Unit 5 Sequential Circuits and Applications

Structure:

5.1 Introduction

In the previous unit we studied about combinational circuits, and different codes like Gray code, BCD code, and Excess-3 code. In 1918 William Eccles and F.W. Jordan invented the first flip flop. Initially it was named after the inventers and was called as Eccles Jordan trigger circuit. The name flip-flop was later derived from the sound produced on a speaker connected to one of the back coupled amplifiers outputs during the trigger process within the circuit. This original *electronic* flip-flop – a simple two-input bistable circuit without any dedicated clock (or even gate) signal, was *transparent*, and thus a device would be labeled as a "latch" in many circles today.

In digital circuits, combinational circuits don’t have memory to store the values and its output depends only on the inputs. A powerful model is required to build a complex digital system. In order to build a powerful model, a digital circuit which has a memory and its output should depend on its previous state and also on the input to the circuit is required. So a device should have following three characteristics to serve as a memory:

- two stable states should exists in the device
- the state of the device should be readable.
- at least once, we should be able to set the state of the device.

The concept of feedback should be used to construct a digital circuit with memory. The digital circuits are said to be acyclic if the logical flow is from input to output. In this unit, we will study about the definition of sequential circuit, latches and flip flops and various types of flip-flops and their operation.

Objectives:

By the end of Unit 5, the learners are able to:

- define sequential circuits
- define Latch
- list and explain different types of Flip-Flops
- discuss on real world applications of sequential circuits

5.2 Definition of Sequential Circuits

Both combinational circuits such as universal gates and memory such as latches and flip flops are used to construct a digital system. In many sequential circuits (like counter and shift registers), we can see the interconnection of both logic gates and memory elements. While designing the digital systems, the additional variable and timing information of the memory should be taken care. Sequential logic operations should be performed, the data is stored in a memory and later combinational operation uses the data when it is released from the memory. The digital systems which perform sequential operations are known as sequentially operated systems. Sequential circuits are those whose outputs depends not only on the present value of its inputs but also on past history of its inputs. There are two types of memory elements which are used in sequential circuits, they are latch and flip flop. Flip flop is a device which changes its state at the positive edge or negative edge (also known as leading edge and trailing edge) of the clock signal. Asynchronous latch is a device which changes its state whenever there is a change in the input signals irrespective of control signal value. A synchronous latch is device which monitors the input signal and changes its state whenever the control signal is high.

Events are used to drive both Asynchronous and Synchronous latches. But the output of the synchronous latch is changed only when the control signal is active.

Self Assessment Questions

1. The digital systems which perform sequential operations are known as sequentially operated systems. (True or False?).
2. _____ is a device which changes its state at the positive edge or negative edge of the clock signal.

5.3 Latch

The S-R Latch

The figure 5.1 shows the SR latch constructed using NOR gates. From figure 5.1, it can be observed that the output of each NOR gate is given as feedback to the other NOR gate.

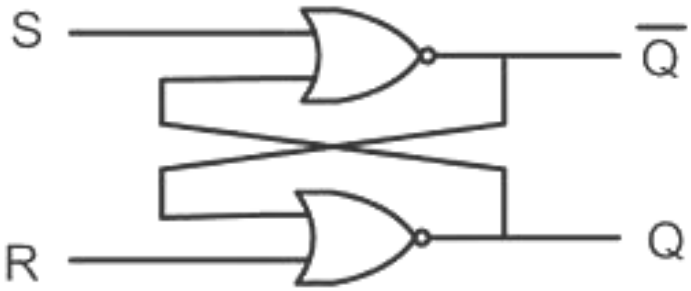

Figure 5.1: The S-R Latch. S sets the latch, causing Q to become true. R resets the latch.

When both the inputs to an NOR gate are low, the output will be high. From the circuit it can be analyzed that among two NOR gate outputs, only one output will be high and other will be low. Consider that the output of the upper NOR gate ($\overline{Q}$) is high, as this output is connected as input to the lower NOR gate its output will be low. This circuit is an **S-R latch.** Where S stands for set and R stands for reset, because of which it is also known as set-reset latch.

By examining the circuit we can see that both the inputs of NOR gate should be low to generate a high output. When the set S button is pressed, the output of the latch will be high i.e.; output of lower gate and its complement output will be low. The latch will be set i.e. its output will be high when the Set S input is high, and the latch will be reset i.e., its output will be low when the reset R input is high. Even when the inputs are removed the circuit will be in the stable state. The inputs which generated a particular output and

unchanged the output value can be removed. The complementary input changes the output and the state of the circuit. As the circuit has two stable states, this kind of circuits are said to be bistable.

The figure 5.2 shows the logic symbol of SR latch.

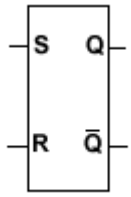

Figure 5.2: The symbol for the S-R latch

The availability of the complemented output pin depends on the packaging of the latch and on the availability of an extra pin. Both the outputs of the circuit will be false if both inputs S and R are equal to 1. This kind of outputs is inconsistent logically and also the circuit will be in unstable state. The circuit will come to stable state from unstable state when one of the inputs is made low and the output also changes. The circuit which stores a single bit is described by a keyword latch.

Timing Considerations

Before discussing further topics, the scenario like the output of the combinational circuit formed from the combination of outputs from two or more gates should also be considered. In the earlier topics, it has been discussed that in a transistor, switching happens in nanoseconds, but even this switching time will cause a finite amount of time delay to change in the output for the given inputs. This time is called **gate delay.** Causing of wrong output because of the timing dependencies is known as **hazard.**

Clocking

A storage element which accepts the input only upon event should be designed to avoid glitches. The event is given only when the input to the storage device which is computed by the combinational circuits have a chance to settle down to the correct value.

