

# 浙江大学

## 本科实验报告

课程名称:	数字逻辑设计
姓 名:	王浩雄
学 院:	竺可桢学院
系:	混合班
专 业:	计算机科学与技术
学 号:	3230106032
指导教师:	马德

2025 年 4 月 18 日

# 浙江大学实验报告

课程名称: 数字逻辑设计 实验类型: \_\_\_\_\_  
实验项目名称: 同步时序电路设计  
学生姓名: 王浩雄 专业: 混合班 学号: 3230106032  
同组学生姓名: 无 指导老师: 马德  
实验地点: 紫金港东 4-509 实验日期: 2025 年 4 月 18 日

## 一、 实验目的和要求

- ① 掌握典型同步时序电路的工作原理和设计方法;
- ② 掌握时序电路的激励函数、状态图、状态方程的运用;
- ③ 掌握用 Verilog 进行有限状态机的设计、调试、仿真;
- ④ 掌握用 FPGA 实现时序电路功能。

## 二、 实验内容

- ① 以原理图方式设计 4 位同步二进制计数器
- ② 以 Verilog 行为描述方式设计 16 位可逆二进制同步计数器

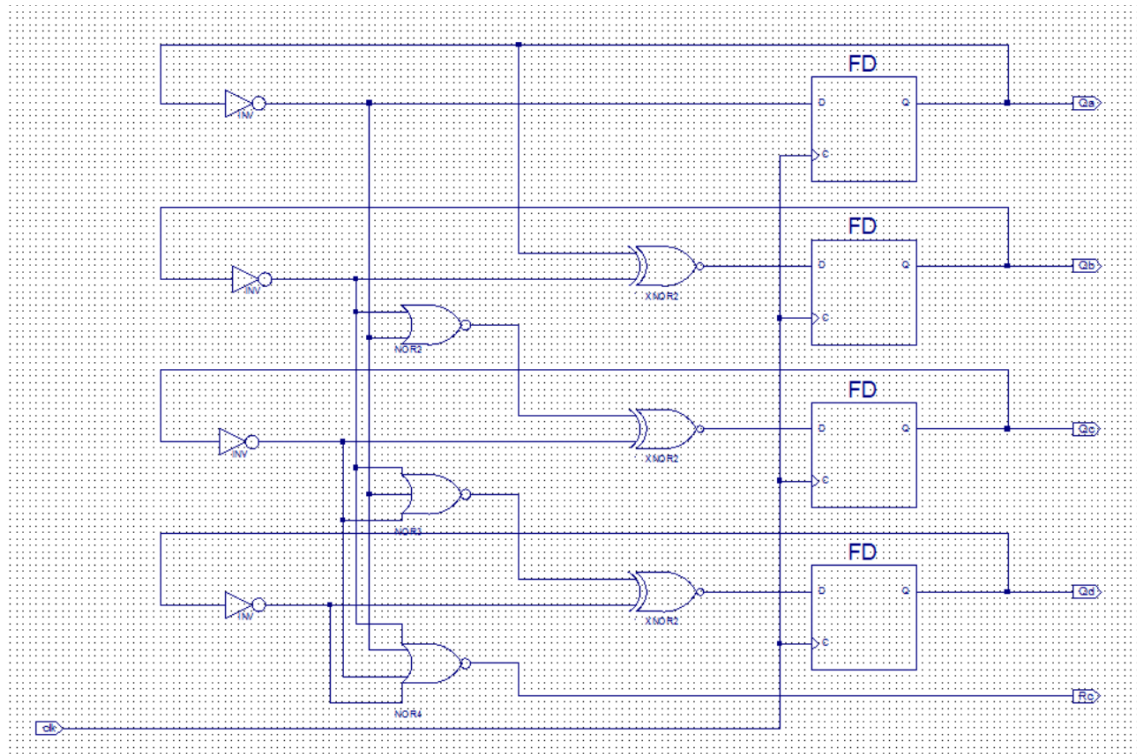
## 三、 主要仪器设备

- ① 装有 ISE 14.7 和 Vivado 2024.2 Enterprise 的计算机 1 台
- ② SWORD 开发板 1 套

## 四、 操作方法与实验步骤

### 1、设计 4 位同步二进制计数器

① 在 Xilinx ISE 中新建 Schematic 文件，结合设计要求和原理图，绘制电路连接图如下：



② 新建仿真激励文件如下：

```
module Counter4b_Counter4b_sch_tb();

    reg clk;

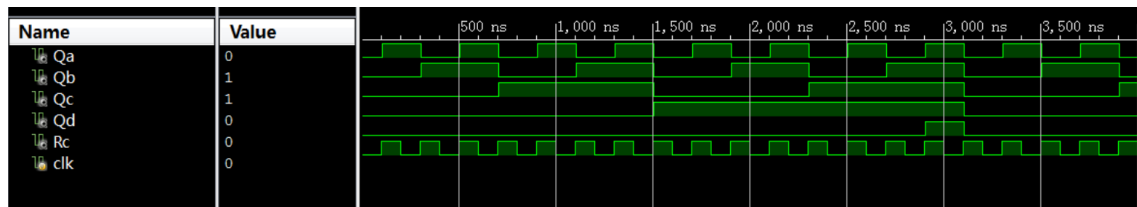
    wire Qa, Qb, Qc, Qd, Rc;

    Counter4b UUT (
        .Qa(Qa), .Qb(Qb), .clk(clk), .Qc(Qc), .Qd(Qd), .Rc(Rc)
    );

    initial forever begin
        clk = 1'b0; #100;
        clk = 1'b1; #100;
    end
endmodule
```

