

中国科学技术大学计算机学院

《数字电路实验报告》



实验题目：使用 Vivado 进行仿真

学生姓名：谭骏飞

学生学号：PB20061276

完成时间：2021.11.17

实验题目

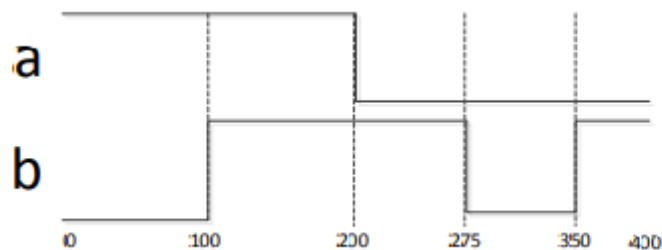
使用 **Vivado** 进行仿真

实验环境

Vivado 2020.1

实验练习

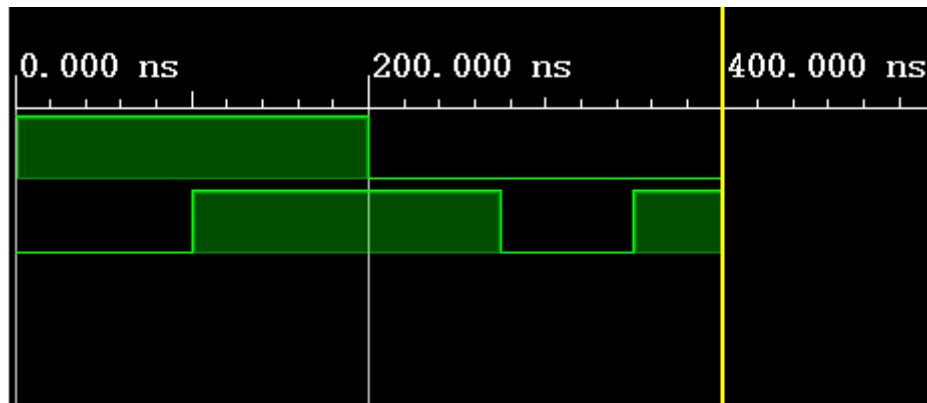
题目 1. 请编写 **Verilog** 仿真文件，生成如下图所示的波形，并在 **Vivado** 中进行仿真。



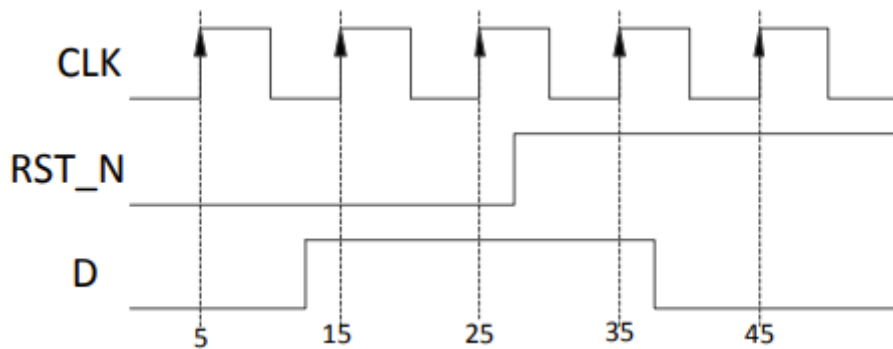
根据波形图的特征，在仿真文件中编写如下代码。

```
module test_bench( );
    reg a,b;
    initial
    begin
        a = 1'b1; b = 1'b0; //根据波形图给a和b赋初值
        #100 a = 1'b1; b = 1'b1;
        #100 a = 1'b0; b = 1'b1; //延时100个时间单位后将A置0
        #75 a = 1'b0; b = 1'b0;
        #75 a = 1'b0; b = 1'b1;
        #50 $finish;
    end
endmodule
```

