Digital Logic Design
(EL-227)
LABORATORY MANUAL
Spring-2021

LAB 11
Flip-Flop

Lab Session 11: Flip-Flop

OBJECTIVES:

- Design D Flip-Flop, JK Flip Flop and verify their functional characteristics
- Define 'Set', 'Reset', 'Preset' and 'Clear' concepts in Flip-Flops
- Extract information of flip-flop from its timing diagram

APPARATUS: Logic trainer, Logic probe

COMPONENTS: ICs 74LS08, 74LS32, 74LS04, 74LS86, 74LS02

Flip-Flop

In digital circuits, a ***flip-flop*** is a term referring to an electronic circuit (a bistable multivibrator) that has two stable states 0 and 1. A flip-flop is a memory device and thus will store a value while power remains in the circuit. Storage elements that operate with signal levels (rather than signal transitions) are referred to as latches; those controlled by a clock transition are flip-flops. Latches are said to be level sensitive devices; flip-flops are edge-sensitive devices. Both are the simplest examples if ***sequential systems***.

Symbol for Flip-Flop

A general type of symbol used for a flip-flop is shown in Figure.1. It has multiple inputs, and two outputs labeled Q & Q' which are the inverse of each other. The Q output is called the ***normal output*** and Q' is the ***inverted output***. If we say that the flip-flop is in the High-state, we mean that Q = 1 & if we say that a flip-flop is in Low-state we mean that Q = 0 of course, the Q' state will always be the inverse of Q.

The high state i.e. Q = 1 & Q' = 0 of a Flip – flop is also known as **SET** state. The low state i.e. Q = 0 & Q' = 1 is then called **RESET** state. The reset state is also called the clear state.

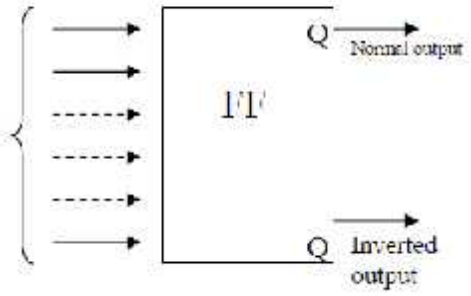
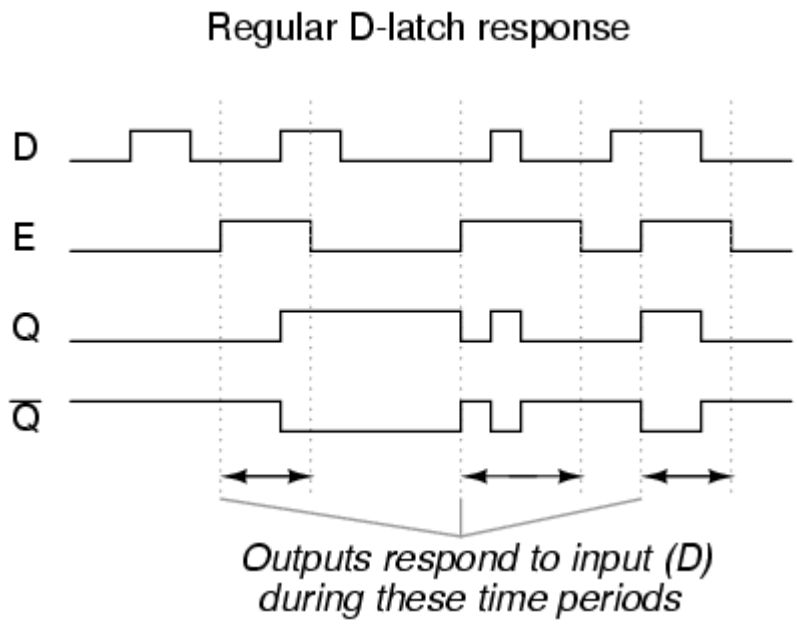
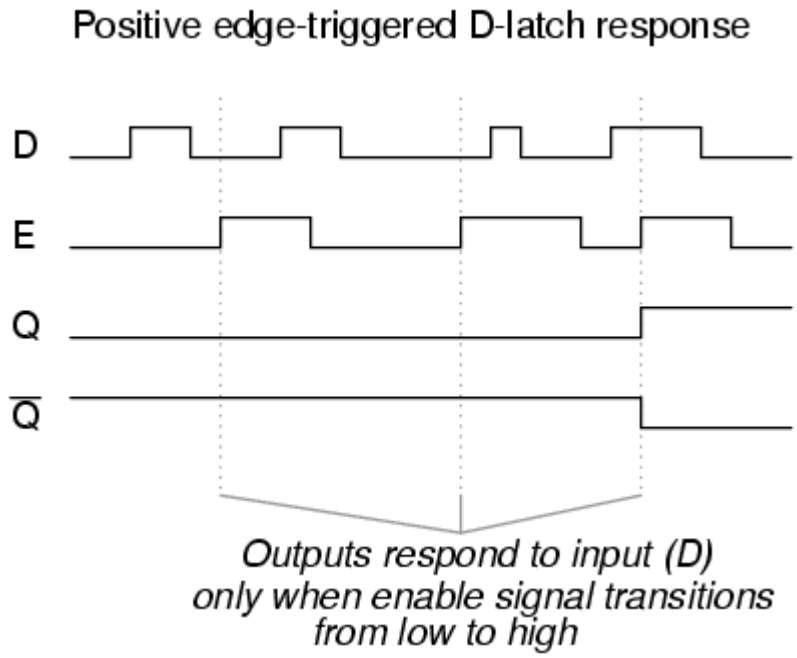


