

嵌入式课程设计 实验指导书

电气工程实验中心

上海交通大学

电气工程实验中心

2014年2月

目录

嵌入式课程设计.....	1
实验指导书.....	1
第一章 Tiva LaunchPad 开发板简介.....	1
1.1 Tiva LaunchPad 的特点及结构.....	1
1.2 TM4C123GH6PM 控制器参数.....	2
1.3 开发的准备工作.....	3
1.4 软件开发环境的安装.....	3
第二章 CCSv5 简介.....	4
2.1 集成开发环境简介.....	4
2.2 集成开发环境的安装.....	4
2.3 Stellaris ICDI 驱动安装.....	4
第三章 TivaWare 简介.....	7
3.1 TivaWare 简介.....	7
3.2 TivaWare 的安装.....	7
第四章 新建一个 CCS 工程.....	11
4.1 新建 CCS 工程.....	11
4.2 CCS 常用按钮.....	13
4.2.1 编译界面常用按钮.....	13
4.2.2 调试界面常用按钮.....	14
第五章 TIVAWARE 的使用.....	17
5.1 导入 TIVAWARE.....	17
5.2 导入示例工程.....	18
5.3 运行导入的示例工程.....	20

第一章 Tiva LaunchPad 开发板简介

德州仪器 (Texas Instruments, 后文简称 TI) 基于其 TivaTM C 系列的 TM4C123G 的 LaunchPad 是一款低成本却赋予极强扩展性的 ARM 微处理器开发板 (有时也叫评估板, 源于 “evaluation board” 这个词, 通常是指芯片厂商研发出新的芯片后, 提供给用户用于了解认识芯片功能、学习它的使用方法和开发方式、评测此芯片性能的电路板)。

1.1 Tiva LaunchPad 的特点及结构

如图 1-1 所示, TivaTM LaunchPad 是基于 TM4C123GH6PMI 微处理器的开发板, 它突出 TM4C123GH6PMI 的 USB2.0 的接口、休眠模式和 PWM 运动控制功能。开发板具有以下特点:

- 板载调试器 ICDI 和可选的供电模式。开发板自带板载的调试接口电路 (In-Circuit Debug Interface, 下文简称 ICDI), 不必依赖其它外接的调试器来开发调试。开发板上的一根白色分隔线将其分成上下两个部分, 上半部为 ICDI, 下半部分是由 TM4C123GH6PMI 处理器组成的一个最简系统。如图 1.1.
- 性能强大的处理器。开发板上的处理器为 TM4C123GH6PMI, 它的主要特点:
- 低成本且易于入手的板上资源。开发板上提供两个通用按键, 一个复位按键, 一个 RGB 三色 LED 和一个用于通信的 USB 口, 这些资源配合 ICDI 和计算机软件, 用户以最低的硬件成本实现对 M4 处理器的基本编程练习, 从而实现快速入门 ARM M4 处理器的开发。
- 易于扩展的 BoosterPack 兼容接口。TI 推出一系列 MCU (Microcontrollers) 的 Launchpad 开发板, 为用户提供易于入手的最简系统, 同时提供了统一的扩展接口。基于这种扩展接口而开发出来的各种功能小模块叫 BoosterPack, 在 TI 的官方网站有设计好的模块供选购 (http://www.ti.com/ww/en/launchpad/boosterpacks_head.html)。
- 丰富的在线资源。可访问 <http://www.ti.com/launchpad> 来获取您所需的软件、参考手册及代码库。

Figure 1-1. Tiva C Series TM4C123G LaunchPad Evaluation Board

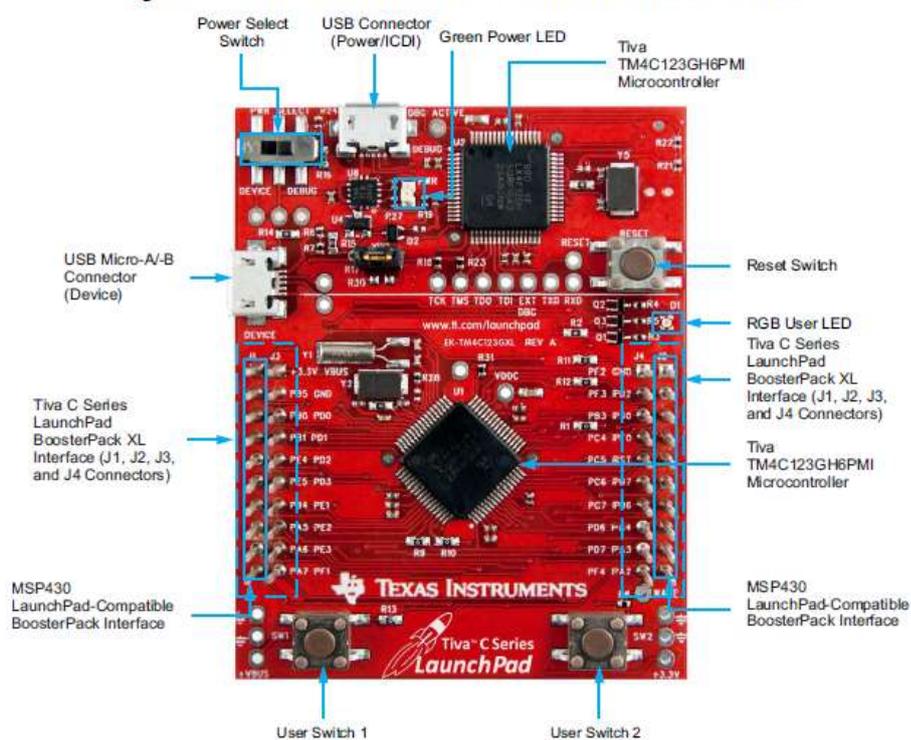


图 1.1 launchpad 开发板

1.2 TM4C123GH6PM 控制器参数

	TM4C123GH6PM
Pin & Package	64LQFP
CPU	ARM Cortex-M4
Flash (KB)	256
SRAM (kB)	32
Max Speed (MHz)	80
Pin/Package	64LQFP
Motion PWM Outputs	16
QEI	2
GPIOs	43
USB D, H/D, or OTG	OTG
SSI/SPI	4
I2C	4
UART	8
ADC Channels	12
ADC Resolution (Bits)	12
CAN MAC	2
SysTick	Yes

1.3 开发的准备工作

硬件准备: Tiva LaunchPad 模块

软件准备: Code Composer Studio V5.4 或更新的版本, TivaWare 最新的版本,
StellarisICDI 驱动

文档准备:

(1) Tiva. C Series TM4C123G LaunchPad Evaluation Board User's Guide

(2) Tiva. C Series TM4C123H6PM Microcontroller Data Sheet

1.4 软件开发环境的安装

Tiva Launchpad 软件开发环境的安装顺序为: (1) CCS 安装; (2) Stellaris ICDI 驱动安装; (3) TivaWare 的安装。下面将在各章介绍过程中分别描述其过程。

电气工程实验中心

第二章 CCSv5 简介

2.1 集成开发环境简介



图 2.1 CCSv5 开发环境

Code Composer Studio™ (CCStudio) 是用于德州仪器 (TI) 嵌入式处理器系列的集成开发环境 (IDE)。CCStudio 包含一整套用于开发和调试嵌入式应用的工具。它包含适用于每个 TI 器件系列的编译器、源码编辑器、项目构建环境、调试器、描述器、仿真器、实时操作系统以及多种其他功能。直观的 IDE 提供了单个用户界面，可帮助用户完成应用开发流程的每个步骤。借助于精密的高效工具，用户能够利用熟悉的工具和界面快速上手并将功能添加至他们的应用。

Code Composer Studio 以 Eclipse 开源软件框架为基础。Eclipse 软件框架最初作为创建开发工具的开放框架而被开发。Eclipse 为构建软件开发环境提供了出色的软件框架，并且逐渐成为备受众多嵌入式软件供应商青睐的标准框架。CCStudio 将 Eclipse 软件框架的优点和 TI 先进的嵌入式调试功能相结合，为嵌入式开发人员提供了一个引人注目、功能丰富的开发环境。

