a negative spike.

•Edge triggering is achieved by designing the flip-flop to neglect one spike and trigger on the occurrence of the other spike.

•Another way to achieve edge triggering is to use a master-slave or edge triggered flip-flop as discussed below.

•**Master-Slave Flip-Flop**

•A master-slave flip-flop is constructed from two separate flip-flops.

•Once circuit serves as a master and the other as a slave, and the overall circuit is referred to as a *master-slave flip-flop.*

•The logic diagram of an RS master-slave flip-flop is shown in Fig 5-9.

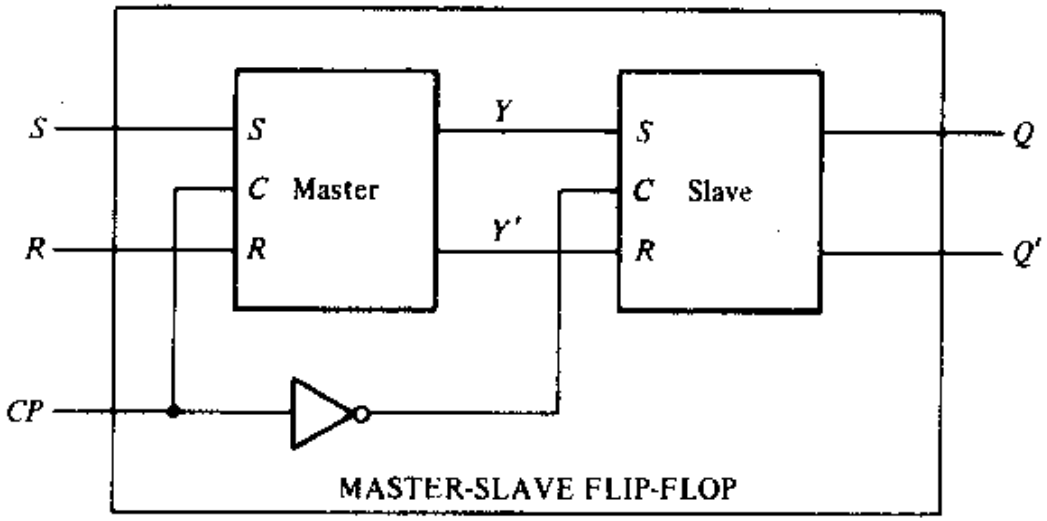


Fig 5-9 Logic Diagram of a master-slave flip-flop

It consists of a master flip-flop, a slave flip-flop, and an inverter.

When clock pulse CP is 0, the output of the inverter is 1.

Since the clock input of the slave is 1, the flip-flop is enabled and output Q is equal to Y, while Q' is equal to Y.

The master flip-flop is disabled because CP=0.

When the pulse becomes 1, the information then at the external R and S inputs is transmitted to the master flip-flop.

The slave flip-flop, however, is isolated as long as the pulse is at its 1 level, because the output of the inverter is 0.

When the pulse return to 0, the master flip-flop is isolated, which prevents the external inputs from affecting it.

•The slave flip-flop then goes to the same state as the master flip-flop.
•The timing relationship shown in Fig 5-10 illustrate the sequence of events that occur in a master-slave flip-flop.

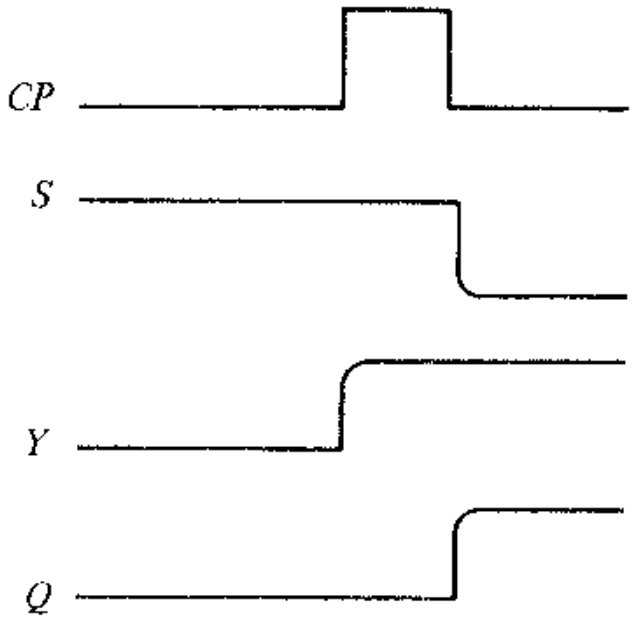


Figure 5-10 Timing relationship in a master-slave flip-flop

•Assume that the flip-flop is in the clear state prior to the occurrence of a pulse, so that Y=0 and Q=0.

•The input conditions are S=1, R=0, and the next clock pulse should change the flip-flop to the set state with Q=1.

•During the pulse transition from 0 to 1, the master flip-flop is set and changes Y to 1.

•The slave flip-flop is not affected because its CP input is 0.

•Since the master flip-flop is an internal circuit, its change of state is not noticeable in the outputs Q and Q'.

•When the pulse returns to 0, the information from the master is allowed to pass through to the slave, making the external output Q=1.

•Once the CP reaches 0, the master is disabled and its R and S inputs have no influence until the next clock pulse occurs.

•Thus, in a master-slave flip-flop, it is possible to switch the output of the flip-flop and its input information with the same clock pulse.

•The behavior of the master-slave flip-flop just described dictates that the sate changes in all flip-flops coincide with the negative-edge transition of the pulse.

•However, some IC master-slave flip-flops change output states in the positive-edge transition of clock pulses.

•This happens in flip-flops that have an inverter between the CP terminal and the input of the master, without the inverter between the CP terminal and the input of the slave.

• Such flip-flops are triggered with negative pulses, so that the negative edge of the pulse affects the master and the positive edge affects the slave and the output terminals.

•**Edge-Triggered Flip-flops**

•Another type of flip-flop that synchronizes the state changes during a clock-pulse transition is the *edge-triggered* flip-flop.

•In this type of flip-flop, output transitions occur at a specific level of the clock pulse.

•When the pulse input level exceeds this threshold level, the inputs are locked out and the flip-flop is therefore unresponsive to further changes in inputs until the clock pulse returns to 0 and another pulse occurs.

•Some edge-triggered flip-flops cause a transition on the positive edge of the pulse, and others cause a transition on the negative edge of the pulse.

•The logic diagram of a D-type positive-edge-triggered flip-flop is shown in Fig 5-12.

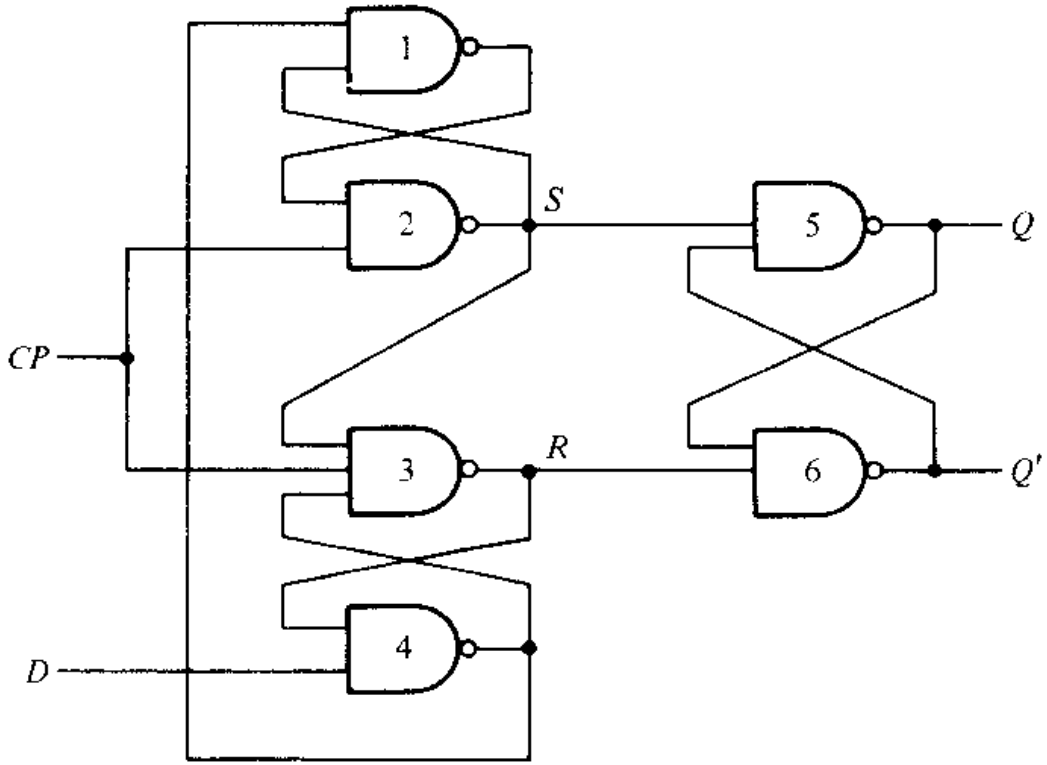


Fig 5-12 D-type positive edge-triggered flip-flop

•It consists of three basic RS flip-flops. NAND gates 1 and 2 make up one basic flip-flop and gates 3 and 4 another.

