Sequential Logic

So far we have investigated combinational logic for which the output of the logic devices/circuits depends only on the **present** state of the inputs.

In sequential logic the output of the logic device is dependent not only on the present inputs to the device, but also on past inputs; i.e., the output of a sequential logic device depends on its present internal state and the present inputs. This implies that a sequential logic device has some kind of **memory** of at least part of its "history" (i.e., its previous inputs).

Sequential logic devices have some sort of feedback, where the output of some logic device is fed back to the input of a logic device. A simple memory circuit constructed from a OR gate is shown on Figure 1.
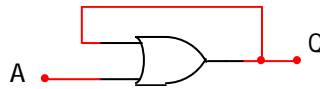

Figure 1.

In this memory device, if A and Q are initially at logic 0, then Q remains at logic 0. However if A becomes a logic 1, then the output Q will be logic 1 ever after, regardless of any further changes in the input at A. Once Q = 1 has been written it can't be changed back but it can be read. You may also think of this as a one time programmable memory element. Not a very useful circuit arrangement but it gave us our first exposure to positive feedback and memory.

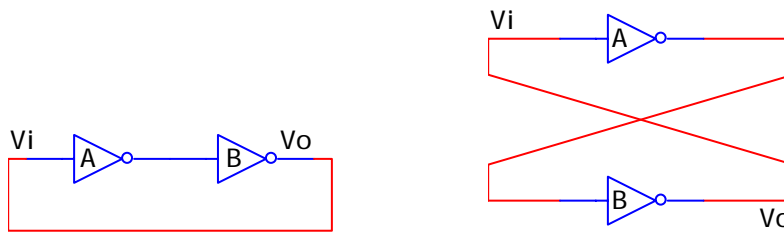Let's now consider a circuit consisting of two inverters connected in a loop as shown on Figure 2.

Figure 2. Fundamental latch

The voltage transfer characteristic of this circuit is shown on Figure 3. Note that the circuit has two stable states and it is thus called **bistable circuit**. This pair of inverters is also called a **latch**.
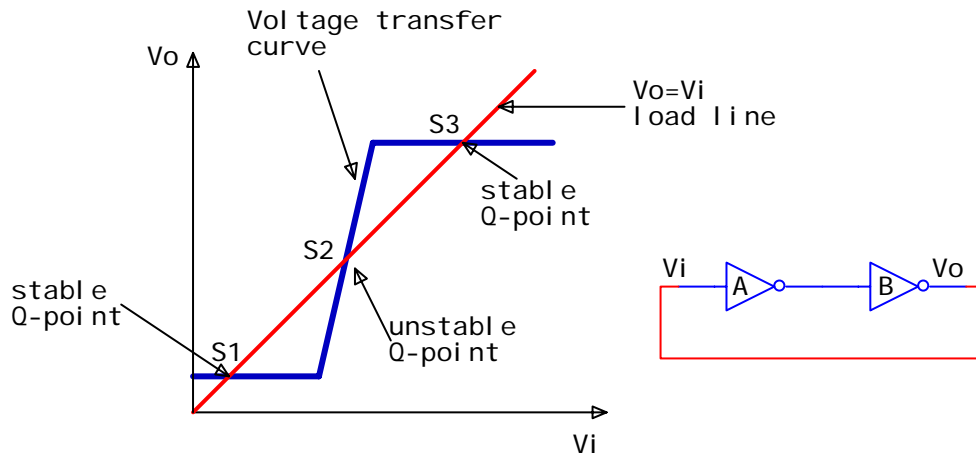
Figure 3. Voltage transfer characteristic of a latch (bistable circuit)

The operating points (Q-points) for this circuit must lie on the voltage transfer curve. Furthermore the loop connection imposes that the input and output voltages must be the same. Therefore the "load line" of this circuit is a line of slope 1 as indicated on Figure 3. The operating points must also lie on the load line curve.

Therefore the only possible operating points are those defined by the interception of the voltage transfer characteristic curve and the load line. These points are denoted by S1, S2 and S3 on Figure 3. Of these 3 possible operating points only 2, S1 and S3, are stable and therefore realizable. Point S2 is unstable. Indeed if we assume that the circuit is operating at S2 then due to the positive feedback characteristic of the system, any small deviation in the input, whether positive or negative, will drive the output to the top operating point S3 or to the bottom operating point S1 respectively.

