

同济大学计算机系

数字逻辑课程实验报告



学 号 2252707

姓 名 陈艺天

专 业 计算机科学与技术

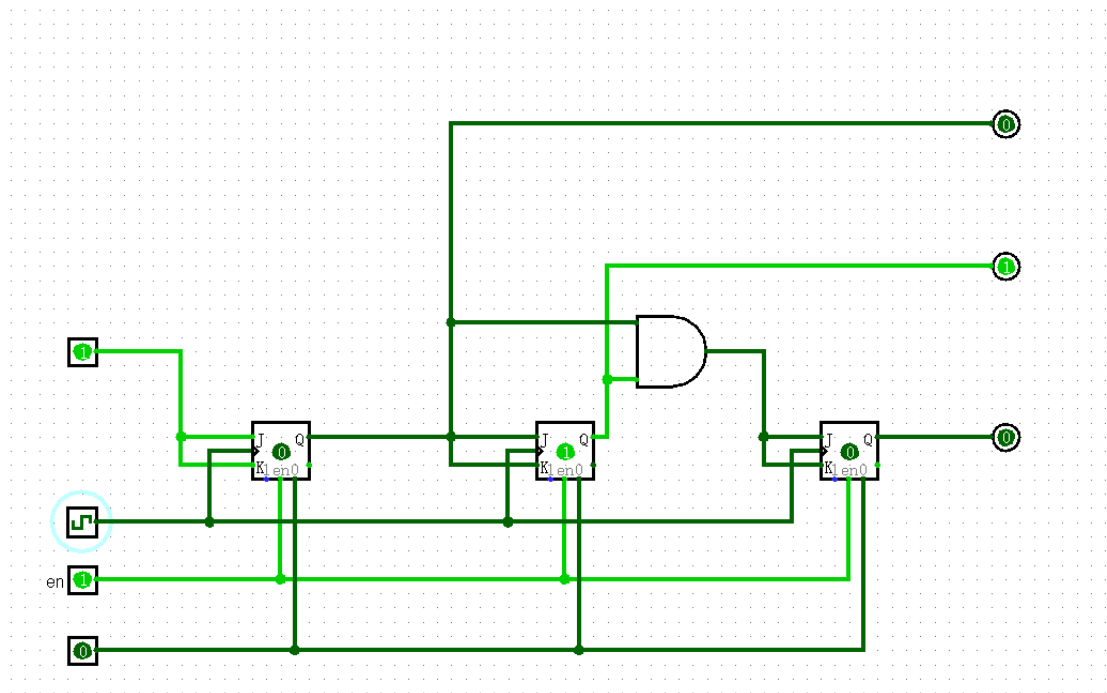
授课老师 张冬冬

1. 计数器实验

一、实验内容

本次实验中，使用 verilog HDL 实现计数器的设计与仿真，以及下板验证，将计数通过数码管显示。

二、硬件逻辑图



三、模块建模

模块一：分频器，倍率为 100000000

```
module Divider(  
    input I_CLK,  
    input rst_n,  
    output O_CLK  
);  
    reg tO_CLK = 0;  
    parameter rate = 100000000;  
    integer cnt = 0;  
  
    always @(posedge I_CLK or negedge rst_n) begin  
        if(~rst_n) begin  
            tO_CLK=0;  
            cnt = 0;  
        end  
    end
```

```

        end

        else if(cnt==rate/2 - 1) begin
            cnt = 0;
            t0_CLK=~t0_CLK;
        end

        else begin
            cnt = cnt + 1;
        end
    end

    assign O_CLK=t0_CLK;

endmodule

```

模块二：JK 触发器

```

module JK_FF(
    input CLK, // 时钟信号 ， 上升沿有效
    input J, // 输入 信号 J
    input K, // 输入 信号 K
    input RST_n, // 复位信号 ， 低电平有效
    output reg Q1, // 输出 信号 Q
    output reg Q2 // 输出 信号
);

always @(posedge CLK or negedge RST_n) begin
    if (!RST_n) begin
        Q1=0;
        Q2=1;
    end
    else begin
        if(J&K) begin
            Q1=~Q1;
            Q2=~Q2;
        end
        else if(~J&~K)
            ;
        else begin
            Q1=J;
            Q2=K;
        end
    end
end
end