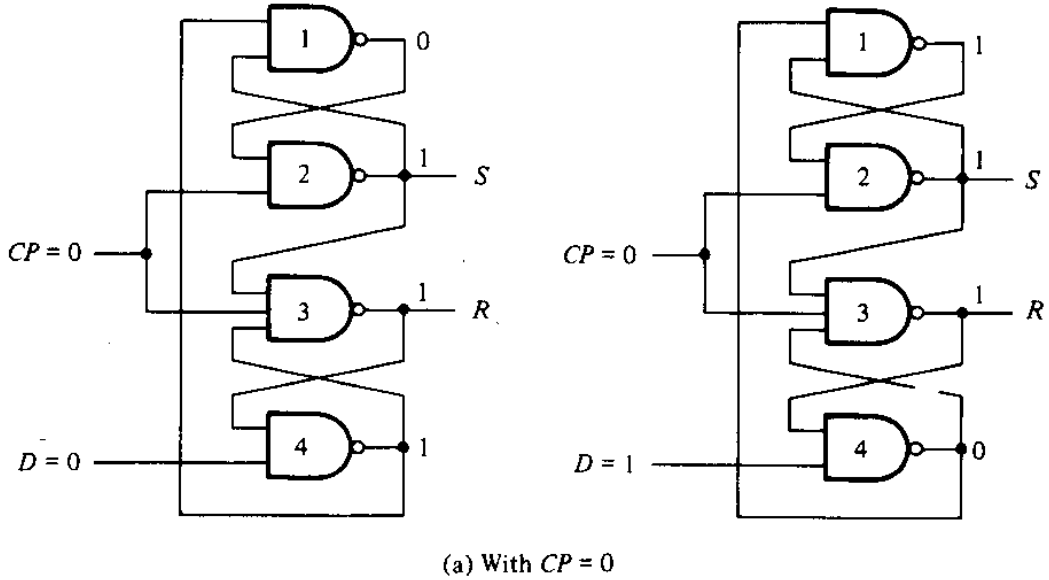
•The third basic flip-flop comprising gates 5 and 6 provides the outputs to the circuits.

•Inputs S and R of the third basic flip-flop must be maintained at logic-1 for the outputs to remain in their steady state values.-
•When S=0 and R=1, the output goes to the set state with Q=1.

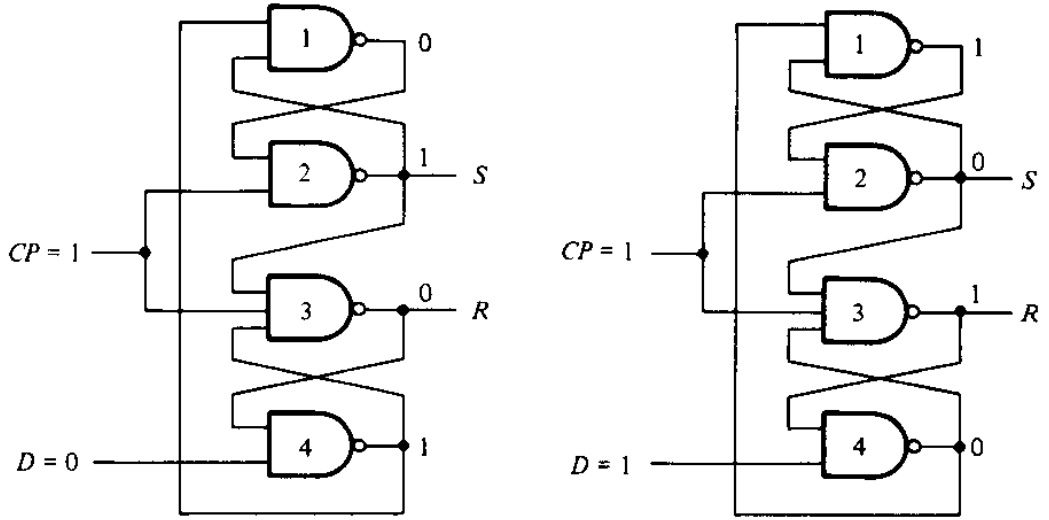
•When S=1 and R=0, the output goes to the clear state with Q=0.

•Inputs S and R are determined from the states of the other two basic flip-flops.

•These two basic flip-flops respond to the external inputs D(data) and CP(clock pulse).
•The operation of the circuit is explained in Fig 5-13, where gates 1-4 are redrawn to show all possible transitions.



(a) With $CP = 0$



(b) With $CP = 1$

Figure 5-13 Operation of D-type edge triggered flip-flop

•Outputs S and R from gates 2 and 3 go to gates 5 and 6, as shown in Fig 5-12, to provide the actual outputs of the flip-flop.

•Figure 5-13(a) shows the binary values at the outputs of the four gates when CP=0.
•Input D may be equal to 0 or 1. In either case, a CP of 0 causes the outputs of gates 2 and 3 to go to 1, thus making S=R=1, which is the condition for a steady-state output.

•When D=0, gate 4 has a 1 output, which causes the output of gate 1 to go to 0.

•When D=1, gate 4 goes to 0, which causes the output of gate 1 to go to 1.

•These are the two possible conditions when CP terminal, being 0, disables any changes at the outputs of the flip-flop, no matter what the value of D happens to be.

•There is a definite time, called the *setup* time, in which the D input must be maintained at a constant value prior to the application of the pulse.

•The setup time is equal to the propagation delay through gates 4 and 1 since change in D causes a change in the outputs of these two gates.

•Assume now that D does not change during the setup time and that input CP become 1. This situation is depicted in Fig 5-13(b).

•If D=0 when CP becomes 1, then S remains 1 but R changes to 0.

•This causes the output of the flip-flop Q to go to 0 (in Fig 5-12).

•If now, while CP=1, there is a change in the D-input, the output of gate 4 will remain at 1 (even if D goes to 1), since one of the gate inputs comes from R, which is maintained at 0.

•Only when CP returns to 0 can the output of gate 4 change; but then both R and S become 1, disabling any changes in the output of the flip-flop.

•However, there is a definite time, called the hold time, that D-input must not change after the application of the positive-going transition of the pulse.

•The hold time is equal to the propagation delay of gate 3, since it must be ensured that R become 0 in order to maintain the output of gate 4 at 1, regardless of the value of D.

•If D=1 when CP=1, then S changes to 0, but R remains at 1, which causes the output of the flip-flop Q to go to 1.

•A change in D while CP=1 does not alter S and R, because gate 1 is maintained at 1 by the 0 signal from S.

•When CP goes to zero, both S and R got to 1 to prevent the output from undergoing any changes.

•In summary, when the input clock pulse makes a positive-going transition, the value of D is transferred to Q.

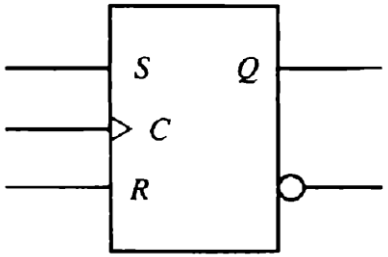
•Changes in D when CP is maintained at a steady 1 value do not affect Q.

• Moreover, a negative pulse transition does not affect the output, nor does it when CP=0.

•Hence, the edge-triggered flip-flop eliminates any feedback problems in sequential circuits just as a master-slave flip-flop does.

•The setup time and hold time must be taken into consideration when using this type of flip-flop.

Graphic Symbols

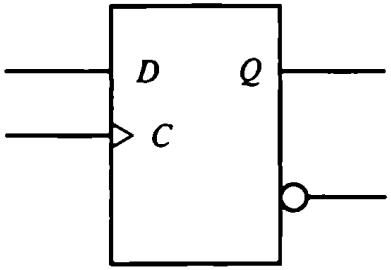


(a) Graphic symbol

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

(b) Characteristic table

Figure 5-14 SR flip-flop

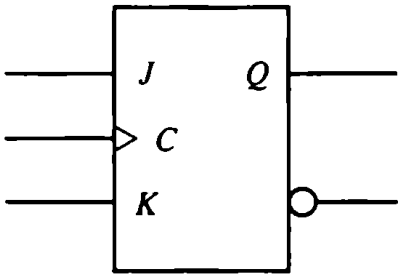


(a) Graphic symbol

D	Q (t + 1)	
0	0	Clear to 0
1	1	Set to 1

(b) Characteristic table

Figure 5-15 D flip-flop

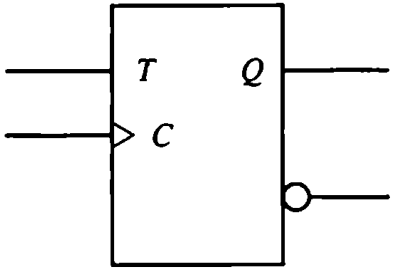


(a) Graphic symbol

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

(b) Characteristic table

Figure 5-16 JK flip-flop



(a) Graphic symbol

T	Q (t + 1)	
0	Q (t)	No change
1	Q′ (t)	Complement

(b) Characteristic table

Figure 1-22 *T* flip-flop.

Figure 5-17 T flip-flop

•5.3 Excitation Tables

•The characteristic tables of flip-flops specify the next state when the inputs and the present state are known.

•During the design of sequential circuits we usually know the required transition from present state to next state and wish to find the flip-flop input conditions that will cause the required transition.

•For this reason, we need a table that lists the required input combinations for a given change of state.

•Such a table is called a flip-flop excitation table.

•Table 5-1 lists the excitation tables for the four types of 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

Table 5-1 Excitation Table for Four Flip Flops

•Each table consists of two columns, Q(t) and Q(t+1), and a column for each input to show how the required transition is achieved.

