

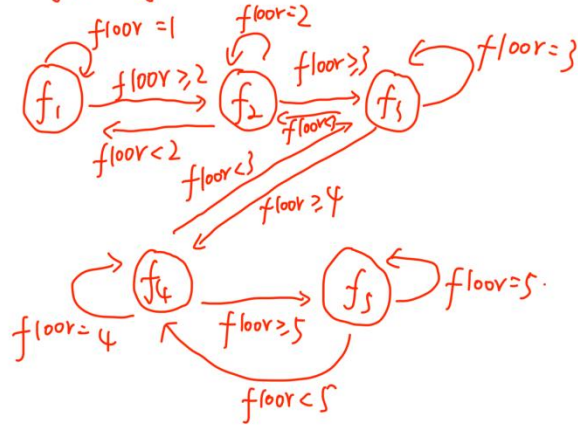
《数字逻辑》实验报告

姓名	文红兵	年级	2021 级
学号	20214590	专业、班级	计卓 2 班
实验名称	电梯控制器设计		
实验时间	2022. 12. 6	实验地点	宿舍
实验成绩		实验性质	<input type="checkbox"/> 验证性 <input type="checkbox"/> 设计性 <input type="checkbox"/> 综合性
<p>教师评价：</p> <p><input type="checkbox"/>算法/实验过程正确； <input type="checkbox"/>源程序/实验内容提交 <input type="checkbox"/>程序结构/实验步骤合理；</p> <p><input type="checkbox"/>实验结果正确； <input type="checkbox"/>语法、语义正确； <input type="checkbox"/>报告规范；</p> <p>评语：</p> <p>评价教师签名（电子签名）：</p>			
<h3>一、实验目的</h3> <p>通过实验，巩固有限状态机设计方法，并设计实现一个电梯控制器</p>			
<h3>二、实验项目内容</h3> <p>利用 BASYS 开发板资源设计一个 5 层楼的电梯控制器系统，并能在开发板上 模拟电梯运行状态，具体要求如下： 1) 利用开发板的 5 个按键作为电梯控制器的呼叫按钮； 2) 利用 led 灯分别显示楼层 1~5 的呼梯状态； 3) 利用数码管显示电梯运行时电梯所在楼层； 4) 利用时钟分频设计电梯控制器控制电梯每秒运行一层。</p>			
<h3>三、实验设计</h3> <p>电梯状态转移图：</p>			

5层楼

$f_1 \sim f_5$: 1~5楼

floor按钮



四、实验过程或算法

电梯有限状态机

```

110 //-----电梯有限状态机
111 parameter f1 = 5'b00001;
112 parameter f2 = 5'b00010;
113 parameter f3 = 5'b00100;
114 parameter f4 = 5'b01000;
115 parameter f5 = 5'b10000;
116 // 初始化
117 always @(posedge clk)
118 begin
119   if (temp_btn == f5) begin tflr <= f5; end
120   else if (temp_btn == f4) begin tflr <= f4; end
121   else if (temp_btn == f3) begin tflr <= f3; end
122   else if (temp_btn == f2) begin tflr <= f2; end
123   else if (temp_btn == f1) begin tflr <= f1; end
124   else begin tflr <= 0; end
125   end
126
127 always @(posedge clk_1s or negedge rst)
128 begin
129   if (rst)
130     state <= f1;
131   else
132     begin
133       case (state)
134         f1: begin
135           if (tflr > f1) begin state <= f2; lamp_out <= tflr; end
136           else

```

```

136 :           else
137 :           begin
138 :               state <= f1;
139 :               lamp_out<=5'b00000;
140 :           end
141 :       end
142 :       f2: begin
143 :           if (tflr > f2) begin state <= f3; lamp_out<=tflr;end
144 :           if(tflr < f2) state <= f1;
145 :           if(tflr ==f2)
146 :           begin
147 :               state <= f2;
148 :               lamp_out<=5'b00000;
149 :           end
150 :       end
151 :       f3: begin
152 :           if (tflr > f3) begin state <= f4;lamp_out<=tflr;end
153 :           if(tflr < f3) state <= f2;
154 :           if(tflr ==f3)
155 :           begin
156 :               state <= f3;
157 :               lamp_out<=5'b00000;
158 :           end
159 :       end
160 :       f4: begin
161 :           if (tflr > f4) begin state <= f5;lamp_out<=tflr;end
162 :           if(tflr < f4) state <= f3;
163 :           if(tflr ==f4)
164 :           begin
165 :               state <= f4;
166 :               lamp_out<=5'b00000;
167 :           end
168 :       end
169 :       f5: begin
170 :           if (tflr != f5) begin state <= f4;lamp_out<=tflr;end
171 :           else begin
172 :               state <= f5;
173 :               lamp_out<=5'b00000;
174 :           end
175 :       end
176 :       default: state <= f1;
177 :   endcase
178 : end
179 : end

