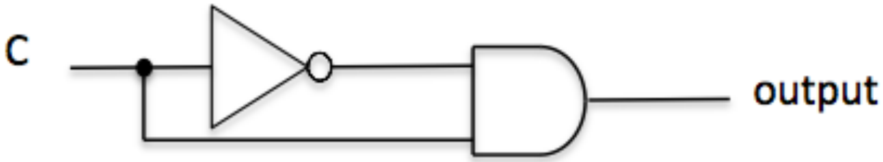
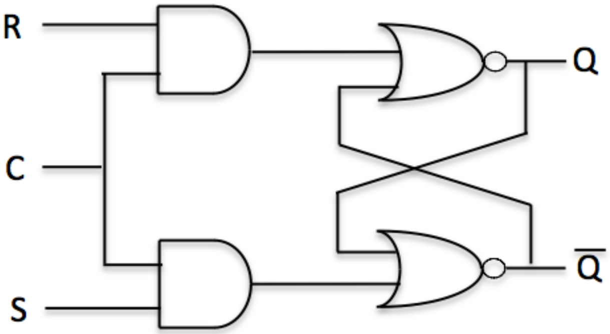
Example Set 5

1. The time interval required for an input signal to travel through a logic gate and appear on the output line is referred to as the gate "propagation delay". A pulse in a signal refers to a change from low to high and back to low. The time required for this low-to-high-to-low transition is the pulse width. Suppose that each gate in the circuit below has a propagation delay of 2ns. If a 4 ns wide positive pulse is applied to the C input, what would be the width of the corresponding pulse output from the circuit?



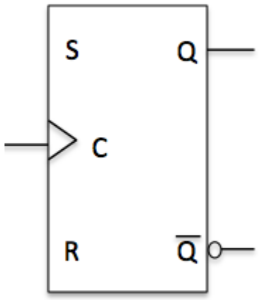
Answer:
C starts out as 0, so the output of the inverter starts at 1. The output of the AND gate starts at 0 AND 1 = 0.
C changes from 0 to 1, it takes 2ns before the output of the inverter to change to 0. During that time both AND gate inputs are 1, so the AND gate will output 1. Once inverter's output changes to 0, the AND gate outputs 0 and remains 0 even when C changes back to 0. Hence the output is a positive pulse and the circuit serves a positive pulse transition detector. Pulses are generated each time C goes from 0 to 1 (i.e., at each positive or leading clock edge).
(Note that such a pulse can be used as the dynamic clock input to cause a flip-flop to change state.)

2. Show an implementation of an SR latch that only changes state while enabled by a clock input.
Answer:



When the C (clock) input is 0, both AND gate output 0, so no state change occurs. When C=1, R as well as S are allowed to pass through the AND gates so the latch is set if S=1 and R=0. It is reset if R=1 and S=0. The output is unpredictable if S and R are both 1.

3. How does the behavior of an SR latch differ from that of an SR flip-flop?
The output from an SR flip-flop only changes at a clock edge while that of an SR latch can change whenever the clock enable is active (i.e., high). The block diagram for an edge triggered SR flip-flop is shown below.

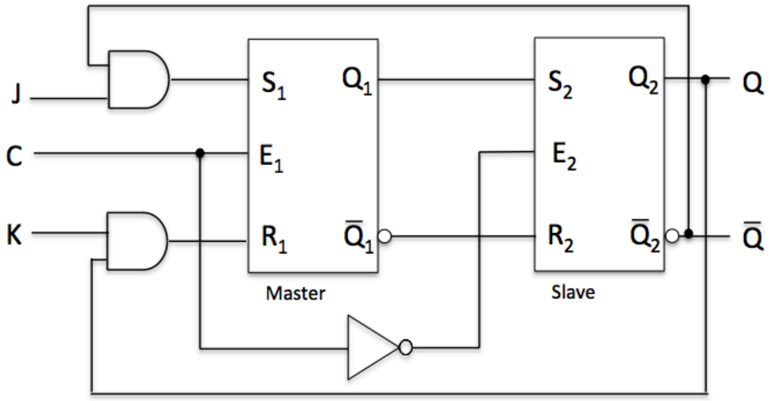


S-R Flip-flop

The trianglar symbol means that C is a dynamic input.

4. The output from an SR flip-flop is unpredictable when both S and R are 1. How can this unusable combination of inputs be made useful.
One way is to place an inverter between the S and R inputs so that they can never both be 1 at the same time. This produces a D flip-flop in which the inputs that were S and R and now derived from a single D input.