•There are four possible transitions from present state Q(t) to next state Q(t +1).

•The required input conditions for each of these transitions are derived from the information available in the characteristic tables.

•The symbol x in the tables represents a don't-care condition; that is, does not matter whether the input to the flip flop is 0 or 1.

•**5-4 Sequential Circuits**

•A sequential circuit is an interconnection of flip-flops and gates.

•The gates by themselves constitute a combinational circuit, but when included with the flip-flops, the overall circuit is called a sequential circuit.

The block diagram of a clocked sequential circuit is show in Fig. 5-18.

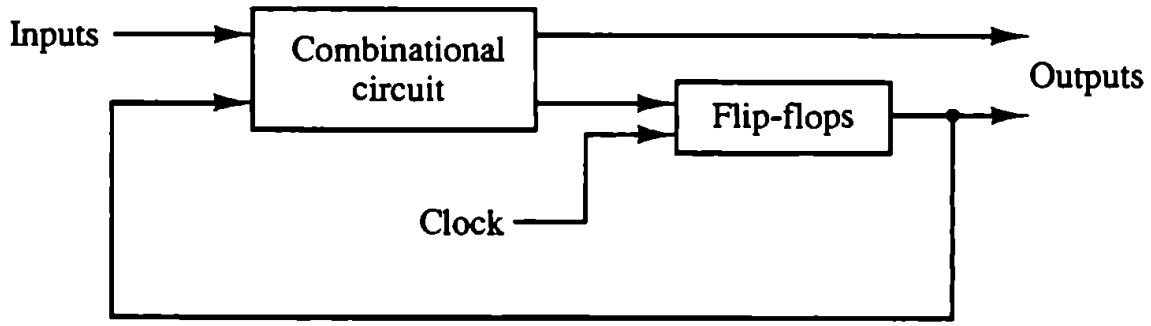


Figure 5-18 Block diagram of a clocked synchronous sequential circuit

- It consists of a combinational circuit and a number of clocked flip-flops.
- In general, any number or type of flip-flops may be included.
- As shown in the diagram, the combinational circuit block receives binary signals from external inputs and from the outputs of flip-flops.

•The outputs of the combinational circuit go to external outputs and to inputs of flip-flops.

•The gates in the combinational circuit determine the binary value to be stored in the flip-flops after each clock transition.

•The outputs of flip-flops, in turn, are applied to the combinational circuit inputs and determine the circuit's behavior.

•This process demonstrates that the external outputs of a sequential circuit are functions of both external inputs and the present state of the flip-flops.

•Moreover, the next state of flip-flops is also a function of their present state and external inputs.

•Thus sequential circuit is specified by a time sequence of external inputs, external outputs and internal flip-flop binary states.

- **Flip-Flop Input Equations**
- An example of a sequential circuit is shown in Fig 5-19.

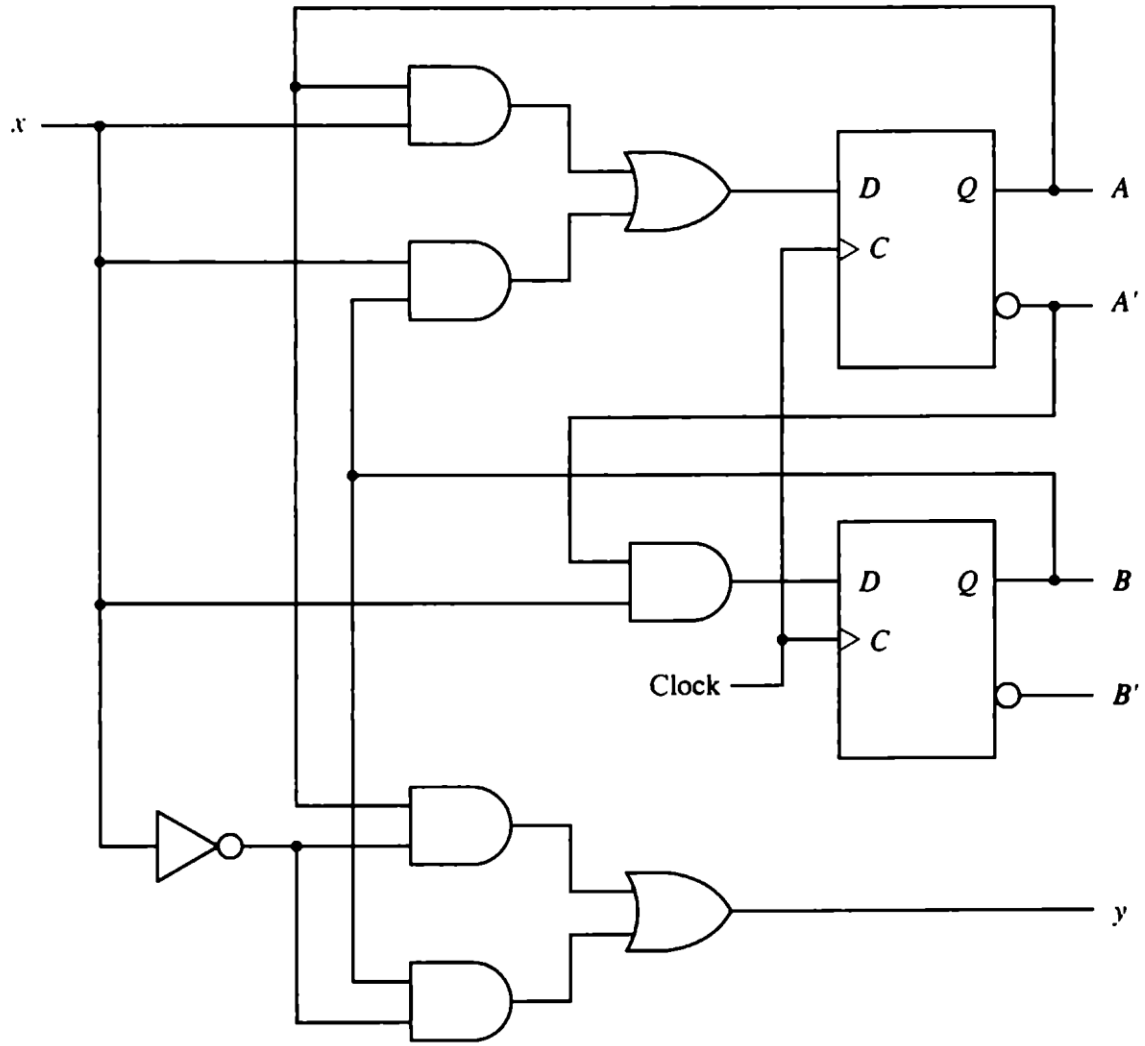


Figure 5-19 Example of a sequential circuit

•It has one input variable x, one output variable y, and two clocked D flip-flops.

•The AND gates, OR gates, and inverter form the combinational logic part of the circuit.

•The interconnections among the gates in the combinational circuit can be specified by a set of Boolean expressions.

•The part of the combinational circuit that generates the inputs to flip-flops are described by a set of Boolean expressions called flip-flop input equations.

•Thus, in Fig 5-19, we have two input equations, designated as D_A and D_B.

•The first letter in each symbol denotes the D input of a D flip-flop.

•The subscript letter is the symbol of the name of flip-flop.

•The input equations are Boolean functions for flip-flop input variables and can be derived by inspection of the circuit.

•Since the output of the OR gate is connected to the D input of flip-flop A, we write the first input equation as

•$D_A = Ax + Bx$

•where A and B are the outputs of the two flip-flops and x is the external input.

•The second input equation is derived from the single AND gate whose output is connected to the D input of flip-flop B:

•$D_B = A'x$

•The sequential circuit also has an external output, which is a function of the input variable and the state of the flip-flops.

•This output can be specified algebraically by the expression

•$y = Ax' + Bx$‘

•From this example we note that a flip-flop input equation is a Boolean expression for a combinational circuit.

•**State Table**

•The behavior of a sequential circuit is determined from the inputs, the outputs, and the state of its flip-flops.

•Both the outputs and the next state are a function of the inputs and the present state.

•A sequential circuit is specified by a state table that relates outputs and next states as a function of inputs and present states.

•In clocked sequential circuits, the transition from present state to next state is activated by the presence of a clock signal.

•The state table for the circuit of Fig 5-19 is shown in Table 5-2.

Present state		Input	Next state		Output
A	B	x	A	B	y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0

Table 5-2 State Table for Circuit of Fig 1-25

•The table consists of four sections, labeled present state, input, next state, and output.

•The present-state section shows the states of flip-flops A and B at any given time t.

•The input section gives a value of x for each possible present state.

•The next state section shows the states of the flip-flops one clock period later at time t+1.

•The output section gives the value of y for each present state and input condition.

•The derivation of a state table consists of first listing all possible binary combinations of present state and inputs.

•In this case we have eight binary combinations from 000 to 111.

•The next-state values are then determined from the logic diagram or from the input equations.
•The input equations for flip-flop A is
•$D_A = Ax + Bx$
•The next-state of each flip-flop is equal to its D input value in the present state.

•The transition from present state to next state occurs after the application of a clock signal.
•Therefore, the next state of A is equal to 1 when the present state and input values satisfy the condition Ax=1 or Bx =1, which makes D equal to 1.

•This is shown in the state table with three 1's under the column for next sate of A.
•Similarly, the input equation for flip-flop B is
•$D_B = A'x$

•The next state of B in the state table is equal to 1 when the present state of A is 0 and input x is equal to 1.