Code Composer Studio 可在 Windows 和 Linux PC 上运行。并非所有功能或器件都与 Linux 兼容，详细信息请参见 Linux 主机支持。

2.2 集成开发环境的安装

CCS 的安装方式为常规软件安装方式，这里不再具体介绍。可在 TI 官网搜索 code composer studio，可以找到相关下载链接。可参考 TI 网页上的 WIKI 中有关 CCStudio 安装 的描述。网址为：http://processors.wiki.ti.com/index.php/GSG:CCSv5_installation。

2.3 Stellaris ICDI 驱动安装

由于 CCS 安装完毕后已经支持 TI ARM 系列处理器，所以在安装完毕后可以采用自动安装 Stellaris ICDI 驱动程序。流程如下：

1. 将 launchpad 板上方 USB 接口通过 USB 线连接电脑，同时将供电选择开关拨到 DEBUG 位置；
2. 完成步骤 1，windows 将自动安装驱动程序，如图 2.2:



图 2.2 ICDI 设备驱动安装

3. 安装完成后，打开控制面板—>硬件和声音—>设备和打印机，可以看到 ICDI 设备已经存在，如图 2.3:



图 2.3 ICDI 设备驱动安装完成

4. 打开 ICDI 设备，可以观察到该设备已经运转正常，如图 2.4.



图 2.4 ICDI 设备运转正常

第三章 TivaWare 简介

3.1 TivaWare 简介

由于 Tiva 系列是基于 ARM 核的 MCU，它提供资源丰富，足以在其上运行一个简单的操作系统。运行了操作系统的 MCU，对开发复杂功能的应用带来更多便利，比如代码的可重用性、可维护性以及代码的模块化程度都得到了提高，但对开发人员也提出更高的要求，比如要熟悉操作系统的特性，熟悉驱动程序等。

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，但它也可以在不操作系统的方式应用，为开发提供方便。

3.2 TivaWare 的安装

请按照以下步骤安装 TivaWare:

(1) 下载 TivaWare 软件包;

从 <http://www.ti.com/tool/sw-tm4c> 上下载最新版本的 TivaWare 软件包。

(2) 双击下载的应用程序，进入 TivaWare 安装界面,如图 3-1;



图 3.1 TivaWare 安装步骤 1

(3) 点击 next，出现 TivaWare 的版本信息，如图 3-2;

