

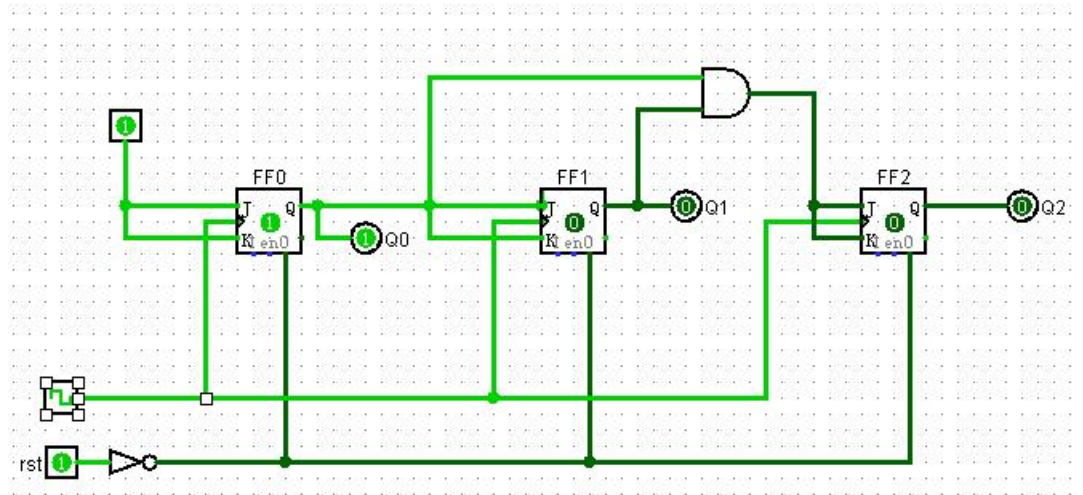
一、实验内容

在本次实验中，将使用 Verilog HDL 语言设计实现：

- (1) 计数器的设计和仿真
- (2) 分频器的设计和仿真

二、硬件逻辑图(逻辑验证图在第五部分)

(1) 计数器的设计



三、模块建模

(1) 同步模 8 计数器的设计和仿真

功能描述：记忆脉冲的个数，所能记忆的最大脉冲数为 8.并且在该模块实例化了分频器，使得实验现象可以肉眼观察。

```
module Counter8(  
    input CLK,  
    input rst_n,  
    output [2:0] oQ,  
    output [6:0] oDisplay  
);  
    wire l0,l1,l2;  
    wire out;  
  
    Divider divv1(  
        .I_CLK(CLK),  
        .rst(rst_n),  
        .O_CLK(out)  
    );
```

```

JK_FF f1(
    .CLK(out),
    .J(1),
    .K(1),
    .RST_n(rst_n),
    .Q1(l0)
);

JK_FF f2(
    .CLK(out),
    .J(l0),
    .K(l0),
    .RST_n(rst_n),
    .Q1(l1)
);

and (l2,l0,l1);

JK_FF f3(
    .CLK(out),
    .J(l2),
    .K(l2),
    .RST_n(rst_n),
    .Q1(oQ[2])
);

assign oQ[0]=l0;
assign oQ[1]=l1;

display7 display1(
    .iData(oQ),
    .oData(oDisplay)
);

endmodule

module JK_FF(
    input CLK,
    input J,
    input K,
    input RST_n,
    output reg  Q1,
    output reg  Q2

```

```

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

end
endmodule

module display7(iData,oData);
    input [2:0] iData;    //d2,d1,d0
    output reg [6:0] oData;    //g~a
    always @(*) begin
        case (iData)
            3'b000: oData = 7'b1000000;
            3'b001: oData = 7'b1111001;
            3'b010: oData = 7'b0100100;
            3'b011: oData = 7'b0110000;
            3'b100: oData = 7'b0011001;
            3'b101: oData = 7'b0010010;
            3'b110: oData = 7'b0000010;
            3'b111: oData = 7'b1111000;
            default: oData = 7'b0000000;
        endcase
    end
end
endmodule

```

```

module Divider(
    input I_CLK,
    input rst,
    output O_CLK
);

    parameter N=10000000;
    reg [31:0]n=32'h00000000;
    reg q=0;
    always @(posedge I_CLK or negedge rst )begin
        if(rst==0)begin
            q<=0;
            n<=32'h00000000;
        end else begin
            if(n==(N/2-1)) begin
                q<=~q;
                n<=32'h00000000;
            end else
                n<=n+1;
        end
    end
    assign O_CLK=q;
endmodule