•The output column is derived from the output equation
•y = A'x + Bx‘

•The state table of any sequential circuit is obtained by the procedure used in in this example.

•In general, a sequential circuit with m flip-flops, n input variables, and p output variables will contain m columns for present state, n columns for inputs, m columns for next state, and p columns for outputs.

•The present state and input columns are combined and under them we list 2^{m+n} binary combinations from 0 up to 2^{m+n} -1.

•The next state and output columns are functions of the present state and input values and are derived directly from the circuit or the Boolean equations that describe the circuit.

•**State Diagram**
•The information available in a state table can be represented graphically in a state diagram.

•In this type of diagram, a state is represented by a circle, and the transition between states is indicated by directed lines connecting the circles.

•The state diagram of the sequential circuit of Fig 5-19 is shown in Fig 5-20.

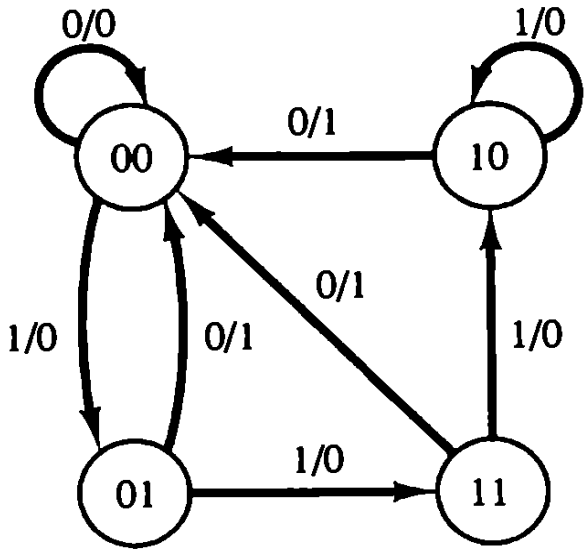


Figure 5-20 State diagram of a sequential circuit

•The state diagram provides the same information as the state table and is obtained directly from Table 5-2.

•The binary number inside each circle identifies the state of the flip-flops.

•The directed lines are labeled with two binary numbers separated by a slash.

•The input value during the present state is labeled first and the number after the slash gives the output during the present state.

•For example, the directed line from state 00 to 01 is labeled 1/0, meaning that when the sequential circuit is in the present state 00 and the input is 1, the output is 0.

•After a clock transition, the circuit goes to the next state 01.

•The same clock transition may change the input value.

•If the input changes to 0, the output becomes 1, but if the input remains at 1, the output stays at 0.

•This information is obtained from the state diagram along the two directed lines emanating from the circle representing state 01.

•A directed line connecting a circle with itself indicates that no change of state occurs.
•There is no difference between a state table and a state diagram except in the manner of representation.

•The state table is easier to derive from a given logic diagram and the state diagram follows directly from the state table.

•The state diagram gives a pictorial view of state transitions and is the form suitable for human interpretation of the circuit operation.

•For example, the state diagram of Fig 5-20 clearly shows that starting form state 00, the output is 0 as long as the input stays at 1.

The first 0 input after a string of 1's gives an output of 1 and transfers the circuit back to the initial state 00.

Design Example

The procedure for designing sequential circuits will be demonstrated by a specific example.

The design procedure consists of first translating the circuit specifications into a state diagram.

The state diagram is then converted into a state table. From the state table we obtain the information for obtaining logic circuit diagram.

We wish to design a clocked sequential circuit that goes through a sequence of repeated binary states 00, 01, 10, and 11 when an external input is equal to 1. The state circuit remains unchanged when x=0.

This type of circuit is called a 2-bit binary counter because the sate sequence is identical to the count sequence of two binary digits.

•Input x is the control variable that specifies when the count should proceed.

•The binary counter needs two flip-flops to represent the two bits.

•The state diagram for the sequential circuit is show in Fig 5-21.

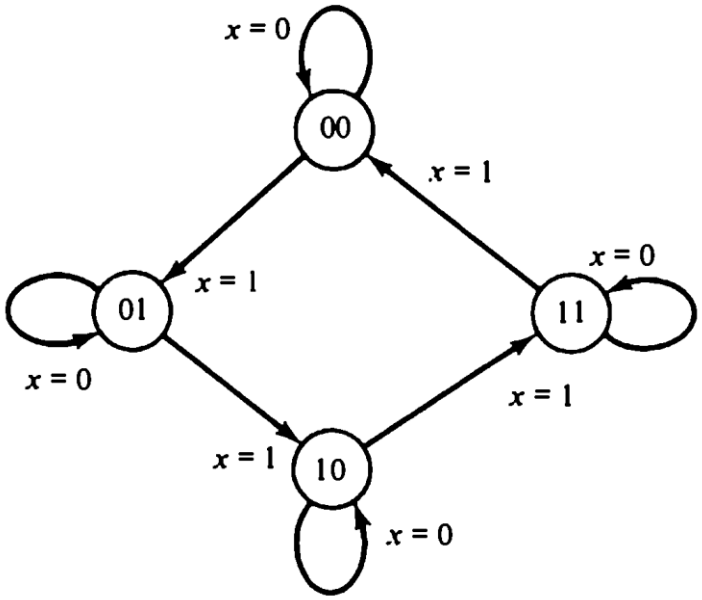


Fig 5-21 State diagram for binary counter

•The diagram is drawn to show that the states of the circuit follow the binary count as long as x=1.

•The state following 11 is 00, which causes the count to be repeated.

•If x=0, the state of the circuit remains unchanged.

•This sequential circuit has no external outputs, and therefore only the input value is labeled in the diagram.

•The state of the flip-flops is considered as the outputs of the counter.

•We have already assigned the symbol x to the input variable.

•We now assign the symbols A and B to the two flip-flop outputs.

The next state of A and B, as a function of the present state and input x, can be transferred from the state diagram into a state table.

The first five column of Table 5-3 constitute the state table. The entries for this table is obtained directly from the state diagram.

Present state		Input	Next state		Flip-flop inputs			
A	B	x	A	B	J_A	K_A	J_B	K_B
0	0	0	0	0	0	×	0	×
0	0	1	0	1	0	×	1	×
0	1	0	0	1	0	×	×	0
0	1	1	1	0	1	×	×	1
1	0	0	1	0	×	0	0	×
1	0	1	1	1	×	0	1	×
1	1	0	1	1	×	0	×	0
1	1	1	0	0	×	1	×	1

Table 5-3 Excitation Table for Binary Counter

- The excitation table of a sequential circuit is an extension of the state table.

- This extension consists of a list of flip-flop input excitations that will cause the required state transitions.

- The flip-flop input conditions are a function of the type of flip-flop used.

- If we employ JK flip flops, we need columns for the J and K inputs of each flip-flop.

- We denote the inputs of flip-flop A by J_A and K_A, and those of flip-flop B by J_B and K_B.

- The excitation table for the JK flip-flop specified in Table 5-1 is now used to derive the excitation table of the sequential circuit.

•Let us now consider the information available in an excitation table such as Table 5-3.

•We already know that a sequential circuit consists of a number of flip-flops and a combinational circuit.

•From the block diagram of Fig 5-18, we note that the outputs of the combinational circuit must go to the four flip-flop inputs J_A, K_A, J_B, and K_B.

•The inputs to the combinational circuit are the external input x and the present-state values of flip-flops A and B.

•Moreover, the Boolean functions that specify a combinational circuit are derived from a truth table that shows the input-output relationship of the circuit.

•The entries that list the combinational circuit inputs are specified under the "present state" and "input columns" in the excitation table.

•Thus an excitation table transforms a state diagram to a truth table needed for the design of the combinational circuit part of the sequential circuit.

•The simplified Boolean functions for the combinational circuit can be derived.

•The inputs are the variables A, B, and x. The outputs are the variables J_A, K_A, J_B, and K_B.

•The information from the excitation table is transferred into the maps of Fig. 5-22, where the four simplified flip-flop input equations are derived:

•$J_A = Bx$ $K_A=Bx$

•$J_B =x$ $K_B=x$

•The logic diagram is drawn in Fig. 5-23 and consists of two JK flip-flop's and an AND gate.