One way to do that is to interpose AND gates between the S and R inputs and the latch circuit. The control signal drives the other input of each AND gate. When the control signal is false, the output of the two AND gates is always low and changes to S and R do not affect the bit stored by the latch.

When the control signal is true, the S and R signals are propagated through the AND gates and the stored value can change. Because the control input is generally driven by a regular train of pulses, it is often called a *clock* input. Figure 5.3-A shows the digital circuit of a clocked S-R latch and figure 5.3-B shows the symbol for the clocked S-R latch.

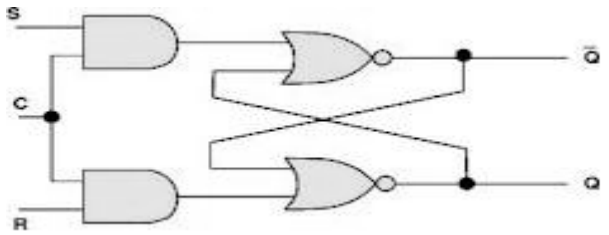

Figure 5.3-A: Clocked S-R latch. The latch can change only when C is true.

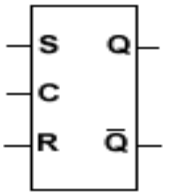

Figure 5.3-B: The symbol for the clocked S-R latch.

First experiment the given digital circuit with different combination of S and R inputs by giving low to C input. Later a high value is applied to C and circuit is experimented with different values of S and R inputs. But when both S and R inputs are high, clocking will not help much. When S and R values are equal to 1, and C is clocking, it is difficult to predict which value will be stored in the memory, either true value or complement value. But latch will settle down to one of the two stable states when both inputs S and R are removed at the same time.

The Clocked D-Latch

When the idea of clocking is applied to S-R latch, the problem of what should happen when S=R=1 can be taken care of and the input to the circuit can also be simplified.

Usually one bit information should be stored in the memory element. The complexity of the circuit increases when the set and reset of the latch are explicitly needed.

The main intention is to design a circuit which has a data input and data output. When the clock signal is high, irrespective of the value at data input D storage device should store the value of D and transfer to the data output Q.

Figure 5.4-A shows the circuit for clocked D latch and the figure 5.4-B shows the logic symbol of clocked D latch.

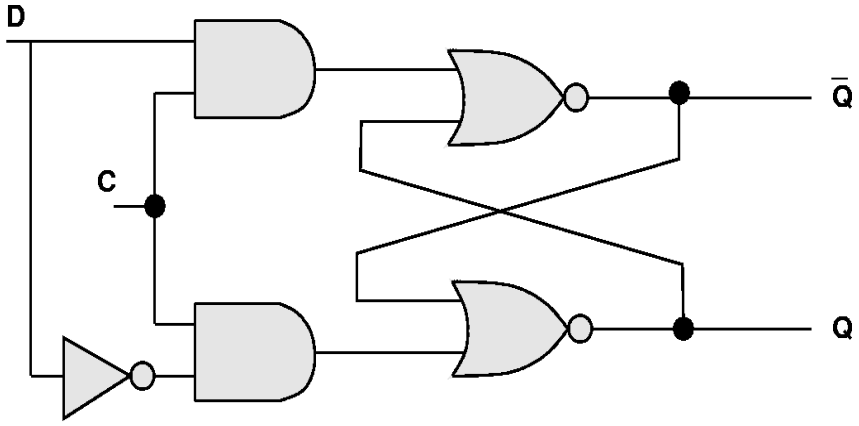

Figure 5.4-A: Cloked D-latch.When C(control) is true,the value at D(data) is stored

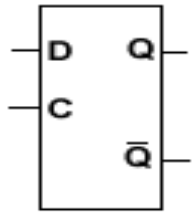

Figure 5.4-B: The symbol for the clocked D latch

The circuit shown in above figure 5.4-A has two inputs. One is control input and other is data input. An AND gate is used to connect the data input to the S input of an S-R latch and the data input is connected to R input through an inverter and an AND gate. The control input C is connected as the second input to both the AND gates. The state of the latch will not change when the control input C is applied with low value. When the control input and the data input are true (i.e. High), the Set input of the latch also will be true and therefore the true value is stored in the element. When the control input is true and the data input is false, the Reset input of the latch also will be true and therefore the false value (i.e low value) is stored in the element.

Here is something to think about: The concept of a D latch, where the bit to be stored is applied to the S input of a latch, and through an inverter to the R input, can only be made to work when the latch is clocked. Why is that?

When the control input is true, the clocked D latch stores the data input D value. The correct value of D will be stored in the latch if the control input is triggered only when the input to the circuit is settled.

The functionality of the circuit is analyzed by asserting the value of C and changing the values of D. The output of the circuit will be equal to the data input D if the C input is high. This kind of clocked D latches are said to be level triggered devices, the level at C says whether to store the data or not.

Self Assessment Questions

3. The latch will be set when the Set S input is _____.
4. Usually one bit information should be stored in the _____element.
5. The control input which is generally driven by a regular train of pulses is often called a _____ input.
6. When the control input is true, the clocked D latch stores the data input D value (True or False?).

5.4 Flip Flop

Usually the master slave devices are implemented using clocked flip flops. The two basic flip flops in master slave devices make the circuit insensitive to the noise and spike which are occurred between two clock transitions. This circuit might have asynchronous reset or set which changes the present value of the output irrespective of clock signal.

