

实验一：GPIO 输出实验——三色 LED 交替闪烁

一、实验目的与意义

- 1、了解 FPGA 输出引脚功能
- 2、了解三色 LED 的特征和应用领域
- 3、掌握 QuartusII 集成开发环境使用方法
- 4、掌握 FPGA 引脚配置方法

二、实验设备及平台

- 1、icore4 开发板
- 2、Blaster 仿真器
- 3、MicroUSB 数据线
- 4、QuartusII 开发平台
- 5、装有 WIN XP(及更高版本)系统的计算机

三、实验原理

本实验根据 led 的驱动原理编写 led 控制程序，led 的驱动原理是通过程序控制 FPGA 数字 IO 的电平变化实现 led 的亮灭。其硬件原理图如图 1-1 所示。



图 1-1

程序实现的功能为，icore4 板上与 FPGA 芯片相连的三色 LED 灯快速交替闪烁。本程序实现原理为：通过硬件语言描述一个周期为 0.2s 的时间闸门信号，以该信号作为状态转移的触发信号来实现状态跳转，通过状态跳转实现不同 led 灯的亮灭。时间闸门信号由以下代码产生：

```
/*clk_25m 上升沿或者 rst_n 下降沿到来执行 always 块内容*/
always @ (posedge clk_25m or negedge rst_n)
if(!rst_n) //rst_n=0 时执行 if 中内容
begin
cnt <= 25'd0; //cnt 赋值为 0
led_cnt <= 1'd0; // led_cnt 赋值为 0
end
else if(cnt == 25'd5000000) //当 cnt=5000000 时执行 else if 中内容
begin
cnt <= 25'd0; //cnt 赋值为 0
if(led_cnt == 3'd2) //当 led_cnt=2 时执行 if 中内容
led_cnt <= 3'd0; //led_cnt 赋值为 0
else
led_cnt <= led_cnt + 1'd1; //led_cnt 自加 1
end
end
```

```

else
begin
cnt <= cnt + 1'd1;           //cnt 自加 1
end

