© 2014 IJIRT | Volume 1 Issue 6 | ISSN : 2349-6002

Flip-flop and Its Applications

Vikash Rai, Tapish Chauhan
Student, Dronacharya College Of Engineering, Gurgaon

Abstract- **The objective of this paper is comprehensive study related to flip-flop and its application. Flip-flops are the building blocks of any sequential logic circuits. Today the word *latch* is mainly used for simple *transparent* storage elements, while slightly more advanced *non-transparent* (or *clocked*) devices are described as flip-flops. Informally, as this distinction is quite new, the two words are sometimes used interchangeably. Flip-flops are the first stage in sequential logic designs which incorporates memory (storage of previous states). Flip-flops can be either simple (transparent or opaque) or clocked (synchronous or edge-triggered). This paper represents the information related to history, implementation, types and applications of the flip-flop.**

I. INTRODUCTION

A digital computer needs devices which can store information. A **flip-flop** is a circuit that has two stable states and can be used to store state information. A flip flop is a binary storage device. It can store binary bit either 0 or 1. It has two stable states HIGH and LOW i.e. 1 and 0. A flip-flop is usually controlled by control signals that can include a clock signal. The outputs usually include the complement as well as the normal output. Flip-flops can be either simple (transparent or opaque) or clocked (synchronous or edge-triggered). Although the term flip-flop has historically referred generically to both simple and clocked circuits, in modern usage it is common to reserve the term *flip-flop* exclusively for discussing clocked circuits; the simple ones are commonly called *latches.* The basic function of flip flop is to store data. They can be used to keep a record or what value of variable (input, output or intermediate). Flip flop are also used to exercise control over the functionality of a digital circuit i.e. change the operation of a circuit depending on the state of one or more flip flops.

II. HISTORY

The first electronic flip-flop was invented in 1918 by William Eccles and F. W. Jordan. It was initially called the *Eccles–Jordan trigger circuit* and consisted of two active elements (vacuum tubes). Such circuits and their transistorized versions were common in computers even after the introduction of integrated circuits, though flip-flops made from logic gates are also common now. Early flip-flops were known variously as trigger circuits or multivibrators. According to P. L. Lindley, a JPL engineer, the flip-flop types discussed below (RS, D, T, JK) were first discussed in a 1954 UCLA course on computer design by Montgomery Phister, and then appeared in his book *Logical Design of Digital Computers.*

III. IMPLEMENTATION

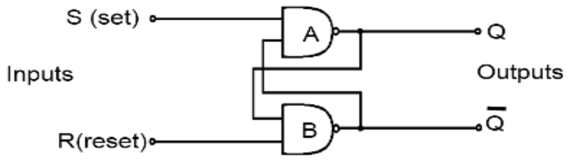
Flip-flops can be either simple (transparent or asynchronous) or clocked (synchronous); the transparent ones are commonly called latches. The word *latch* is mainly used for storage elements, while clocked devices are described as *flip-flops*. Simple flip-flops can be built around a pair of cross-coupled inverting elements: vacuum tubes, bipolar transistors, field effect transistors, inverters, and inverting logic gates have all been used in practical circuits. Clocked devices are specially designed for synchronous systems; such devices ignore their inputs except at the transition of a dedicated clock signal (known as clocking, pulsing). Clocking causes the flip-flop to either change or retain its output signal based upon the values of the input signals at the transition. Some flip-flops change output on the rising edge of the clock, others on the falling edge. Since the elementary amplifying stages are inverting, two stages can be connected in succession (as a cascade) to form the needed non-inverting amplifier.

IV. TYPES OF FLIP-FLOP

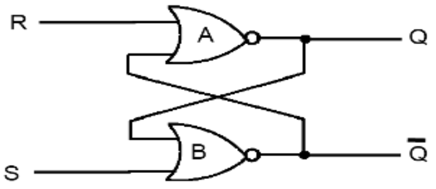
S-R FLIP-FLOP

The R-S (Reset Set) flip flop is the simplest flip flop of all and easiest to understand. The SR flip-flop can also have a clock input for a level driven circuit as opposed to a pulse driven circuit. Besides the CLOCK input, an SR flip-flop has two inputs, labeled SET and RESET. If the SET input is HIGH when the clock is triggered, the Q output goes HIGH. If the RESET input is HIGH when the clock is triggered, the Q output goes LOW.

Note that in an SR flip-flop, the SET and RESET inputs shouldn't both be HIGH when the clock is triggered. This is considered an invalid input condition, and the resulting output isn't predictable if this condition occurs. Such flip flop can be made simply by cross coupling two inverting gates either NAND or NOR gate could be used.