③ 运行仿真，得到如下的仿真波形。在该波形的时钟上升沿处，由 Qd、Qc、Qb、Qa 表示的四位二进制数进行自增，并在归零时输出 Rc 信号。该波形与预期相符，表明原理图绘制的正确性。

（注：由于显示问题，下图的变量名与波形没有对齐）



④ 新建 Verilog 文件，以行为描述方式完成时钟分频器的设计。该模块接收频率为 50MHz 的时钟信号，并将其转化为周期为 1s 的时钟信号。

```
module clk_1s(clk, clk_1s);

input wire clk;
output reg clk_1s;

reg [31:0] cnt;

always @ (posedge clk) begin
    if (cnt < 50_000_000) begin
        cnt <= cnt + 1;
    end
    else begin
        cnt <= 0;
        clk_1s <= ~clk_1s;
    end
end
endmodule
```

⑤ 新建 Verilog 文件，以行为描述方式完成顶层模块的设计。该模块将板载时钟输入到分频器，获得周期为 1s 的时钟并提供给计数器模块，将计数器的 4 位二进制数输出提供给 Lab7 实现的显示模块，将计数器的 Rc 输出用于驱动 LED 的显示。

```
module top(
    input wire clk,
    output wire LED,
    output wire [7:0] SEGMENT,
```

```

        output wire [3:0] AN
    );

    wire Qa, Qb, Qc, Qd, clk_1s;
    wire [15:0] Hex;

    assign Hex = {12'b0, Qd, Qc, Qb, Qa};

    clk_1s u0(.clk(clk),.clk_1s(clk_1s));

    Counter4b u1(.clk(clk_1s), .Qa(Qa), .Qb(Qb), .Qc(Qc), .Qd(Qd), .Rc(LED));

    DispNum
    u2(.clk(clk), .HEXS(Hex), .RST(1'b0), .points(4'b0000), .LES(4'b1110) ,
    .AN(AN), .Segment(SEGMENT));

endmodule

```

⑥ 根据顶层模块的输入输出，完成如下的引脚约束文件。

```

NET "SEGMENT[0]" LOC = AB22 | IOSTANDARD = LVCMOS33;#a
NET "SEGMENT[1]" LOC = AD24 | IOSTANDARD = LVCMOS33;#b
NET "SEGMENT[2]" LOC = AD23 | IOSTANDARD = LVCMOS33;#c
NET "SEGMENT[3]" LOC = Y21 | IOSTANDARD = LVCMOS33;#d
NET "SEGMENT[4]" LOC = W20 | IOSTANDARD = LVCMOS33;#e
NET "SEGMENT[5]" LOC = AC24 | IOSTANDARD = LVCMOS33;#f
NET "SEGMENT[6]" LOC = AC23 | IOSTANDARD = LVCMOS33;#g
NET "SEGMENT[7]" LOC = AA22 | IOSTANDARD = LVCMOS33;#point
NET "AN[0]" LOC = AD21 | IOSTANDARD = LVCMOS33;
NET "AN[1]" LOC = AC21 | IOSTANDARD = LVCMOS33;
NET "AN[2]" LOC = AB21 | IOSTANDARD = LVCMOS33;
NET "AN[3]" LOC = AC22 | IOSTANDARD = LVCMOS33;
NET "LED" LOC = AF24 | IOSTANDARD = LVCMOS33;
NET "clk" LOC=AC18 | IOSTANDARD=LVCMOS18| CLOCK_DEDICATED_ROUTE=TRUE;

```

⑦ 下板，按其功能设计要求进行验证。观察到七段数码管显示 1 位十六进制数字，该数字每秒自增一次，并在归零时亮起 LED 灯。该结果与设计要求和预期相符。

 <p>(归零时 LED 灯亮)</p>	
	
	
	

## 2、设计 16 位可逆同步二进制计数器

① 在 Xilinx ISE 中新建 Verilog 文件，结合设计要求，编写硬件描述代码如下：

```
module counter_16bit_rev(clk, s, cnt, Rc);

input wire clk, s;
output reg [15:0] cnt;
output wire Rc;
```

```

initial cnt = 0;
assign Rc = (~s & (~|cnt)) | (s & (&cnt));

always @ (posedge clk) begin
    if (s)
        cnt <= cnt + 1;
    else
        cnt <= cnt - 1;
end
endmodule

