



**北京理工大学**  
BEIJING INSTITUTE OF TECHNOLOGY

# 单片机课程设计

## 软件使用与焊接

良乡工训楼810实验室



# KEIL软件使用



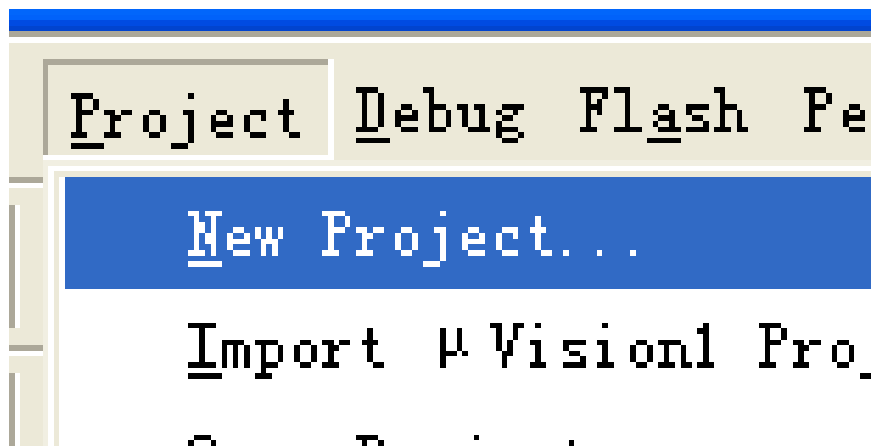
# 主要步骤

- 建立工程
- 编译链接
- 调试程序

# 建立工程

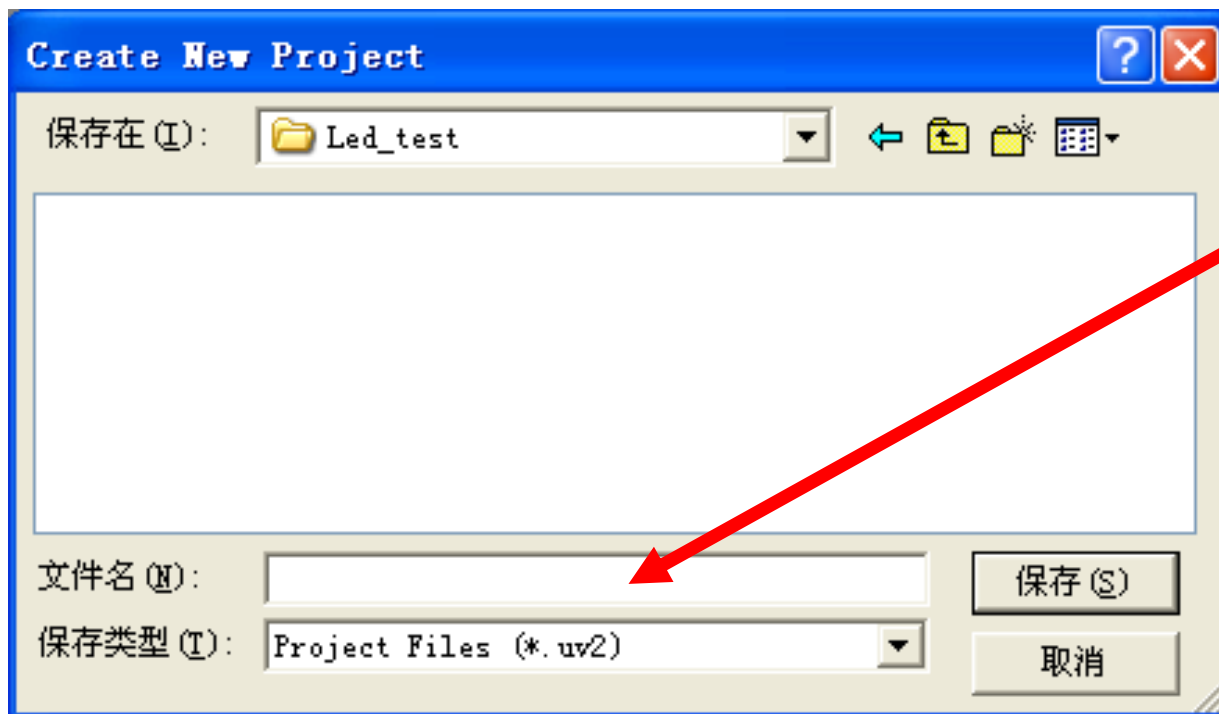


- **打开软件** (双击桌面上的Keil软件图标  进入)
- **建立新的项目文件**
  - 在下拉菜单中选择Project选项，然后选择New Project。