The fundamental principles of sequential logic show us how to construct circuits that switch from one operating point to the other.

The Set-Reset (SR) Flip-Flop.
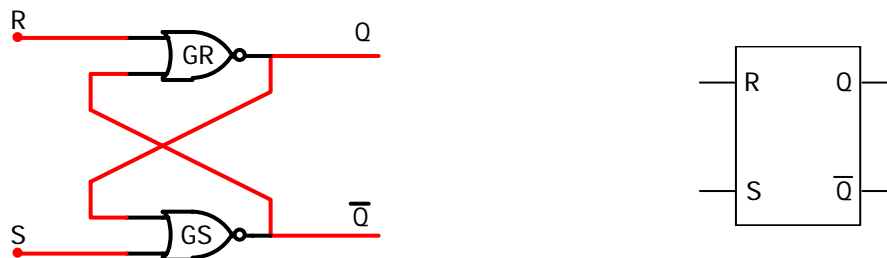
The SR flip-flop is shown on Figure 4.

Figure 4. SR Flip-Flop. Circuit and symbol

This sequential logic circuit is constructed with NOR gates and it has inputs labeled R and S which may assume the values 1 or 0. An equivalent representation of the circuit is shown on Figure 5 where the feedback loop is clearly shown.
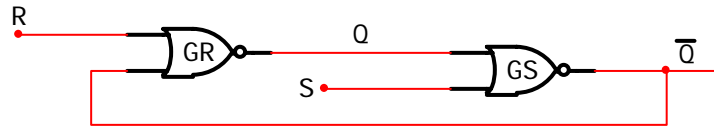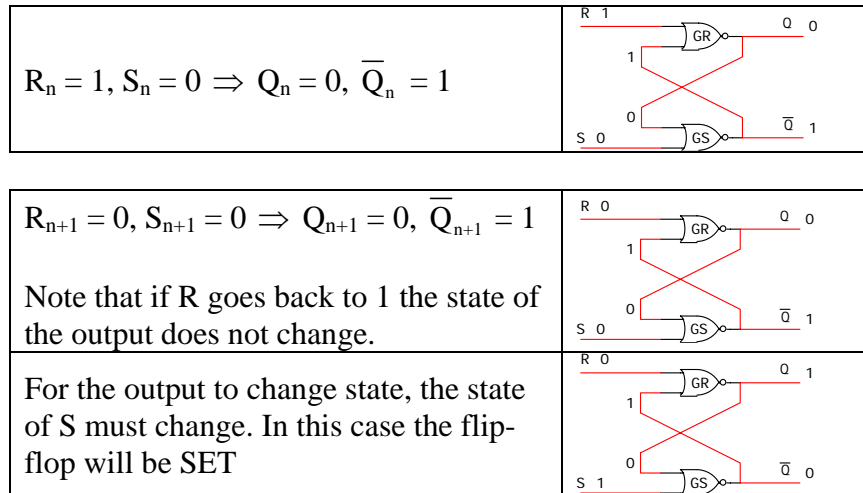

Figure 5. SR Flip-Flop

The outputs Q and $\overline{Q}$ are complementary. The function of the circuit is described by the table

R	S	Q_n
0	0	Q_{n-1}
0	1	1
1	0	0
1	1	Not allowed

This table, called the function table, represents the dynamics of the system, not just the static logic described in combinational logic circuits and represented by the truth table. Q_n represents the current state of the output and Q_{n-1} corresponds to the previous state.

In order to understand the behavior of the circuit let's first consider the following scenaria.

1. R = 0, S = 1. In this case the output of NOR gate GS must be 0 ($\overline{Q}$ = 0). Now both inputs of gate GR are 0 and so the output of GR must be 1. So Q = 1 and $\overline{Q}$ = 0. In this case we say that the flip-flop is SET

2. R = 1, S = 0. In this case the output of NOR gate GR must be 0 (Q = 0). Now both inputs of gate GS are 0 and so the output of GS must be 1. So Q = 0 and $\overline{Q}$ = 1. In this case we say that the flip-flop is RESET. Note that the notation make sense. When R changed from 0 to 1 the output was RESET.