Flip-flops can be further divided into types that have found common applicability in both asynchronous and clocked sequential systems: the **SR** ("set-reset"), **D** ("data" or "delay"), **T** ("toggle"), and **JK** types are the common ones; all of which may be synthesized from (most) other types by a few logic gates. The behavior of a particular type can be described by what is termed the characteristic equation, which derives the "next" (i.e., after the next clock pulse) output, Q_{next}, in terms of the input signal(s) and/or the current output, Q. The circuit which is capable of remembering the applied input is called a flip flop. As the flip flop stores the given values, in a memory device it can be used as basic storage device. Only one bit information can be stored in a Flip Flop.

Basic flip-flops

Two NOR gates or two NAND gates can be used to construct a flip flop. Usually flip flop are used in constructing the sequential circuits. As Flip flop has two stable states, it is also known as bi-state gate. Until the trigger is received by the flip flop, it maintains its state for indefinite amount of time. When the trigger is applied according to the predefined rules, the state of the flip flop changes and the new state will be unchanged till another trigger is applied.

Flip – Flop circuit using NOR gates

The functionality of the flip flop can be explained using cross coupled two NOR gates or NAND gates. The cross coupled circuits will have the feedback paths. The figure 5.5 shows the logic diagram of SR Flip flop using NOR gates

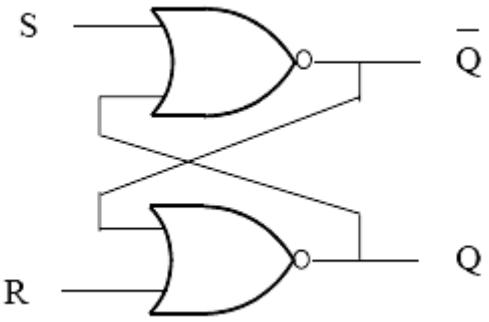

Figure 5.5: SR Flip flop using NOR gates

Normal value (Q) and the complement value ($\overline{Q}$) are the two outputs of a flip flop. Set(S) and Reset (R) are the two inputs for above circuit. The current state of the output is determined by the feeding back the previous states. i.e output is feedback to the input as shown in figure 5.5. Flip-flops which are constructed using NOR gates works normally at input equal to logic zero.

When the S input is equal to 1 and R input is 0, the output $\overline{Q}$ is equal to 0. As this $\overline{Q}$ value is given again along with R, the output Q will become 1. Therefore, from the above discussion we can say that when S = 1and R = 0, outputs Q will be 1 and $\overline{Q}$ will be 0.

The state of outputs will be unchanged even when the Set input is made 0 and R is also 0. When the S input is equal to 0 and R input is 1, the output

Q is equal to 0. As this Q value is given again along with S, the output $\overline{Q}$ will become 1. Therefore, from the above discussion we can say that when S = 0 and R = 1, outputs Q will be 0 and $\overline{Q}$ will be 1. The state of outputs will be unchanged even when the Set input is 0 and R is made 0. The truth table for S-R flip flop is shown in the table 5.1.

Table 5.1: Truth table for S-R flip flop

S	R	Q	$\overline{Q}$	
1	0	1	0	
0	0	1	0	(after S =1 and R = 0)
0	1	0	1	
0	0	0	1	(after S =0 and R = 1)

The outputs Q and $\overline{Q}$ are 0's when inputs S = 1 and R = 1. This violates the fact that Q and $\overline{Q}$ are complement to each other. This condition must be avoided in normal operations.

From the above discussion, it can be observed that the flip flop has two stable state i.e., set state (Q =1, $\overline{Q}$ = 0) and reset state (Q = 0, $\overline{Q}$ = 1).

Flip – Flop Circuit using NAND Gates

By cross coupling the NAND gates a flip flop can also be constructed. This is shown in figure 5.6.

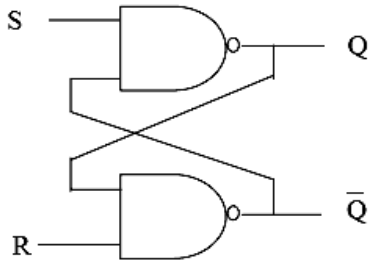

Figure 5.6: SR Flip flop using NAND gates

The corresponding truth table is given in table 5.2.

Table 5.2: Truth table for S-R flip flop using NAND gates

S	R	Q	$\overline{Q}$	
1	0	0	1	
1	1	0	1	(after S =1 and R = 0)
0	1	1	0	
1	1	1	0	(after S =0 and R = 1)

Flip flops which are constructed using NAND gates works normally at input equal to logic one. When the input S is applied with logic 0, then the outputs Q will be 1 and $\overline{Q}$ will be 0. Then flip flop is said to be in set sate. If R input is applied with logic 0, then outputs Q will be 0 and $\overline{Q}$ will be 1. Then flip flop is said to be in reset state. Both the output Q and $\overline{Q}$ will be 1 if S and R inputs are applied with logic 0. In normal operation this condition must be avoided.

R-S, D, J-K and T are the different types of flip flops used in designing sequential circuits. The digital circuits like memories and microprocessors can be constructed by interconnecting flip-flops to form the logic gates.

5.4.1 SR flip-Flop

The figure 5.7 shows the logic symbol of Set-Reset flip-flops (SR flip-flop).

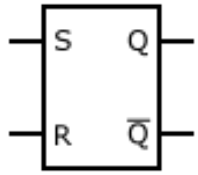

Figure 5.7: The symbol for an SR flip flop

Normally, in storage mode, the S and R inputs are both low, and feedback maintains the Q and Q outputs in a constant state, with Q the complement of Q. If S (*Set*) is pulsed high while R is held low, then the Q output is forced high, and stays high even after S returns low; similarly, if R (*Reset*) is pulsed high while S is held low, then the Q output is forced low, and stays low even after R returns low. The table 5.3 shows the truth table SR flip-flop which shows the operation of SR flip-flop.