## ➤ 给新建的项目取一个名字

➤ 取名的时候请注意，一般项目的名字都要能代表项目的内容，不要取一些没有任何意义的名字。

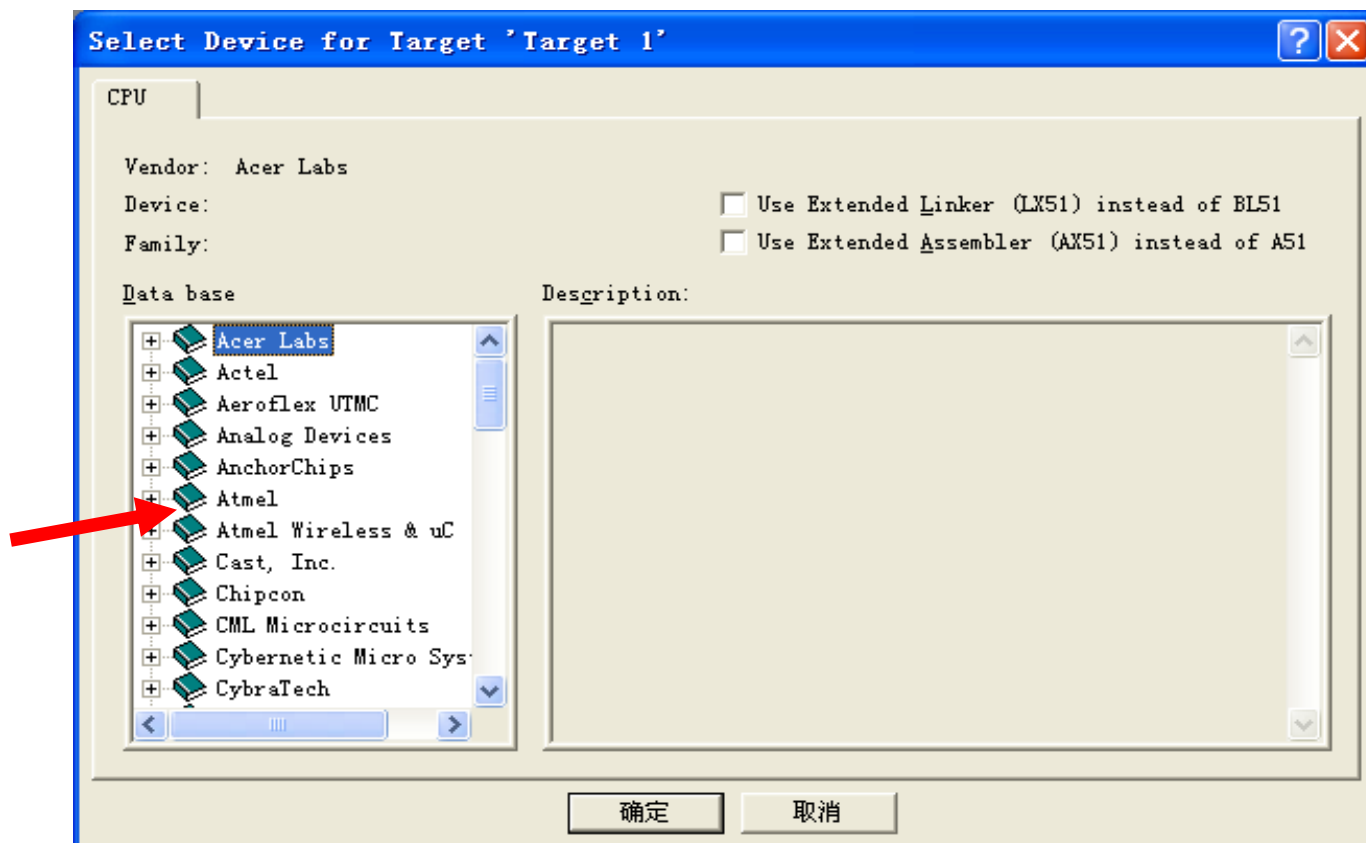


**注意：**在这里只需要写出项目的文件名即可，不需要添加后缀，后缀名默认为.uv2

## ➤ 选择CPU类型

- 在以下的列表中选择所用的CPU型号，选好了以后双击该型号，或者选中后点击“确定”即可。

在本学期的实验中，我们选择Atmel公司的AT89C52型号的单片机就可以了。或者说我们选择是51内核的单片机就可以了。

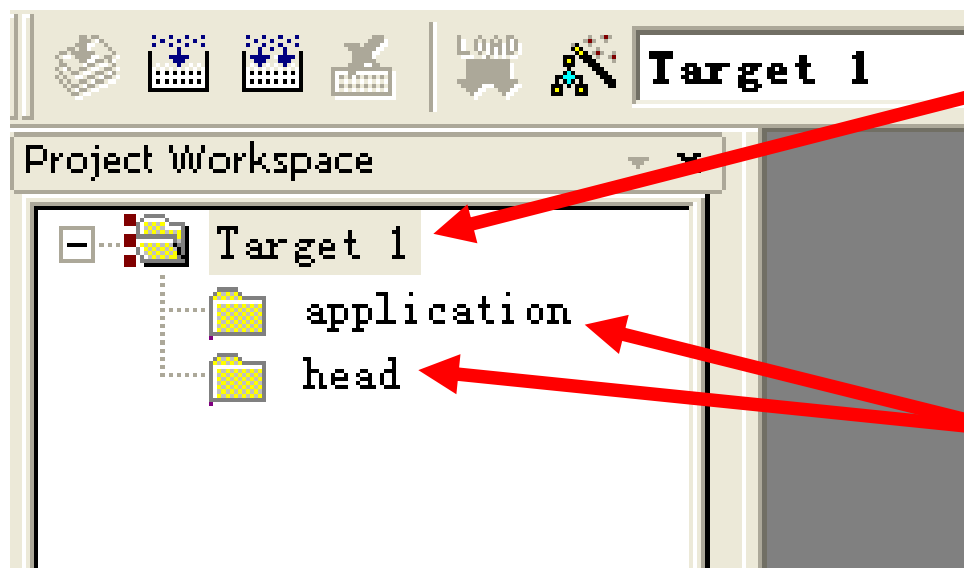


## ➤ 选择是否添加启动代码

- 在选择完CPU的型号以后，会出现是否添加启动代码选项，在这里选择“是 (Y)” 选项即可。



➤ 选择完以后就可以看到已经建立好的项目在Project Window中显示出来了。



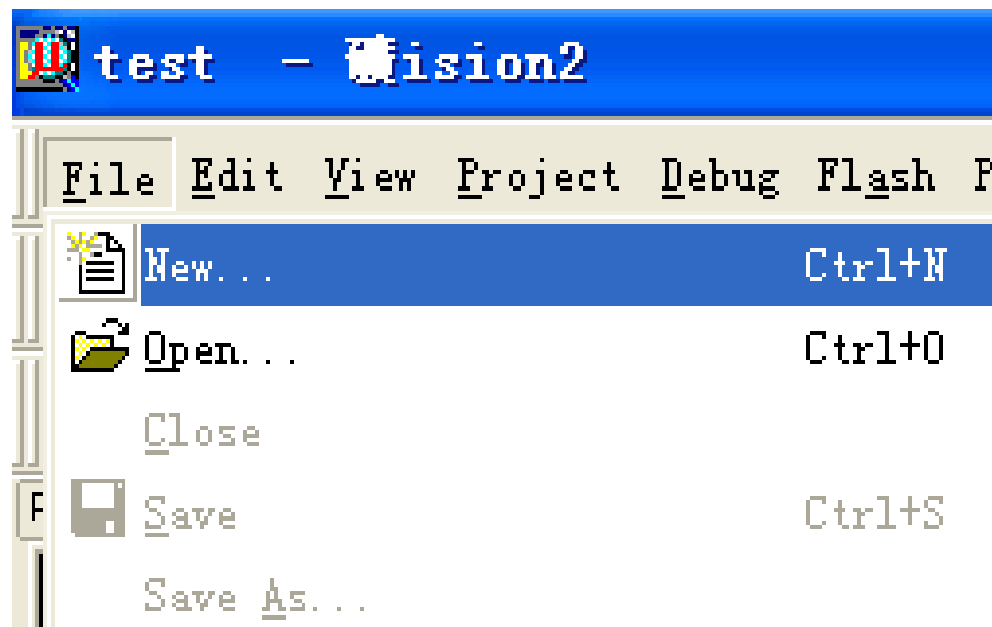
项目文件夹

项目中文件夹，可以自己建立多个，放置不同的文件。



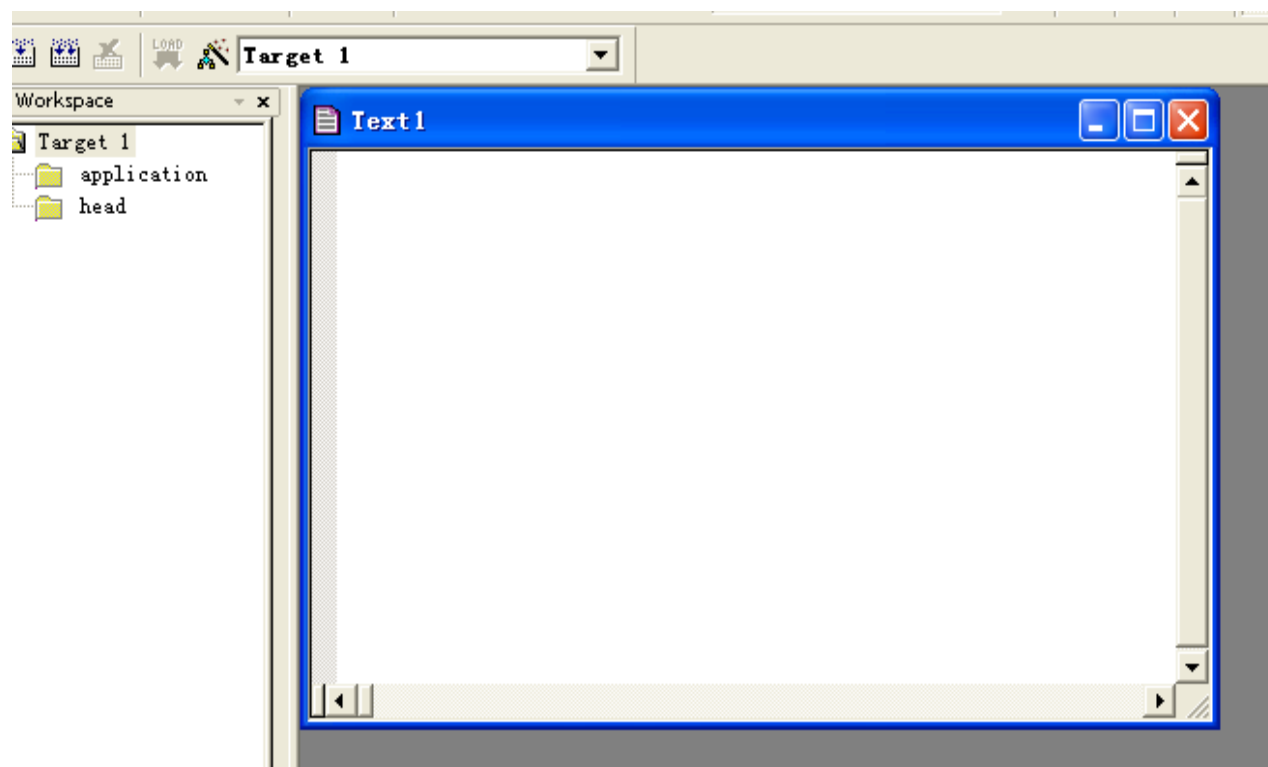
## ➤ 建立新的程序文件

➤ 在菜单中选择File，然后在选择New File，建立一个新的程序文件。



或者在快捷图标栏中  
单击图标 

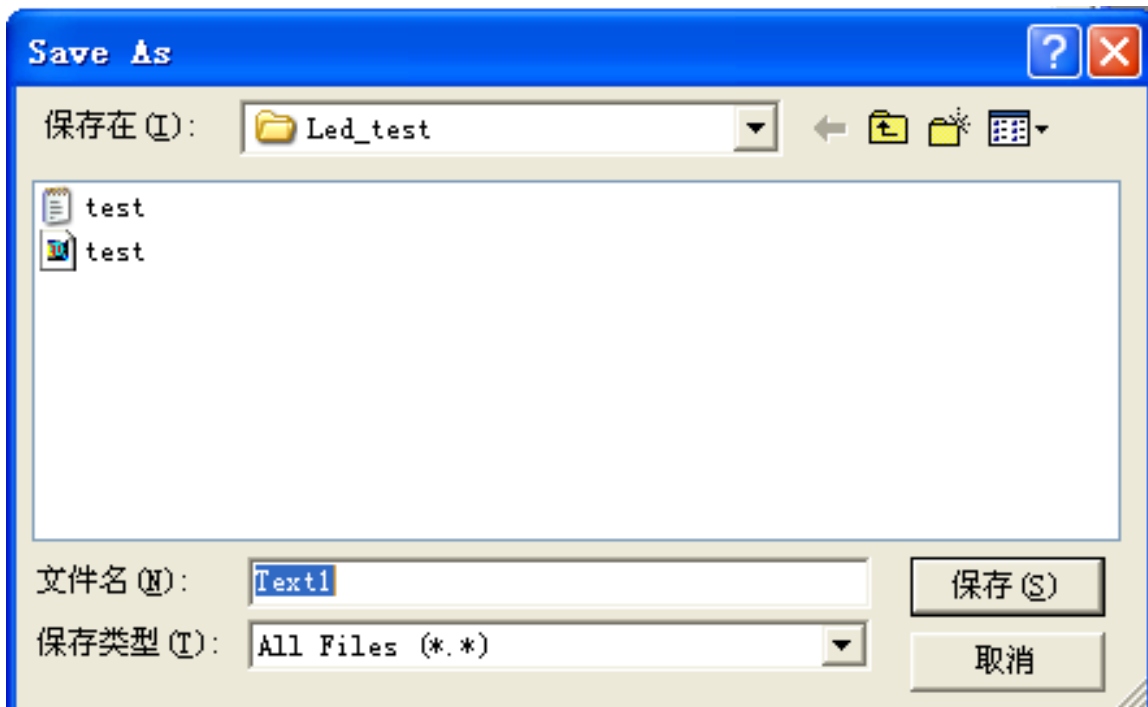
➤新建的程序文件类似一个空白的文本框，文本框标注的文件名为Text 1，Text 2，Text 3...，



然后选择存盘文件，  
会出现新的对话框。

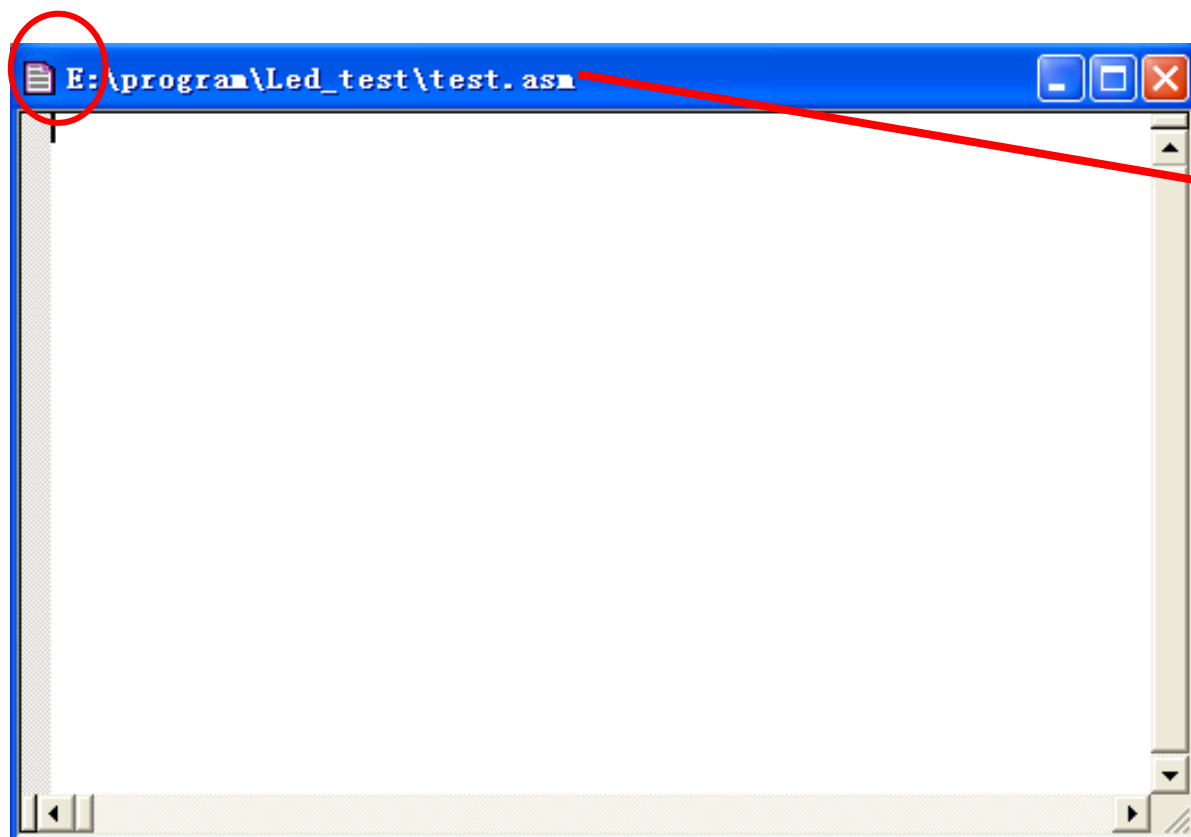
## ➤ 给新建的程序文件命名

➤ **注意：**文件名也应该有实际的意义。在取名时，**一定要给程序文件名带上后缀**，如果选择用汇编语言编写程序，添加的后缀名为`.asm`，如果选择用C语言来编写程序添加的后缀名为`.c`，如果不添加后缀名，将导致导入文件时出错。



