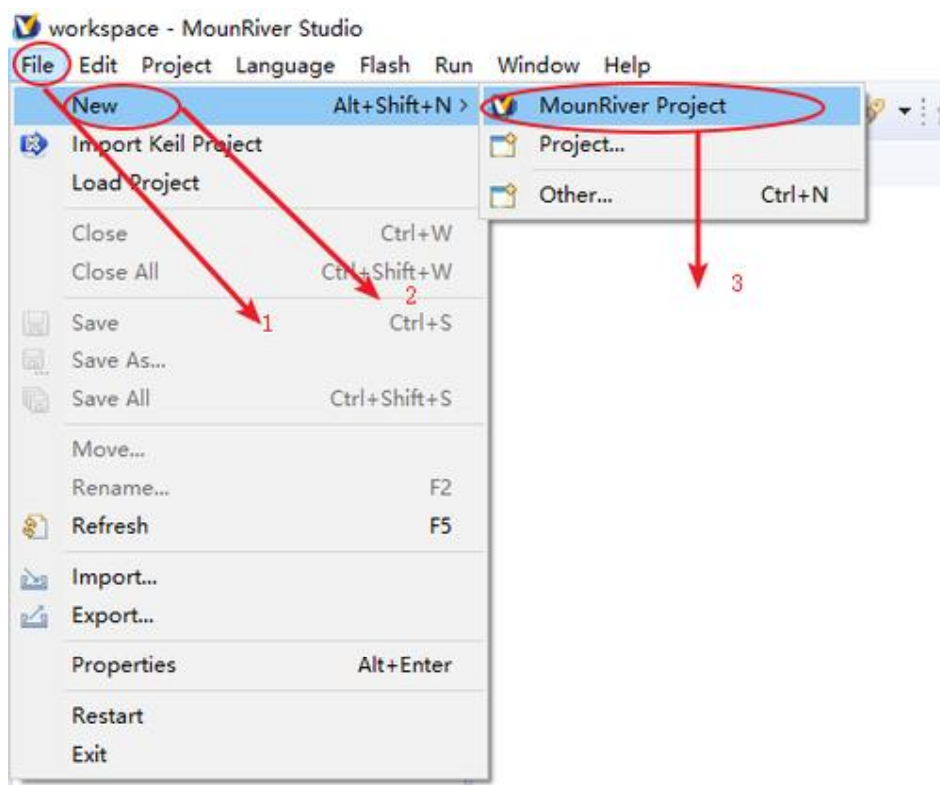


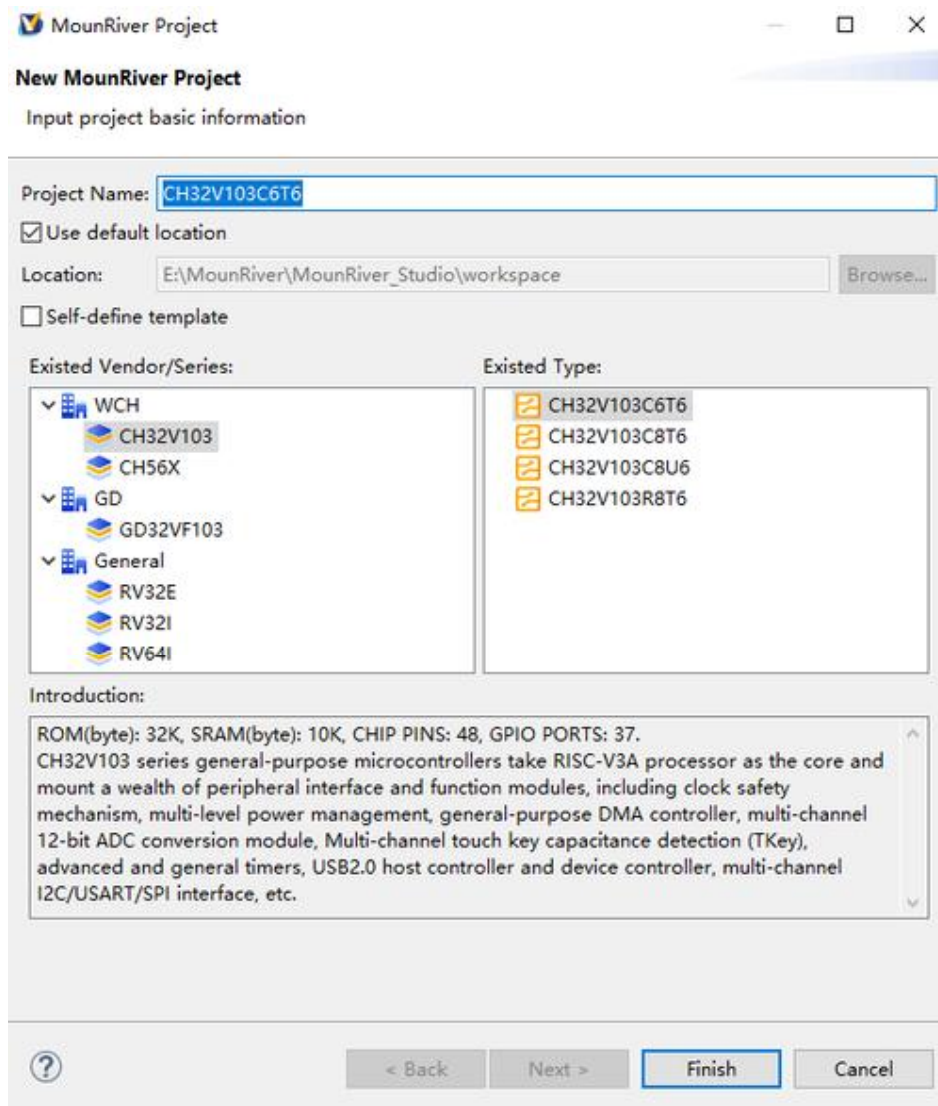
## 1、新建项目工程

在进行CH32V103相关外设应用学习之前，我们首先使用MounRiver Studio新建一个工程，以GPIO教程（点亮LED）为例，具体操作步骤如下：

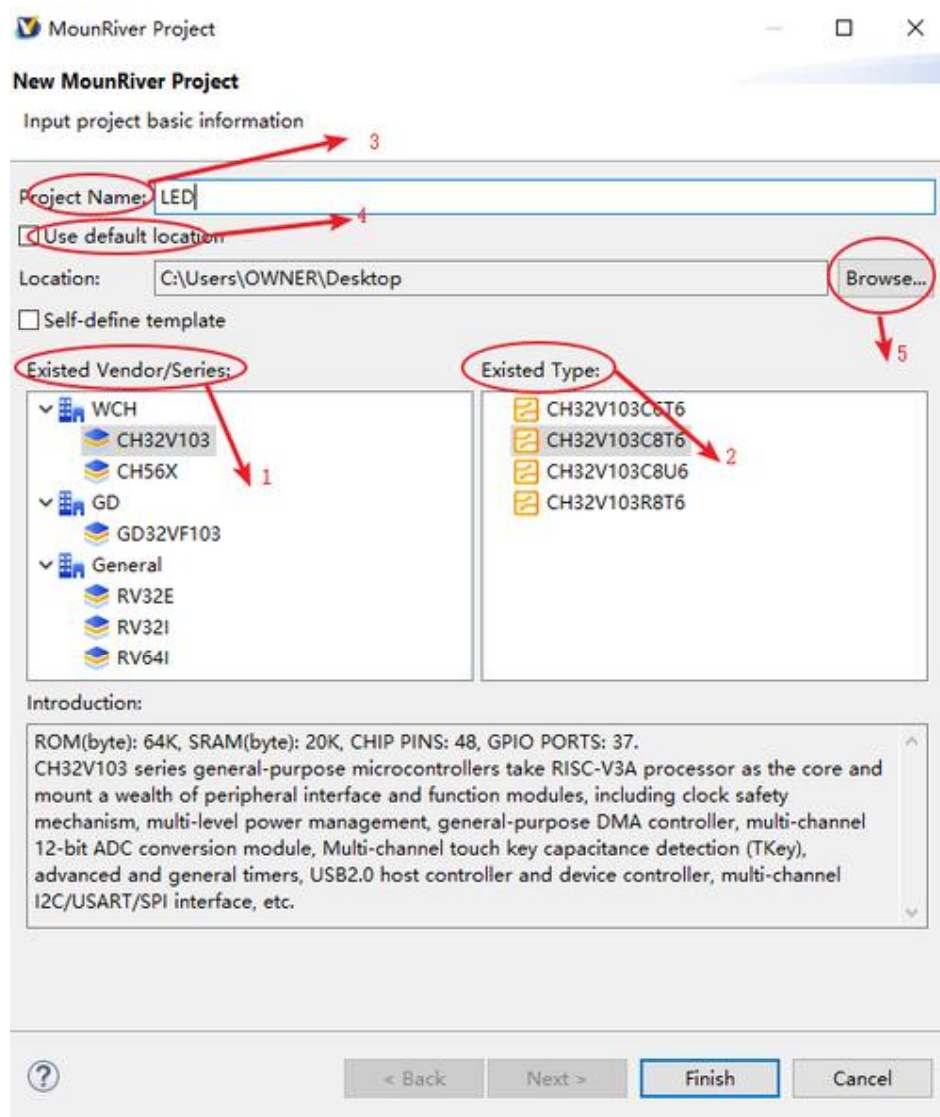
**第一步：**创建LED工程。打开MounRiver Studio,点击File-->New-->MounRiver Project