Table 5.3: Truth table for the operation of SR flip-flop

SR Flip-Flop operation							
Characteristic table			Excitation table				
S	R	Action	Q(t)	Q(t+1)	S	R	Action
0	0	Keep state	0	0	0	X	No change
0	1	Q = 0	0	1	1	0	Set
1	0	Q = 1	1	0	0	1	Reset
1	1	Unstable combination,	1	1	X	0	No change

5.4.2 JK flip-flop

The figure 5.8 shows the timing diagram for JK flip-flop.

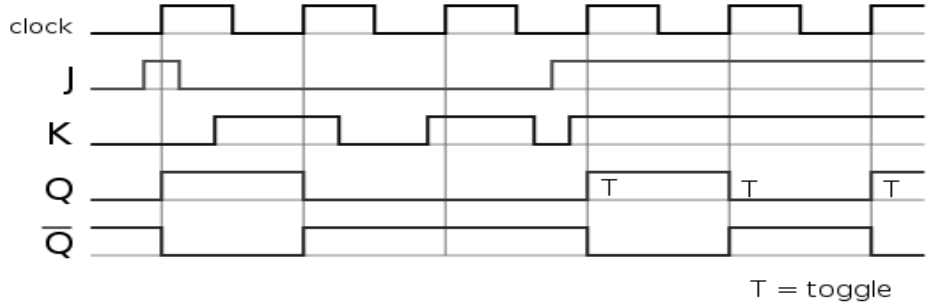

Figure 5.8: JK flip-flop timing diagram

The **JK** flip-flop augments the behavior of the SR flip-flop (J=Set, K=Reset) by interpreting the S = R = 1 condition as a "flip" or toggle command. The flip flop will be set when J is 1 and K is 0; and flip flop will be reset when J is 0 and K is 1; the output of the flip flop will toggle when both J and K are 1.Setting J = K = 0 does NOT result in a D flip-flop, but rather, will hold the current state. The D flip flop can be obtained from JK flip flop by setting complement of J to K. As JK flip flop can be configured to any of other 3 flip flops (SR, D, T), it is considered as Universal flip flop. The figure 5.9 shows the logic symbol of JK flip-flop.

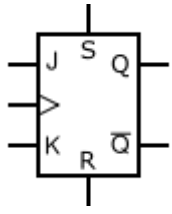

Figure 5.9: The symbol of JK flip flop

In the JK flip flop circuit symbol, the clock input is shown as **>** and data inputs are J, K. Q and $\overline{Q}$ are the data outputs. The characteristic equation of the JK flip-flop is:

$$Q_{next} = J\overline{Q} + \overline{K}Q$$

And the corresponding truth table is shown n in the table 5.4.

Table 5.4: Truth table for the operation of JK flip-flop

JK Flip Flop operation								
Characteristic table				Excitation table				
J	K	Q_{next}	Comment	Q	Q_{next}	J	K	Comment
0	0	Q_{prev}	hold state	0	0	0	X	No change
0	1	0	Reset	0	1	1	X	Set
1	0	1	Set	1	0	X	1	Reset
1	1	$\overline{Q_{prev}}$	toggle	1	1	X	0	No change

The origin of the name for the JK flip-flop is detailed by P. L. Lindley, a JPL engineer, in a letter to *EDN*, an electronics design magazine. The letter is dated June 13, 1968, and was published in the August edition of the newsletter. In the letter, Mr. Lindley explains that he heard the story of the JK flip-flop from Dr. Eldred Nelson, who was responsible for coining the term while working at Hughes Aircraft. Flip-flops in use at Hughes at the time were all of the type that came to be known as J-K. Another theory holds that the set and reset inputs were given the symbols "J" and "K" after one of the engineers that helped design the J-K flip-flop, Jack Kilby.

5.4.3 D flip-flop

The symbol of D flip flop is shown in the figure 5.10.

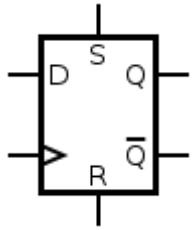

Figure 5.10: Symbol of D flip flop

During the positive edge of the clock or negative edge of the clock, the output of the flip flop will take the value of D. The flip flop output should not take the values of inputs on both the edges of the clock. As the output of the flip flop takes the value of input after one clock cycle, it is called the D or Delay flip flop. The D flip flop can also be used as a memory cell or zero order hold or a delay line. The table 5.5 shows the truth table of D flip-flop.

Table 5.5: Truth table of D flip-flop

Clock	D	Q	Q_{prev}
Rising edge	0	0	X
Rising edge	1	1	X
Non-Rising	X	constant	

('X' denotes a *Don't care* condition, meaning the signal is irrelevant)

All the shift registers which are important part of digital system are constructed using flip flops. When compared to D latch, the advantage of the D flip flop is that the input is captured only during positive edge or negative edge of the clock and any changes in input line will not be stored in flip flop at the high or low level of the clock. A flip flop can have a reset signal which can be used to reset the output to zero; the reset to the flip flop can be synchronous or asynchronous.

Master-slave (pulse-triggered) D flip-flop

By connecting two gated D latches in series as shown in figure 5.11, a master-slave D flip flop can be constructed. In this master-slave D flip flop one of the two gated is connected to active low enable. As the second flip flop responds for the changes from first flip flop (master) this circuit is called master slave. The circuit is said to be pulse triggered if the data is accepted at the positive edge of the clock and that value is reflected at the output at the negative edge of the clock.