Figure.1 Symbol for Flip-Flop

Edge Triggered D-Flip-Flop

One method of enabling a multivibrator circuit is called *edge triggering*, where the circuit's data inputs have control only during the time that the enable input is *transitioning* from one state to another. Let's compare timing diagrams for a normal D latch versus one that is edge-triggered:

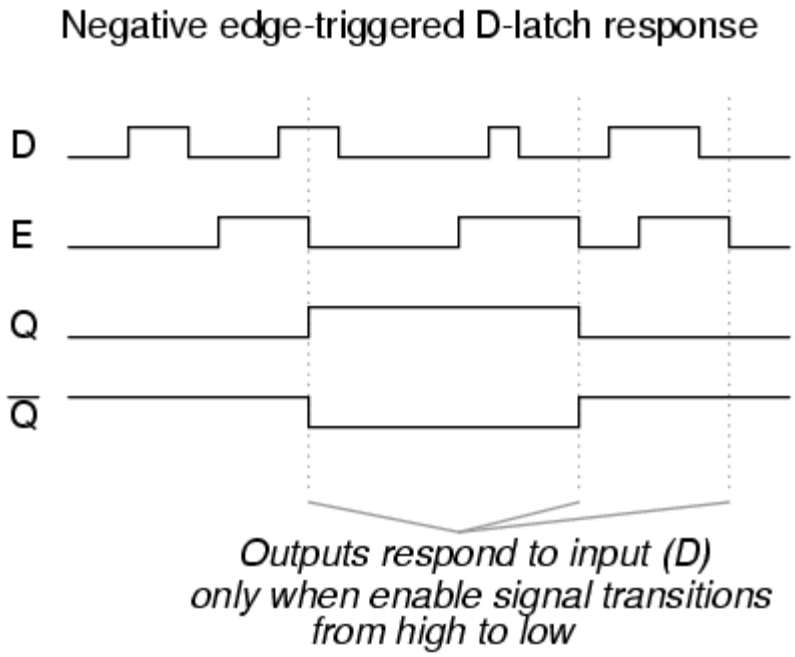




In the first timing diagram, the outputs respond to input D whenever the enable (E) input is high, for however long it remains high. When the enable signal falls back to a low state, the circuit remains latched.

In the second timing diagram, we note a distinctly different response in the circuit output(s): it only responds to the D input during that brief moment of time when the enable signal *changes*, or *transitions*, from low to high. This is known as *positive* edge-triggering.