```

endmodule

模块三：数码管显示器

```
module display7(  
    input[3:0] iData,  
    output[6:0] oData  
);  
    reg[6:0] oData_tmp;  
    always @(*) begin  
        case(iData)  
            4'b0000:  
                oData_tmp=7'b1000000;  
            4'b0001:  
                oData_tmp=7'b1111001;  
            4'b0010:  
                oData_tmp=7'b0100100;  
            4'b0011:  
                oData_tmp=7'b0110000;  
            4'b0100:  
                oData_tmp=7'b0011001;  
            4'b0101:  
                oData_tmp=7'b0010010;  
            4'b0110:  
                oData_tmp=7'b0000010;  
            4'b0111:  
                oData_tmp=7'b1111000;  
            4'b1000:  
                oData_tmp=7'b0000000;  
            4'b1001:  
                oData_tmp=7'b0010000;  
            default:  
                oData_tmp=7'b1111111;  
        endcase  
    end  
    assign oData=oData_tmp;  
endmodule
```

模块四：计数器

```
module Counter8(  
    input CLK,  
    input rst_n,  
    output [2:0] oQ,  
    output [6:0] oDisplay  
);
```

```

wire [3:0] oQ4; // 辅助的 4 位信号
wire O_CLK;
Divider Div(CLK,rst_n,O_CLK);
JK_FF FF0(O_CLK, 1, 1, rst_n, oQ[0]);
JK_FF FF1(O_CLK, oQ[0], oQ[0], rst_n, oQ[1]);
JK_FF FF2(O_CLK, oQ[1] & oQ[0], oQ[1] & oQ[0], rst_n, oQ[2]);

assign oQ4 = {1'b0, oQ}; // 将最高位设置为 0
display7 disp(oQ4, oDisplay);
endmodule