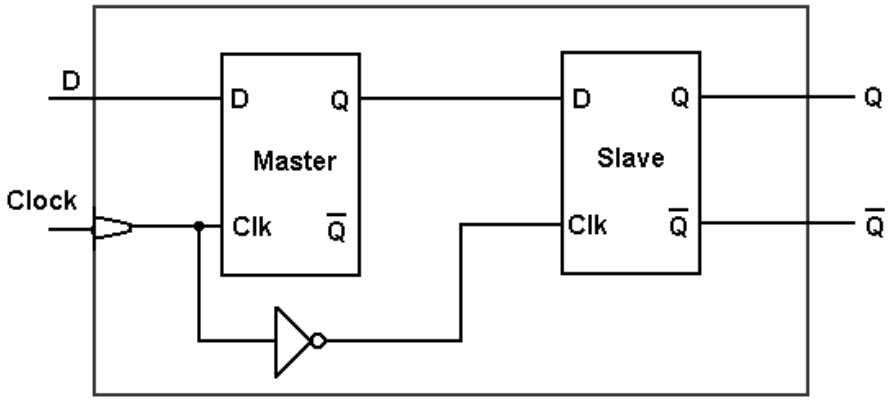

Figure 5.11: Master-slave (pulse-triggered) D flip-flop

Usually a master slave flip flop responds during the negative edge of the enable input. When the clock input low for a positive edge triggered master slave D flip flop and enable to the master is high. During the transition of clock from low to high, the input value will be latched. When the clock is changed from 0 to 1, the enable to the master will go low and its locks the value which is at master's input. At the same time, the enable to the slave will be changed from low to high and the signal which is captured by the master is latched by the slave. When the clock signal is changed from high to low, the output of the slave is locked and holds the value which is seen at the last positive edge while the master accepts the new value. The figure 5.12 shows the circuit for positive edge triggered master slave D flip flop.

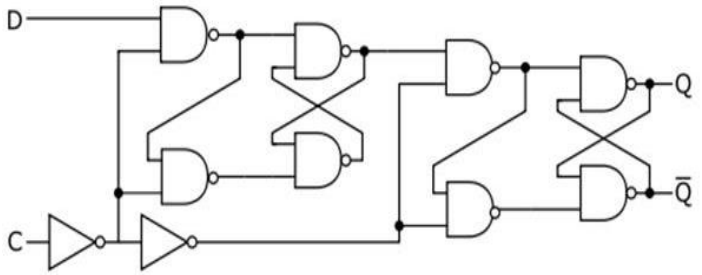

Figure 5.12: Circuit for positive edge triggered master slave D flip flop

A negative edge triggered flip flop can be determined by removing the left most inverter in the above circuit. This has a truth table as shown in table 5.6.

Table 5.6: Truth table for negative edge triggered D flip- flop

D	Q	>	Q_{next}
0	X	Falling	0
1	X	Falling	1

Like SR flip flop, most of the D flip flops in digital systems has set and reset capability. In D flip flop the illegal condition i.e., S=R=1 in SR flip flop is resolved (refer to the truth table shown in table 5.7).

Table 5.7: Truth table for D flip-flop with S=R=1 resolved

Inputs				Outputs	
S	**R**	**D**	**>**	**Q**	**Q'**
0	1	X	X	0	1
1	0	X	X	1	0
1	1	X	X	1	1

By setting S = R = 0, the flip-flop can be used as described above.

Edge-triggered D flip-flop

A more efficient way to make a D flip-flop is not as easy to understand, but it works the same way. While the master-slave D flip flop is also triggered on the edge of a clock, its components are each triggered by clock levels. The "edge-triggered D flip-flop" does not have the master slave properties.

5.4.4 J-K Master Slave Flip Flop

The logic symbol for the master-slave flip-flop shown in figure 5.13 only indicates the initial inputs to the master and the outputs from the slave. Master-slave flip flop is designed using two separate flip flops. One acts as the master and the other as a slave as shown in figure 5.13.

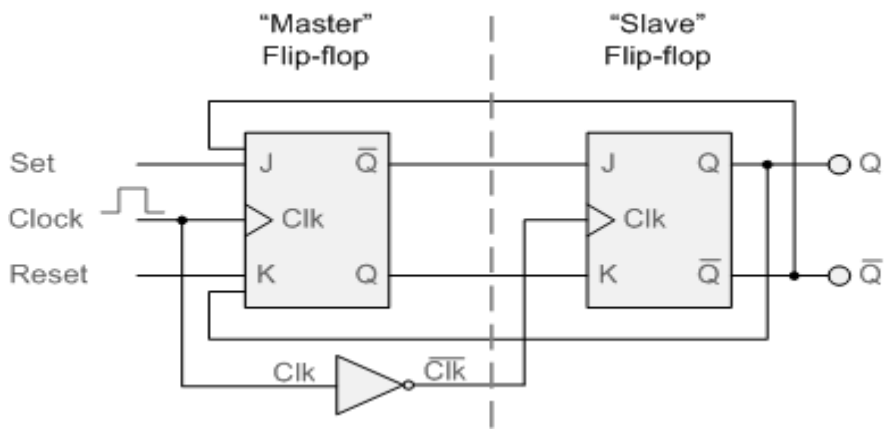

Figure 5.13: Symbol for the master-slave JK flip-flop

Let us now examine the operation of the master-slave J-K flip-flop:

When clock signal i.e. Clk=1, the master J-K flip flop gets disabled. The Clk input of the master input will be the opposite of the slave input. So the master flip flop output will be recognized by the slave flip flop only when the Clk value becomes 0. So when the clock pulse makes a transition from 1 to 0, the locked outputs of the master flip flop are fed through to the inputs of the slave flip-flop making this flip flop edge or pulse-triggered. To understand this operation, take a look at the timing diagram shown in figure 5.14.