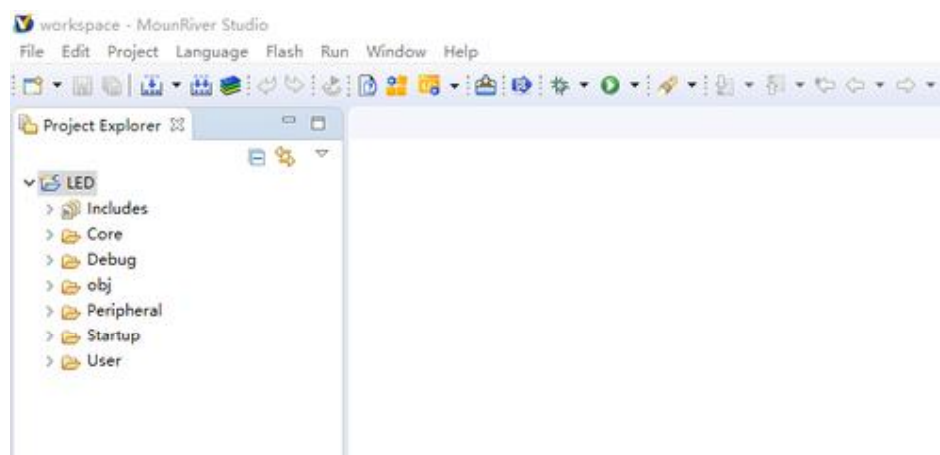
点击MounRiver Project之后，出现如下界面：



**第二步：**设置型号选型、工程名称和存放位置。选择Existed Vendor/Series和Existed Type-->更改工程名称-->取消Use default location勾选-->自定义文件位置；其中“更改文件名称”和“取消Use default location勾选”可不进行操作，使用默认设置，但此处建议进行修改。由于本次实验以点亮LED为例，修改工程名称为LED，工程存放位置设置为桌面，Existed Vendor/Series这一项选择CH32V103，Existed Type选择CH32V103C8T6。最后点击Finish完成工程创建。



工程创建完成之后，MounRiver Studio编译页面及桌面文件夹如图所示：

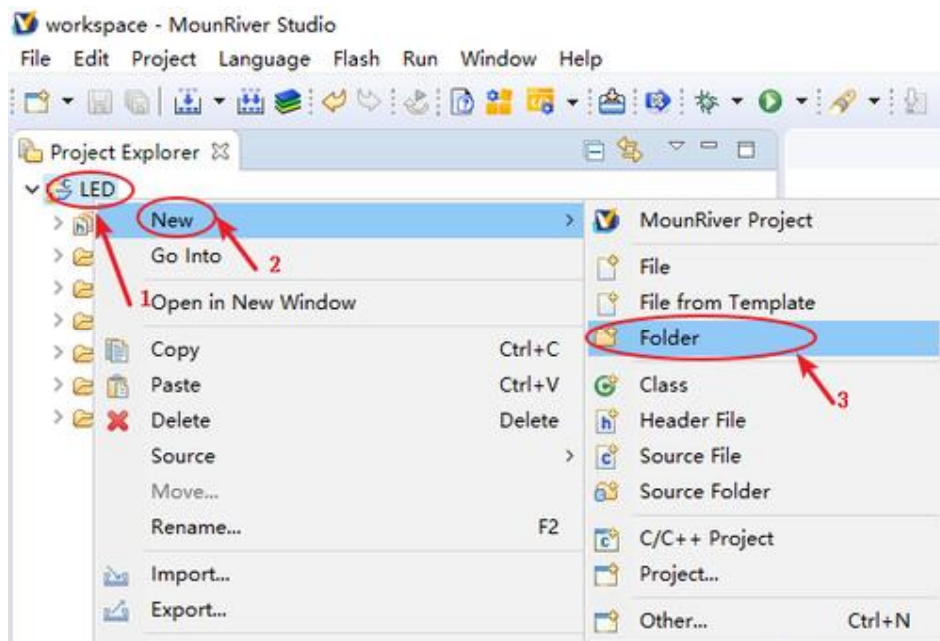


名称	修改日期	类型	大小
.settings	2020/9/28 19:58	文件夹	
Core	2020/9/28 19:58	文件夹	
Debug	2020/9/28 19:58	文件夹	
obj	2020/9/28 19:58	文件夹	
Peripheral	2020/9/28 19:58	文件夹	
Startup	2020/9/28 19:58	文件夹	
User	2020/9/28 19:58	文件夹	
.cproject	2020/9/28 19:58	CPROJECT 文件	24 KB
.launch	2020/9/28 19:58	LAUNCH 文件	6 KB
.ld	2020/8/24 16:16	LD 文件	4 KB
.project	2020/9/28 19:58	PROJECT 文件	1 KB
.tmpinfo	2020/9/28 19:58	TMPINFO 文件	1 KB
LED.wvproj	2020/9/28 19:58	WVPROJ 文件	1 KB