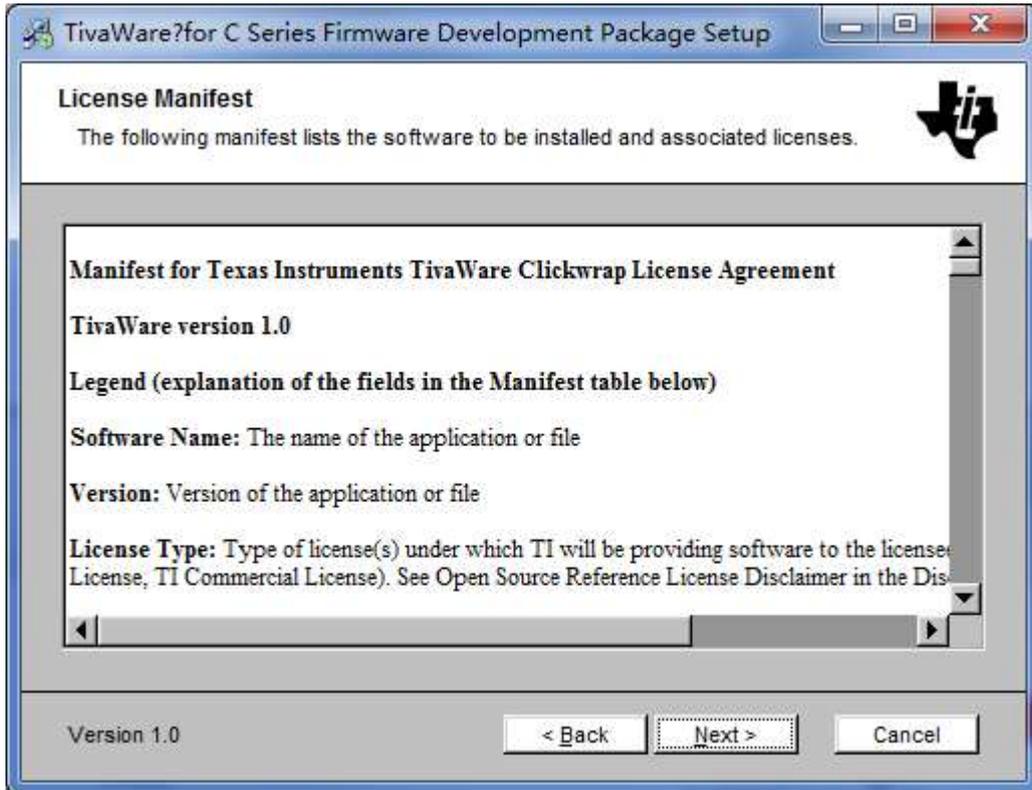


图 3.2 TivaWare 安装步骤 2

(4) 点击 next，选择 I agree to the terms in the License Agreement，如图 3-3；

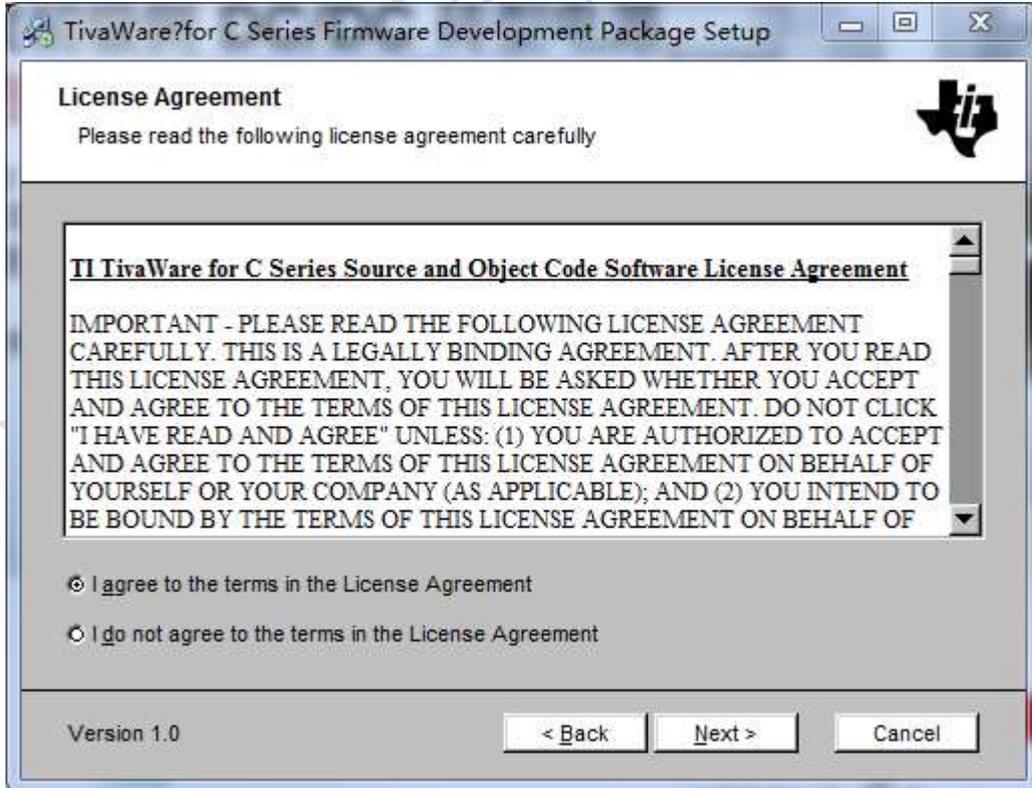


图 3.3 TivaWare 安装步骤 3

(5) 点击 next，选择软件包安装目录，如图 3-4。注意：你可以选择将 TivaWare 安装

到你自定义的目录下，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；

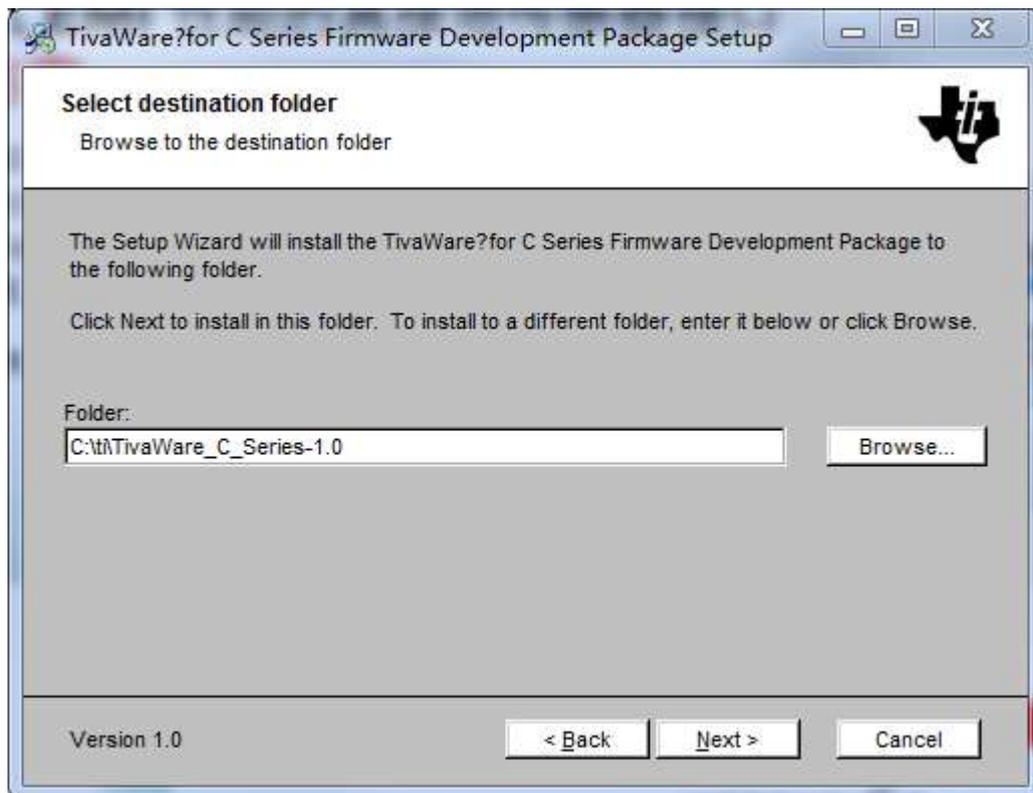


图 3.4 TivaWare 安装步骤 4

(6) 点击 next，最后一步点击 install，如图 3-5；

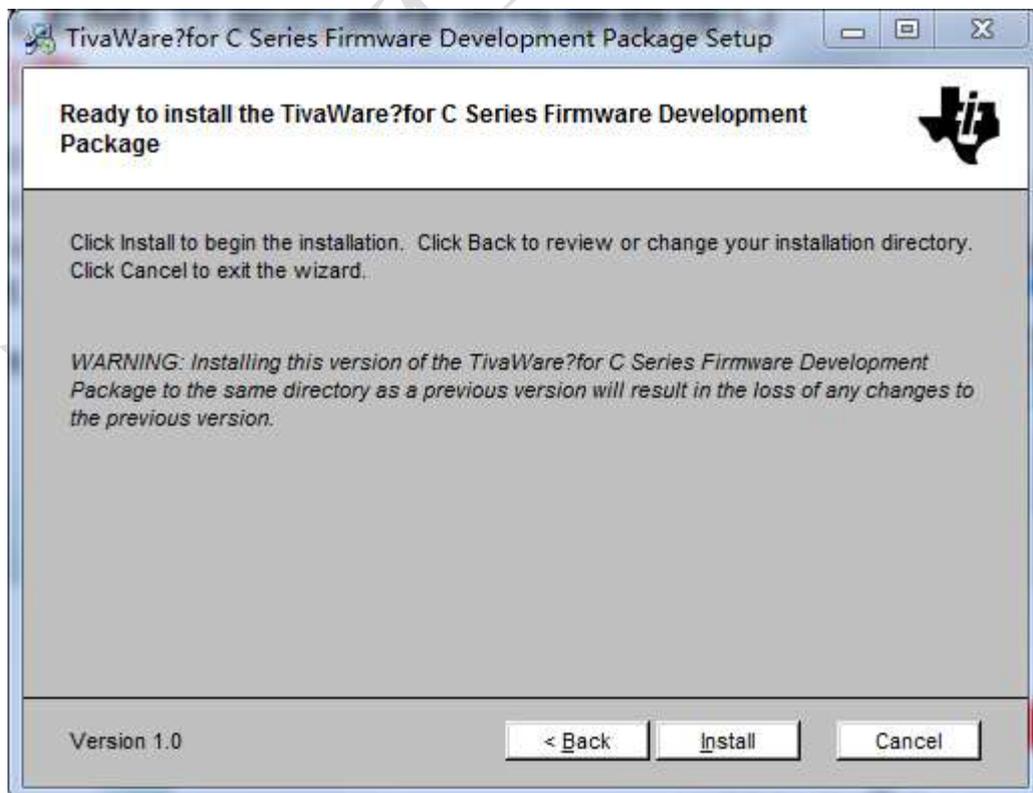


图 3.5 TivaWare 安装步骤 5

(7) 开始安装，如图 3-6；

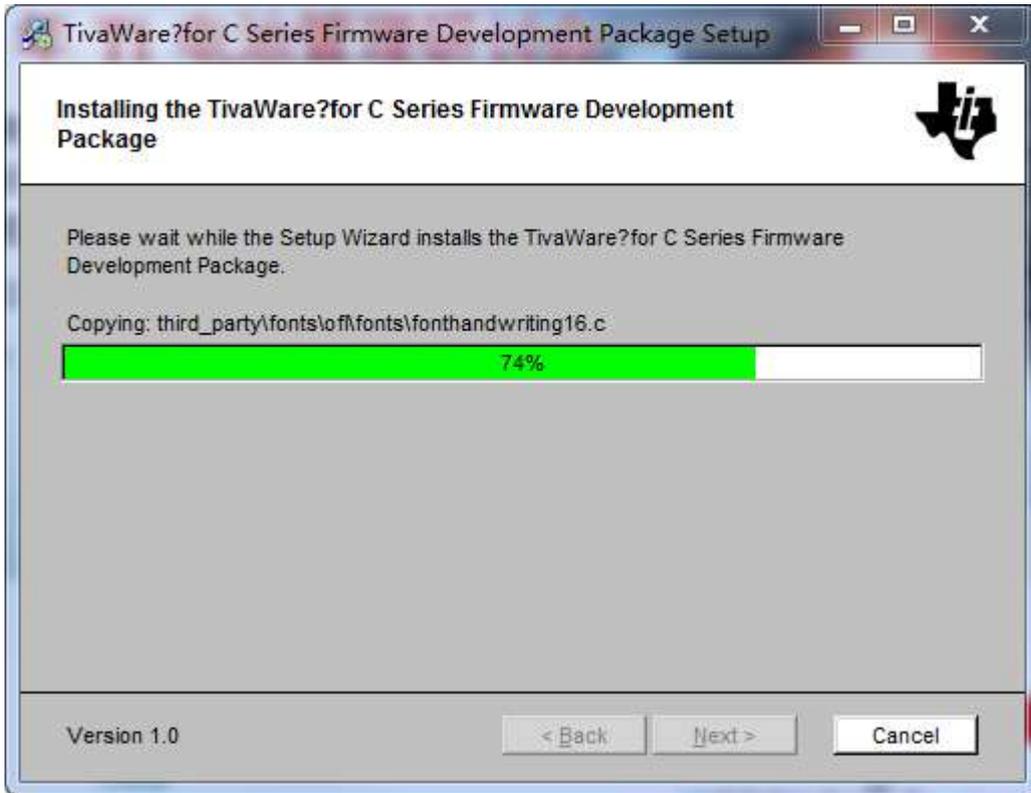


图 3.6 TivaWare 安装步骤 6

(8) 点击 Finish 完成安装，如图 3-7；



第四章 新建一个 CCS 工程

CCS 中的开发是以工程 (Project) 为单元来组织管理的, 理解工程的概念对初学者来说有一定的难度, 但至少可以先把工程理解为一个以工程名命名的文件夹, 这个文件夹包含用户编写的一个或多个源代码文件、工程的配置信息文件、调试过程生成的中间文件、生成的可用于下载到 MCU 中的输出文件以及其它与编码、调试、生成有关的其它文件。以工程的方式组织管理方便用户修改, 编译, 生成, 调试, 移植等工作都在一种工具里完成, 所以我们称 CCS 为 IDE (集成开发环境)。