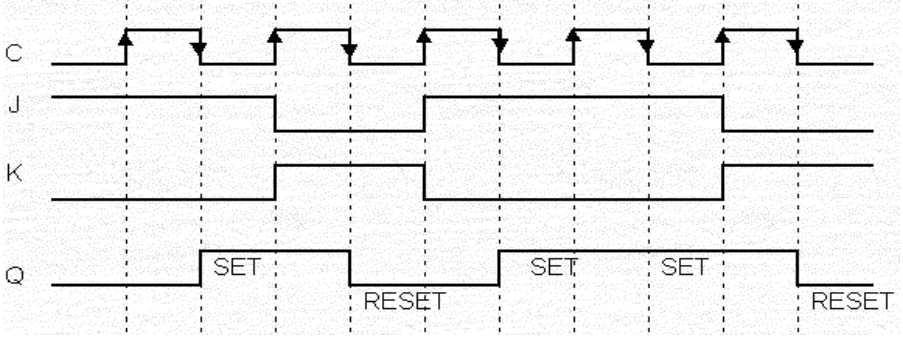

Figure 5.14: Timing Diagram for the master-slave JK flip-flop

From the timing diagram, it is clear the circuit accepts the value in the input when the clock is HIGH, and passes the data to the output on the falling-edge of the clock signal (i.e when clock pulse is making transition from 1 to 0). Thus Master-Slave J-K flip flop is a Synchronous device since it only passes data with the timing of the clock signal.

Application of flip-flops

An important application of flip-flops is in the design of digital counters. These devices generate binary numbers in a specified count sequence when triggered by an incoming clock waveform. On each trigger, the counter advances to the next number in the sequence. After reaching the final state in the sequence, the counter then recycles. Counters may be used to count up or down, to cycle through memory addresses in microprocessors applications, to generate waveforms of particular patterns and frequencies, and to activate other logic circuits in a complex process.

5.5 Real World Applications of Sequential Circuits

Design of Sequential Circuits

The design of a synchronous sequential circuit starts from a set of specifications and culminates in a logic diagram or a list of Boolean functions from which a logic diagram can be obtained. In contrast to a combinational logic, which is fully specified by a truth table, a sequential circuit requires a state table for its specification. The first step in the design of sequential circuits is to obtain a state table or an equivalence representation, such as a state diagram. A synchronous sequential circuit is made up of flip-flops and combinational gates. The design of the circuit consists of choosing the flip-flops and then finding the combinational structure which, together with the flip-flops, produces a circuit that fulfills the required specifications. The number of flip-flops is determined from the number of states needed in the circuit. The recommended steps for the design of sequential circuits are set out as shown in the figure 5.15.

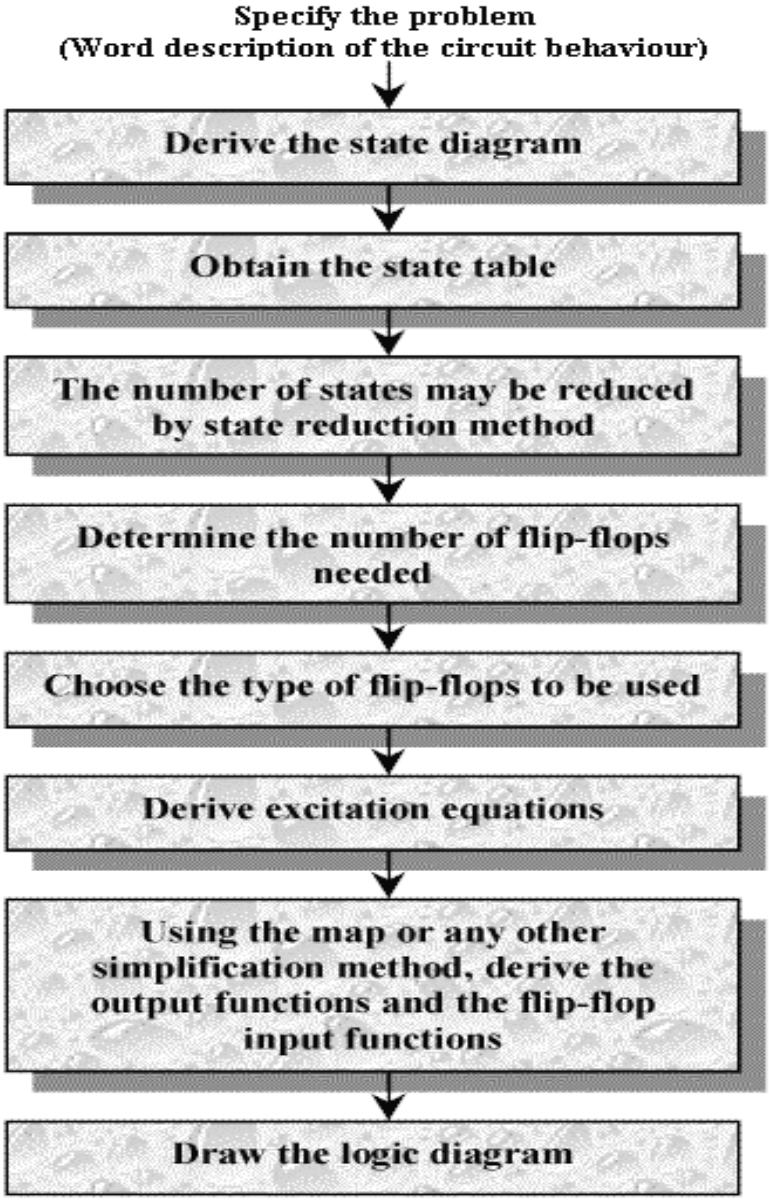

Figure 5.15: Steps for the design of sequential circuits

Example 1: We wish to design a synchronous sequential circuit whose state diagram is shown in figure 5.16. The type of flip-flop to be use is J-K.