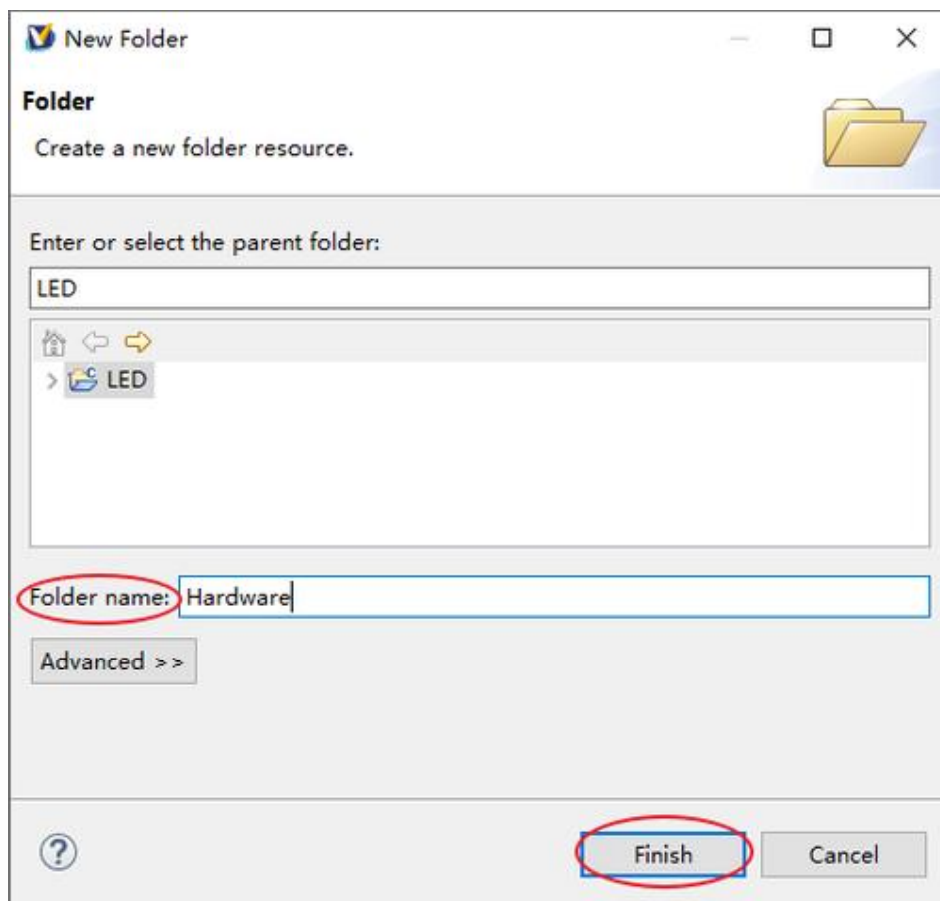
创建的工程目录清单说明如下：



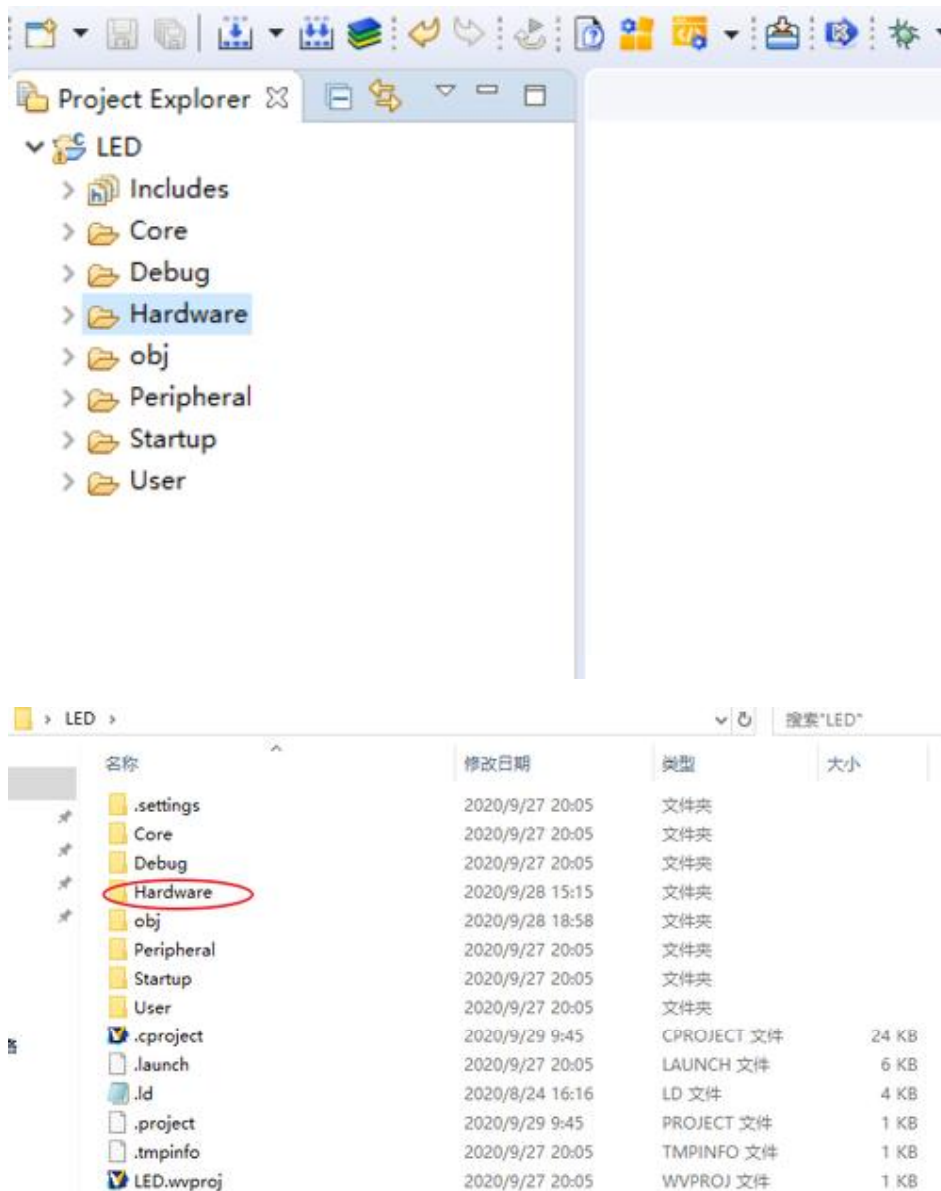
**第三步：**添加Hardware文件夹。在工程下面建立一个Hardware文件夹，用于存放保存每个实验外设驱动文件的子文件夹，具体步骤如下：选中LED右击-->选中New-->选中Folder左击，操作步骤如图所示：