3. R = 0, S = 0. In this case the output can be either 0 or 1. In fact the output in this case depends on its previous value. To see this explicitly consider the sequence of events.

$R_n = 1,\ S_n = 0 \Rightarrow Q_n = 0,\ \overline{Q}_n = 1$	

$R_{n+1} = 0,\ S_{n+1} = 0 \Rightarrow Q_{n+1} = 0,\ \overline{Q}_{n+1} = 1$ Note that if R goes back to 1 the state of the output does not change.	
For the output to change state, the state of S must change. In this case the flip-flop will be SET	

4. $R = 1$, $S = 1$. Not Allowed. In this case both outputs will try to go to 0. It is impossible to predict which output will go to 1 and which will stay at 0.

The dynamic characteristics of the flip-flop can be explicitly represented by the timing diagram shown below. The diagram on the left shows the transitions without any time delay. On the right the incorporation of time delay, td, through each gate is indicated in an exaggerated way for the purpose of the demonstration.
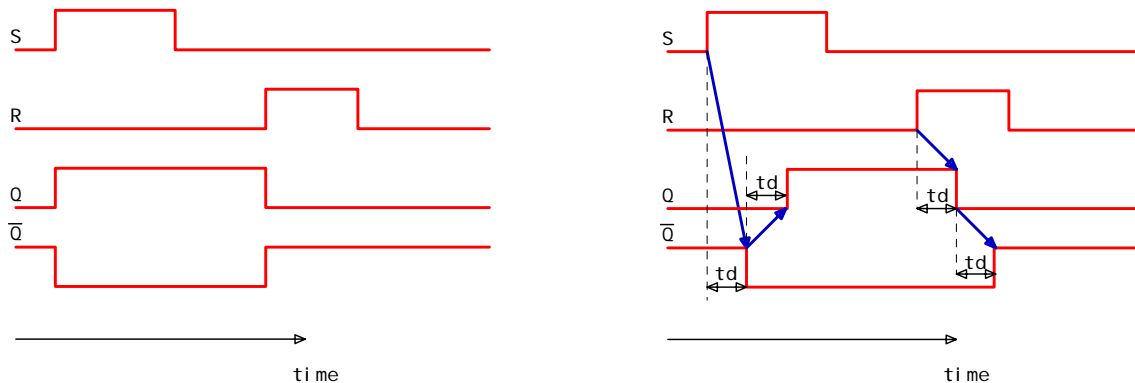
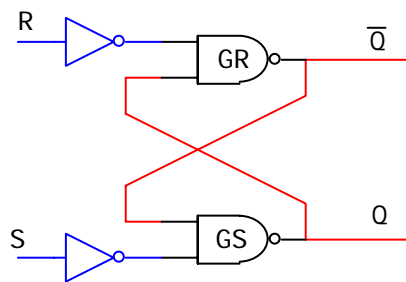The RS flip-flop may also be constructed with NAND gates as shown on Figure 6.

Figure 6. NAND implementation of SR flip-flop

Clocked SR Flip-Flop

Flip-Flops become very useful devices once we control their operation with some type of control signal. For example we might be interested in capturing data that is available for a short amount of time or at regular intervals.

When a clock pulse is used as the control signal the inputs R and S may be clocked (entered) into the flip-flop by a set of AND gates as shown on Figure 7.
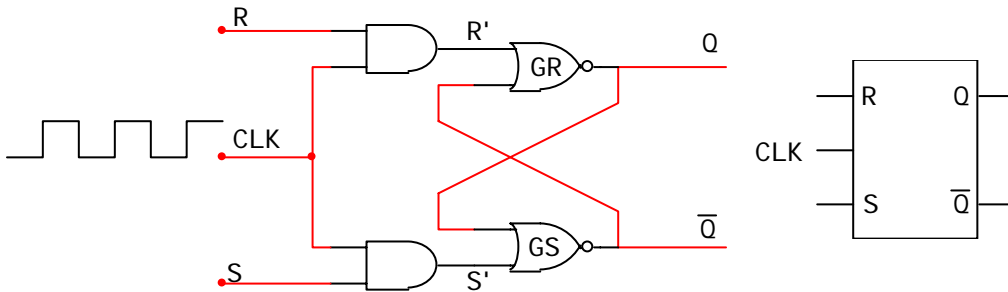

Figure 7. Clocked SR Flip-Flop and its symbol.

