

浙江大学

本科实验报告

课程名称： 数字逻辑设计

姓 名：

学 院： 计算机科学与技术学院

专 业： 计算机科学与技术

学 号：

指导教师： 马德

2021 年 12 月 4 日

浙江大学实验报告

课程名称：数字逻辑设计

实验项目名称：多路选择器设计与应用

学生姓名：

专业：计算机科学与技术

学号：

指导老师：马德

实验日期：2021年11月15日

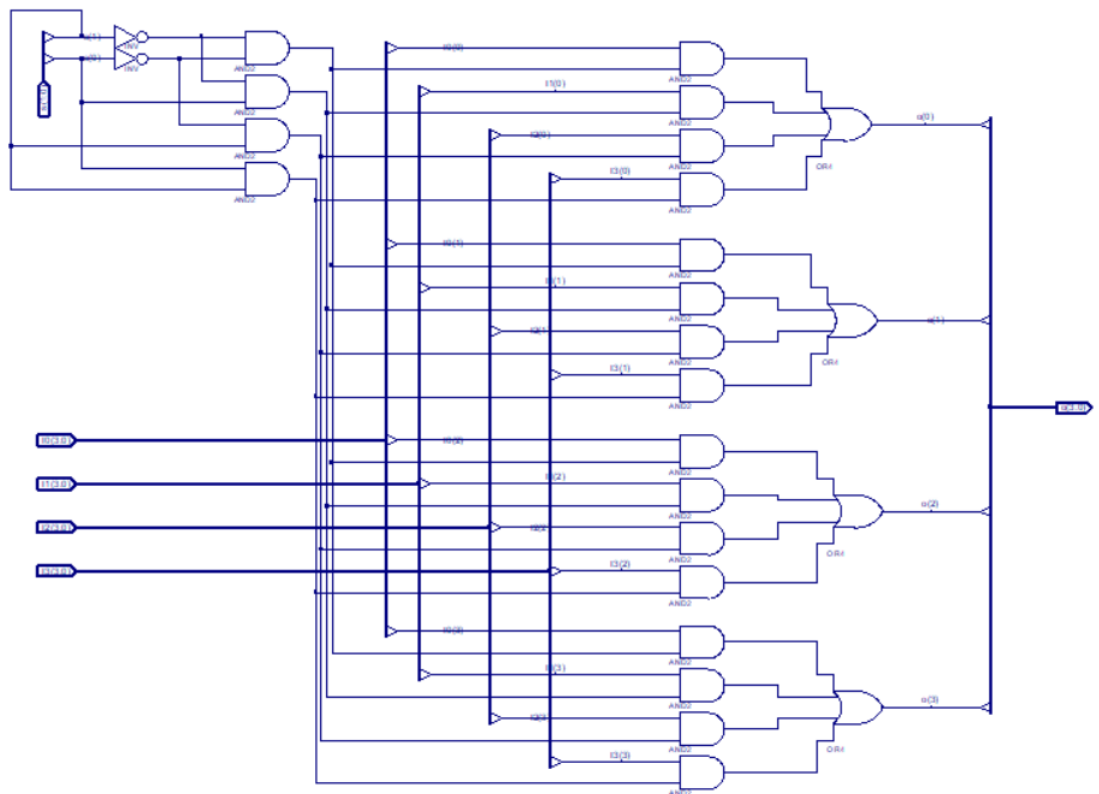
一、实验目的

1. 掌握数据选择器的工作原理和逻辑功能
2. 掌握数据选择器的使用方法
3. 掌握4位数码管扫描显示方法
4. 4位数码管显示应用——记分板设计

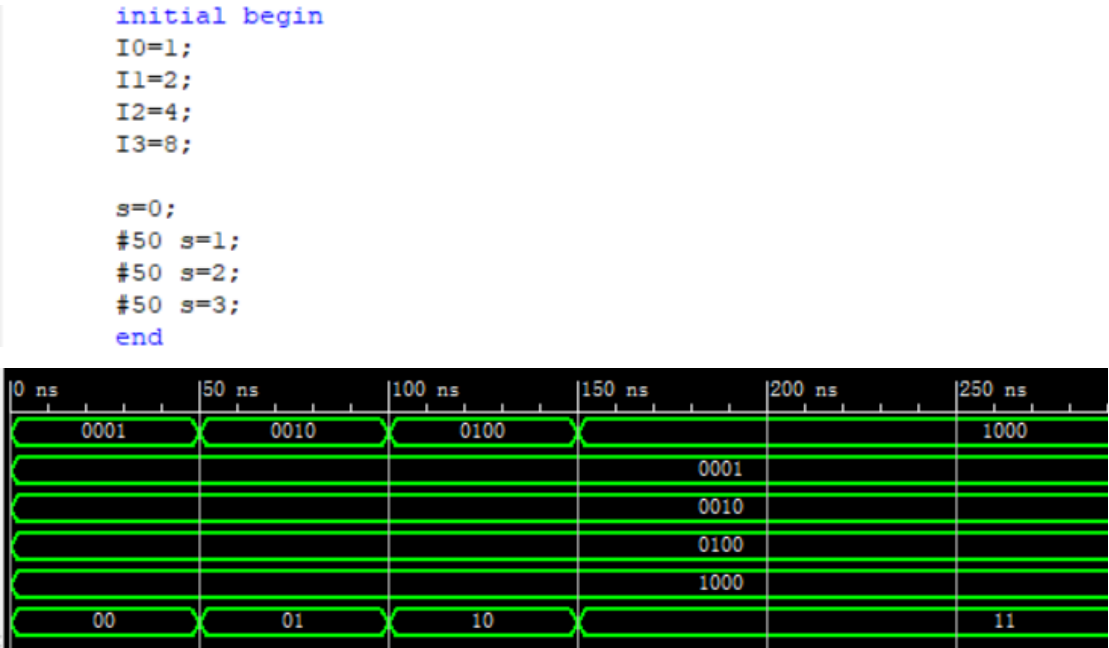
二、操作方法与实验步骤

1. 数据选择器设计




