

《EDA技术》试验

实验三

组合逻辑电路设计、仿 真与实现

专业： 计算机科学与技术
姓名： 高星杰
学号： 2021307220712
班级： 计科 2102

报告上交时间：2023年 4月 3日

一、实验目的

- 1、掌握常用组合逻辑电路的 EDA 设计方法；
- 2、熟练掌握基于 QuartusII 集成开发环境的组合逻辑电路设计流程；
- 3、加深对 VerilogHDL 语言的理解；
- 4、熟练掌握 DE2-115 开发板的使用方法；
- 5、掌握多层电路在 QuartusII 集成开发环境中的实现；

二、实验任务及要求

任务一：4-7BCD 码译码器设计/仿真/与实现

要求：用 DE2-115 开发板上的四个拨动开关(从板子上的 18 个拨动开关中任选 4 个)输入 BCD 码，用 DE2-115 开发板上的一个七段数码管（从 8 个七段数码管中任选一个）显示译码结果。

任务二：4 位全加器的设计、仿真与实现

要求：(1) 将两个输入数据和运算结果均显示在 DE2-115 开发板的七段数码管上；用拨动开关输入数据。

(2) 分两个模块实现：运算模块、显示模块。运算模块是主模块，调用显示模块实现输入和结果显示。

任务三：4 位比较器设计与实现

要求：比较器要求用开关输入，用 led 指示比较结果

附加任务：3-8 译码器设计/仿真/与实现

要求：选用 DE2-115 开发板上的 3 个拨动开关(从板子上的 18 个拨动开关中任选 4 个)输入，选用 DE2-115 开发板上的 8 个 LED 灯（从 18 个 led 红灯中任选 8 个）显示译码结果。

三、实验原理与步骤

任务一：4-7BCD 码译码器设计/仿真/与实现

原理分析：4-7BCD 译码器的原理就是将 4 位输入的 BCD 码转换成 7 端数码管的信号。本次实验用的开发板的 bcd 码是共阳极，所以是低电平亮。然后根据真值表，就可以设计出 Verilog 程序，最后下载到板子上就可以实现了。

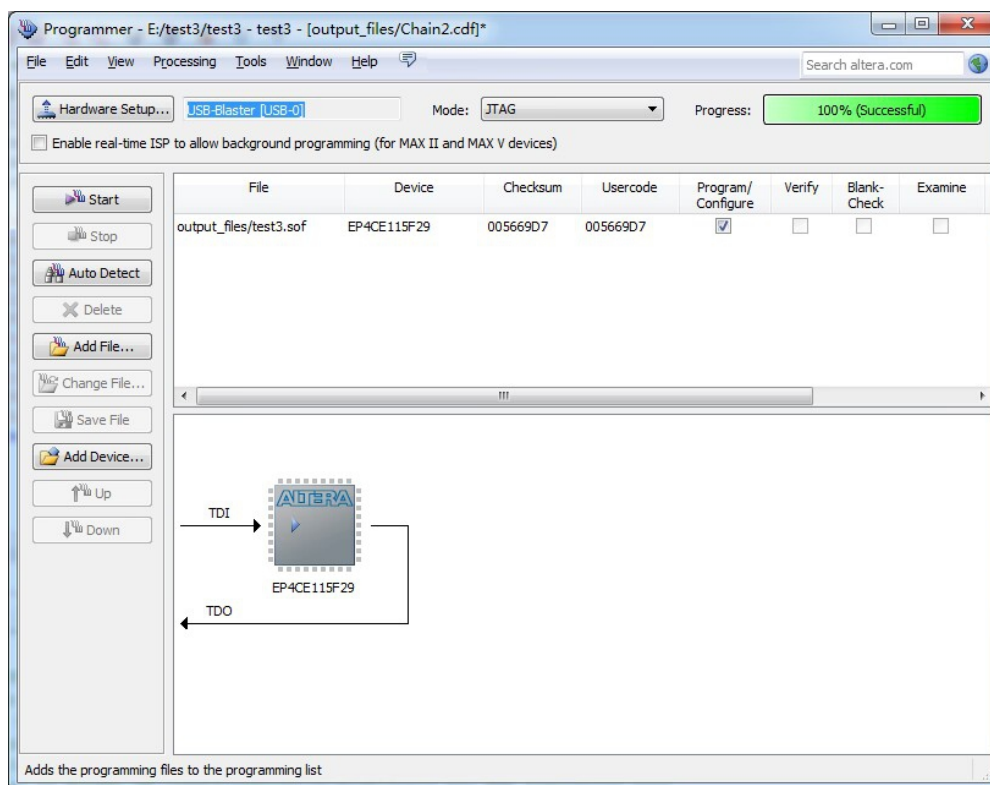
设计的程序：

```
module test3(in,out);
  (*chip_pin="AB28,AC28,AC27,AD27"*)input [3:0] in;
  (*chip_pin="H22,J22,L25,L26,E17,F22,G18"*)output reg[6:0] out;
  always@(in)
  begin
    case(in)
      4'b0000:out=7'b1000000;
      4'b0001:out=7'b1111001;
      4'b0010:out=7'b0100100;
      4'b0011:out=7'b0110000;
      4'b0100:out=7'b0011001;
      4'b0101:out=7'b0010010;
      4'b0110:out=7'b0000010;
      4'b0111:out=7'b1111000;
      4'b1000:out=7'b0000000;
      4'b1001:out=7'b0010000;
    endcase;
  end
endmodule
```