文件存储的路径默认为项目所在的目录中，也可以选择将程序文件存在其他的路径中，这不会影响到程序的编译及调试过程。

建立好了文件以后就可以看到原来的文件对话框上面就已经有了文件地址和文件名了，这表示文件命名和确定存储地址已经完成。



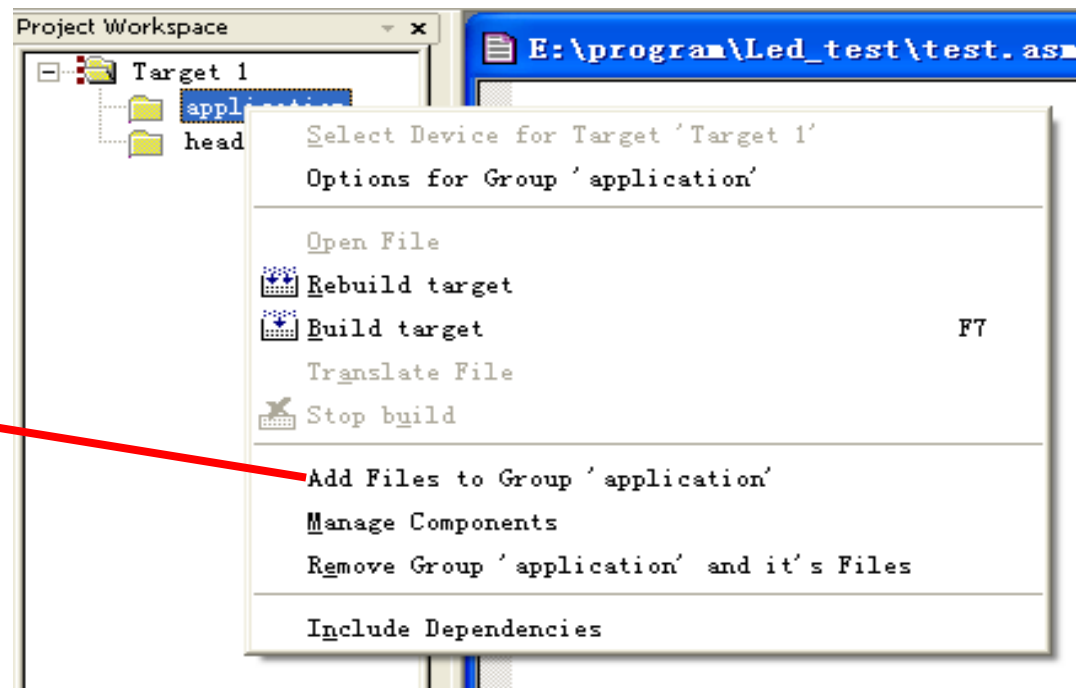
文件存放的路径和文件名。

## ➤在项目中添加程序文件

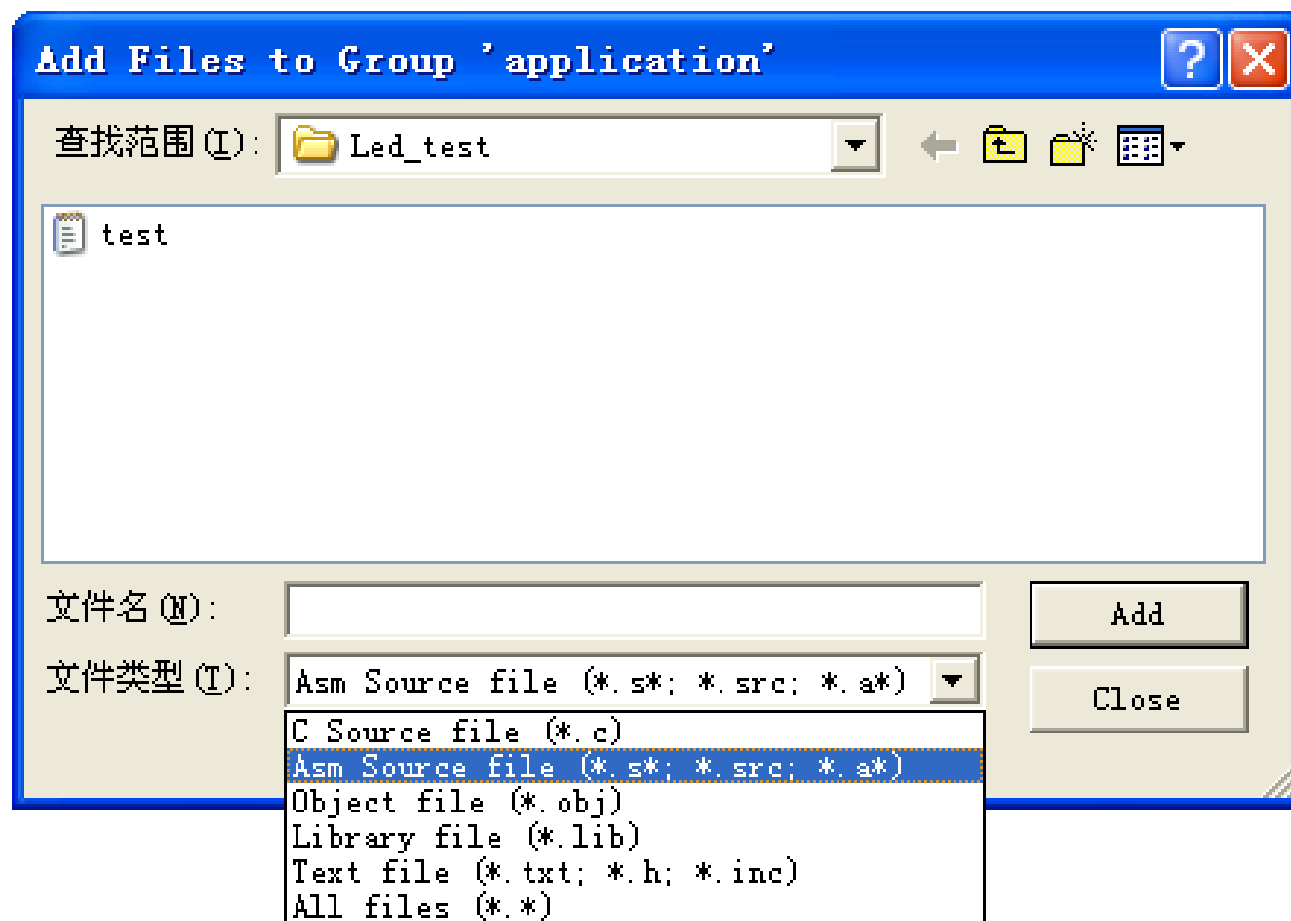
➤接下来，将程序文件添加到项目中去，这样程序文件就可以为项目服务了。添加的方法如下：

用鼠标右键单击项目文件中，所要加入程序文件的文件夹，出现了菜单后，选择“Add Files to Group ‘文件夹名’”选项。

文件夹名根据实际  
给出的项目的文件  
夹名字决定




➤选择后会出现如下的对话框，选择需要添加的文件，更改文件类型选项，可以更加快速的找到需要的程序文件。找到文件以后双击该文件，或者选中该文件，点击Add按钮即可将文件添加进项目中，如果多次添加，软件则会报告错误。

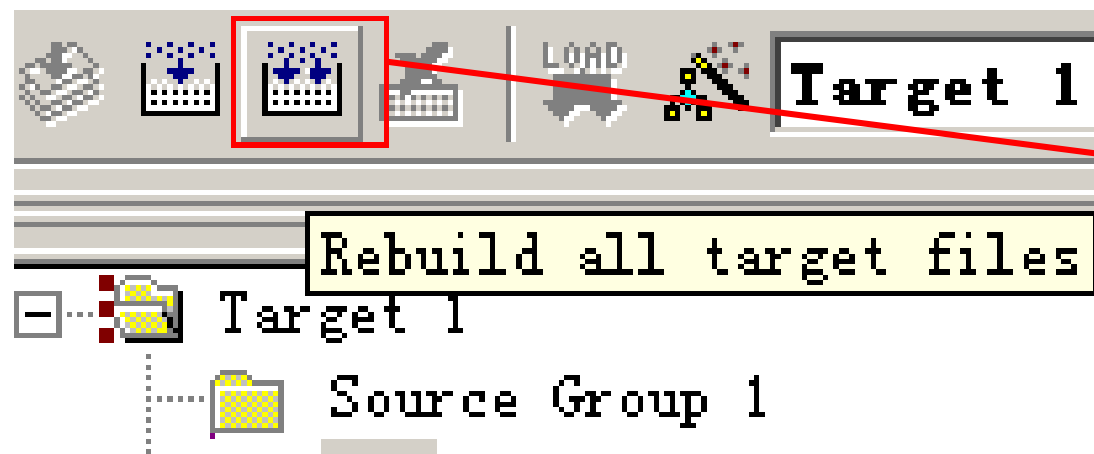


- **完成上面的工作后，就可以在文本框里编写程序，编写完的程序就为当前的项目服务。**
- **编写的程序文件如果添加到别的项目中，也可以为别的项目服务。**
- **程序的编写需要有一定的格式，要将程序对齐书写，每行写一条指令。**

# 编译链接

## ➤ 编译程序

- 将程序写好后就可以进行编译了。编译的目的是为了让程序变成单片机能识别的机器文件，并且可以帮助程序员找出程序中的语法错误。
- 编译时只需要点击快捷图标中的  按钮即可。



该按钮表示编译该项目中的所有程序文件。





程序编译完成后会在输出窗口中显示编译结果，只有当结果出现“0 error(s),0 Warning(s)”才表示程序在语法上没有错误了。


另外，在编译器编译程序时，编译结果只能判断程序中是否有语法错误，而对于逻辑错误，编译器是无法识别的，需要程序员自己检查出现的错误。也就是说编译后没有错误，不代表程序的结果就一定正确。

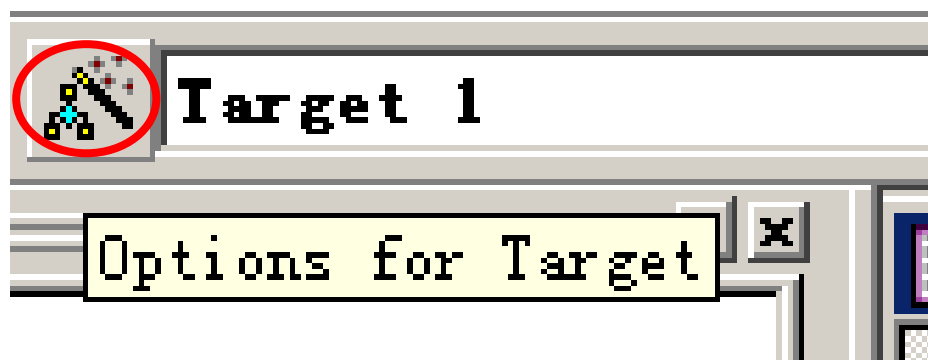
为了帮助程序员检查逻辑错误，该软件还可以进行软件的调试，进入软件的调试后，可以根据自己的需要设置需要查看的寄存器，I/O口和内部RAM的值，以判断程序是否有逻辑错误。

A screenshot of a compiler's output window. The window has a title bar with a close button (X) and a scroll bar. The text inside the window reads: "Build target 'Target 1'", "assembling main.asm...", "linking...", "Program Size: data=8.0 xdata=0 code=258", and "\"ledtest\" - 0 Error(s), 0 Warning(s).". At the bottom of the window, there is a toolbar with buttons for "Build", "Command", and "Find in Files".

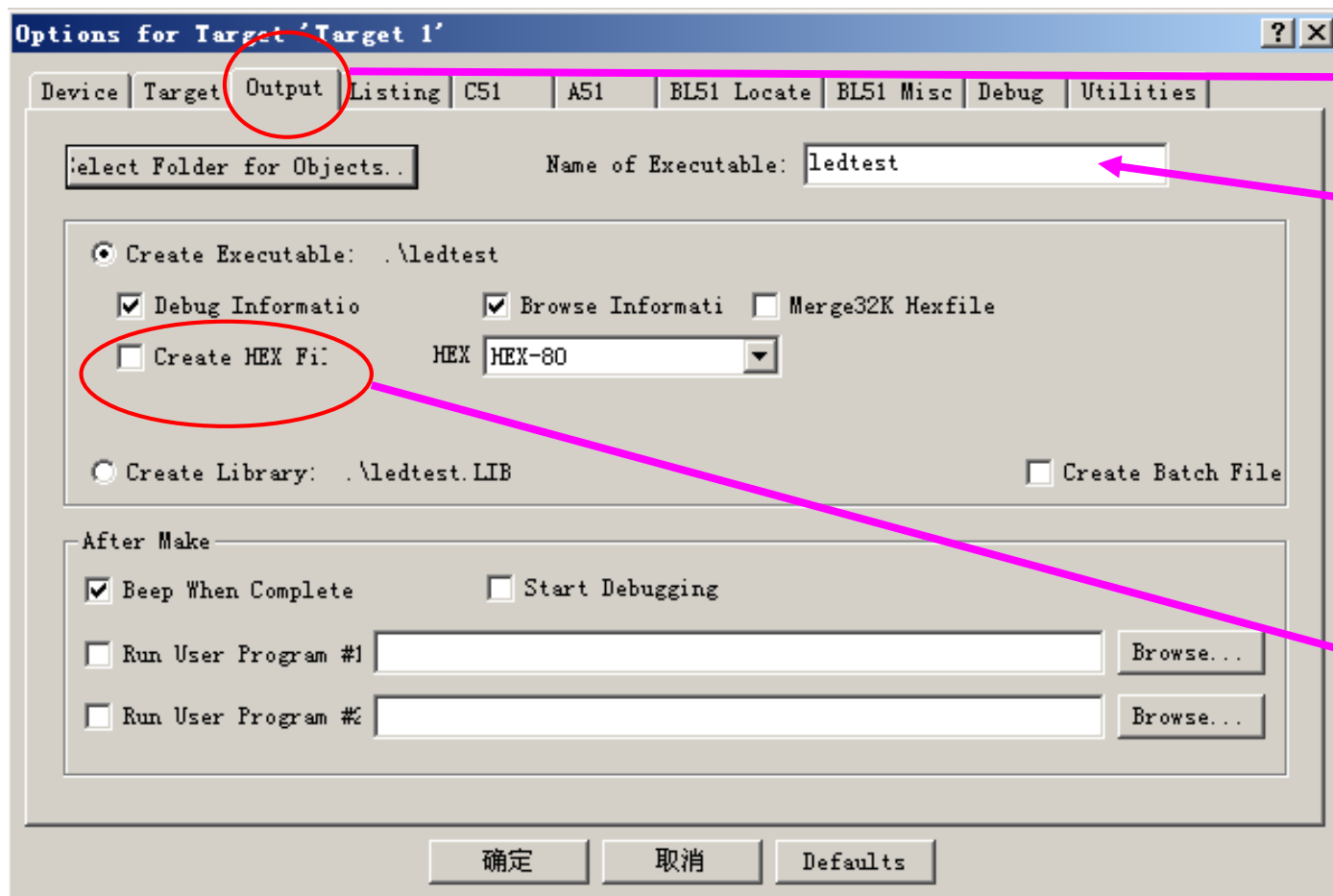
```
Build target 'Target 1'
assembling main.asm...
linking...
Program Size: data=8.0 xdata=0 code=258
"ledtest" - 0 Error(s), 0 Warning(s).
```