点击选中Folder之后，出现如下界面，在Folder name中输入Hardware，点击Finish完成。

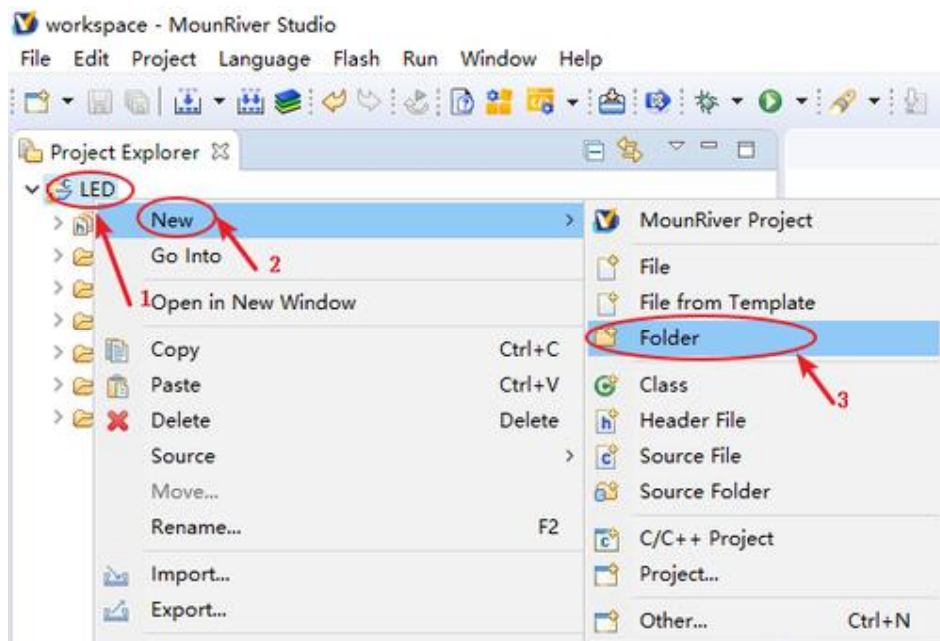


点击Finish完成之后编译器界面及文件夹界面如下：

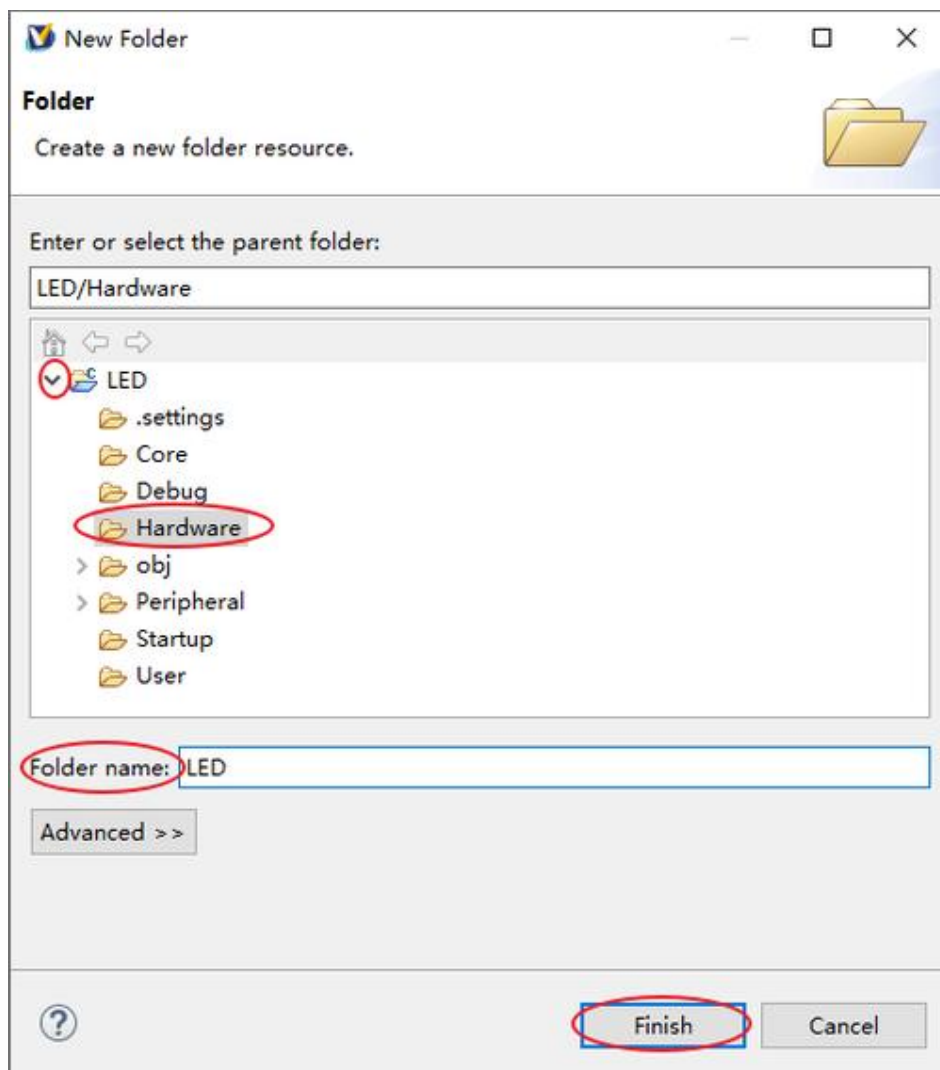


**第四步：**添加LED子文件夹。因为本次教程以点亮LED为例，因此可在Hardware文件夹下新建一个LED子文件夹，用于接下程序编写存放led.c和led.h文件，具体操作步骤与第三步类似，如图所示：

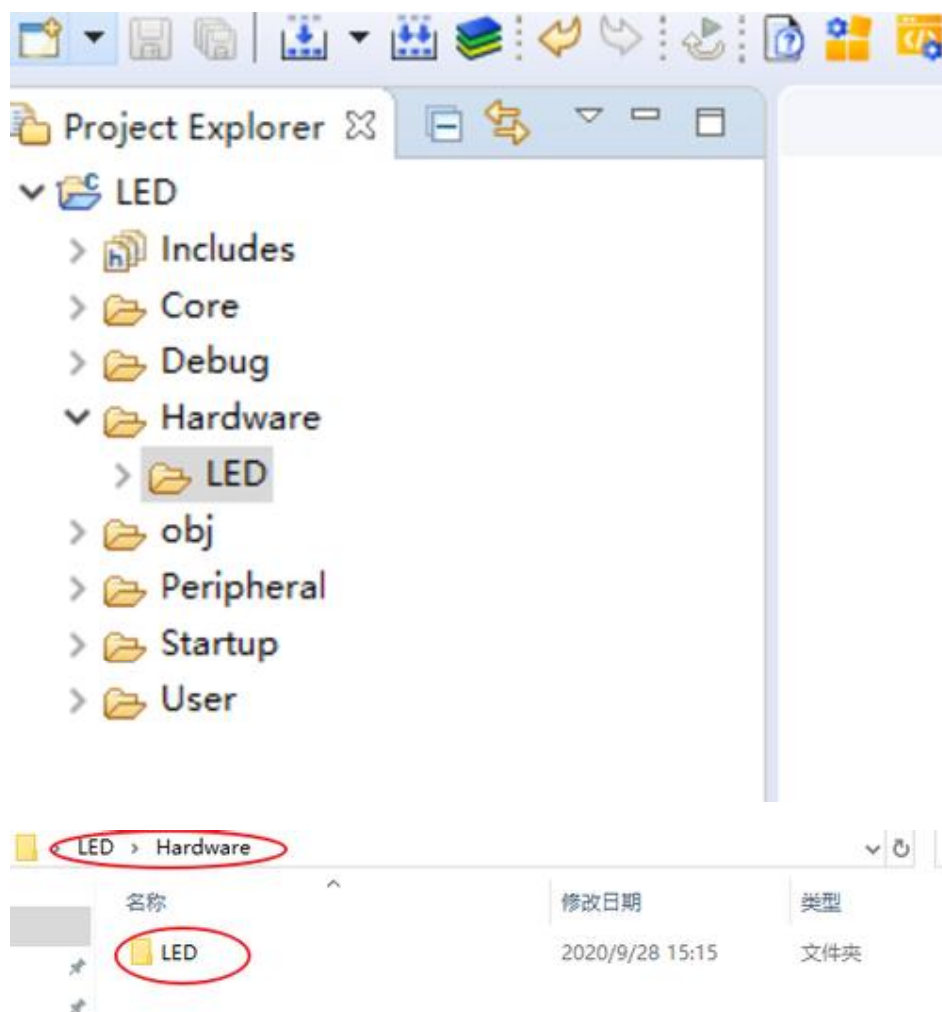