```

四、新建工程

1、双击 Quartus 桌面图标打开软件，出现下面界面，如图 1-2 所示。

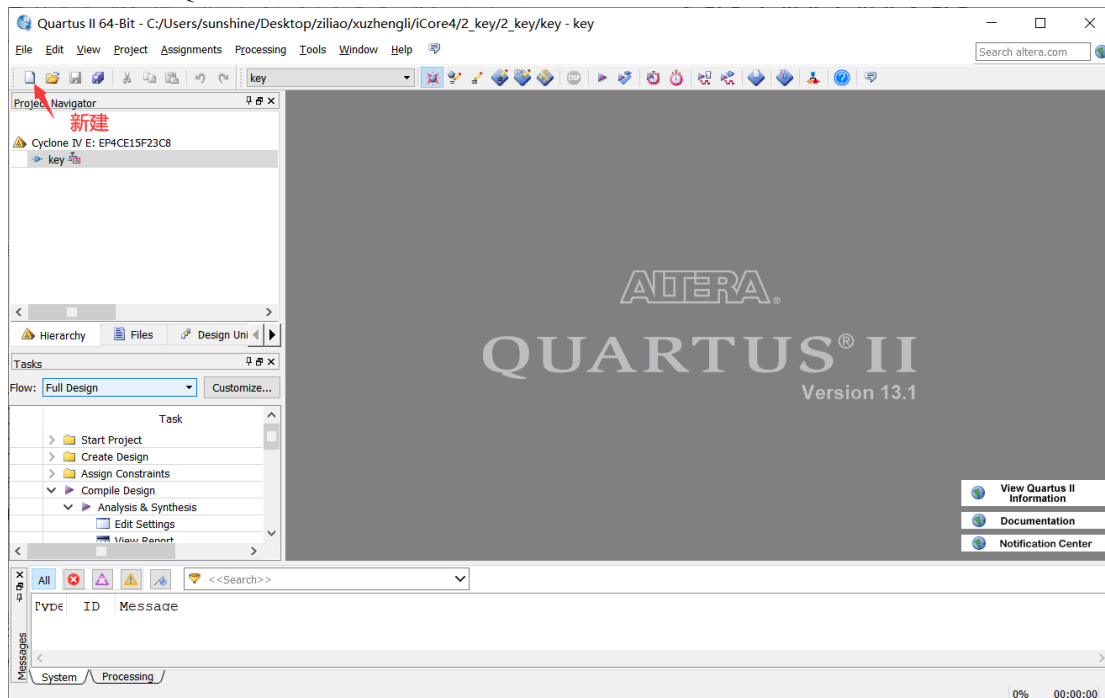


图 1-2

2、点击新建图标出现下面对话框，选择 New Quartus II Project 选项，如图 1-3 所示。

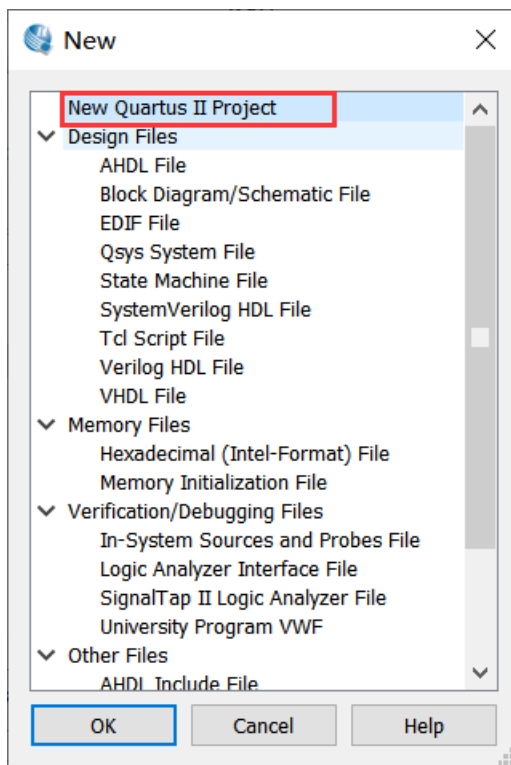


图 1-3

3、直接点击 Next，如图 1-4 所示。

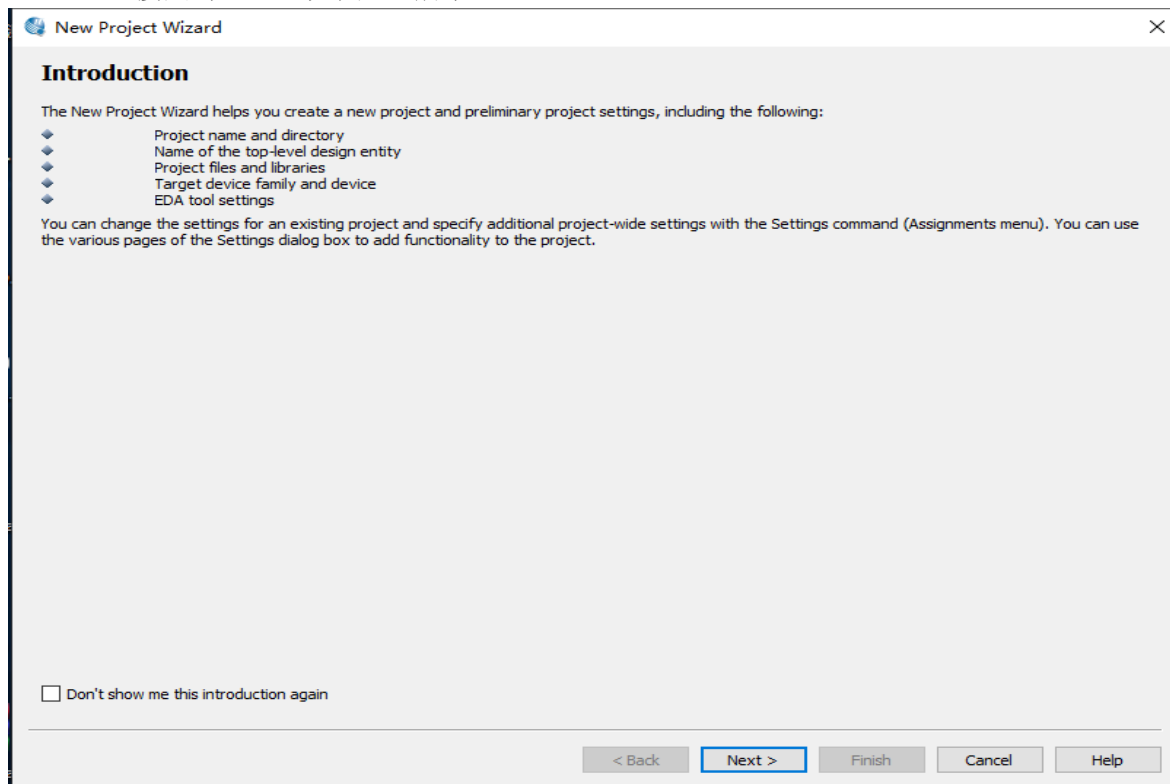


图 1-4

3、在此界面选择工程存放的路径，设置工程名及顶层文件名称，如图 1-5 所示。

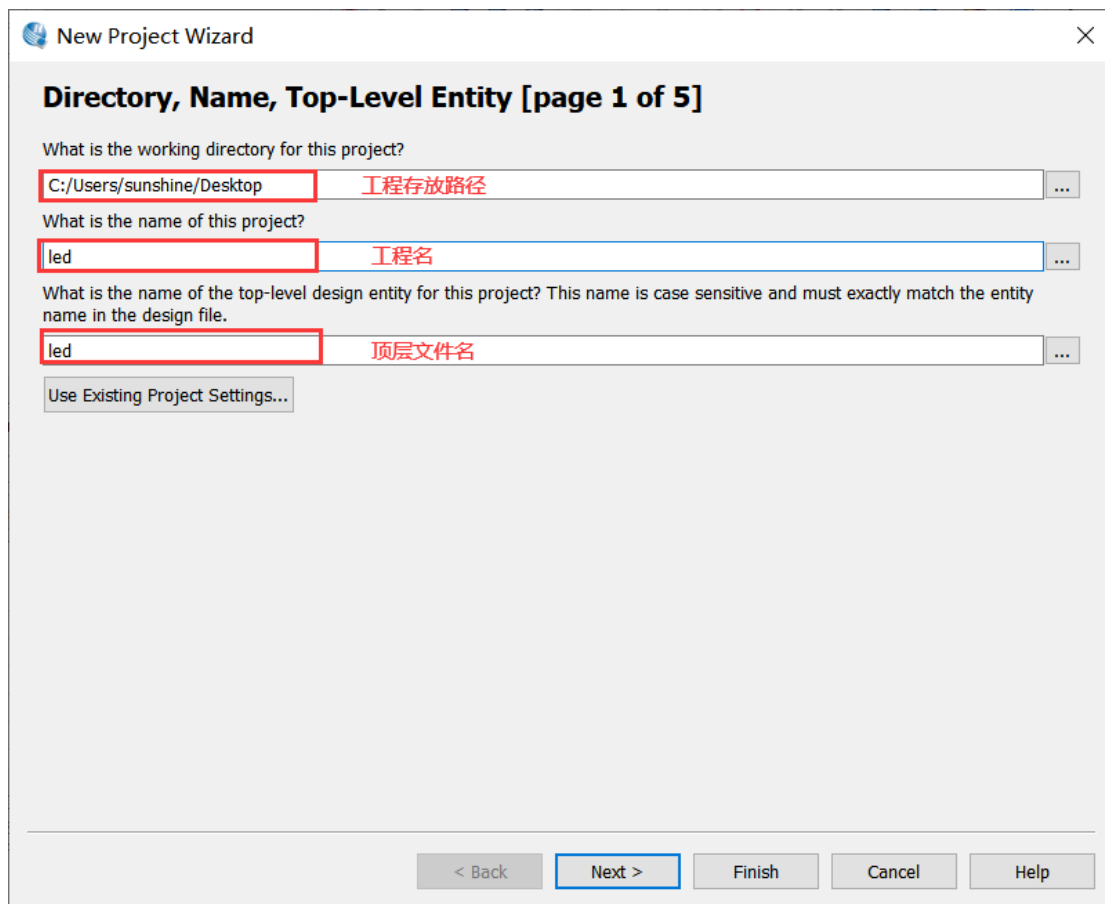


图 1-5

5、此界面主要用来添加已经编辑好的模块文件，若没有直接下一步即可，如图 1-6 所示。

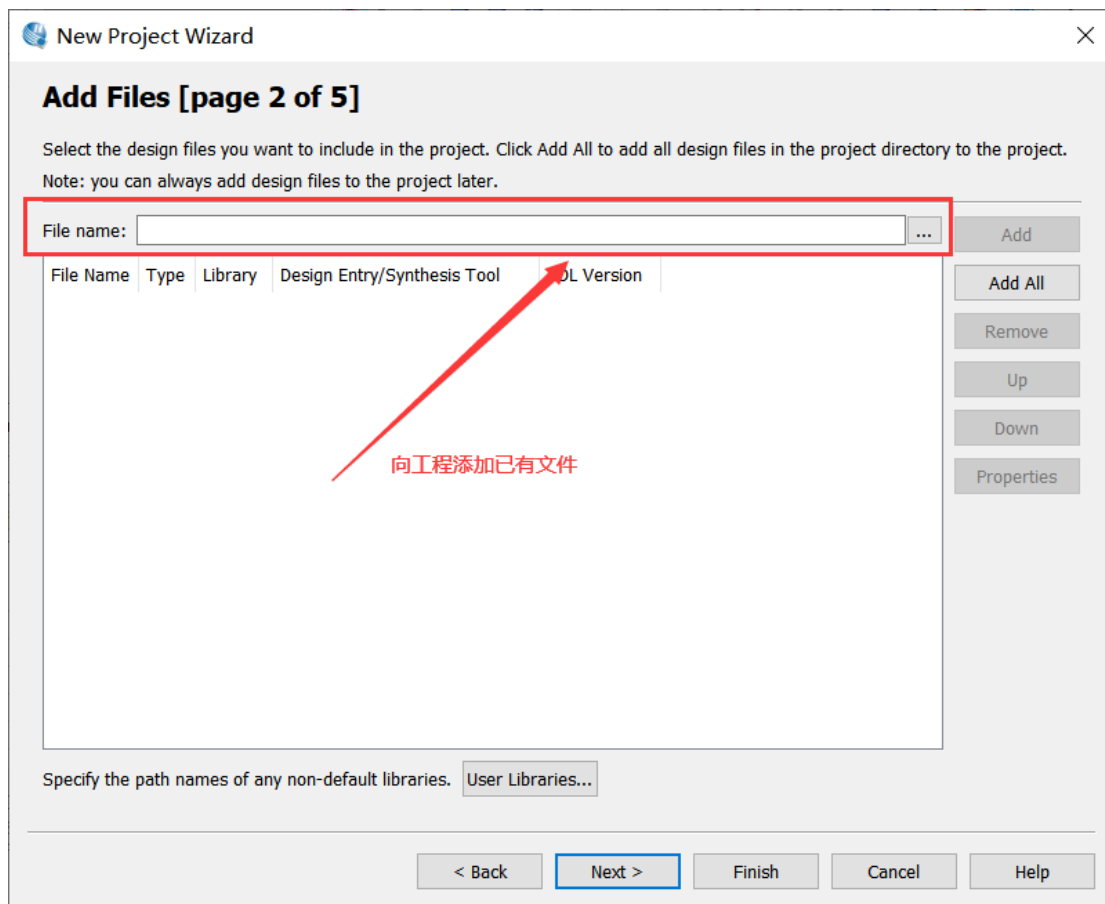


图 1-6

6、选择设备所用的芯片型号，芯片的系列、封装、引脚数、速度等级的选项方便用户更快速精确的找到目标芯片的型号，如图 1-7 所示。

New Project Wizard

Family & Device Settings [page 3 of 5]

Select the family and device you want to target for compilation.
You can install additional device support with the Install Devices command on the Tools menu.

Device family
Family: Cyclone IV E 芯片所属系列
Devices: All

Show in 'Available devices' list
Package: FBGA 芯片封装
Pin count: 484 芯片引脚数
Speed grade: 8 芯片速度等级
Name filter:
☒ Show advanced devices

Target device
☐ Auto device selected by the Fitter
☒ Specific device selected in 'Available devices' list
☐ Other: n/a

Available devices:

Name	Core Voltage	LEs	User I/Os	Memory Bits	Embedded multiplier 9-bit elem
EP4CE15F23C8	1.2V	15408	344	516096	112
EP4CE30F23C8	1.2V	28848	329	608256	132
EP4CE40F23C8	1.2V	39600	329	1161216	232
EP4CE55F23C8	1.2V	55856	325	2396160	308
EP4CE75F23C8	1.2V	75408	293	2810880	400
EP4CE115F23C8	1.2V	114480	281	3981312	532

