



南京邮电大学
Nanjing University of Posts and Telecommunications

电工电子实验报告

课程名称： 电工电子实验（二）
实验项目： 动态显示电路

学 院： 贝尔英才学院
班 级：
学 号：
姓 名：
学 期： 2022-2023学年第1学期

动态显示电路

一、实验目的

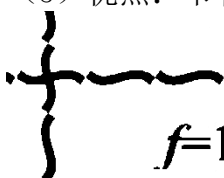
1. 使用ISE软件完成时序逻辑电路的设计输入并仿真。
2. 掌握Testbench中时序逻辑测试文件的写法。
3. 下载并测试实现的逻辑功能。

二、实验仪器

ISE Design Suite 14.7

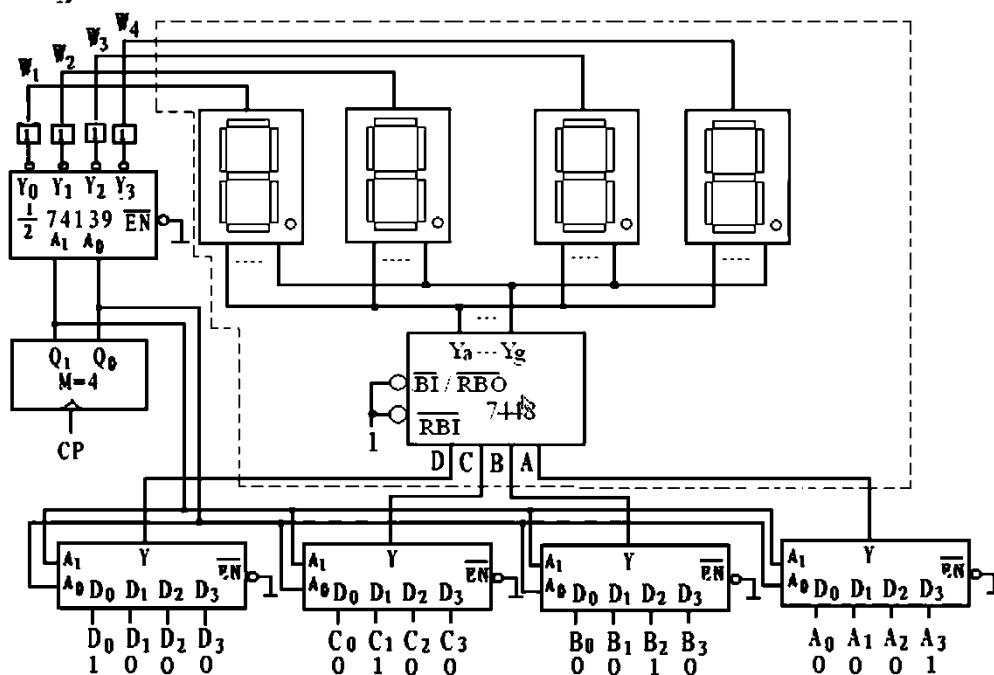
三、实验原理

1. 通过M=4的计数器同时控制译码器实现位选和数据选择器实现数选。
 - (1) 控制部分：提供数码管的位选信号和数据选择器的地址信号
 - (2) 数据选择部分：提供显示的数据信息
 - (3) 译码驱动：二—十进制译码
 - (4) 显示部分：十进制数字显示
 - (5) 优点：节省七段显示译码器和能源



实验原理

$f=1\text{Hz}$ ，显示的数据为：8421。



四、实验内容

1. 设计一个4位动态显示电路，显示内容为本人学号后四位“0115”，完成设计代码及测试代码，记录显示译码器输入端和数码管位选端的输入波形。
2. 拓展要求：设计六位动态显示电路，显示本人学号后六位“010115”，完成设计代码及测试代码，记录显示译码器输入端和数码管位选端的输入波形。
3. 拓展要求：设计一个模10计数器

五、实验数据

1. 设计一个4位动态显示电路，显示内容为本人学号后四位“0115”，完成设计代码及测试代码，记录显示译码器输入端和数码管位选端的输入波形。

(1) 设计代码

```

module four(D0,D1,D2,D3,clk,rst,Dout,Eout
);
    input[3:0] D0;
    input[3:0] D1;
    input[3:0] D2;
    input[3:0] D3;
    input clk;
    input rst;
    output reg [3:0] Dout;
    output reg [3:0] Eout;
    reg [1:0] cnt;
always@(posedge clk or negedge rst)
begin
    if(!rst==1)
        cnt<=2'b00;
    else
        cnt<=cnt+2'b01;
    end
always@(cnt)
begin
    case(cnt)
        2'b00 : Eout<=4'b0111;
        2'b01 : Eout<=4'b1011;
        2'b10 : Eout<=4'b1101;
        default : Eout<=4'b1110;
    endcase
end
always@(cnt or D3 or D2 or D1 or D0)
begin
    case(cnt)
        2'b00 : Dout <= D3;
        2'b01 : Dout <= D2;
        2'b10 : Dout <= D1;
        default : Dout <= D0;
    endcase
end
endmodule

