



**BITS Pilani**

Hyderabad Campus

Department of Electrical Engineering

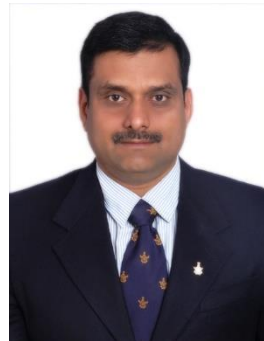


**Digital Design : 2020-21**

**Lab 10**

**Sequential Circuit Design using  
Flip-flops in Xilinx**

**By Dr. Sanjay Vidhyadharan**



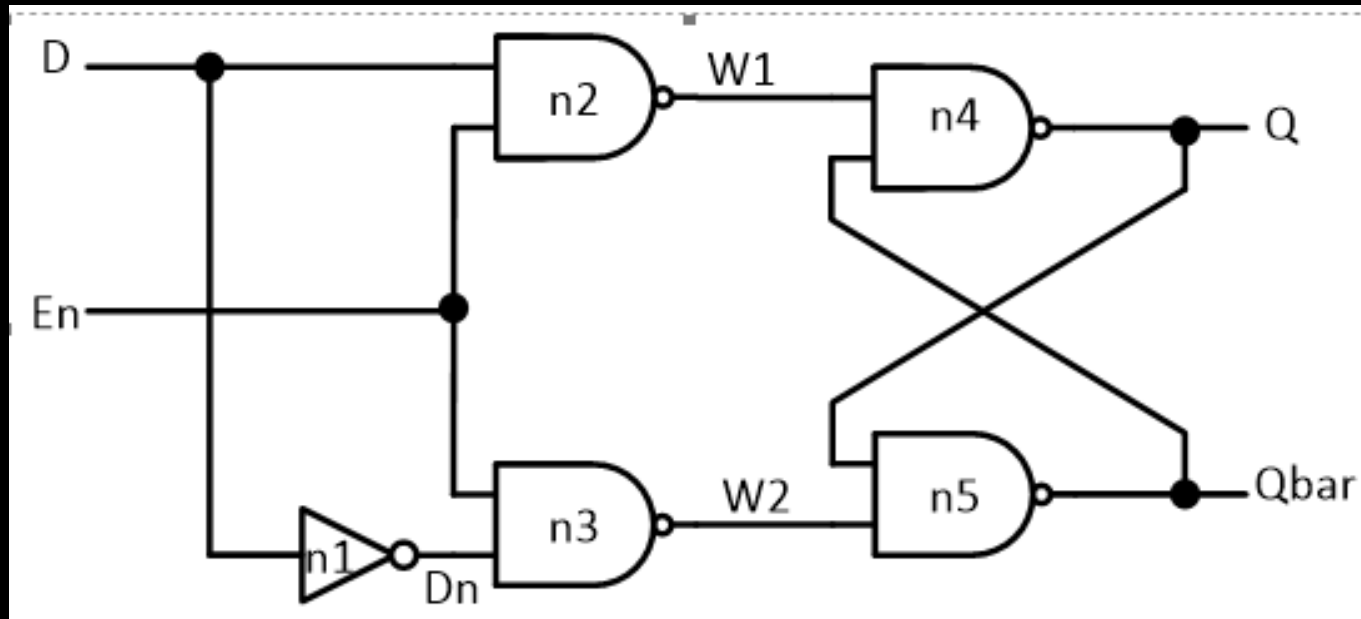
# **New Concepts**

**Flip-flop code in Verilog**

**Test bench for Sequential Circuits**



# D Latch





# D Latch

```
module D_latch(  
    input D,  
    input En,  
    output Q,  
    output Qbar  
);  
wire Dn, W1, W2;  
  
not n1(Dn,D); // NOT gate instance  
  
nand n2(W1, D, En); // NAND gate instance for n2  
  
//similarly write the NAND instances n3.  
  
//Q and Qbar can also be used as inputs for example  
//n4 NAND instance can be written as shown below  
nand n4(Q, W1, Qbar);  
  
//similarly write the NAND instances n5.  
  
endmodule
```

Complete the code and Test operation of D latch



# D Flip-flop

```
module DFF_CLK (  
input D,  
input Clk,  
output reg Q);  
always @ (posedge Clk)  
begin  
Q <= D;  
end  
endmodule
```