```

(2) 分频器的设计和仿真

功能描述：利用计数器由一个输入时钟信号获得分频后的时钟信号，每一个计数器的脉冲输出频率等于其输入时钟频率除以计数模值。

```

module Divider(
    input I_CLK,
    input rst,
    output O_CLK
);

    parameter N=20;
    reg [31:0]n=32'h00000000;
    reg q=0;
    always @(posedge I_CLK )begin
        if(rst==1)begin
            q<=0;
            n<=32'h00000000;
        end else begin
            if(n==(N/2-1)) begin
                q<=~q;
                n<=32'h00000000;
            end
        end
    end
    assign O_CLK=q;
endmodule

```

```

end else
                                n<=n+1;
                                end
                                end
                                assign O_CLK=q;

endmodule

```

四、测试模块建模

（1）计数器的设计和仿真

```

module Counter8_tb;
    reg CLK;
    reg rst_n;
    wire [2:0] oQ;
    wire [6:0] oDisplay;

    Counter8 uut (
        .CLK(CLK),
        .rst_n(rst_n),
        .oQ(oQ),
        .oDisplay(oDisplay)
    );

    initial begin
        CLK = 0;
        forever #5 CLK = ~CLK; // 10ns 时钟周期
    end

    initial begin
        rst_n = 0;
        #20;
        rst_n = 1; // 释放复位信号

        #200;
        #20;
        rst_n=0;    //测试异步复位
        #10;
        rst_n=1;

        #200;

    end
end

```

(2) 分频器的设计和仿真

```
module Divider_tb;
    reg I_CLK;
    reg rst;
    wire O_CLK;

    Divider uut (
        .I_CLK(I_CLK),
        .rst(rst),
        .O_CLK(O_CLK)
    );

    initial begin
        I_CLK = 0;
        forever #5 I_CLK = ~I_CLK; //周期为 10
    end

    initial begin

        rst = 0;
        #380;

        rst=1; //验证同步复位
        #5;

    end

end
endmodule
```