```

(2) 测试代码

```

module four1;

    // Inputs
    reg [3:0] D0;
    reg [3:0] D1;
    reg [3:0] D2;
    reg [3:0] D3;
    reg clk;
    reg rst;

    // Outputs
    wire [3:0] Dout;
    wire [3:0] Eout;

    // Instantiate the Unit Under Test (UUT)
    four uut (
        .D0(D0),
        .D1(D1),
        .D2(D2),
        .D3(D3),
        .clk(clk),
        .rst(rst),
        .Dout(Dout),
        .Eout(Eout)
    );

```

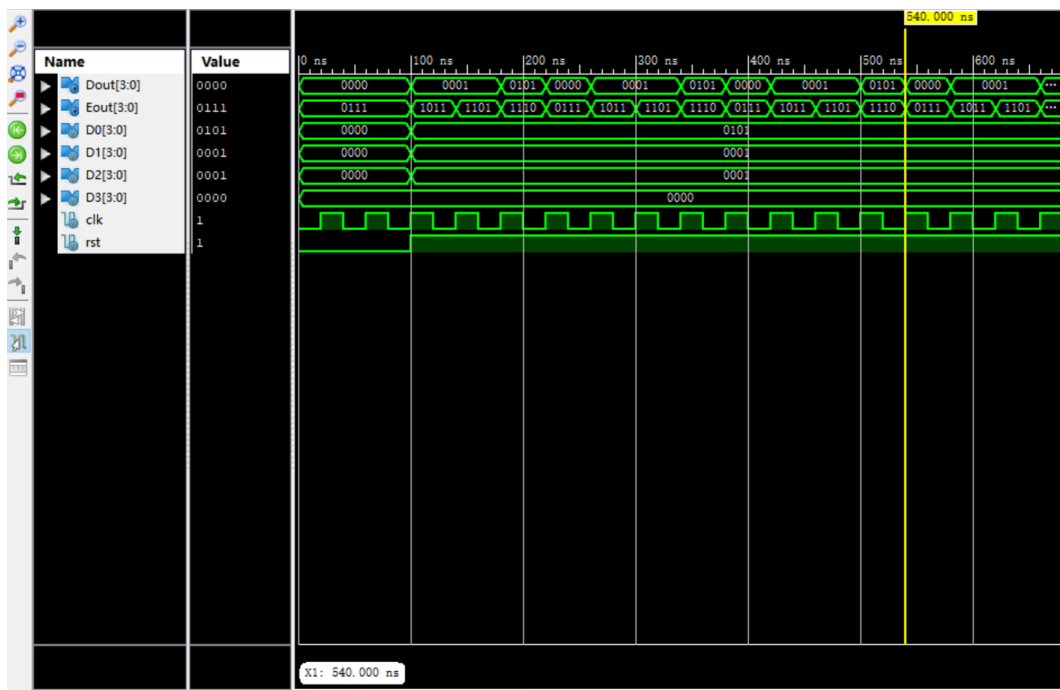
```

initial begin
    // Initialize Inputs
    D0 = 0;
    D1 = 0;
    D2 = 0;
    D3 = 0;
    clk = 0;
    rst = 0;

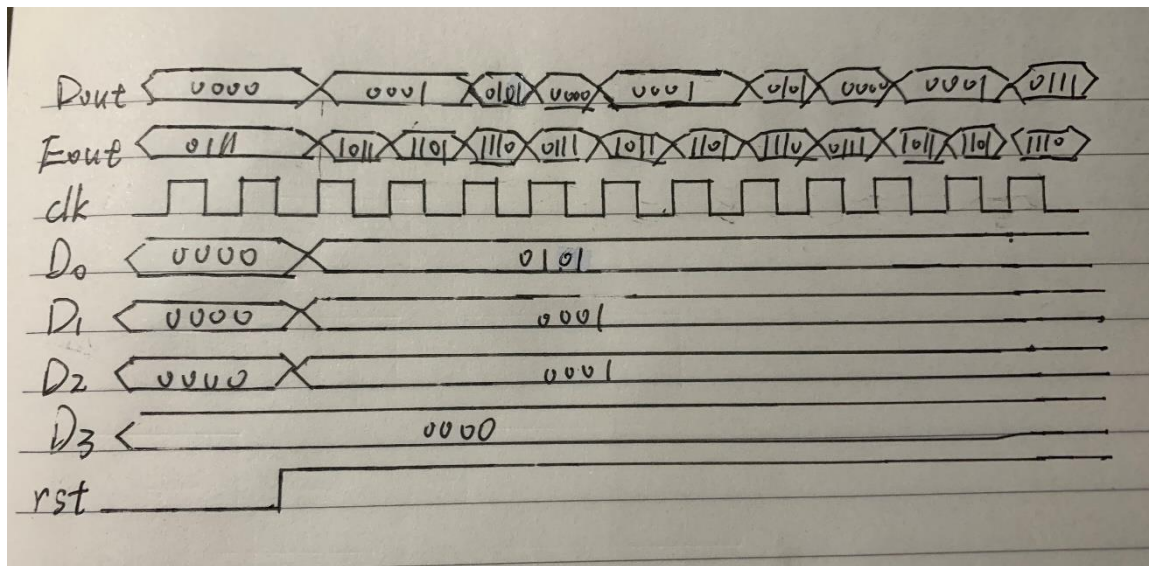
    // Wait 100 ns for global reset to finish
    #100;
    rst=1;
    D3 = 4'b0000;
    D2 = 4'b0001;
    D1 = 4'b0001;
    D0 = 4'b0101;
    // Add stimulus here

end
    always #20 clk=~clk;
endmodule
    
```