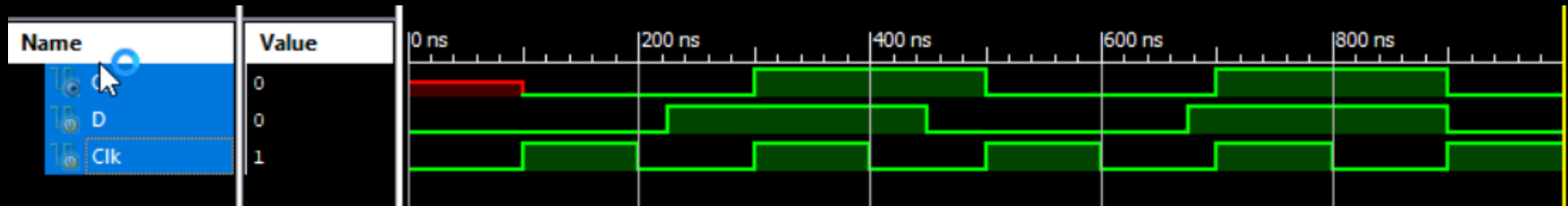


# Testbench for D Flip-flop

```
module DDF1_test;
    // Inputs reg D; reg Clk;
    // Outputs wire Q;
    // Instantiate the Unit Under Test (UUT)
    DFF_CLK uut ( .D(D), .Clk(Clk), .Q(Q) );
    initial begin
        // Initialize Inputs
        Clk = 0;
        repeat (9)
            #100 Clk = ~ Clk;
    end
    initial begin
        // Initialize Inputs
        D = 0;
        repeat (6)
            #225 D = ~ D;
    end
end
endmodule
```



# Simulation Plots of D Flip-flop





# D Flip-flop with Reset

```
module DD_FF_R(input D, input Clk, input
Reset, output reg Q);
always @ (posedge Clk, posedge Reset)
begin
if (Reset==1)
Q <= 1'b0;
else
Q <= D;
end
endmodule
```





# Testbench for D Flip-flop with Reset

```
initial begin
```

```
    Clk = 0;  
    repeat (9)  
        #100 Clk = ~ Clk;
```

```
end
```

```
initial begin
```

```
    D = 0;  
    repeat (6)  
        #225 D = ~ D;
```

```
end
```

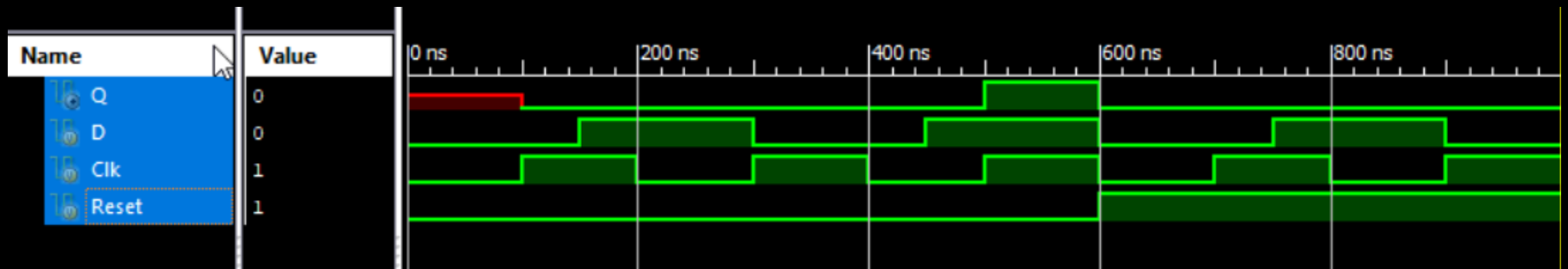
```
initial begin
```

```
    Reset = 0;  
    repeat (2)  
        #600 Reset = ~ Reset;
```

```
end
```

