Lab Program -8

Design Verilog program for implementing various types of Flip-Flops such as SR, JK and D.

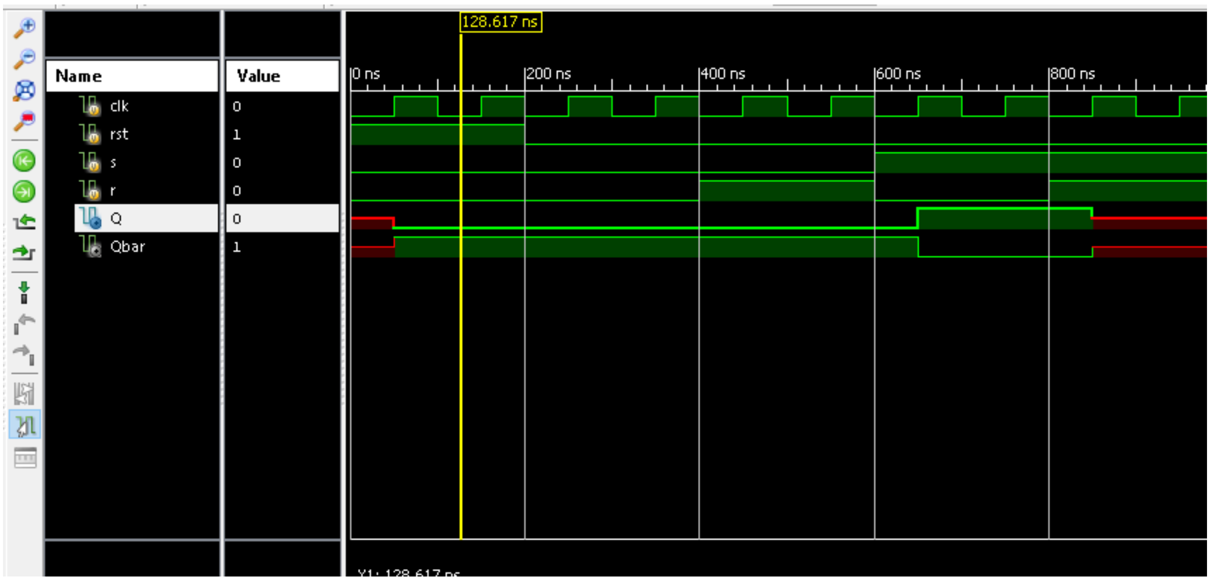
```
module SRff(
    input s,r,clk,en,
    output reg Q,
    output Qbar
    );
always@(posedge clk)
begin
if(en==1)
Q<=0;
else if(s==0&&r==0)
Q<=Q;
else if(s==0&&r==1)
Q<=0;
else if(s==1&&r==0)
Q<=1;
else
Q<=1'bx;
end
assign Qbar=~Q;
endmodule
```

Test Benches

```
module test_srr;
            // Inputs
            reg s;
            reg r;
            reg clk;
            reg en;
```

```
    // Outputs
    wire Q;
    wire Qbar;
    // Instantiate the Unit Under Test (UUT)
    SRff uut (
            .s(s),
            .r(r),
            .clk(clk),
            .en(en),
            .Q(Q),
            .Qbar(Qbar)
    );
    initial begin
            // Initialize Inputs
            s = 0;
            r = 0;
            clk = 0;
            en = 1;
            #200 en=0; s=0;r=0;
            #200 s=0; r=1;
            #200 s=1; r=0;
            #200 s=1; r=1;
            end
            always
            begin
            #50  clk= ~clk;
    end
endmodule
```

Output:



Name	Value
clk	0
rst	1
s	0
r	0
Q	0
Qbar	1

Program JK Flip flop

```
module jk(
    input j,k,clk,en,
    output reg Q,
    output Qbar
    );
always@(posedge clk)
begin
if(en= =1)
Q<=0;
else begin
case({j,k})
2'b00:Q<=Q;
2'b01:Q<=0;
2'b10:Q<=1;
2'b11:Q<=~Q;
default:Q<=1'bx;
endcase
end
```