点击选中Folder之后，出现如下界面，点击图中标注处将LED工程展开，选中Hardware，在Folder name中输入LED，点击Finish完成。



点击Finish完成之后编译器界面及文件夹界面如下：

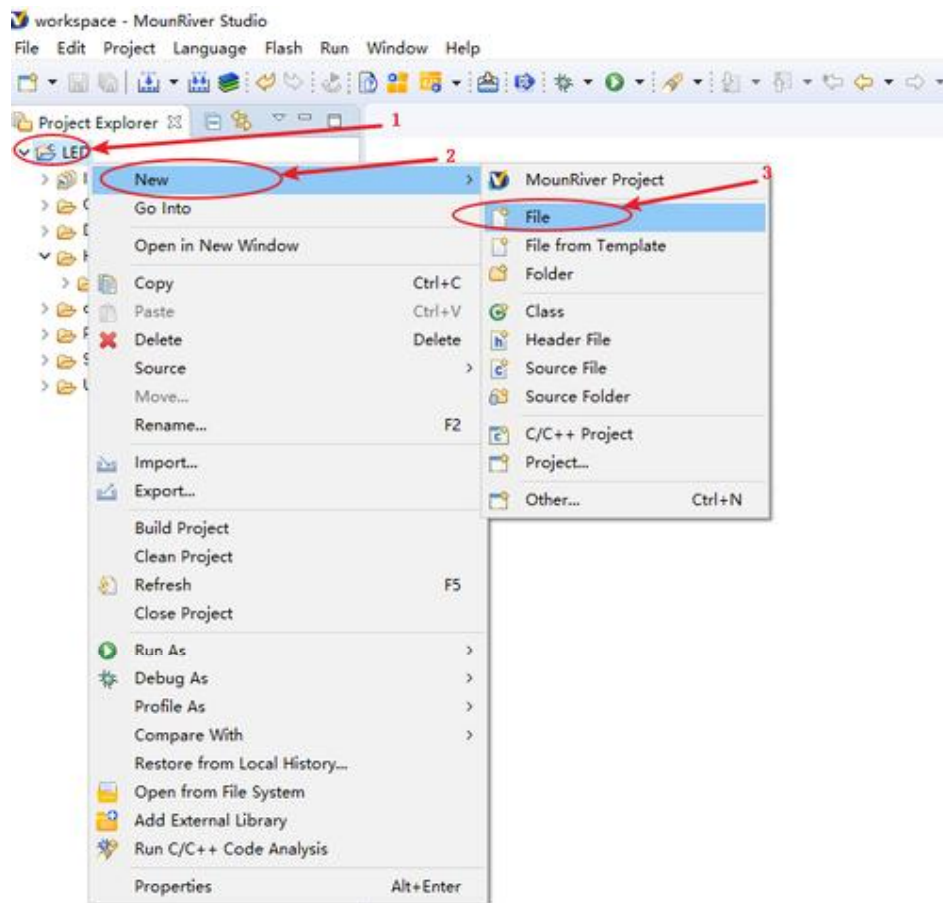


## 2、添加相关外设驱动文件

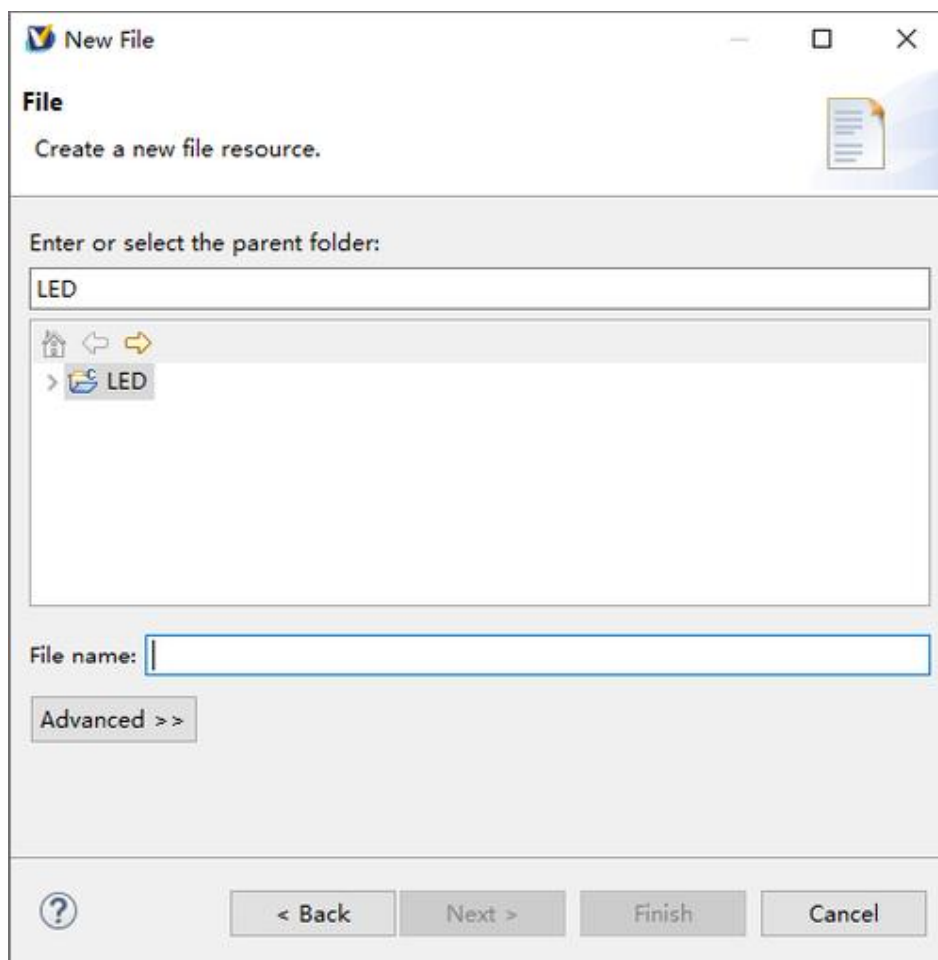
第一节第四步讲述如何在Hardware文件夹下添加一个LED子文件夹，本节我们将讲述如何添加led.c和led.h两个文件并放入LED子文件夹中，具体步骤如下：

