

FLIP FLOPS

- A flip-flop is the basic memory element for storing a bit of information.
 - It is an edge-triggered device.
 - That is, it reacts to the edge of a pulse.
 - A simple flip-flop has two stable states.
 - Flopflops are preffered over latches,because they are synchronous,more stable,easier to design and more flexible.
-

D FLIP FLOP

- D flip flop is one of the most widely used flopflop in digital circuits.
- It is reliable.
- Less immune to noise.
- D flipflop can be easily cascaded in series to form more complex circuit,such as counters and shift registers.

TRUTH TABLE

D	Qn+1
0	0
1	1

Code(SYSTEM VERILOG)

```
module d_ff(clk,rst,d,q);
  input logic clk,rst,d;
  output logic q;

  always_ff@(posedge clk)
  begin
    if(rst)
      q<=1'b0;
    else
      q<=d;
    end
endmodule
```

T FLIP FLOP

- A T flip flop is known as a toggle flip flop because of its toggling operation. It is a modified form of the [JK flip flop](#).
- A T flip flop is constructed by connecting J and K inputs, creating a single input called T.
- Hence why a T flip flop is also known as a single input JK flip flop.

TRUTH TABLE

T	Q _{n+1}
0	Q _n
1	~Q _n

Code:

```
module t_ff(clk,rst,t,q);
  input logic rst,clk,t;
  output logic q;
  always_ff@(posedge clk)
  begin
    if(rst)
      q <= 1'b0;
    else
      begin
        if(t==1)
          q <= ~q;
        else
          q <= q;
        end
      end
  end
endmodule
```

S R FLIP FLOP

- The SR flip flop is a 1-bit memory bistable device having two inputs, i.e., SET and RESET.
- The SET input 'S' set the device or produce the output 1, and the RESET input 'R' reset the device or produce the output 0.
- The SET and RESET inputs are labeled as **S** and **R**, respectively..

TRUTH TABLE

S	R	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	X

Code:

```
module sr_ff(clk,rst,s,r,q);  
  input logic s,r,clk,rst;  
  output logic q;  
  always_ff@(posedge clk)  
  begin  
    if(rst)  
      q <= 1'b0;  
    else  
      begin  
        case({s,r})  
          2'b00 : q<=q;  
          2'b01 : q <=0;  
          2'b10 : q <=1;  
          2'b11 : q <=1'bx;  
        endcase  
      end  
    end  
end
```

```

    endcase
end
end
endmodule

```

<https://www.edaplayground.com/x/rw4Y>

J K FLIP FLOP

JK Flip flop is an improved version of [SR flip flop](#) where the undefined state of SR Flip flop is eliminated by providing feedback.

TRUTH TABLE

J	K	Q _{n+1}
0	0	Q _n
0	1	0
1	0	1
1	1	~Q _n

Code

```

module jk_flipflop(j,k,clk,rst,q);
    input logic clk,rst,j,k;
    output logic q;
    always_ff@(posedge clk)
    begin
        if(!rst)
            q<= 1'b0;
        else
            begin
                case({j,k})
                    2'b00 : q<=q;
                    2'b01 : q<=0;
                    2'b10 : q<=1;
                    2'b11 : q<=~q;
                endcase
            end
    end
endmodule

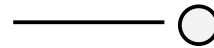
```

```
end
end
endmodule
```

<https://www.edaplayground.com/x/XZaR>

d LATCH using 2:1 MUX

```
module mux(a,b,s,y);
  input a,b,s;
  output reg y;
  always @(a,b,s)
  begin
    y=(~s & a) | (s & b);
  end
endmodule
```



```
module d_latch(d,q,en);
  input d,en;
  inout q;
  mux mux_1(.a(q),.b(d),.s(en),.y(q));
endmodule
```

<https://www.edaplayground.com/x/RjB4>