```

四、测试模块建模

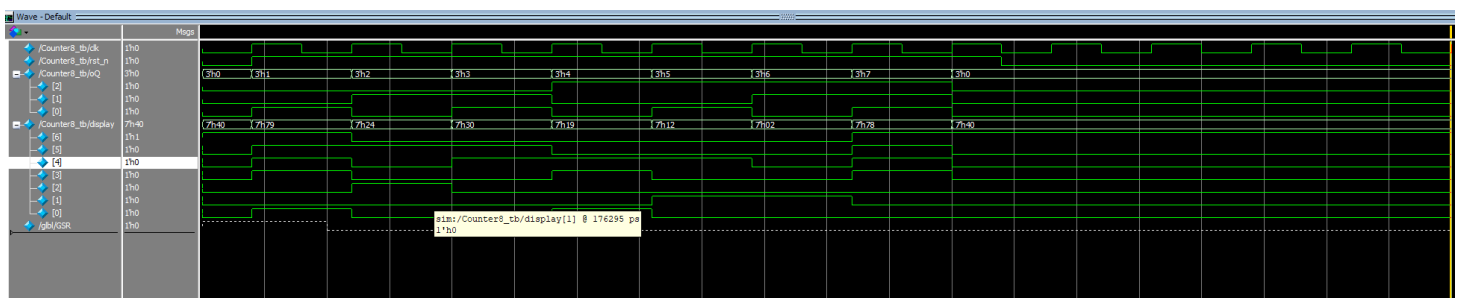
```

module Counter8_tb;
    reg clk;
    reg rst_n;
    wire [2:0]oQ;
    wire [6:0]display;
    Counter8 inst(clk,rst_n,oQ,display);
    parameter t = 40;
    initial begin
        clk=0;
        repeat(t) begin
            #40 clk=~clk;
        end
    end
    initial begin
        rst_n=0;
        #40 rst_n=1;
        #600 rst_n=0;
    end
endmodule

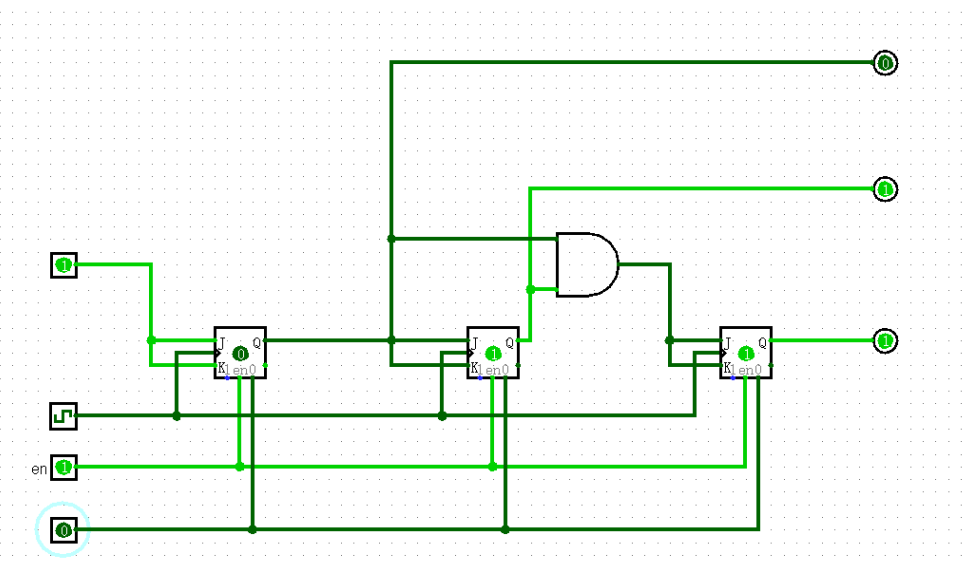
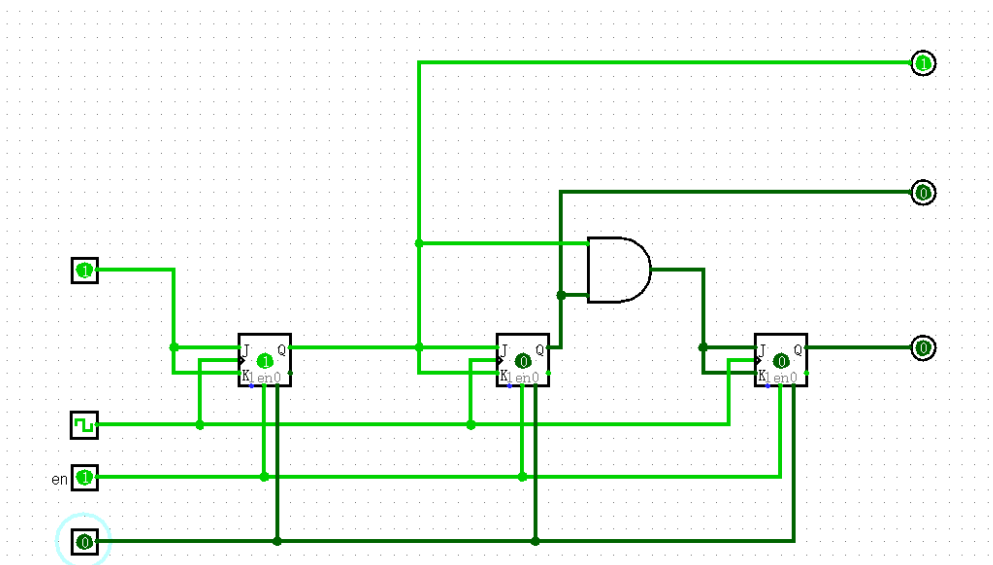
```