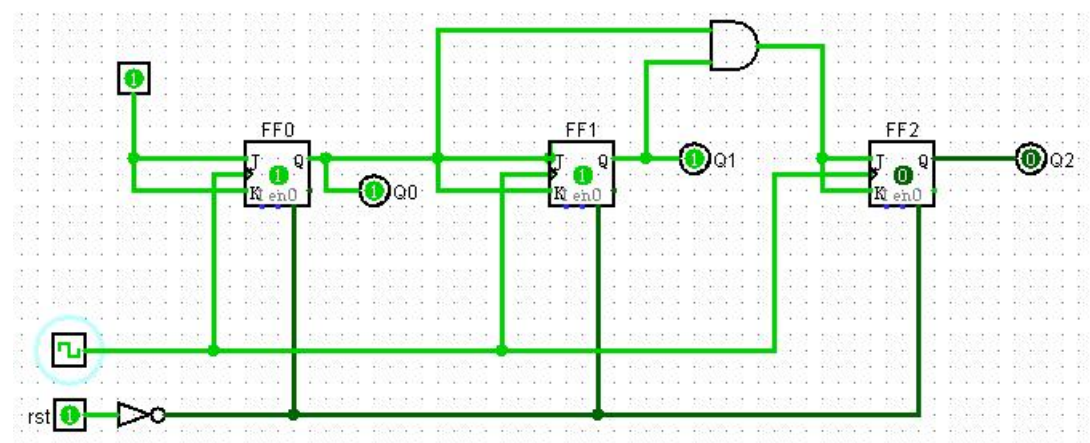
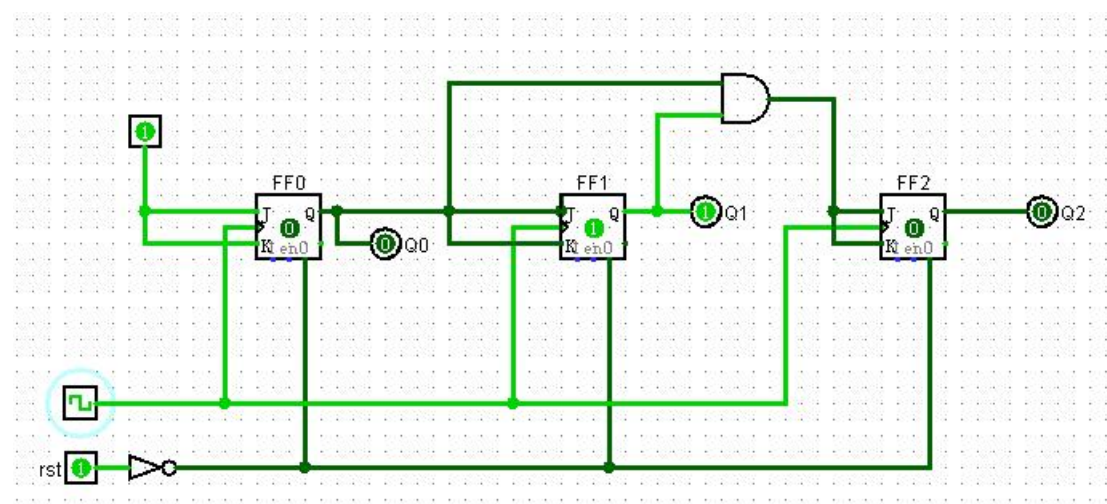
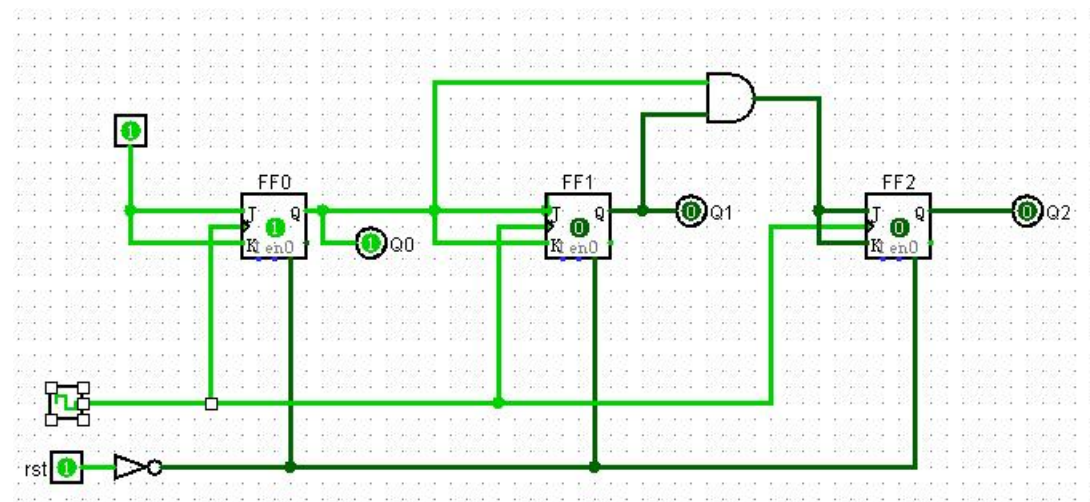
五、实验结果