图 1-7

7、进行仿真器的一些设置，可忽略，如图 1-8 所示。

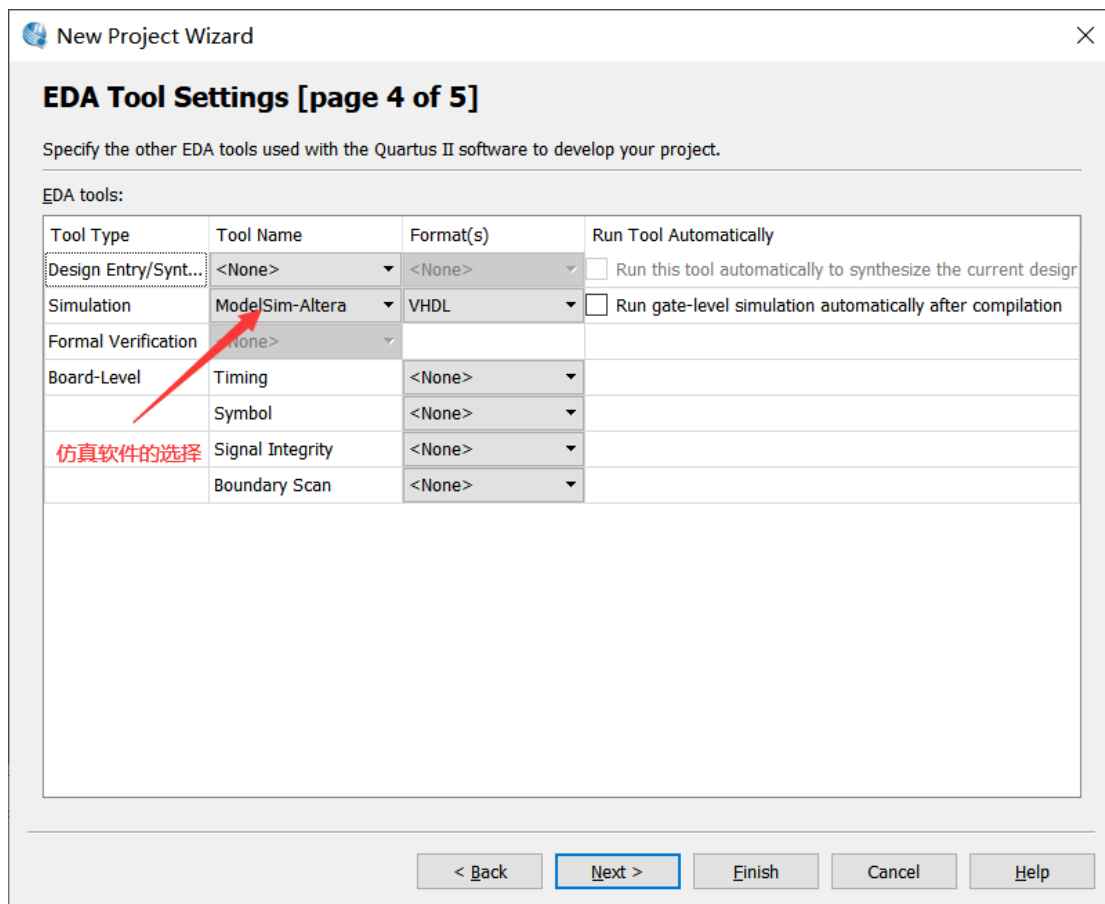


图 1-8

8、至此，工程建立完成，如图 1-9 所示。

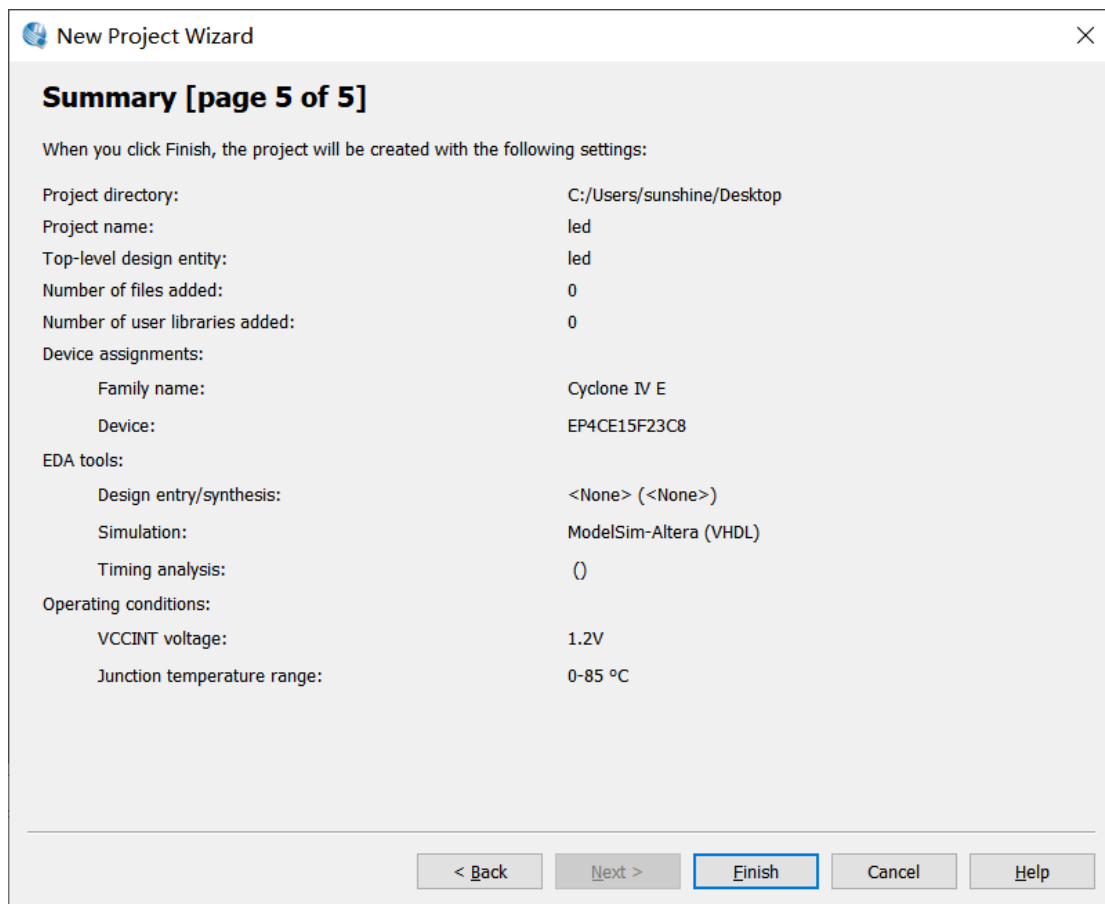


图 1-9

9、回到软件主界面可以看到新建的工程已经显示在左边列表中，如图 1-10 所示。

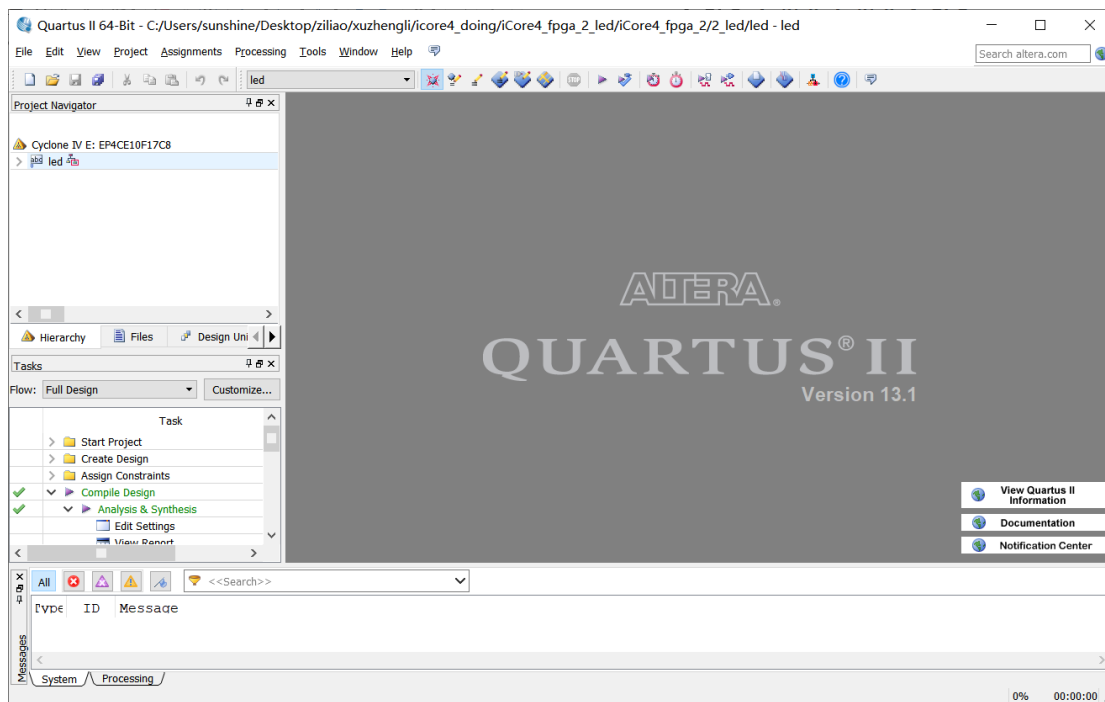


图 1-10

10、重新点击新建图标进入下面界面选择将要编译的文件类型，此处选择 Verilog HDL 硬件语言描述类型，如图 1-11 所示。

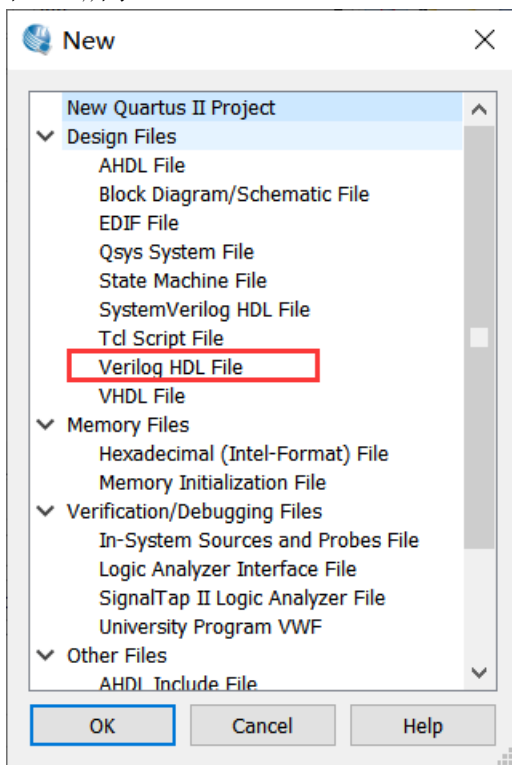


图 1-11

