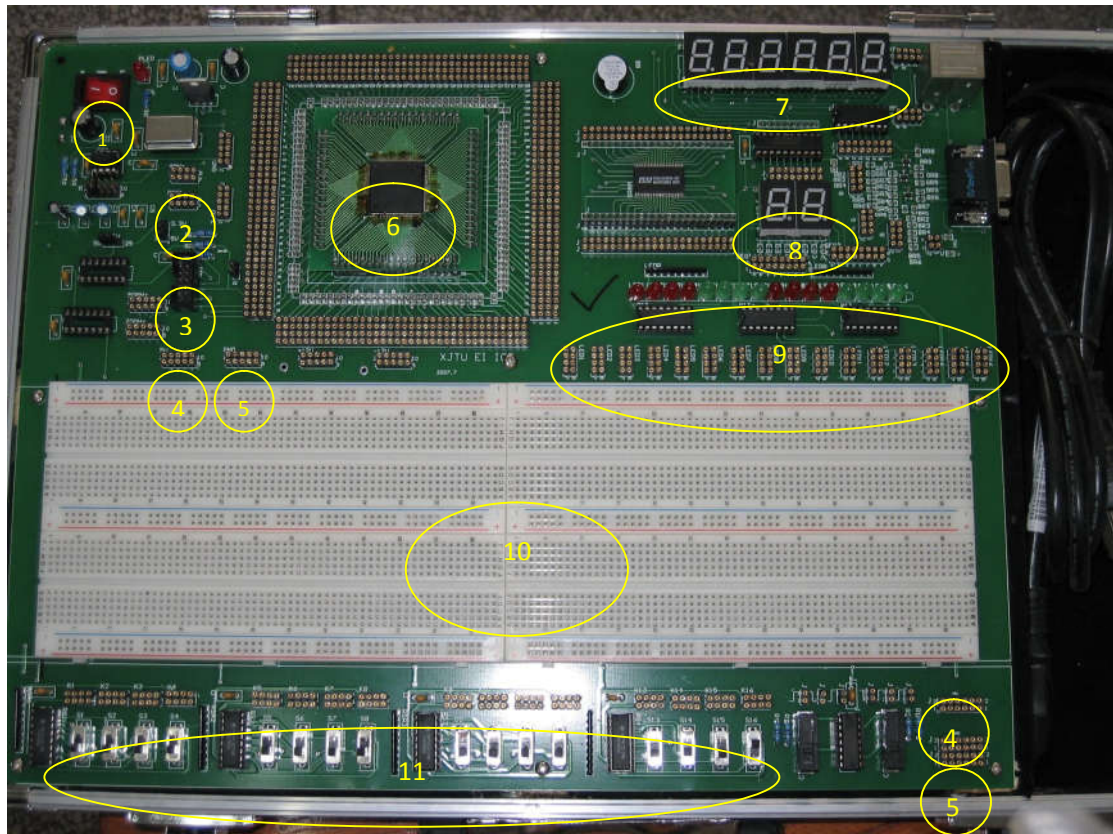
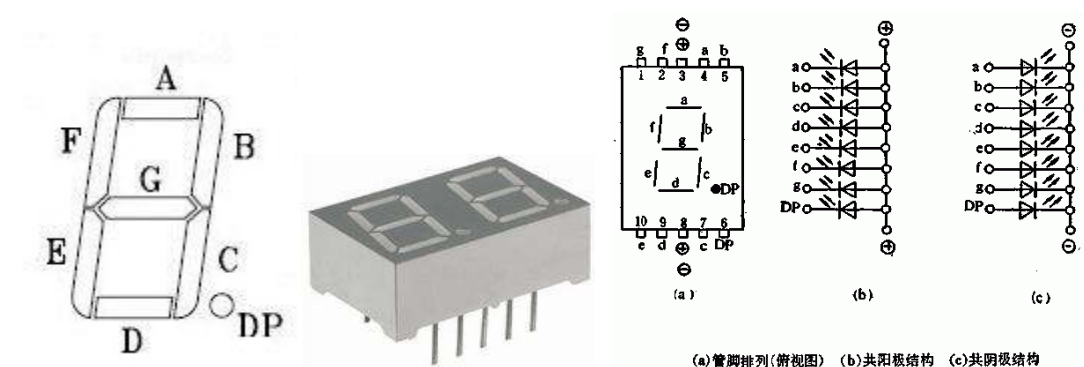