**第一步：**选中LED-->选中New-->点击File

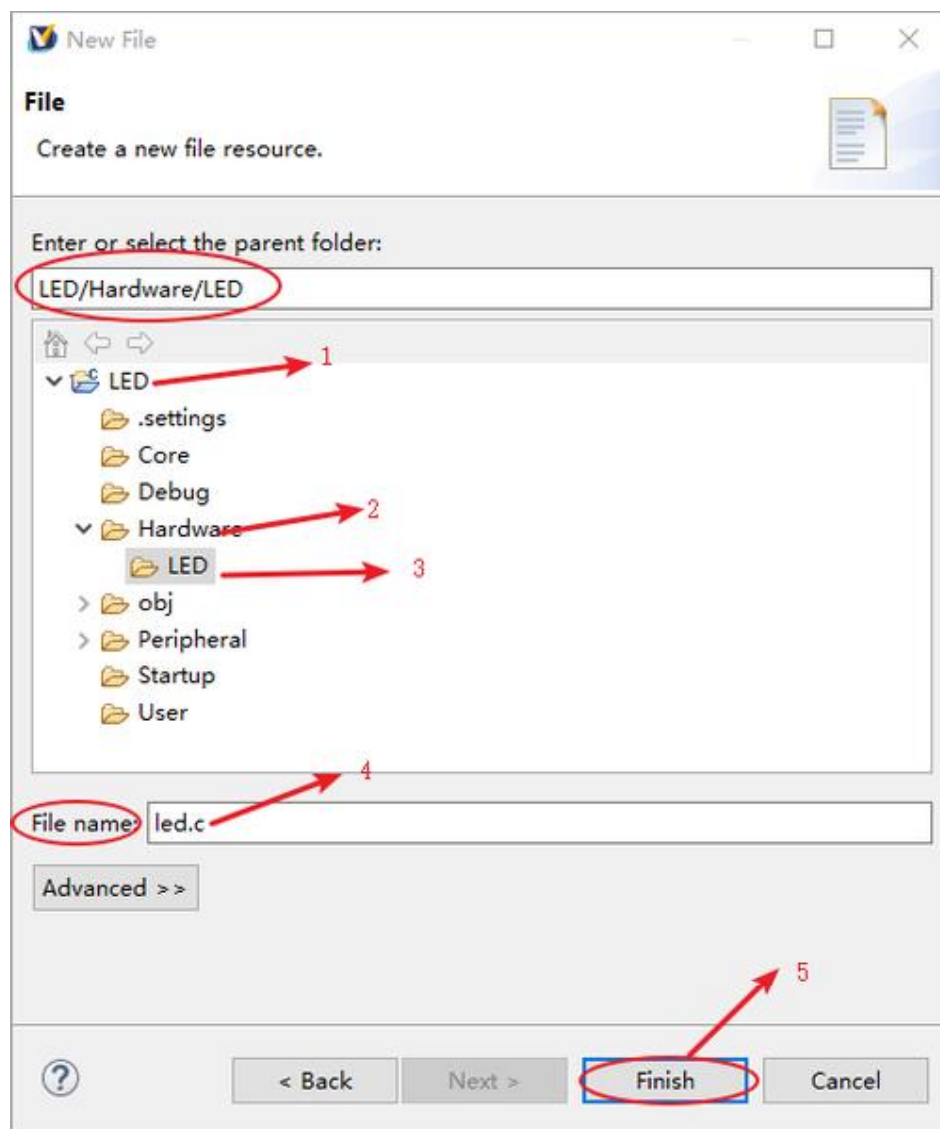




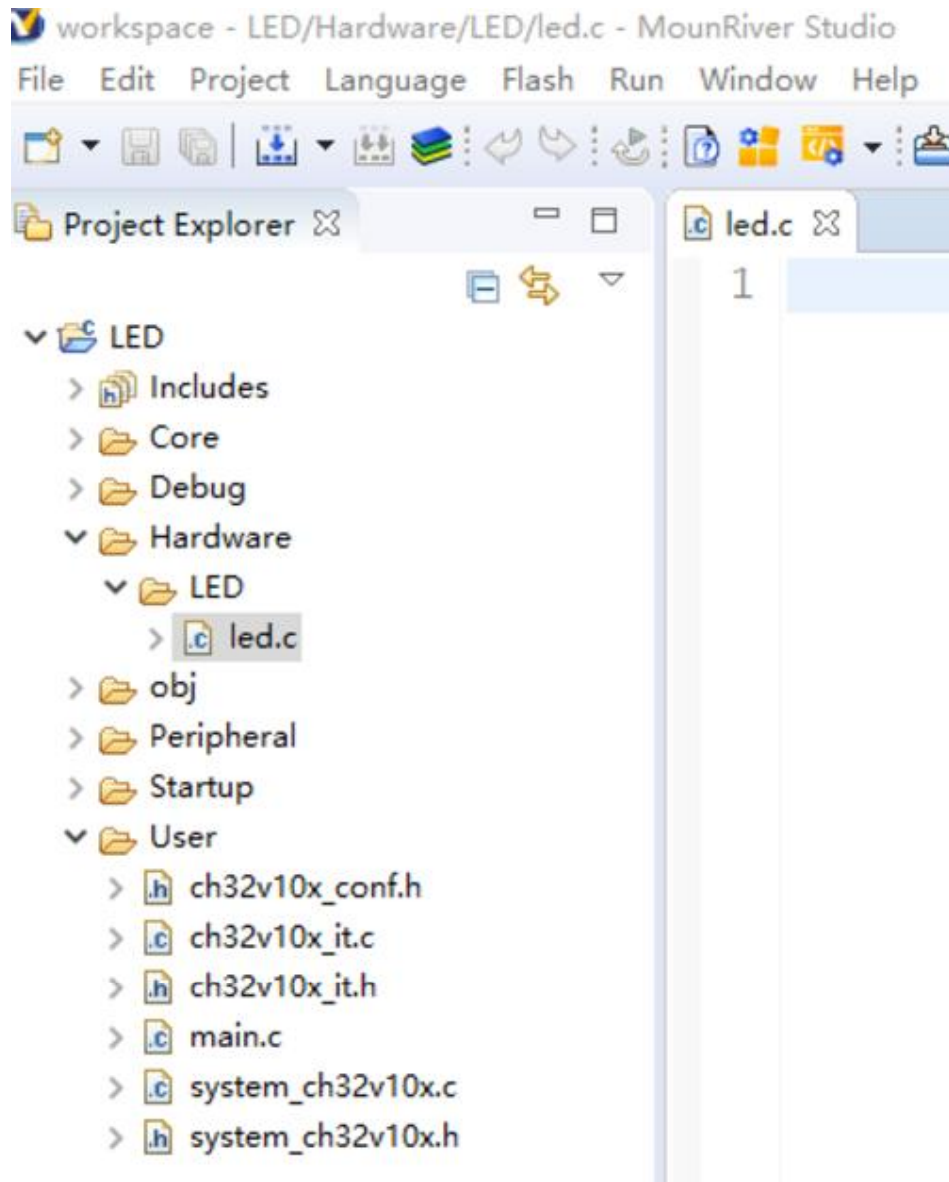
点击File之后，出现如下界面：



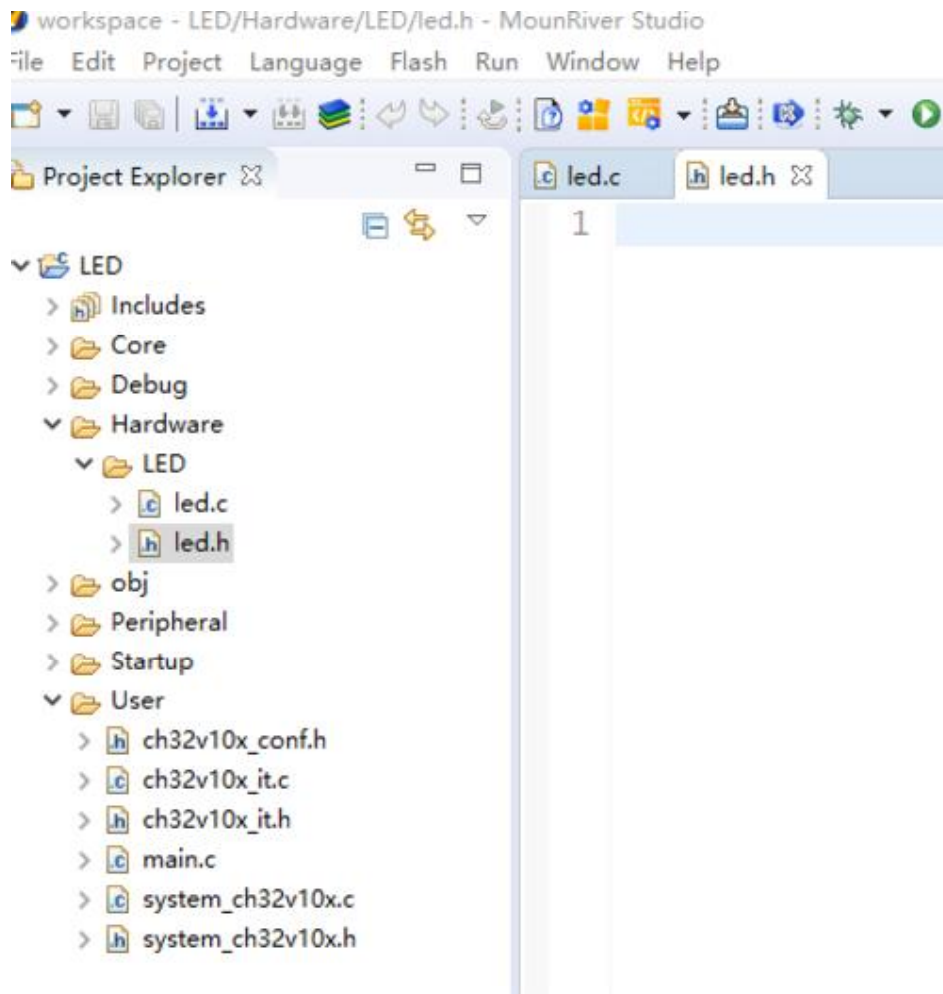
**第二步：** 点击LED展开，选中Hardware文件夹下LED子文件夹，File name中命名为led.c，点击Finish完成。如图所示：



点击Finish完成之后，界面如图所示：



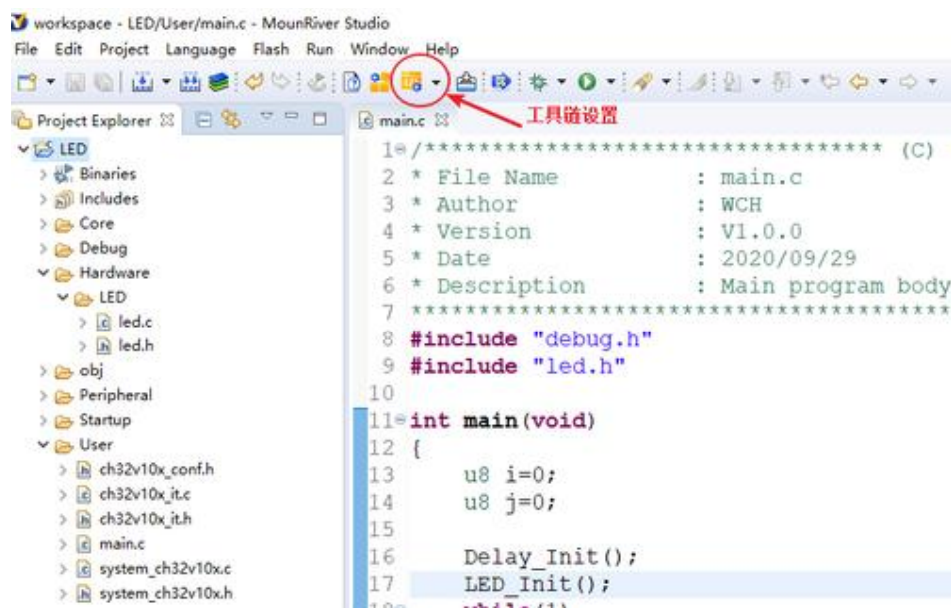
led.h文件建立过程与上述步骤一致，led.c和led.h两个文件建立完成之后，显示界面如下所示：



