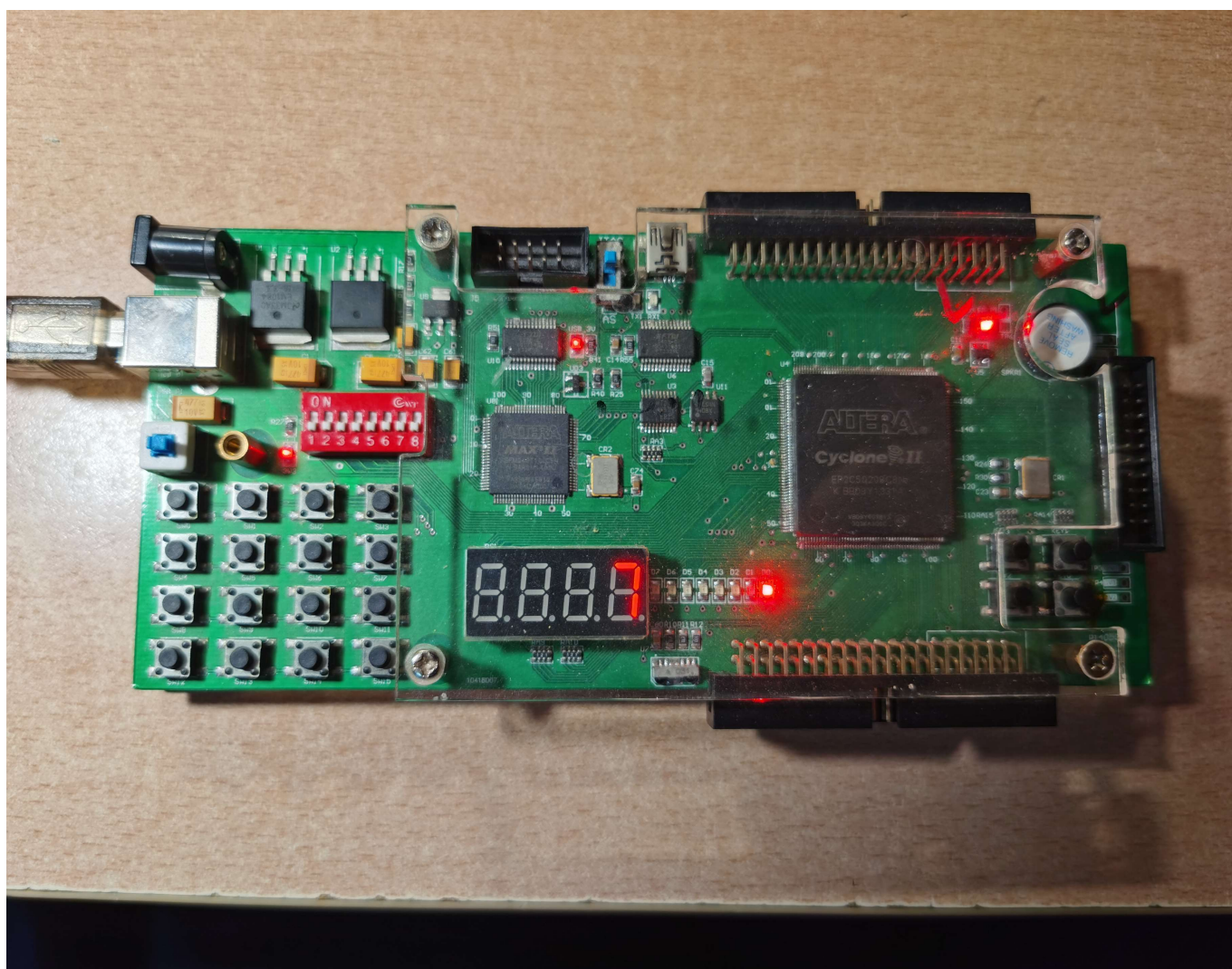


EDA03实验报告



自动化系

李昭阳 2021013445

充电仪模块设计与选择

(1) 分频器Clock

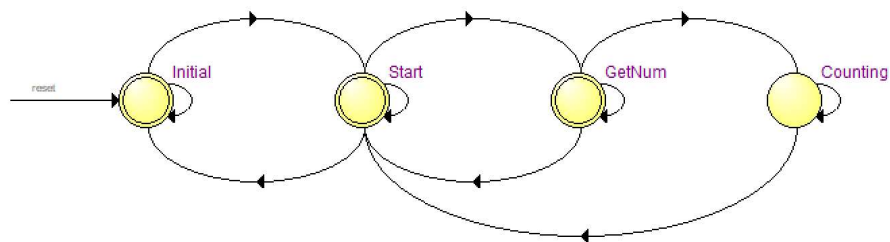
分频器模块的作用是将FPGA板的晶振产生的 50MHZ 的高频时钟信号降成 10000HZ 和 250HZ 的适合充电仪工作的较低频时钟信号，代码如下。

```
module Clock(input wire BoardCLK, output reg clk_250Hz, output reg clk_10000Hz);
initial clk_250Hz=1'b0;
initial clk_10000Hz=1'b0;
reg [17:0] i=18'b0;
reg [17:0] j=18'b0;
always @(posedge BoardCLK)
begin
    if(i==100000)
    begin
        clk_250Hz <= ~clk_250Hz;
        i<=18'b0;
    end
    else
        i<=i+1;
    end
end

always @(posedge BoardCLK)
begin
    if(j==2500)
    begin
        clk_10000Hz<=~clk_10000Hz;
        j<=18'b0;
    end
    else
        j<=j+1;
    end
end
endmodule
```

(2) 键盘扫描模块KeyBoard

该键盘扫描实现是，在每一个时钟信号下扫描键盘中的一列，如果这一列对应的行有某位为低电平，则将代表前一时钟键盘有按下的变量，last置为1。在接下来的时间内，如果该按键能够维持20次时钟脉冲，则认为该信号不是按键抖动的误触，实现了防抖。按键信息到输出数据的转换，是通过case语句进行实现的。大部分逻辑相似，故给出部分精简代码如下。



(4) 显示模块Show

该模块的作用是实时地将投币数和倒计时数显示在数码管上（Initial状态不显示）。通过高频选通信号扫描，利用视觉残留，使得数码管“假并真串”使数码管同时显现四个不同数字。代码如下。

```

module Show(input wire clk, input [19:0] TimeLeft, input [19:0] Money, output reg [3:0] DIG, output reg [3:0] Num);
initial DIG=4'b0000;
initial Num=4'b0000;
always @(posedge clk)
begin
    case(DIG)
        4'b1000:
            begin
                DIG<=4'b0001;
                Num<=TimeLeft%10;
            end
        4'b0001:
            begin
                DIG<=4'b0010;
                Num<=TimeLeft/10;
            end
        4'b0010:
            begin
                DIG<=4'b0100;
                Num<=Money%10;
            end
        4'b0100:
            begin
                DIG<=4'b1000;
                Num<=Money/10;
            end
        default:
            begin
                DIG<=4'b0001;
                Num<=0;