(a) Latch Flip Flop NAND Gate



(b) RS Latch Flip Flop NOR Gate

Table 1: The truth table for the NAND R-S flip flop

Initial Conditions	Inputs (Pulsed)		Final Output	
Q	S	R	Q	$\bar{Q}$
1	0	0	indeterminate	
1	0	1	1	0
1	1	0	0	1
1	1	1	1	0
0	0	0	indeterminate	
0	0	1	1	0
0	1	0	0	1
0	1	1	0	1

Table 2: Simple NAND R-S Flip Flop Truth Table

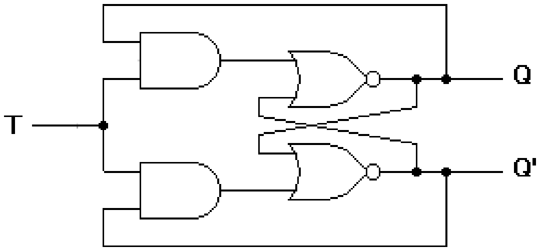
S	R	Q
0	0	indeterminate
0	1	Set (1)
1	0	Reset(0)
1	1	No Change

Table 3: NOR Gate R-S Flip Flop Truth Table

S	R	Q
0	0	No Change
0	1	Reset (0)
1	0	Set (1)
1	1	Indeterminate

T Flip-Flop

This is simply a JK flip-flop whose output alternates between HIGH and LOW with each clock pulse. Toggles are widely used in logic circuits because they can be combined to form counting circuits that count the number of clock pulses received. Toggle means to change in the previous stage i.e. switch to opposite state. It can be constructed from clocked RS flip flop be incorporating feedback from output to input.



T Flip Flop

Truth table for T Flip Flop

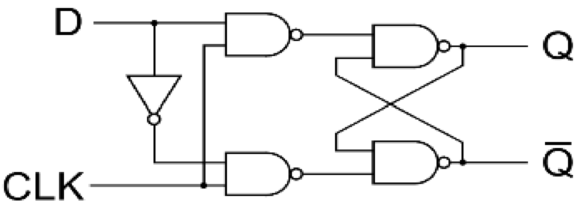
Q_n	T	$Q_n + 1$
0	0	0
0	1	1
1	0	1
1	1	0

Excitation table for T Flip Flop

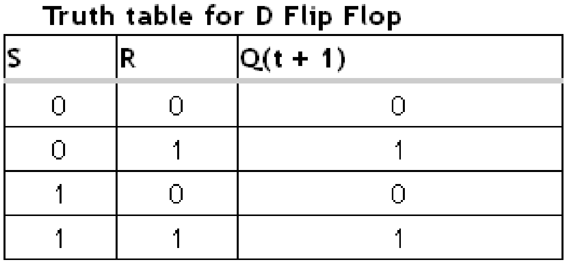
T	Q
0	Q_n
1	$\bar{Q}_n$

D Flip-Flop

Has just one input in addition to the CLOCK input. This input is called the DATA input. When the clock is triggered, the Q output is matched to the DATA input. Thus, if the DATA input is HIGH, the Q output goes HIGH, and if the DATA input is LOW, the Q output goes LOW.

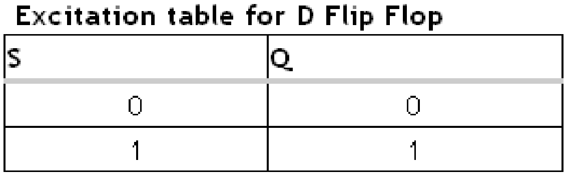


D Flip Flop



Truth table for D Flip Flop

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

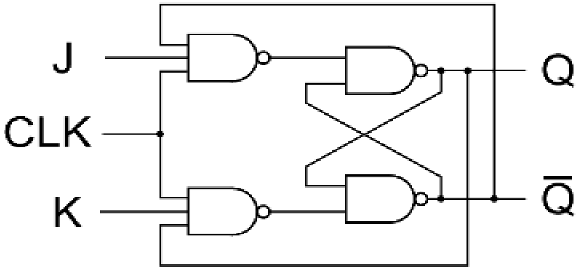


Excitation table for D Flip Flop

S	Q
0	0
1	1

J-K Flip-Flop

A JK flip-flop has two inputs, labeled *J* and *K.* The J input corresponds to the SET input in an SR flip-flop, and the K input corresponds to the RESET input. The difference between a JK flip-flop and an SR flip-flop is that in a JK flip-flop, both inputs can be HIGH. When both the J and K inputs are HIGH, the Q output is *toggled*, which means that the output alternates between HIGH and LOW.



JK Flip Flop

When J = 0 and K = 0, These J and K inputs disable the AND gates, therefore clock pulse have no effect on the flip flop. In other words, Q returns it last value.

When J = 0 and K = 1,

The upper AND gate is disabled the lower AND gate is enabled if Q is 1 therefore, flip flop will be reset (Q = 0 , $\overline{Q}$ =1)if not already in that state.