- ① 建立 Mux4to1b4 工程,按如下原理图绘制 Mux4to1b4 器件电路结构。



② 按照参考激励代码进行仿真激励，得到波形图如下。

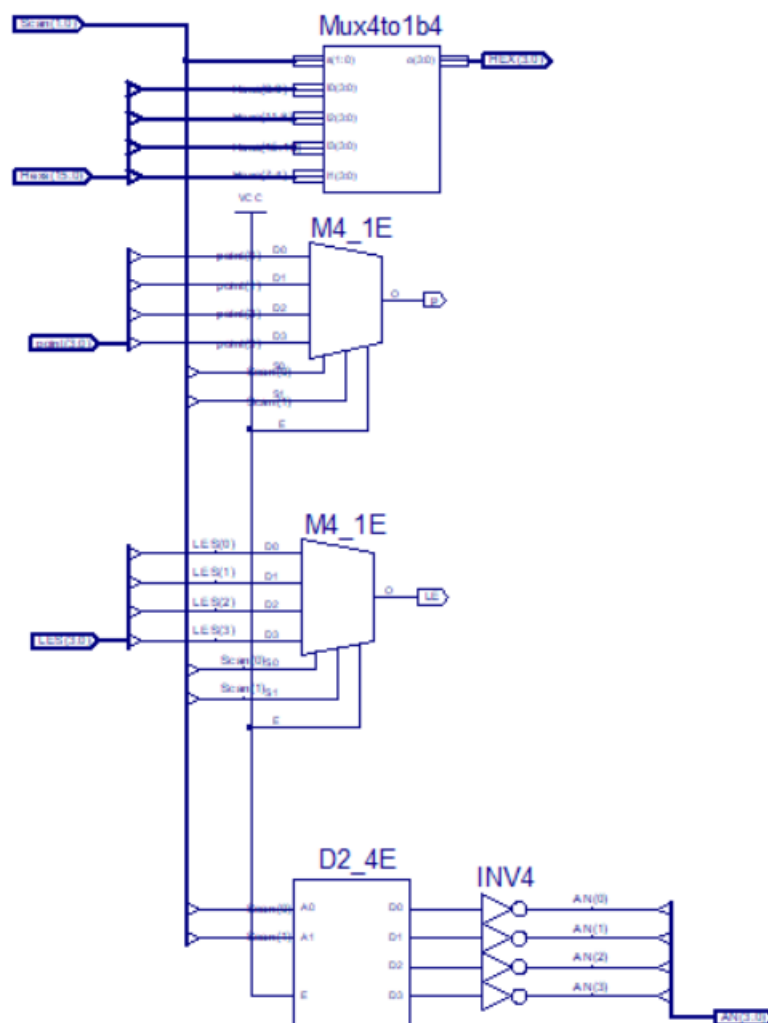


③ 检查无误后，通过 Create Schematic Symbol 生成 Mux4to1b4 器件的逻辑文件。

 Mux4to1b4.sch	2021/11/13 14:00	SCH 文件	30 KB
 Mux4to1b4.schlog	2021/12/5 14:11	SCHLOG 文件	0 KB
 Mux4to1b4.sym	2021/11/13 14:02	SYM 文件	3 KB

2. 记分板应用设计

- ① 建立 ScoreBoard 工程，将 Mux4to1b4 器件、MyMC14495 器件的相关文件复制到工程根目录。