## ➤ 机器文件的产生

- 为了让单片机按照编写的程序工作，必须给单片机内写入机器码文件，这个文件也由该软件生成。
- 快捷图标处单击图标  用来设置输出的机器文件。



点击后出现如下的选择对话框，在output标签页中将creat .HEX File选项选中，点击确定即可。

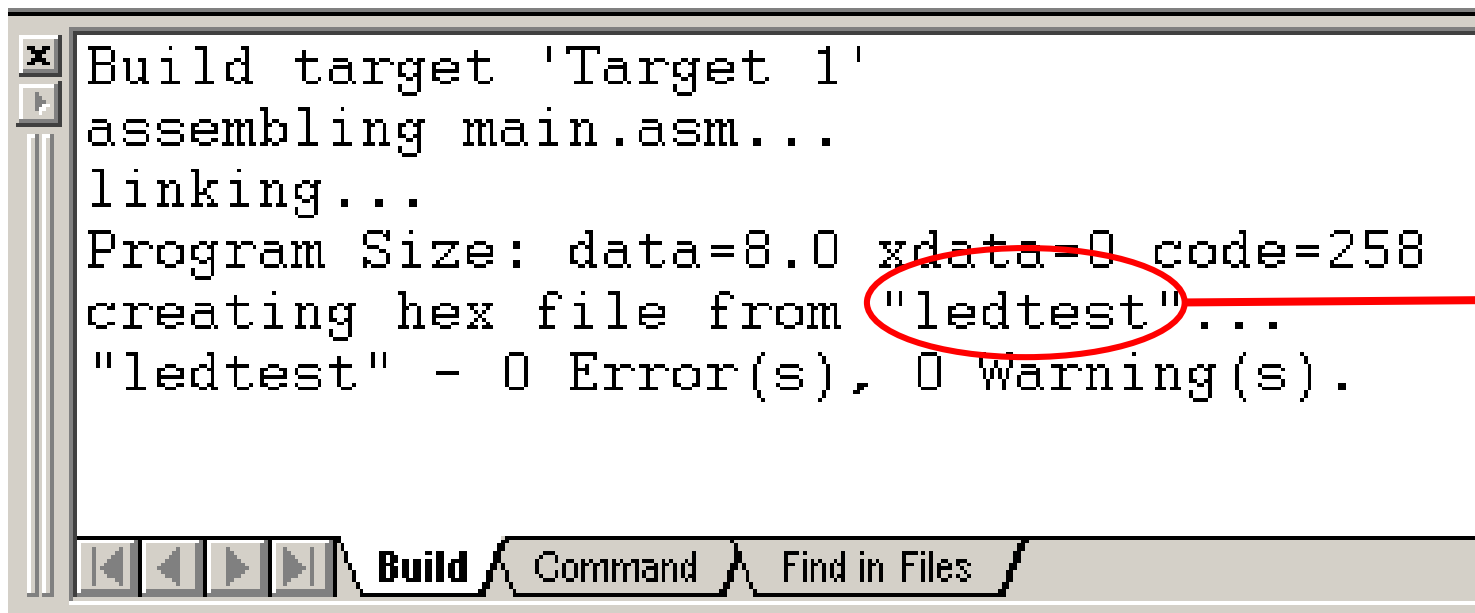


标签页选择

可以在此处修改生成的机器码文件的文件名，文件存储的路径为项目所在文件夹。

在该选项前的空格处打勾，表示选中，这样编译器在编译结束后会自动生成一个机器码的文件，该文件的后缀名为.hex，文件名默认与项目名相同。

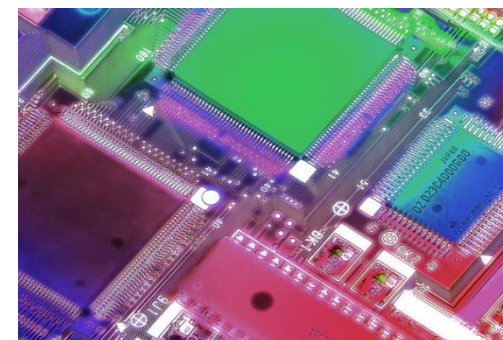
➤ 设置完成后，重新编译一次项目，就可以**得到机器码文件**。



```
Build target 'Target 1'
assembling main.asm...
linking...
Program Size: data=8.0 xdata=0 code=258
creating hex file from "ledtest"...
"ledtest" - 0 Error(s), 0 Warning(s).
```

生成了一个机器码文件，  
文件名为**ledtest.hex**，  
该文件可以在项目所在  
的文件夹中找到。

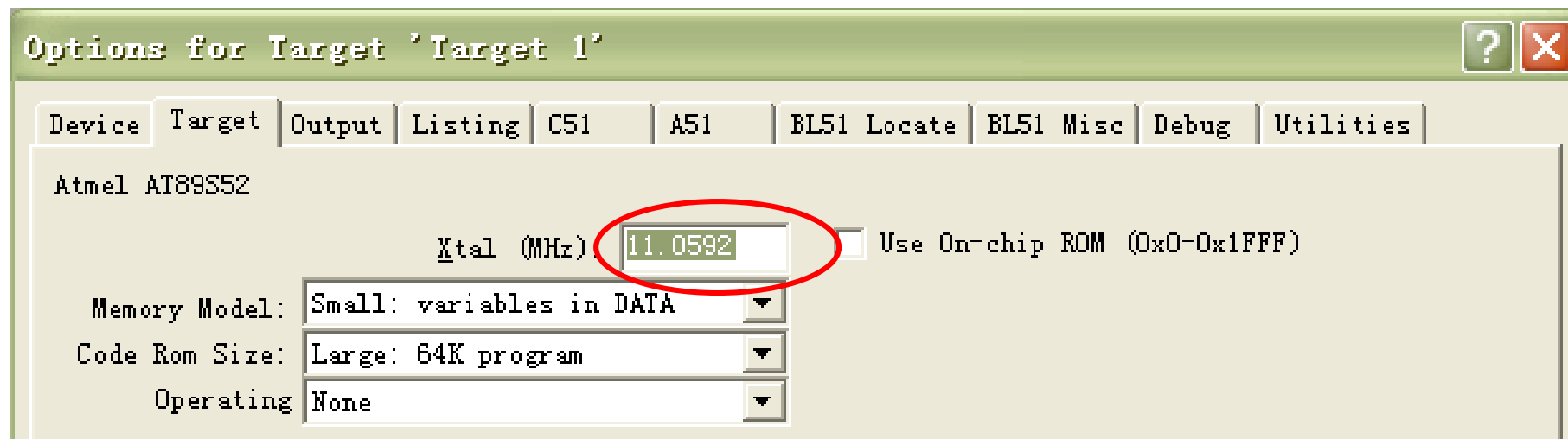
➤ 下载机器码文件到芯片中，就可以让单片机在程序员的程序控制下工作。



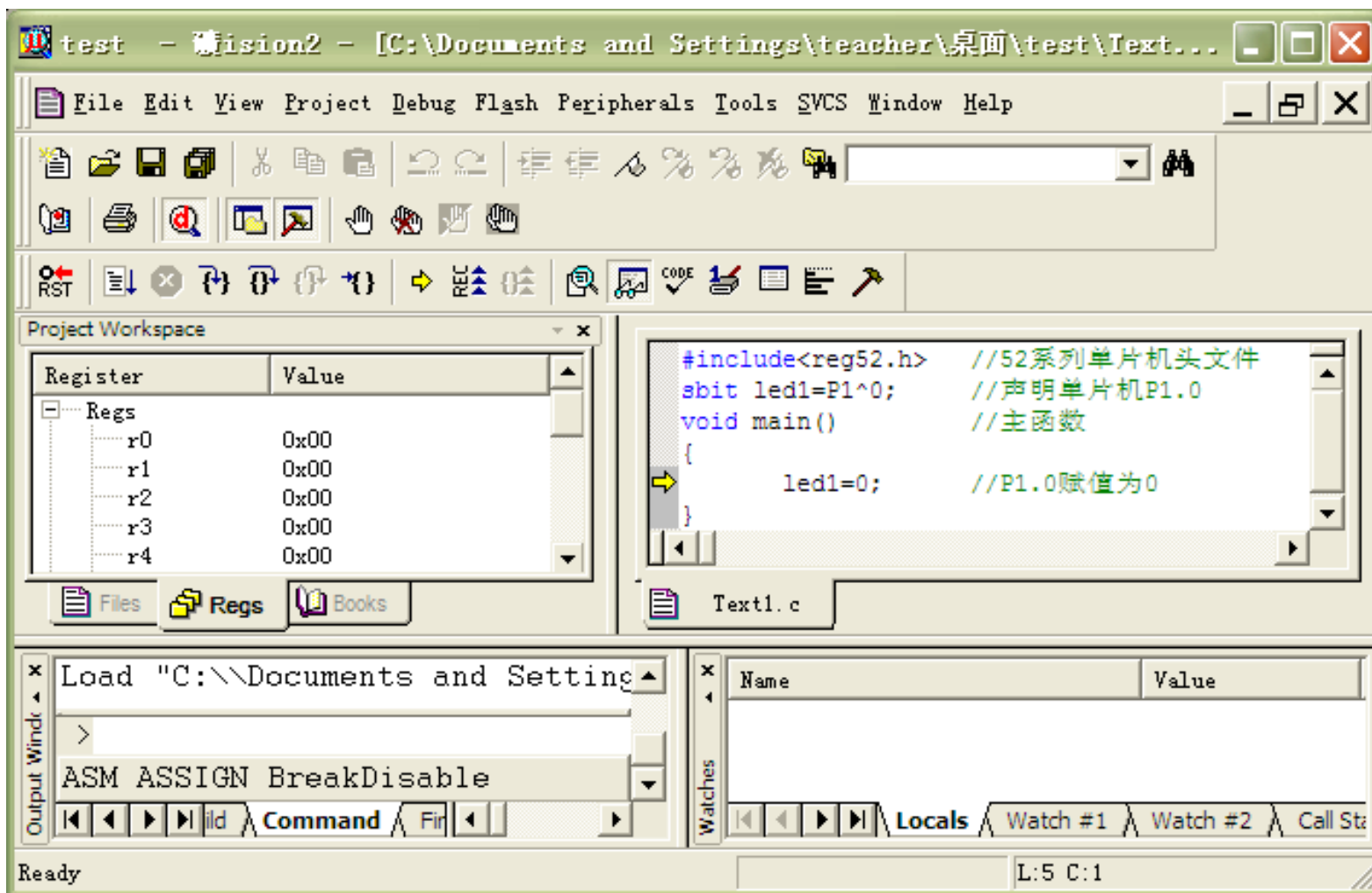
# 调试程序






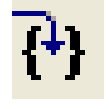
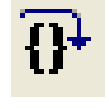





- 首先设置仿真晶振频率，打开工程设置对话框，在“Target”标签下的“Xtal (MHz)”后面将原来的默认值修改为课程设计实验板上晶振频率值11.0592MHz，Keil编译器在编译程序时，计算代码执行时间与该数值有关。