编译成功：

| Task | |
|------|---------------------------|
| ✓ | ▶▶ Compile Design |
| ✓ | ▶▶ Analysis & Synthesis |
| ✓ | ▶▶ Fitter (Place & Route) |

下载到板上：



任务二：4 位全加器的设计、仿真与实现

实验原理：分两个模块写，一个是展示模块负责把二进制的数转换为 7 段数码管上的数展示出来，另一个是计算模块负责把 4 位全加器的功能实现包括输入的进位，输出的进位等。

难点：因为是四位全加器，所以范围包括十位然后调用展示模块时需要将十位和个位分开来。

Verilog 程序：计算模块：

```
module test32(in1,in2,out1,out12,out2,out22,out3,out32);
    (*chip_pin="Y23,Y24,AA22,AA23"*)input[3:0]in1;
    (*chip_pin="AB23,AB24,AC24,AB25"*)input[3:0]in2;
    (*chip_pin="AA14,AG18,AF17,AH17,AG17,AE17,AD17"*)output [6:0]out1;//十位
    (*chip_pin="AC17,AA15,AB15,AB17,AA16,AB16,AA17"*)output [6:0]out12;//个位
    (*chip_pin="AH18,AF18,AG19,AH19,AB18,AC18,AD18"*)output [6:0]out2;//十位
    (*chip_pin="AE18,AF19,AE19,AH21,AG21,AA19,AB19"*)output [6:0]out22;//个位
    (*chip_pin="Y19,AF23,AD24,AA21,AB20,U21,V21"*)output [6:0]out3;
    (*chip_pin="W28,W27,Y26,W26,Y25,AA26,AA25"*)output [6:0]out32;
    wire [4:0]out;
    show(in1%D10,out12);//个位
    show((in1-in1%D10)/D10,out1)//十位
    show(in2%D10,out22);//个位
    show((in2-in2%D10)/D10,out2)//十位
    show2(out%10,out32)//个位
    show2((out-out%10)/10,out3)//十位
    assign out=in1+in2;
endmodule
```

展示模块：

```
module show2(in1,out);
    input [4:0] in1;
    output reg[6:0] out;
    always@(in1)
    begin
        case(in1)
            'b00000:out=7'b1000000;
            'b00001:out=7'b1111001;
            'b00010:out=7'b0100100;
            'b00011:out=7'b0110000;
            'b00100:out=7'b0011001;
            'b00101:out=7'b0010010;
            'b00110:out=7'b0000010;
            'b00111:out=7'b1111000;
            'b01000:out=7'b0000000;
            'b01001:out=7'b0010000;
        endcase;
    end
endmodule
```