(3) 仿真结果（截图及手绘）



4位动态显示电路仿真截图



4位动态显示电路手绘波形图

2. 拓展要求：设计六位动态显示电路，显示本人学号后六位“010115”，完成设计代码及测试代码，记录显示译码器输入端和数码管位选端的输入波形。

(1) 设计代码

```

module six(D0,D1,D2,D3,D4,D5,clk,rst,Dout,Eout,en
);
    input [5:0] D0;
    input [5:0] D1;
    input [5:0] D2;
    input [5:0] D3;
    input [5:0] D4;
    input [5:0] D5;
    input clk;
    input rst;
    input en;
    output reg [5:0] Dout;
    output reg [5:0] Eout;
    reg [3:0] cnt;
    always@(posedge clk or negedge rst)
    begin
        if(!rst)
            cnt<=3'b000;
        else
            if(cnt==3'b101)
                cnt<=3'b000;
            else
                cnt<=cnt+3'b001;
        end
    end
    always@(cnt)
    begin
        case(cnt)
            3'b000 : Eout<=6'b011111;
            3'b001 : Eout<=6'b101111;
            3'b010 : Eout<=6'b110111;
            3'b011 : Eout<=6'b111011;
            3'b100 : Eout<=6'b111101;
            default : Eout<=6'b111110;
        endcase
    end

    always@(cnt or D5 or D4 or D3 or D2 or D1 or D0)
    begin
        case(cnt)
            3'b000 : Dout <= D5;
            3'b001 : Dout <= D4;
            3'b010 : Dout <= D3;
            3'b011 : Dout <= D2;
            3'b100 : Dout <= D1;
            default : Dout <= D0;
        endcase
    end
endmodule