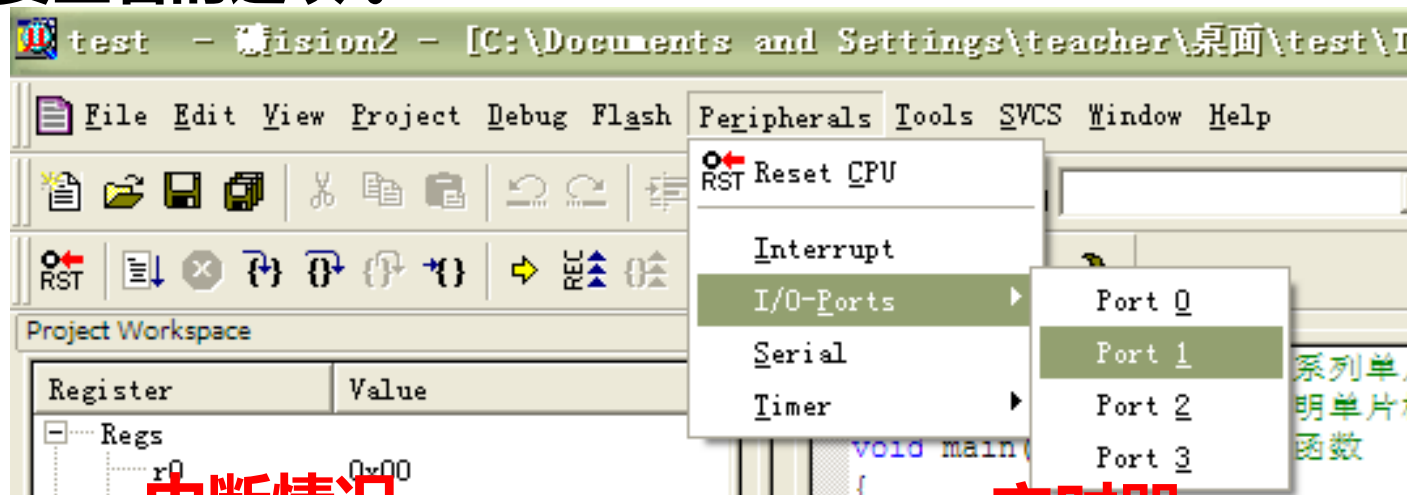
➤ 单击窗口上的调试按钮  快捷图标，进入到软件模拟调试模式



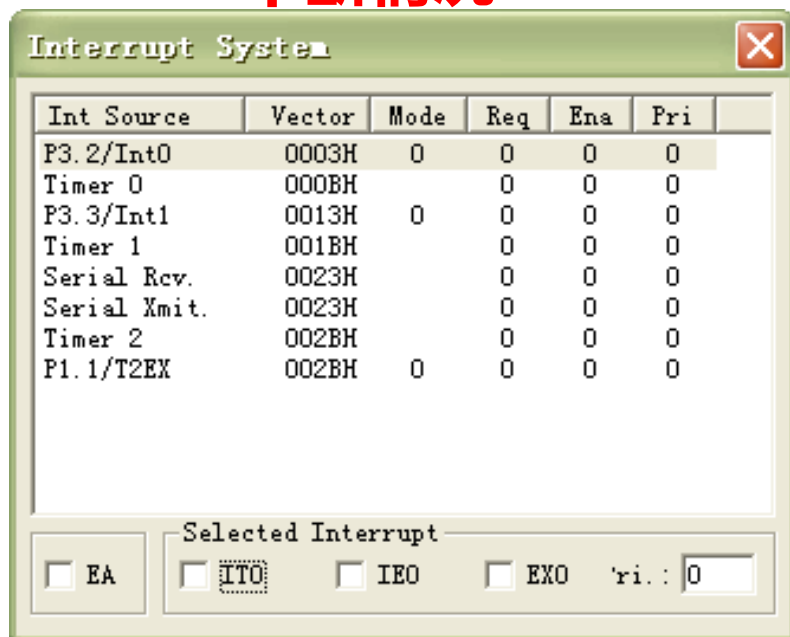


- ——将程序复位到主函数的最开始处，准备重新运行程序。
- ——全速运行，运行程序时中间不停止。
- ——停止全速运行，全速运行程序时激活该按钮，用来停止正全速运行的程序。
- ——进入子函数内部。
- ——单步执行代码，它不会进入子函数内部，可直接跳过函数。
- ——跳出当前进入的函数，只有进入子函数内部该按钮才被激活。
- ——程序直接运行至当前光标所在行。
- ——显示/隐藏编译窗口，可以查看每句C语言编译后所对应的汇编代码。
- ——显示/隐藏变量观察窗口，可以查看各个变量值的变化状态。
- ——设置断点，程序运行到此处停止。

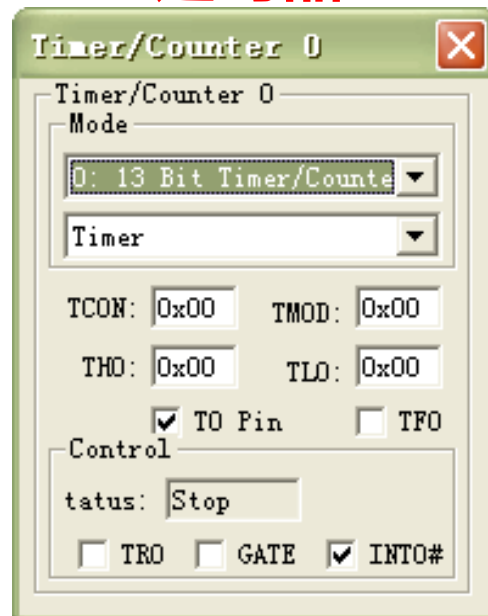
➤ 查看硬件中断、I/O口、串口、定时器情况时，点击菜单“Peripherals”，选中所要查看的选项。



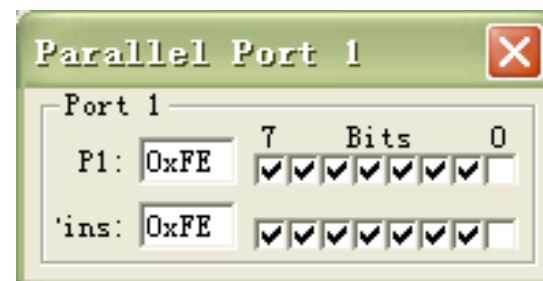
中断情况



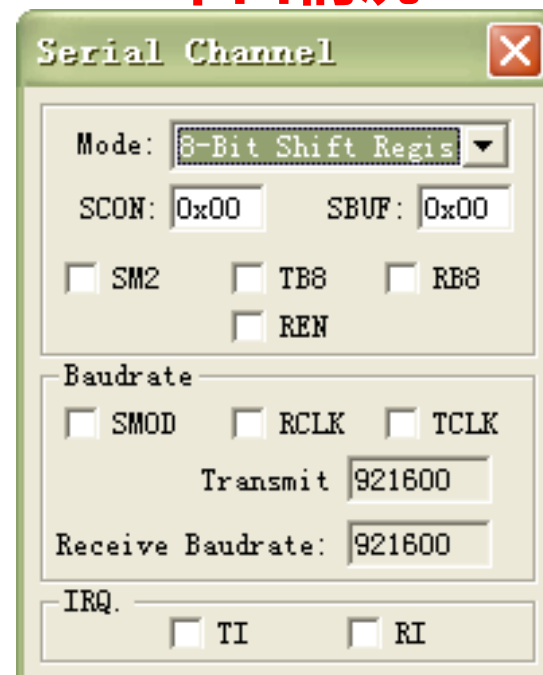
定时器



I/O口

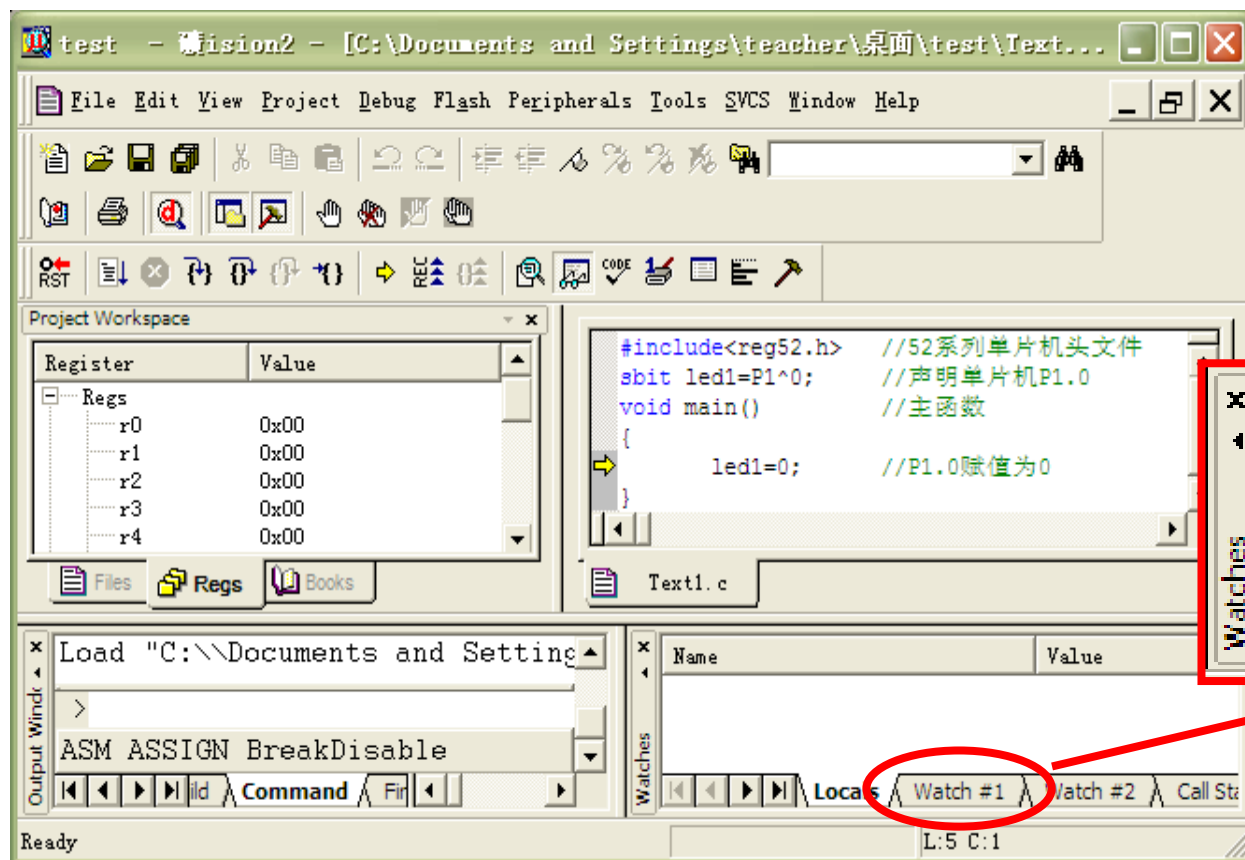


串口情况

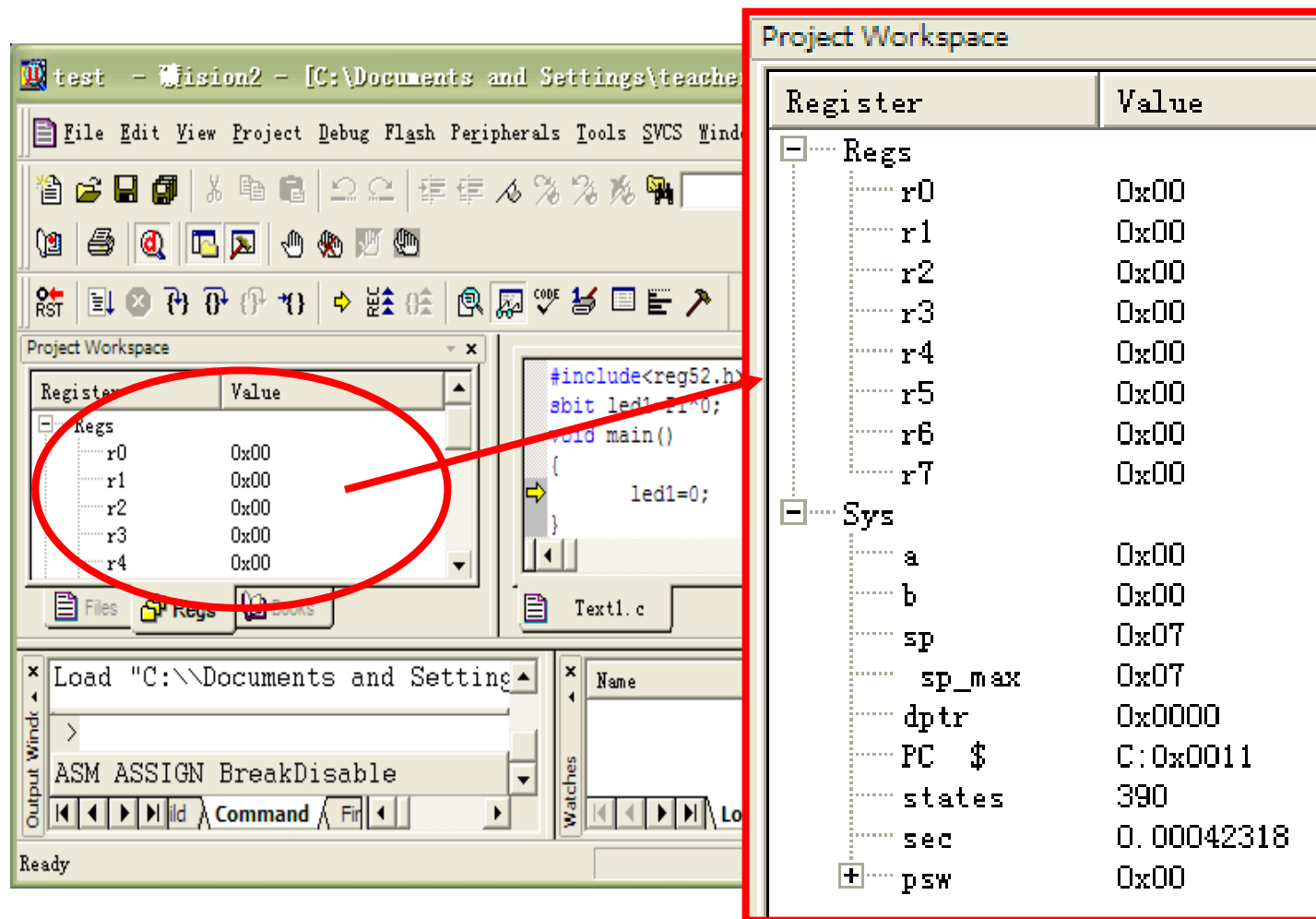




- 单击右下角变量观察窗口的“Watch #1”标签，窗口变成图示，可以在“type F2 to edit”处按F2键，输入程序中的变量，以便观察。



左侧的寄存器窗口中可以看到一些寄存器名称及它们的值



The screenshot shows a debugger interface with a red circle highlighting the Register window on the left and a red rectangle highlighting the Project Workspace window on the right. A red arrow points from the Register window to the Project Workspace window.