There is such a thing as *negative* edge triggering as well, and it produces the following response to the same input signals:



Whenever we enable a multivibrator circuit on the transitional edge of a square-wave enable signal, we call it a *flip-flop* instead of a *latch*. Consequently, and edge-triggered S-R circuit is more properly known as an S-R flip-flop, and an edge-triggered D circuit as a D flip-flop

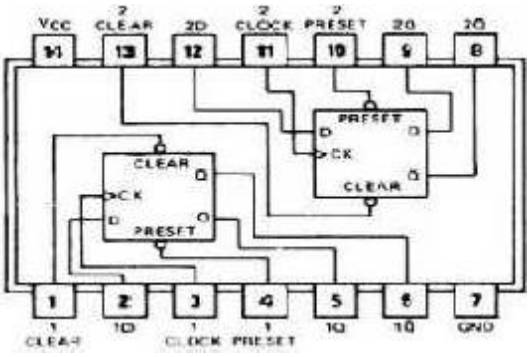


Figure.2 IC 74LS74PinConfiguration

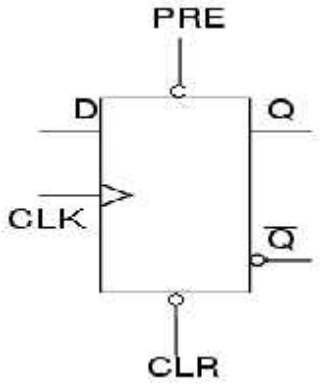


Figure.3 Graphic symbol Dflip-flop

Function Table of D Flip-Flop

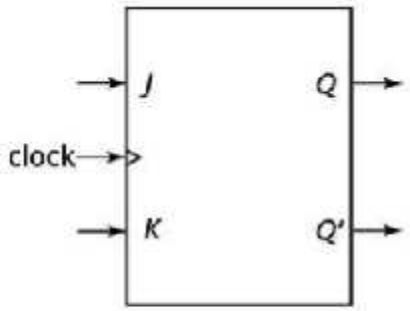
INPUTS				OUTPUTS	
$\overline{PR}$	$\overline{CLR}$	CLK	D	Q	$\overline{Q}$
0	1	X	X	1	0
1	0	X	X	0	1
0	0	X	X	X	X
1	1	↑	1	1	0
1	1	↑	0	0	1
1	1	0	X	Q_0	$\overline{Q}_0$

The preset and clear input are active-low, because there are an inverting bubble at that input lead on the block symbol, just like the negative edge-trigger clock inputs.
When the preset input is activated, the flip-flop will be reset (Q=0, not-Q=1) regardless of any of the synchronous inputs or the clock. When the clear input is activated, the flip-flop will be set (Q=1, not-Q=0), regardless of any of the synchronous inputs or the clock. So, what happens if both preset and clear inputs are not activated (both of them 0) X,X state: we get an invalid state on the output.

When both preset and clear inputs are activated then the flip flop will work normally.

1. JK flip-flop

The JK type flip-flop consists of two data inputs: J and K, and one clock input. There are again two outputs Q and Q' (where Q' is the reverse of Q).

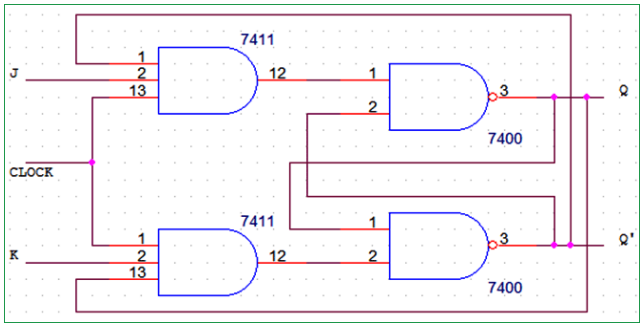


J	K	clock	Q
X	X	not ↑	Q_0
0	0	↑	Q_0
0	1	↑	0
1	0	↑	1
1	1	↑	Q_0'

Lab Tasks

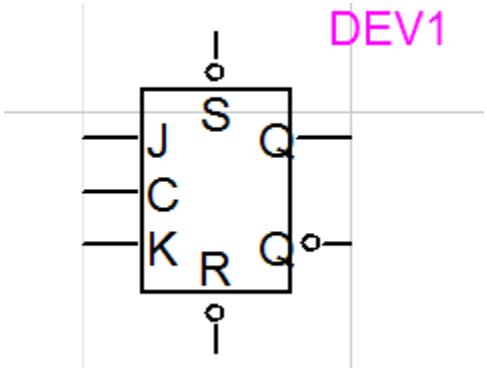
Lab Task #1

Implement JK flip- flop by using logic gates.



Lab Task #2

Implement JK flip- flop IC present in logic gates.



Lab Task #3

Implement D Latch by using logic gates and draw the TT according to inputs present in the image.