The obvious advantage of this clocked SR flip-flop is that the inputs R and S are considered only when the clock pulse is high. As before the condition R = S = 1 is indeterminate and should be avoided. A typical timing diagram for the clocked SR flip flop is shown on Figure 8.
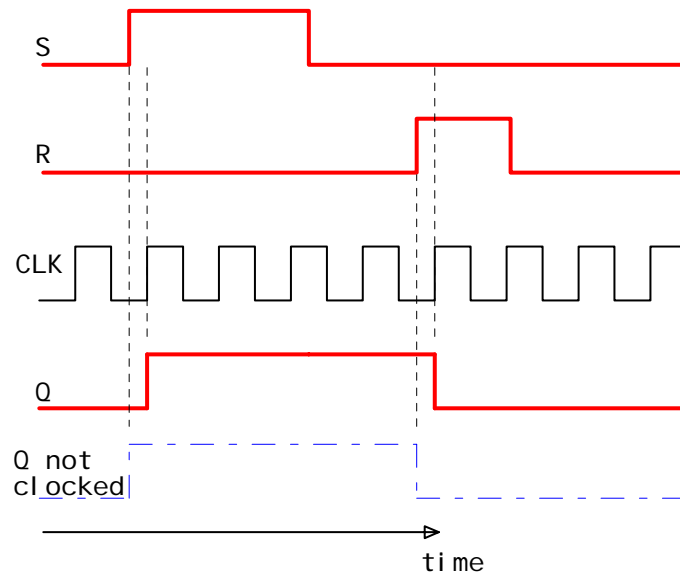

Figure 8. Timing diagram of clocked SR flip-flop

Note that the output responds to the clock pulse. The dashed line labeled "Q not clocked" represents the output of the non-clocked flip-flop.

For a finer control of the SR flip we may, with the circuit modification shown on Figure 9, enable signals R and S during the raising edge of the clock pulse. The delay is introduced by the inverter and the clocking is accomplished during the delay time.
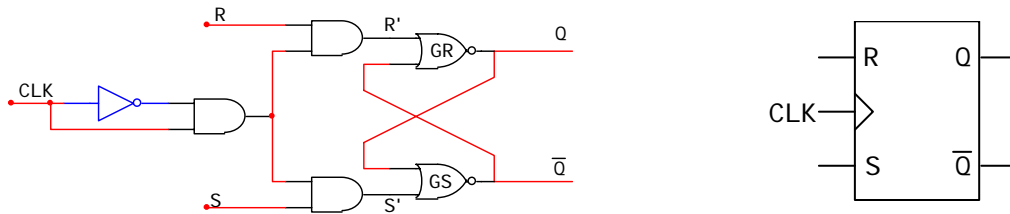