仿真生成如下波形。



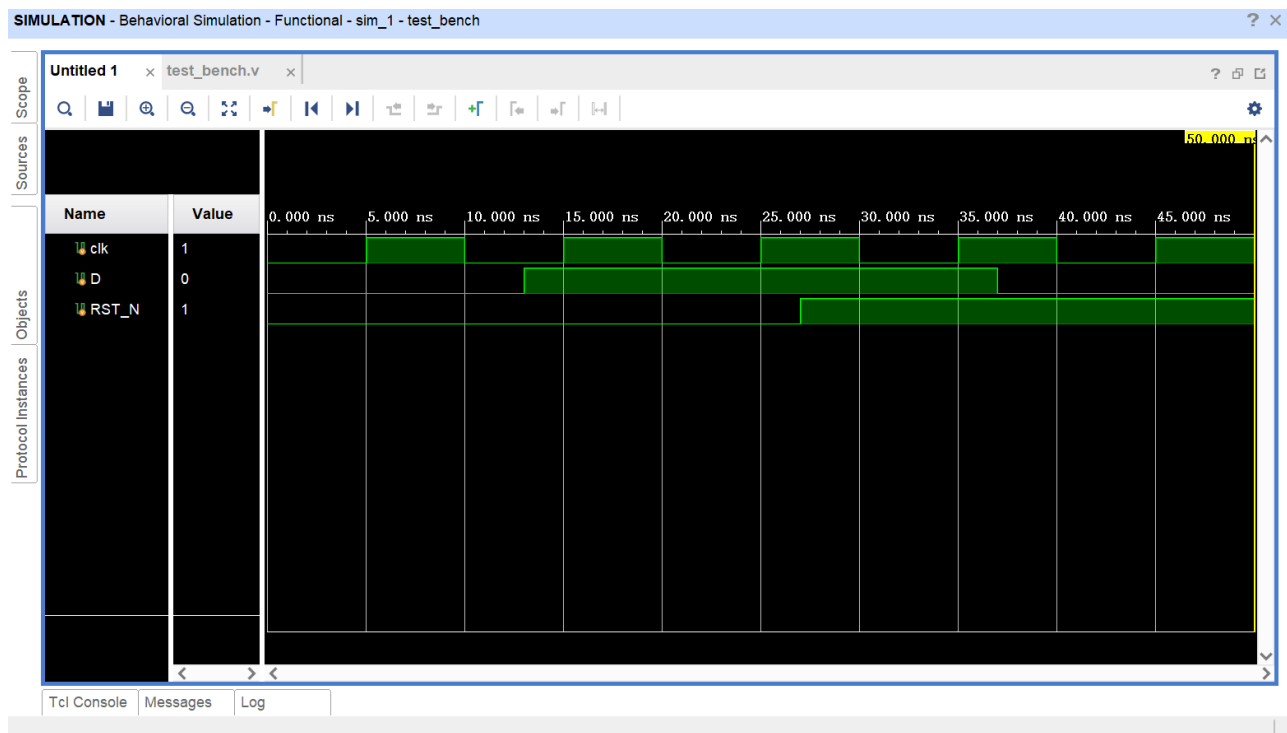
题目 2. 请编写 **Verilog** 仿真文件，生成如下图所示的波形，并在 **Vivado** 中进行仿真。



根据波形图的特征，在仿真文件中编写如下代码。

```
module test_bench( );
    reg clk,D,RST_N;
    initial clk = 1'b0;
    always #5 clk = ~clk; //每隔五个时间单位，clk取反
    initial
    begin
        D = 1'b0; RST_N = 1'b0;
        #13 D = ~D;
        #14 RST_N = ~RST_N;
        #10 D = ~D;
        #13 $finish;
    end
endmodule
```

仿真生成如下波形。



题目 3. 利用题目 2 中的信号作为以下代码的输入，在 **Vivado** 中对其仿真，并观察仿真波形。

```
module d_ff_r(  
    input clk,rst_n,d,  
    output reg q);  
    always@(posedge clk)  
    begin  
        if(rst_n==0)  
            q <= 1'b0;  
        else  
            q <= d;  
        end  
    endmodule
```

在仿真文件中编写如下代码。

```
module test_bench( );  
    reg clk,D,RST_N;  
    wire q;
```

