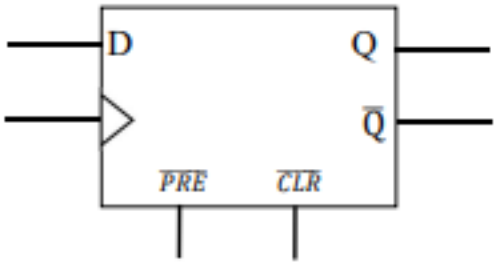
Experiment 5

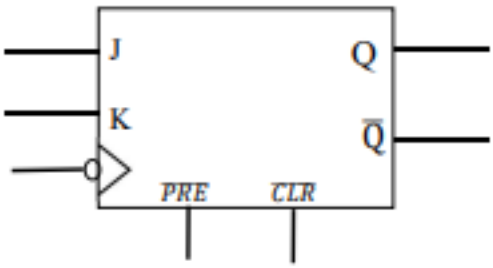
D Flip-flop:




```
module DFF(output Q,output Qb, input D,input clk,input CLR,input PRE);

reg Q;      // output Q and Qb defined as register
reg Qb;
 initial
   begin
     Q=0;
     Qb=1;
   end
 always @(posedge clk or negedge CLR or negedge PRE)
begin
 if  (CLR == 0)
   begin
   Q = 0;
   Qb = 1;
   end
 else
  if  (PRE == 0)
   begin
   Q  = 1;
   Qb = 0;
   end
 else
   begin
   Q = D;
   Qb = ~D;
   end
end
endmodule
```

JK Flip-flop:




```
module JKFF(output Q,output Qb, input J,input K, input clk,input CLR,input PRE);

reg Q; // output Q and Qb defined as register
reg Qb;
 initial
   begin
     Q=0;
     Qb=1;
   end
 always @ (negedge clk or negedge CLR or negedge PRE)
begin
 if (CLR == 0)
   begin
   Q = 0;
   Qb = 1;
   end
 else
  if (PRE == 0)
   begin
   Q  = 1;
   Qb = 0;
   end
 else
   if ( J == 0 & K == 0 )
   begin
   Q = Q;
   Qb = ~Q;
   end
 else
   if ( J == 0 & K == 1 )
   begin
   Q = 0;
   Qb = 1;
   end
 else
   if ( J == 1 & K == 0 )
   begin
   Q = 1;
   Qb = 0;
```

```
    end
  else
    if ( J == 1 & K == 1 )
    begin
    Q = ~Q;
    Qb = ~Qb;
    end

end
endmodule
```

Test-bench Code for D Flip-flop

```
module test_DFF( );
reg D;
reg clk;
reg PRE;
reg CLR;
wire Q;
wire Qb;

 DFF FF1 (Q,Qb,D,clk,CLR,CLR);

initial
  begin
  clk=0;
  forever #10 clk = ~clk;
end
initial
begin
 CLR=1;PRE=1; D = 0; #20
 CLR=0;#10
 CLR=1; D = 1; #20;
 D = 0; #20;
 PRE=0;#20
 PRE=1;#10
 D = 1; #10
  $finish;
end
endmodule
```

Test-bench code for JK FF

```
module test_JKFF();
reg J;
reg K;
reg clk;
reg PRE;
reg CLR;
wire Q;
wire Qb;

 JKFF FF1 (Q,Qb,J,K,clk,CLR,PRE);

initial
 begin
 clk = 0;
 forever #10 clk = ~clk;
end
initial
begin
 CLR = 1; PRE = 1; J = 0; K = 0; #20
 PRE = 0; #10
 PRE = 1; J = 0; K = 1; #20;
 J = 1; K = 0;#20
 CLR = 0;#20
 CLR = 1;J = 1; K = 1; #20;
 $finish;
end
endmodule
```