```

② 新建仿真激励文件如下。其中，s 表示计数的正反方向。在测试时，通过手动修改 s 的值分别运行仿真，从而验证其功能正确性。

```

module Counter4b_Counter4b_sch_tb();
module RevCounter_tb;

reg clk, s;

wire [15:0] cnt;
wire Rc;

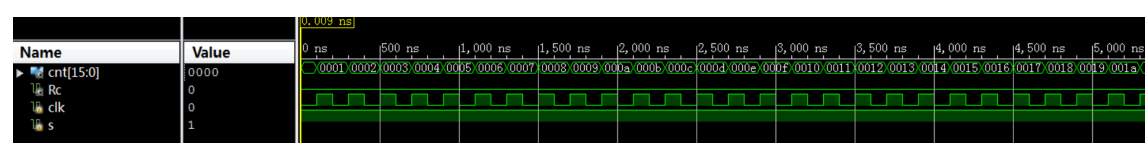
counter_16bit_rev uut (.clk(clk), .s(s), .cnt(cnt), .Rc(Rc));

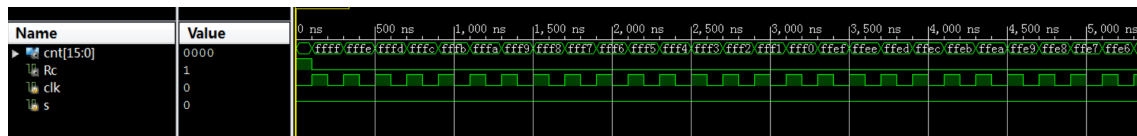
initial forever begin
    s = 1'b0;
    clk = 1'b0; #100;
    clk = 1'b1; #100;
end
endmodule

```

③ 运行仿真，得到如下的仿真波形。在该波形的时钟上升沿处，s=1 时，由 cnt 表示的四位十六进制数进行自增；s=0 时，由 cnt 表示的四位十六进制数进行自减。该波形与预期相符，表明原理图绘制的正确性。

（注：由于显示问题，下图的变量名与波形没有对齐）





④ 新建 Verilog 文件，以行为描述方式完成时钟分频器的设计。该模块接收频率为 50MHz 的时钟信号，并将其转化为周期为 100ms 的时钟信号。

```
module clk_1s(clk, clk_1s);
module clk_100ms(clk, clk_100ms);

input wire clk;
output reg clk_100ms;

reg [31:0] cnt;

always @ (posedge clk) begin
    if (cnt < 5_000_000) begin
        cnt <= cnt + 1;
    end
    else begin
        cnt <= 0;
        clk_100ms <= ~clk_100ms;
    end
end
endmodule
```

⑤ 新建 Verilog 文件，以行为描述方式完成顶层模块的设计。该模块将板载时钟输入到分频器，获得周期为 100ms 的时钟并提供给计数器模块，将计数器的 16 位二进制数输出提供给 Lab7 实现的显示模块，将计数器的 Rc 输出用于驱动 LED 的显示。

```
module top(
    input wire clk,
    input wire SW,
    output wire LED,
    output wire [7:0] SEGMENT,
    output wire [3:0] AN
);

wire clk_100ms;
wire [15:0] Hex;

clk_100ms u0(.clk(clk),.clk_100ms(clk_100ms));
```



```

counter_16bit_rev u1(.clk(clk_100ms), .s(SW), .cnt(Hex), .Rc(LED));

DispNum
u2(.clk(clk), .HEXS(Hex), .RST(1'b0), .points(4'b0000), .LES(4'b0000) ,
  .AN(AN), .Segment(SEGMENT));

endmodule

```

⑥ 根据顶层模块的输入输出，为引脚约束文件增添以下条目。

```
NET "SW" LOC = AA10 | IOSTANDARD = LVCMOS15;
```

⑦ 下板，按其功能设计要求进行验证。观察到七段数码管显示 4 位十六进制数字，拨片开关 SW[0]=1 时该数字每 0.1 秒自增一次，SW[0]=0 时该数字每 0.1 秒自减一次，并在归零时亮起 LED 灯。该结果与设计要求和预期相符。

 <p>(正向计数)</p>	 <p>(正向计数)</p>
 <p>(正向计数)</p>	 <p>(正向计数)</p>

 <p>(归零时 LED 灯亮)</p>	 <p>(反向计数)</p>
 <p>(反向计数)</p>	 <p>(反向计数)</p>

## 五、 实验结果与分析

(请见上方分析)

## 六、 讨论、心得

在本次实验中，我使用画图和行为描述分别实现了 4 位同步二进制计数器和 16 位可逆同步二进制计数器的设计，并成功通过仿真和下板，验证其正确性。本次实验使我对时序逻辑的理解和认识大为加深。此外，通过自己编写顶层模块，我对动态扫描同步输出模块和通用计数分频模块的原理和功能有了更为直观的了解，进一步掌握了分层次、模块化设计的方法。