**lab4——实验内容：使用 QuartusII 软件在数电实验箱上设计和实现 7 段译码管程序并测试**  
 一：数电实验箱介绍



- 1.电源开关
- 2.时钟信号源
- 3.JTAG 编程接口
4. 5V 电源插孔
5. GND 信号插孔
- 6.主控 CPLD 芯片，型号为：
7. 6 位共阳极 7 段数码管
8. 两位共阴极 7 段数码管
- 9.LED 灯
- 10.面包板
- 11.拨位开关

二：7 段数码管原理

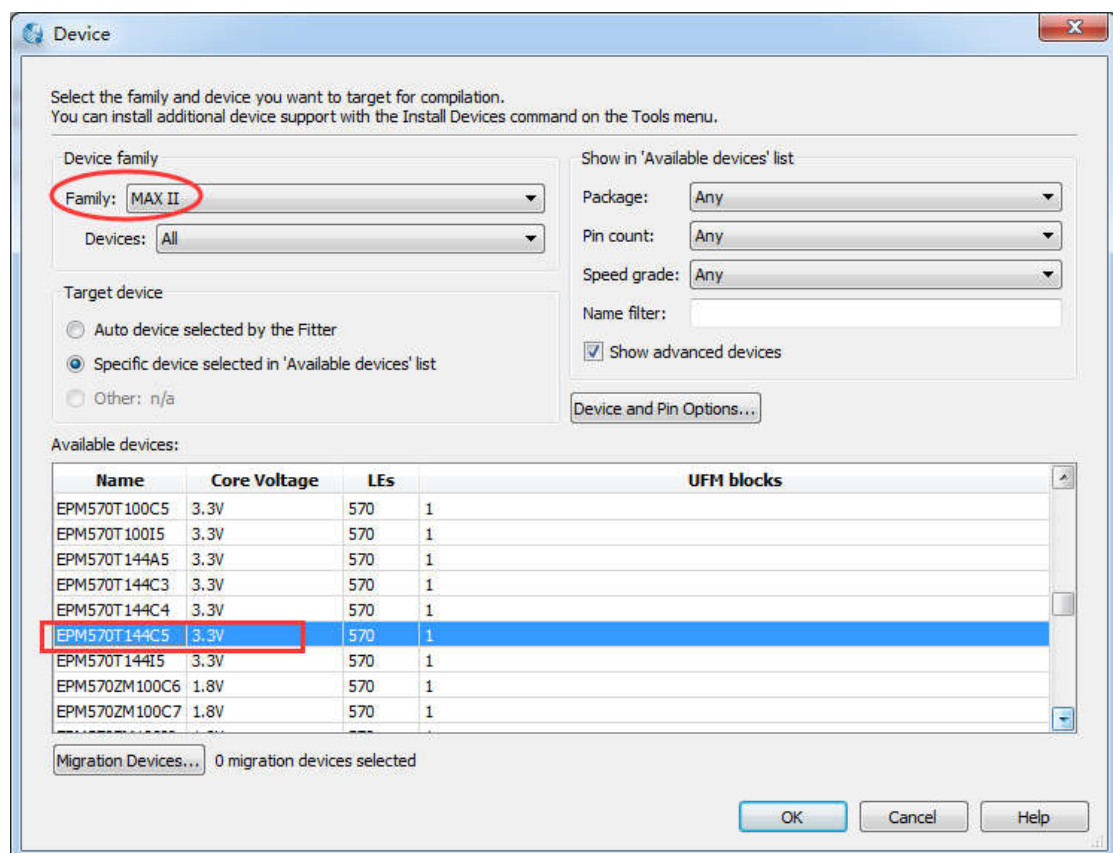


7 段数码管有 a b c d e f g dp 八段，靠每一段的亮灭状态的不同来显示不同的字符，可以显示 16 进制 0-F。分为共阴极和共阳极两种，共阳极数码管是指数码管的八段发光二极管的阳极(正极)都连在一起，而阴极对应的各段可分别控制，如图(b)所示，此时控制各段的信号为低时该段点亮；共阴极数码管是指数码管的八段发光二极管的阴极(负极)都连在一起，而阳极对应的各段可分别控制，如图(c)所示，此时控制各段的信号为高时该段点亮。例：要让共阴极数码管显示 0，需使 abcdefg=7'b1111110，而要让共阳极数码管显示 0，则需使 abcdefg=7'b0000001。

### 三：实验步骤

#### 3.1: 创建工程

按照实验二的内容在 E 盘自己学号的文件夹下为今天的实验创建一个工程目录。使用 QuartusII 软件在该工程目录下创建一个新工程，注意新工程选择芯片型号为 MAX II 系列的 EPM570T144C5。



#### 3.2: 编写 7 段译码管的程序

为新建的工程添加一个 verilog 文件，并使用 case 语句完成 7 段译码管的程序编写。对工程进行编译和仿真，记录仿真结果。

```
module seven_segment(
    input [3:0] data,
    output reg a,b,c,d,e,f,g
);
```