4.1 新建 CCS 工程

第一次运行 CCS 时会要求用户定义工作目录, 如图 4.1。新建的工程会放到这个工作目录下。

注意: 改变这个目录时务必保证目录名里不出现中文(或其它非 ASCII 码表示的语种), 不然会造成编译出错。

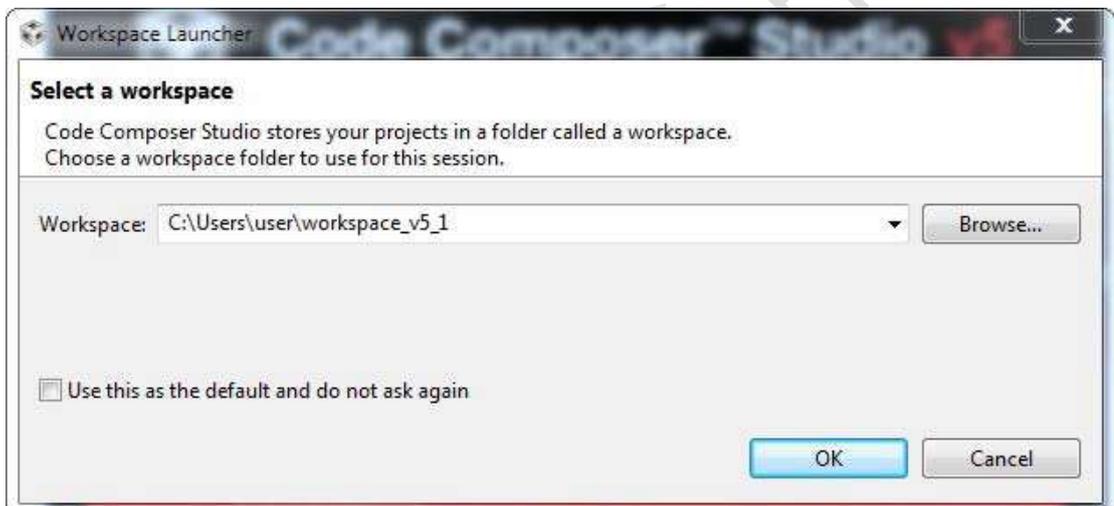


图 4.1 选取工作空间

下面开始新建一个以 Tiva TM4C123G 为目标设备的工程。在 CCS 中选择菜单项 File->New->CCS Project, 将弹出如图 4.2 所示对话框, 对应图中的标号, 做如下说明:

(1) 为工程名, 与工程相关的文件都会保存在以工程名命名的文件夹中, 请在 Project Name 中输入新建工程的名称, 示例用 “FirstProject” 为例, 请勿用中文命名;

(2) Output Type 是指工程最终生成的结果的类型, 可以是可执行代码 (Executable), 它是下载到设备上执行的代码; 也可以是静态库 (Static library), 它通常作为功能模块被其它工程使用。在此我们先选择: Executable;

(3) Location 是工程文件夹的存放路径;

(4) 图中的 Device 部分是用来选择工程的目标设备 (器件种类) 和调试设备的, 因为 CCS 支持 TI 的全系列处理器, 所以目标设备的列表会非常长, 为了方便用户选择, 按器件的大类 (Family)、系列 (Variant) 和目标设备的具体型号进行分类, 请在 Family 选择 ARM;

(5) Variant (目标设备所属的系列), 选择 Tiva C Series;

(6) 目标设备的型号 (指开发板上 MCU 的型号), 选择 Tiva TM4C123GH6PM;

(7) 目标设备与计算机的联接方式 (Connection) 选择 Stellaris In-Circuit Debug Interface。

(8) 选择空工程（带 main.c），然后单击 Finish 完成新工程的创建。

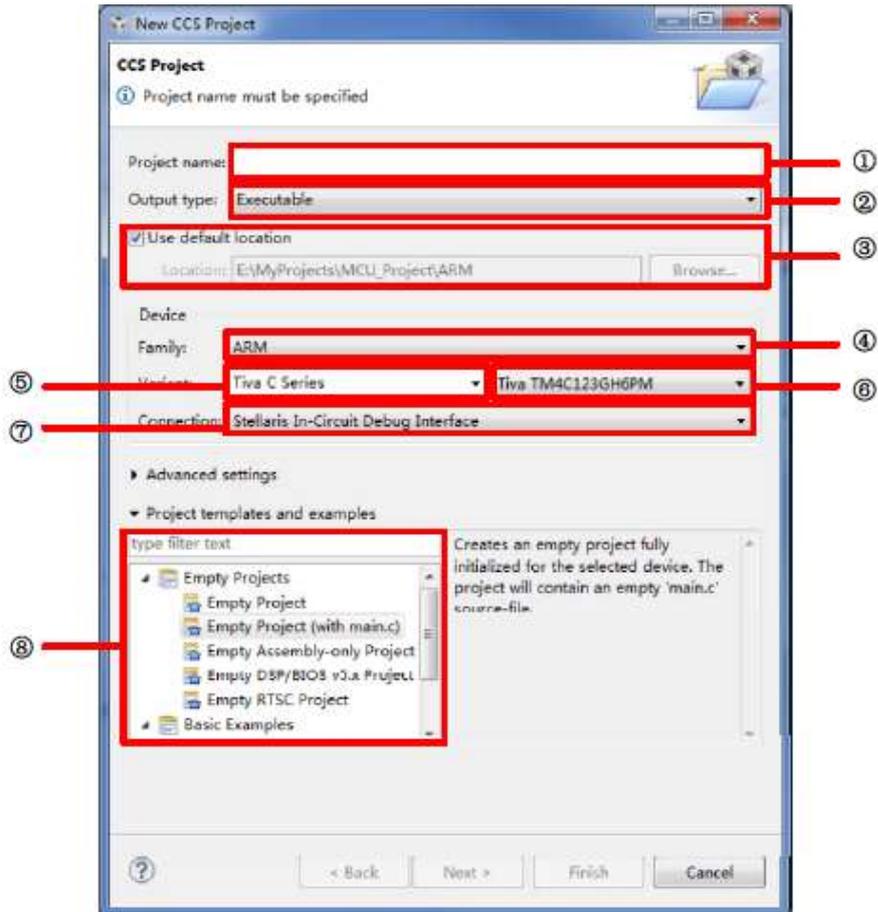


图 4.2 项目配置

工程创建后 CCS 进入如图 4-3 所示窗口。窗口左侧显示为项目的文件组成，右侧为文件内容。在工程上右键单击，选择 Properties，出现如图 4-4 界面。选择 Build->ARMCompiler->Include Options。