```

(2) 测试代码


```

module six1;

    // Inputs
    reg [5:0] D0;
    reg [5:0] D1;
    reg [5:0] D2;
    reg [5:0] D3;
    reg [5:0] D4;
    reg [5:0] D5;
    reg clk;
    reg rst;
    reg en;

    // Outputs
    wire [5:0] Dout;
    wire [5:0] Eout;

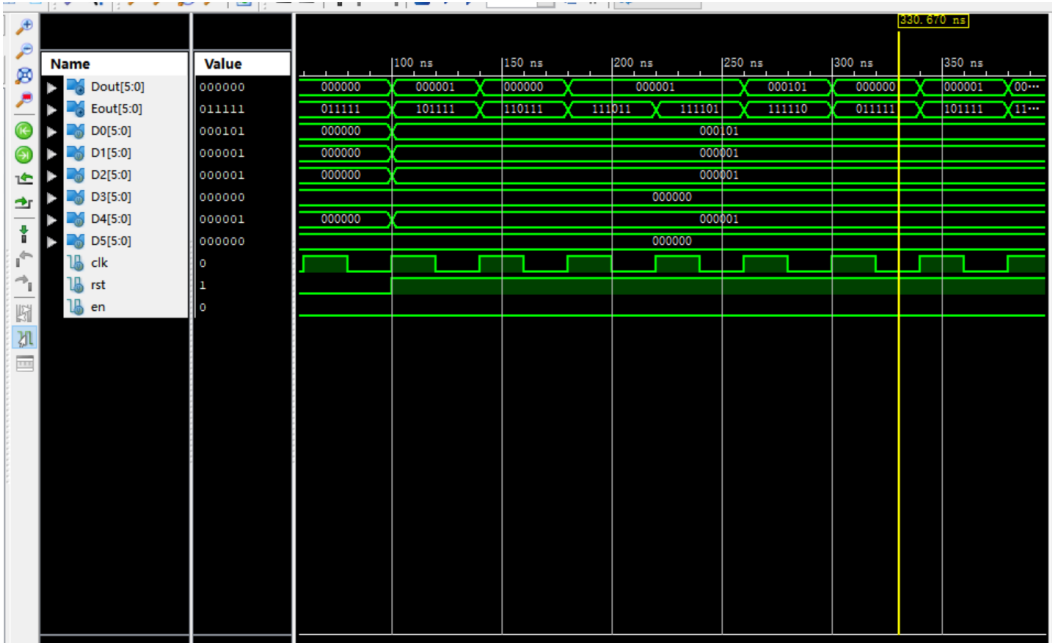
    // Instantiate the Unit Under Test (UUT)
    six uut (
        .D0(D0),
        .D1(D1),
        .D2(D2),
        .D3(D3),
        .D4(D4),
        .D5(D5),
        .clk(clk),
        .rst(rst),
        .Dout(Dout),
        .Eout(Eout),
        .en(en)
    );

```

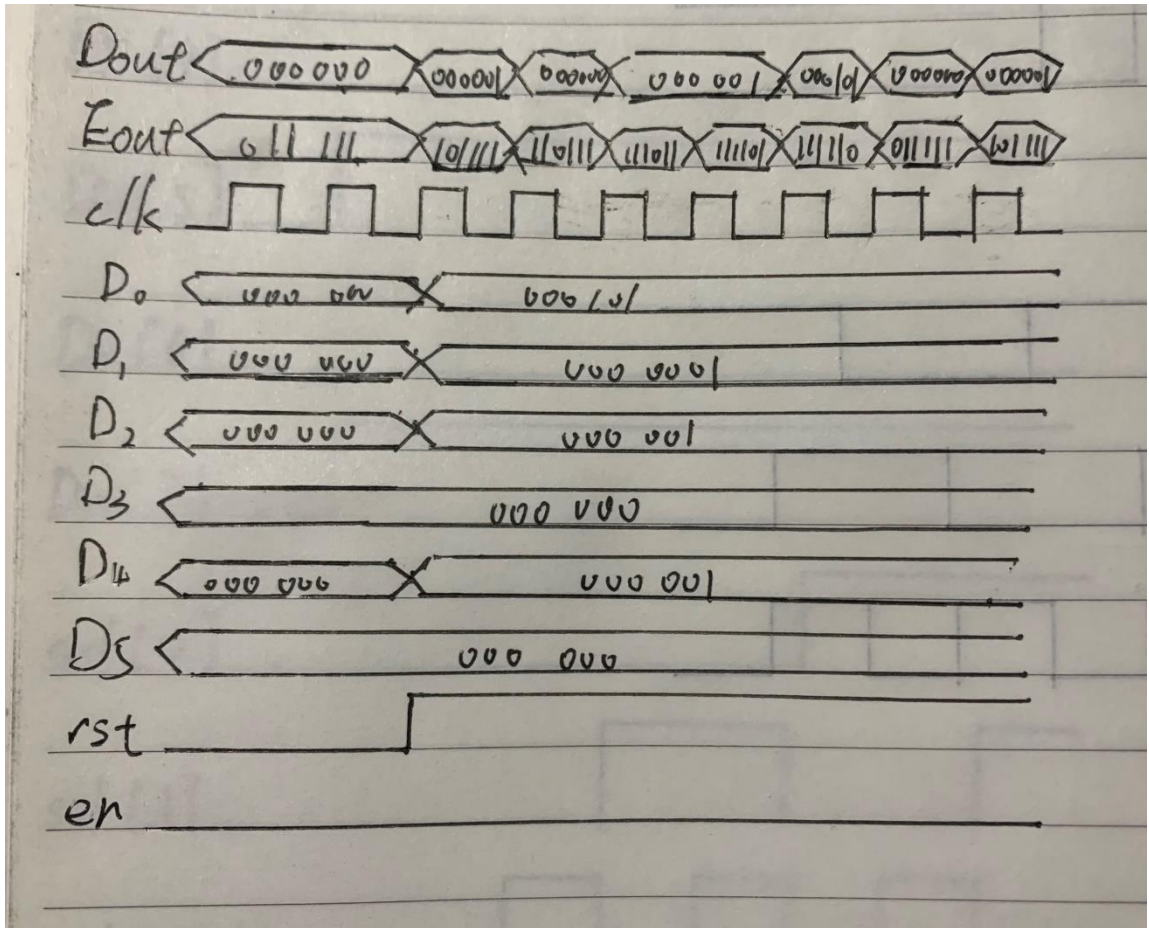
```
initial begin
    // Initialize Inputs
    D0 = 0;
    D1 = 0;
    D2 = 0;
    D3 = 0;
    D4 = 0;
    D5 = 0;
    clk = 0;
    rst = 0;
    en = 0;

    // Wait 100 ns for global reset to finish
    #100;
    rst=1;
    D5 =6'b0000000;
    D4 =6'b0000001;
    D3 =6'b0000000;
    D2 =6'b0000001;
    D1 =6'b0000001;
    D0 =6'b000101;
    // Add stimulus here
end
    always #20 clk =~clk;
endmodule
```