- ② 建立显示同步器 DisplaySync 器件的原理图文件，绘制原理图，并进行检查、实例化，生成该器件。



- ③ 根据示例代码，使用 Verilog HDL 编写时钟分频器 `clkdiv` 器件如下，并进行实例化。

```

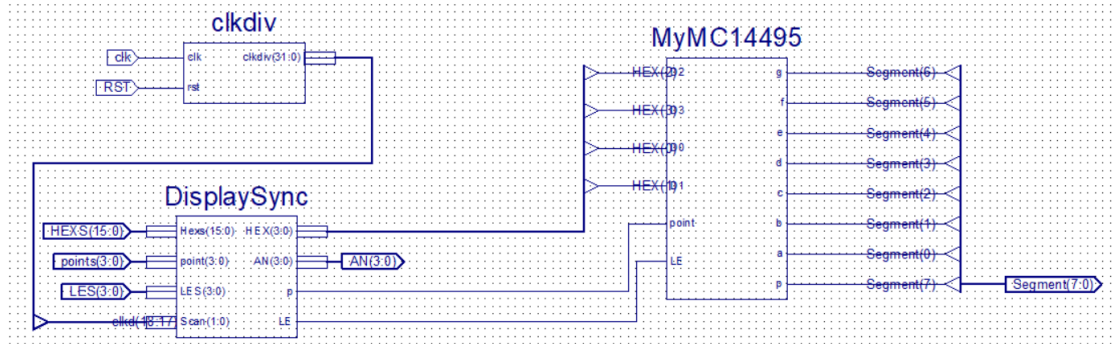
module clkdiv(
    input wire clk,
    input wire rst,
    output reg [31:0] clkdiv
);

    always@(posedge clk or posedge rst) begin
        if(rst) clkdiv<=1;
        else clkdiv<=clkdiv+1'b1;
    end

endmodule