**Register Window (Left):**

Register	Value
r0	0x00
r1	0x00
r2	0x00
r3	0x00
r4	0x00

**Project Workspace Window (Right):**

Register	Value
----- Regs	
r0	0x00
r1	0x00
r2	0x00
r3	0x00
r4	0x00
r5	0x00
r6	0x00
r7	0x00
----- Sys	
a	0x00
b	0x00
sp	0x07
sp_max	0x07
dp_ptr	0x0000
PC \$	C:0x0011
states	390
sec	0.00042318
psw	0x00

The Project Workspace window also displays the following code snippet:

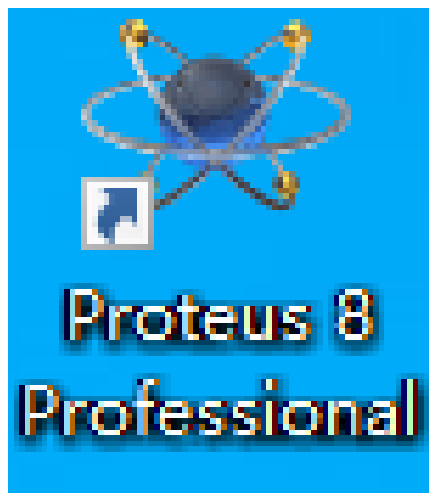
```
#include<reg52.h>
sbit led1=P1^0;
void main()
{
    led1=0;
}
```



# Proteus使用

# 主要步骤

## 1、双击图标



打开Proteus软件

## 2、新建工程



UNTITLED - Proteus 8 Professional - 主页

文件(F) 系统(Y) 帮助(H)

主页 X

# PROTEUS DESIGN SUITE

## 第1步

### 使用教程

- Schematic and PCB (Basic)
- Schematic and PCB (Advanced)
- 仿真
- What's New

### 帮助中心

- 帮助主页
- 原理图绘制
- PCB板设计
- 仿真

### 关于软件

(C) Labcenter Electronics 1989-2021  
发布版本 8.12 SP0 (内部版本30713) 具有高级仿真功能  
[www.labcenter.com](http://www.labcenter.com)

注册给:

用户编号: 01-17378-370  
更新许可期限: 01/04/2025

Free Memory: 2,998 MB  
Windows 10 (x64) v10.00, Build 19042

### 开始设计

打开工程 新建工程 New Flowchart 打开示例工程

### 最近打开的工程

- D:\Proteusproject\1.pdsprj
- D:\Proteusproject\新工程.pdsprj

### 最新消息

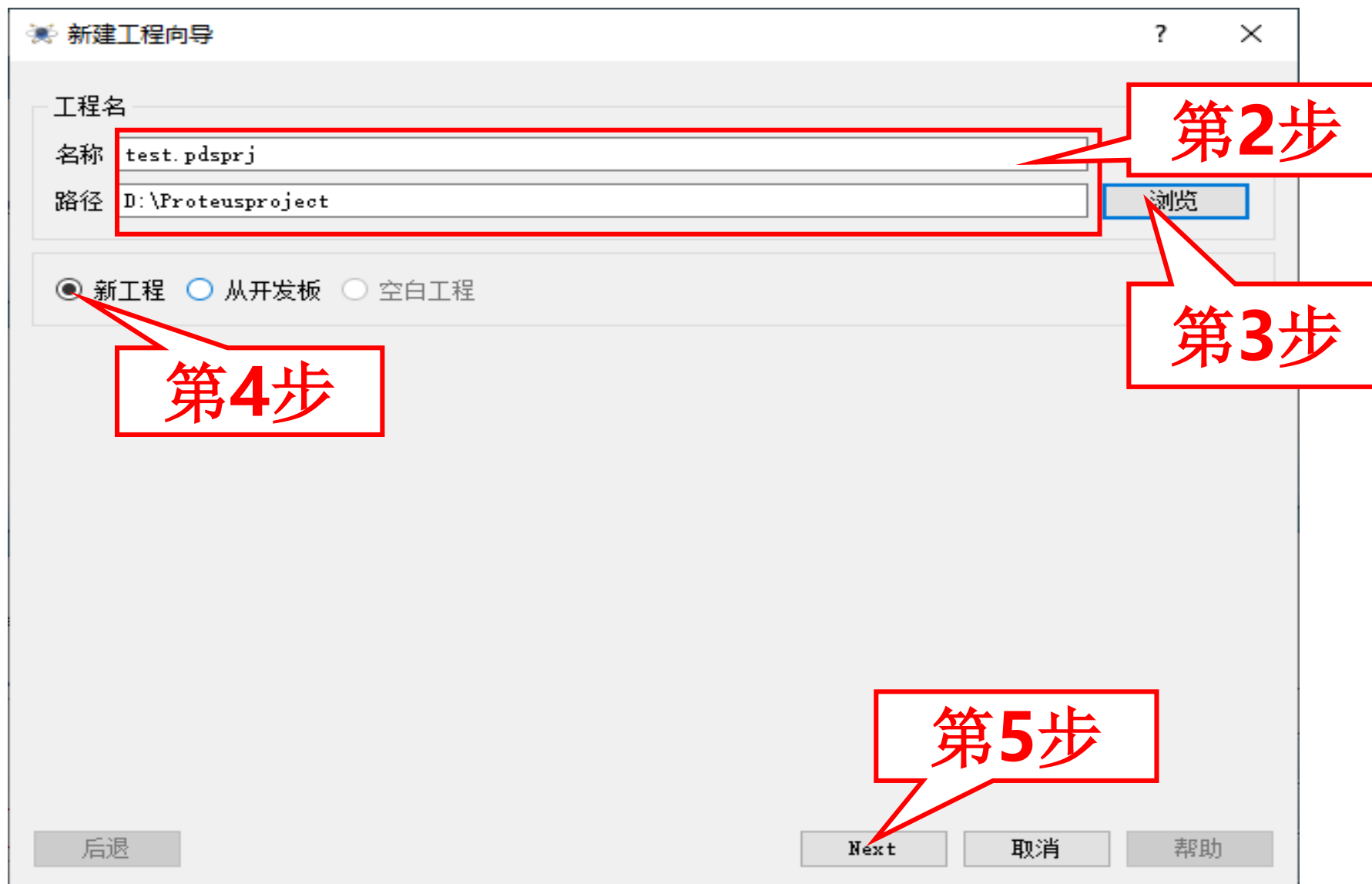
#### Proteus Design Suite Professional

##### 有新版本可升级

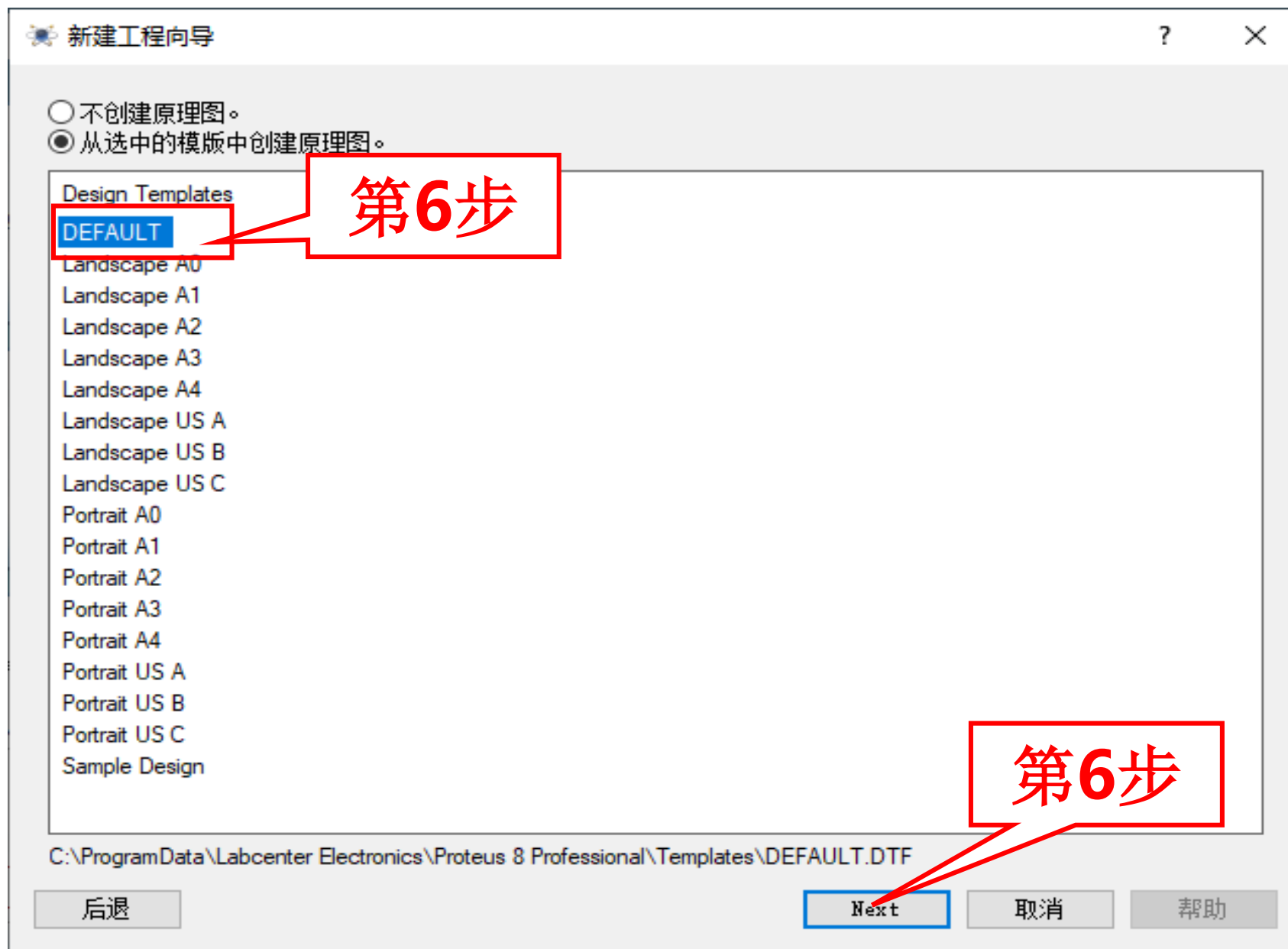
描述	发布日期	USC有效	
<a href="#">Proteus Professional 8.14 SP1 [8.14.32850]</a>	28/04/2022	是	<a href="#">下载</a>
<a href="#">Proteus Professional 8.13 SP1 [8.13.32171]</a>	07/01/2022	是	<a href="#">下载</a>
<a href="#">Proteus Professional 8.12 SP2 [8.12.31155]</a>	17/06/2021	是	<a href="#">下载</a>