```

注解：

1、Lamp_in 是按键的输入，tflr 是目标的楼层，state 是电梯此时在的楼层

2、电梯应该一层一层的走，所以在状态转移的时候需要判断是向上移动还是向下移动

3、一楼和五楼作为特殊情况，只能向上和向下。

4、在二三楼的时候分别判断当前 state 和目标的楼层的大小，慢慢的向上或向下移动一层。如：在 flag 为 4 且 state 为 2 时，那么 state 一层一层移动，先移到 3，再到 4。

按键消抖

```

23 //-----防抖输入模块
24 parameter DELAY_TIME = 1000_000; //延时200ms也就是1000_000个时钟周期
25
26 //信号定义
27 reg[4:0] k0 ; //同步 当前时钟周期输入状态
28 reg[4:0] k1 ; //打拍 前一个时钟周期输入的状态
29 reg[4:0] lamp_in_xiaodou;
30 wire k_xia ; //下降
31 reg[19:0] cnt ; //计数20ms, 需要20ms/20ns = 1000_000个时钟周期
32 reg anfg ; //按下的下降沿标志
33
34 //计数实现
35 always @(posedge clk or negedge rst) begin
36     if(rst == 1'b1) begin
37         k0 <= 5'b00000;
38         k1 <= 5'b00000;
39     end
40     else begin
41         k0 <= lamp_in;
42         k1 <= k0;
43     end
44 end
45 assign k_xia = ~k0 & k1; //检测下降沿
46
47 //cnt 计数器计满1000_000个时钟周期
48 always @(posedge clk or negedge rst) begin
49     if(rst) begin
50         cnt <= 0;
51     end
52     else if(anfg) begin
53         //按下的下降沿标志出现, 则执行
54         if(cnt == DELAY_TIME - 1) begin
55             cnt <= 0;
56         end
57         else begin
58             cnt <= cnt + 1'b1;
59         end
60     end
61 end
62
63 //delay_flag
64 always @(posedge clk or negedge rst) begin
65     if(rst) begin
66         anfg <= 1'b0;
67     end
68     else if(k_xia) begin
69         anfg <= 1'b1;
70     end
71     else if(cnt == DELAY_TIME - 1) begin
72         anfg <= 1'b0;
73     end
74 end
75
76 //lamp_in_xiaodou
77 always @(posedge clk or negedge rst) begin
78     if(rst) begin
79         lamp_in_xiaodou <= 5'b00000;
80     end
81     else if(cnt == DELAY_TIME - 1) begin
82         lamp_in_xiaodou <= ~k0;
83     end
84     else begin
85         lamp_in_xiaodou <= 5'b00000;
86     end
87 end

```

寄存器储存按下的值

```

88 //-----寄存器储存按下的值
89 reg [4:0] temp_btn;
90 always @(posedge clk)
91 begin
92     if(lamp_in_xiaodou)
93     begin
94         temp_btn<=lamp_in_xiaodou;
95     end
96 end

```

一秒分频器

```

97 //-----1秒分频器
98 reg clk_1s;
99 reg[27:0] tmp;
100 always @(posedge clk)
101 begin
102     if(tmp>=28'd50000000)
103     begin
104         tmp <= 28'd0;
105         clk_1s <= ~clk_1s;
106     end
107     else
108         tmp <= tmp + 1'b1;
109 end

```

七段数码管显示

```

23 module show(input clk,input [4:0]state,output reg[3:0] an,output reg[6:0] a_to_g);
24     reg[1:0] sign;
25     parameter update_interval = 50000000 / 160 ;
26     integer selcnt;
27     always @(posedge clk) //分频160Hz
28     begin
29         selcnt <= selcnt + 1;
30         if (selcnt == update_interval)
31         begin
32             selcnt <= 0;
33             sign <= sign+ 1;
34         end
35     end
36     always @(*) //选择位
37     case(sign)
38         0:begin an=4'b1110;end
39     endcase
40     always @(*)//显示段数
41     case(state)
42         5'b00001:a_to_g=7'b1001111;
43         5'b00010:a_to_g=7'b0010010;
44         5'b00100:a_to_g=7'b0000110;
45         5'b01000:a_to_g=7'b1001100;
46         5'b10000:a_to_g=7'b0100100;
47         default:a_to_g=7'b0000000;
48     endcase
49 endmodule

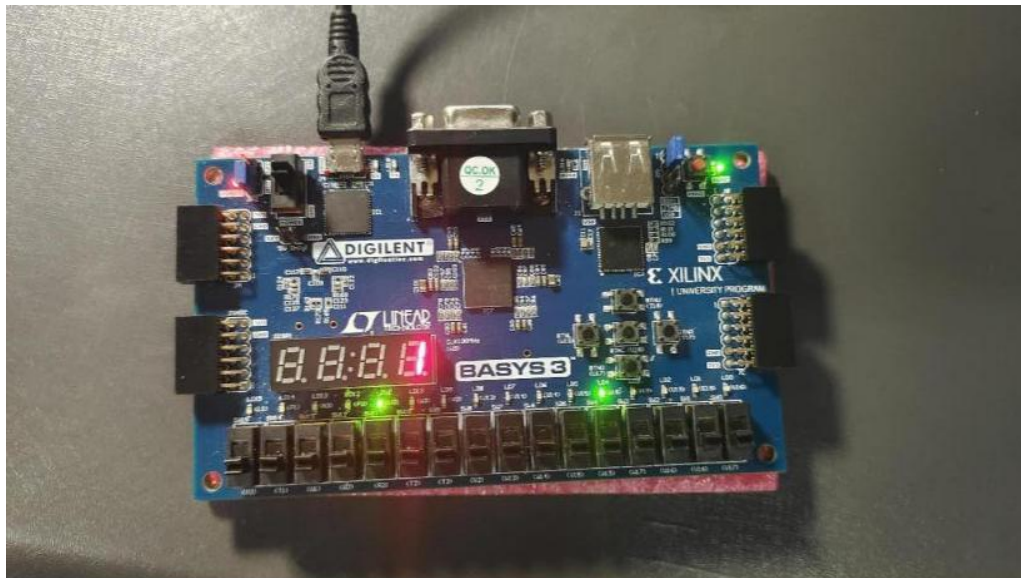
```