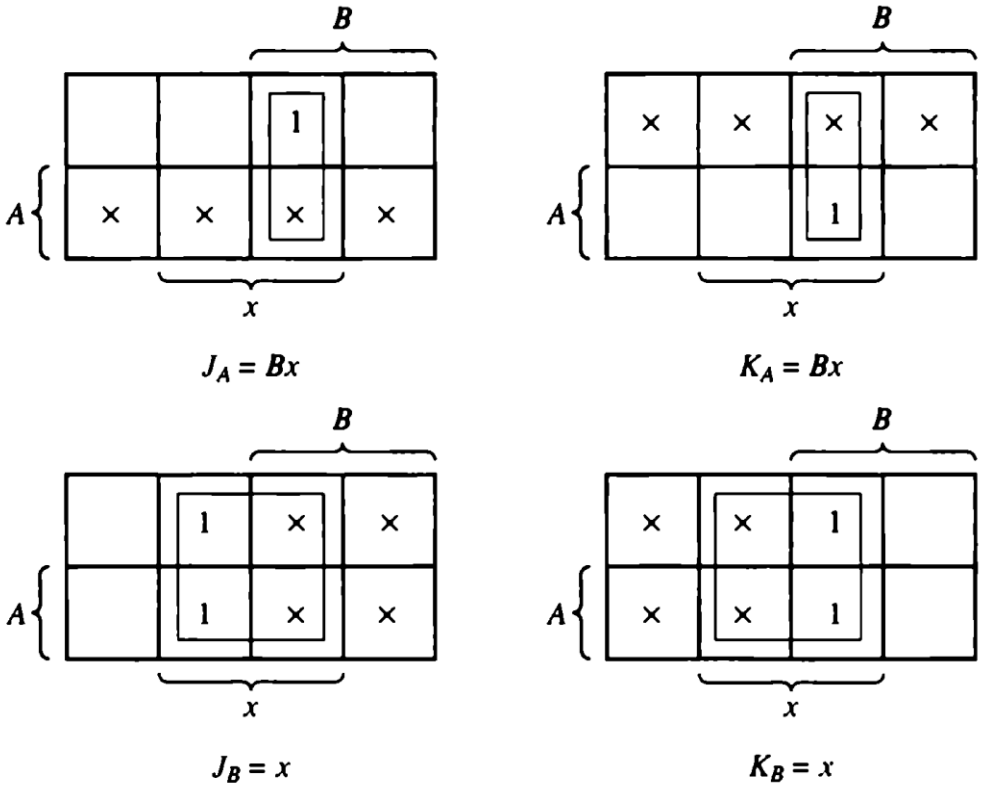


Fig 5-22 Maps for combinational circuit of counter

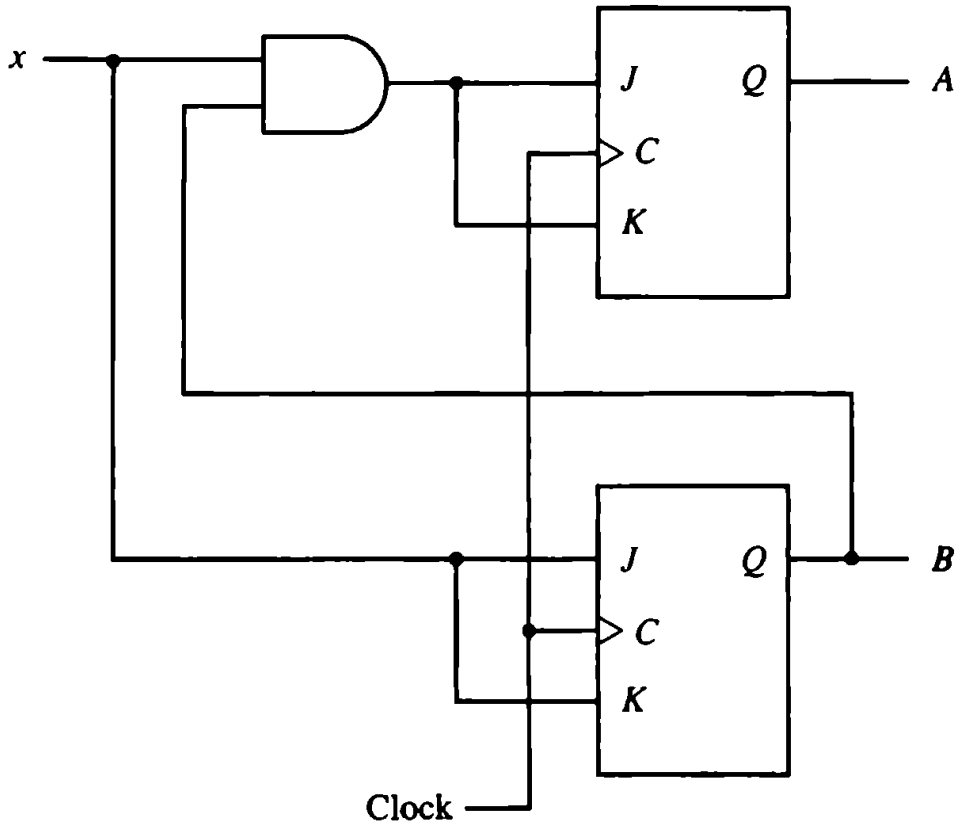


Figure 5-23 Logic diagram of a 2-bit binary counter

•Note that inputs J and K determine the next state of the counter when the clock signal occurs.

•If both J and K are equal to 0, a clock signal will have no effect; that is, the state of the flip-flops will not change.

•Thus when x=0, all four inputs of the flip-flops are equal to 0 and the state of the flip-flops remains unchanged even though clock pulses are applied continuously.

•**5.5 Registers**

•A register is a group of flip-flops with each flip-flop capable of storing one bit of information.

•An n-bit register has a group of n flip-flops and is capable of storing any binary information of n bits.

•A register consists of a group of flip-flops and gates that effect their transition.

•The flip-flops hold the binary information and the gates control when and how new information is transferred into the registers.

•The simplest register is one that consists only of flip-flops, with no external gates. Fig 2-6 shows such a register constructed with four D flip-flops.

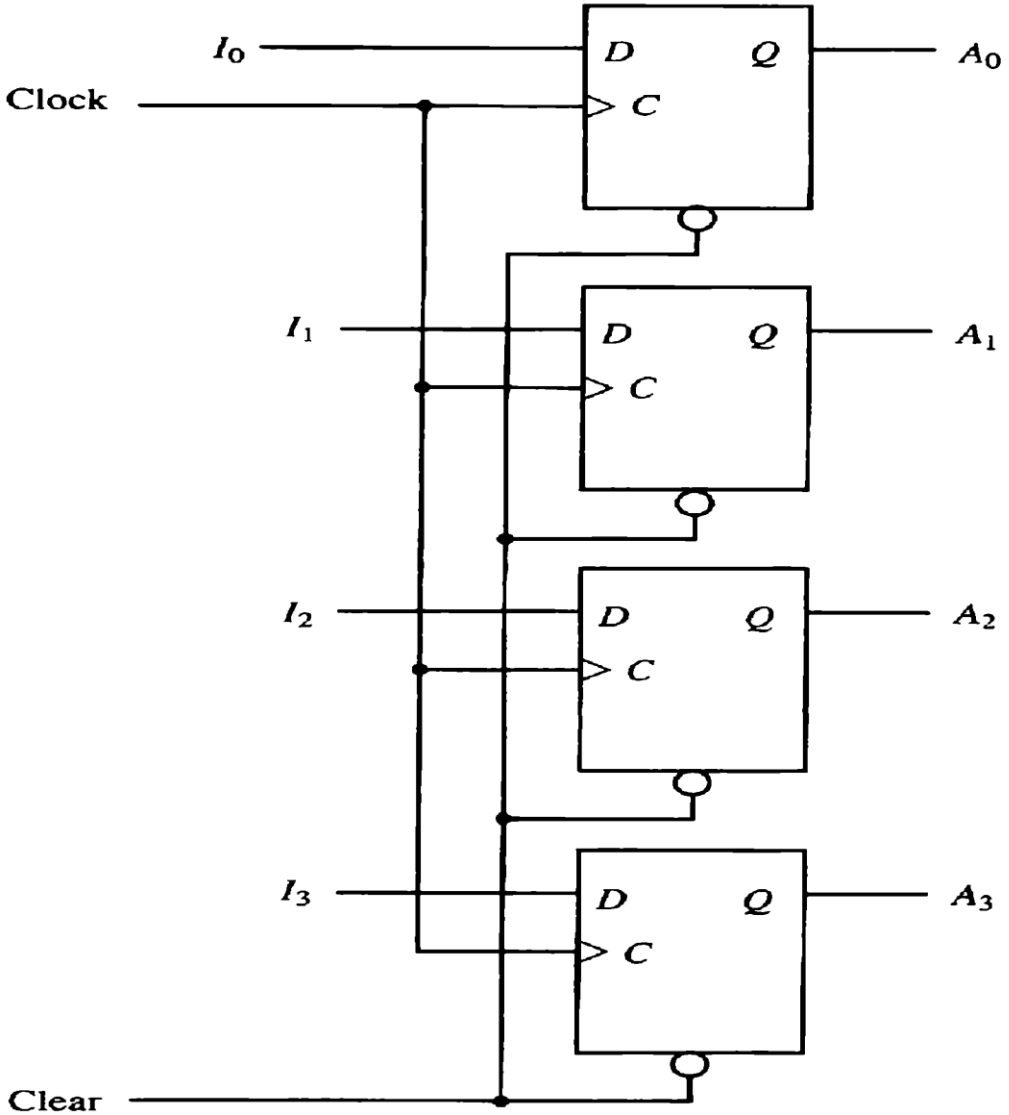


Figure 5-24 4-bit Register

•The clear input must be maintained at logic 1 during normal clocked operation.

•The transfer of new information into a register is referred to as **loading** the register.

•If all the bits of the register are loaded simultaneously with common clock pulse transition, we say that the loading is done in **parallel.**

•A clock transition applied to the C inputs of the register of Fig 2-6 will load all four inputs I_0 through I_3 in parallel.

•In this configuration, the clock must be inhibited from the circuit if the content of the register must be left unchanged.

•5-6 Shift Registers

•A register capable of shifting its binary information in one or both directions is called a shift register.

•The logical configuration of a shift register consists of a chain of flip-flops in cascade, with the output of one flip-flop connected to the input of the next flip-flop.

•All flip-flops receive common clock pulses that initiate the shift from one stage to the next.

•As shown in Fig 2-8, the serial input determines what goes into the leftmost position during the shift.

•The serial output is taken from the output of the rightmost flip-flop.