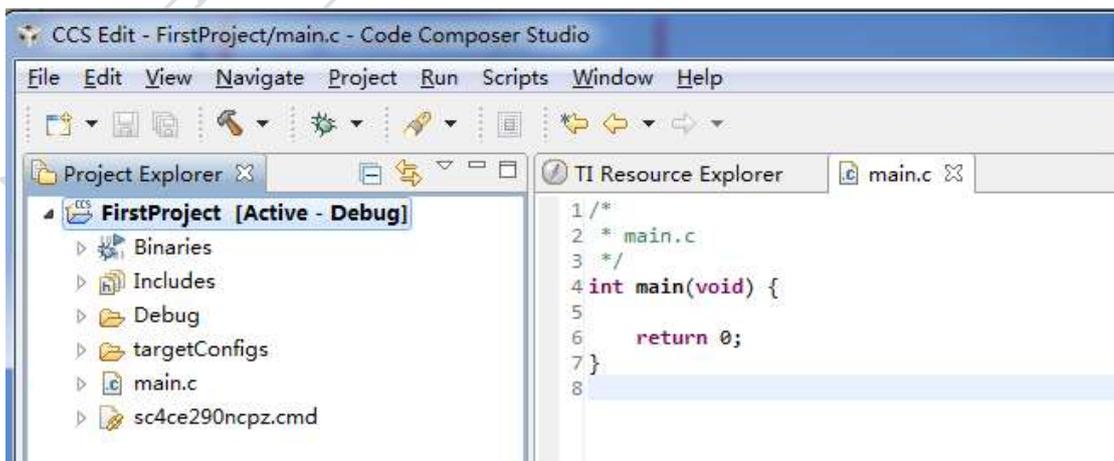


图 4.3 初始工程

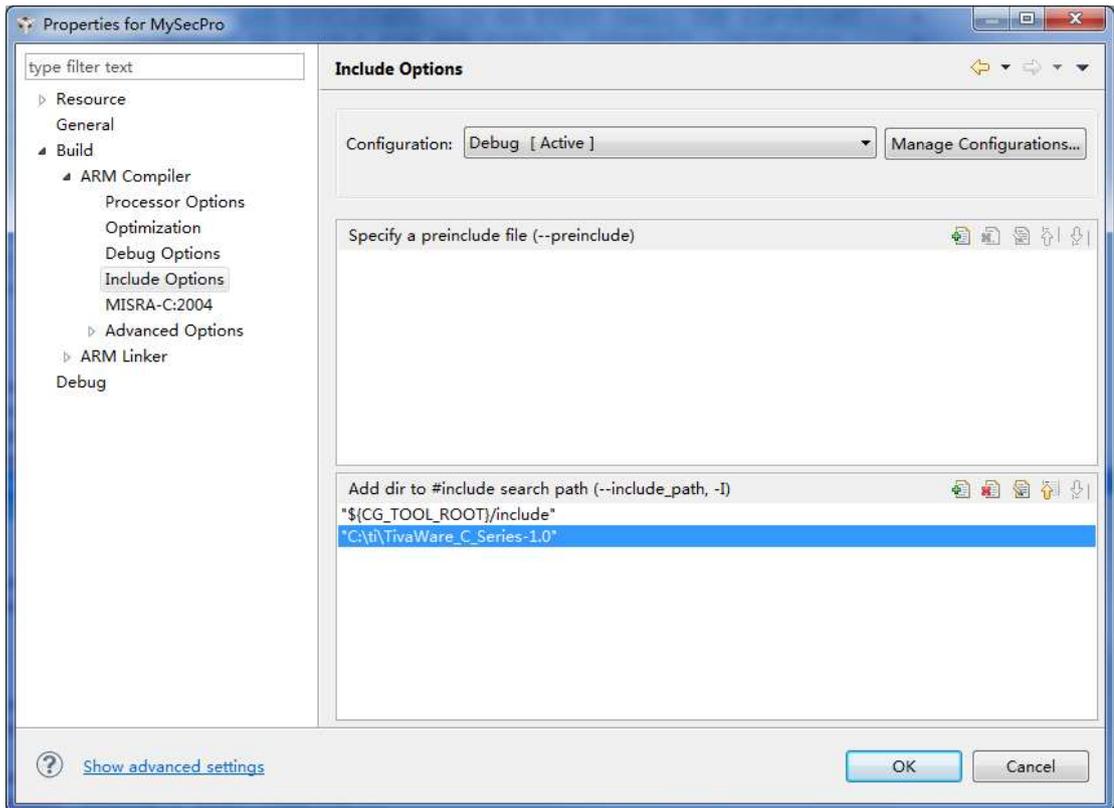


图 4.4 添加 include 路径

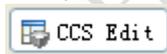
因为开发过程中会包含各种不同的头文件，这些头文件在用户的机器在 `Add dir to #include search path` 中添加路径 `"C:\ti\TivaWare_C_Series-1.0"`，（注意：您安装 TivaWare 的路径有可能与示例的不同，请按您的安装路径来设置），该文件夹内中包含了开发所需的头文件。

至此，初始工程建立完毕。

4.2 CCS 常用按钮

4.2.1 编译界面常用按钮

图 4.1 为构建一个新的项目或者导入一个已经存在的项目时的编辑界面，此时，



图标处在激活状态（底色为白色）。

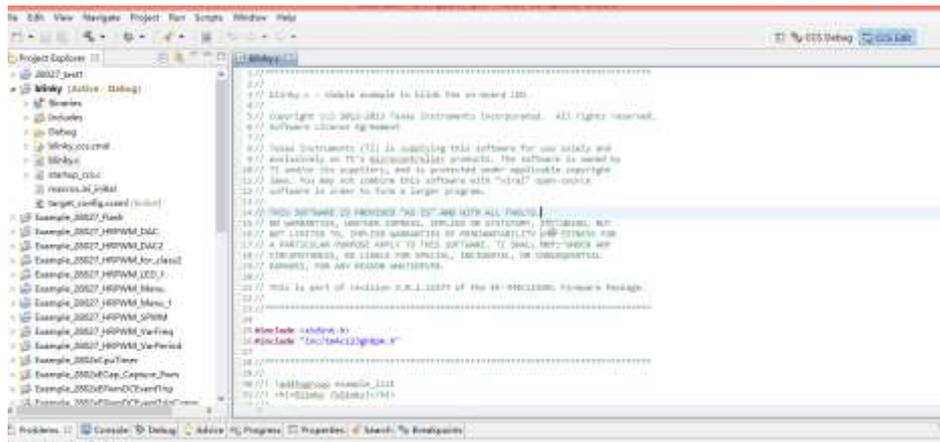


图 4.1 项目编辑页面

表 4.1 为 CCSv5 编辑状态下的常用按钮。

表 4.1 CCSv5 编译状态下的常用按钮

图标	名称	说明
	Build 'Debug'	根据实际使用的情况，该按钮对自建项目有效，其下拉菜单有两个可选项： Debug 允许对自建项目进行编译； Release 将自建项目释放成配置前的状态，此时编译将不能通过。
	Debug	编译调试按钮，该按钮在项目激活时有效，其下拉菜单有历次编译记录，可激活选定的项目并进行编译。
	Search	搜索按钮，其下拉菜单有 3 种搜索方式： File Search 文件搜索； c\c++ Search c\c++ 搜索； Git Search Git 搜索
	Back to ...	记录通过项目管理器（Project Explorer）导入的项目（或文件），按该按钮时，将根据后进先出的顺序激活对应的项目
	Last Edit Location	记录通过  按钮依次导入的项目或文件，按该按钮时，将根据后进先出的顺序激活对应的项目。当最后一个被激活后，该按钮关闭
	CCS Debug	切换到当前的调试界面
	CCS Edit	切换到当前的编辑界面

4.2.2 调试界面常用按钮

在编译状态下，按调试按钮 ，当编译链接全部通过后，可进入图 4.2 所示的项目调试界面。上部椭圆形红框内为调试状态下的常用按钮，下部椭圆形红框标注的阴影条左

端有一个箭头表示程序将从这一行开始运行。

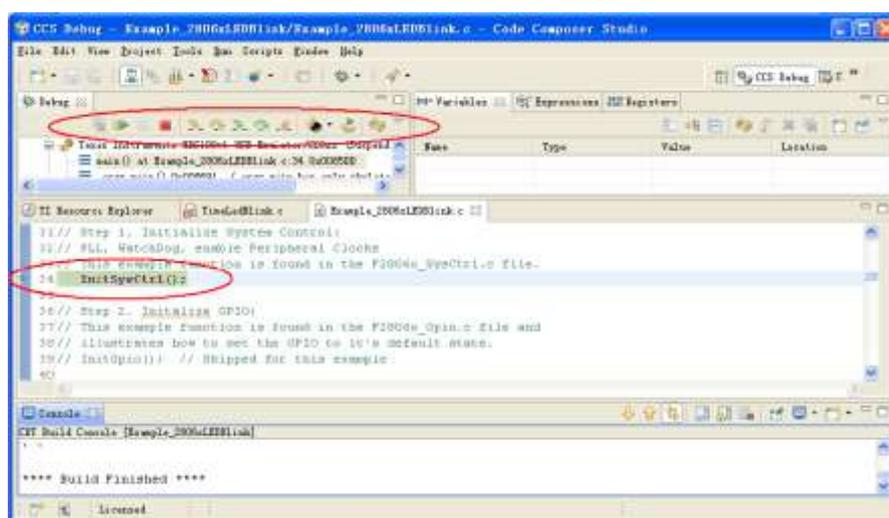


图 4.2 项目的调试界面

表 4.2 为调试界面的常用按钮。

表 4.2 CCSv5 调试界面的常用按钮

图标	名称	说明
	Resume	运行按钮。
	Suspend	暂停按钮。
	Terminate	结束按钮
	Step Into	步入按钮
	Step Over	步出按钮
	Assembly Step Into	汇编步入按钮
	Assembly Step Over	汇编步出按钮
	Step Return	单步返回按钮
	Reset CPU or Rese Emulatort	复位 CPU 或者复位仿真机按钮
	Restart	重新开始
	Refresh	刷新按钮
	Real Time Mode	实时模式按钮