An alternative solution is to feed back the Q and Q' outputs to produce what is known as a J-K flip-flop shown below:



The Characteristic table for the J-K flip-flop is shown below:

INPUTS		OUTPUTS		Comments
J	K	Q	Q'	
0	0	Q_0	Q_0'	No change
0	1	0	1	RESET
1	0	1	0	SET
1	1	Q_0'	Q_0	toggle

Each latch reacts to its inputs only when enabled.
Q_0 represents the previous output.

When C is high the master is enabled and responds to the signals from the AND gates. The inverter disables the slave when C is high.
Assume that initially the Q output is 0, so Q'=1.
If J and K are both 0, each AND gate outputs 0 so there is no change in the outputs.
If J =1 and K=0, Q1 is set to 1 (since S1 = 1 AND 1 = 1) and Q1'=0
When C goes back to 0, the slave is enabled and has S2=Q1=1, R2=Q1'=0 so Q2 is set to 1

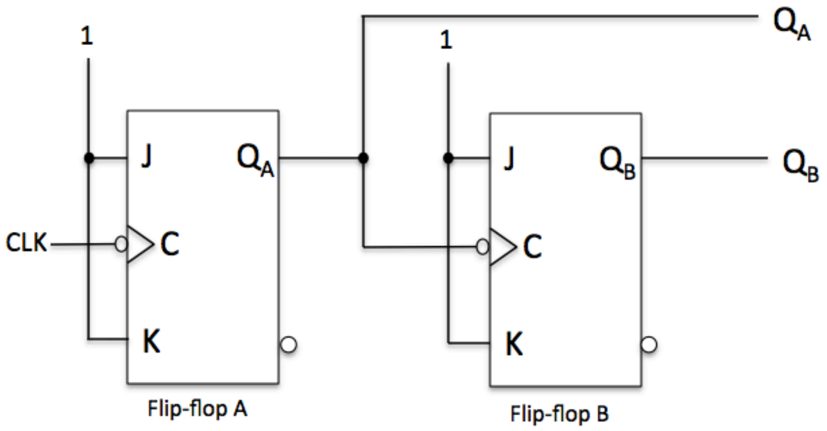
Assume the 1 state, that is, Q=1.
If J=0 and K=1, Q1 is reset to 0 (since R1 = 1 AND 1 = 1) and Q1'=1.
When C goes back to 0, the slave is enabled and has S2=Q1=0, R2=Q1'=1 so Q2 is reset to 0.

Assume the 0 state (Q=0).
If both J=1 and K=1, S1 = J AND Q' = 1 AND 1 = 1; R1= K AND Q = 1 AND 0 = 0
When the clock goes back to 0, the slave is enabled and has S2=1, R2=0 so the output toggles from 0 to 1.

Assume the 1 state(Q=1).
If both J=1 and K=1, S1 = J AND Q' = 1 AND 0 = 1; R1 = K AND Q = 1 AND 1 = 1
When the clock goes back to 0, the slave is enabled and has S2=0, R2=1 so the output toggles from 1 to 0.

5. What is an example of an application of J-K flip-flops with both J and K set to 1?
One application is in the implementation of a binary counter in which each bit in the value is generated by a separate flip-flop. As each bit position increments, it changes from 0 to 1 or from 1 to 0 (i.e., it toggles).
Another is in producing a frequency divider. Since the output only changes on either the leading or trailing clock edge, the rate at which the Q output changes will be one half the rate at which the flip-flop is clocked (i.e., the frequency is divded by 2).

6. Shown below is a block diagram representing two J-K flip-flops.
The J and K inputs for both are tied to a value of 1. The Q output of flip-flop A is used to clock or activate flip-flop B. Assume that the output from each flip-flop is initially 0. Show the outputs Q_A and Q_B for the first 4 clock transitions.



Recall that where lines cross there is no connection unless a dot is present.

The bubble on the clock input denotes that the flip-flop changes on the falling or trailing clock edge when the clock goes from 1 to 0.

Initially Q_B and Q_A = 0 0
At the first clock transition Q_A toggles from 0 to 1 but the second flip-flop does not react since its clock input is Q_A (which went from 0 to 1, not 1 to 0).
Hence Q_B and Q_A = 0 1.