11、一个新的文件就建好了，此时文件是空的，点击 file->save as 保存文件。常将源文件保存在 src 文件夹下，且文件名称必须与模块名称一致，如图 1-12 所示。

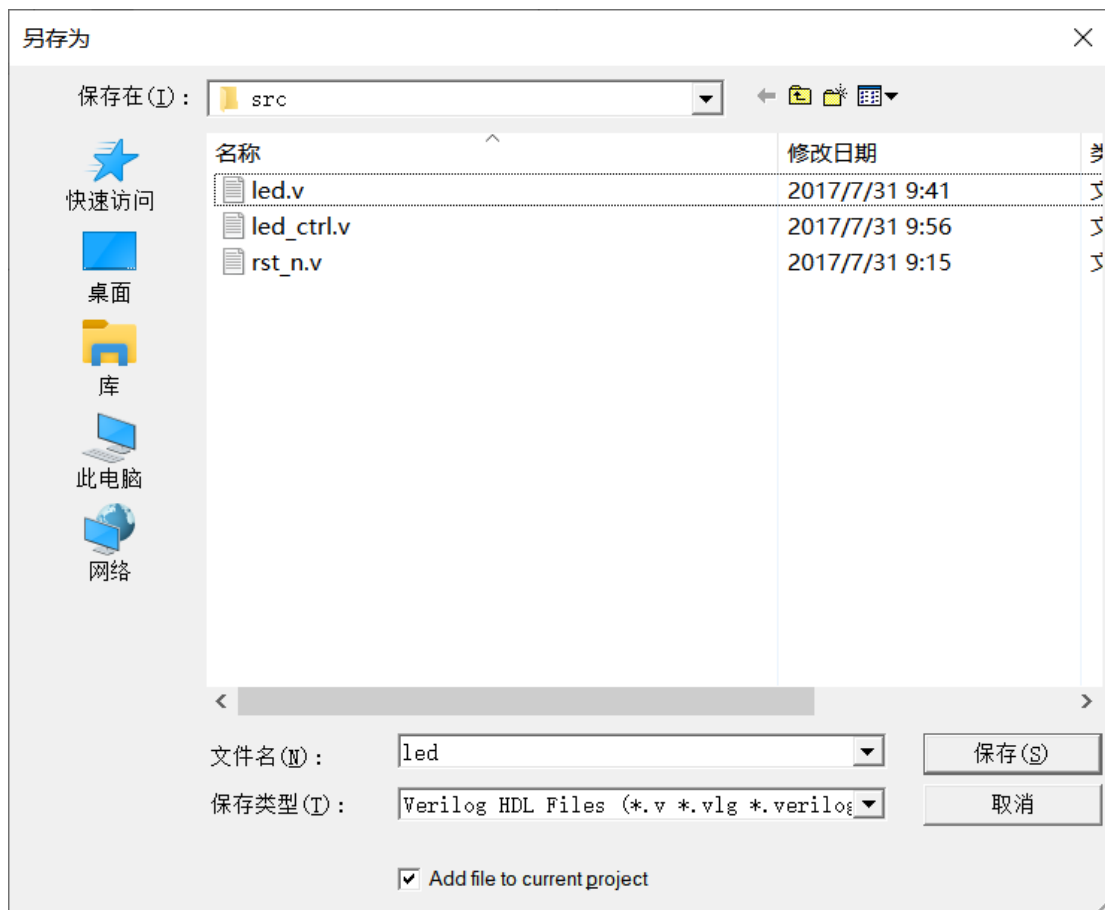


图 1-12

12、此时就可以写代码了，写好以后点击方框处的全部保存，如图 1-13。

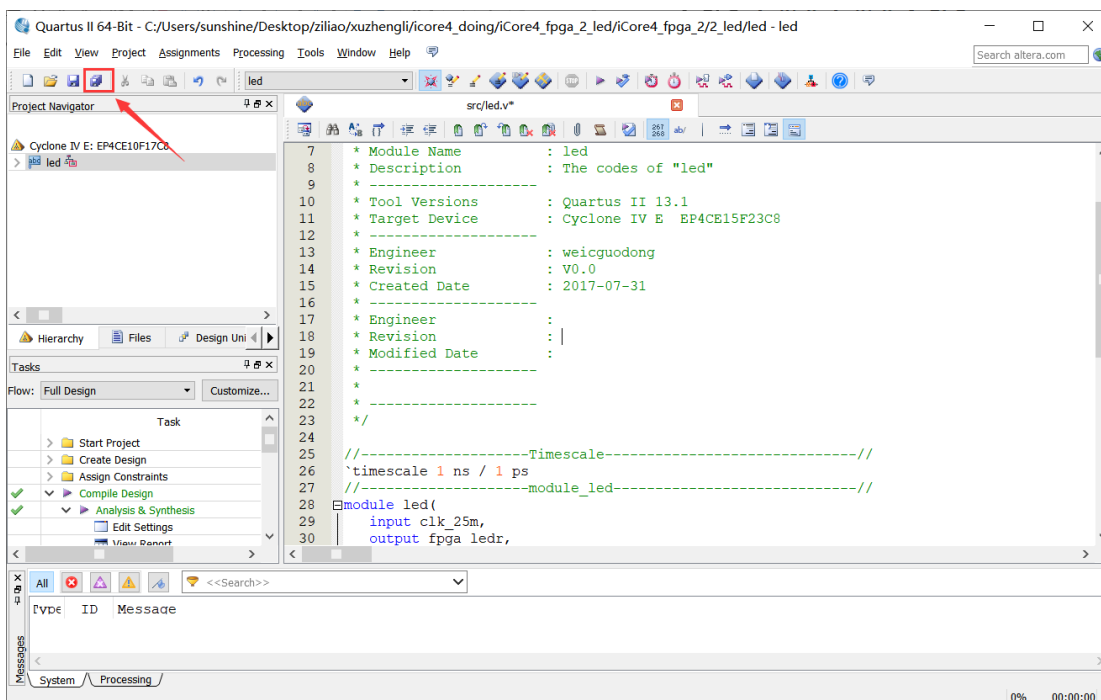


图 1-13

13、点击 assignments->device,出现下边窗口,再点击方框处的“devices and pin options”,在弹出的窗口左栏中选择箭头 1 所示的“unused pins”,在相应的右边栏红框处的下拉菜单中选择 “as input tri-stated”。

