**Objective:**

* Understanding the internal logic of J-K Flip-flops and implement one using basic logic gates
* Understand the relationship between J-K, T and D flip-flops and observe the characteristics of all three
* Implementing a shift register using D flip-flops and analyze its operation
* Learning about the concept of states in digital logic and how flip-flop circuits can e used to store state information.

**List of Equipment**

* 1 \* IC 7402 2-input NOR gate
* 2 \* IC 7411 3-input AND gate
* 1 \* IC 7404 Hex inverter
* 1 \* IC 7474 D flip-flop
* Trainer board
* Wires
* IC Extractor

**Theory:**

Flip-Flops:

A flip-flop is a binary storage device that has two stable states and is capable of storing one bit of information. In a stable state, the output of a flip-flop is either 0 or 1. The output can only change when a clock pulse is supplied to the flip-flop. The value that is stored in a flip-flop when the clock pulse occurs is determined by the inputs to the flip-flop at that time or the values presently stored in the flip-flop (or both). The new value is stored (i.e., the flip-flop is updated) when a pulse of the clock signal occurs.

https://pixel-geo.prfct.co/cs/?partnerId=mrinhttps://pixel-geo.prfct.co/cs/?partnerId=twtrhttps://pixel-geo.prfct.co/cs/?partnerId=yahhttps://pixel-geo.prfct.co/cs/?partnerId=opxhttps://pixel-geo.prfct.co/cs/?partnerId=pubhttps://pixel-geo.prfct.co/cs/?partnerId=rbcnhttps://pixel-geo.prfct.co/cs/?partnerId=goohttps://pixel-geo.prfct.co/seg/?add=2863386,8013780,4696903,8013781,2988646,8013792,2990199,8013834&source=js_tag&a_id=45525https://secure.adnxs.com/seg?t=2&add=8013834https://secure.adnxs.com/seg?t=2&add=2990199https://secure.adnxs.com/seg?t=2&add=8013792

Latches vs Flip-Flops:

Latches and flip – flops are both 1 – bit binary data storage devices. The main difference between a latch and a flip – flop is the triggering mechanism. Latches are transparent when enabled ,whereas flip – flops are dependent on the transition of the clock signal i.e. either positive edge or negative edge.

The modern usage of the term flip – flop is reserved to clocked devices and term latch is to describe much simpler devices. Some of the other differences between latches and flip – flops are listed in below table.

### Types of flip flops

Based on their operations, flip flops are basically 4 types. They are

1. R-S flip flop
2. D flip flop
3. J-K flip flop
4. T flip flop

### **S-R Flip Flop**

The S-R flip-flop is basic flip-flop among all the flip-flops. All the other flip flops are developed after SR-flip-flop.

SR flip flop is represented as shown below:

S-R stands for SET and RESET. This can also be called RS flip-flop. Difference is RS is inverted SR flip-flop. Any flip flop can be build using logic gates. NAND and NOR gates were used as they are universal gates.

Here is the SR flip-flop using NAND gates.

#### **Working**

From the above truth table it is clear that SR flip flop will be set or reset for four conditions.

1. For last condition it will be in invalid state.
2. SR Flip-flop will be set when S=1 and R=0, if S=1 and R=1 then previous state is remembered by the flip flop.
3. Flip-flop will be reset when S=0 and R=1, if S=1 and R=1, then it will remember the previous state.
4. But when both the inputs are zeros, SR Flip flop will be in an uncertain state where both Q and Q’ will be same. This is not same allowed..

This is indeterminate state is avoided by adding gates extra gates to the existing flip flop. This is  called clocked or gated SR Flip flop. This produces the output only for the  High clock pulse.

The circuit of a clocked SR flip – flop using NAND gates is shown below.

### D flip flop

In the SR flip flop an uncertain state occurred. This can be avoided by using D flip flop. Here D stands for “Data”.  
It is constructed from SR flip flop. The two inputs (S &R) of the clocked SR flip flop are connected to an inverter.

It is one of the most widely used flip – flops. It has a clock signal (Clk) as one input and Data (D) as other. There are two outputs and these outputs are complement to each other.

The symbol of D flip – flop is shown below.

#### Working

* D flip flop will work depending on the clock signal.
* When the clock is low there will be no change in the output of the flip flop i.e. it remembers the previous state.
* When the clock signal is high and if it receives any data on its data pin, it Changes the state of output.
* When data is high Q reset to 0,while Q is set to 0 if data is low.

A master slave D flip flop can be constructed using D-flip flop.

**J-K Flip Flop**

JK flip – flop is named after Jack Kilby, an electrical engineer who invented IC.

A JK flip – flop is a modification of SR flip – flop. In this the J input is similar to the set input of SR flip – flop and the K input is similar to the reset input of SR flip – flop. The condition J = K = 1 which is not allowed in SR flip – flop (S = R = 1) is interpreted as a toggle command.

The JK flip flop has

* Two data inputs J and K.
* One clock signal input (CLK).
* Two outputs Q and Q’.

The symbol of a JK flip – flop is shown below.

#### Working

* When J is low and K is low, then Q returns its previous state value i.e. it holds the current state.
* When J is low and K is high, then flip – flop will be in reset state i.e. Q = 0, Q’ =1.
* When J is high and K is low then flip – flop will be in set state i.e. Q = 1, Q’ =0.
* When J is high and K is high then flip – flop will be in Toggle state or flip state. This means that the output will complement to the previous state value.

**T Flip Flop**

T flip flop is also known as “Toggle Flip – flop”. Toggle is to change the output to complement of the previous state in the presence of clock input signal.

The T flip flop has

* T input.
* One clock signal input (CLK).
* Two outputs Q and Q’.