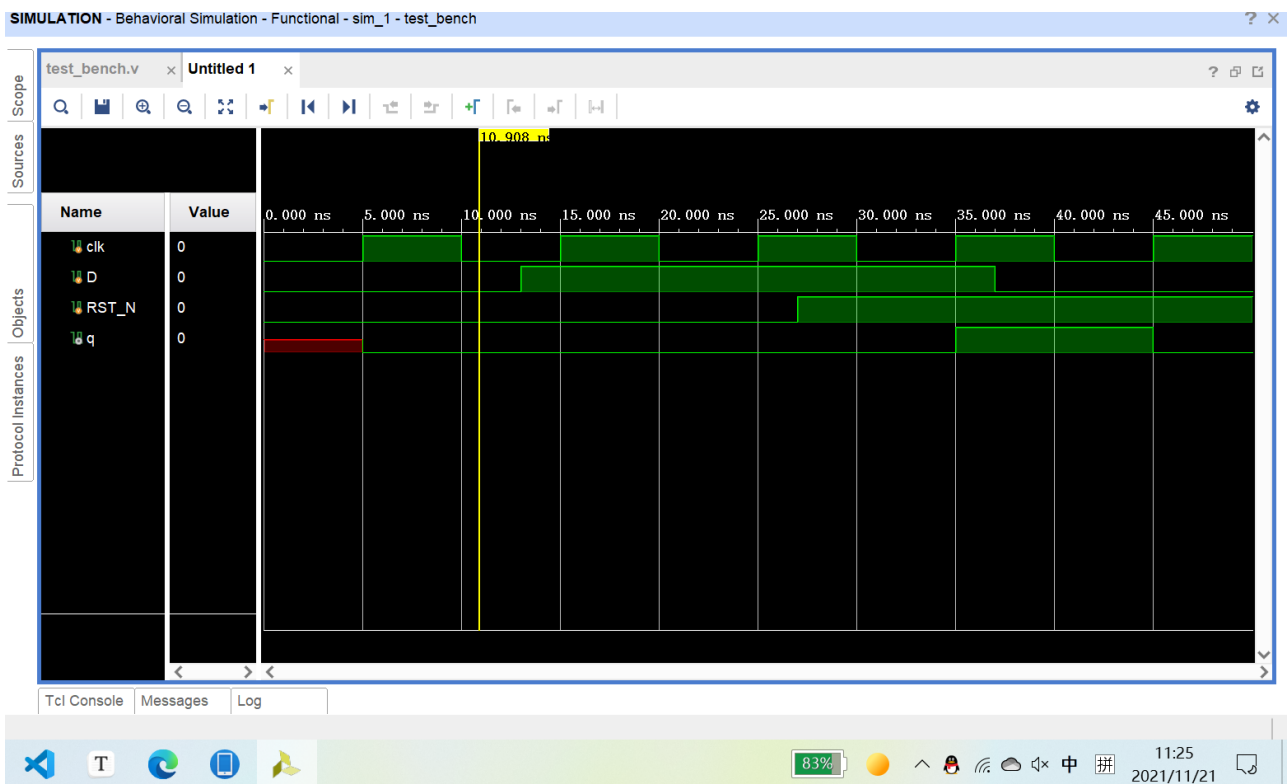
```

//assign q=1'b0;
d_ff_r ins(.clk(clk),.d(D),.rst_n(RST_N),.q(q));
initial clk = 1'b0;
always #5 clk = ~clk;
initial
begin
    D = 1'b0; RST_N = 1'b0;
    #13 D = ~D;
    #14 RST_N = ~RST_N;
    #10 D = ~D;
    #13 $finish;
end

endmodule

```

仿真生成如下波形。



注：由于一开始并未达到时钟上升沿，并且 q 初始状态未知，因此 q 的波形图在到达第一个时钟上升沿之前都处于红色未知状态。

题目 4. 设计一个 **3-8** 译码器，编写仿真测试文件，在 **Vivado** 中对其进行仿真。要求仿真时遍历所有的输入情况组合，给出源代码和仿真截图。

3-8译码器的真值表如下。

$I_2I_1I_0$	$O_7O_6O_5O_4O_3O_2O_1O_0$
000	11111110(xFE)
001	11111101(xFD)
010	11111011(xFB)
011	11110111(xF7)
100	11101111(xEF)
101	11011111(xDF)
110	10111111(xBF)
111	01111111(x7F)