任务三：

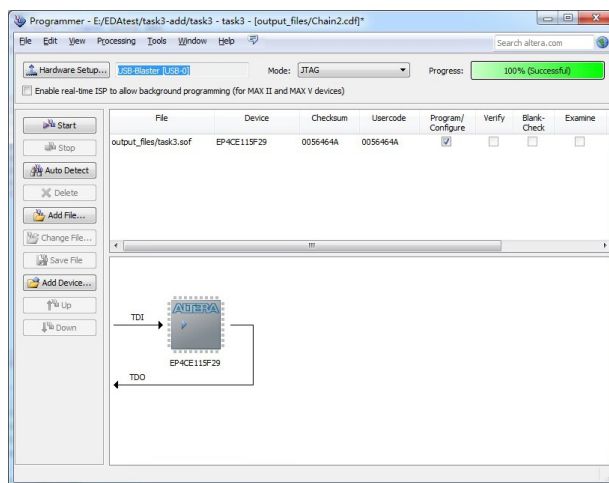
实验原理：添加了新的功能：不仅输出输入的数还输出最大的数在 7 段数码管

上。类似实验二，仅把加的功能改为判别大小的功能就可以，然后再绑定三个 LED 的引脚，分别代表三种状态:大于小于等于 LED 是共阴极，所以是高电平亮。

Verilog 程序:

```
show(in1%D10,out12);//个位
show((in1-in1%D10)/'D10,out1);//十位
show(in2%D10,out22);//个位
show((in2-in2%D10)/'D10,out2);//十位
show2(out%10,out32);//个位
show2((out-out%10)/10,out3);//十位
always@(in1,in2)
begin
if(in1>in2)
begin
state='b001;
out =in1;
end
else if(in1==in2)
begin state='b010;out=in2;end
else begin state='b100;out=in2;end
end
endmodule
```

编译成功后下载到板上:



附加任务: 3-8 译码器设计/仿真/与实现

实验原理: 类似实验一，不同的是译码后的位数和结果不同，但是相同的是只要写出真值表后之后的步骤都是相同的。

Verilog 程序:

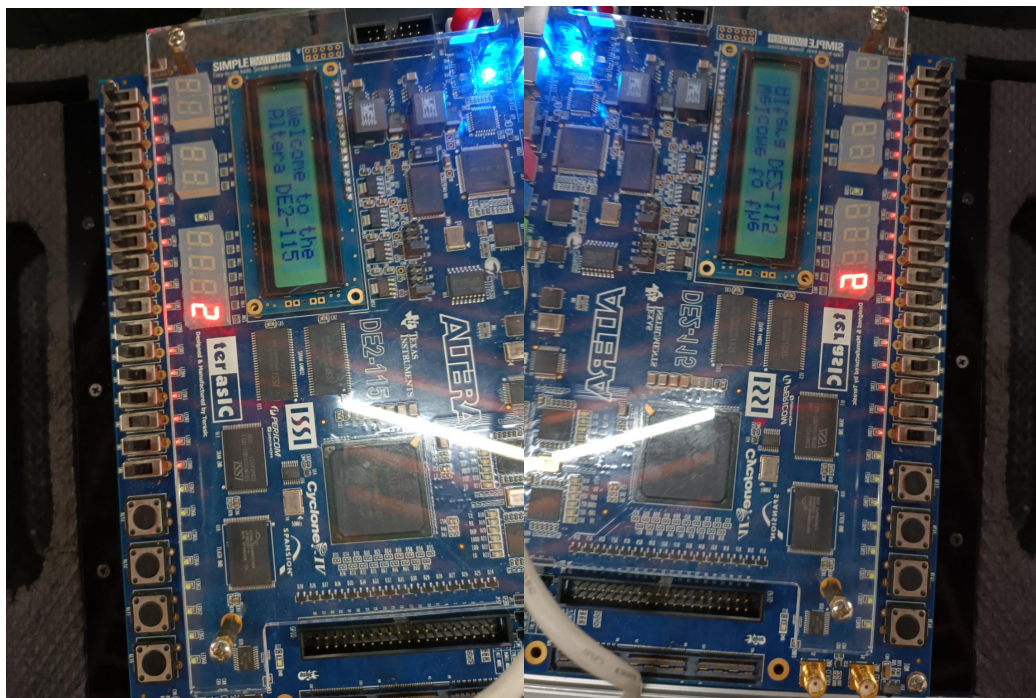
```
task3.v
Completion Report - task3

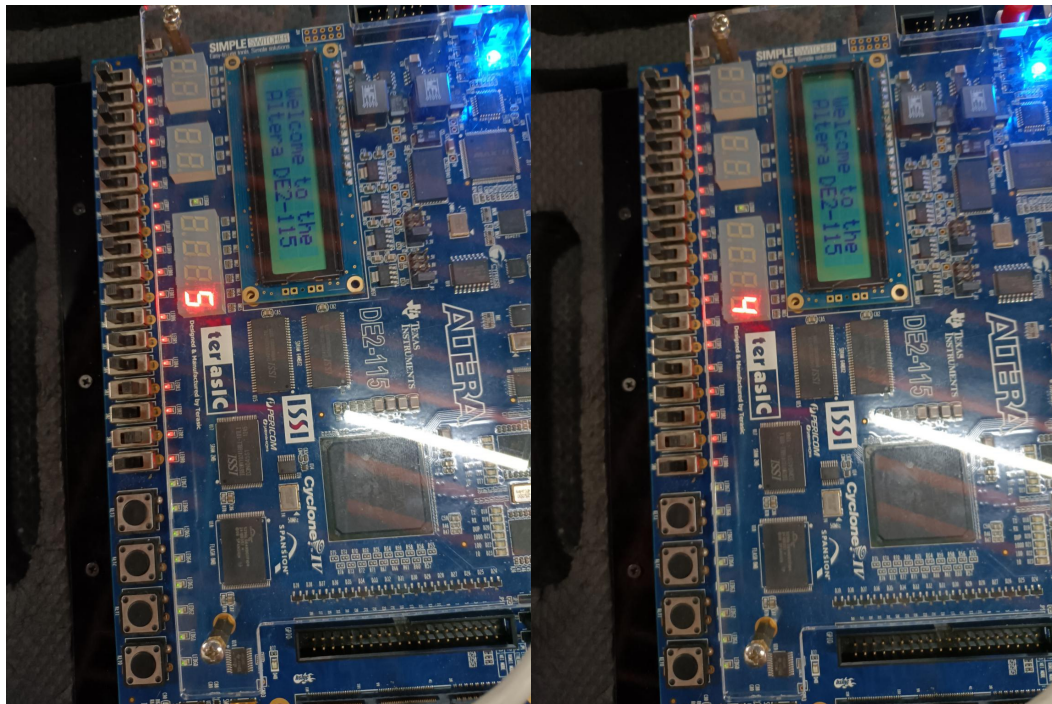
1 module task3(in, en, out);
2   (*chip_pin="Y3, Y24, AA22"*) input [2:0] in;
3   (*chip_pin="AA23"*) input en;
4   (*chip_pin="G19, F19, E19, F21, F18, E18, J19, H19"*) output reg [7:0] out;
5   always@(in, en)
6   begin
7     case((en, in))
8       'b1000: out = 'b00000001;
9       'b1001: out = 'b00000010;
10      'b1010: out = 'b00000100;
11      'b1011: out = 'b00001000;
12      'b1100: out = 'b00010000;
13      'b1101: out = 'b00100000;
14      'b1110: out = 'b01000000;
15      'b1111: out = 'b10000000;
16      default: out = 'b00000000;
17    endcase;
18  end
19 endmodule

1200mv_85c_vhd_slow.sdo in folder "E:/EDAtest/task3-add/simulation/modelsim/" for EDA simulation tool
1200mv_0c_vhd_slow.sdo in folder "E:/EDAtest/task3-add/simulation/modelsim/" for EDA simulation tool
in_1200mv_0c_vhd_fast.sdo in folder "E:/EDAtest/task3-add/simulation/modelsim/" for EDA simulation tool
ns.sdo in folder "E:/EDAtest/task3-add/simulation/modelsim/" for EDA simulation tool
testlist Writer was successful. 0 errors, 0 warnings
simulation was successful. 0 errors, 7 warnings
```