            end
    endcase;
end
endmodule
  
```

(5) 音乐播放模块Music

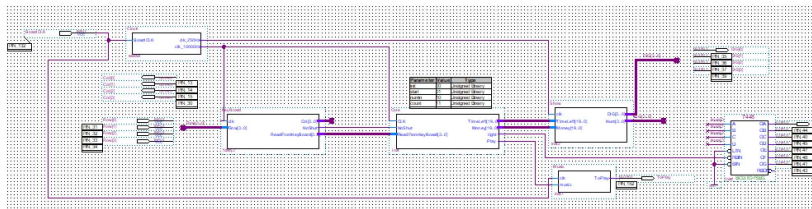
该模块的作用是播放一段20s的音乐，部分代码如下。

```

module Music(input wire clk, input music, output reg ToPlay);
initial ToPlay=1'b0;
localparam M1=95600;
localparam M2=85150;
localparam M3=75850;
localparam M5=63750;
localparam M6=56800;
localparam M7=50600;
localparam H1=47750;
reg [16:0] cnt0=0;
reg [10:0] cnt1=0;
reg [5:0] cnt2=0;
reg [16:0] temp=0;
wire [16:0] midtemp;
reg [10:0] num=0;
wire [10:0] musicnum;
reg [10:0] nota=60;
assign midtemp=temp/2;
assign musicnum=num*7/8;
always @(posedge clk)
begin
    if(music==0)
    begin
        cnt0<=0;
        cnt1<=0;
        cnt2<=0;
        ToPlay<=1'b0;
    end
    else
    begin
        case (cnt2)
            0:temp=M1;
            1:temp=M2;
            2:temp=M3;
            3:temp=M5;
            4:temp=M3;
            5:temp=M2;
            6:temp=M1;
            7:temp=M1;
            8:temp=M1;
            9:temp=M2;
            10:temp=M3;
        endcase
        cnt0<=midtemp;
        if(cnt0<cnt0)
        begin
            ToPlay<=1'b1;
        end
        else
        begin
            ToPlay<=1'b0;
        end
        if(cnt0==temp)
        begin
            cnt0<=0;
            if(cnt1==num)
            begin
                cnt1<=0;
                if(cnt2==nota-1)
                begin
                    cnt2<=0;
                end
                else
                begin
                    cnt2<=cnt2+1;
                end
            end
            else
            begin
                cnt1<=cnt1+1;
            end
        end
        else
        begin
            cnt0<=cnt0+1;
        end
    end
end
endmodule

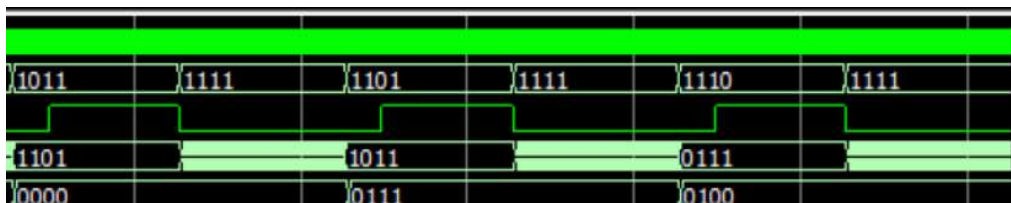
```