☐ 忽略测试版升级

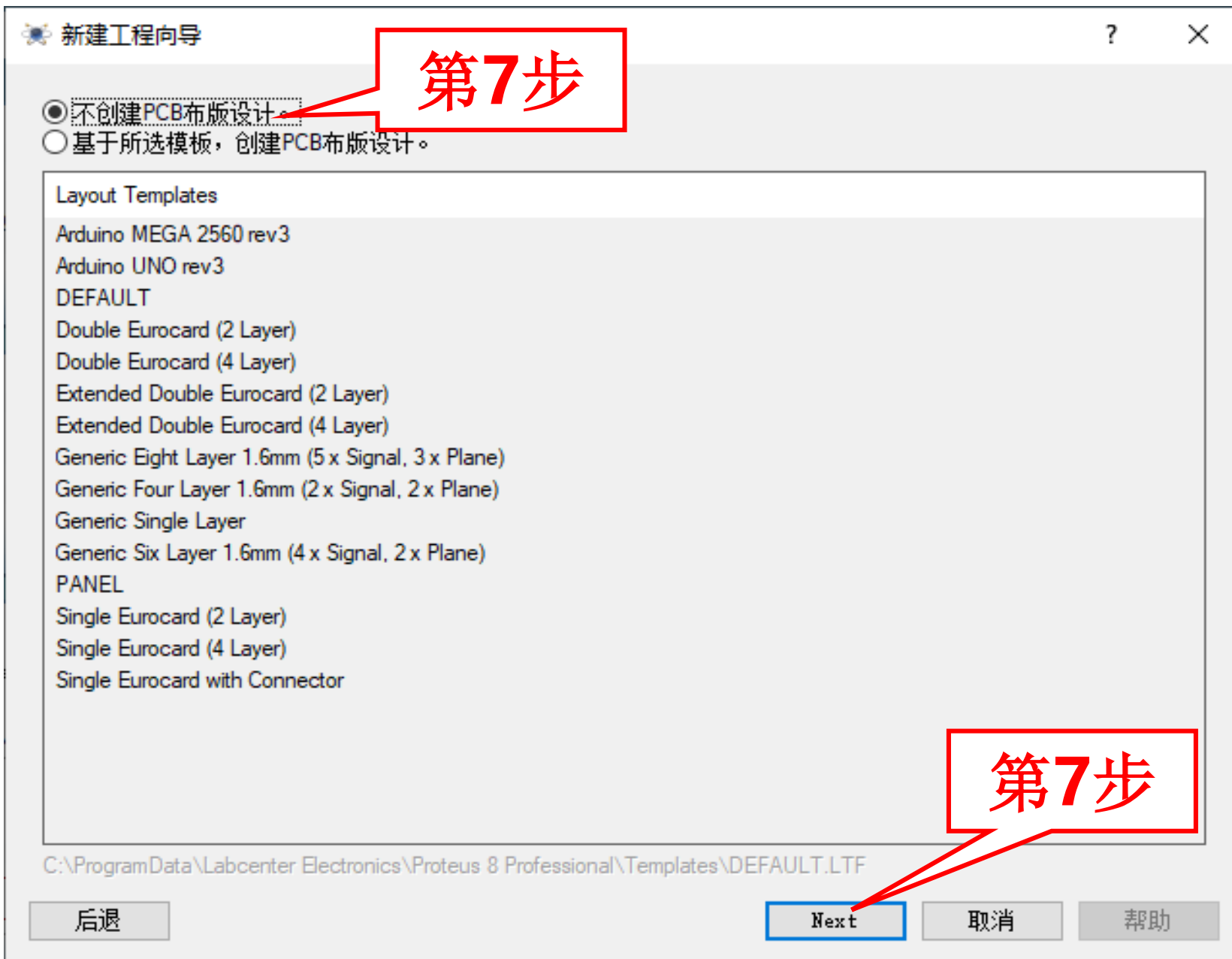
3、输入工程名称，点击浏览选择需要保存的位置（一般新建一个专用文件夹存放课程设计的仿真工程文件）