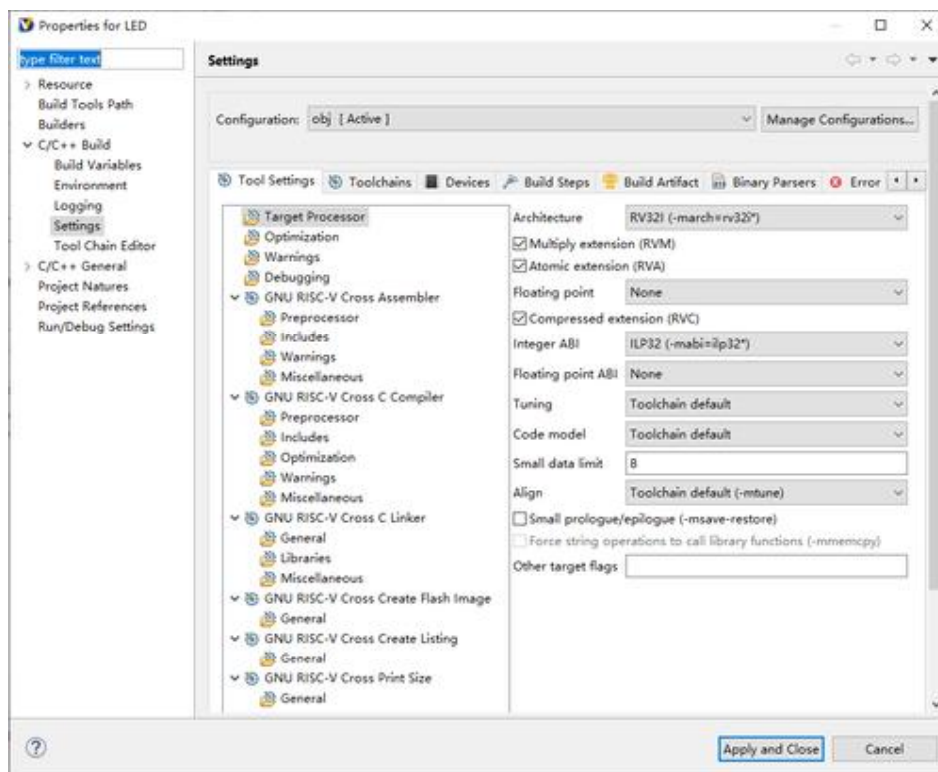
### 3、添加路径

完成工程建立之后，需添加相关文件路径，否则程序代码编写完成编译会出错。本节介绍如何添加Hardware文件夹下子文件夹及相关文件的路径，具体步骤如下图所示：

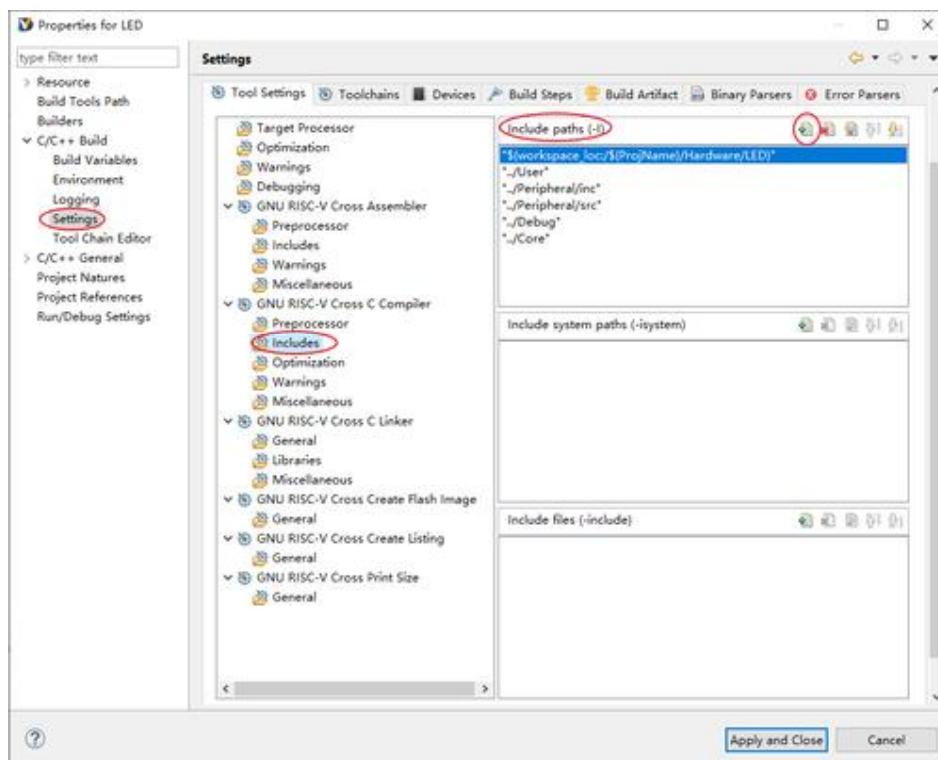
**第一步：**选中工具链设置按钮左击；



工具链设置按钮左击之后出现如下界面：

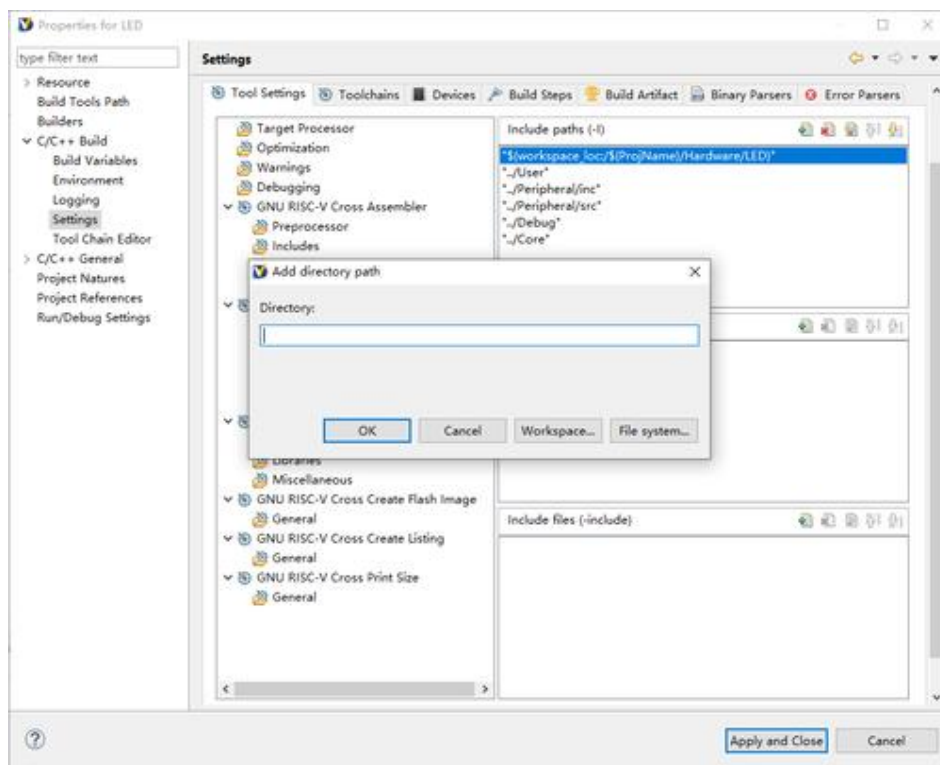


**第二步：**左击选中C/C++ Build下的Settings-->选择Tool Settings-->选择GNU RISC-V Cross C Compiler下的includes-->点击Include paths右边的绿色加号按键；