(8) 整体设计图

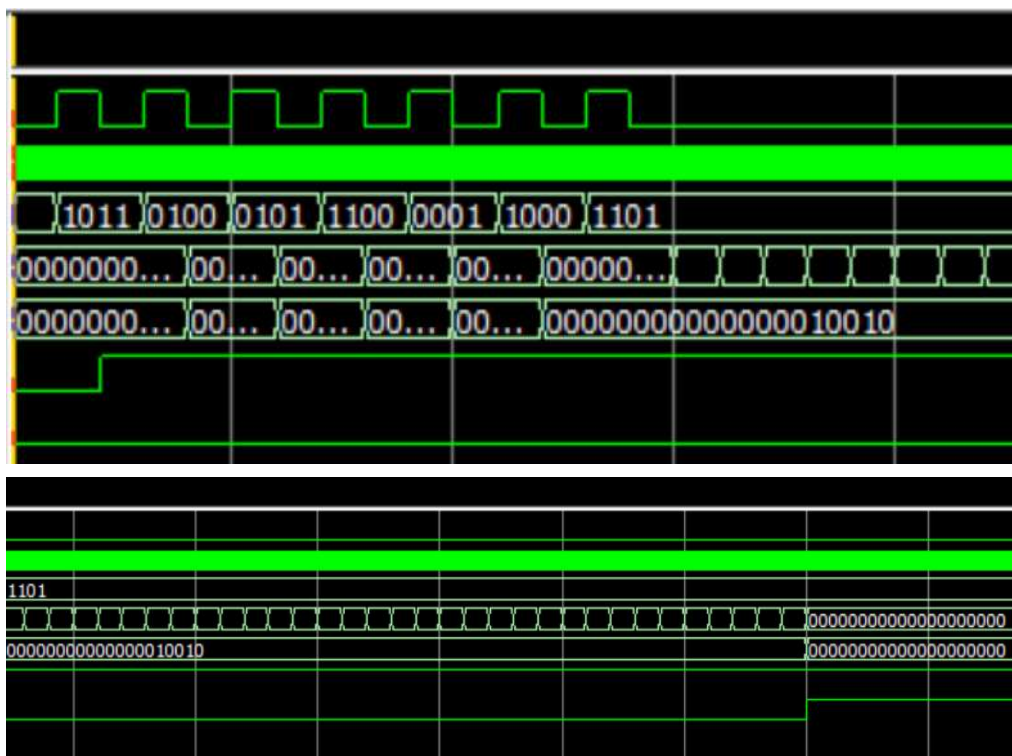


波形图与验证文件

键盘扫描模块仿真



核心模块仿真



图中为开始、长按4、长按5、清零、长按1、长按8、确认。当按下开始键（“1011”）后进入数字输入状态。长按数字键（“0100”和“0101”），输出相应的金额和时间；按下清零键（“1100”），金额和时间被清零；再按下数字键（“0001”和“0100”）重新输入金额和时间。仿真符合预期的需求。按下确认键（“1101”）后，进入计时状态，充电时间逐渐减小，减为0时，播放音乐。音乐播放结束，变回输入状态，然后经过10s，数码管熄灭，变为初始状态。仿真与预期相符，设计合理。

总结

通过本次实验，我对verilog语言的理解更加深刻，也逐渐脱离原理图进行纯语言的设计。我相信这次EDA3作业培养了我可编程逻辑器件的操作能力，同时在解决bug的过程中也锻炼了我的耐心和查阅资料的能力。