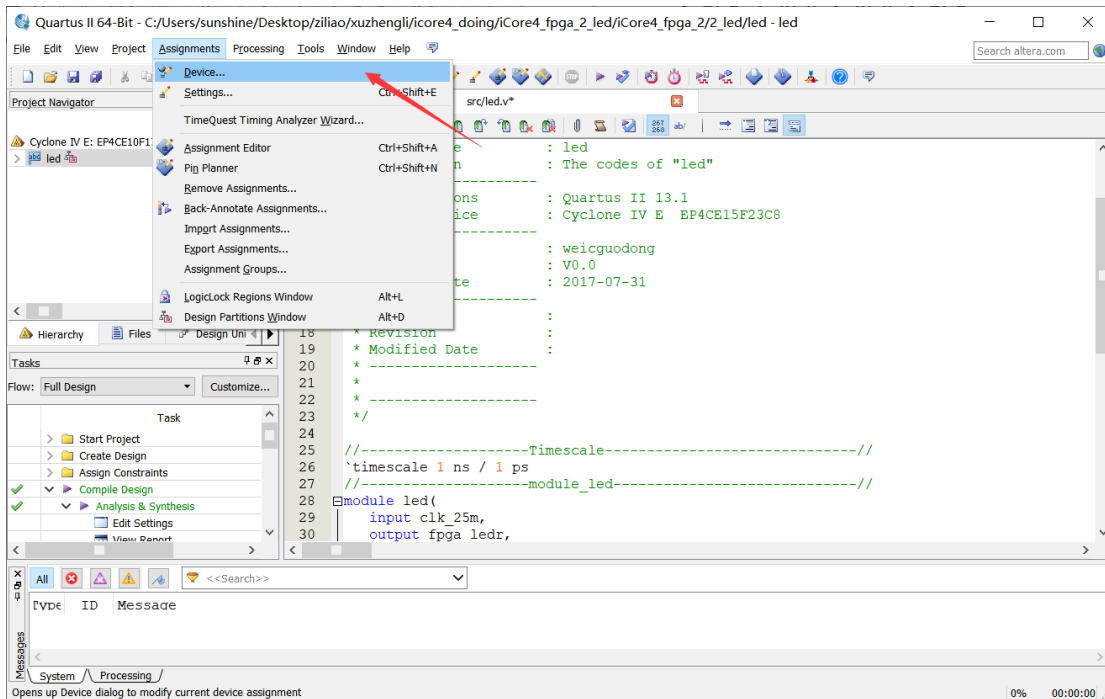


图 1-14

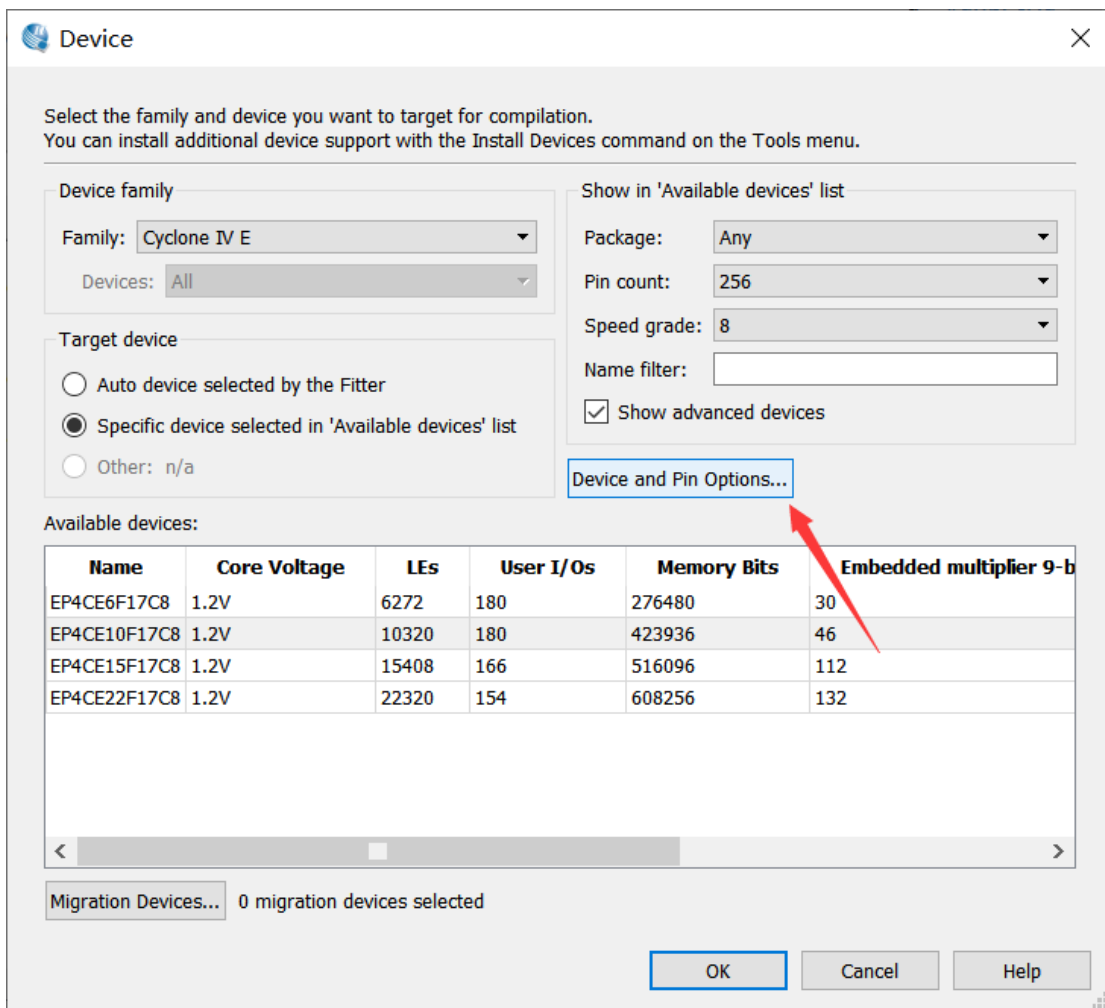


图 1-15

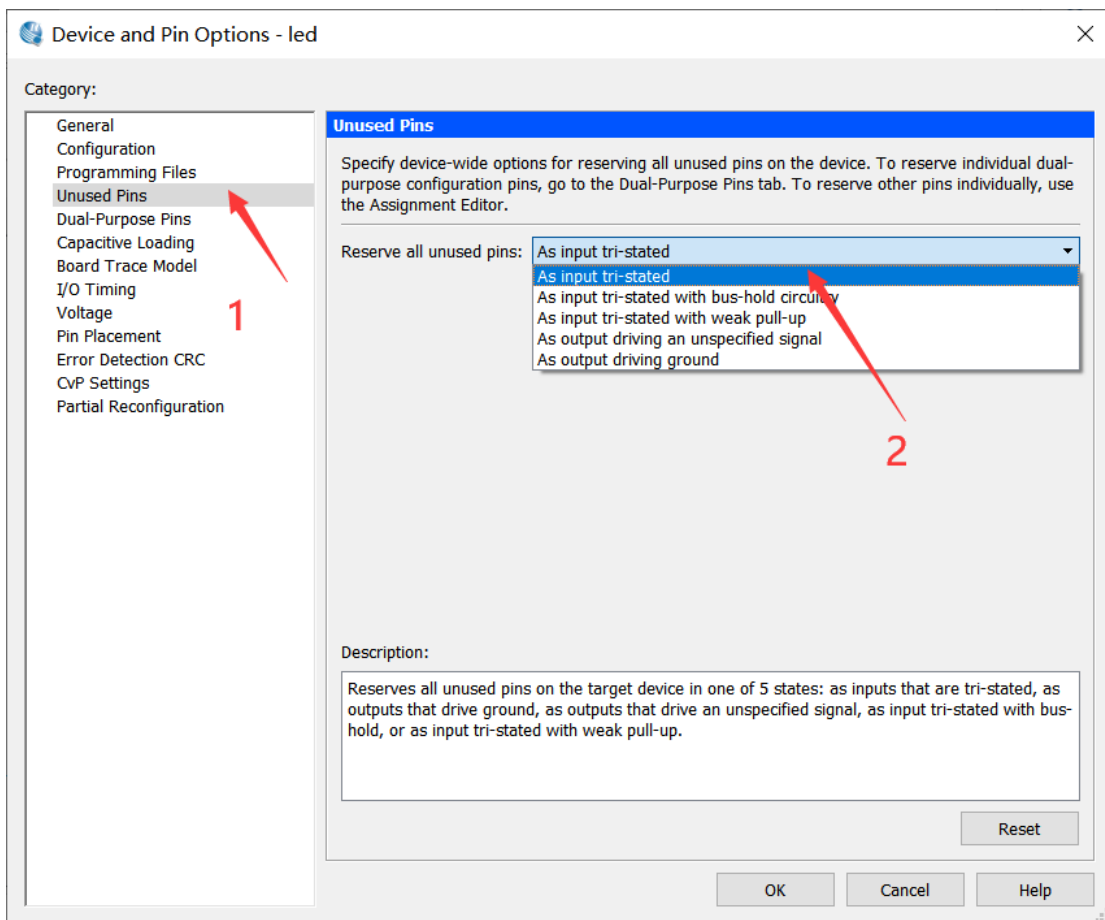


图 1-16

14、然后在左边栏中选择”Dual-purpose pins”,在相应的右边栏中双击图示位置会出现下拉菜单，选择”use as regular I/O”。

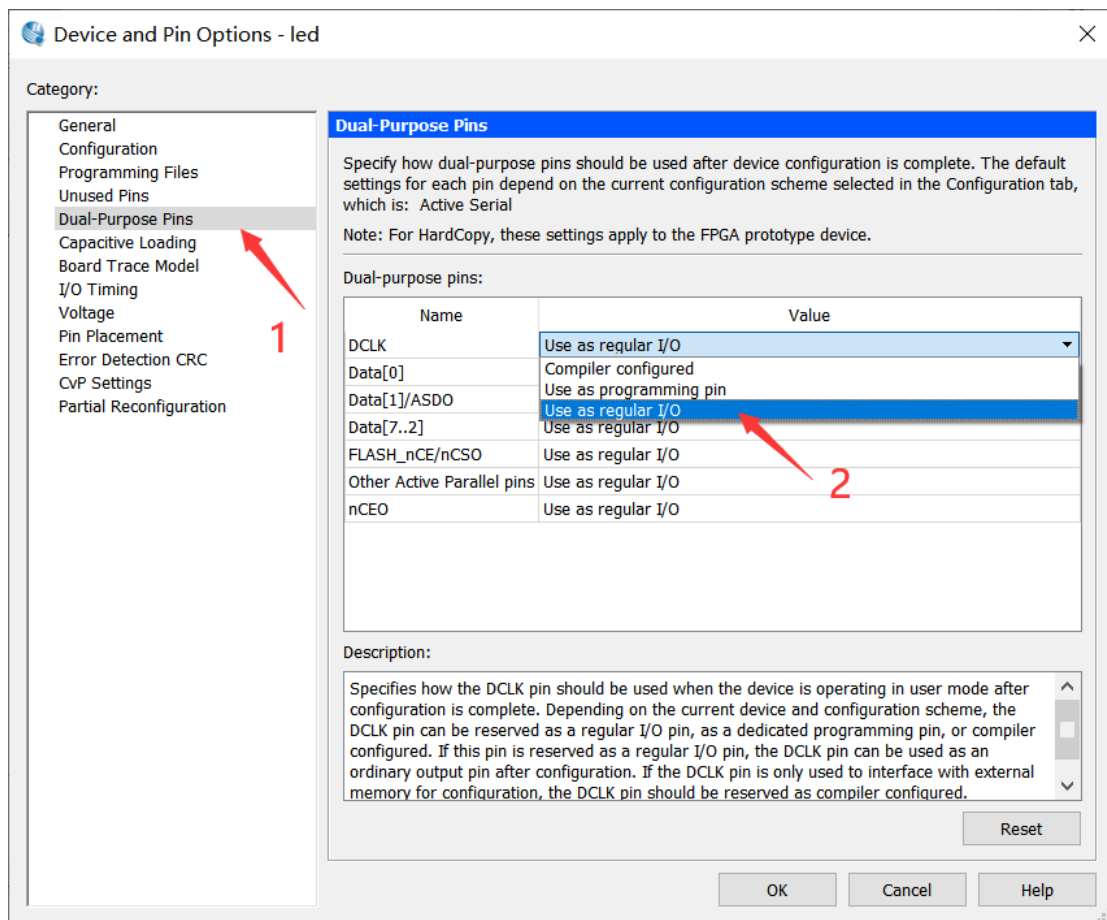


图 1-17

15、这是选择好的界面，点击 OK 即可。