四、实验结果与分析

任务一：板子实现的检验：（因数字有九个数量大空间有限仅展示部分）





任务二：实验波形仿真：

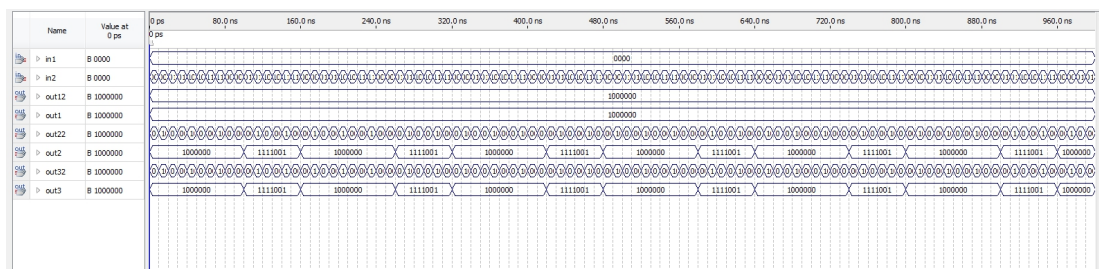


图 1 输入仅为一个数时

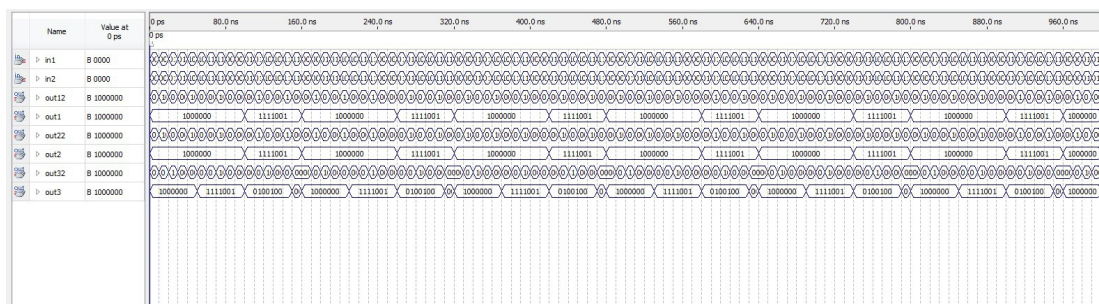
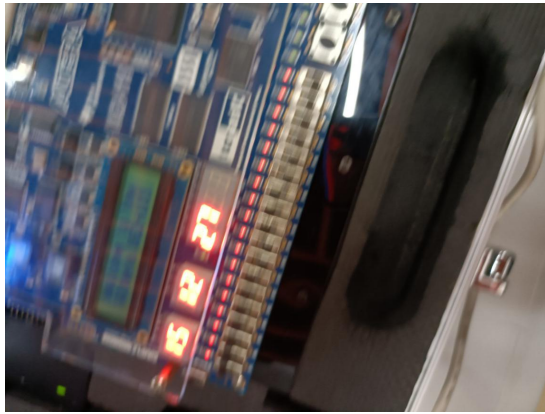
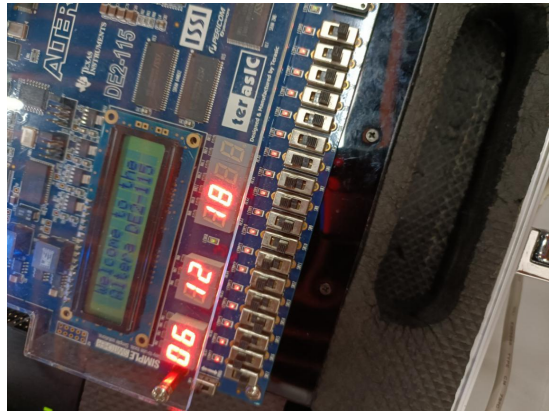
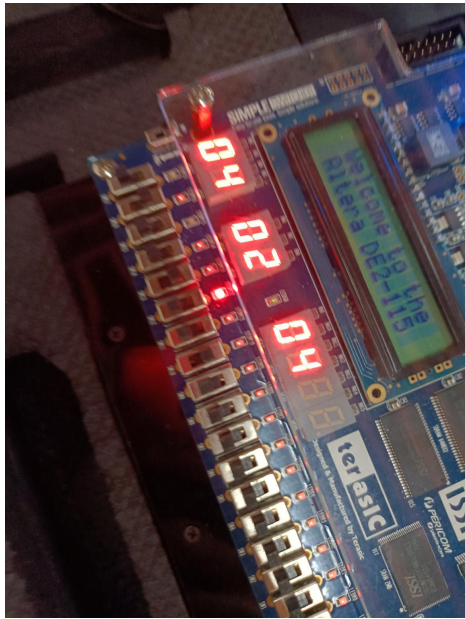
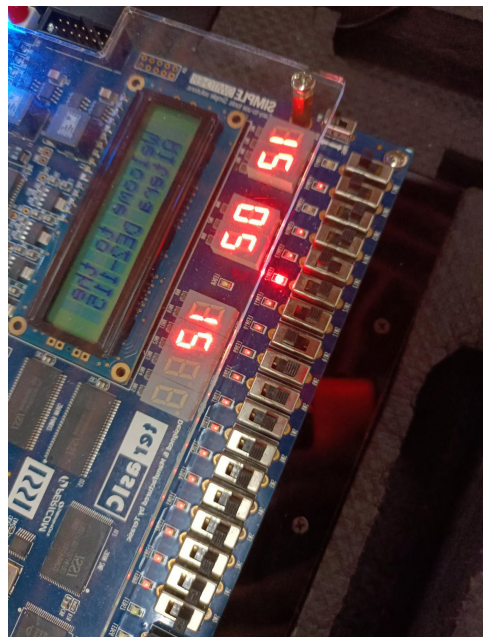