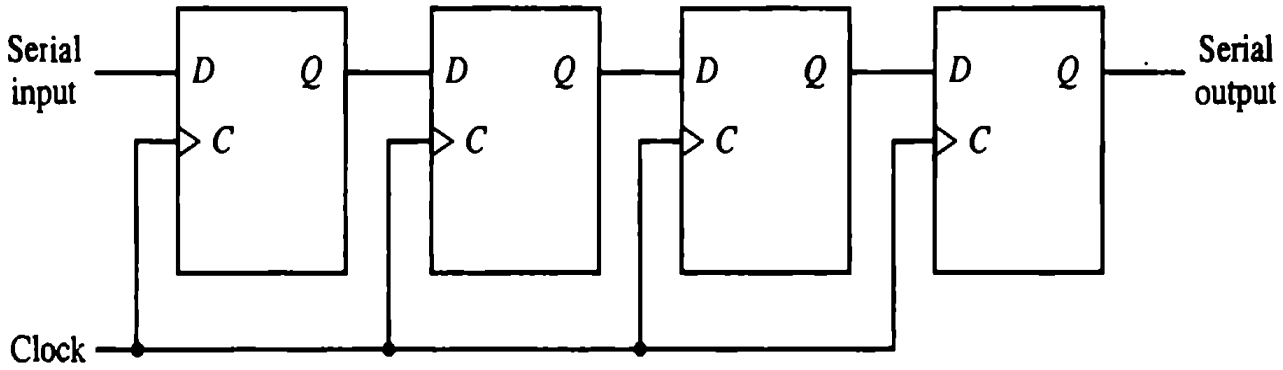


Fig 5-25: 4-bit shift register

•**Bidirectional Shift Register with Parallel Load**

•A register is capable of shifting in one direction only is called a unidirectional shift register.

•A register that can shift in both directions is called a bidirectional shift register.

•Some registers provide the necessary input and output terminals for parallel transfer.

•A 4-bit bidirectional shift register with parallel load is shown in Fig 3-9.

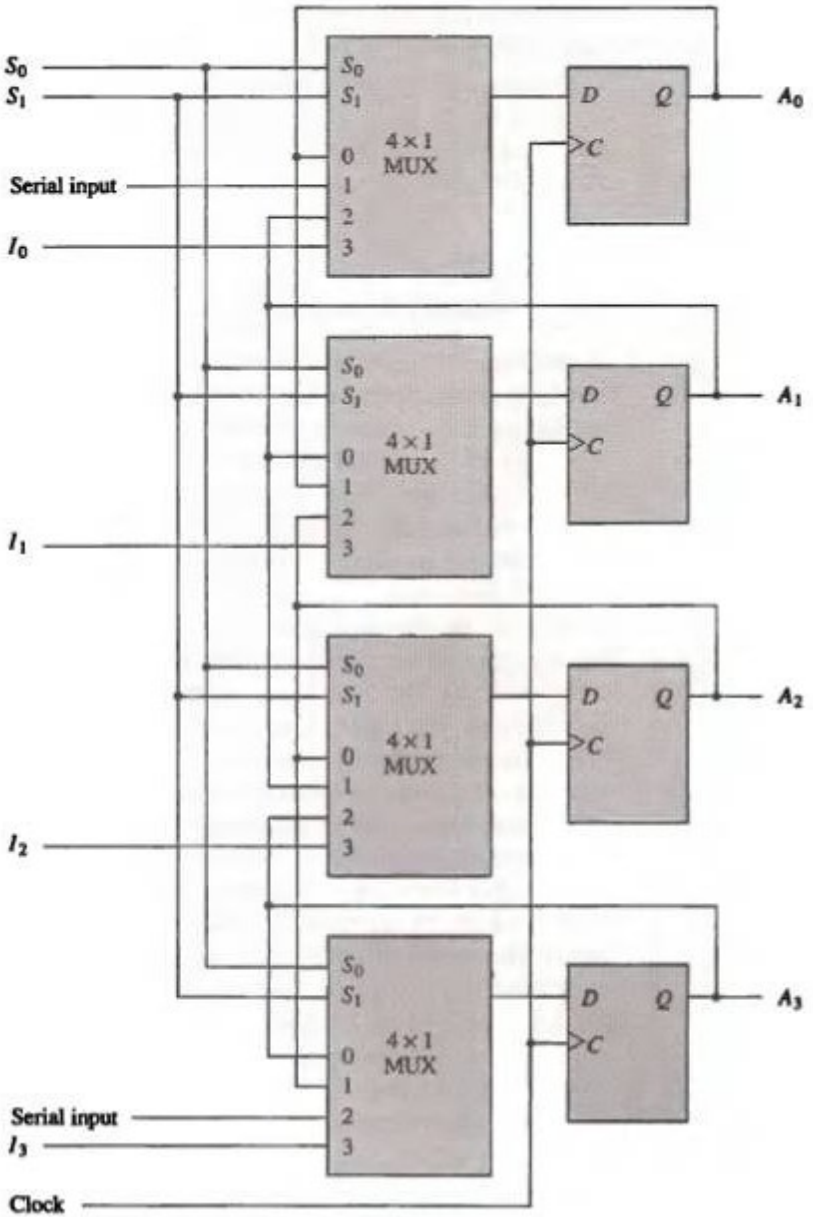


Figure 5-26: Bidirectional shift register with parallel load

•Each stage consists of a D flip-flop and a 4x1 multiplexer.

•The two selection inputs S_1 and S_0 select one of the multiplexer data inputs for the D flip-flop.

•The selection lines control the mode of operation of the register according to function table shown in Table 2-4.

Mode of Control		
S_1	S_0	Register Operation
0	0	No change
0	1	Shift right (down)
1	0	Shift left (up)
1	1	Parallel load

Table 2-4 Function Table for Register of Fig 3-9

•When the mode control S_1S_0=00, data input 0 of each multiplexer is selected.

•This condition forms a path from the output of each flip-flop into the input of the same flip-flop.

•The next clock transition transfers into each flip-flop the binary value it held previously, and no change of state occurs.

•When S_1S_0 = 01, the terminal marked 1 in each multiplexer has a path to the D input of the corresponding flip-flop.

•This causes a shift-right operation, with the serial input data transferred into A_0 and the content of each flip-flop A_{i-1} transferred into flip-flop A_i for i=1, 2, 3.

•When S_1S_0=10 a shift-left operation results, with the other serial input going into flip-flop A_3 and the content of flip-flop A_{i+1} transferred into flip-flop A_i for i=0,1,2.

•When S_1S_0=11, the binary information from each input I_0 through I_3 is transferred into the corresponding flip-flop, resulting in parallel load operation.

•Note the way the diagram is drawn, the shift-right operation shifts the contents of the register in the down direction while the shift left operation causes the contents of the register to shift in the upward direction.

•**5-7 Binary Counters**

•A register that goes through a predetermined sequence of states upon the application of input pulses is called a counter.

•The input pulses may be clock pulses or may originate from external source.

•Counters are found in almost all equipment containing digital logic.

•They are used for counting the number of occurrences of an event and are useful for generating timing signals to control the sequence of operations in digital computers.

•A counter that follows the binary number sequence is called a binary counter.

•An n-bit binary counter is a register of n flip-flops and associated gates that follows a sequence of states according to the binary count of n bits, from 0 to 2^n-1.

•Going through a sequence of binary numbers such as 0000, 0001,0010,0011, 0100 and so on, we note that the lower order bit is complemented after each count and every other bit is complemented from one count to the next if and only if all its lower-order bits are equal to 1.

•For example, the binary count from 0111(7) to 1000(8) is obtained (a) complementing the low-order bit, (b)complementing the second-order bit because the first bit of 0111 is 1, (c)complementing the third-order bit because the first two bits of 0111 are 1's, and (d) complementing the fourth-order bit because the first three bits of 0111 are all 1's.

•A counter circuit will employ flip-flops with complement capabilities.

•A J-K flip-flop can be used for counters since when the inputs to J and K are both equal to 1, the output is complemented and when they are both at 0 the output remains the same.

•In addition, the counter may be controlled with an enable input that turns the counter on or off without removing the clock signal from the flip-flops.

•Synchronous binary counters have a regular pattern, as can be seen from the 4-bit binary counter shown in Fig 2-10.