```

- ④ 使用已经构造好的三种器件，按原理图方式嵌套绘制 `DispNum` 器件，如下图，并进行检查。



⑤ 使用 Verilog HDL 编写 CreateNumber 器件，完成计数功能。

```
module CreateNumber(
    input wire [3:0] btn,
    output reg [15:0] num
);

    wire [3:0] A,B,C,D;

    initial num<=16'b1010_1011_1100_1101;

    assign A=num[ 3: 0]+4'd1;
    assign B=num[ 7: 4]+4'd1;
    assign C=num[11: 8]+4'd1;
    assign D=num[15:12]+4'd1;

    always@(posedge btn[0]) num[ 3: 0]<=A;
    always@(posedge btn[1]) num[ 7: 4]<=B;
    always@(posedge btn[2]) num[11: 8]<=C;
    always@(posedge btn[3]) num[15:12]<=D;

endmodule
```

⑥ 根据示例代码，使用 Verilog HDL 编写顶层模块，如下。

```
module top(
    input wire clk,
    input wire [7:0] SW,
    input wire [3:0] btn,
    output wire [3:0] AN,
    output wire [7:0] SEGMENT,
    output wire BTNX4);

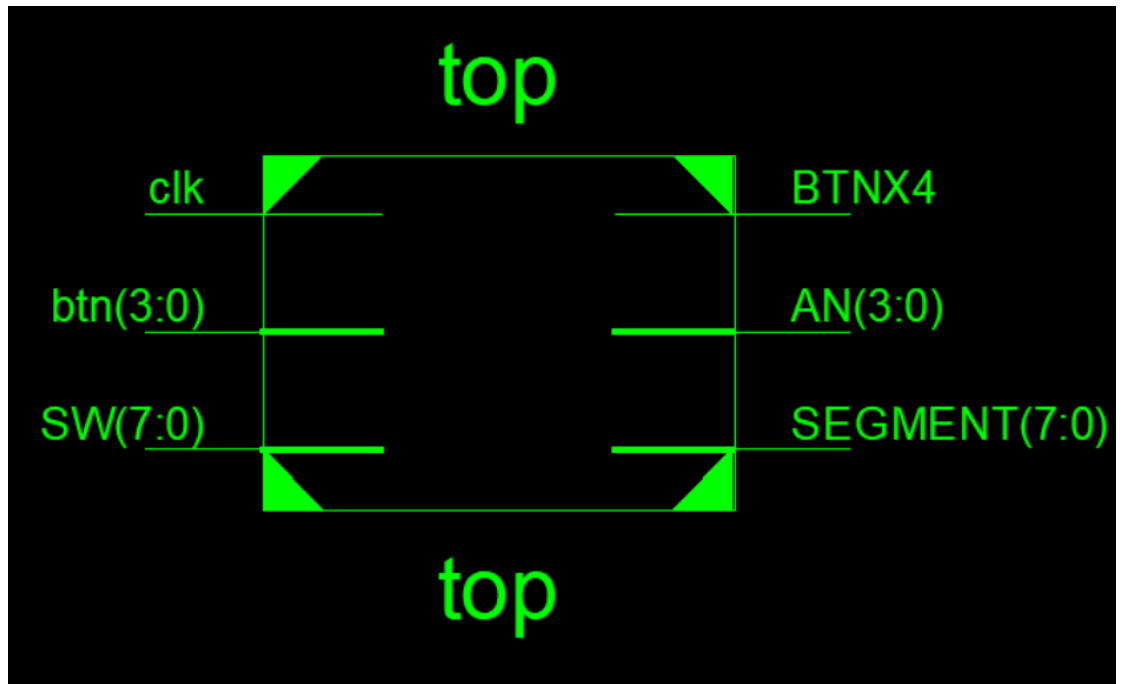
    wire [15:0] num;

    CreateNumber
        c0(btn[3:0],num[15:0]);

    DispNum
        d0(clk,num[15:0],SW[7:4],SW[3:0],1'b0,AN[3:0],SEGMENT[7:0]);

    assign BTNX4=1'b0;//Enable button inputs

endmodule
```