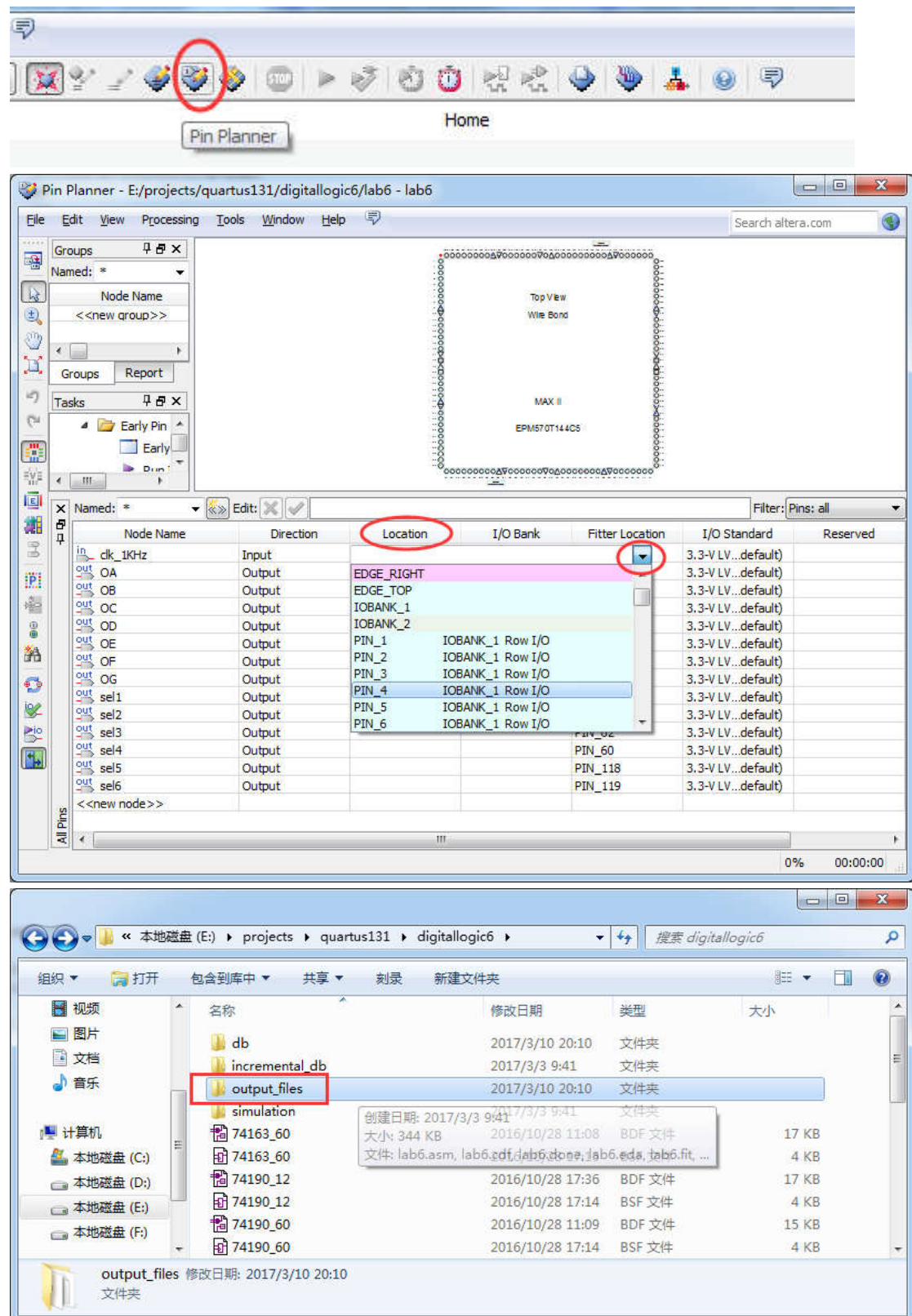
对各个输入输出信号进行类型声明

请用 case 语句实现内部逻辑

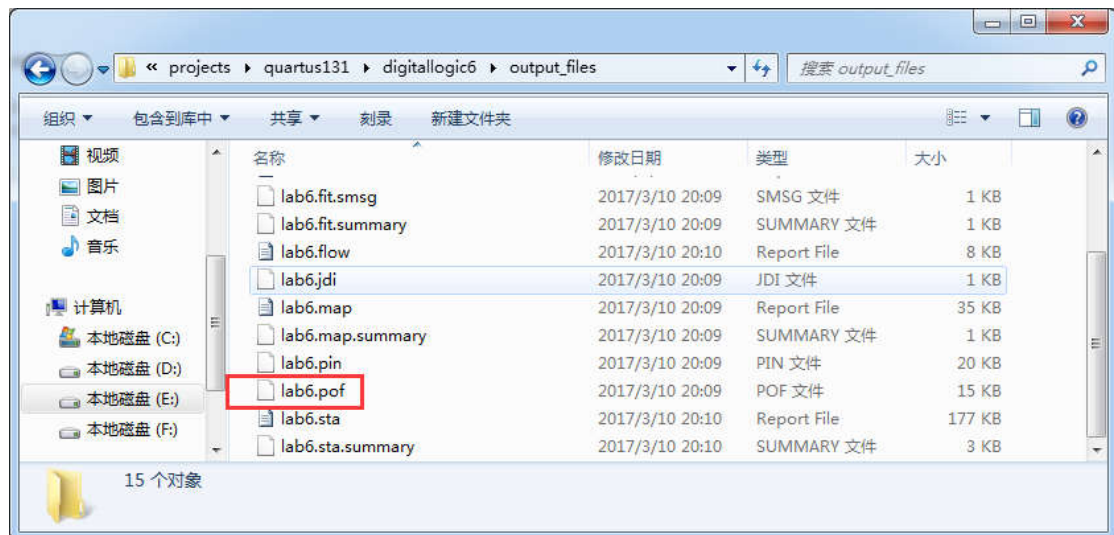
```
endmodule
```

### 3.3: 分配管脚后重新编译工程

工程编译成功后，点击 Pin Planner，打开管脚分配界面。在 location 位置对设计中的端口信号进行管脚分配。待所有信号管脚分配完成后，再次对工程编译。