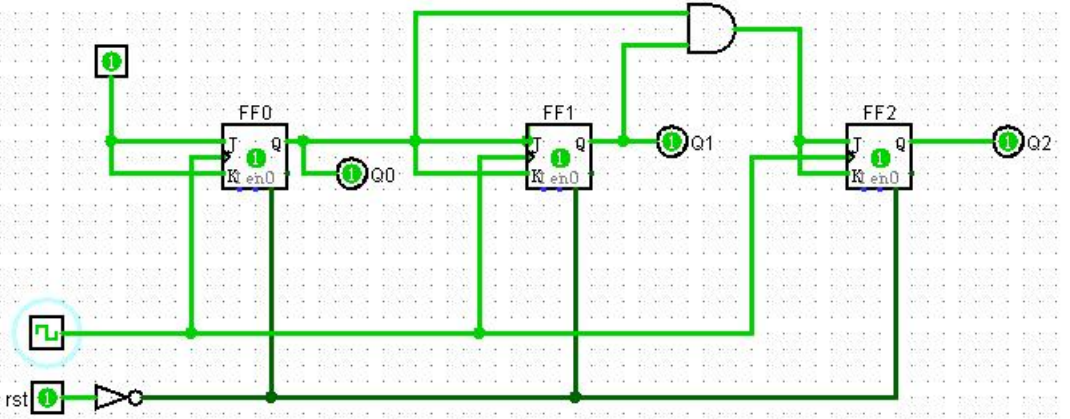
(1) 计数器的设计和仿真

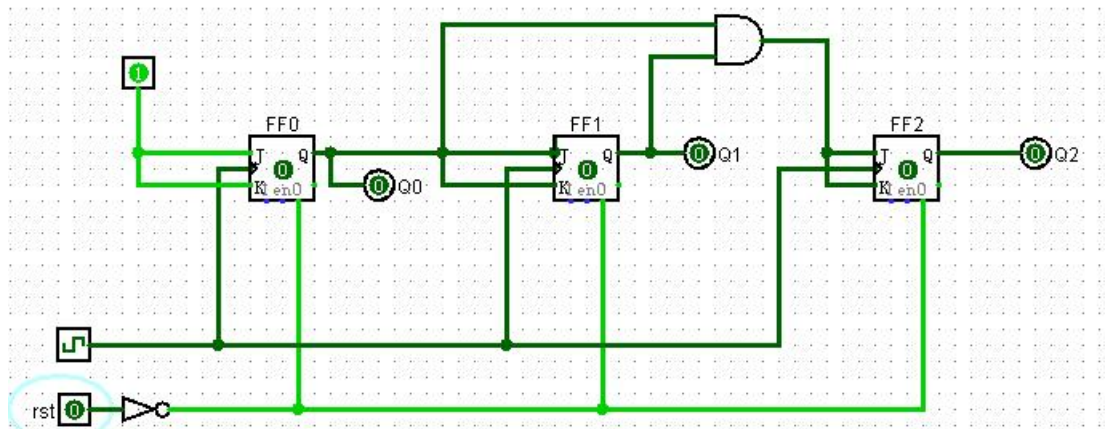
1. logisim 逻辑验证图

表 6.7.1 模 8 计数器的状态转移表

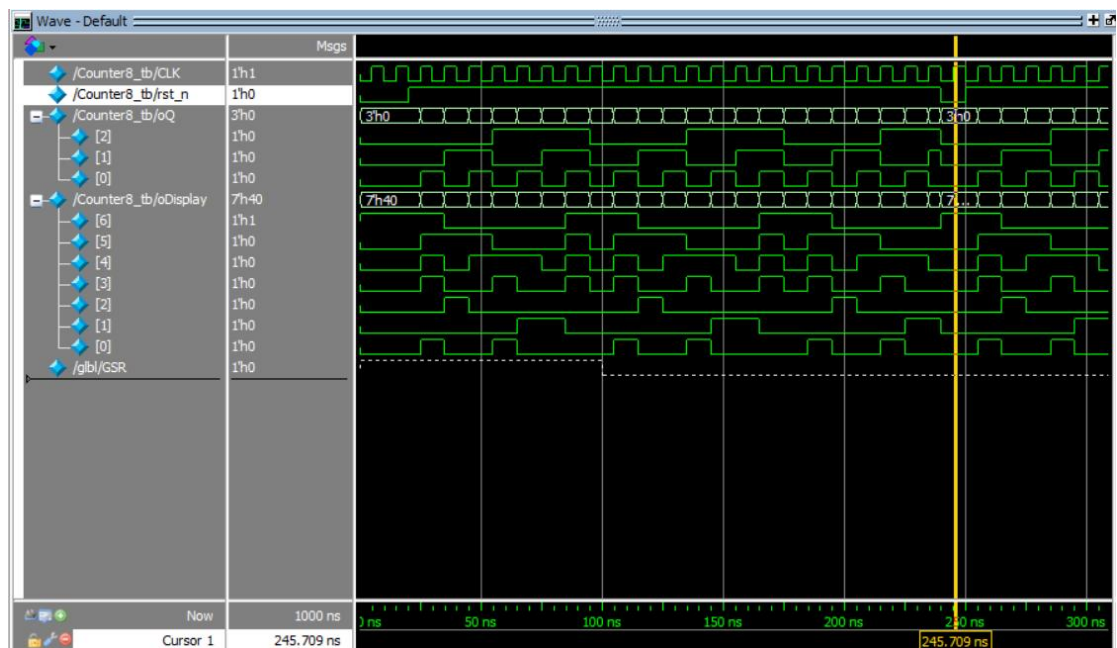
时钟 个数	PS (现 态)			NS (次 态)			时钟 个数	PS (现态)			NS (次态)		
	Q ₂	Q ₁	Q ₀	Q ₂	Q ₁	Q ₀		Q ₂	Q ₁	Q ₀	Q ₂	Q ₁	Q ₀
1	0	0	0	0	0	1	5	1	0	0	1	0	1
2	0	0	1	0	1	0	6	1	0	1	1	1	0
3	0	1	0	0	1	1	7	1	1	0	1	1	1
4	0	1	1	1	0	0	8	1	1	1	0	0	0
							9 (循 环)	0	0	0	0	0	1