	Continuous Reflash	连续刷新按钮
	Load	下载按钮
	Show logical structure	逻辑结构显示按钮
	Collapse all	全部折叠
	Open new View	打开新的观察窗
	Pin to debug Context	
	Open Perspective	打开透视图
	CCS edit Perspective	CCS 编辑透视图
	disconnect hardware	断开硬件

第五章 TIVWARE的使用

本章将会介绍 TIVWARE 的使用，即通过 CCS 导入 TIVWARE 工程，同时可以验证 ICD1 驱动程序等是否已安装完成。

5.1 导入 TIVWARE

TIVWARE 中包含了 TI 官方的实力程序及相关驱动库，通过使用这些驱动库和学习相关例程，将大大加快开发的步骤。导入 TIVWARE 的流程如下：

1. 打开 CCS，选择菜单栏 View—>TI Resource Explorer，即可打开资源页面，如图 5.1；

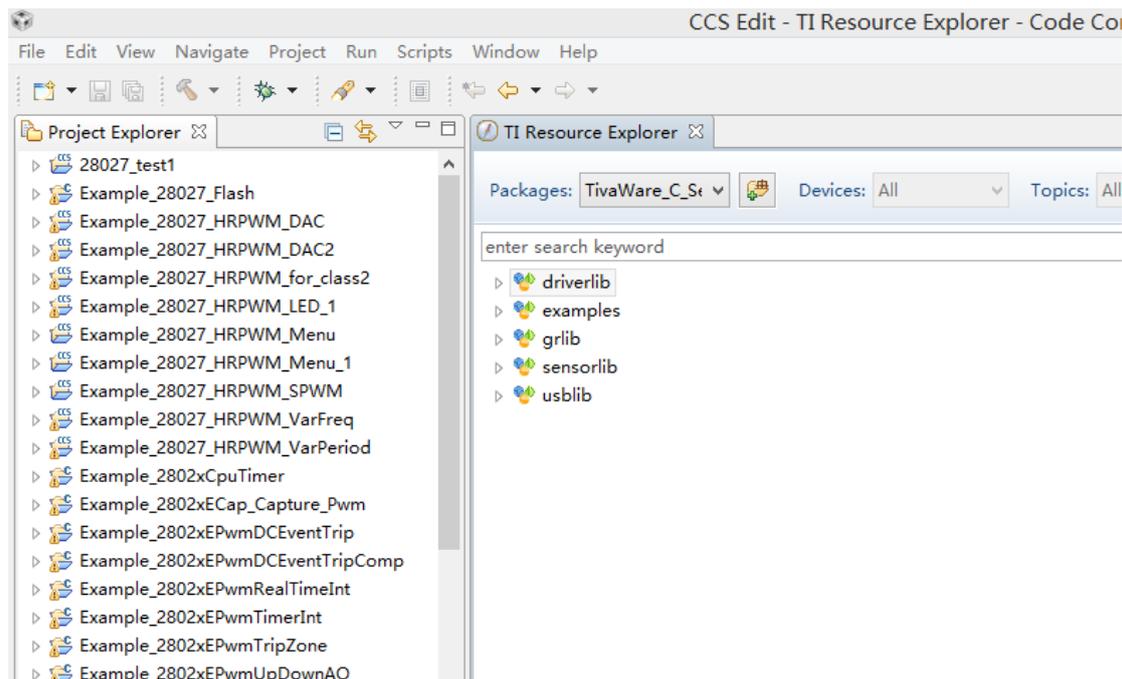


图 5.1 TI Resource Explorer

2. 初次使用 TIVWARE 需要首先导入安装的 TIVWARE，点击 explorer 中 packages 右边的添加按钮，如图 5.2；

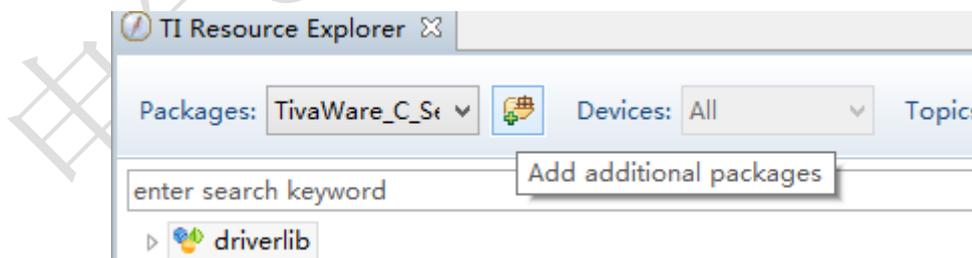


图 5.2 添加 packages

3. 在点击添加后，在弹出的 package configuration 中添加安装好的 tivaware，选取其根目录即可，如图 5.3；

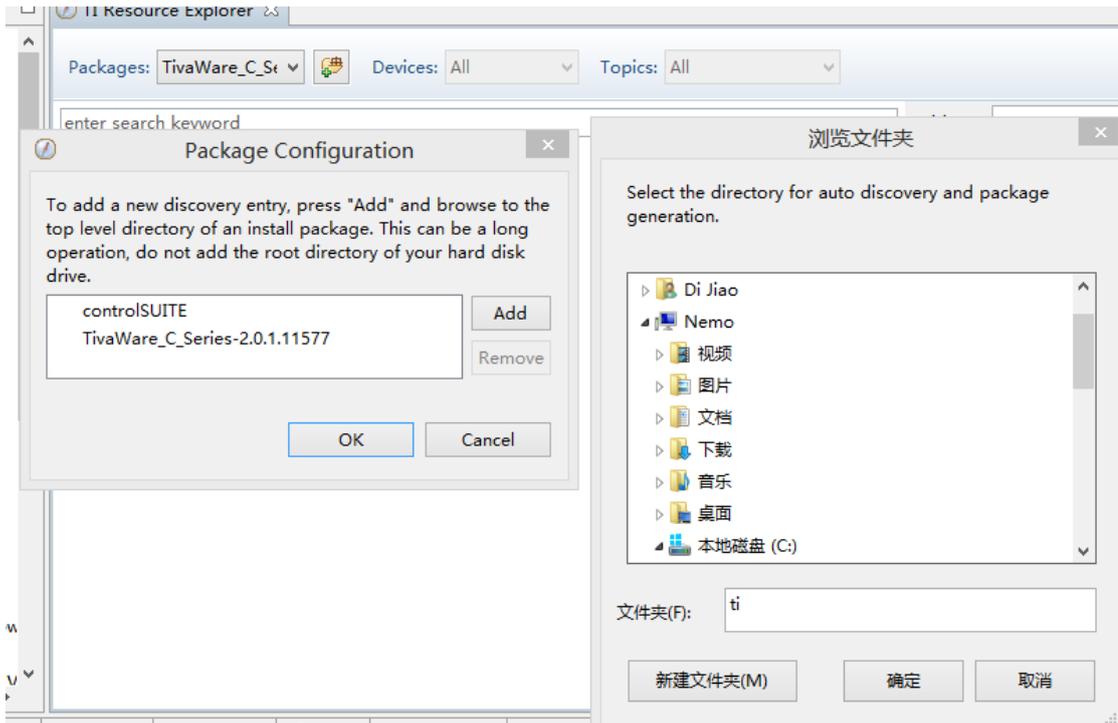


图 5.3 添加 packages

4. 添加后可打开 tivaware，可观察其包含的资源，如图 5.4;