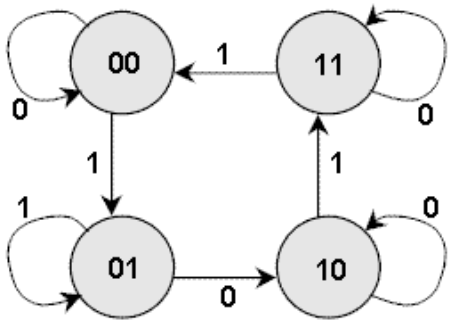

Figure 5.16: State diagram

From the state diagram, we can generate the state table shown in table 5.8 Note that there is no output section for this circuit. Two flip-flops are needed to represent the four states and are designated Q0Q1. The input variable is labeled x.

Table 5.8: State table

Present State	Next State	
Q0 Q1	x = 0	x = 1
0 0	0 0	0 1
0 1	1 0	0 1
1 0	1 0	1 1
1 1	1 1	0 0

We shall now derive the excitation table and the combinational structure. The table is now arranged in a different form as shown in table 5.9, where the present state and input variables are arranged in the form of a truth table.

Table 5.9: Excitation table of the circuit

Output Transitions Q →Q (next)	Flip-flop inputs J K
0 → 0	0 X
0 → 1	1 X
1 → 0	X 1
1 → 1	X 0

The Table 5.10 shows the Excitation table with present state and next state.

Table 5.10: Excitation table with present state and next state

Present State	Next State	Input	Flip-flop Inputs	
Q0 Q1	Q0 Q1	x	J0K0	J1K1
0 0	0 0	0	0 X	0 X
0 0	0 1	1	0 X	1 X
0 1	1 0	0	1 X	X 1
0 1	0 1	1	0 X	X 0
1 0	1 0	0	X 0	0 X
1 0	1 1	1	X 0	1 X
1 1	1 1	0	X 0	X 0
1 1	0 0	1	X 1	X 1

In the first row of table 5.10 we have a transition for flip-flop Q0 from 0 in the present state to 0 in the next state. In table 5.10 we find that a transition of states from 0 to 0 requires that input J = 0 and input K = X. So 0 and X are copied in the first row under J0 and K0 respectively. Since the first row also shows a transition for the flip-flop Q1 from 0 in the present state to 0 in the next state, 0 and X are copied in the first row under J1 and K1. This process is continued for each row of the table and for each flip-flop, with the input conditions as specified in table 5.10.

The simplified Boolean functions for the combinational circuit can now be derived. The input variables are Q0, Q1, and x; the outputs are the variables J0, K0, J1 and K1.The information from the truth table is plotted on the Karnaugh maps shown in figure 5.17.

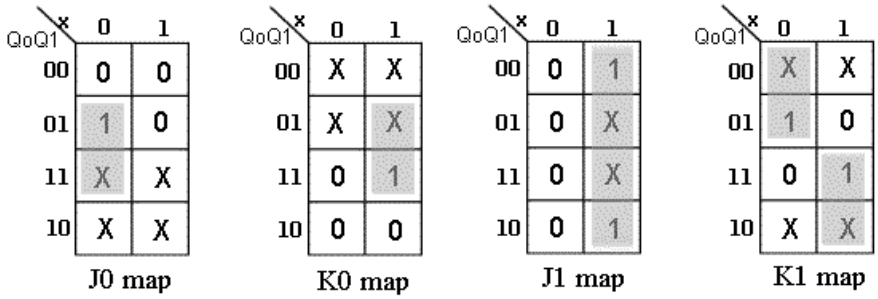

Figure 5.17: Karnaugh Maps

The flip-flop input functions are derived:

J0 = Q1*x' K0 = Q1*x

J1 = x K1 = Q0'*x' + Q0*x = Q0⊙x

Note: the symbol ⊙ is exclusive-NOR.

The logic diagram is drawn in figure 5.18.

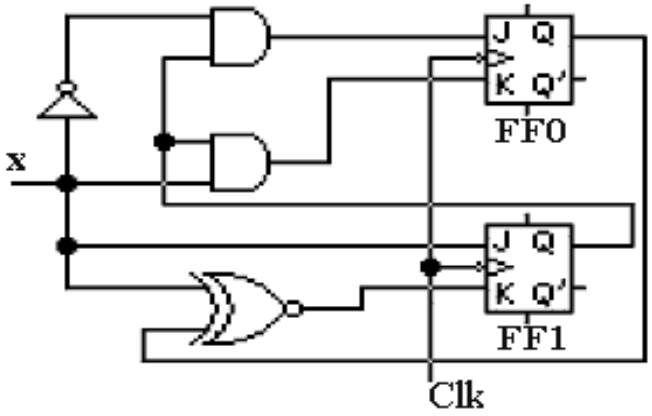

Figure 5.18: Logic diagram of the sequential circuit

Example 2: Design a sequential circuit whose state tables are specified in table 5.11, using D flip-flops.