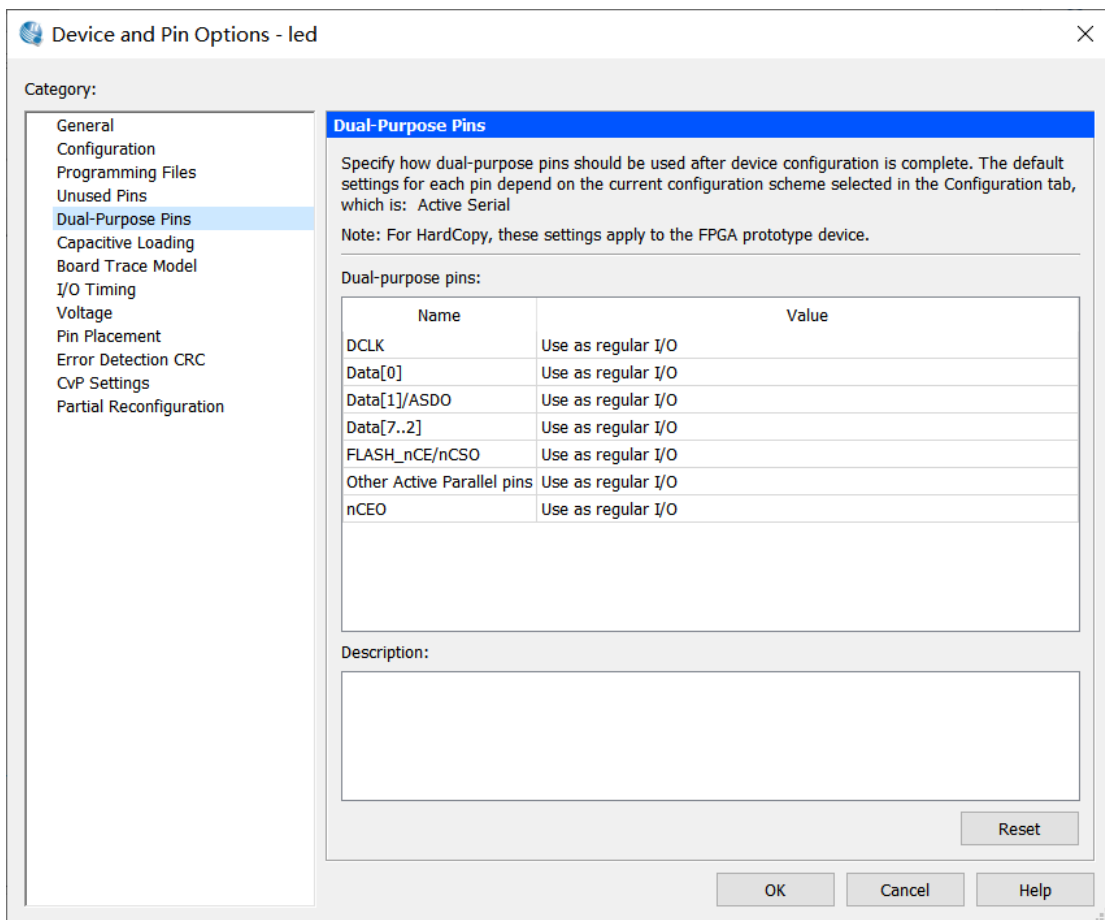


图 1-18

16、点击图示方框中的按钮进行编译。

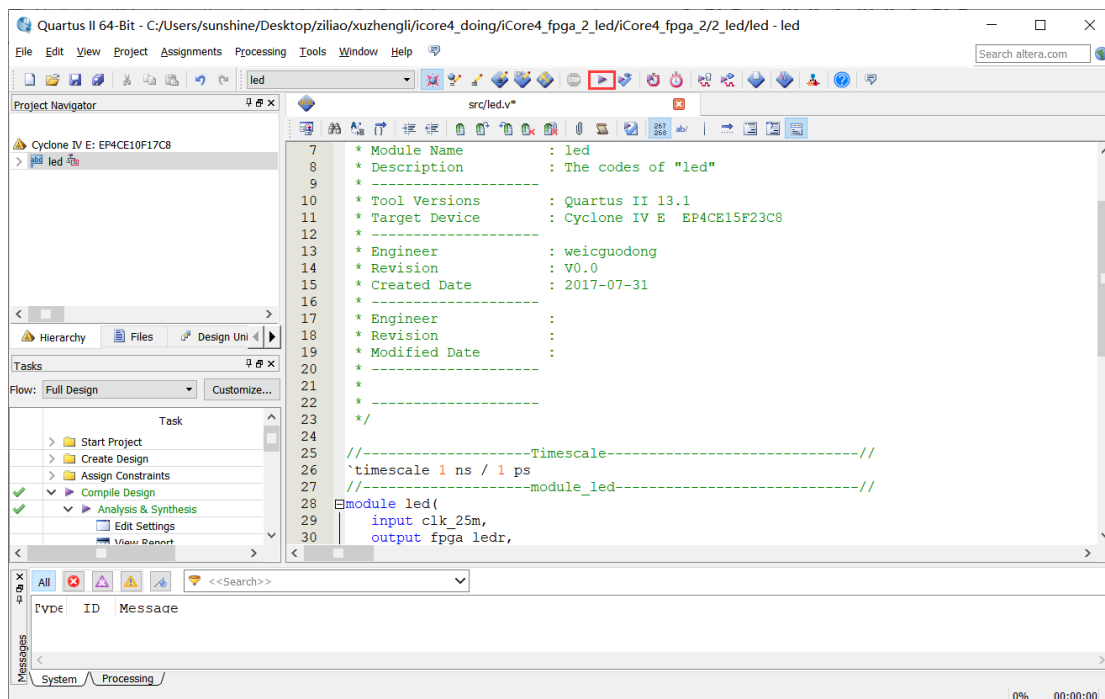


图 1-19

17.编译完成后，点击 Assignment->pin planner 或直接点击图 1-20 方框中的按钮，进行引脚分配。

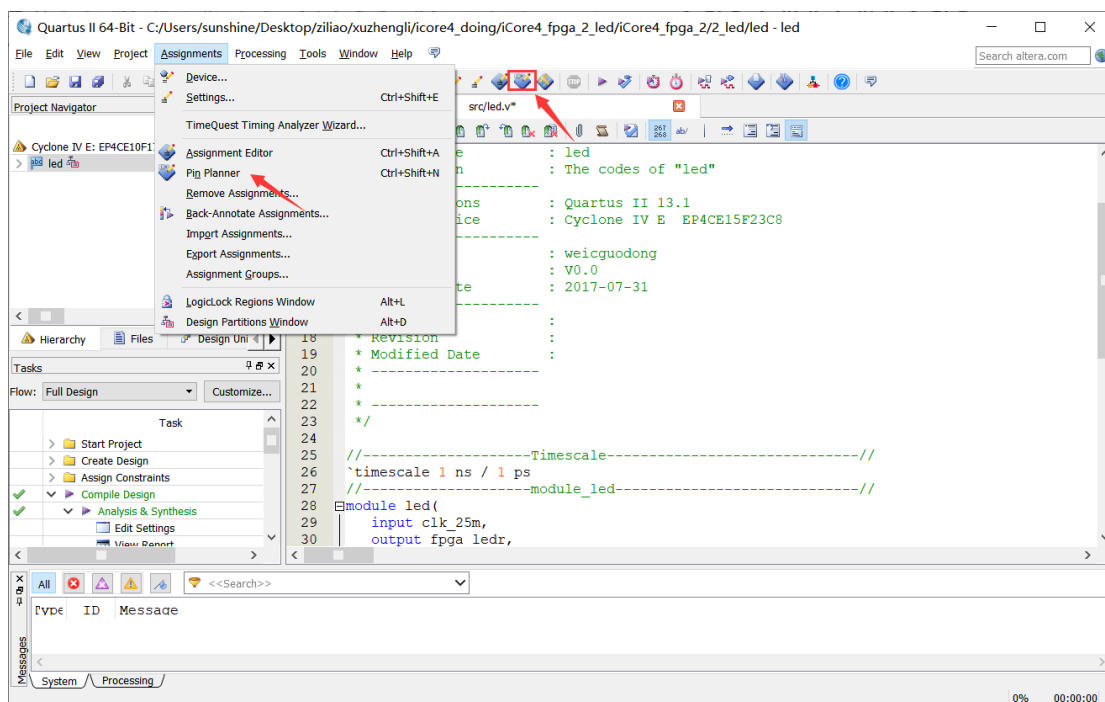


图 1-20

18、在引脚分配窗口下方的列表中显示工程的输入输出引脚，双击 location 栏下（箭头指示的位置）会出现下拉菜单，根据自己的硬件连接，选择正确的引脚，完成之后关闭窗口即可。