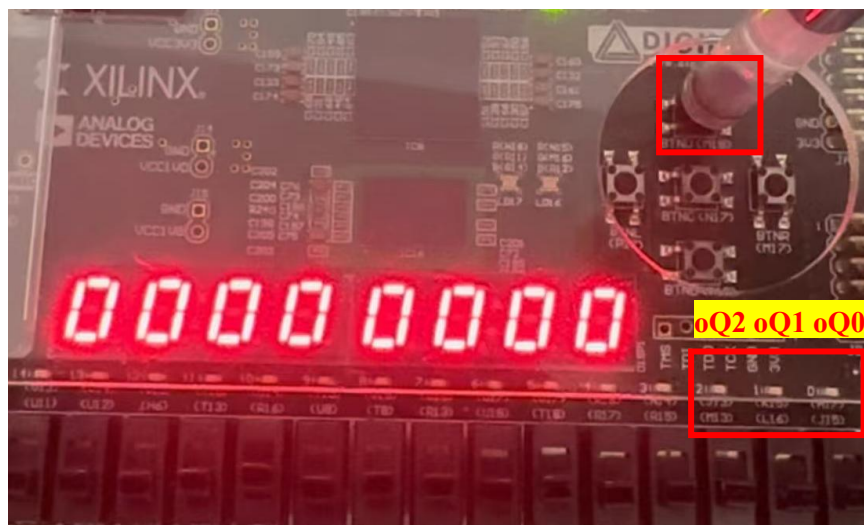


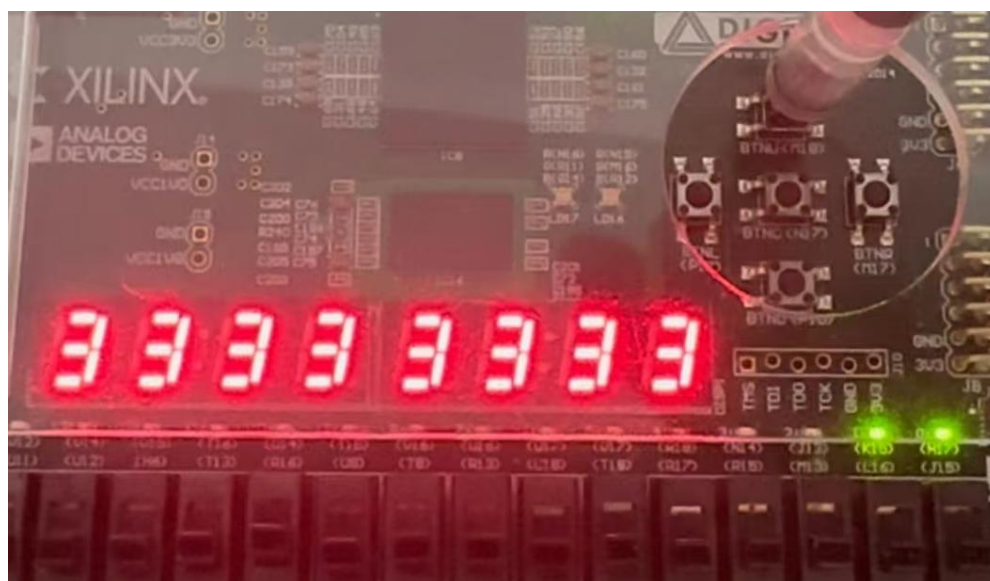