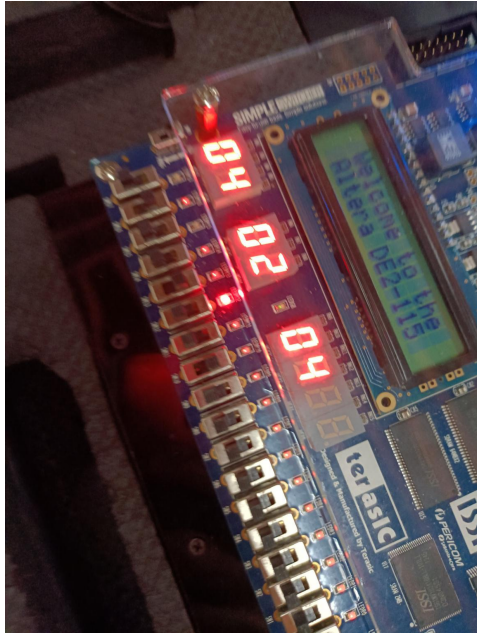
图 2 输入为两个数时

通过对比 in 和 out 的波形图可以看出输出是没有问题的意味着 Verilog 程序设计无误
开发板的实验结果展示



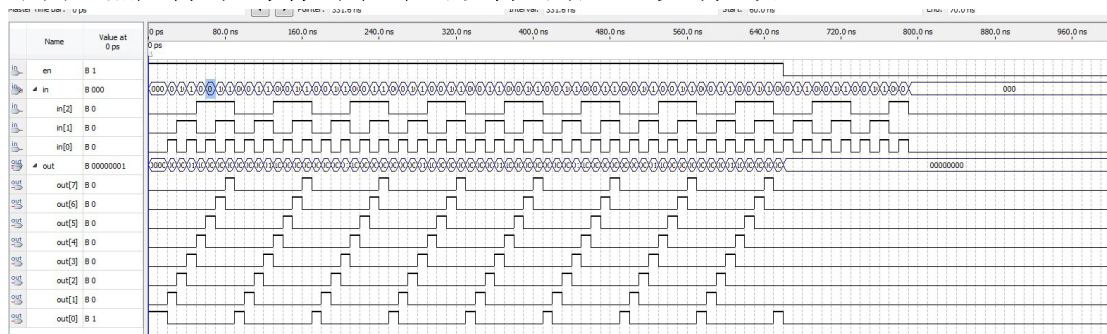
实验三：实验结果：



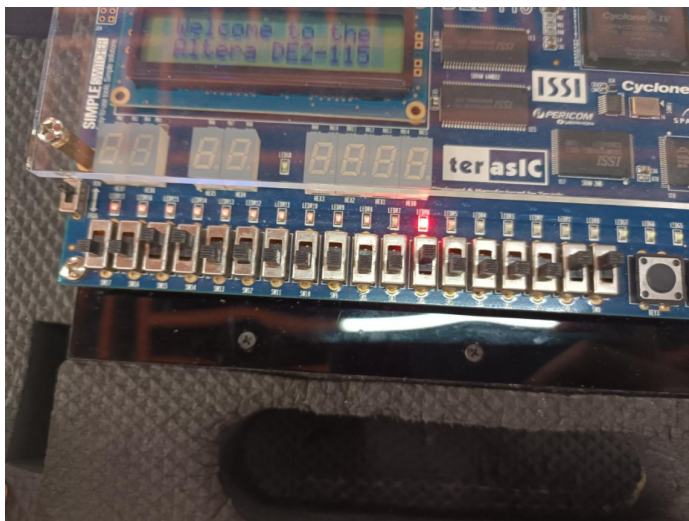


附加任务：波形仿真：

通过对比 in 三位和 out 八位可以看出来时不输入的数不同时对应亮的 led 灯也不同，然后再对比真值表后就可以得出结论，设计无误。



实验板结果：



五、实验体会与讨论

本次实验卡住我的地方主要还是 Verilog 程序的语法，语法还有些不过关，如对变量的声明、使用，对三大结构的运用不够熟练，对 Verilog 所独有的符号使用不确定如 {} 等等，还需要巩固下之前学习的 Verilog 语法，其次要复习下组合逻辑电路这些经典的电路如全加器的原理，译码器的原理，以及实现的功能，这样写 Verilog 程序才能游刃有余。

任务二花了较长时间原因是：使用变量时忘记提前声明位数，导致默认一位，然后出错，找原因找了很长时间，没想到是这么个错误。然后对变量赋值时声明的位数与赋值时的位数不同出现了错误，直接把常量里的位数声明删除后，程序不再出错。