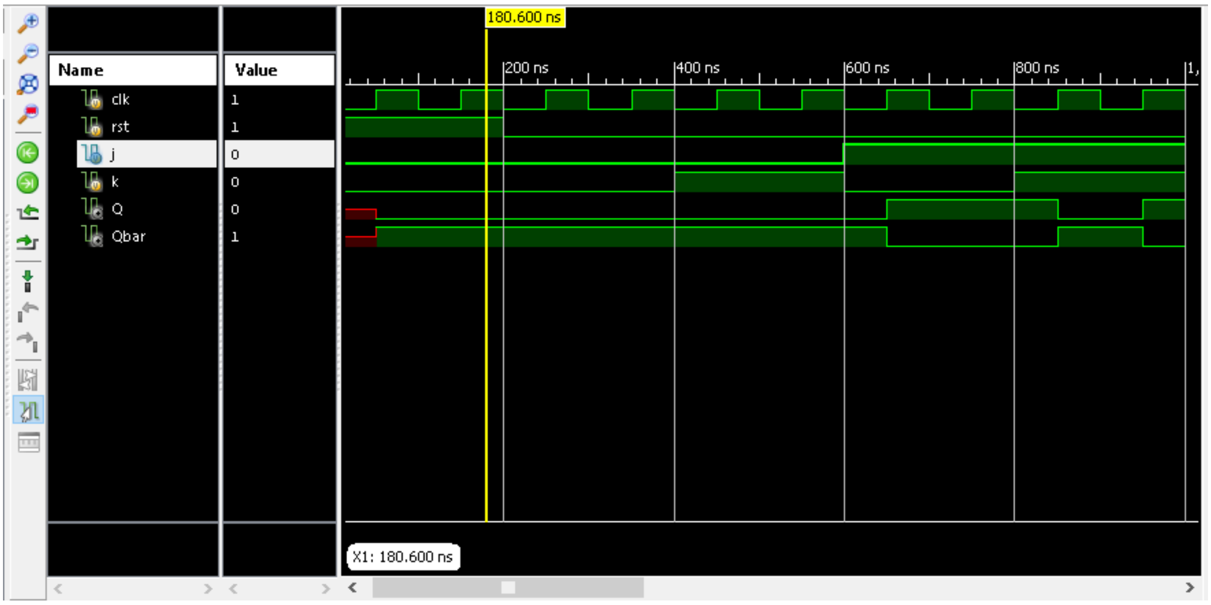
```
end
assign Qbar=~Q;
endmodule
```

TESTBENCH

```
module test_jk;
        // Inputs
        reg j;
        reg k;
        reg clk;
        reg en;
        // Outputs
        wire Q;
        wire Qbar;
        // Instantiate the Unit Under Test (UUT)
        jk uut (
                .j(j),
                .k(k),
                .clk(clk),
                .en(en),
                .Q(Q),
                .Qbar(Qbar)
        );
        initial begin
                // Initialize Inputs
                j = 0;
                k = 0;
                clk = 0;
                en = 1;
                #200 en=0;j=0; k=0;
```

```
            #200    j=0; k=1;

            #200    j=1; k=0;

    #200    j=1; k=1;

            end

            always

            begin

            #50     clk= ~clk;

        end

endmodule
```

OUTPUT



Name	Value
clk	1
rst	1
j	0
k	0
Q	0
Qbar	1

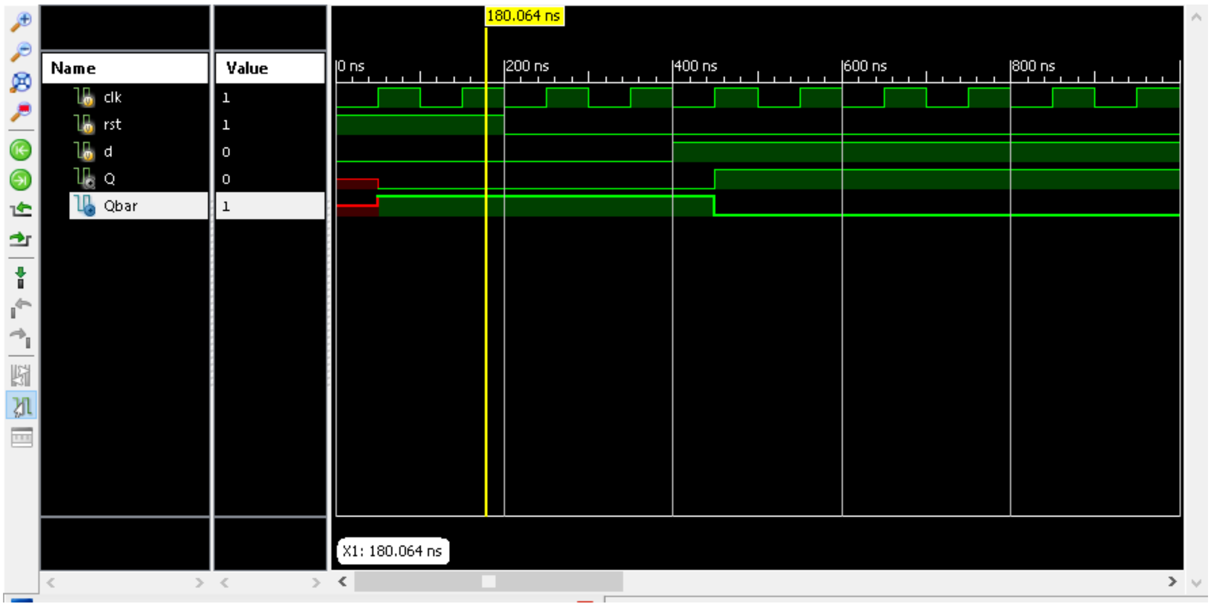
Program for D FLIP-FLOP

```
module DFF(
   input d,clk,en,
   output reg Q,
   output Qbar
   );
always@(posedge clk)
begin
if(en==1)
Q<=0;
else
Q<=d;
end
assign Qbar=~Q;
endmodule
```

TESTBENCH

```
initial begin
                  d = 0;
                  clk = 0;
                  en = 1;
                  #200     en=0; d=0;
                  #200 d=1;
                  end
                  always
                  begin
                  #50      clk=  ~clk;
         end
endmodule
```

OUTPUT



Name	Value
clk	1
rst	1
d	0
Q	0
Qbar	1