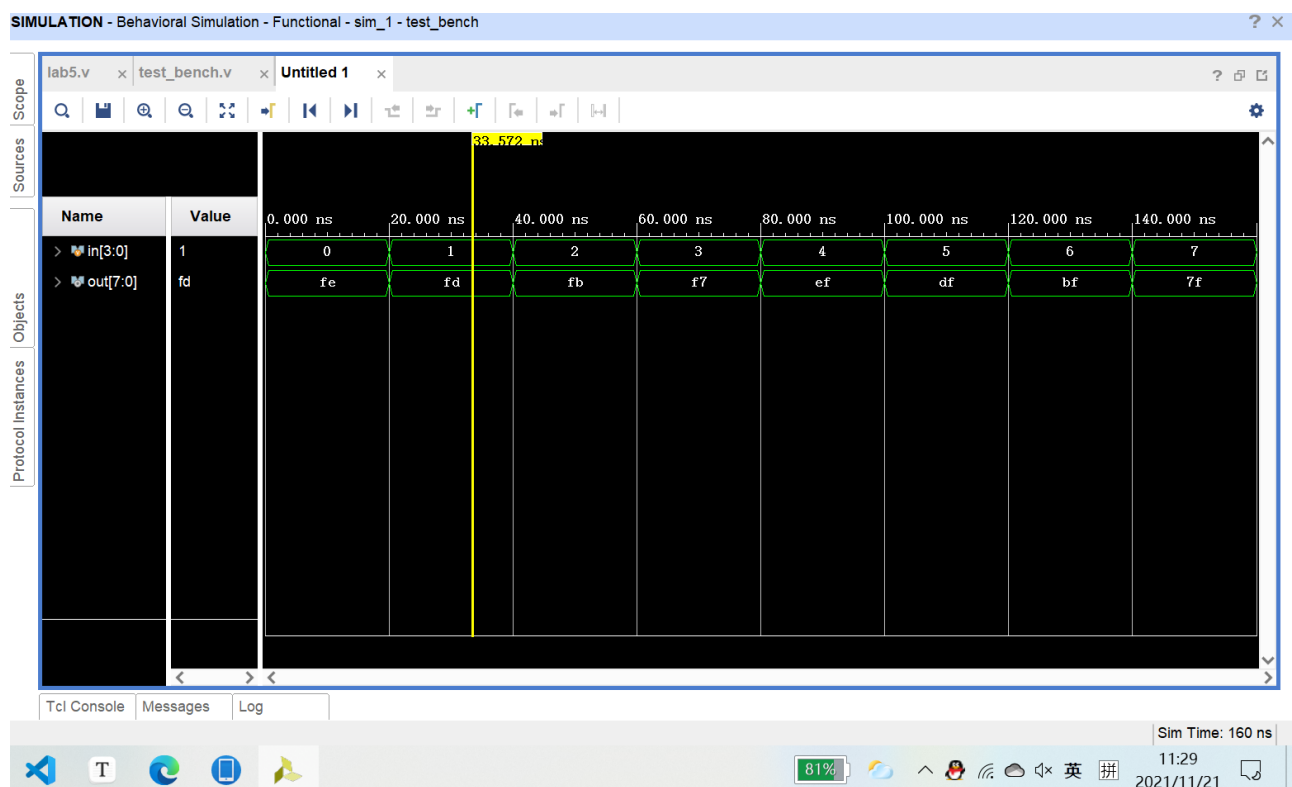
有关于**3-8**译码器的**Verilog**设计文件如下。

```
module lab5(  
    input [2:0] in,  
    output reg [7:0] out  
);  
    always@(*)//组合逻辑电路  
begin  
    case(in)  
        3'b000: out = 8'b11111110;  
        3'b001: out = 8'b11111101;  
        3'b010: out = 8'b11111011;  
        3'b011: out = 8'b11110111;  
        3'b100: out = 8'b11101111;  
        3'b101: out = 8'b11011111;  
        3'b110: out = 8'b10111111;  
        3'b111: out = 8'b01111111;  
    endcase  
end  
endmodule
```

相应的仿真文件内容如下。

```
module test_bench( );
reg [3:0] in;
wire [7:0] out;
    lab5 ins(.in(in),.out(out)); //将设计文件中的模块实例化
initial
begin
    in = 3'b000;
    #20 in = 3'b001;
    #20 in = 3'b010;
    #20 in = 3'b011;
    #20 in = 3'b100;
    #20 in = 3'b101;
    #20 in = 3'b110;
    #20 in = 3'b111; //遍历所有的输入情况
    #20 $finish;
end
endmodule
```

仿真后的得到以下波形图。



总结与思考

本次实验初步学习了**Vivado**的使用，包括**Verilog**设计文件、仿真文件的添加，仿真文件的编写，以及如何进行代码仿真。从而之前的**Verilog**代码得以以一种更为直观的波形图的形式呈现出来，更有利于检查**Verilog**代码的正确性，以及加深对**Verilog**语法和编写的理解，为以后进一步利用**Vivado**进行一些更大项目的仿真打下基础。