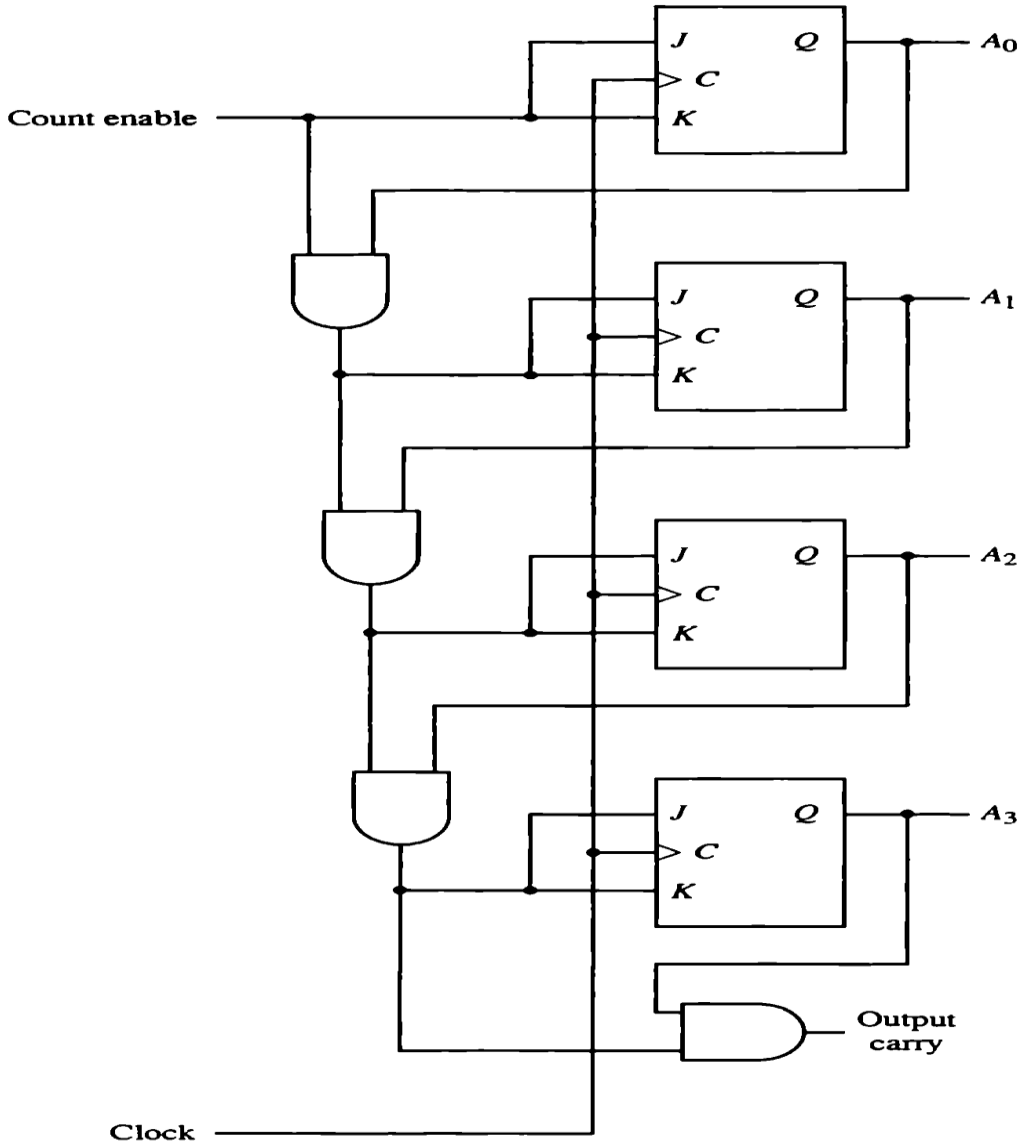


Figure 5- 27 4-bit Synchronous binary counter

•The C inputs of all flip-flops receive the common clock.

•If the count enable is 0, all J and K inputs are maintained at 0 and the output of the counter does not change.

•The first state A_0 is complemented when the counter is enabled and the clock goes through a positive transition.

•Each of the other three flip-flops are complemented when all previous least significant flip-flops are equal to 1 and the count is enabled.

•The chain of AND gates generate the required logic for the J and K inputs.

•The output carry can be used to extend the counter to more stages, with each having an additional flip -flop and an AND gate.

•**Binary Counter with Parallel load**

•Counters employed in digital systems quite often require a parallel load capability for transferring an initial binary number prior to the count operation.

•Fig 2-11 shows the logic diagram of a binary counter that has a parallel load capability and can also be cleared to 0 synchronous with the clock.

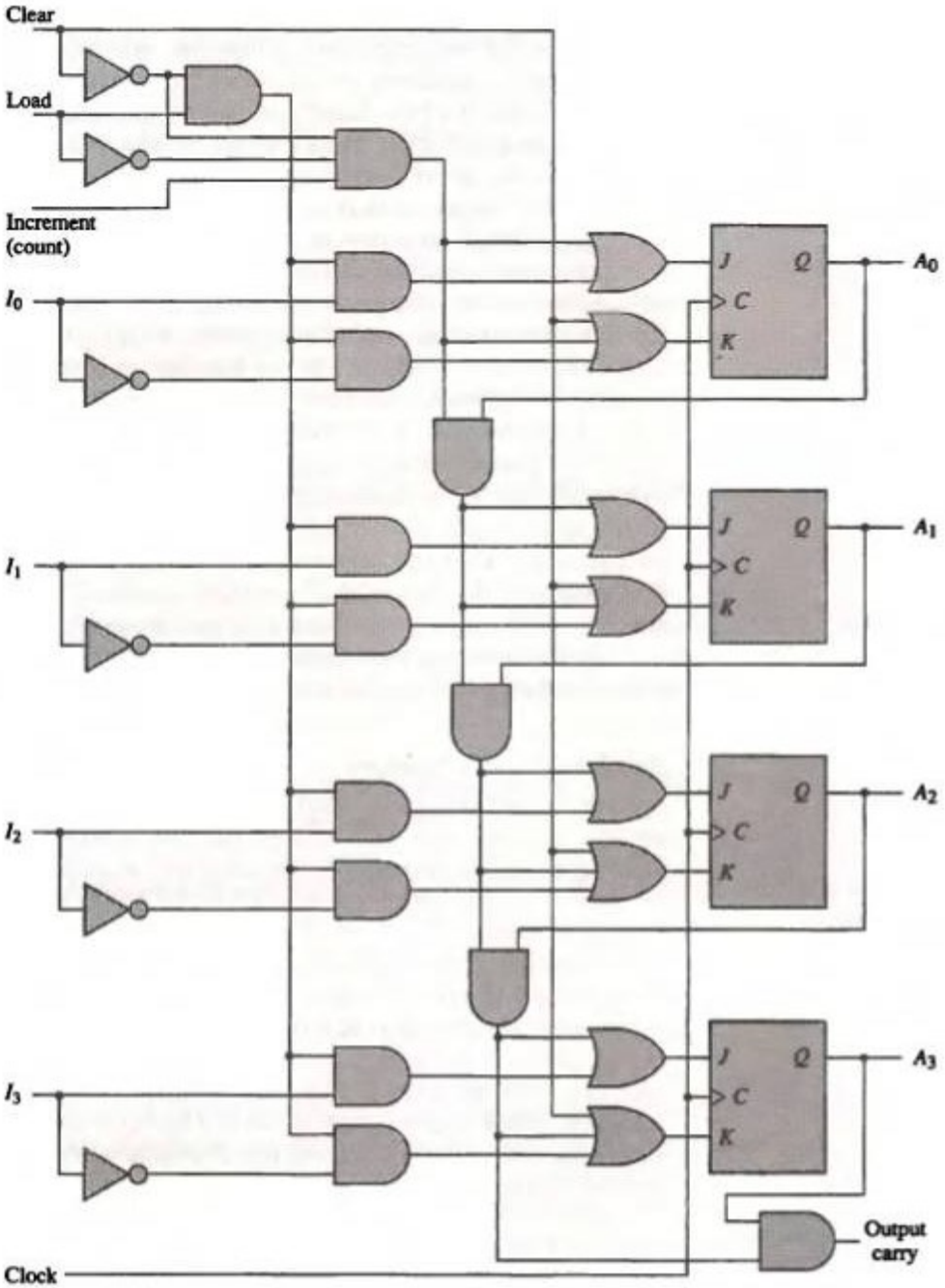


Fig 5-28 4-bit binary counter with parallel load and synchronous clear

•When equal to 1, the clear input sets all K inputs to 1, thus clearing all flip-flops with the next clock transition.

•The input load control when equal to 1, disables the count operation and causes a transfer of data from the four parallel inputs into the four flip-flops (provided that the clear input is 0).

•If the clear and load input are both 0 and the increment input is 1, the circuit operates as a binary counter.

•The operation of the circuit is summarized in Table 2-5.

Clock	Clear	Load	Increment	Operation
↑	0	0	0	No change
↑	0	0	1	Increment count by 1
↑	0	1	×	Load inputs I_0 through I_3
↑	1	×	×	Clear outputs to 0

Table 2-5 Function table for the Counter of Fig 5-28

•Counters with parallel load are very useful in the design of digital computers.

•we refer to them as registers with load and increment operations.

•The increment operation adds 1 to the content of the register.

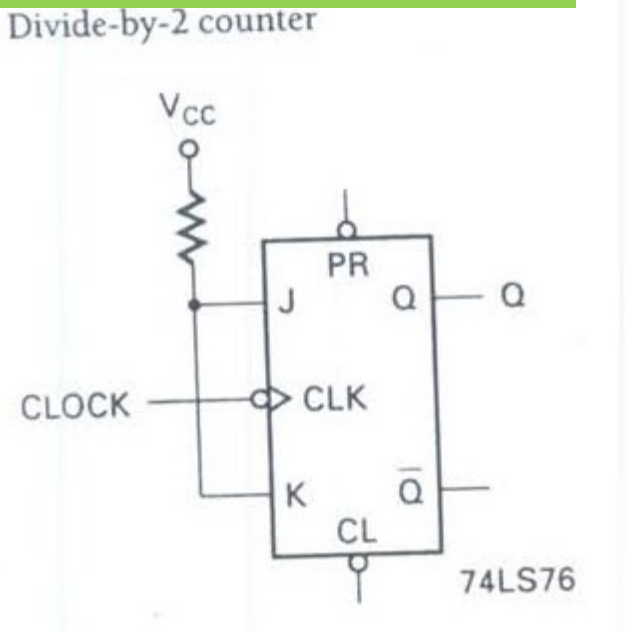
•By enabling the count input during one clock period, the content of the register can be incremented by one.