Table 5.11: State table of a sequential circuit

Present State	Next State		Output	
Q0 Q1	X = 0	x = 1	x = 0	x = 1
0 0	0 0	0 1	0	0
0 1	0 0	1 0	0	0
1 0	1 1	1 0	0	0
1 1	0 0	0 1	0	1

Table 5.12: Excitation table for a D flip-flop

Output Transitions Q→ Q(next)	Flip-flop inputs D
0 → 0	0
0 → 1	1
1 → 0	0
1 → 1	1

The Table 5.12 shows the excitation table for D flip-flop. Next step is to derive the excitation table for the design circuit, which is shown in table 5.13. The output of the circuit is labeled Z.

Table 5.13: Excitation table

Present State	Next State	Input	Flip-flop Input		Output
Q0 Q1	Q0 Q1	x	D0	D1	Z
0 0	0 0	0	0	0	0
0 0	0 1	1	0	1	0
0 1	0 0	0	0	0	0
0 1	1 0	1	1	0	0
1 0	1 1	0	1	1	0
1 0	1 0	1	1	0	0
1 1	0 0	0	0	0	0
1 1	0 1	1	0	1	1

Now plot the flip-flop inputs and output functions on the Karnaugh map to derive the Boolean expressions, which is shown in figure 5.19.

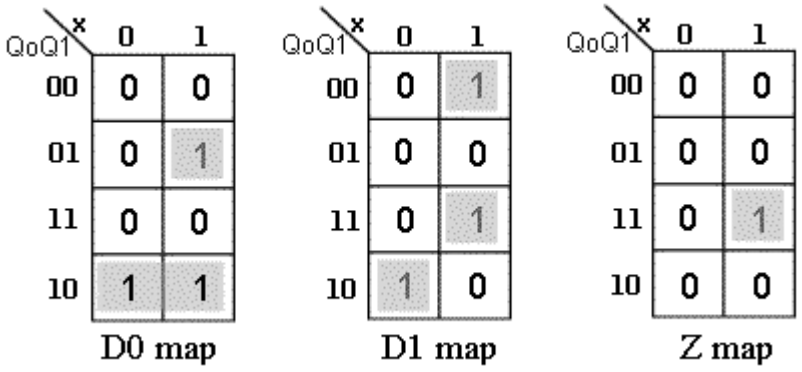

Figure 5.19: Karnaugh maps

The simplified Boolean expressions are:

D0 = Q0*Q1' + Q0'*Q1*x

D1 = Q0'*Q1'*x + Q0*Q1*x + Q0*Q1'*x'

Z = Q0*Q1*x

Finally, draw the logic diagram as shown in the figure 5.20

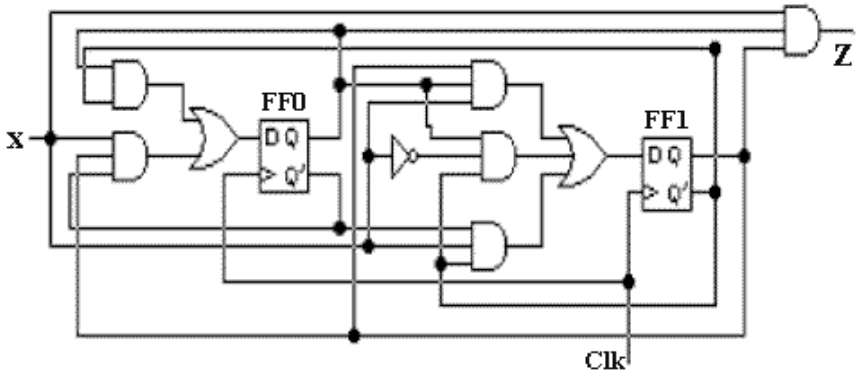

Figure 5.20: Logic diagram of the sequential circuit

Self Assessment Questions

7. As the flip flop stores the given values, in a memory device it can be used as _____device.
8. In case of flip flop using NAND gates, both the output Q and Qbar will be 1 if S and R inputs are applied with logic 0 and in normal operation this condition must be avoided. (State True or False?)
9. The design of the circuit consists of choosing the flip-flops and then finding the sequential structure which produces a circuit that fulfills the required specifications. (True or False?)
10. The master J-K flip flop gets enabled when clock signal i.e._____.
11. A synchronous sequential circuit is made up of _____and combinational gates
12. The first step in the design of sequential circuits is to obtain a _____ or an equivalence representation, such as a state diagram.

5.6 Summary

Let us recapitulate the important concepts discussed in this unit:

- Flip flop is a device which changes its state at the positive edge or negative edge (also known as leading edge and trailing edge) of the clock signal. A synchronous latch is device which monitors the input signal and changes its state whenever the control signal is high.
- As the flip flop stores the given values, in a memory device it can be used as basic storage device. Only one bit information can be stored in a Flip Flop.

- Two NOR gates or two NAND gates can be used to construct a basic flip flop.
- Flip-flops types are: SR ("set-reset"), D ("data" or "delay"), T ("toggle"), and JK flip flop.
- The output of the flip flop takes the value of input after one clock cycle in case of D flip flop.
- All the shift registers which are important part of digital system are constructed using flip flops.
- Master-slave flip flop is designed using two separate flip flops. One acts as the master and another acts as slave.

5.7 Terminal Questions

1. Define Sequential Circuits.
2. Draw and explain the working of JK, S-R, D Flip-Flops.
3. Write a short note on J-K Master Slave Flip-Flop.
4. What are the steps recommended for the design of sequential circuits?

5.8 Answers

Self Assessment Questions

1. True
2. Flip flop
3. High
4. Memory
5. Clock
6. True
7. Basic storage
8. True
9. True
10. Clk=1
11. Flip-flops
12. State table

Terminal Questions

1. Refer to section 5.2
2. Refer to section 5.4
3. Refer to sub-section 5.4.4
4. Refer to section 5.5