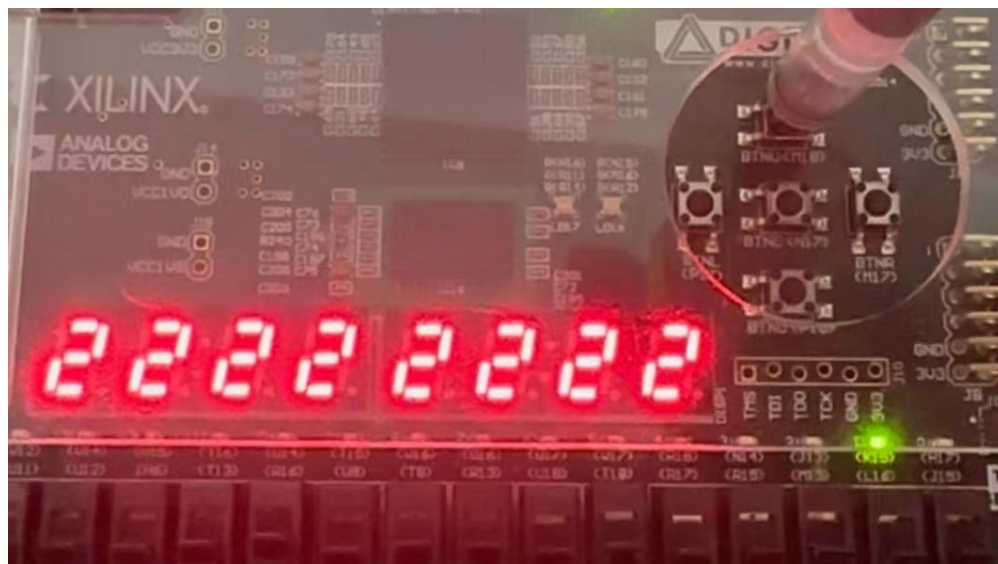
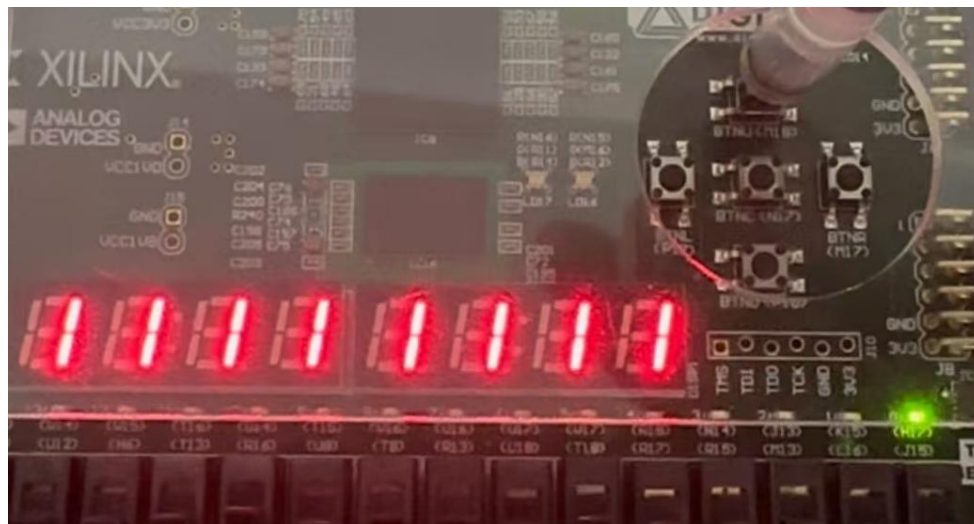
2. modelsim 仿真波形图