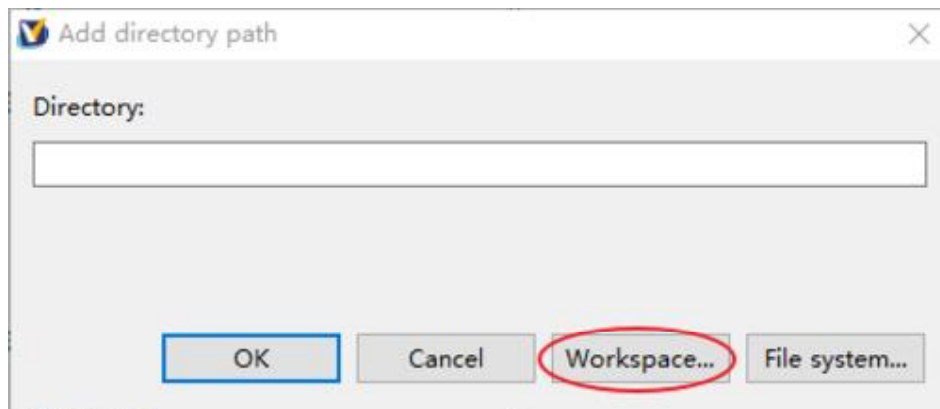




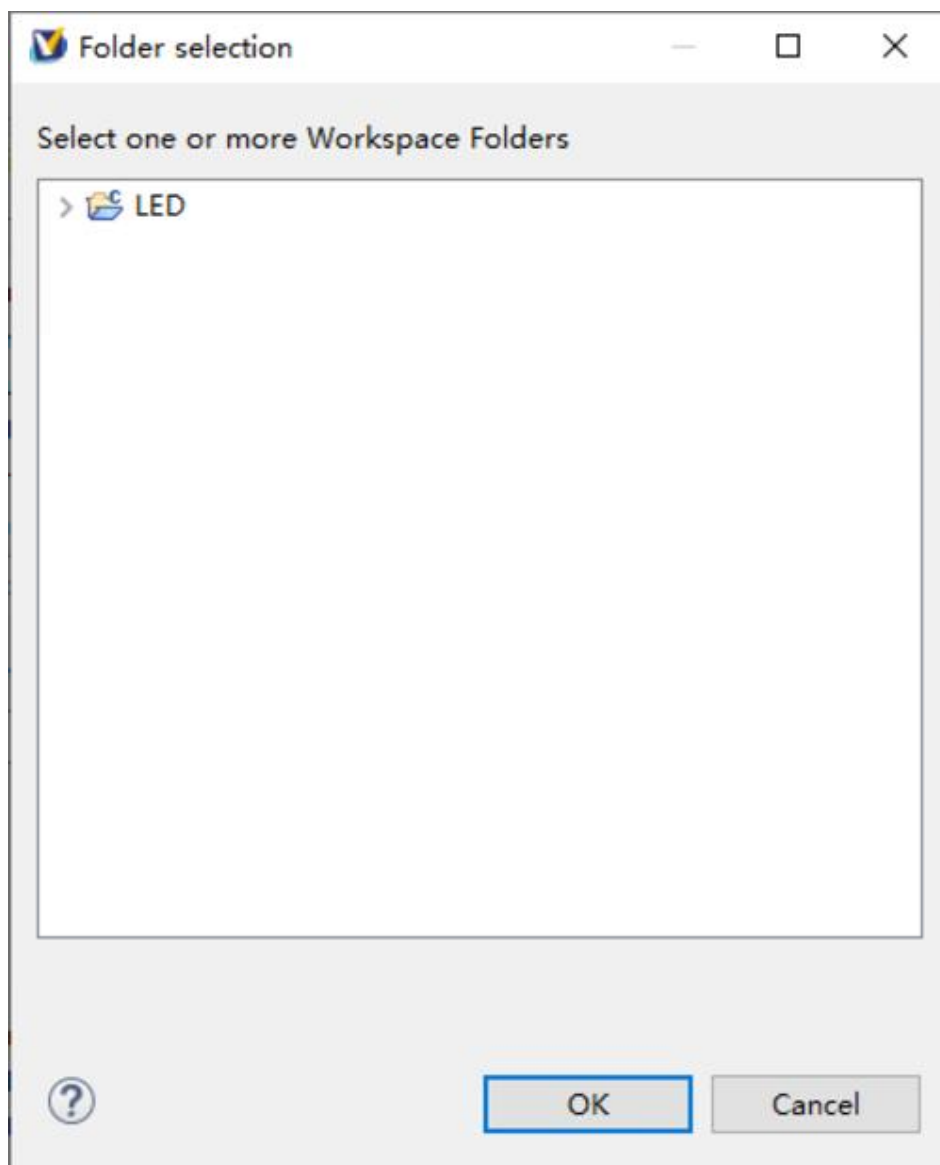
点击Include paths右边的绿色加号按键之后出现如下界面：



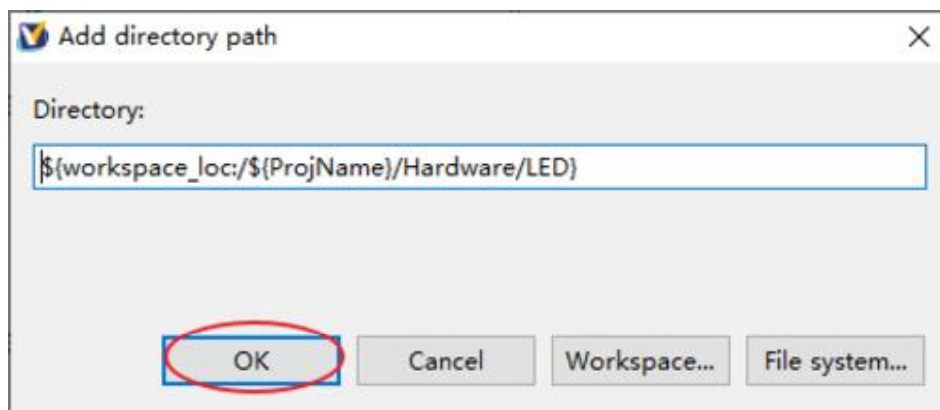
**第三步：** 点击Workspace；



点击Workspace之后出现如下界面：



**第四步：**展开LED文件下-->选中Hardware文件夹下LED子文件夹-->点击OK之后出现如下界面-->再次点击OK-->然后点击Apply and Close;



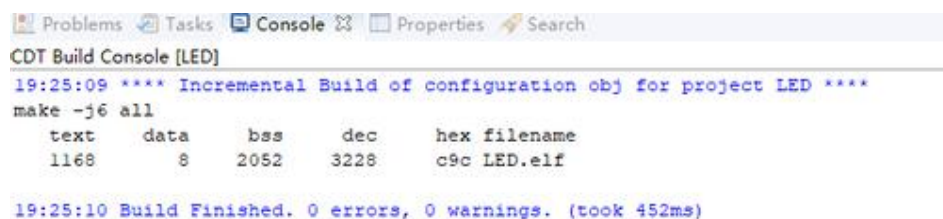
当添加路径完成之后，即可编写代码并编译运行。

## 4、编译及下载验证

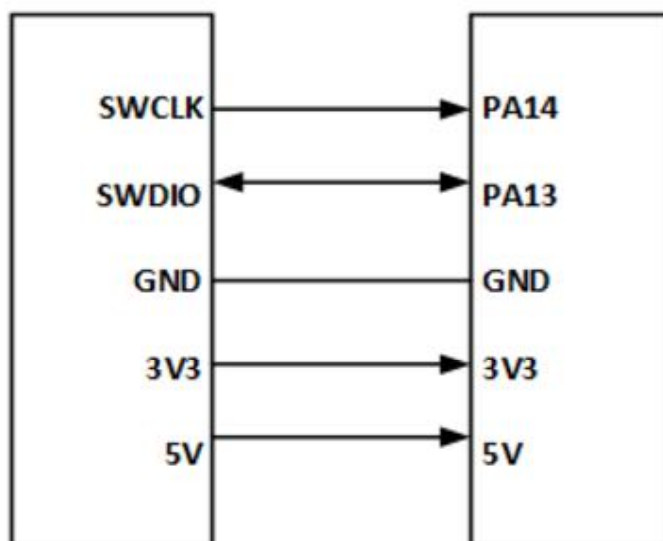
代码编写完成之后，需要对当前工程进行编译运行，具体编译按钮操作如图所示：



图中：1为增量编译当前工程；2为重新编译；3为编译全部工程。编译无错后会出现如下提示：



确认程序没有错误之后，下载到开发板运行，此处需要用到WCH-Link模块。WCH-Link模块与开发板的具体连接如下图所示（其中3.3V和5V任选一个连接即可）：



将WCH-Link模块与开发板连接之后，WCH-Link模块插入电脑USB口

进行程序下载，具体操作步骤如下：

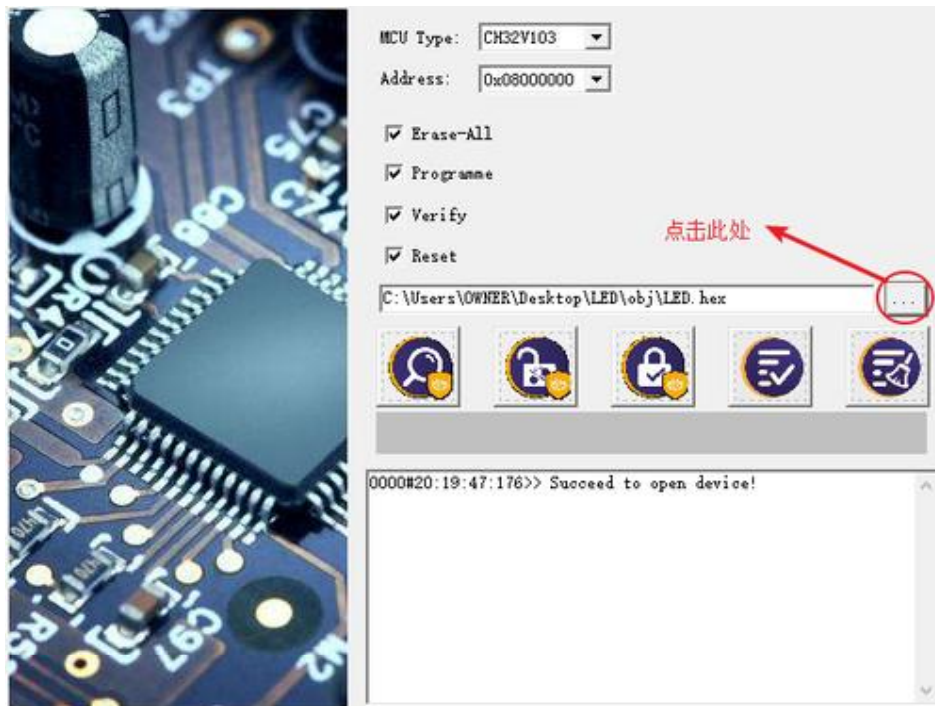
**第一步：** 点击MounRiver Studio上方Download按钮，如图所示：



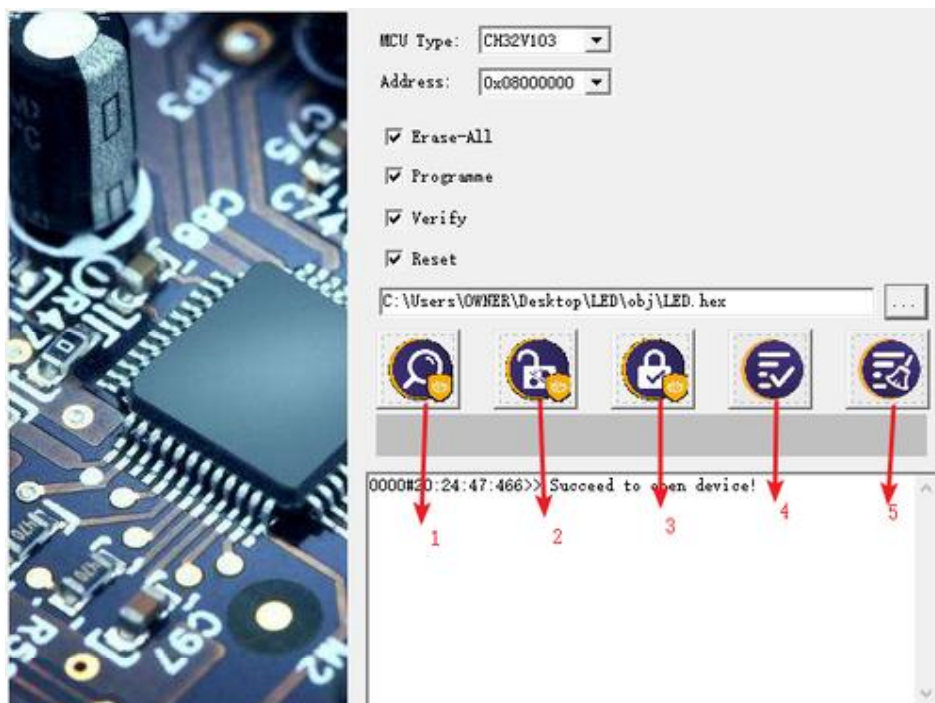
点击Download按钮之后出现如下界面：



**第二步：** 点击下图标注处，选择工程目录obj文件夹下hex文件打开，之后如下图所示：



**第三步：**检查是否处于读保护状态，若处于读保护状态，则取消读保护，然后点击下载按钮，进行程序下载，具体操作如下图所示。图中，1为查询设备读保护状态，2为解除读保护，3为设置读保护，4为执行下载按钮，5为清除输出信息按钮。



程序下载成功之后，会出现如下界面，表示下载成功；