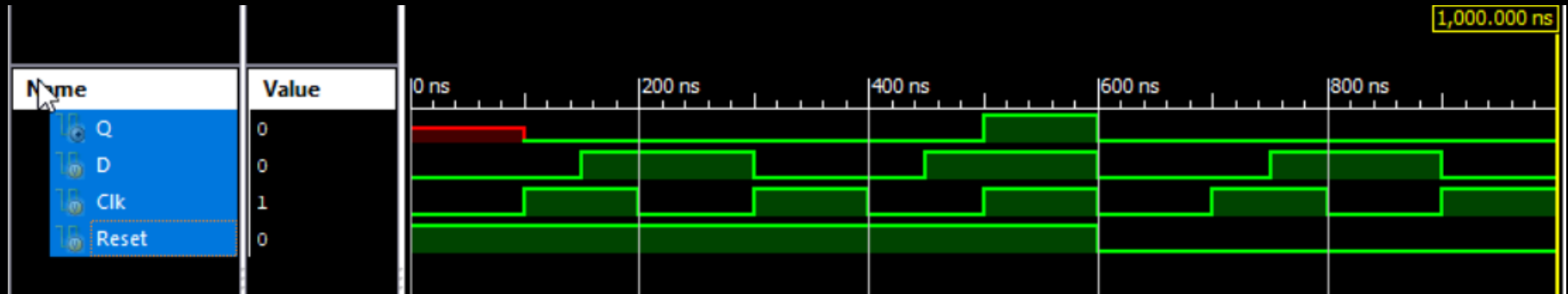


# Simulation Plots of D Flip-flop With Reset





# Simulation Plots of D Flip-flop With Reset



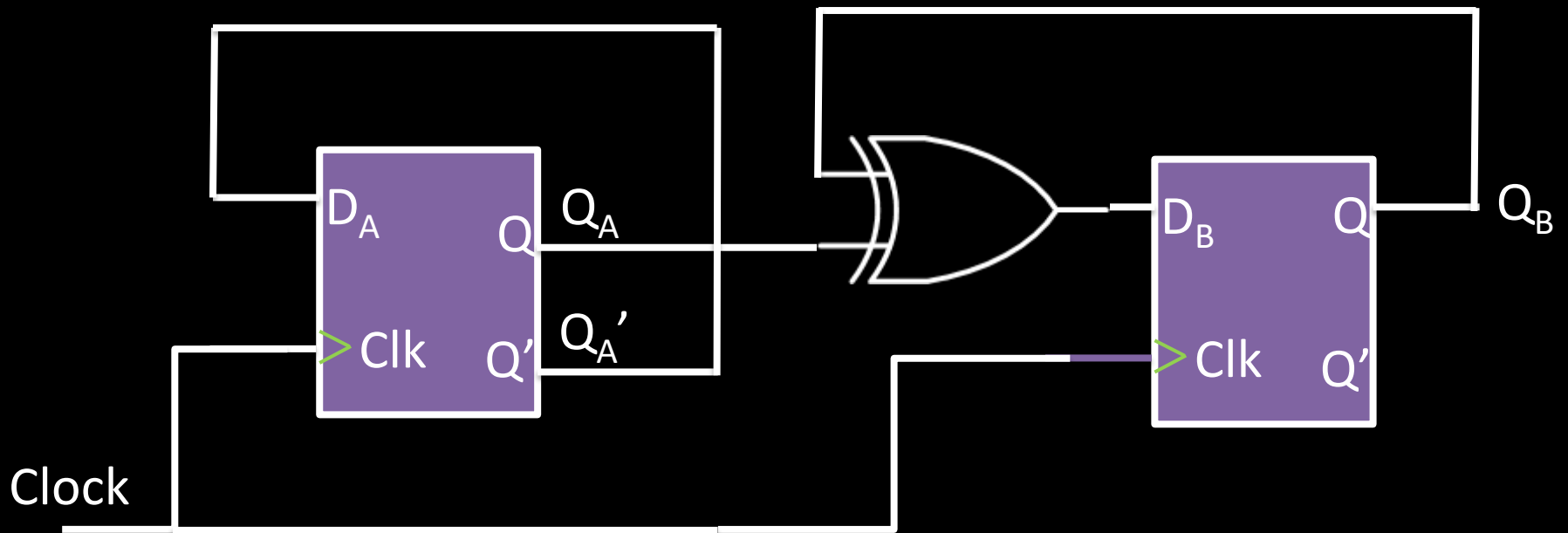
```
module DD_FF_R(input D, input Clk, input
Reset, output reg Q);
always @ (posedge Clk, negedge Reset)
begin
if (Reset==0)
Q <= 1'b0;
else
Q <= D;
end
endmodule
```

```
initial begin
Reset = 1;
repeat (2)
#600 Reset = ~ Reset;
end
```



# Problem 1

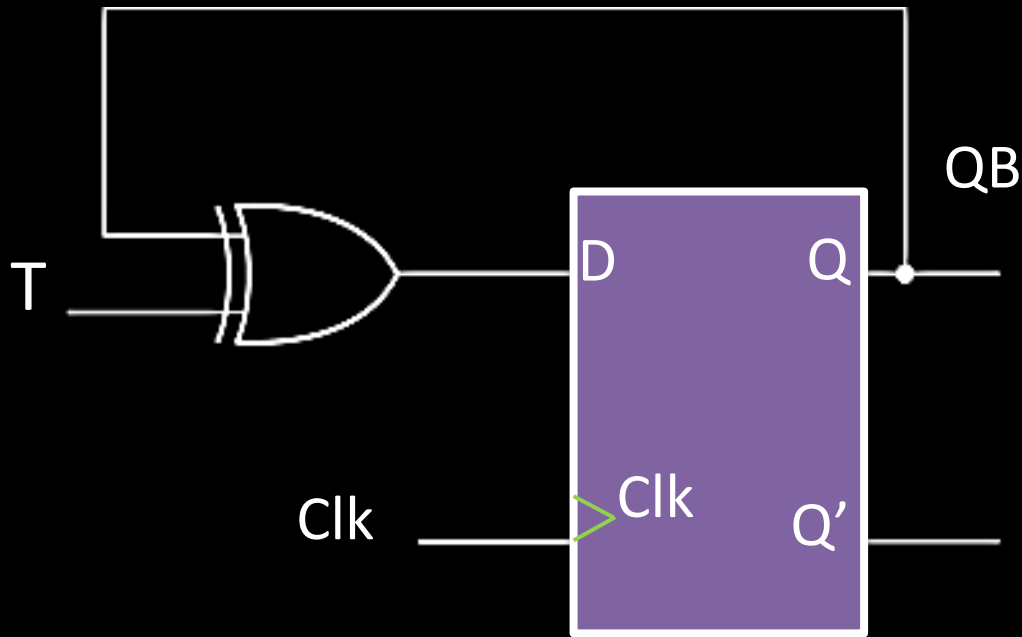
Make a 2 Bit Binary Counter





## Problem 2

Design a T flip-flop using D flip-flop and XOR Gate





## Problem 3

Design a JK flip-flop using D flip-flop



Thank You