When J = 1 and K = 0

The lower AND gate is disabled and the upper AND gate is enabled if $\overline{Q}$ is at 1, As a result we will be able to set the flip flop (Q = 1, $\overline{Q}$ = 0) if not already set

When J = 1 and K = 1

If Q = 0 the lower AND gate is disabled the upper AND gate is enabled. This will set the flip flop and hence Q will be 1. On the other hand if Q = 1, the lower AND gate is enabled and flip flop will be reset and hence Q will be 0. In other words , when J and K are both high, the clock pulses cause the JK flip flop to toggle.

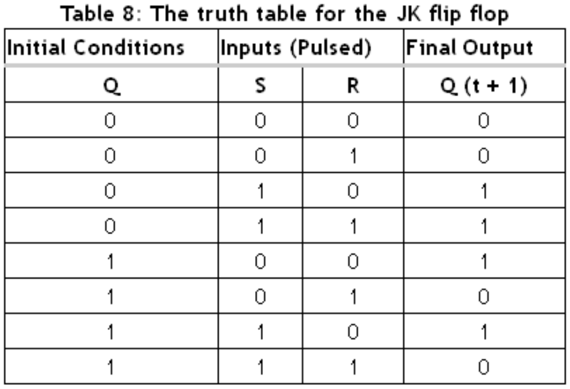


Table 8: The truth table for the JK flip flop

Initial Conditions	Inputs (Pulsed)		Final Output
Q	S	R	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

V. APPLICATIONS OF FLIP-FLOP

a. Data Storage:

A flip flop store one bit at a time in digital circuit. In order to store more than one bit flip flop can be connected in series and parallel called registers. A register is simply a data storage device for a number of bits in which each flip flop store one bit of information (0 or 1). Thus a 4 bit register consists of 4 individual flip flops, each able to store one bit of information at a time.

b. Data Transfer:

Flip flops can also be used extensively to transfer the data. For this purpose shift register is used. A shift register is a register which is able to shift or transfer it content within itself without changing the order of the bits. It may be designed to shift or transfer data either left or right. The data is shifted or transferred one bit at a time, when a clock pulse is applied. The shift register can be used for temporary storage of data. The shift register is used for multiplication and division where bit shifting is required. The shift register can be built using RS, JK or D flip flops.

c. Counter:

Another major application of flip flops is a digital counter. It is used to count pulses or events and it can be made by connecting a series of flip flops. Counter can count up to 2^n. Where **n** is the number of flip flops.

d. Frequency Division:

Flip flops can divide the frequency of periodic waveform. When a pulse wave is used to toggle an flip flop, the output frequency becomes one half the input frequency. The output of each flip flop is half the frequency of an input. If the input frequency is 160 KHz then output of each flip flop would be so after first flip flop, 40 after second flip flop and 20 after third flip flop.

Input frequency 160 KHz
Frequency of first flip flop 80 KHz
Frequency of 2nd flips flop 40 KHz
Frequency of 3rd flips flop 20 KHz

REFERENCES

[1] Volnei A. Pedroni (2008). *Digital electronics and design with VHDL*. Morgan Kaufmann. p. 329. ISBN 978-0-12-374270-4.

[2] Latches and Flip Flops (EE 42/100 Lecture 24 from Berkeley).

[3] http://www.daenotes.com/

[4] William Henry Eccles and Frank Wilfred Jordan, "Improvements in ionic relays" British patent number: GB 148582 (filed: 21 June 1918; published: 5 August 1920).

[5] http://blog.oureducation.in/

[6] W. H. Eccles and F. W. Jordan (19 September 1919) "A trigger relay utilizing three-electrode thermionic vacuum tubes," *The Electrician*, vol. 83, page 298. Reprinted in: *Radio Review*, vol. 1, no. 3, pages 143–146 (December 1919).

[7] http://www.princeton.edu/

[8] Emerson W. Pugh; Lyle R. Johnson; John H. Palmer (1991). *IBM's 360 and early 370 systems*. MIT Press. p. 10. ISBN 978-0-262-16123-7.

[9] http://www.ee.surrey.ac.uk/

[10] Earl D. Gates (2000-12-01). *Introduction to electronics* (4th ed.). Delmar Thomson (Cengage) Learning. p. 299. ISBN 978-0-7668-1698-5.

[11] By Doug Lowe from Electronics All-in-One For Dummies, http://www.dummies.com/

[12] Max Fogiel; You-Liang Gu (1998). *The Electronics problem solver, Volume 1* (revised ed.). Research & Education Assoc. p. 1223. ISBN 978-0-87891-543-9.

[13] P. L. Lindley, Aug. 1968, EDN (magazine), (letter dated June 13, 1968).

[14] Montgomery Phister (1958). *Logical Design of Digital Computers.*. Wiley. p. 128.

[15] http://en.wikipedia.org/