五、实验结果

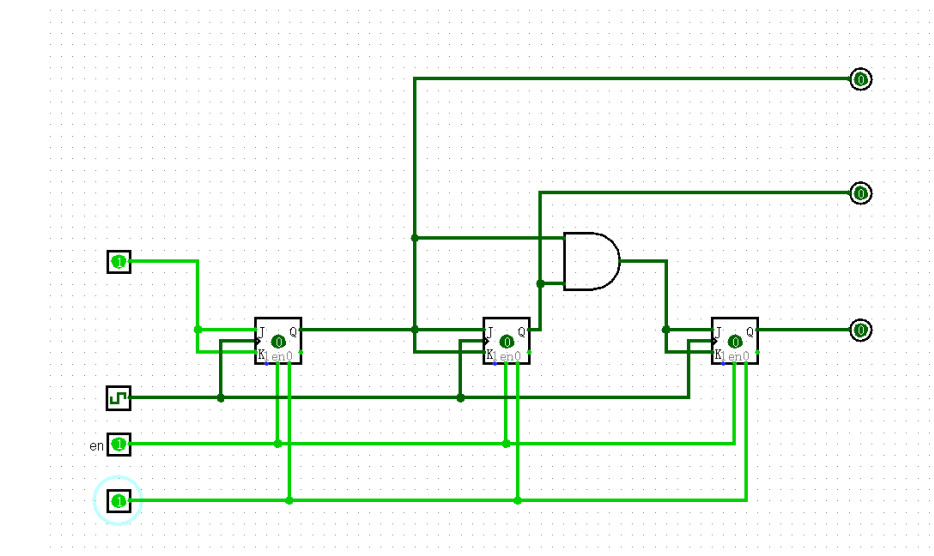
(1) Modelsim 仿真结果



(2) logisim 逻辑验证

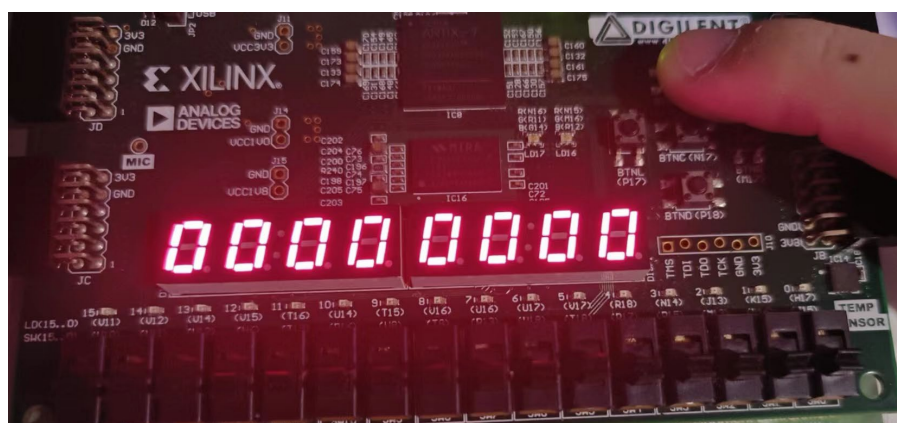
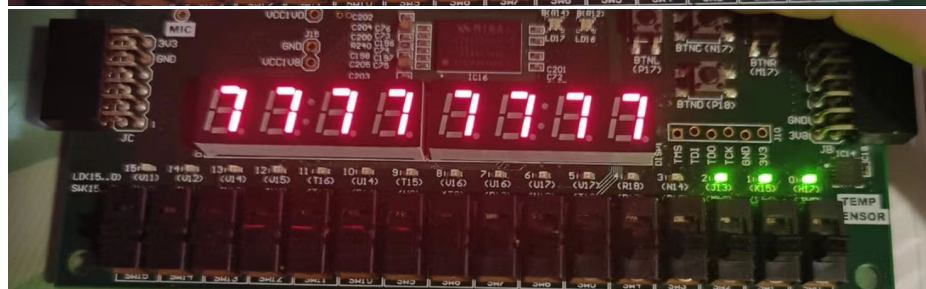
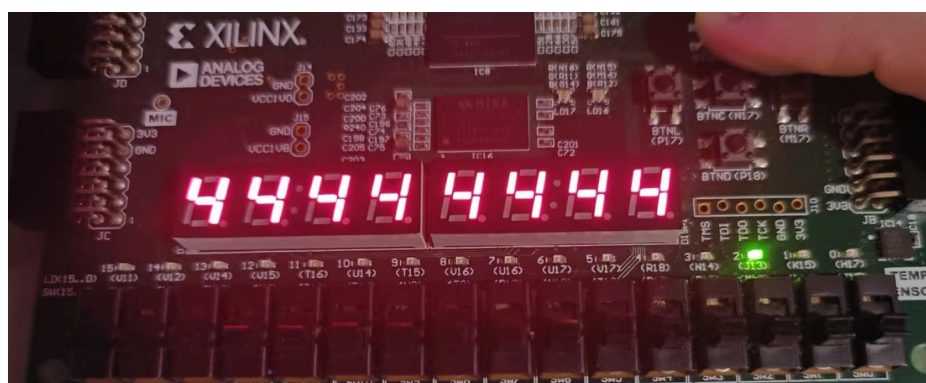
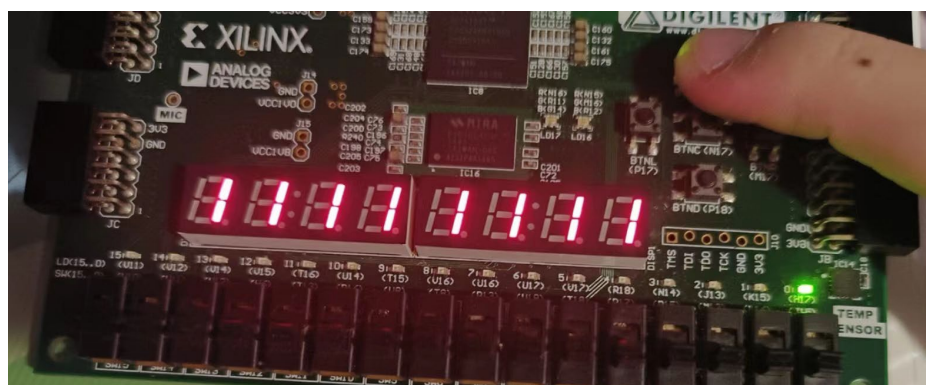