五、实验过程中遇到的问题及解决情况

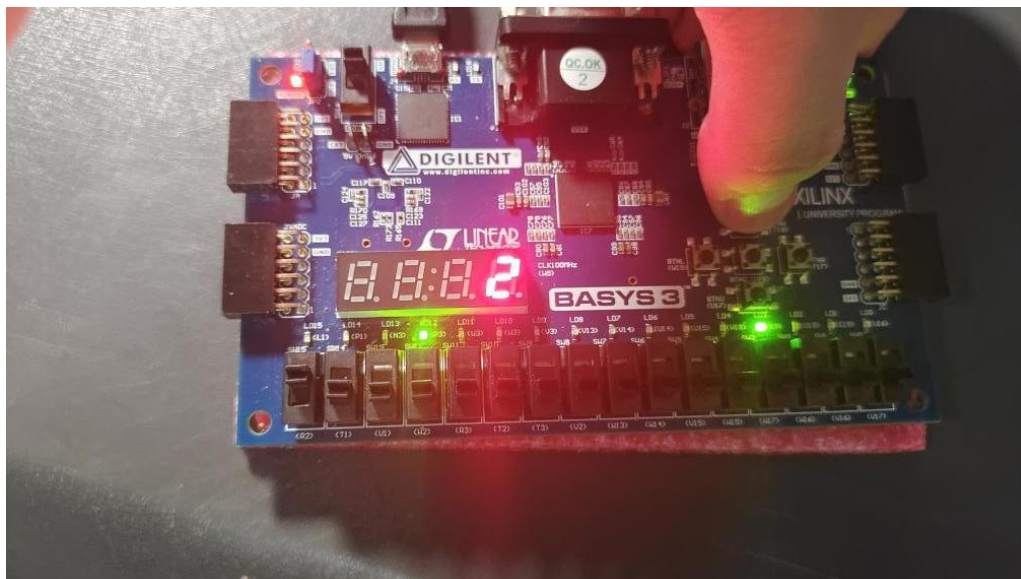
在按键防抖模块中，需要设置一个寄存器储存按键按下来的值，否则就会出现要一直按着按键才能触发代码

六、实验结果及分析和（或）源程序调试过程

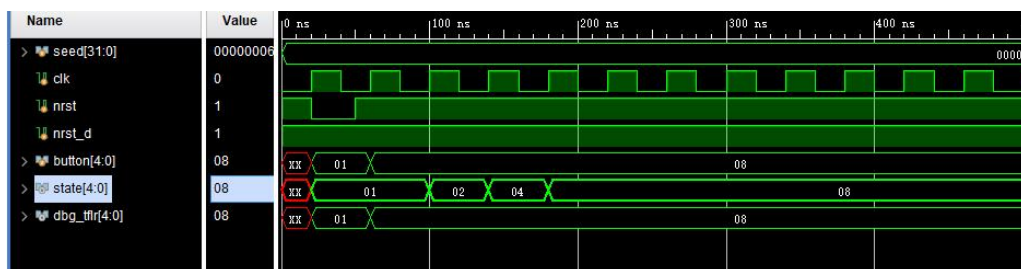
下面是默认情况，电梯在一楼



下面是按着按键后电梯的跳转情况：



下面是代码仿真：



可以看到，楼层状态 state 是一层一层向上的。由于楼层设为 00001、00010、00100....所以在十进制下为 1、2、4、8.

前面红色的是没有进行初始化

七、小组分工情况说明

二个人一起讨论总体思路，并且一起给出电梯的状态转移图和按键消抖输入的思路