Figure 9. Positive (raising) edge triggered SR flip-flop and related symbol

Similarly the triggering, i.e enabling of the R and S signals may be accomplished during the falling edge with the circuit implementation shown on Figure 10.
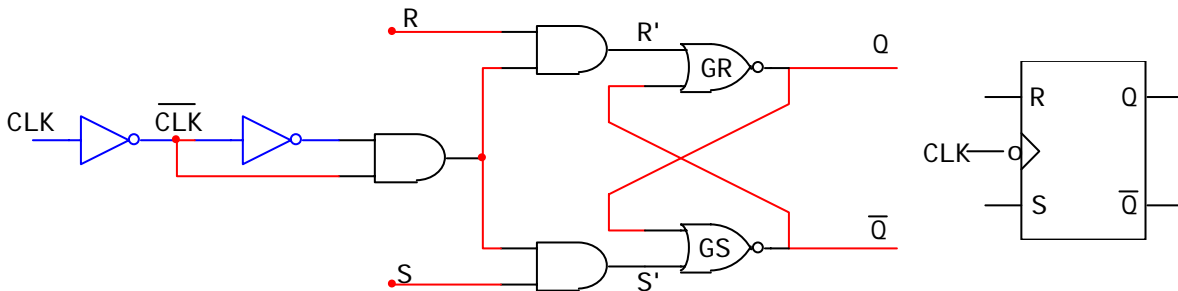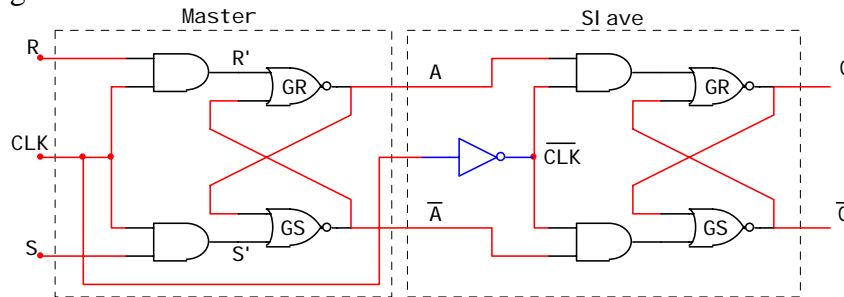

Figure 10. Negative (falling) edge triggered SR flip-flop and related symbol

A variation of the standard SR flip-flop is the Master-Slave SR flip-flop. The corresponding circuit schematic is

This flip-flop is made up of two SR flip-flops connected in series. The clock pulse to the second flip-flop (the Slave) is inverted. A the positive clock transition the output of the Master gets SET but without affecting the Slave. The Slave is in turn triggered during the negative (falling) clock pulse transition.

JK Flip-Flop

The fundamental disadvantage of the SR flip-flop is the indeterminate state of the output when the inputs S and R simultaneously assume the state of 1. A modification of the SR flip-flop, called the JK flip flop removes this problem. The schematic of the JK flip-flop is shown on Figure 11.
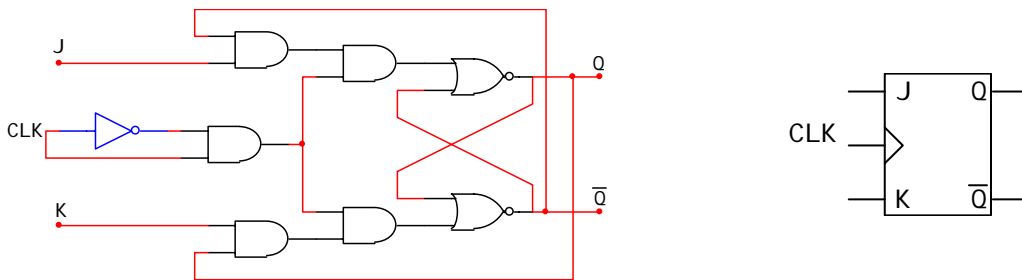

Figure 11. JK flip-flop and its symbol.

The fundamental difference of this device is the feedback paths to the AND gates of the input. The function table for this very useful device is

J	K	CLK	Q_n
0	0	↑	Q_{n-1}
0	1	↑	0
1	0	↑	1
1	1	↑	$\overline{Q}_{n-1}$

For enhanced functionality the JK flip-flop is designed with a PRESET and a CLEAR pin. The schematic and the functional table for this JK flip-flop is shown below.

PRE

J	Q
CLK	
K	$\overline{Q}$

CLR
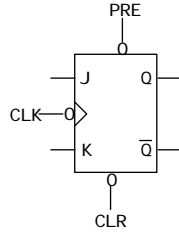
PRE	CLR	CLK	J	K	Q_n	$\overline{Q}_n$	
0	1	X	X	X	1	0	
1	0	X	X	X	0	1	
0	0	X	X	X	Not used		
1	1	⌐↓	0	0	Q_{n-1}	$\overline{Q}_{n-1}$	
1	1	⌐↓	1	0	1	0	
1	1	⌐↓	0	1	0	1	
1	1	⌐↓	1	1	$\overline{Q}_{n-1}$	Q_{n-1}	
1	1	1	X	X	Q_{n-1}	$\overline{Q}_{n-1}$	toggle

Note that the PRESET and CLEAR override J, K and CLK completely. So if PRESET goes to 0, then Q goes to 1; and if CLEAR goes to 0, then Q goes to 0 no matter what J, K and CLK are doing.