通过电路验证，观察到随着时钟上升沿到来，结果在递增，递增到 8 时归 0；
如果设置 rst 端，由于是异步方式实现，计数器会立马归零。



(3) 下板结果验证

经过分频器的分频，频率调整为 1Hz，观察到数字每秒递增，增到 8 归零；或者设置 rst 也实现清零。



2. 分频器实验

一、 实验内容

本次实验中使用 verilog HDL 实现分频器的设计与仿真以及下板验证。

二、 模块建模

```
`timescale 1ns / 1ps

module Divider(
    input I_CLK,
    input rst,
    output O_CLK
);
    reg tO_CLK = 0;
    parameter rate = 20; //下板时设置为 100000000
    integer cnt = 0;

    always @(posedge I_CLK) begin
        if(rst) begin
            tO_CLK=0;
            cnt = 0;
        end
        else if(cnt==rate/2 - 1) begin
            cnt = 0;
            tO_CLK=~tO_CLK;
        end
        else begin
            cnt = cnt + 1;
        end
    end

    assign O_CLK=tO_CLK;
endmodule
```

三、 测试模块建模

```
`timescale 1ns / 1ps

module Divider_tb;
    reg I_CLK;
```



```

reg rst;
wire O_CLK;
Divider inst(I_CLK,rst,O_CLK);

parameter t = 880;
initial begin
    I_CLK=0;
    repeat(t) begin
        #5 I_CLK=~I_CLK;
    end
end

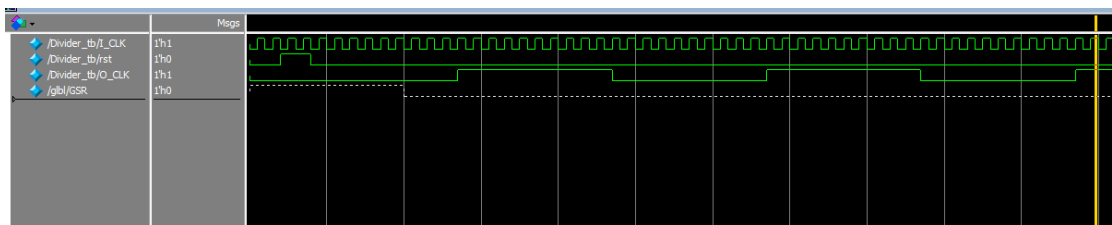
initial begin
    rst = 0;
    #20 rst = 1;
    #20 rst = 0;
end

endmodule

```

四、 实验结果

(1) Modelsim 仿真结果



(2) 下板实验验证