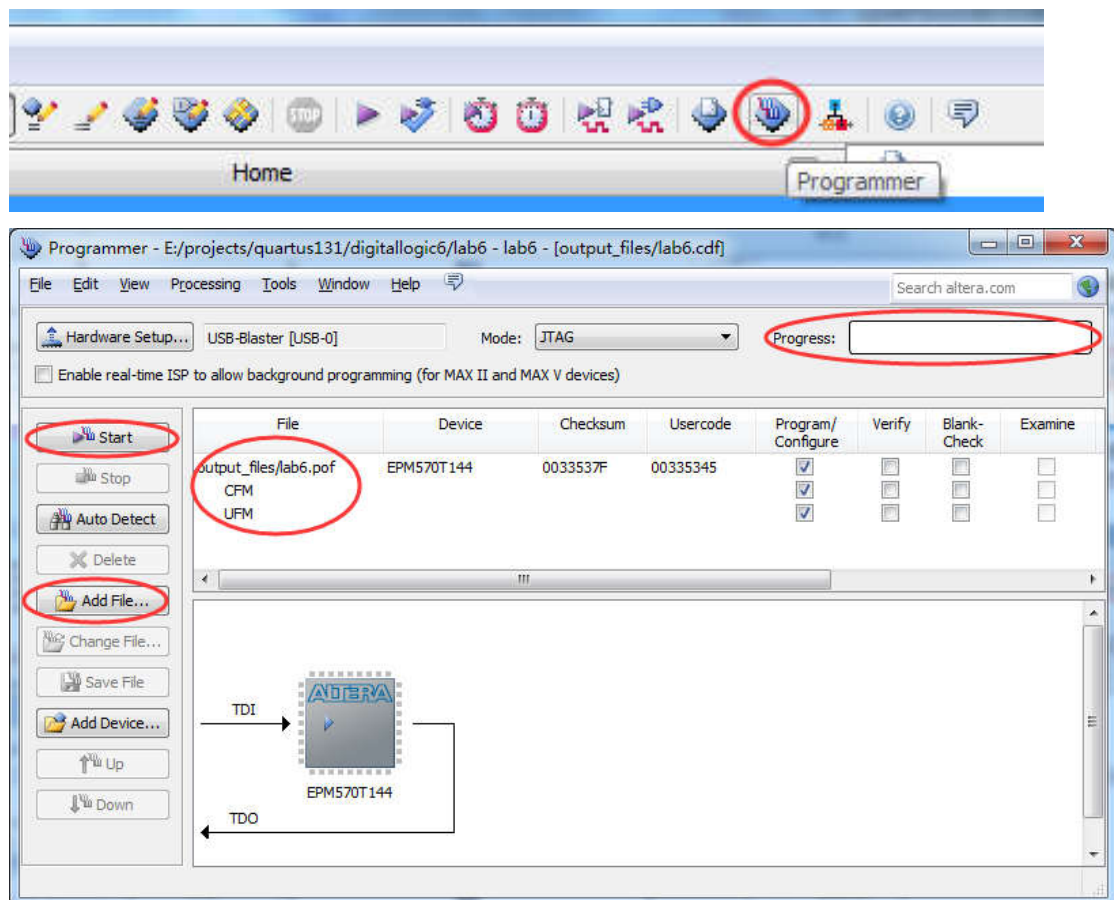




如果整个工程编译成功的话，在工程路径下 `output_files` 文件夹下会出现一个 `pof` 后缀的下载文件。

### 3.4：下载程序

打开实验箱的电源，连接 JTAG 线到实验箱的 JTAG 编程接口。点击 **Programmer**，调出下载对话框。这时 **File** 下会自动出现该工程的 `pof` 下载文件，若没有出现，可以通过 **Add File** 浏览到之前生成的 `pof` 文件并打开。最后点击 **Start** 开始下载，当 **progress** 进度条走完时表示下载完毕。



### 3.5 接线验证

用导线连接输出信号引脚到 7 段译码管，连接拨位开关到输入信号引脚上，通过拨动四个拨位开关来验证设计的正确与否。

四：实验报告应至少包含：

- 1.描述 7 段数码管的显示原理，写出真值表，写出代码的核心 case 语句；
- 2.描述从分配管脚到下载验证的关键步骤。

五：预习数电教材上竞争与现象相关知识