•**Asynchronous Counters**

•Asynchronous counters are counters that don't have a common clock. The clock input is used as a data input.

•We can use a J-K flip flop to construct a T-flip flop which will be used to design Divide by 2, 4 and 8 Asynchronous counters.

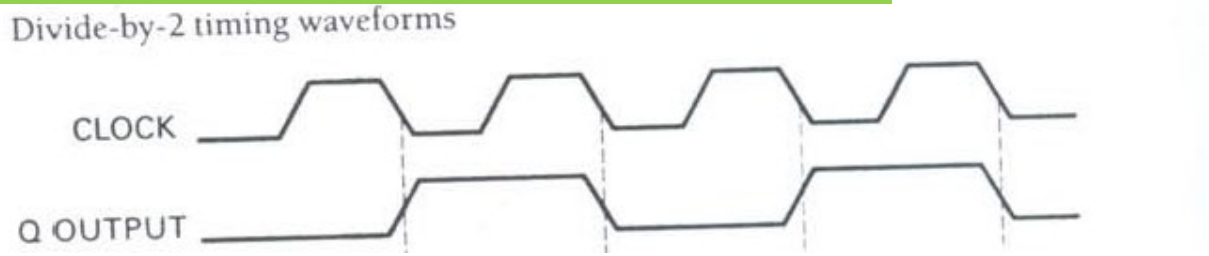
Figure 5-29 Divide-by-2 Counter



•When both the J and K inputs are high, the output Q will be complemented at every clock pulse.

•When this T-flip flop is a negative edge triggered flip flop, the output Q will change states from 0 to 1 and from 1 to 0 at the negative edge of the clock pulse.

Figure 5-30 Divide-by-2 Timing Waveforms



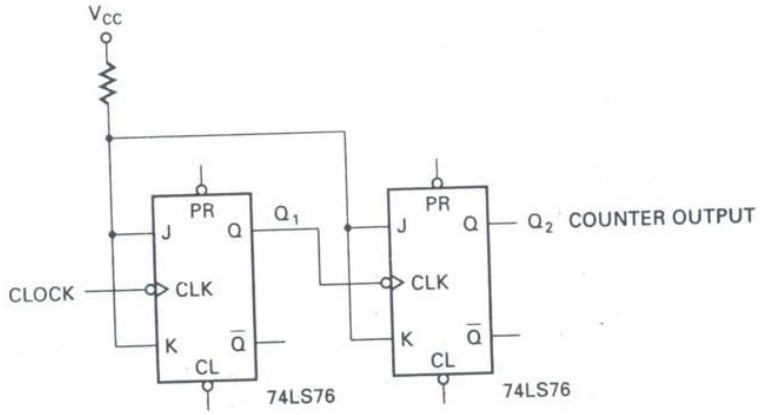
•And this will sample the clock pulse, which is the data input, into a bigger width clock pulse, where two pulses will merge into one pulse. Thus dividing the clock frequency by two and increasing its amplitude.

•**Divide by 2, 4, 8 Asynchronous Counters**

•A divide-by-2 counter can be constructed with a single T-flip flop.

•A divide-by-4 counter is constructed with two T-flip flops the output of one T-flip flop going into the second and only the first receiving the clock pulse as a data input.

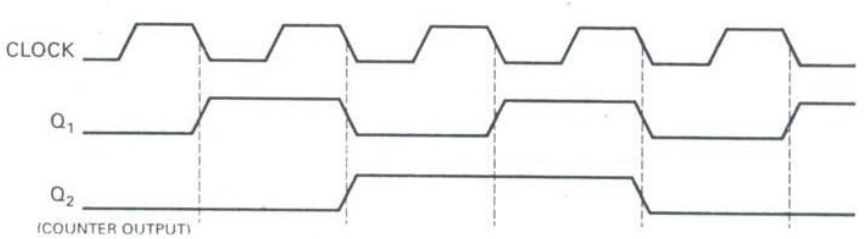
Figure 5.60
Divide-by-4 counter



- The first Q output will divide the clock frequency by two and this frequency of the Q output will be further divided by two.

- Thus, at the output of the second flip-flop, the original clock frequency will be divided by four.

Figure 5.61
Divide-by-4 counter timing diagram



•We can label the binary states, with Q2 being the most significant bit, and Q1 being the list significant bit and obtain a total of 2^2=4 possible combinations of bits, considering each state of the clock pulse, Q1 and Q2 as inputs.

•A divide-by-8 counter will require three flip flops, the output of each flip flop diving the clock frequency by two and the last output having divided the clock frequency by 8.

Figure 5.62
Divide-by-8 counter

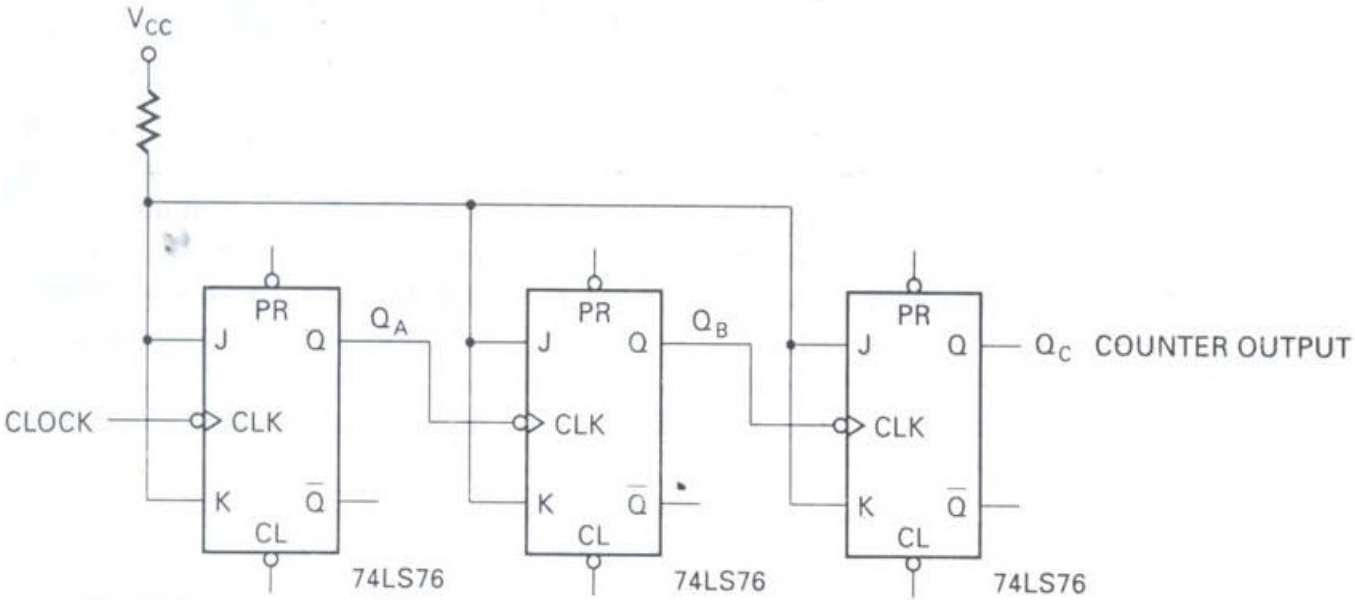
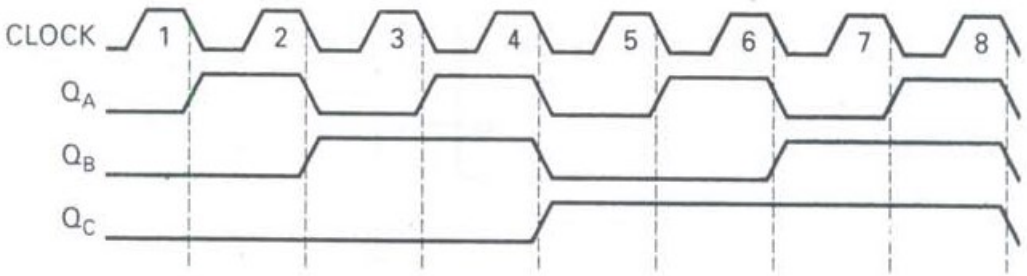


Figure 5.63
Divide-by-8 counter timing diagram



A divide-by-8 counter has 8 states from the output of each flip flop, QA, QB, QC with QC being the most significant bit: 000,001,010,011,100,101,110,111

When we focus on the state of flip-flops and all the possible combinations of the output bit values, we label it a modulo-8 counter.

The relationship between the number of flip-flops and the number of states(n) can be expressed as:

$n = 2^{\text{no. of flip-flops}}$

Each flip-flop output can be considered as a boolean variable.

The timing diagram, such as that shown in Fig. 5-63 is a time dependent truth table.
Each of the variables(the flip-flop Q outputs) changes with time and at a given time has a unique value.

Review Exercises

1. Design a 2-bit count-down counter. This is a sequential circuit with two flip-flops and one input x. When x=0, the state of the flip-flops does not change. When x=1, the state sequence is 11, 10, 01, 00, 11 and repeat.

2. The content of a 4-bit register is initially 1101. The register is shifted six times to the right with the serial input being 101101. What is the content of the register after each shift?

3.Show the connections between two 4-bit binary counters with parallel load to produce an 8-bit binary counter with parallel load. Use the block diagram for each 4-bit counter.