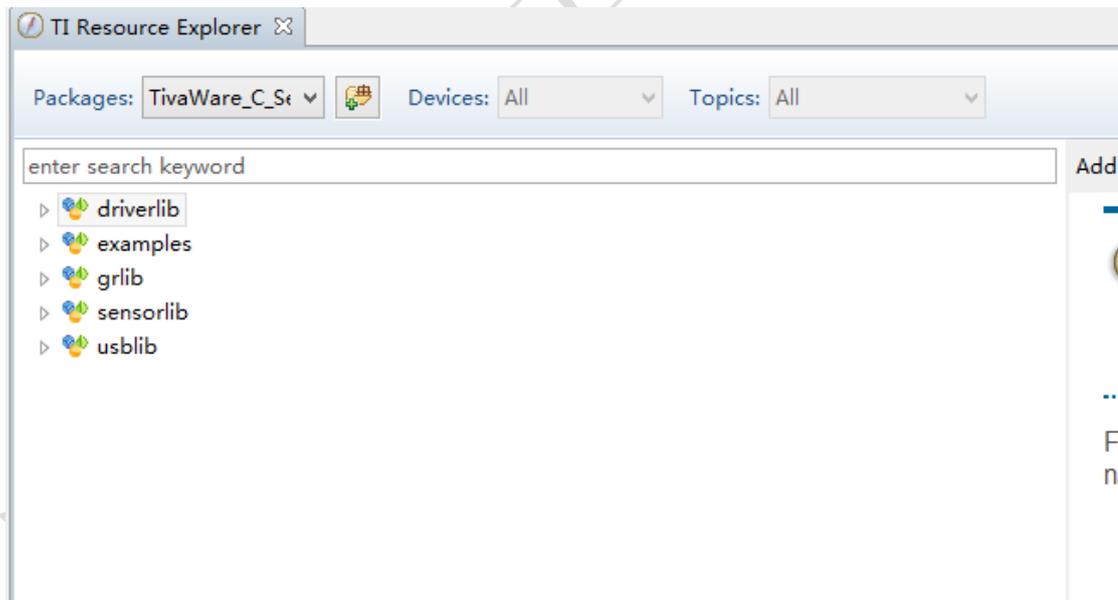


图 5.4 tivaware 资源

5. 至此，tivaware 导入完毕。

5.2 导入示例工程

这里我们以导入 tivaware 工程中的示例程序为例，流程如下：

1. 打开 TI Resource Explorer 中 tivaware，选择列表中的 example，如图 5.5;

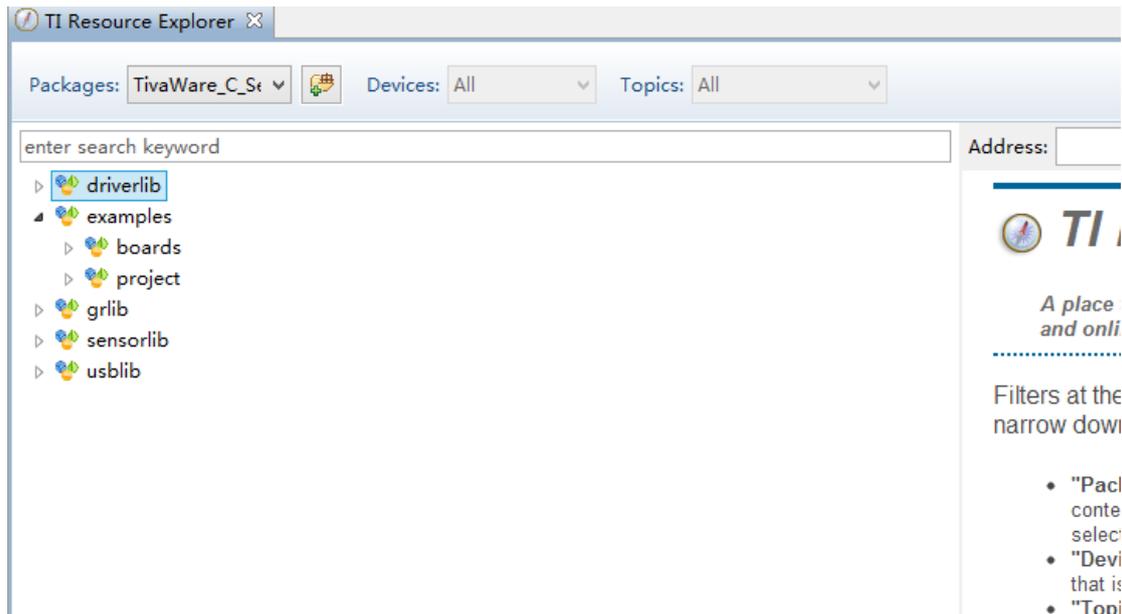


图 5.5 tivaware 资源

2. 选择 examples→boards→ek-tm4c123gxl→blinky, 如图 5.6;

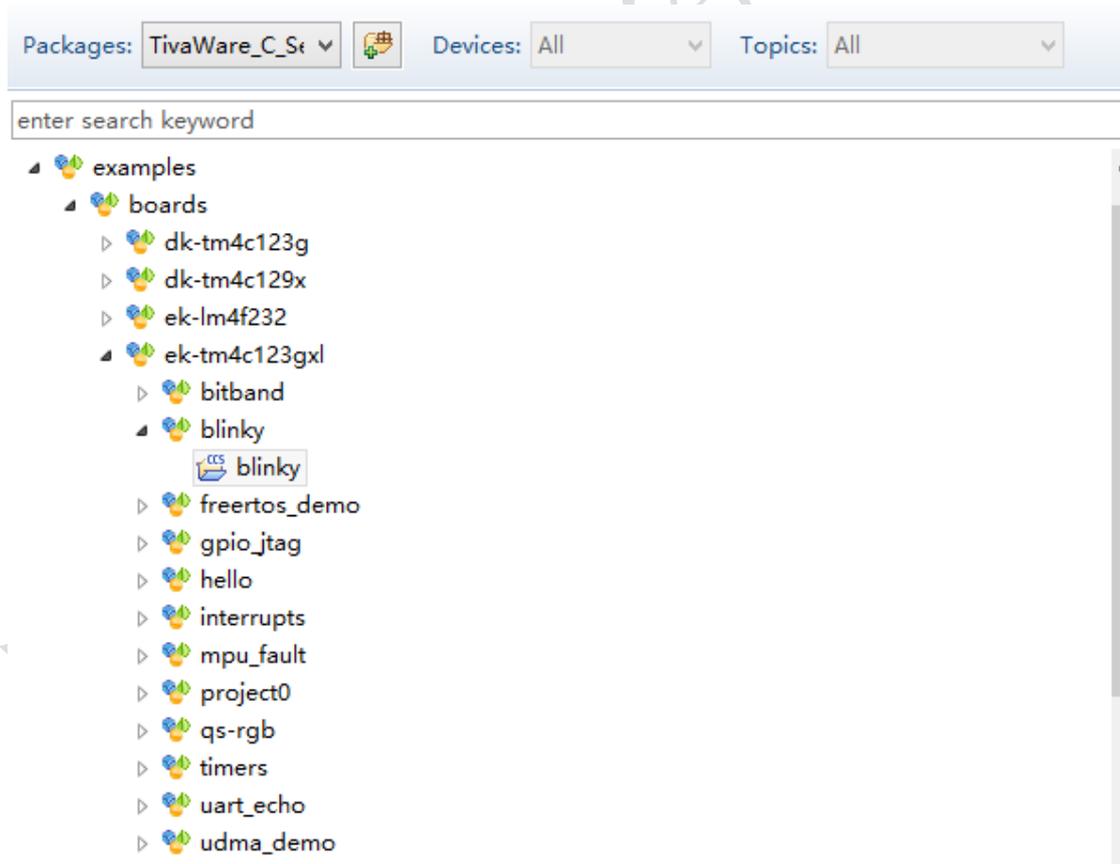


图 5.6 tivaware 资源

3. 此时，可以看到 blinky 工程已出现在 explorer 右侧，如图 5.7;