## 4、选择默认原理图，再下一步



## 5、不创建PCB





## 6、创建固件项目（务必提前安装好Keil软件，在编译器中点击“检查全部”后按图选择）



新建工程向导

☐ 没有固件项目

☒ 创建固件项目 **第8步**

☐ Create Flowchart Project

系列: 8051

Controller: AT89C52 **第9步**

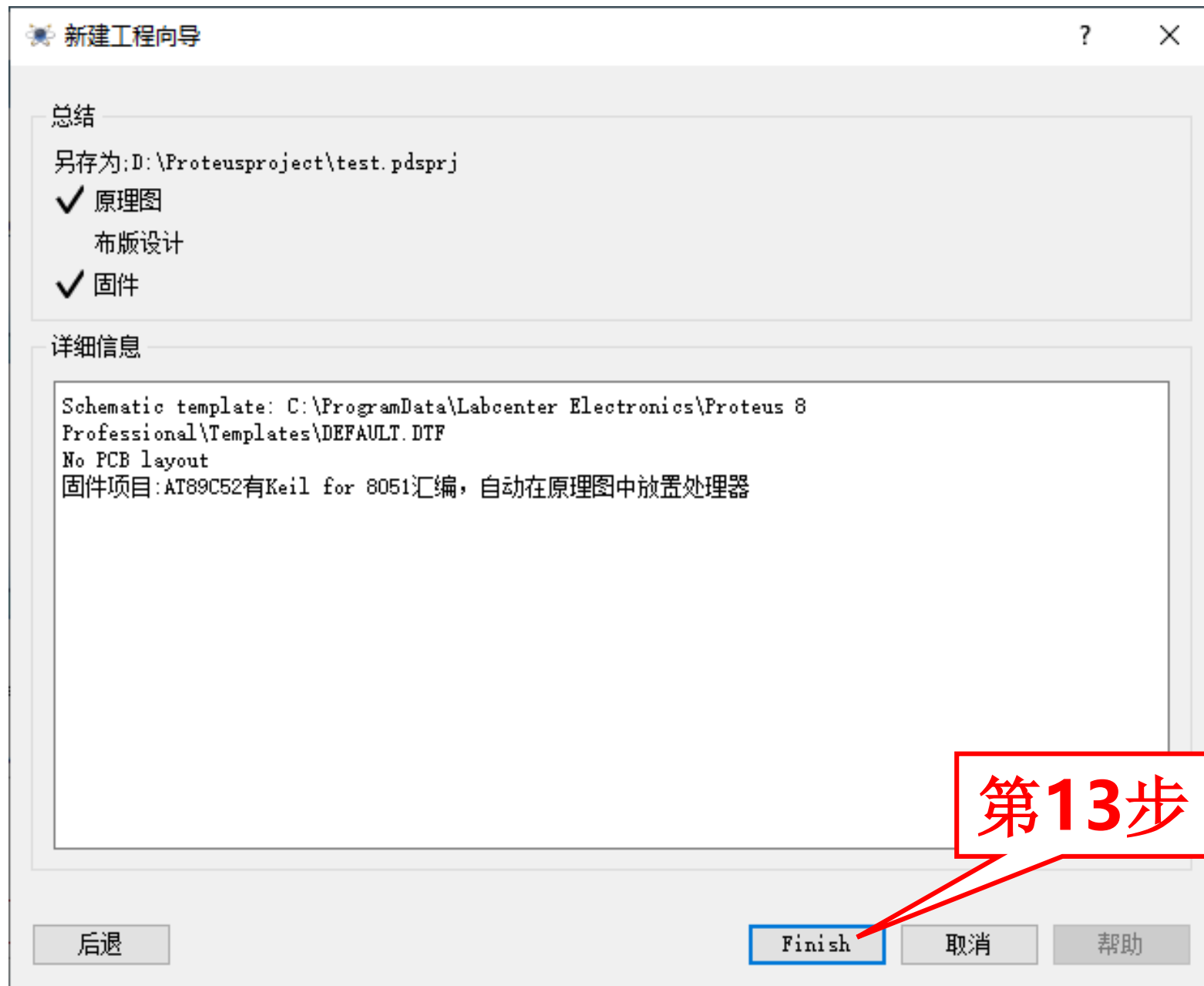
编译器: Keil for 8051 **第10步**

创建快速启动文件 ☒ **第11步**

Create Peripherals ☐

后退 Next 取消 帮助 **第12步**

## 7、完成项目创建

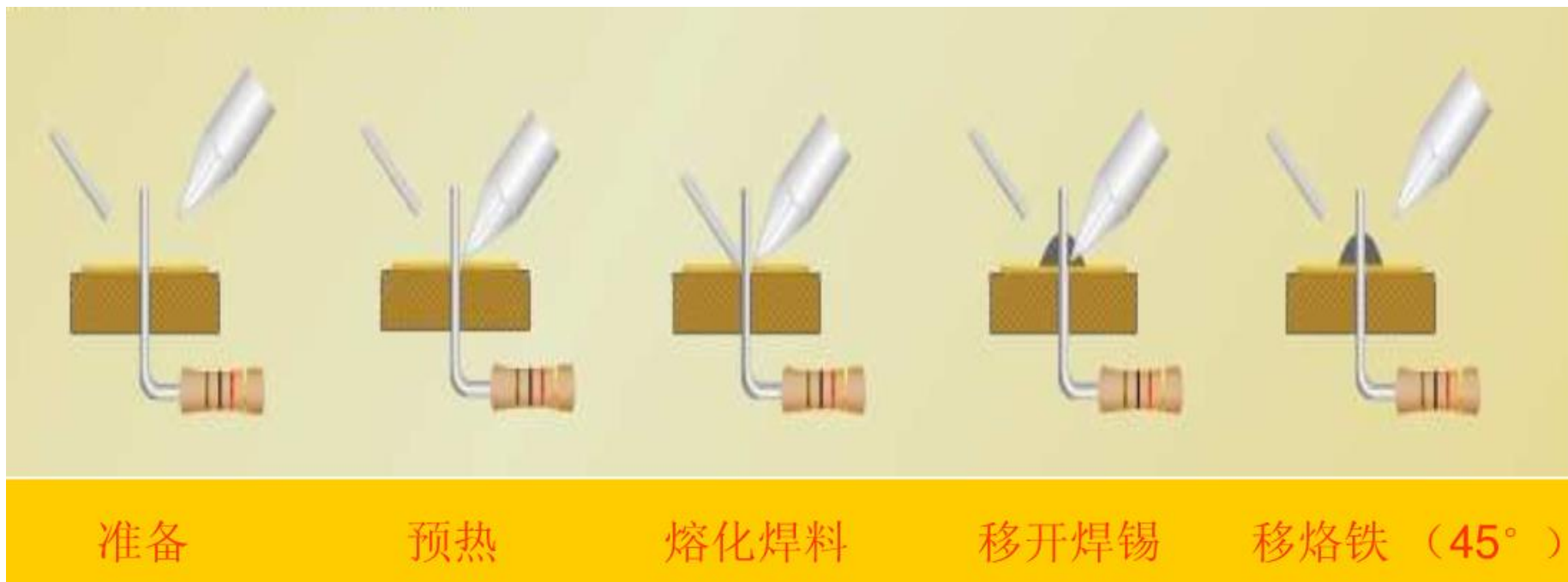


第13步

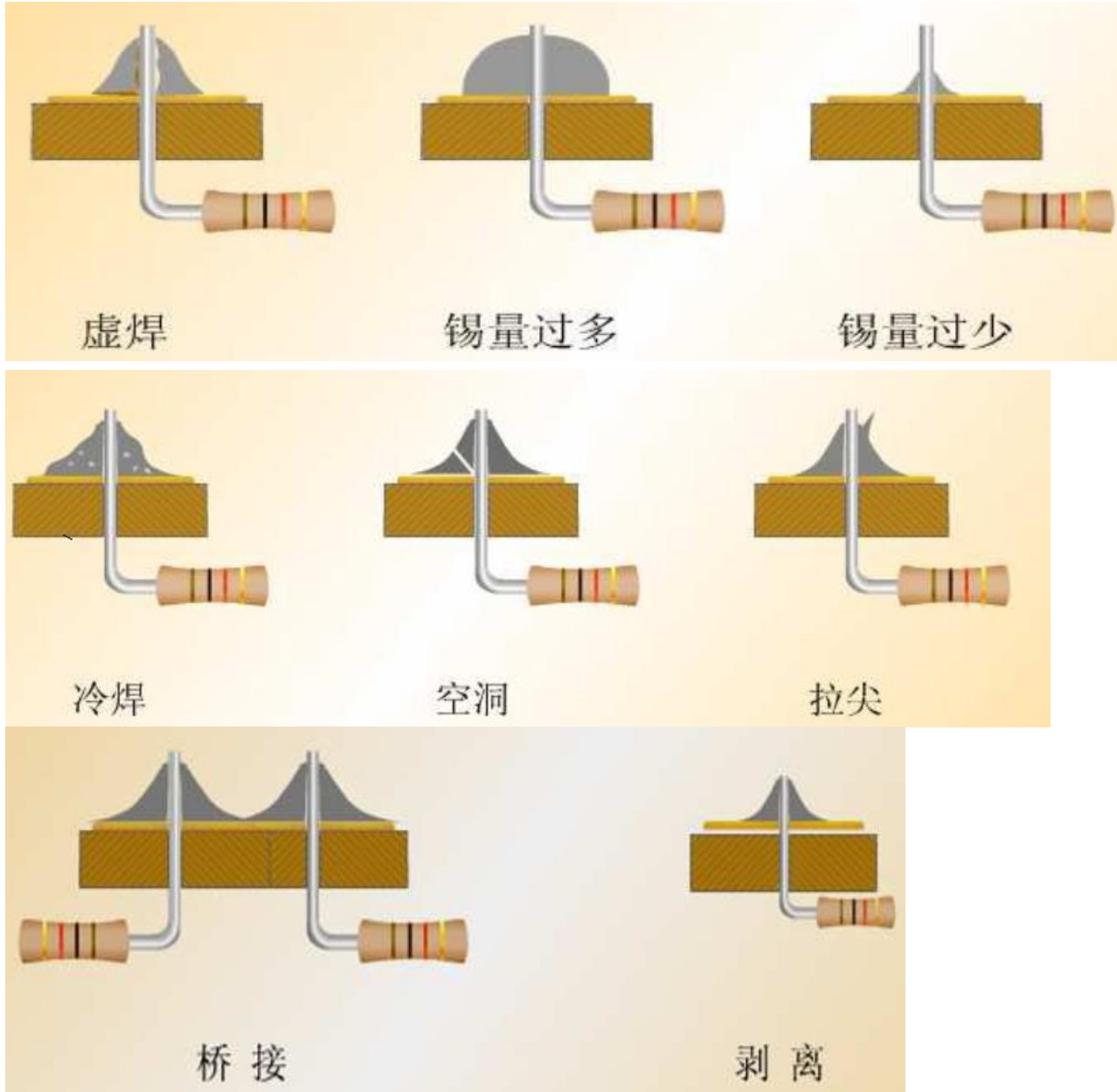


# 焊接注意事项

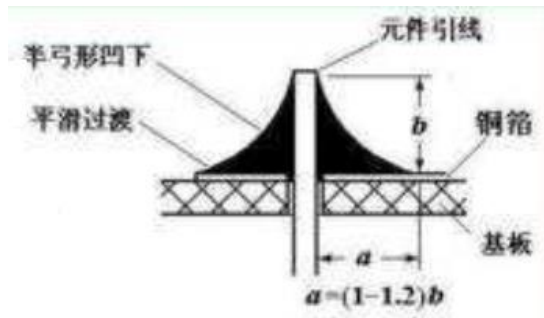
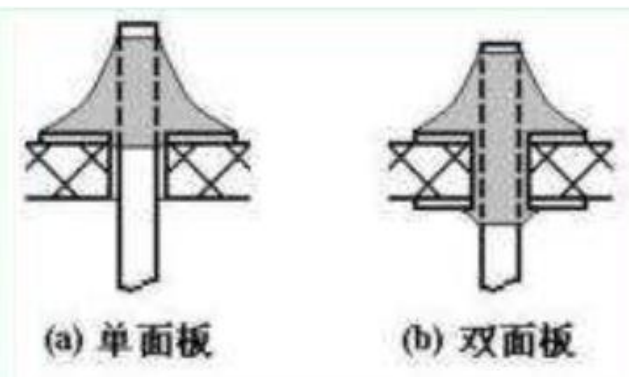
# 加热焊接（5步法）



# 常见焊点缺陷

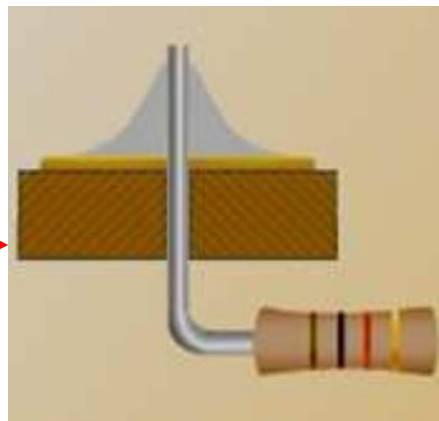


# 单个焊点的外观要求

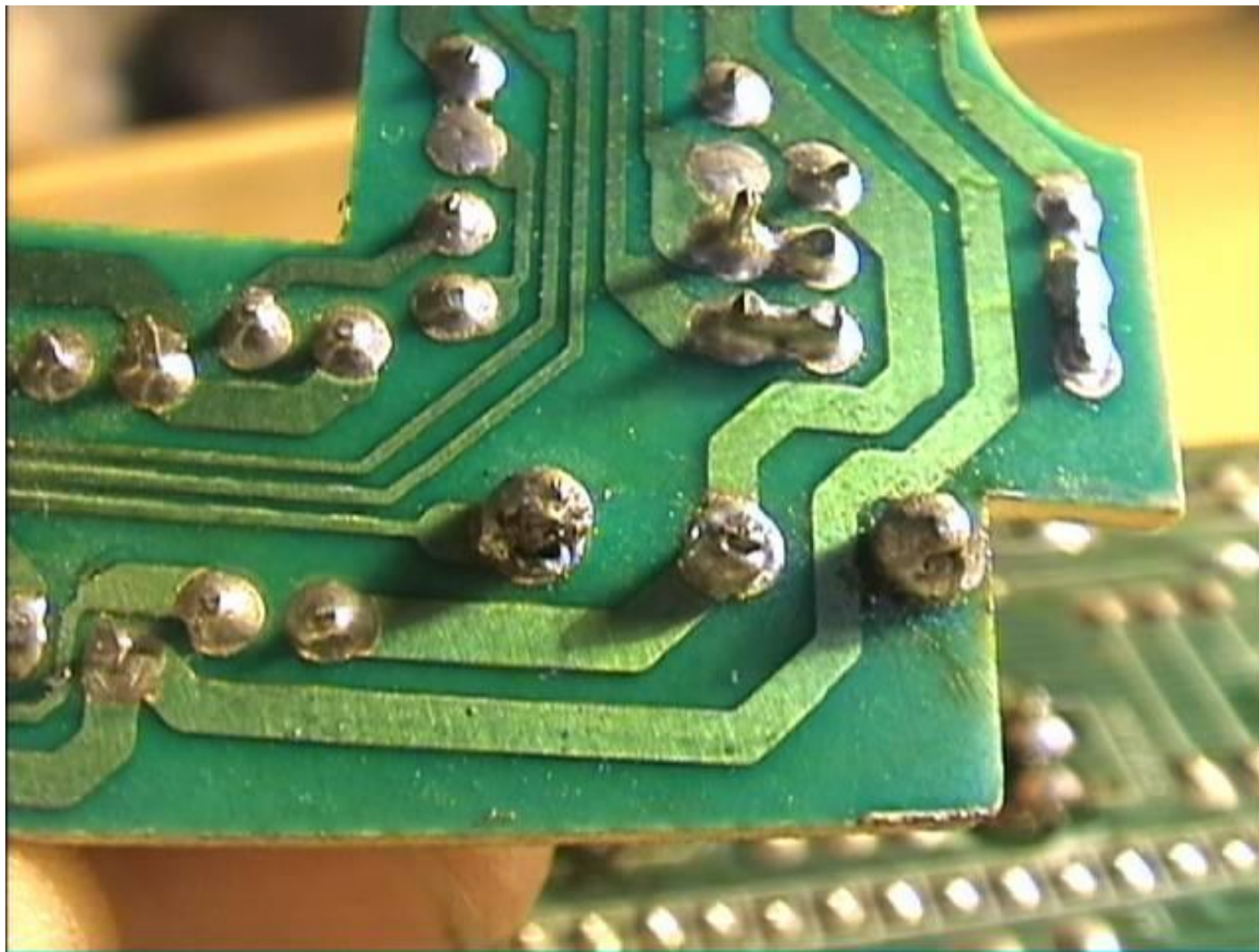


1. 形状为近视圆锥而表面微凹呈漫坡状，（以焊接导线为中心，对称成裙形拉开）。虚焊点表面往往成凸状，可以鉴别出来。
2. 焊料的连接面呈半弓形凹面，焊料与焊件交界处平滑，接触角尽可能小。
3. 表面有光泽且平滑。
4. 无裂纹、针孔、夹渣。

合格的焊点



# 举例





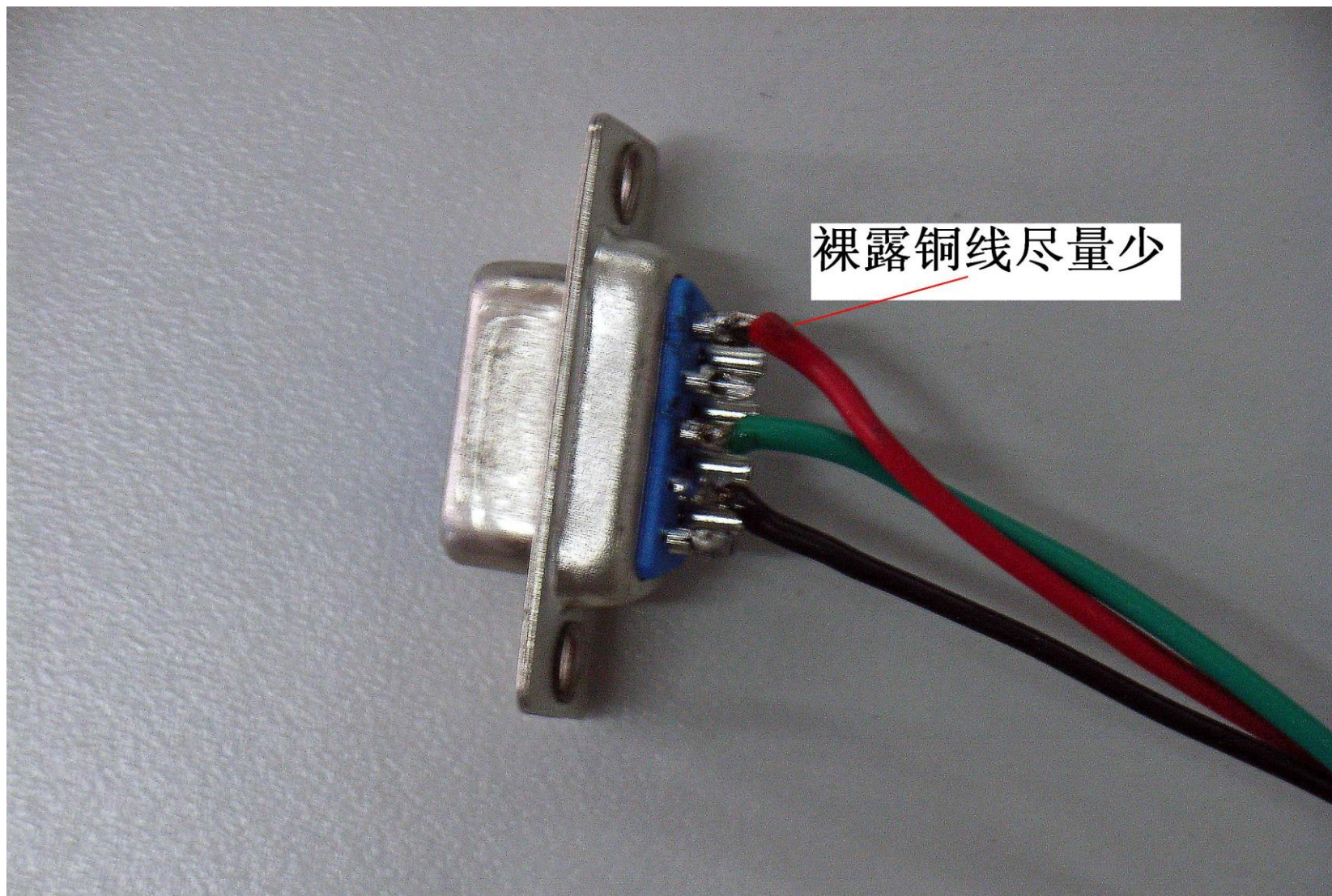
# 焊接时的注意事项

- 注意电烙铁的安全使用和科学使用
- 焊接时不可施加压力
- 注意区分元器件的极性
- 尽量避免重复焊接

**注意：请爱护所使用的烙铁**

- 有段时间不使用烙铁，拔掉电源。
- 隔段时间给烙铁头镀锡一次，防止氧化。







# 调试时的注意事项

- 搬运过程中注意是否断线及短路
- 先做通断检查，重点电源和地必须正确。
- 连接232口必须断电
- 拔插元件和连接导线必须断电
- 实验中小心短路，线路板上下不要有金属物，不要将线路板放在计算机外壳上。