(3) 仿真结果（截图及手绘）



6位动态显示电路仿真截图



6位动态显示电路手绘图

3. 设计一个模10计数器（反馈置零法）

(1) 设计代码

```

module count10t;

    // Inputs
    reg clk;
    reg rst;

    // Outputs
    wire [3:0] Dout;

    // Instantiate the Unit Under Test (UUT)
    count10 uut (
        .clk(clk),
        .rst(rst),
        .Dout(Dout)
    );

    always
    begin
        #100 clk=~clk;
    end
    initial begin
        // Initialize Inputs
        clk = 0;
        rst = 0;
        #100 rst=1;
        // Wait 100 ns for global reset to finish

        // Add stimulus here

    end

endmodule

```

(2) 测试代码

```

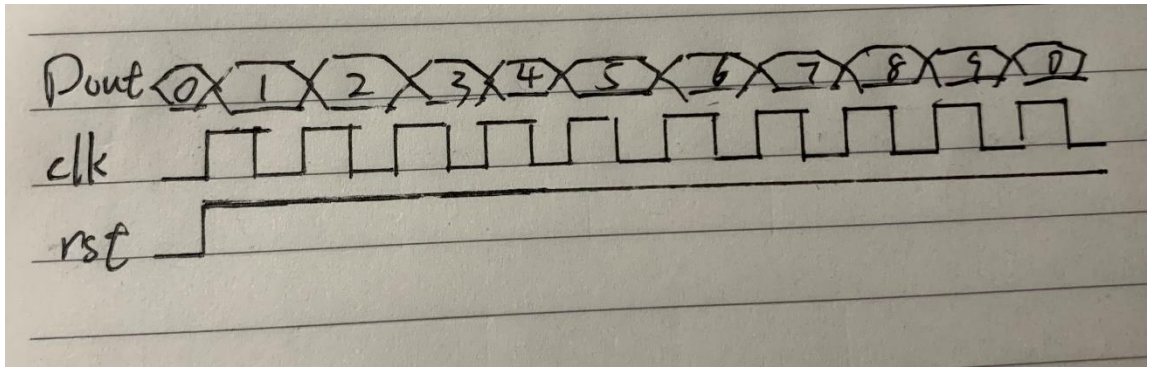
module count10(clk,rst,Dout
);
input clk,rst;
output [3:0] Dout;
reg [3:0] Dout;
always@(posedge clk or negedge rst)
begin
if(!rst)
Dout<=4'b0000;
else if(Dout==4'b1001)
Dout<=4'b0000;
else
Dout<=Dout+4'b0001;
end
endmodule

```

(3) 仿真结果（截图及手绘）



模10计数器仿真截图



模10计数器手绘图

六、实验小结

这次实验通过对动态显示电路以及计数器的具体实现，更加深刻地了解了Verilog语言的编码规则以及方法。掌握了使用ISE软件完成时序逻辑电路的设计输入并仿真和Testbench中时序逻辑测试文件的写法。对动态译码显示的工作过程有了更深的理解。