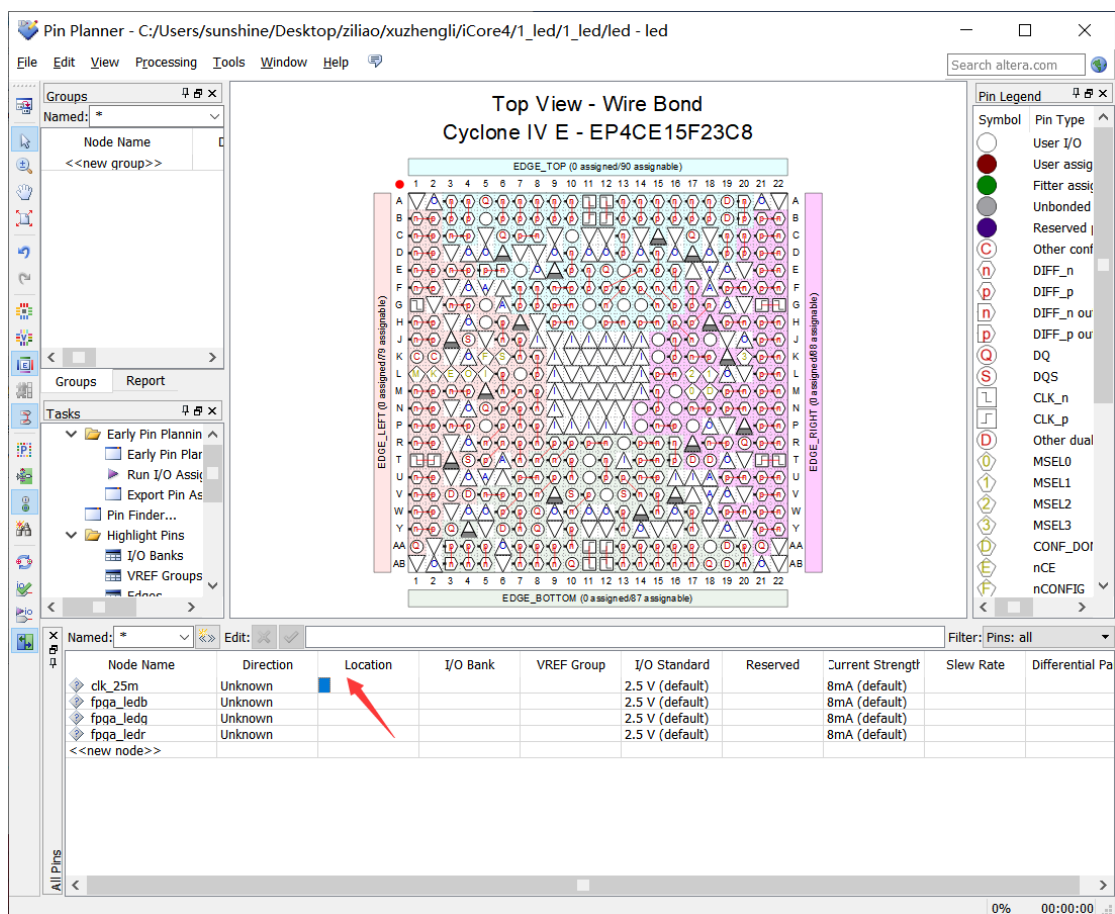


图 1-21

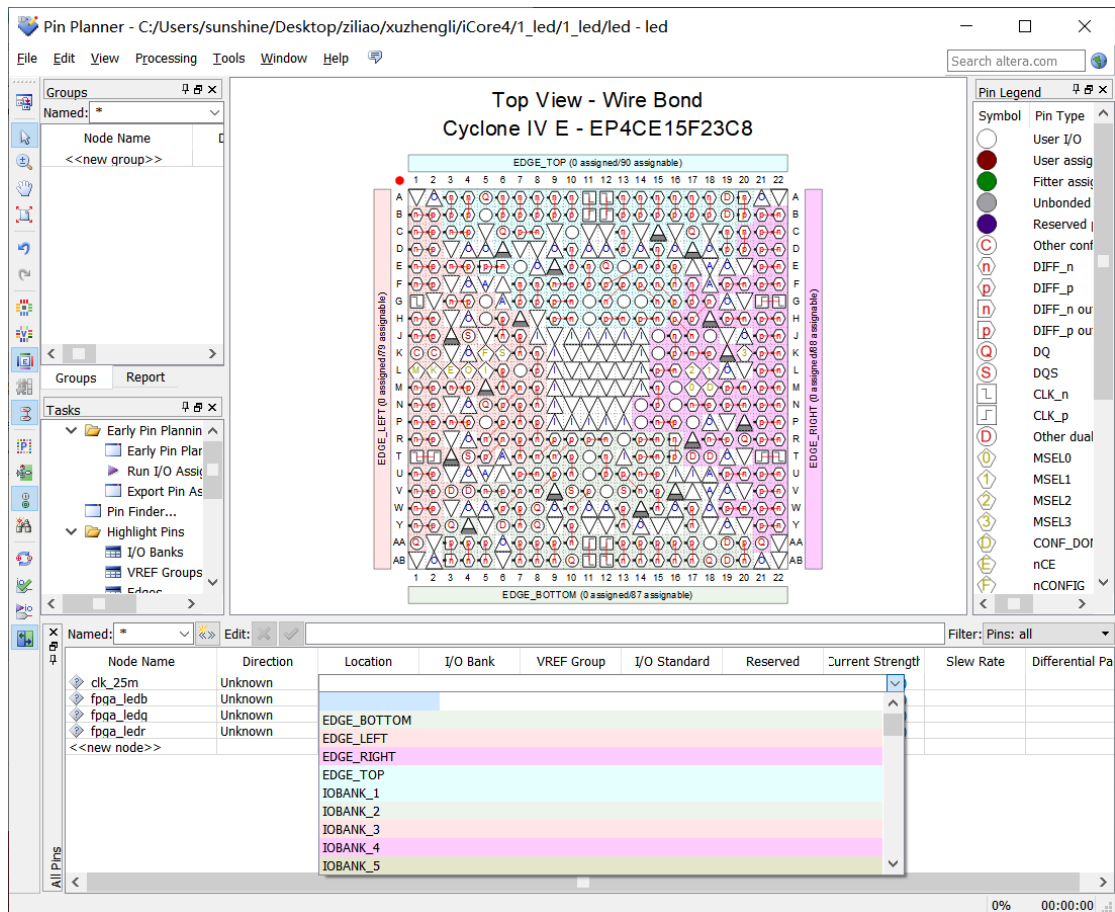


图 1-22

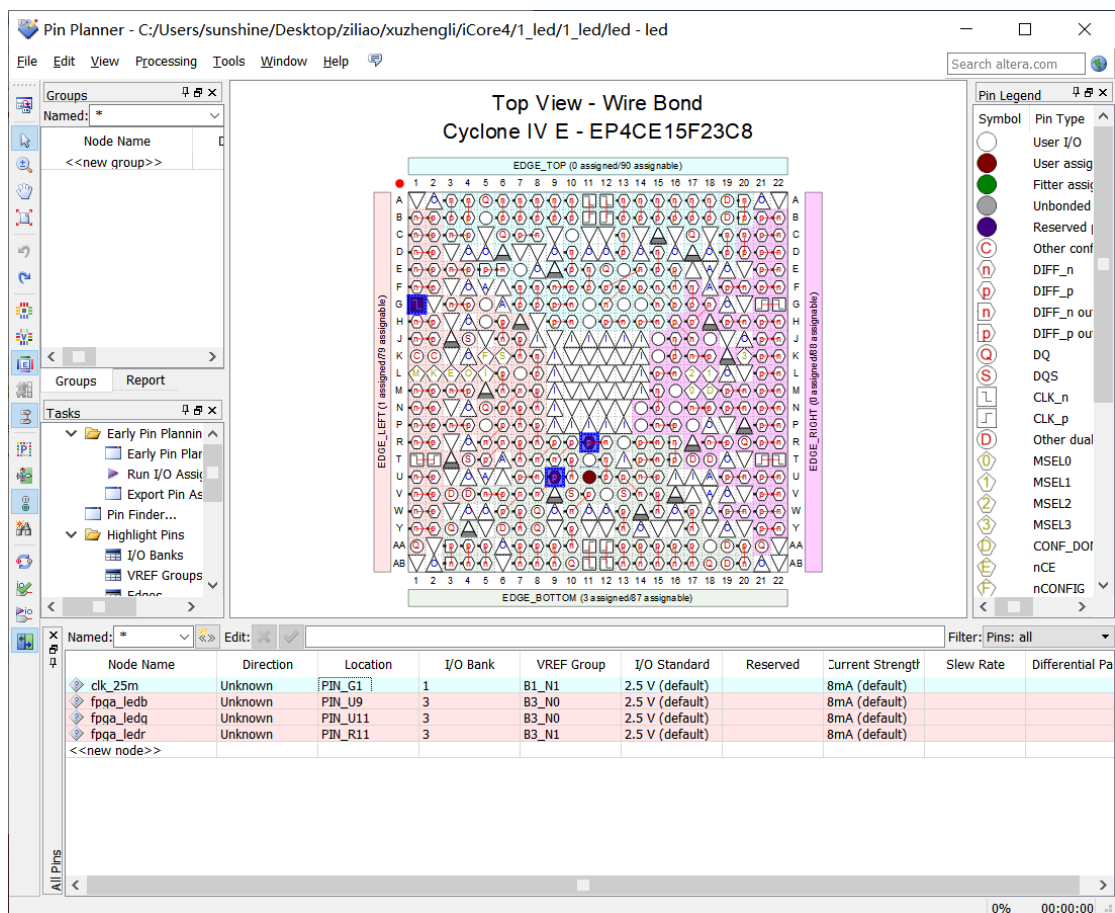


图 1-23

19、接下来再次编译程序，点击 processing->start compilation 或者直接点击图 1-24 小方框中的图标。可以在图 1-25 左侧看到编译进度，图示是编译完成的状态，右侧方框处是生成的报告（工程占用的资源数及引脚数等信息）。