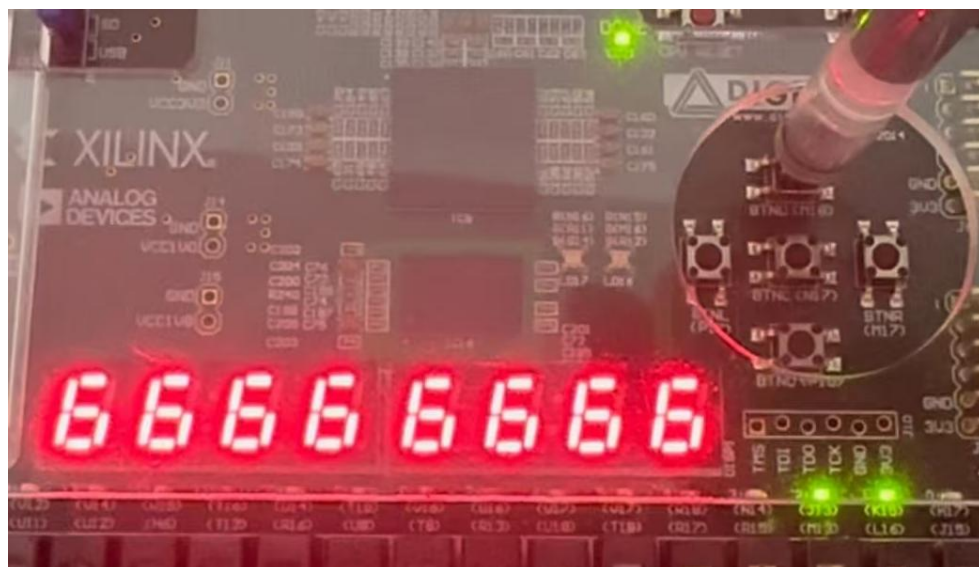
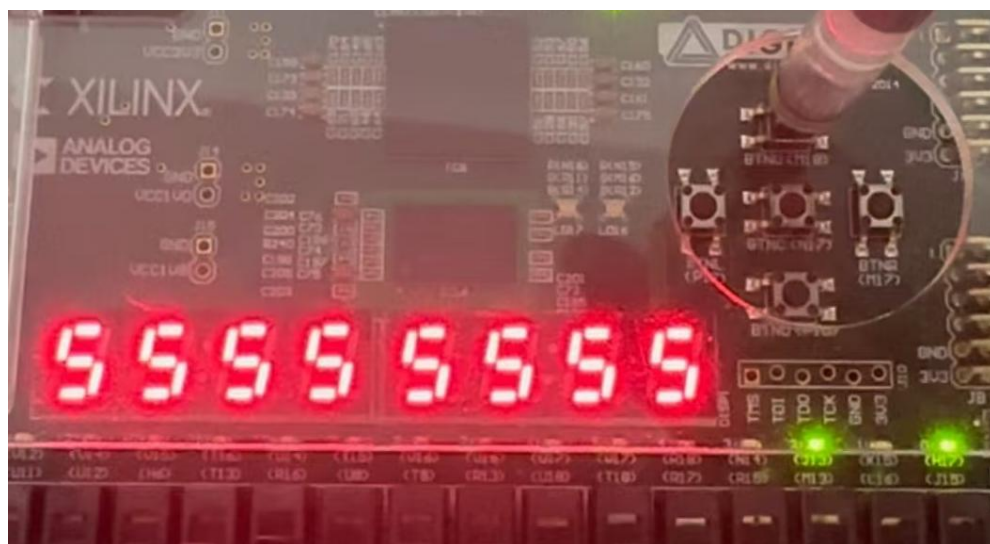
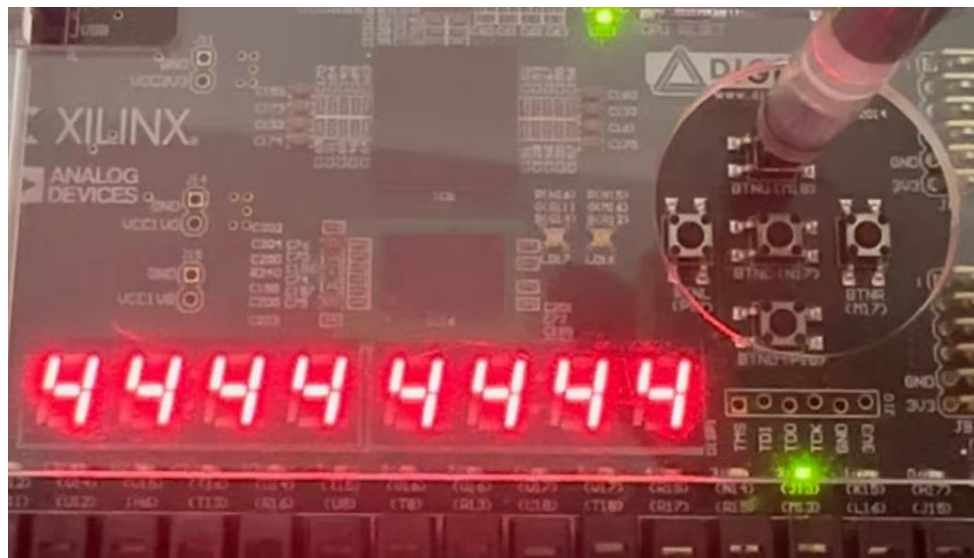