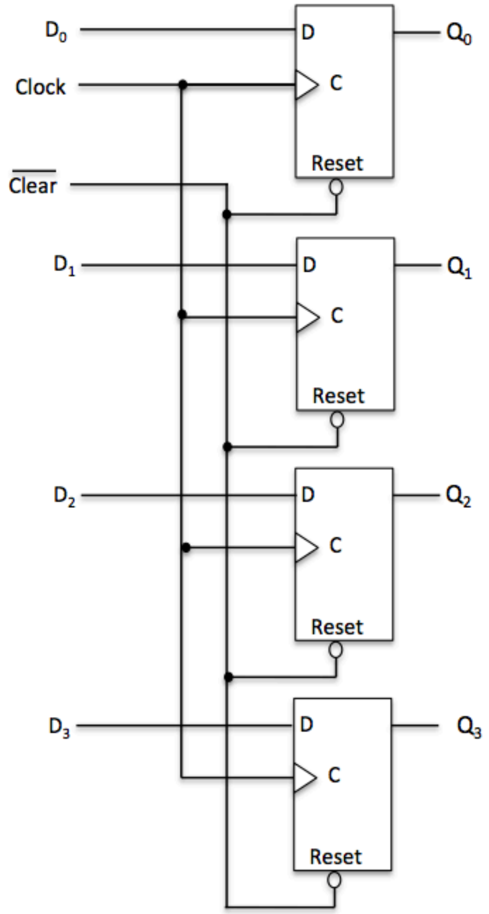
At the second clock transition Q_A toggles from 1 to 0 and causes flip-flop B to toggle Q_B from 0 to 1. Hence Q_B and Q_A = 1 0.

At the third clock transition Q_A toggles from 0 to 1 but the second flip-flop does not react. Hence Q_B and Q_A = 1 1.

At the 4th clock transition Q_A toggles from 1 to 0 and causes flip-flop B to toggle Q_B from 1 to 0. Hence Q_B and Q_A = 0 0.
Therefore the behavior is that of a 2-bit binary counter that increments by 1 for each clock transition and rolls back to 00 from 11. More flip-flops could be included to create a binary counter with the desired number of bits.

7. Show how a 4-bit register could be implemented using D flip-flops.

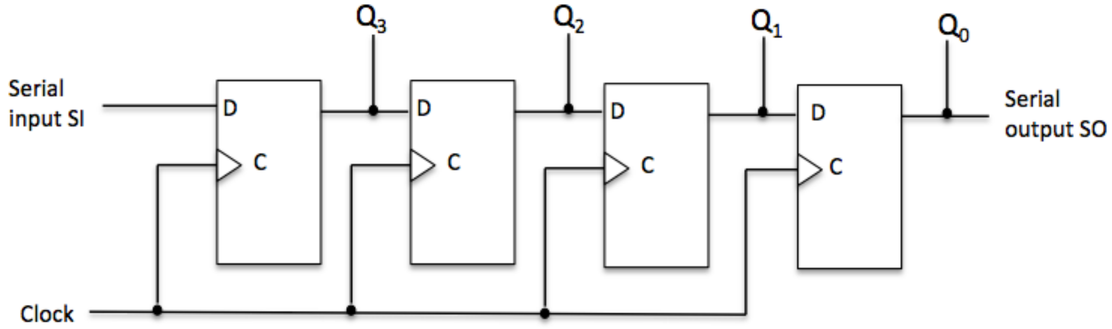
Answer:
One implementation is shown below:



The Clear signal when set to 0 clears the 4-bit register (i.e., to reset the register). When the clock makes a positive transition (from 0 to 1) each of the 4 data inputs is stored into a separate D flip-flop. The Q outputs show the current contents of the register.

8. Show how D flip-flops can be used to create a 4-bit serial shift register.
Answer:

One simple implementation is shown below:



This is a chain of flip-flops in cascade, with the output of one flip-flop connected to the input of the next flip-flop on the right. All flip-flops are controlled by a common clock pulse. For each clock pulse the current serial input SI is loaded into the leftmost flip-flop; the output of each flip-flop is loaded into its neighbor on the right and the output of the rightmost flip-flop appears on SO (the serial out). Hence the behavior is that of a right shift. After four clock pulses, the register is filled and the 4 bits contained in the register can be read as $Q_3Q_2Q_1Q_0$. The original 4 bits in the register will have been serially shifted out (i.e., one at a time) through SO.

9 How can a J-K flip-flop be transformed into a D flip-flop.

This can be done by inserting an inverter between the J and K inputs. With this change the output will be set to 1 when J=1 and the output will be reset to 0 when J=0 (since K=J' = 1). So the flip-flop behaves like D flip-flop with J as the D input.

10. A T flip-flop (toggle flip-flop) is one that has a single input T and complements its output each time that T makes a transition from 0 to 1. How can a J-K flip-flop be transformed into a T flip-flop?

If the J and K inputs are connected together the desired behavior will be obtained. This makes J and K equal. When the input is 1, J=1 and K=1 which causes the output to toggle. If the input is 0, J=0 and K=0 so there is no change in the output.