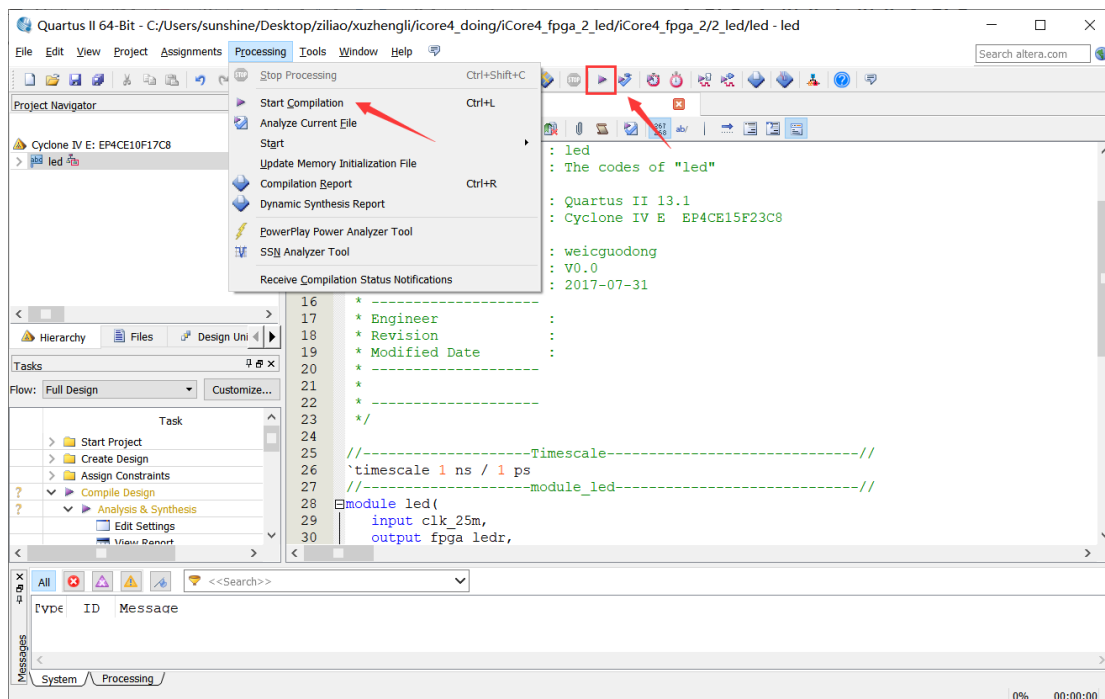


图 1-24

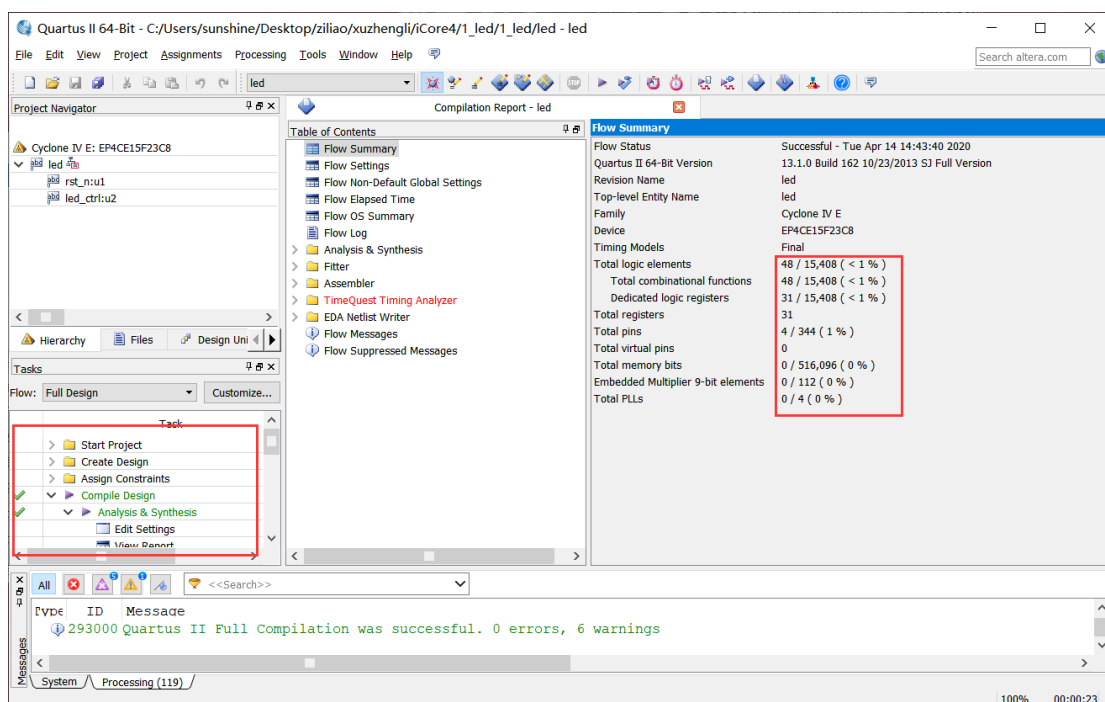


图 1-25

20、按照上一实验中介绍的配置文件下载教程将配置文件下载至硬件，观察实验现象。

五、实验现象

iCore4 双核心板上与 FPGA 相连的三色 LED (PCB 上标示为 FPGA-LED), 红色、绿色、蓝色, 快速交替闪烁。