The symbol of a T flip – flop is shown below.

We can construct a T flip – flop by using any other flip – flops.

* SR flip – flop: By connecting the feedback of outputs of SR flip – flop to the inputs (S & R).
* D flip – flop: Connecting the Q’ to its Data input of D flip – flop as feedback path.
* J K flip – flop: By combing the J & K inputs of JK flip – flop, to make as single input, we can design the T flip – flop.

#### **Working:**

The operation of the T flip flop is explained below.

When the T input is low, then the next sate of the T flip – flop is same as the present state i.e. it holds the current state.

* T = 0 and present state = 0 then the next state = 0.
* T = 0 and present state = 1 then the next state = 1.

When the T input is high, then the next sate of the T flip – flop is toggled i.e. it is same as the complement of present state on clock transition.

* T = 1 and present state = 0 then the next state = 1.
* T = 1 and present state = 1 then the next state = 0.

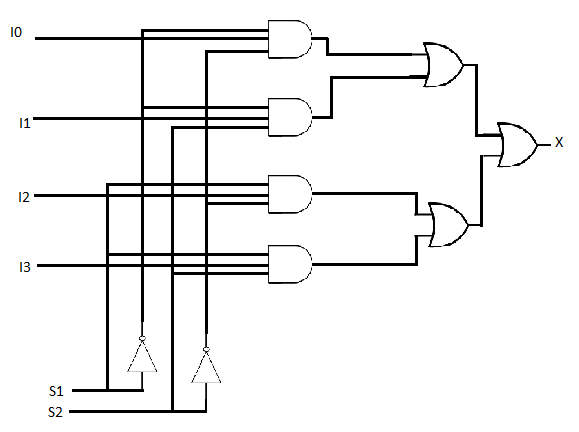
**Registers:**

Flip-flop is a 1 bit memory cell which can be used for storing the digital data. To increase the storage capacity in terms of number of bits, we have to use a group of flip-flop. Such a group of flip-flop is known as a **Register**. The **n-bit register** will consist of **n** number of flip-flop and it is capable of storing an **n-bit** word.

The binary data in a register can be moved within the register from one flip-flop to another. The registers that allow such data transfers are called as **shift registers**. There are four mode of operations of a shift register.

* Serial Input Serial Output
* Serial Input Parallel Output
* Parallel Input Serial Output
* Parallel Input Parallel Output

**Circuit Diagram:**



**Data/Truth table:**

|  |  |  |  |
| --- | --- | --- | --- |
| **J** | **K** | **Q** | **Q’** |
| 1 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

Table: J-K Flip Flop using AND and NOR gates

|  |  |
| --- | --- |
| **T** | **Q** |
| 0 | 1 |
| 0 | 1 |

|  |  |
| --- | --- |
| **D** | **Q** |
| 0 | 0 |
| 1 | 1 |

Table: T and D Flip-flops using J-K flip-flops

|  |  |  |
| --- | --- | --- |
| **States** | **Input** | **Output** |
| Initial State | X | XXXX |
| T1 | 1 | 1XXX |
| T2 | 0 | 01XX |
| T3 | 1 | 101X |
| T4 | 0 | 0101 |

Table: Right Shift register using D flip-flops

**Discussion:**

Today we learnt about the MUX and Decoders. At first we created a MUX using basic gates. While using the gates we didn’t get expected results. Then we changed our trainer board and then checked it again. But we got the same result. After that we checked every IC separately. Then we came to know that the AND gate IC had problem. Then we changed the IC. Then our result came as expected.

Next, we had to create a circuit using a MUX IC. In this case our IC was working properly. So, we did not face any problems. And this time we finished the circuit very quickly.

After that, we build a circuit using decoder. As decoders cannot be checked before implementing the circuit, we made the whole circuit to see whether it works properly or not. The IC worked properly. But couple of the output lights didn’t blow. But we managed with that. As all others bulbs worked properly and gave us the expected value we considered as the circuit is working properly.

Finally, after all our work has done cleared our table and rearranged our chairs. Then it was the end of the lab that day.

* 1. Simulate the circuit you built for the JK flip-Flop using Logisim. Include a screenshot of the circuit with report.
  2. Briefly Explain the difference between T and D Flip-Flops.

Ans :

The TFF and DFF are very similar, T stands for Toggle, and D stands for Data. The differences here are that a TFF will toggle its output every time it receives a signal, and a DFF will change its output based on what is on the data line when it is clocked.

One way to tell them apart is by what they have, aTFF has an input and an output and will toggle no matter what a DFF has an input, an output, and a clock, and will only change its output when the clock is on (it also changes its output to whatever the input is, which is why it is very useful in memory/ram)

The T-Flip flop will change its output from on to off, or vice versa, each time it receives an input. The D-Flip flop will change its output to whatever the signal at the other input is, each time it receives an input.

* 1. Why do we need shift registers?

Ans:

The applications of shift registers makes it important for microcontrollers. A few of the applications of shift registers are:

**'Serial to Parallel' and 'Parallel to Serial' Conversion**  
 Usually microprocessors and controllers handle data as bytes (8 bits) or words (16 bit, 32 bit ). But serial interfaces in them allow transmitting or receiving data as a series of bits (serial data). So there should be a mechanism that will convert the serial data into parallel data (or vice versa). The Serial-In-Parallel-Out (SIPO) and Parallel-in-Serial-Out (PISO) shift registers makes this possible.

**Multipliers**  
 The basic steps involved in binary multiplication is shifting and adding. Hence shift registers are a part of binary multipliers.

**Registers to store data**  
 Used in ALU's to store operands, intermediate results and final results.

* 1. What Would happen if the output of the last D flip-flop in the register was connected to the input of the first d flip-flop?

Ans: