

Tiva LaunchPad 快速入门

V1.0

目录

1	Tiva LaunchPad 简介	3
1.1	电源	4
1.2	复位电路	5
1.3	时钟电路	5
1.4	用户按键	6
1.5	三色 LED 灯显示	6
1.6	虚拟串口	7
1.7	调试接口	8
1.8	USB 接口电路	8
2	Tiva 的开发环境搭建	9
2.1	CCS 的下载与安装	10
2.2	Stellaris ICDI 驱动安装	10
2.3	LM Flash Programmer 安装与使用	13
2.4	TivaWare 安装与使用	14
2.4.1	PC 机上安装 TivaWare	14
2.4.2	CCS5.4 上使用 TivaWare	15
2.4.3	CCS6.0 上使用 TivaWare (待补充)	16
3	Tiva 板卡入门实验	17
3.1	下载 Bin 文件	17
3.1.1	找到编译好的 BIN 文件	17
3.1.2	程序下载演示	18

3.2	CCS5.4 中导入例程, 编译下载.....	20
3.2.1	导入 TivaWare 例程	20
3.2.2	编译工程.....	21
3.3	CCS5.4 新建空白工程.....	24
3.3.1	新建一个空白工程.....	24
3.3.2	基本文件路径设置.....	25
3.3.3	示例——点亮 LED 灯.....	26
3.4	导入他人的工程.....	28
4	Tiva 参考资料.....	29
5	常见问题解决	30
5.1	找不到 Tivaware	30
5.2	TivaWare 的函数定义.....	30
5.3	编译时提示找不到头文件	30
5.4	找不到变量	31
5.5	错误: unresolved symbol	31
6	附录	32

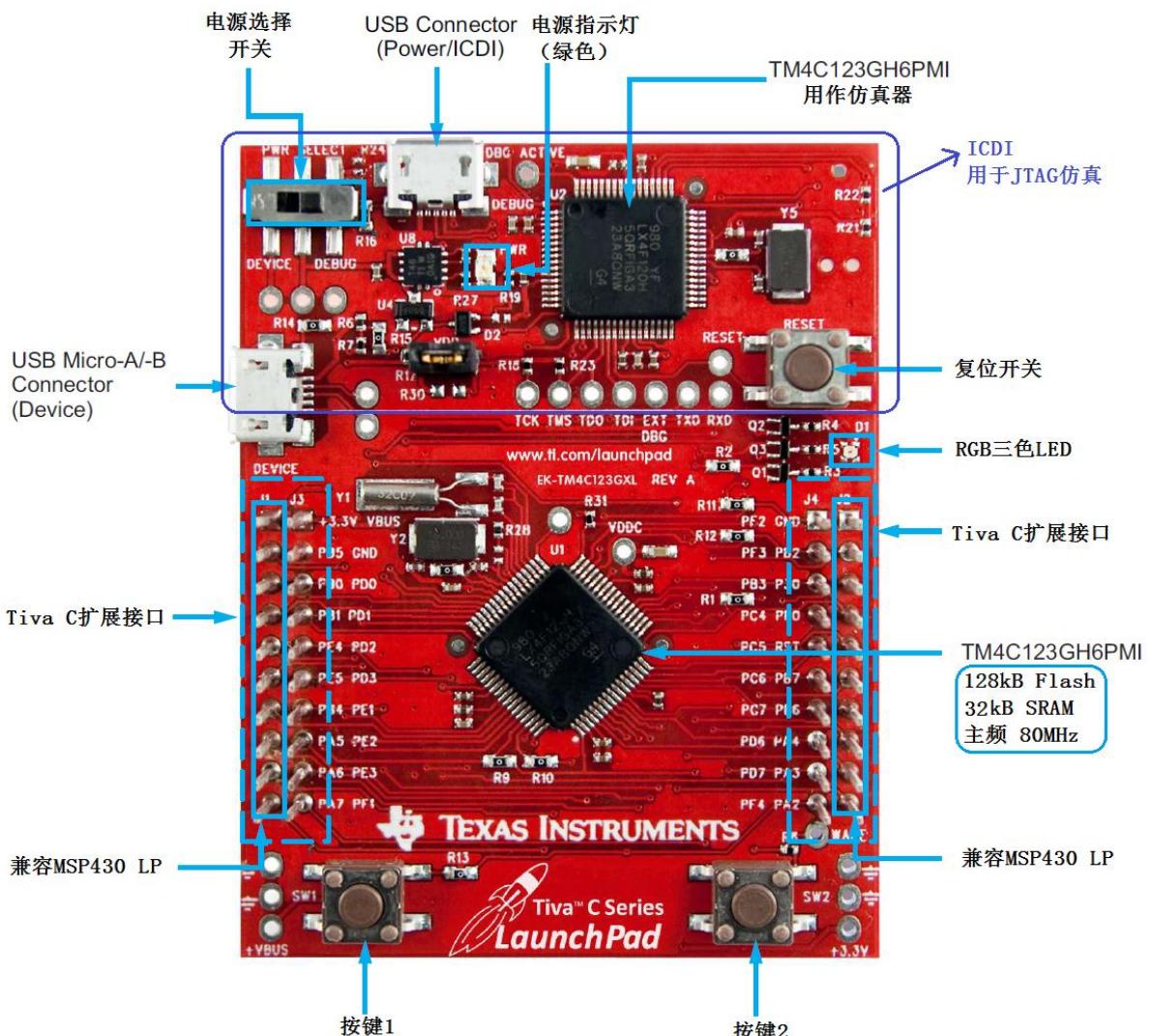
1 Tiva LaunchPad 简介

TM4C123GH6PM 是 TI 公司推出的一款 32 位基于 ARM Cortex-M4 的处理器，主频 80MHz，256kB Flash，32kB SRAM，具有 USB Host，Device 和 OTG 的能力。

另：TM4C129 系列带有 LCD、以太网控制器等，主频可到 120M，更多信息[参考 TI 官网](#)。

TivaC LaunchPad 是基于 TM4C123GH6PM 控制器的实验板卡，自带仿真器，连接上 USB 即可进行 Corte-M4 的学习。

板子上带 2 个用户按键和 1 个三色的 LED 灯；对外引出的 IO 口符合 BoosterPack 40Pin 标准定义（详见 [TI 的文档](#)），也就是说可以兼容 MSP430 LaunchPad 板卡，整个板卡示意图如下：



拿到 TivaC LAUNCHPAD 板子时，不用急于上电操作，“磨刀不误砍柴工”，先花点时间大致了解下板卡的硬件电路连接。使用前，只要了解大概即可😊。

1.1 电源

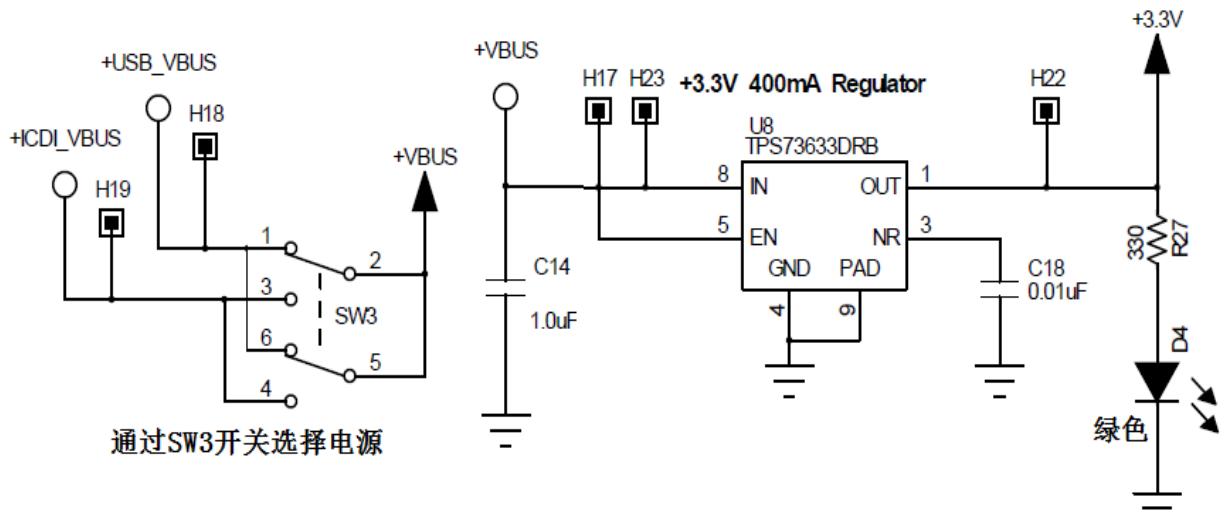
板子通过 USB 口供电。有两个 USB 口（Device 和 Debug；USB 座子边上有丝印字符表示），这两个 USB 口都可以给板子供电，用开关来选择。

板子上有一个绿色的 LED 灯，用来指示 3.3V 供电。



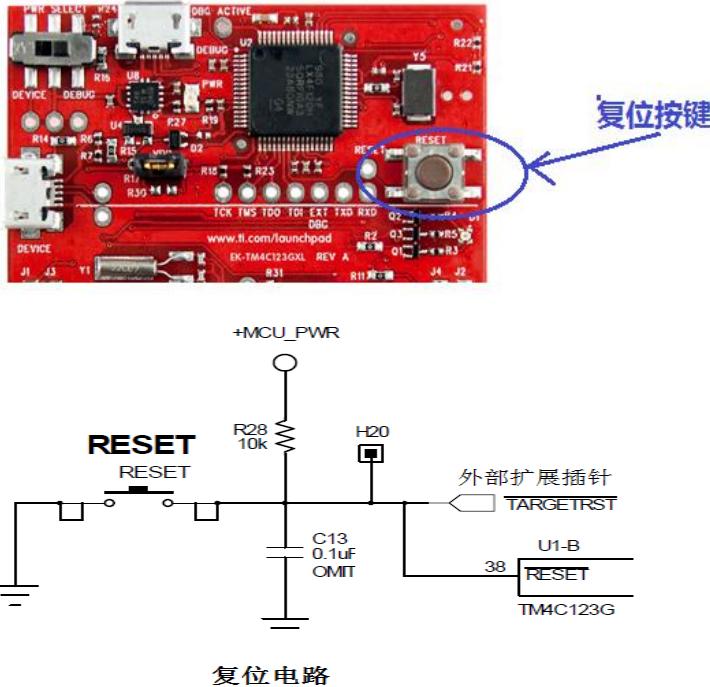
对外输出两路电压：

- **3.3VDC 最大 300mA**;
- **5.0VDC 输出能力 23mA~323mA**, 与 **3.3VDC** 的使用有关。**3.3V** 用的多了，供这边使用的就少了。



1.2 复位电路

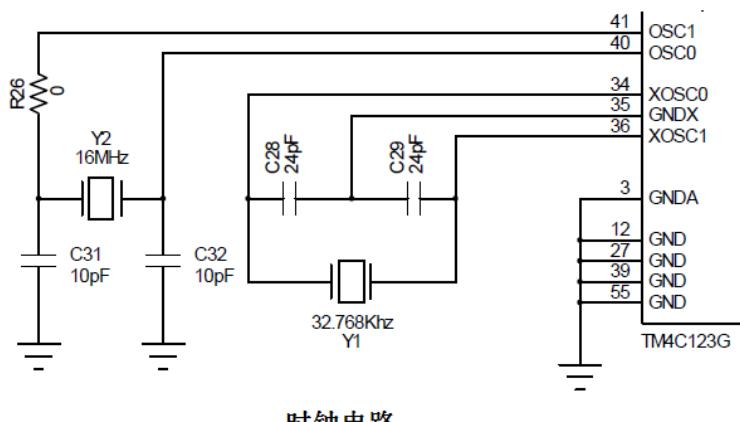
低电平有效复位。当按键按下时，为低电平。



1.3 时钟电路

板卡上有两个晶振。16MHz 晶振（Y2）提供给处理器，通过内部 PLL，倍频后再分频给内核和外设使用；

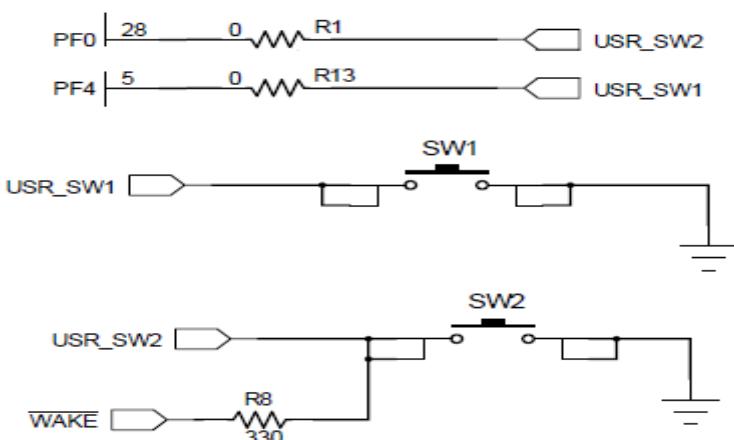
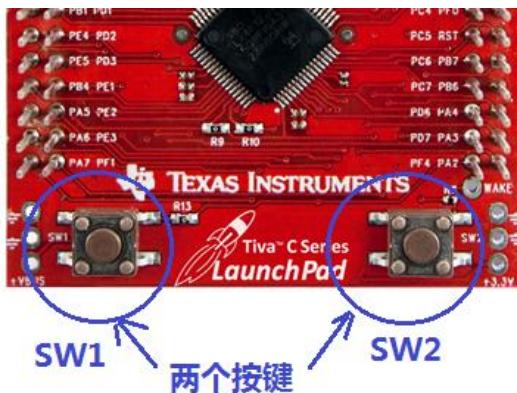
晶振 32.768KHz（Y1）用于休眠系统的时钟源；休眠系统也是一大特色，后续有时间再补充哈。



1.4 用户按键

按键按下时，IO 口接地，为低电平；

- GPIO-PF4---->按键 SW1
- GPIO-PF0---->按键 SW2



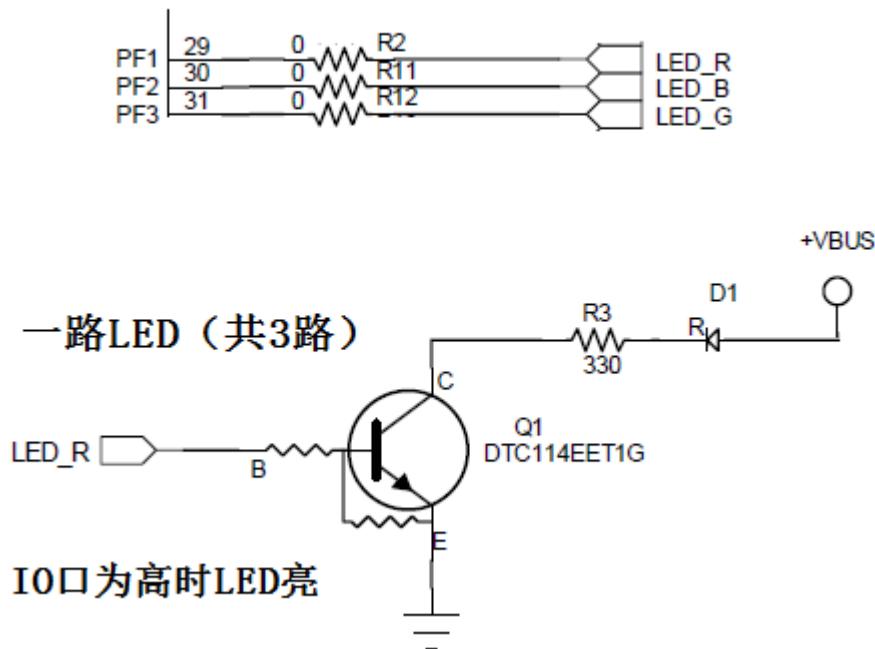
1.5 三色 LED 灯显示

LaunchPad 板子上带一个三色的 LED 灯，位于复位按键下边。



IO 口输出高电平时，三极管导通，LED 灯亮。

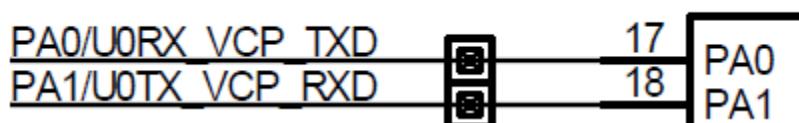
- GPIO-PF1---->RGB LED (红色)
- GPIO-PF2---->RGB LED (蓝色)
- GPIO-PF3---->RGB LED (绿色)



1.6 虚拟串口

当 LaunchPad 板卡连接到电脑时，作为调试器的同时，也会有一个虚拟的串口可以使用。处理器上使用的串口引脚为：

- PA0 ---->U0RX
- PA1 ---->U0TX



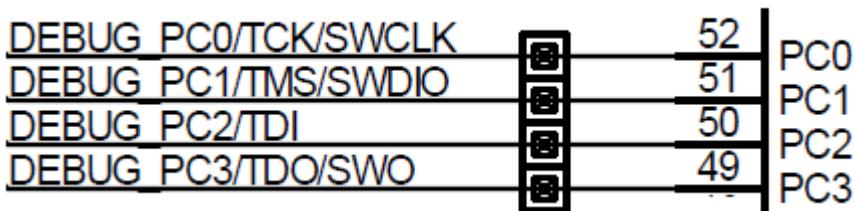
连接到仿真器的 TM4C 芯片上

1.7 调试接口

LaunchPad 板自带一个板上仿真接口 ICDI (In-Circuit Debug Interface)，通过 USB 先连接电脑后即可使用。ICDI 可用于编程或者调试 TM4C123GH6PM。支持 LM Flash Programmer 或者兼容的工具。

其它的调试接口可以连接到 SWD (Serial Wire Debug) 和 SWO。注意，ICDI 仅支持 JTAG 调试。

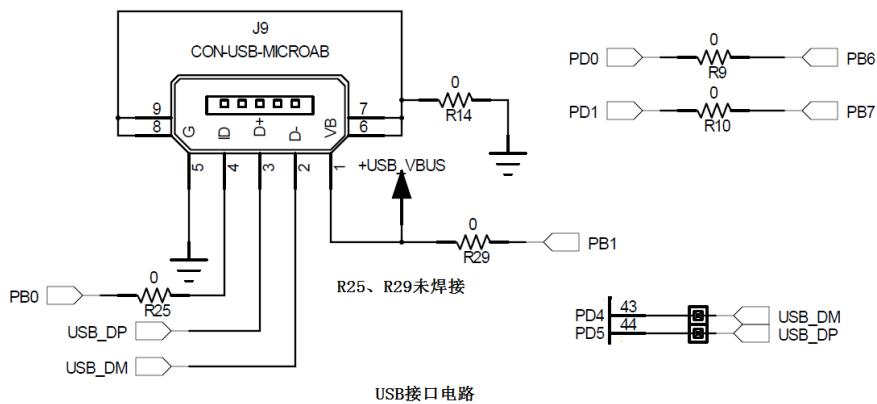
- PC0---->TCK/SWCLK
- PC1---->TMS/SWDIO
- PC2---->TDI
- PC3---->TDO/SWO



1.8 USB 接口电路

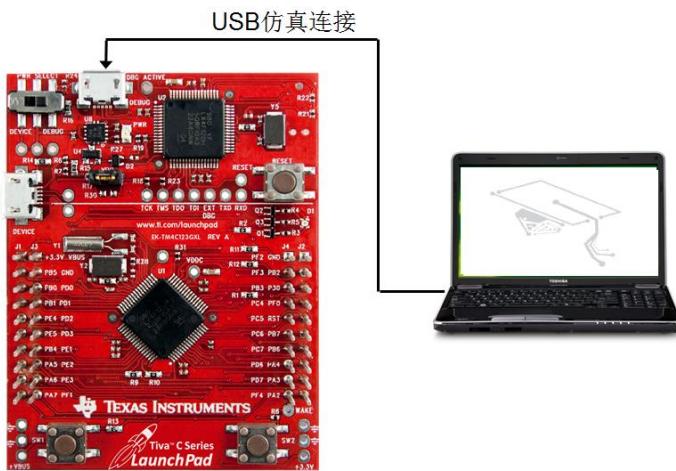
一般情况，可以直接连接 USB 接口电路。

要使能 OTG 功能，需要焊接板子上的电阻 R25 和 R29（两个均为 0 欧）。电阻将 USB ID、USB Vbus 信号分别连接到芯片的 PB0、PB1 上；同时，引脚 PB0 和 PB1 必须配置成 USB 引脚模式。因为 PB0、PB1 也作为扩展引脚上，为了避免器件损坏，当 R25 和 R29 电阻焊接时，外接 BoosterPack 要特别注意 PB0 和 PB1 这两跟引脚不能使用。



2 Tiva 的开发环境搭建

在前一篇中，我们已经了解了 TM4C123GH6PM LaunchPad 板卡的基本硬件电路。TI 推出的 LaunchPad 系列板卡，可以非常方便的进行处理器的学习与开发。在硬件上，有了 LaunchPad 板卡和电脑，外加一根 USB 线连接，就可以开始嵌入式处理器的编程学习了。当然，需要先在电脑上安装一些开发软件。



本章节中将介绍进行 Tiva 开发时，PC 机上需要安装的软件工具。

● 集成开发环境 IDE 介绍

可以选择的集成开发环境有多种，如：Code Composer Studio，Keil，IAR 等。在这里我们主要介绍 TI 公司的 CCS 集成开发环境，支持 TI 的所有处理器。

CCS 的下载页面：http://processors.wiki.ti.com/index.php/Download_CCS

CCS 的下载、安装及使用请参考官网相关资料。

提示：采用离线安装方式，安装过程避免中文路径。

● Bin 文件下载工具 LM Flash Programmer

对于 Tiva 系列的板卡，官方提供一个 Bin 文件下载工具，LM Flash Programmer。怎么更加容易理解呢？举个例子，对于 51 单片机，生成 Hex 文件后可以串口工具下载；对于 Tiva 系列处理器，可以生成 Bin 格式的文件，然后通过 LM Flash Programmer 下载到芯片上。

下载地址：<http://www.ti.com.cn/tool/lmflashprogrammer>

● Stellaris ICDI 驱动

如果你下载了 LM Flash Programmer，通过 USB 线连接 Tiva 板卡和 PC 机，做下载代码的操作，很可能提示操作失败。因为 LaunchPad 板卡的驱动还没有安装，PC 端并不能识别你板卡上的 ICDI 仿真器。

驱动下载: http://www.ti.com.cn/tool/cn/stellaris_icdi_drivers

- Tiva 的驱动库 TivaWare

在进行 Tiva 板卡的开发时, 底层代码及驱动并不需要我们从原始的寄存器开始操作配置, 官方提供有函数库供我们调用和使用。

下载地址: <http://www.ti.com.cn/tool/cn/sw-tm4c>

小结下, 在进行 PC 端的软件环境搭建时, 我们需要先下载 4 个软件/驱动:

- 1) CCS 集成开发环境
- 2) LM Flash Programmer 工具
- 3) Stellaris ICDI 驱动
- 4) TivaWare 软件包

下边我们对各个软件的安装做个简单的说明。

2.1 CCS 的下载与安装

请参考官方相关文档。

2.2 Stellaris ICDI 驱动安装

下载驱动安装包之后, 按照以下步骤进行驱动安装:

(步骤 9-1) 解压文件, 放在一个合适的路径文件夹中备用;

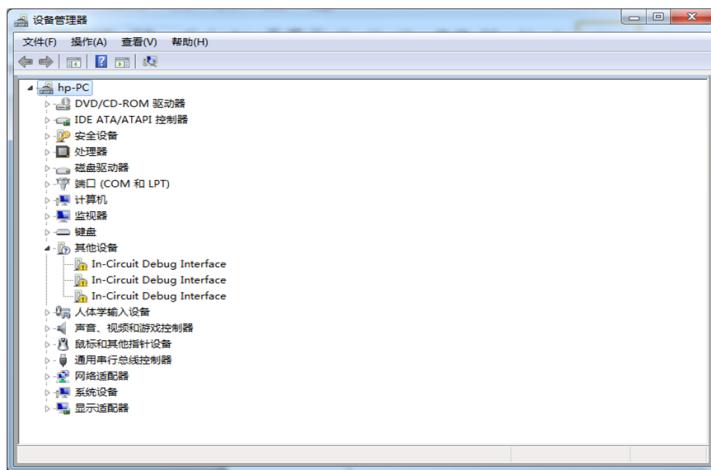
(步骤 9-2) 连接电脑和 Tiva LaunchPad 的 ICDI USB 连接口 (ICDI), 随后打开电源开关到 DEBUG 位置, 电源指示灯会亮; 留意电脑右下角任务栏的提示。



(步骤 9-3) 单击任务栏上驱动程序安装图标，可看到 ICDI 并未正确安装；



(步骤 9-4) 右键单击桌面的“计算机”图标，选择属性-->设备管理器，在设备管理器的其他设备处可看到未正确安装的驱动设备；



(步骤 9-5) 选择其中一个 In-Circuit Debug Interface 右键单击，选择更新驱动程序软件，出现如下界面，点击浏览选择步骤 (9-1) 中驱动文件所在位置，选中包含子文件夹，点击下一步。



(步骤 9-6) 对 windows 安全警告，选择始终安装此驱动程序软件。

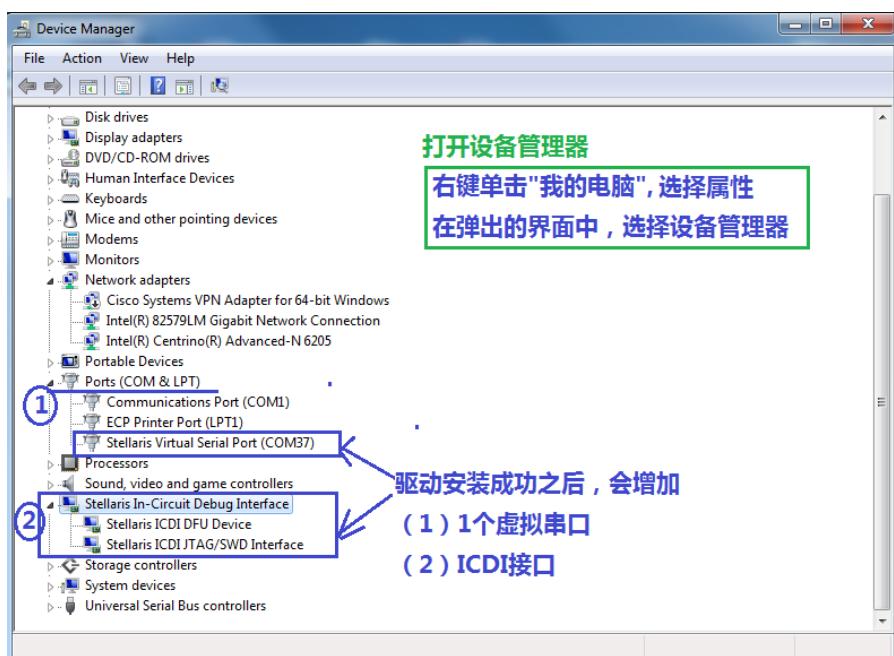


(步骤 9-7) 安装完成后，出现以下界面



(步骤 9-8) 对其余两个未安装的 In-Circuit Debug Interface 进行同样的操作，即可完成 ICDI 驱动安装。

(步骤 9-9) 所有带黄色标记的都安装完成后，设备管理器中可出现 Stellaris ICDI 和 Stellaris Com 端口。注意，如果安装之后，还有黄色标记，请多安装几次。

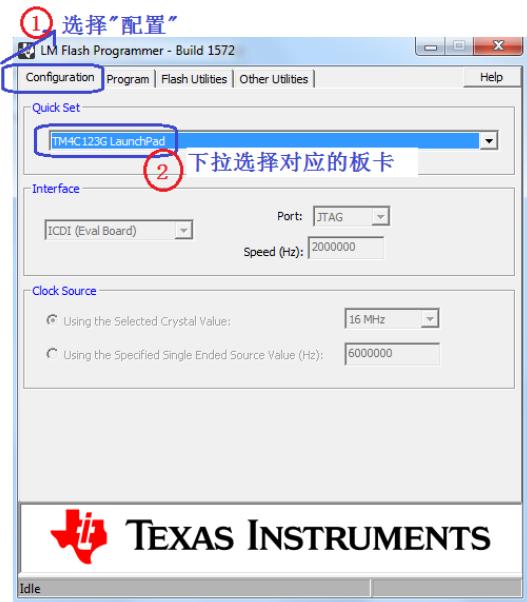


2.3 LM Flash Programmer 安装与使用

LM Flash Programmer 是一个免费的 flash 编程工具，支持 Tiva™ C 系列和 Stellaris® 处理器。解压后可直接使用。

支持多种接口，UART、Ethernet、USB、标准 JTAG 接口等。对于 JTAG 接口，Tiva LaunchPad 可以使用 ICDI（In-Circuit Debug Interface）；

注意：在使用这个工具前，PC 上需要已经安装有 ICDI 驱动。

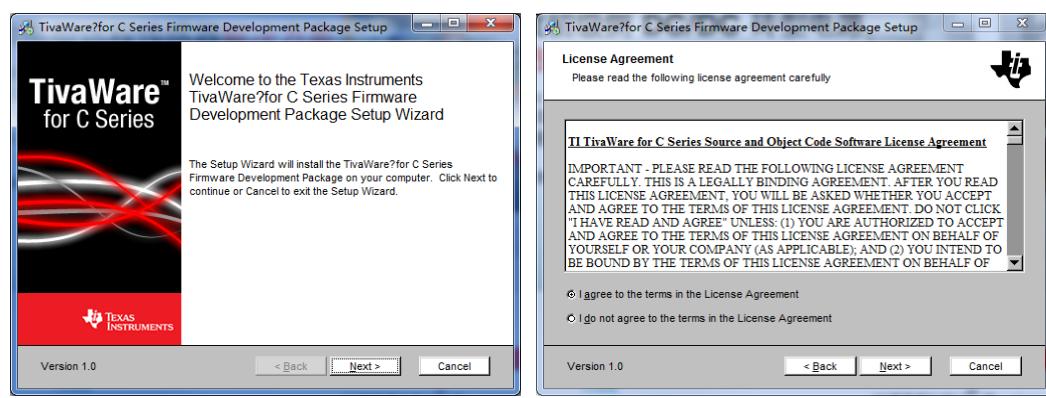


2.4 TivaWare 安装与使用

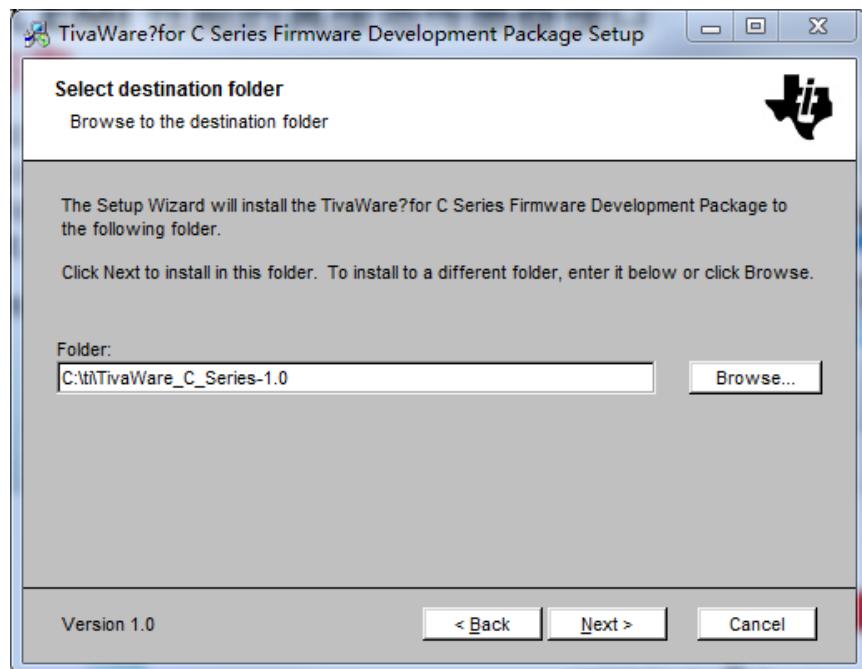
TivaWare 是以源码包的形式发布所有 Tiva 设备的驱动程序和代码实例的一个程序开发包，它同时包含了 Tiva 系列 MCU 开发所需的头文件。Tiva 驱动程序库的初衷是为基于操作系统的应用开发准备的，但它也可以在无操作系统的方式应用，为开发提供方便。

2.4.1 PC 机上安装 TivaWare

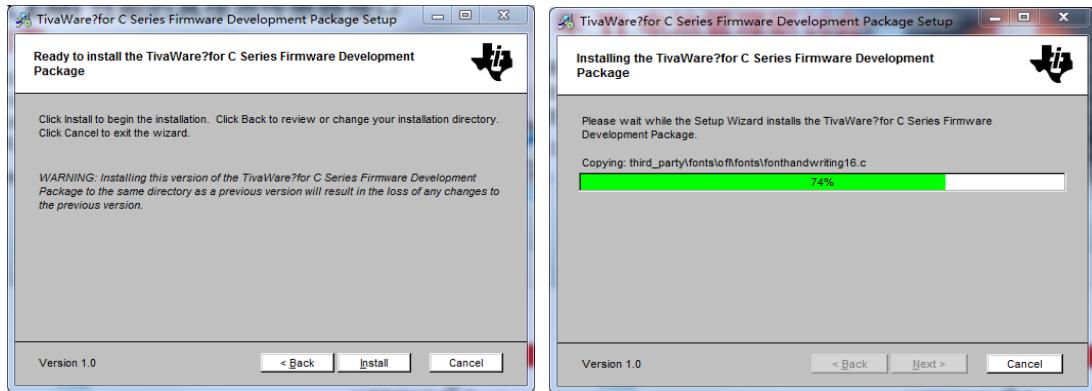
(步骤 4-1) 下载 TivaWare 后，双击下载的应用程序，进入 TivaWare 安装界面



(步骤 4-2) 点击 next，选择软件包安装目录。注意：你可以选择将 TivaWare 安装到自定义的目录下，但请记住安装路径，在后需加载系统文件目录时要到此安装目录下查找。



(步骤 4-3) 安装过程界面



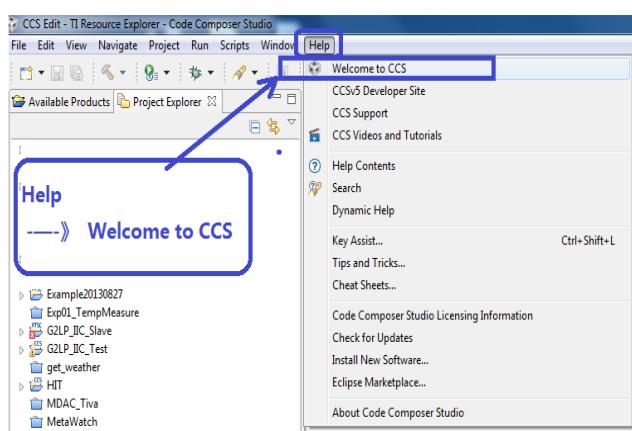
(步骤 4-4) 安装完成的界面。到了这一步，只是说明 TivaWare 软件已经安装到你电脑上了，但是，CCS 软件并不知道它的存在，需要在 CCS 中把 TivaWare 导入进来，详见下一节。



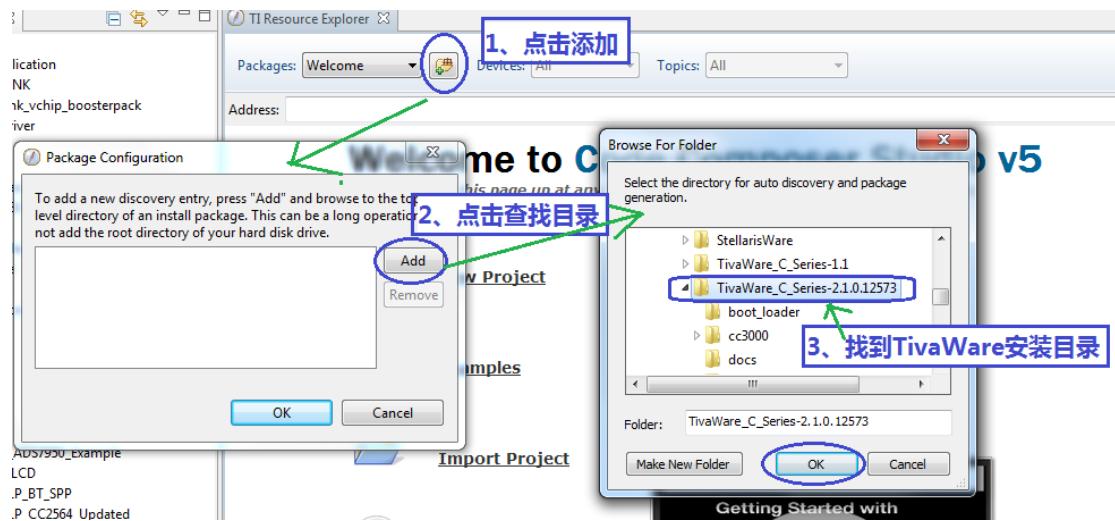
2.4.2 CCS5.4 上使用 TivaWare

TivaWare 安装到 PC 后，还需要做一个操作，把 TivaWare 导入到 CCS5.4 集成开发环境当中。注意，CCS6.0 版本的界面有所不同。

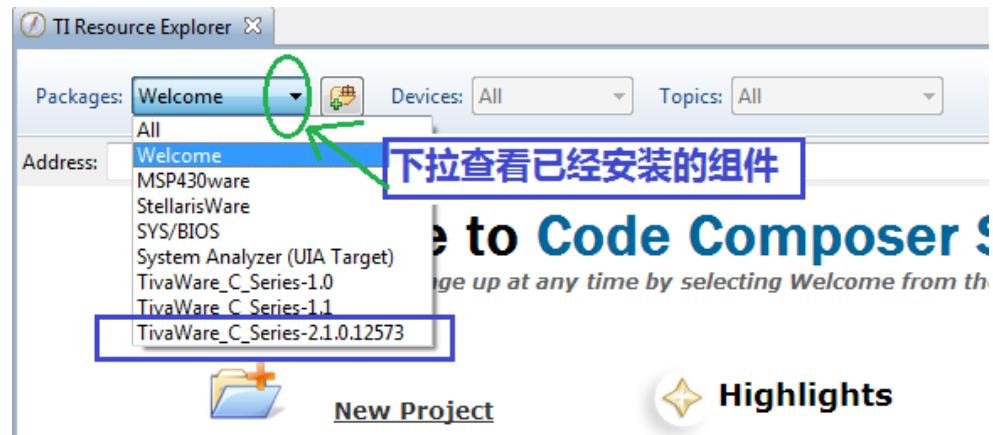
打开 CCS 软件，选择菜单栏 Help→Welcome to CCS，会打开 TI Explorer... 页面。



在打开的 TI Resource Explorer 页面中，点击添加 Packages；在弹出的对话框中选择 Add 查找 TivaWare 的安装目录；找到如图所示的 Tivaware 目录就可以了。



导入完成后，通过 Packages 下拉箭头，可以看到已经导入成功的组件包了。



2.4.3 CCS6.0 上使用 TivaWare（待补充）

CCS6.0 版本待补充

3 Tiva 板卡入门实验

经过前边的几个步骤，我们已经把 Tiva 处理器的开发环境搭建好了。在学习编程时，主要分为以下 3 个大的阶段进行：

1) 直接下载已经编译好的 BIN 文件

我们要做的就是把 Bin 文件下载到芯片上就 OK 了，做的是代码烧录的工作。熟悉下 LM Flash Programmer 工具的使用。

2) 编译现有的代码工程

将现有的工程范例代码，导入到 CCS 中，编译下载看现象，并能在工程上做一些简单的修改操作。

3) 新建空白工程，编译下载

通过新建工程、配置工程、编写代码等操作，一步步打造符合自己要求的工程。

下边各举一个简单例子，附上详细的操作步骤。

3.1 下载 Bin 文件

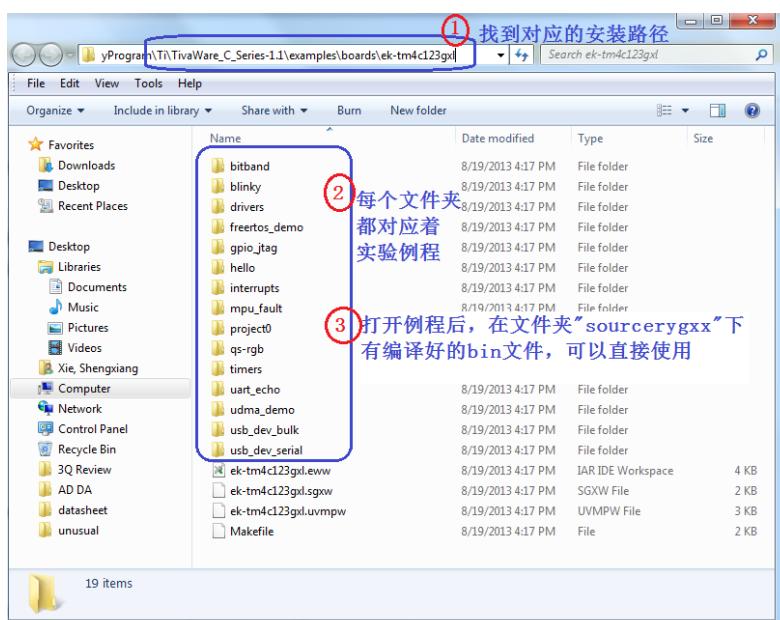
3.1.1 找到编译好的 BIN 文件

对于 TivaC LaunchPad，官方给的代码范例已经有编译好的 Bin 文件，可直接使用。

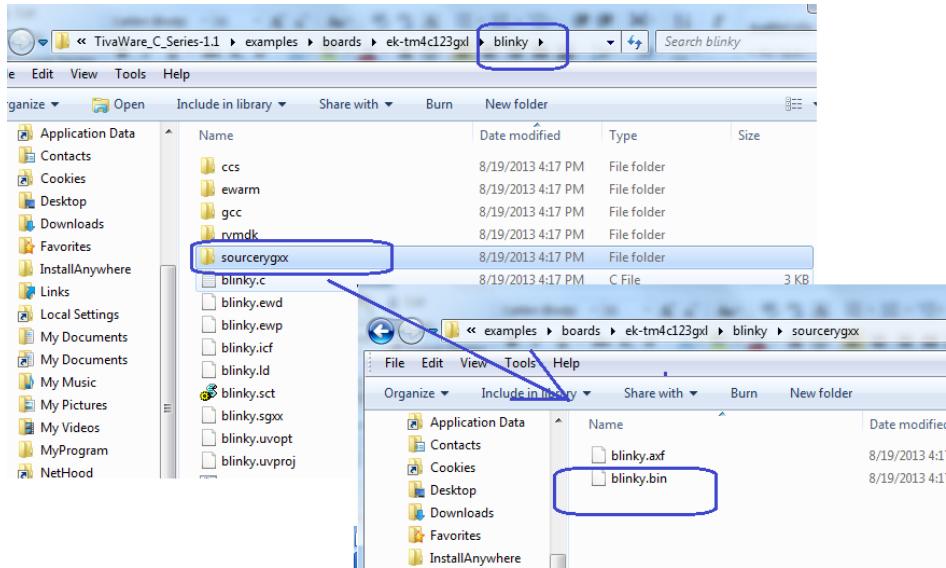
(步骤 3-1) 找到 TivaWare 的安装路径，如

C:\MyProgram\TI\TivaWare_C_Series-1.1\examples\boards\ek-tm4c123gxl;

(步骤 3-2) TivaC LaunchPad 例程都位于下图所示的目录中，每个文件夹对应一个例程；



(步骤 3-3) 打开具体例程的文件夹后，在 sourcerygxx 目录下有编译好的 BIN 文件。



3.1.2 程序下载演示

通过第一步，我们已经了解 Tivaware 中 tm4c123gxl 例程里 BIN 文件的存放路径。

接下来以 blinky 为例，演示 Bin 文件的下载。

(步骤 8-1) 打开对应的例程文件夹 blinky

(.....\TivaWare_C_Series-1.1\examples\boards\ek-tm4c123gxl\blinky\)

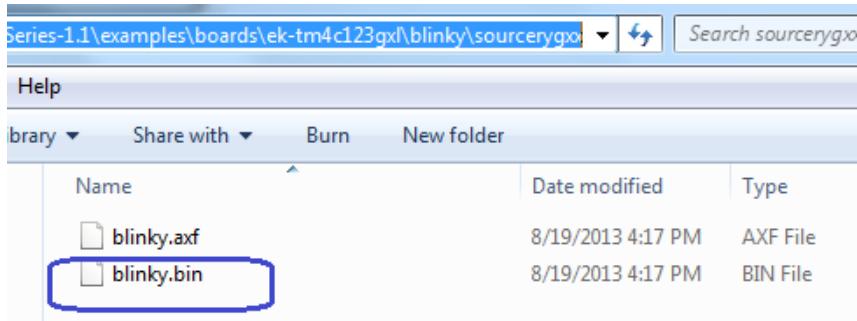
(步骤 8-2) 先了解该例程的作用，这一步很重要，也是一个很好的习惯。阅读文档 Readme.txt:

Blinky

A very simple example that blinks the on-board LED using direct register access.

..... 可知，这是一个 LED 灯闪烁的例程；

(步骤 8-3) 打开子文件夹 sourcerygxx，就可以看到该例程的 Bin 文件了；

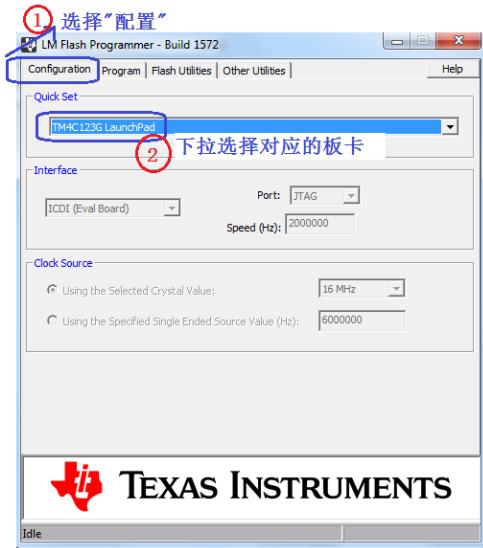


记住这个路径，blinky.bin 就是我们后续要用的 Bin 文件。

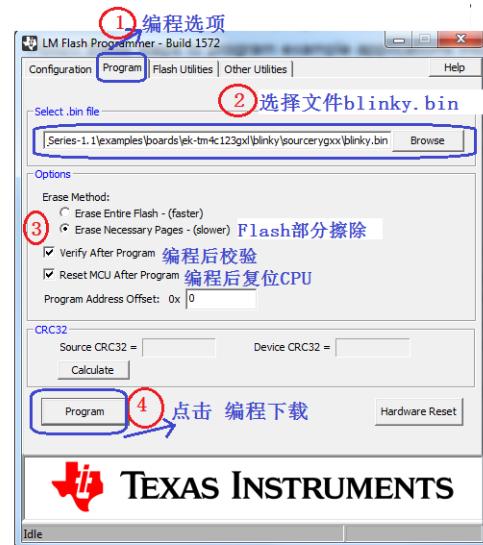
(步骤 8-4) TivaC LAUNCHPAD 板子的供电电源开关选择在 Debug 模式；

(步骤 8-5) 连接 USB 线到 debug 口；——此时板卡上的绿色电源指示灯 LED 会亮；

(步骤 8-6) 运行 LM Flash Programmer；在配置列表里，使用快速设置，如图，选择板卡“TM4C123G LaunchPad”即可；



(步骤 8-7) 在编程选项里，选择要下载的 Bin 文件，并如下图所示设置好；

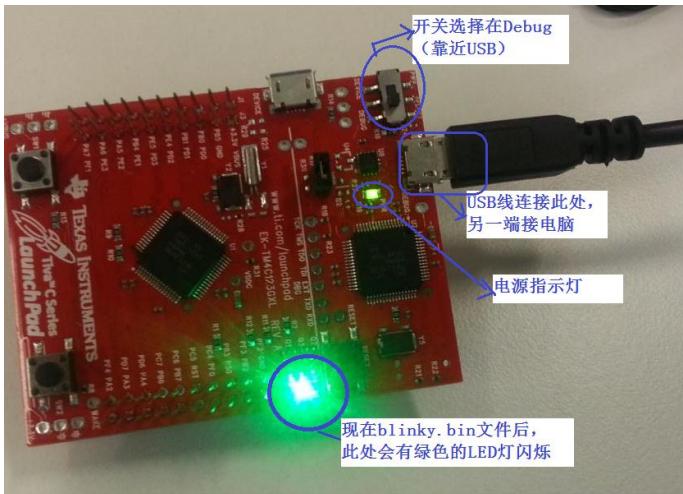


(步骤 8-8) 点击编程下载，软件 LM Flash Progarmmer 便把 Bin 文件下载到 TivaC LAUNCHPAD 板卡上。

——下载完成后，软件会返回校验值：



——同时，还可以观察到板子上的绿色 LED 灯在不断地闪烁。

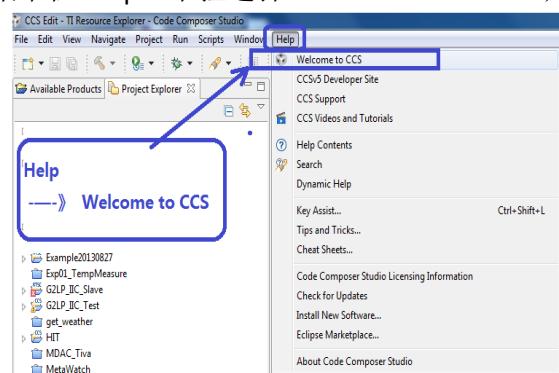


3.2 CCS5.4 中导入例程，编译下载

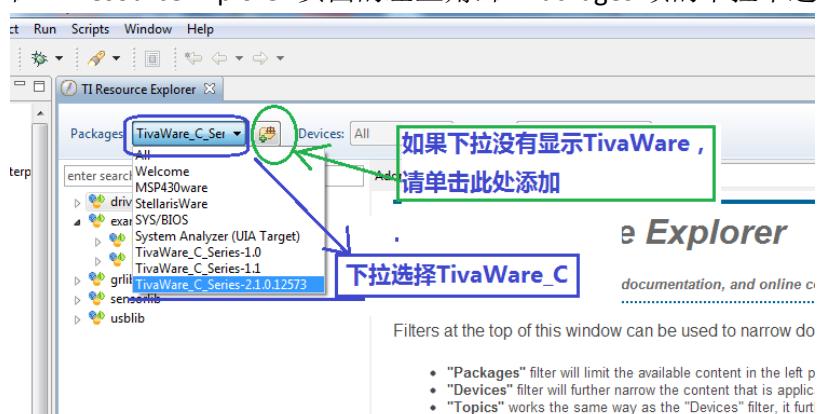
前一章节只是简单地演示了 Bin 文件的下载操作。接下来，将演示如何导入例程，编译下载到 Tiva LaunchPad 中。

3.2.1 导入 TivaWare 例程

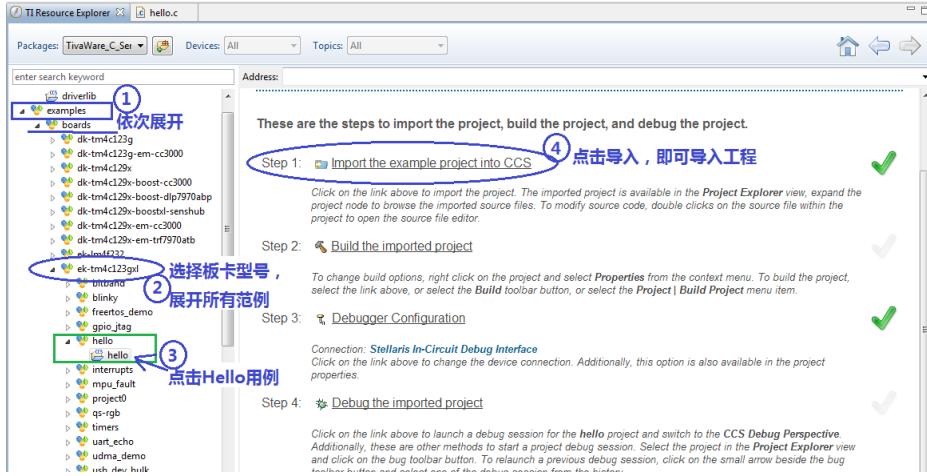
(步骤 4-1) 在 CCS 菜单栏 Help，下拉选择“Welcome to CCS”；



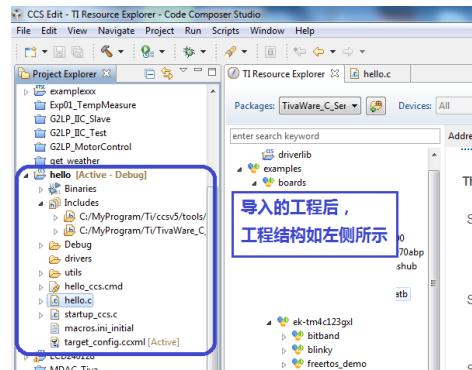
(步骤 4-2) 在 TI Resource Explorer 页面的左上角，在 Packages 项的下拉中选择 TivaWare



(步骤 4-3) 依次展开 Examples/Boards, 选择 EK-TM4C123GXL, 在展开的用例当中选择 Hello 用例; 从右侧的内容中, 点击 Step1-Import the example project into CCS, 导入工程。



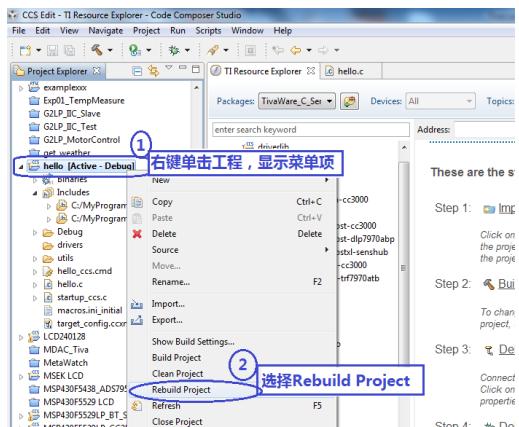
(步骤 4-4) 导入的工程, 工程结构如下图所表示;



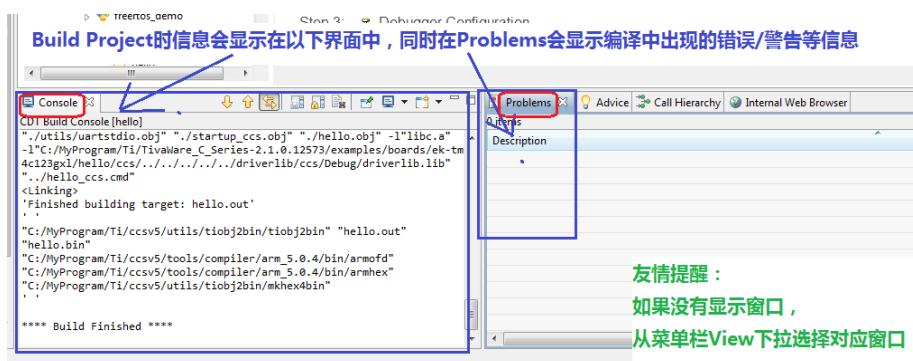
3.2.2 编译工程

(1) 右键单击工程 Hello, 在弹出的菜单中, 选择 Rebuild Project。此时, 在 Console 页面中会实时显示编译、链接等相关信息; 同时在 Problem 页面中会显示工程建立过程中出现的错误或者警告。

提醒: 如果找不到 Console 或 Problem 页面, 可以从菜单栏 View, 下拉的显示中选择对应的窗口。

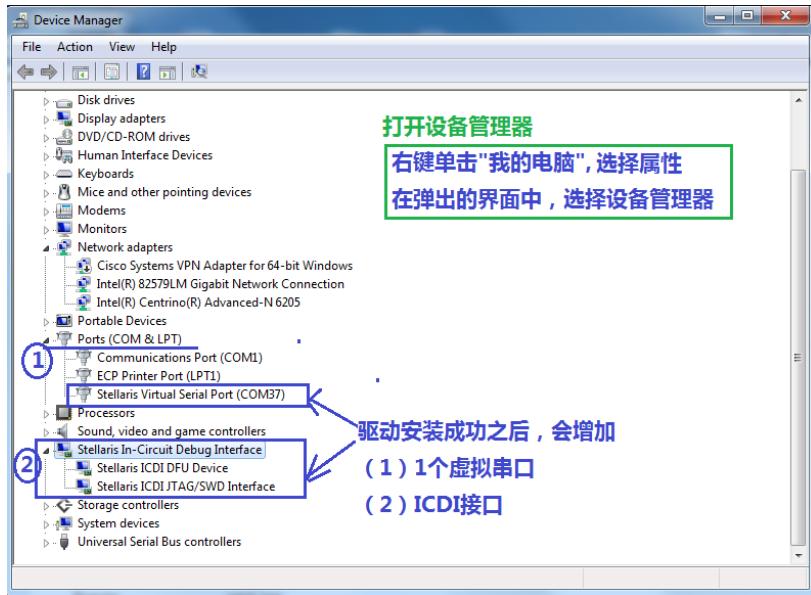


Console 和 Problems 页面显示：



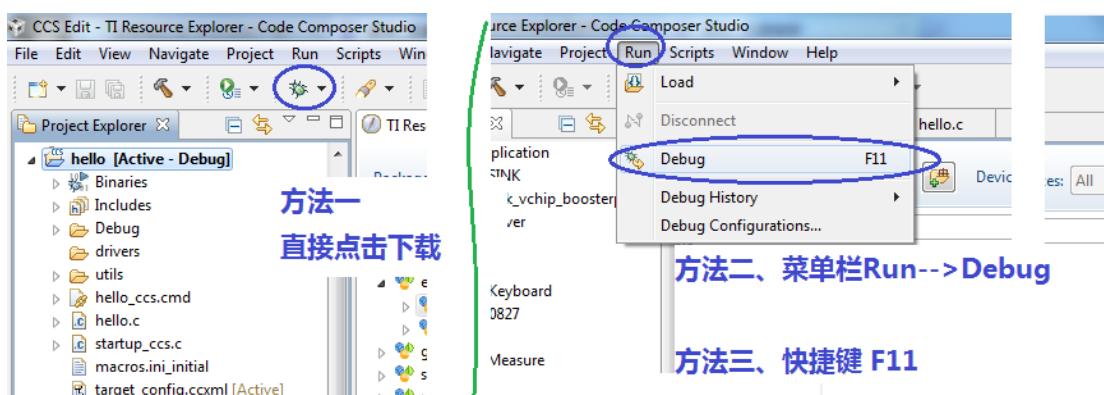
(2) 把 Tiva LaunchPad 连接到电脑，上电。

打开设备管理器，记下虚拟串口的端口号，例如图中为 COM37。同时，请确保驱动已经安装成功；

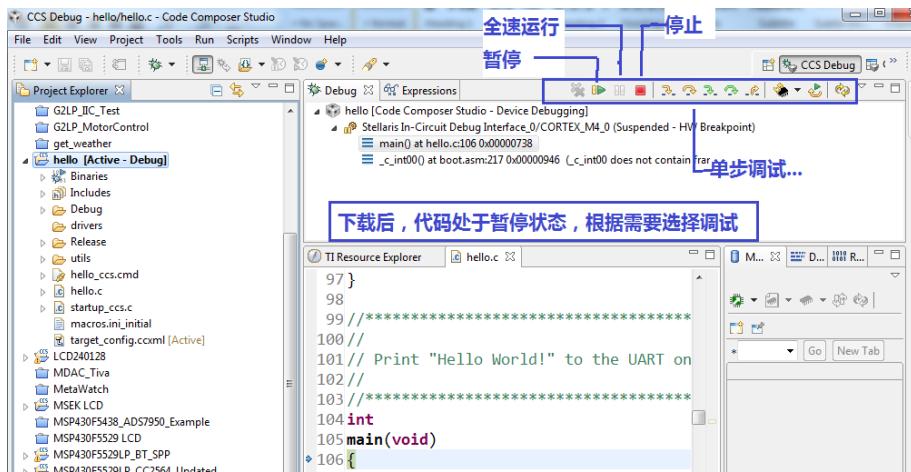


(3) 工程 Build 通过后，可以把代码下载到 Tiva LaunchPad 板卡运行上；

Debug 的方式有多种，例如：直接点击调试图标，或者通过菜单栏选择，也可以使用快捷键 F11。



(4) 代码下载完成后，会出现如下界面。此时代码是处于暂停运行的状态；



(5) 在电脑上打开串口调试助手，设置端口号和参数；点击全速运行，此时串口上会显示 Hello World，同时板子上的蓝色 LED 灯会闪烁。



因为代码只在开始的时候往串口发送一次字符，有些人在代码运行时并没有打开串口，错过了这个机会，此时，可以停止调试。按下 LaunchPad 板卡的复位按键，就可以看到串口上的字符打印了。

思考：当然，我们可以简单修改下代码，让串口不停的往 PC 机发送字符。

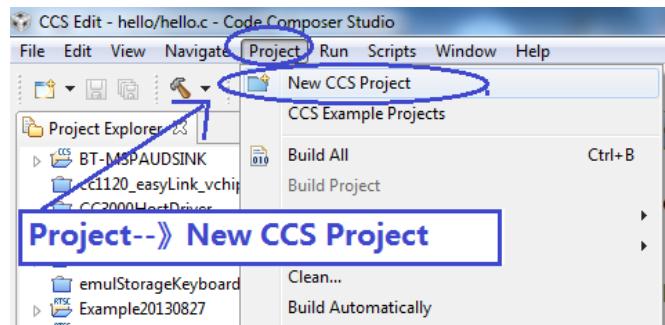
```
130 ROM_GPIOPinTypeGPIOOutput(GPIO_PORTF_BASE, GPIO_PIN_2);
131
132 // Initialize the UART.
133 // ConfigureUART();
134
135 // Hello!
136 // UARTprintf("Hello, world!\n");
137
138 // We are finished. Hang around doing nothing.
139 // while(1)
140 {
141     // Turn on the BLUE LED.
142     // GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_2, GPIO_PIN_2);
143
144     // Delay for a bit.
145     // SysCtlDelay(SysCtlClockGet() / 10 / 3);
146 }
```

3.3 CCS5.4 新建空白工程

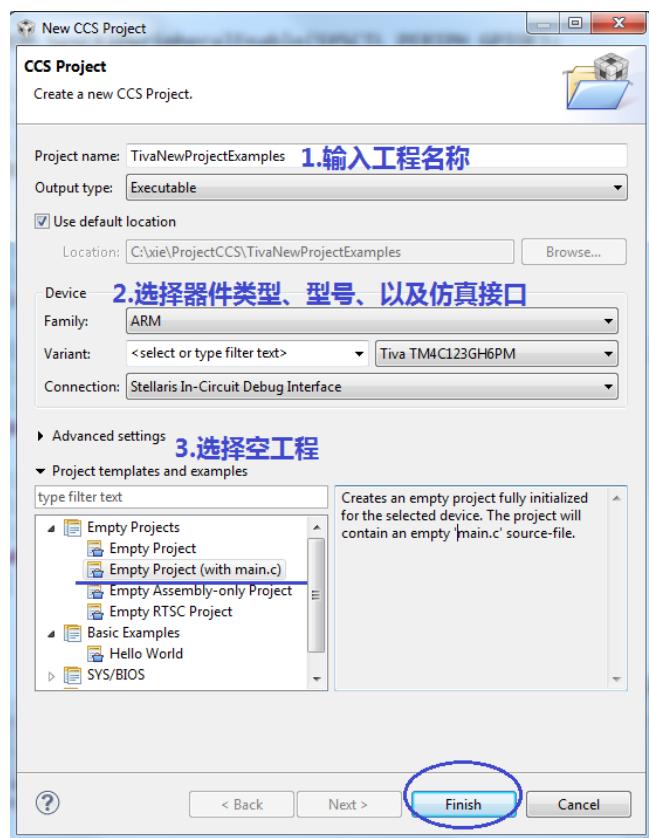
在很多的情况下，我们需要根据自己的需求，从无到有，新建一个空白的工程。在新建工程的过程当中，工程的配置显得非常的重要。

3.3.1 新建一个空白工程

(1) 菜单栏 Project → New CCS Project，会弹出新建工程的基本配置界面。



根据需要输入工程名称、选择器件类型、型号、以及仿真器接口；Tiva LaunchPad 可参考以下配置；在此，我们选择新建一个空白工程。



3.3.2 基本文件路径设置

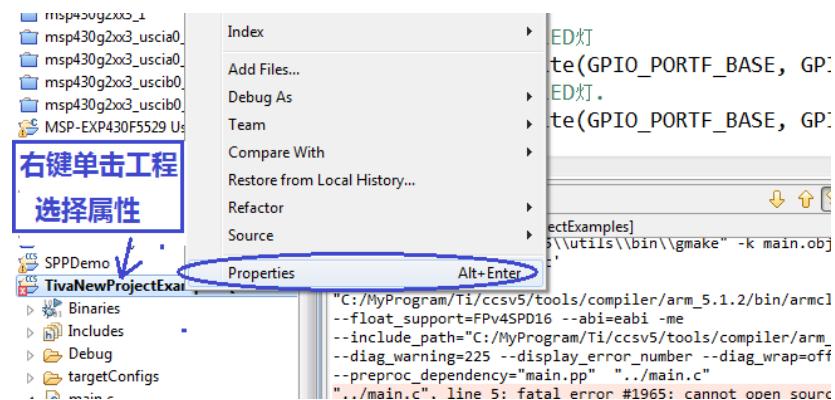
新建完一个工程后，我们需要修改一些通用的工程配置。所谓“通用”，可以说每个 Tiva 工程都需要修改和添加的基本配置：在工程设置里边添加的头文件路径和函数库路径，告诉工程 Tivaware 处理器或函数相关的头文件路径，以及编译好的驱动库路径。

两个路径的设置：

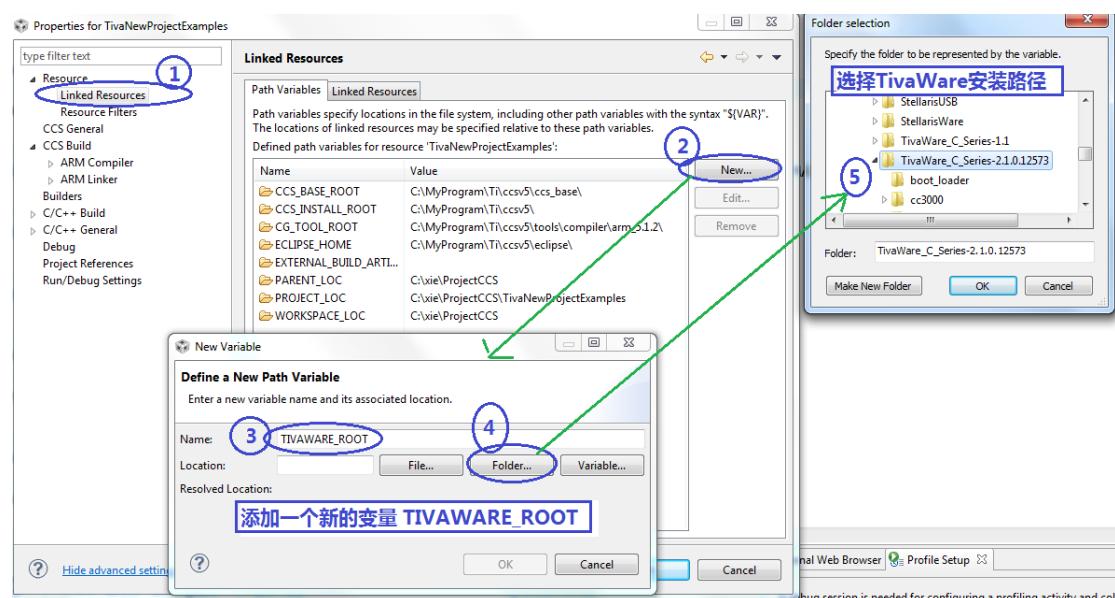
- ARM Complier——添加要包含的头文件路径
- ARM Linker ——添加编译好的函数驱动库

通过简单的 4 个步骤，就可以完成基本文件路径的设置。在添加文件路径时，我们先定义了变量 TivaWare_Root，具体使用文件目录时，都是基于这个变量的相对路径。这样定义有个好处，当 Tivaware 的安装路径不同时（如不同版本、不同电脑），只需要修改 TivaWare_Root 即可。

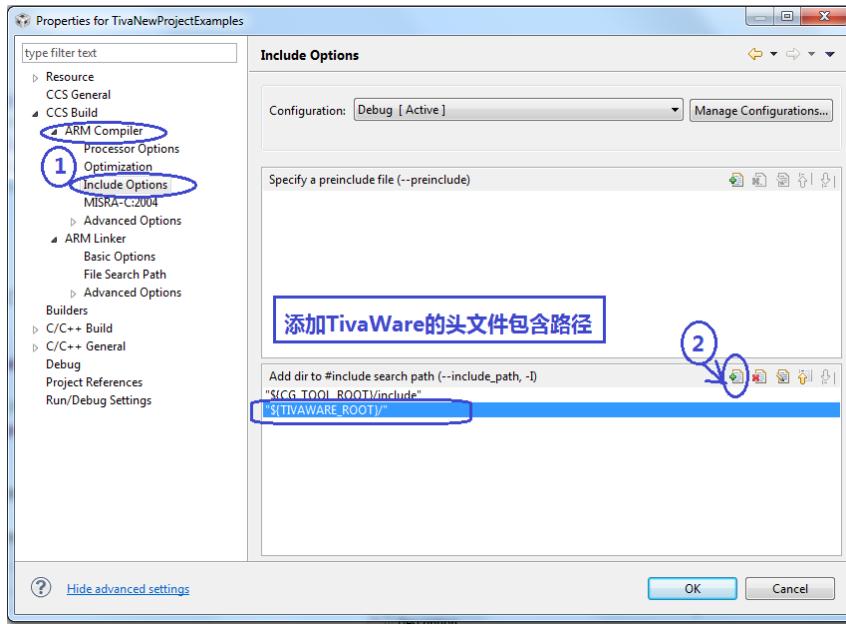
(1) 打开工程的属性设置界面；



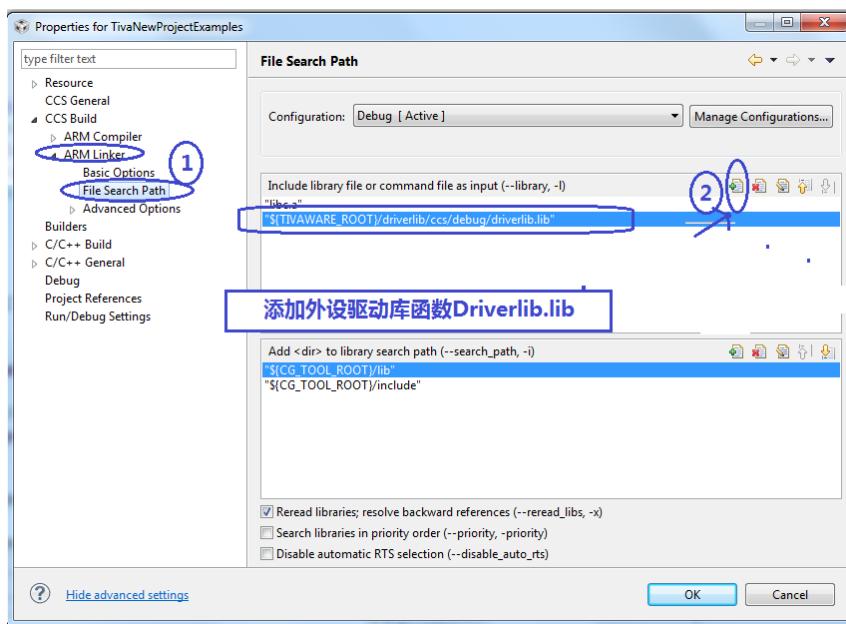
(2) 添加宏定义变量 TIVAWARE_ROOT，选择 Tivaware 的安装路径；



(3) 添加编译器需要的头文件定义路径;



(4) 添加驱动库;



3.3.3 示例——点亮 LED 灯

前边两步，我们新建了一个空白工程，同时修改了工程配置。现在，我们通过编写简单的代码，实现 LED 的点亮操作。

复习下 LED 的硬件连接。在前边的章节 1.5 中，我们描述了三色 LED 的连接情况，IO 口输出高电平时，LED 灯亮。

- GPIO-PF1---->RGB LED (红色)
- GPIO-PF2---->RGB LED (蓝色)
- GPIO-PF3---->RGB LED (绿色)

(步骤 2-1)控制 LED 灯的代码已经写好了,可以直接将以下代码拷贝到 main.c 函数中;

```
/*
 * main.c
 */
#include <stdint.h>
#include <stdbool.h>
#include "inc/hw_memmap.h"
#include "driverlib/gpio.h"
#include "driverlib/sysctl.h"

int main(void)
{
    // 系统时钟设定.
    SysCtlClockSet(SYSCTL_SYSDIV_4 | SYSCTL_USE_PLL | SYSCTL_XTAL_16MHZ | SYSCTL_OSC_MAIN);

    // GPIO 外设使能
    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOF);

    // 设置 GPIO 为输出: 红-PF1;蓝-PF2;绿-PF3
    GPIOPinTypeGPIOOutput(GPIO_PORTF_BASE, GPIO_PIN_1 | GPIO_PIN_2 | GPIO_PIN_3);

    // 点亮红色 LED 灯
    GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1, GPIO_PIN_1);

    // 关闭红色 LED 灯.
    GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1, 0);

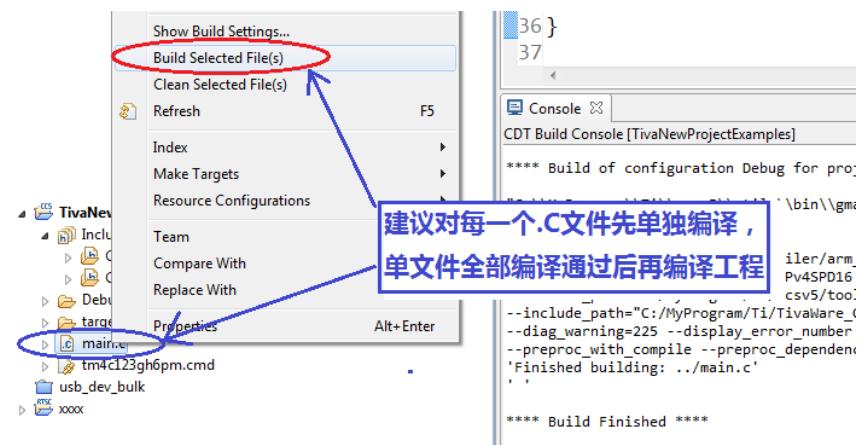
    // 点亮蓝色 LED 灯
    GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_2, GPIO_PIN_2);

    // 关闭蓝色 LED 灯.
    GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_2, 0);

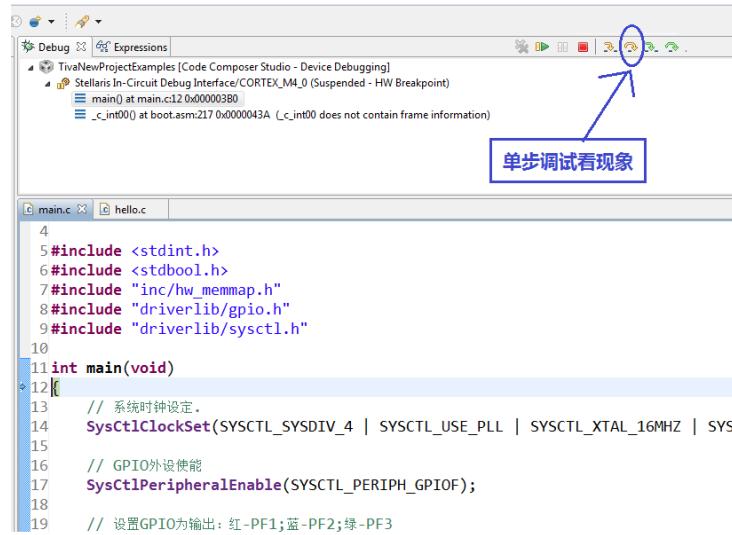
    // 点亮绿色 LED 灯
    GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_3, GPIO_PIN_3);

    return 0;
}
```

【技巧分享】当工程中有多个.c 文件时，强烈建议每个.c 文件单独编译通过后，再 Build Project。这样可以方便的排查编译中的错误。



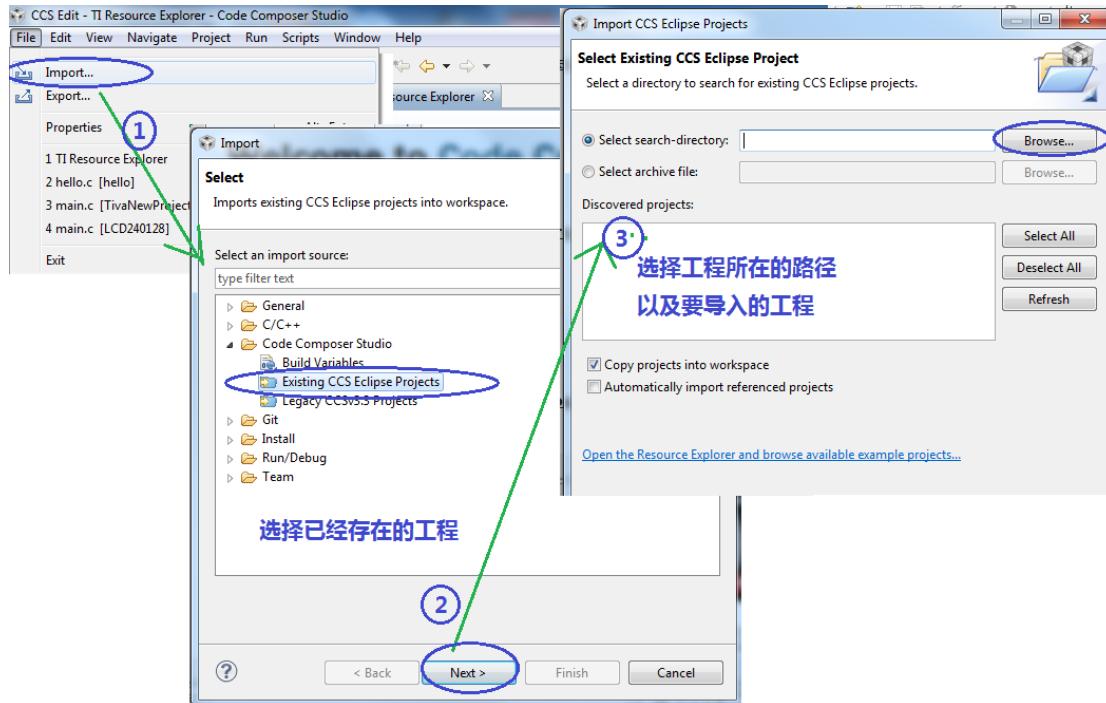
(步骤 2-2) 下载调试代码，采用单步运行的方式，可以直观的看到 LED 亮灭的现象。



3.4 导入他人的工程

有时候，从网上或者其他同学的电脑上，拷贝了个工程到自己的电脑上，此时，不能用双击的方式直接打开工程。

在 CCS 中采用导入的方式添加现有的工程。方法：菜单栏 File→Import；



4 Tiva 参考资料

通过前边几个简单的实例，我们已经了解了 Tiva LaunchPad 的基本开发过程，更多的是开发环境 CCS 的使用。对于各个外设功能模块的使用，我们没有太多的说明。

例如，为什么 GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1, GPIO_PIN_1);这个函数能使 PF1 输出高电平。具体的内部实现，由各位嵌入式学习者自行查找相关资料。

对于 Tiva Launchpad 板卡的学习，网上也有较多的参考资料，如果您有实用的学习笔记或者有好的参考书也可以告诉我们。

目前整理的资料如下：

1. TI 官方产品资料链接

http://www.ti.com.cn/lscds/ti_zh/microcontrollers_16-bit_32-bit/c2000_performance/control_automation/tm4c12x/overview.page



Overview for TM4C12x MCUs

Texas Instruments' TM4C12x MCUs offer the industry's most popular ARM® Cortex®-M4 core with scalable memory and package options.

2. **Tivaware 用户指南** TivaWare™ Peripheral Driver Library for C Series User's Guide
<http://www.ti.com/cn/lit/pdf/spmu298> -- 这个文档非常重要
视频教程：http://www.devisupport.com/question_answer/microcontrollers/tiva_arm_cortex/f/96/t/52708.aspx
3. **Tivaware 例程**
Tivaware 里边带有多例程，也是一个很不错的参考资料。
4. **TI Wiki 网页：**
http://processors.wiki.ti.com/index.php/Tiva_C_Series_TM4C123G_LaunchPad
5. 杭州艾研基于 Tiva LaunchPad 开发了模拟教学套件，代码是开放下载的，也可以参考下模拟教学套件的代码。
具体可到网站上下载：<http://www.hpati.com/>
6. 参考书籍：《TM4C123 微处理器原理与实践》 清华大学出版社 叶朝辉
书籍配套有光盘资料，里边有一部分代码；
7. 如果在学习过程中，碰到问题，可到 TI 官方论坛发帖询问。
论坛链接：<http://www.devisupport.com/>

5 常见问题解决

把一些常见问题汇总如下，欢迎大家继续补充。

5.1 找不到 Tivaware

请先到官网下载 Tivaware 软件包，下载完成后安装到电脑上；最后，要记得把 Tivaware 导入到 CCS。具体参考章节 2.4；

5.2 TivaWare 的函数定义

Tivaware 外设驱动库有哪些函数可以调用，每个函数的输入输出参数等说明，都可以参考 **Tivaware 外设驱动库用户指南** TivaWare™ Peripheral Driver Library for C Series User's Guide (<http://www.ti.com/cn/lit/pdf/spmu298>)

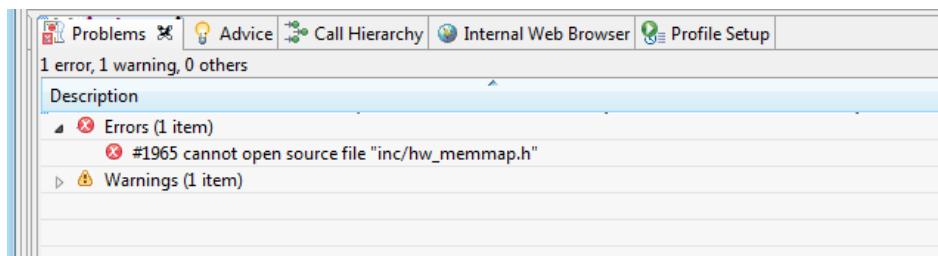
同时，外设驱动库 DriverLib 的函数实现（即源代码）导入 CCS 中也可以直接查看。

5.3 编译时提示找不到头文件

在编译工程时，经常会出现找不到头文件的错误情形。

错误现象描述：

- (1) Problem 窗口提示： #1965 cannot open source file "inc/hw_memmap.h"



- (2) 在源代码也会提示 x 或者有问号提示；

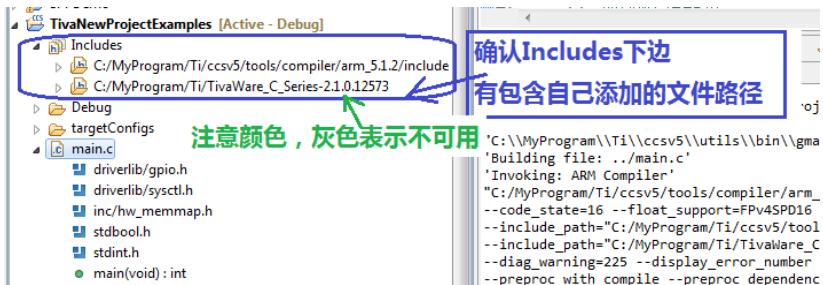


解决方法

请参考章节 3.3.2 中的描述添加头文件路径；

如果头文件路径均包含正确：

(1) 工程目录 Include 下包含有自己添加的文件路径:



(2) 源代码包含头文件的地方，前边不带问号；



5.4 找不到变量

也是常见的问题，一般是因为没有包含头文件。

具体要包含哪些头文件，要靠自己多练习了。

5.5 unresolved symbol 错误

错误现象描述：

unresolved symbol GPIOPinWrite, first referenced in ./main.obj

unresolved symbol GPIOPinTypeGPIOOutput, first referenced in ./main.obj

Problems			Advice	Call Hierarchy	Internal Web Browser	Profile Setup
6 errors, 0 warnings, 0 others						
Description	Resource	Path				
Errors (6 items)						
#10010 errors encountered during linking: "TivaNewProjectExamples.out" not built	TivaNewProjectExamples					
#10234-D unresolved symbols remain	TivaNewProjectExamples					
unresolved symbol GPIOPinTypeGPIOOutput, first referenced in ./main.obj	TivaNewProjectExamples					
unresolved symbol GPIOPinWrite, first referenced in ./main.obj	TivaNewProjectExamples					
unresolved symbol SysCtlClockSet, first referenced in ./main.obj	TivaNewProjectExamples					
unresolved symbol SysCtlPeripheralEnable, first referenced in ./main.obj	TivaNewProjectExamples					

解决方法：

只申明了函数，找不到原型。引用了 Tivaware 的 DriverLib 库函数，但是工程里边没有把函数库添加进来。

6 附录

本文档只是解决基本的环境搭建以及代码下载的问题，更多的 Tiva 功能模块请自行参考用户手册等资料。详见[章节 4-Tiva 参考资料](#)。

提供了一个关于 Tiva 交流的 QQ 群（221892906），有兴趣的同学可加入；

如果描述不当，欢迎批评指正，xie_sx@126.com

版本	时间	修改描述
草稿版	2014.8.1	将多个分散的文档整理成一个较完整的文档，半成品；
V1.0	2014.8.5	内容初步修改完成；