⑦ 建立引脚约束如下，进行综合和模拟设计过程，将编写好的工程下载到实验板上进行验证。

```

1  NET "clk" LOC = AC18 | IOSTANDARD = LVCMOS18;
2  NET "clk" TNM_NET = TM_CLK;
3  TIMESPEC TS_CLK_100M = PERIOD "TM_CLK" 10 ns HIGH 50%;
4
5  NET "SW[0]" LOC = AD10 | IOSTANDARD = LVCMOS15;
6  NET "SW[1]" LOC = AD11 | IOSTANDARD = LVCMOS15;
7  NET "SW[2]" LOC = Y12 | IOSTANDARD = LVCMOS15;
8  NET "SW[3]" LOC = Y13 | IOSTANDARD = LVCMOS15;
9  NET "SW[4]" LOC = AA12 | IOSTANDARD = LVCMOS15;
10 NET "SW[5]" LOC = AA13 | IOSTANDARD = LVCMOS15;
11 NET "SW[6]" LOC = AB10 | IOSTANDARD = LVCMOS15;
12 NET "SW[7]" LOC = AA10 | IOSTANDARD = LVCMOS15;
13
14 NET "btn[0]" LOC = W14 | IOSTANDARD = LVCMOS18;
15 NET "btn[0]" CLOCK_DEDICATED_ROUTE = FALSE;
16 NET "btn[1]" LOC = V14 | IOSTANDARD = LVCMOS18;
17 NET "btn[1]" CLOCK_DEDICATED_ROUTE = FALSE;
18 NET "btn[2]" LOC = V19 | IOSTANDARD = LVCMOS18;
19 NET "btn[2]" CLOCK_DEDICATED_ROUTE = FALSE;
20 NET "btn[3]" LOC = V18 | IOSTANDARD = LVCMOS18;
21 NET "btn[3]" CLOCK_DEDICATED_ROUTE = FALSE;
22 NET "BTNX4" LOC = W16 | IOSTANDARD = LVCMOS18;
23
24 NET "SEGMENT[0]" LOC = AB22 | IOSTANDARD = LVCMOS33;#a
25 NET "SEGMENT[1]" LOC = AD24 | IOSTANDARD = LVCMOS33;#b
26 NET "SEGMENT[2]" LOC = AD23 | IOSTANDARD = LVCMOS33;#c
27 NET "SEGMENT[3]" LOC = Y21 | IOSTANDARD = LVCMOS33;#d
28 NET "SEGMENT[4]" LOC = W20 | IOSTANDARD = LVCMOS33;#e
29 NET "SEGMENT[5]" LOC = AC24 | IOSTANDARD = LVCMOS33;#f
30 NET "SEGMENT[6]" LOC = AC23 | IOSTANDARD = LVCMOS33;#g
31 NET "SEGMENT[7]" LOC = AA22 | IOSTANDARD = LVCMOS33;#point

```

三、实验数据记录和处理

1. 数据选择器设计

经测试，通过原理图方式绘制的 Mux4to1b4 器件能够完成 4 位多路选择器的功能，即能通过 s[1:0]信号完成对 I0、I1、I2、I3 输入信号的选择。00 对应 I0，01 对应 I1，等等。

经模拟过程的波形验证，器件设计无误。

2. 记分板应用设计

经过简单的编写及调试，在实验板上验收时，SW[7-4]对应开关能完成数码管显示与否的转换，SW[3-0]为各位小数点显示与否的转换，BTNX4Y3-BTNX4Y0 的四个按钮为计数器增加。在每一位增加到 F 后，再按下会回到 1。

实验结果经验证无误。

四、实验结果与分析

1. 实验结果正常，经过验收。
2. 注意到在记分板上进行验证时，计数器增长速度较快，可能是因为时钟周期设置较小。

五、心得

继续学习了较为复杂模块的电路设计。

在面对较为复杂电路的实验时，我经过多次尝试，最终选取了自底向上的设计方法。我发现在各器件的设计原理比较明白、容易参照时，这样设计出来的电路逻辑比较清晰，实现也较为简单。