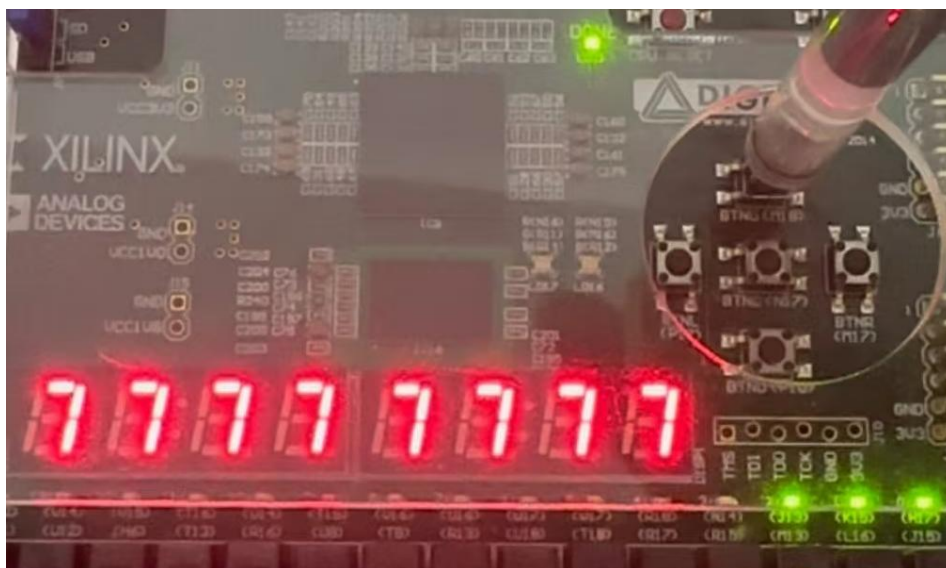
3. 下板实验图

说明：该实验结果是在分频 1000000 倍的情况下拍摄得到的。



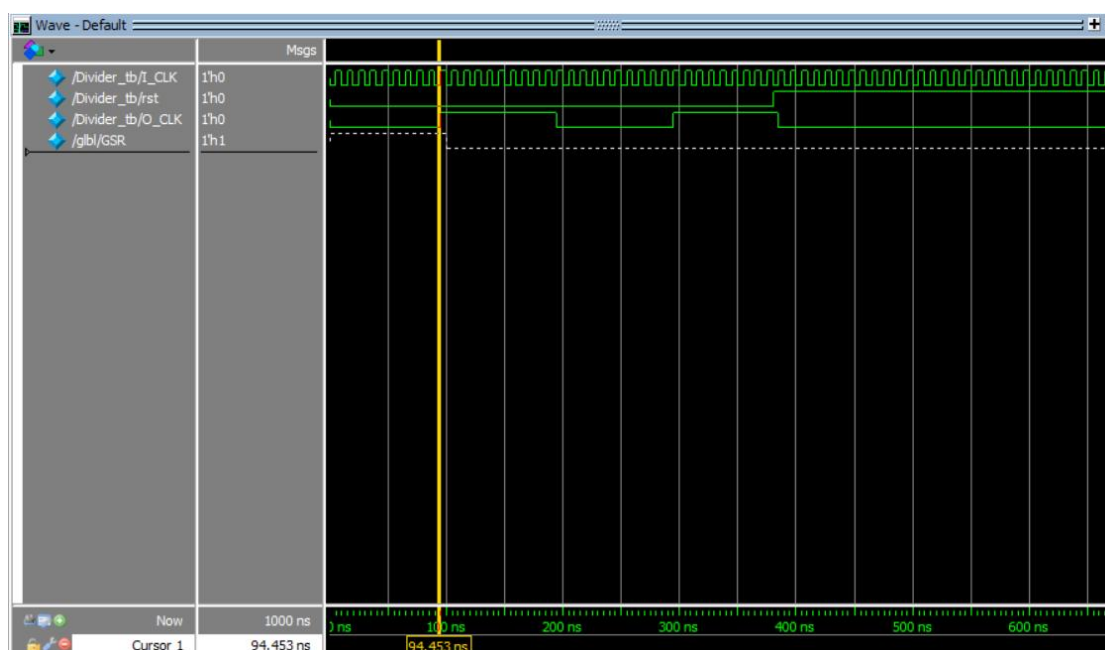






(2) 分频器的设计和仿真

1. modelsim 仿真波形图



Testbench 文件中仿真输入的时钟周期是 10，扩大二十倍后为 200.仿真结果是正确的。

2. 下板实验图

略。因为在实验一中已经运用了分频器，其实验现象足以证明分频器程序的正确性。而且本实验默认分频数为 20，实验现象肉眼并不能观察到时钟闪烁的周期，故没有下板实验的必要性。