If both PRESET and CLEAR are 1, then J, K and CLK can contribute as we have learned above for a JK flip-flop. So at the falling edge the J and K inputs are stored ($Q_n = J$ and $\overline{Q}_n = K$) if J and K are opposites.

 If both J and K are 1 at the low-going edge, then the outputs toggle.

The T Flip-Flop

The T (trigger) flip-flop is a one input flip-flop which may be constructed by simply connecting the inputs of the JK flip-flop together as shown on Figure 12. Equivalently the T flip-flop may be constructed by connecting and setting to 1 the inputs of the JK flip-flop. In this case the output simply toggles after each pulse.
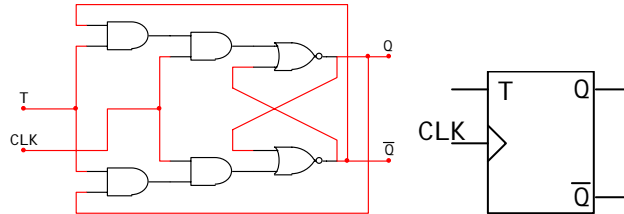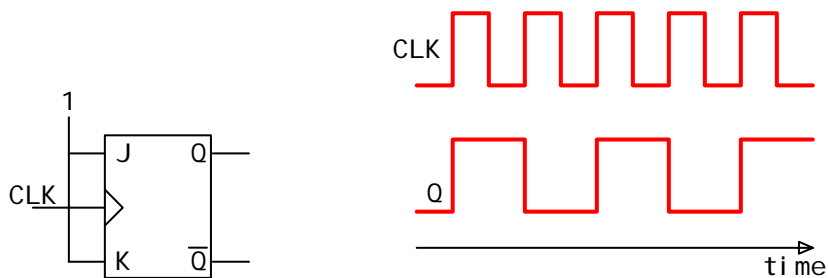

Figure 12. T flip-flop and related symbol.

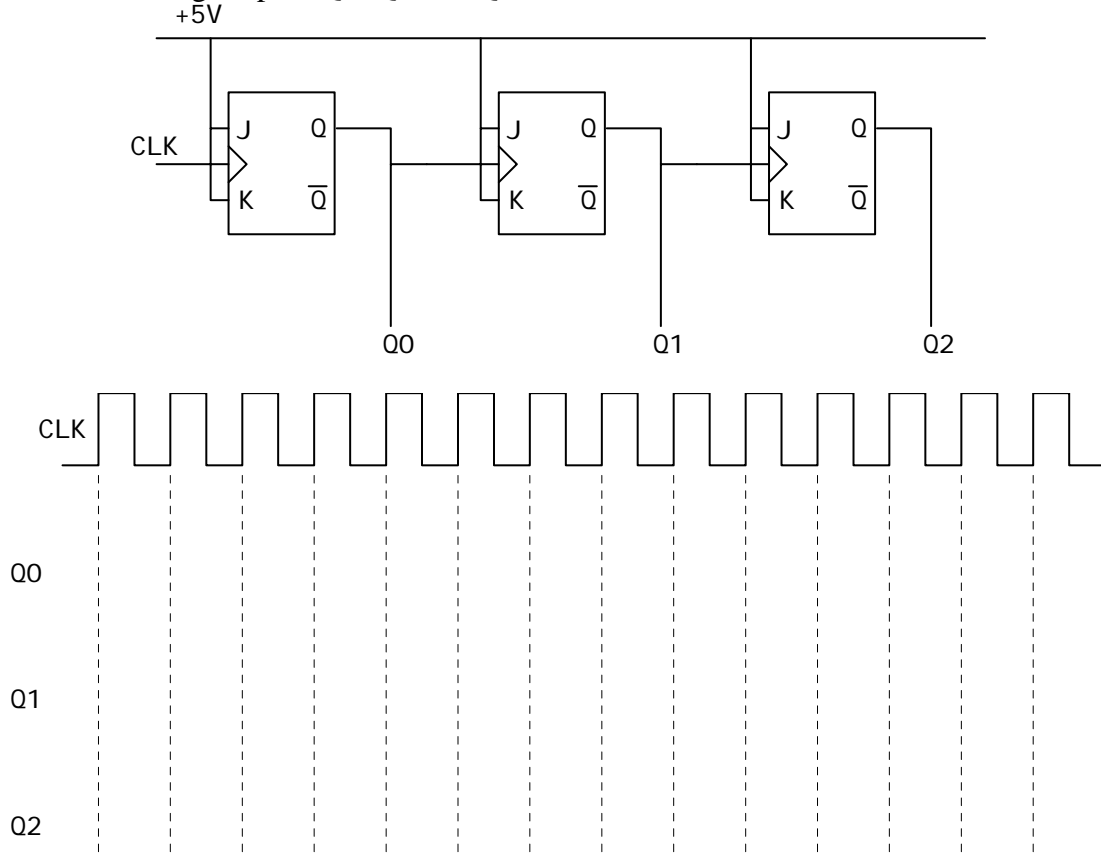The function table of the T flip-flop is

CLK	T	Q_n
⬆	0	Q_{n-1}
⬆	1	$\overline{Q}_{n-1}$

With the JK inputs connected to 1 as shown below the output of the flip-flop results in a signal with half the clock frequency

In the following circuit three JK flip flops (in a T configuration) are arranged as indicated. Draw the resulting outputs Q0, Q1 and Q2

The D Flip-Flop

The D (Delay or Data) flip-flop is a one input flip-flop which may be constructed from the standard SR flip-flop as shown on Figure 13.
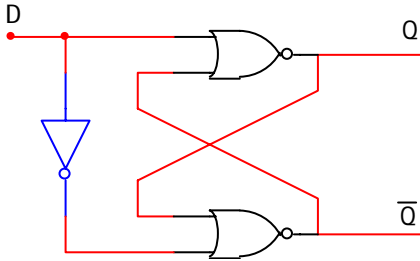

Figure 13. D Flip-Flop

The function table for the D flip-flop is shown below. Note that the conditions R=S=1 and R=S=0 can not exist in this configuration.

D	Q_n	$\overline{Q}_n$	
0	1	0	SET
1	0	1	RESET

A clocked D flip-flop is shown on Figure 14.
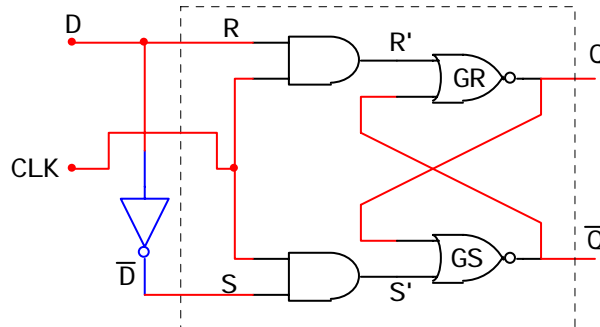
Figure 14. Clocked D flip-flop

The functional characteristics of the clocked D flip-flop are:

CLK	D	Q_n	
⊓	0	1	SET
⊓	1	0	RESET

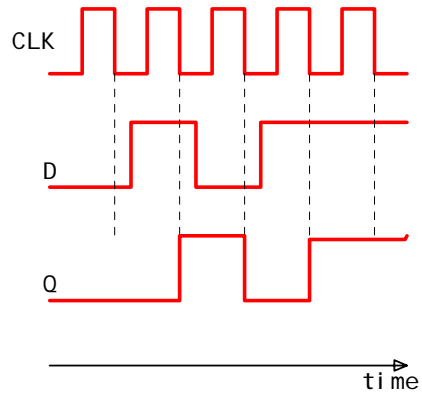A typical timing diagram of the clocked D flip-flop is shown on Figure 14. The dotted vertical lines are the times at which transitions occur

Figure 14. Timing diagram of D Flip-Flop