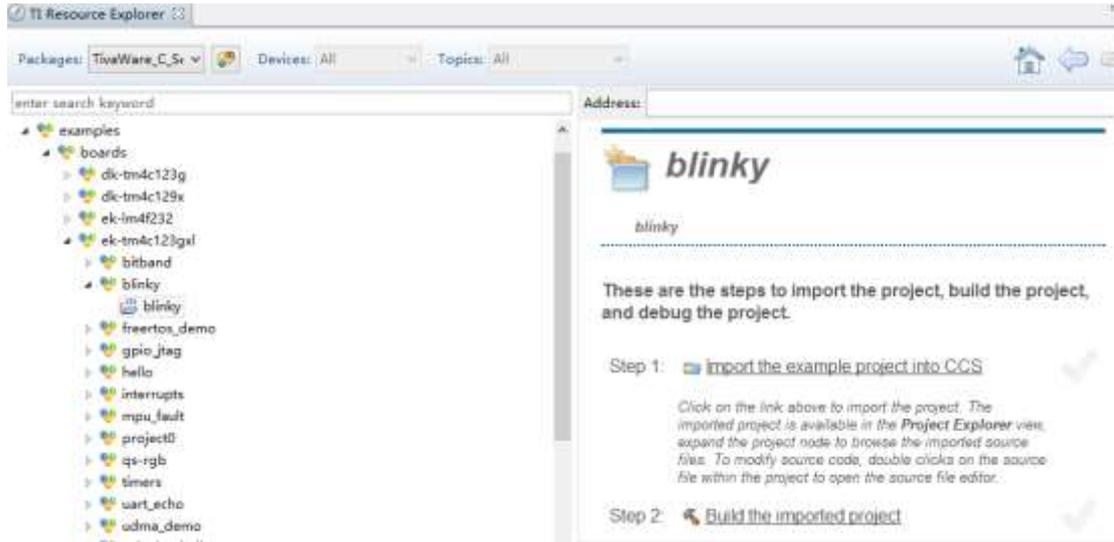


图 5.7 导入工程

4. 选择 import the example project into CCS，即可完成工程的导入，此时，在 CCS 界面 project explorer 中可以看到 blinky 工程已经存在，如图 5.8。

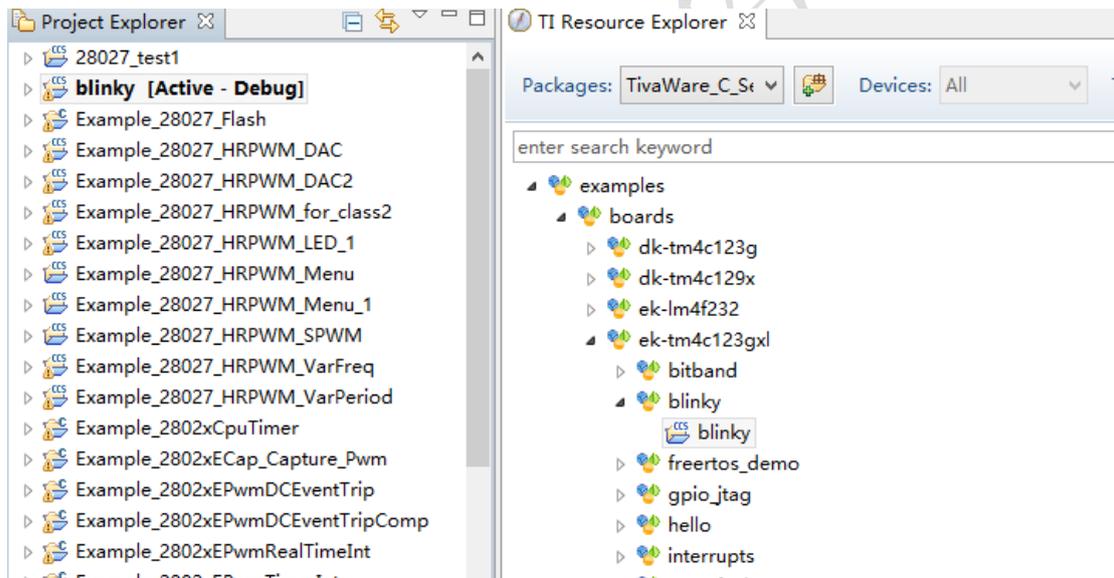


图 5.8 导入工程

5.3 运行导入的示例工程

这里我们仍以导入的工程 blinky 为例，流程如下；

1. 选择 blinky 工程，使其处于 active-debug 状态，如图 5.9；

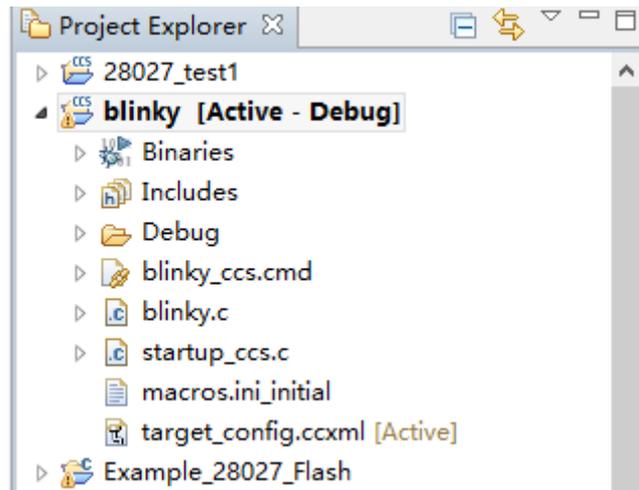


图 5.9 运行 blinky 工程

2. 选择工具栏  debug 按钮，程序即可烧入开发板，此时，程序尚未运行，程序指向 main 中第一行语句，如图 5.10；

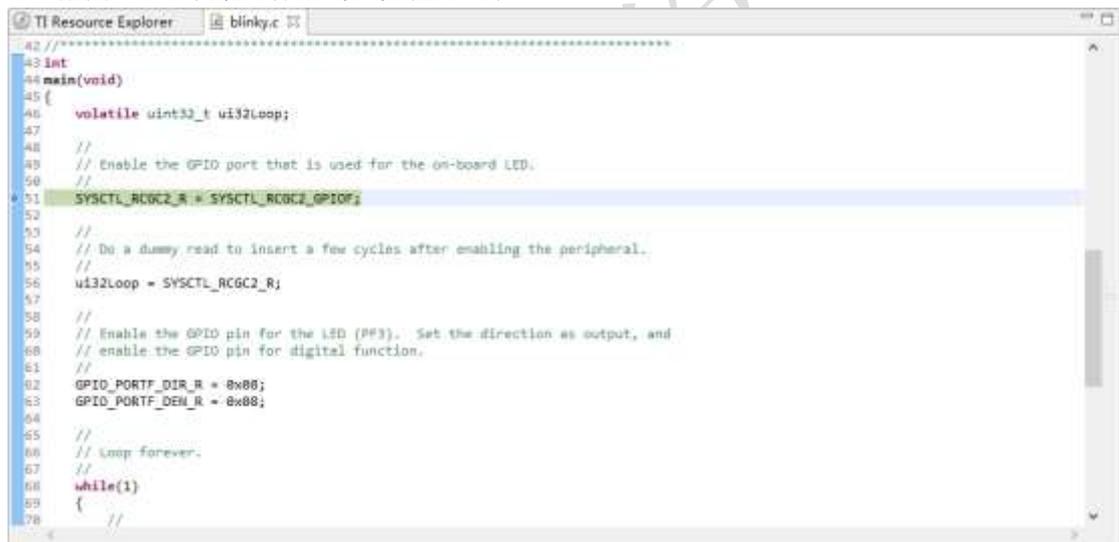


图 5.10 运行 blinky 工程

3. 选择 debug 下  中第一个按钮，即运行，程序即可正常运行；
4. 此时可以观察到开发板 D1 灯快速闪烁，如图 5.11.

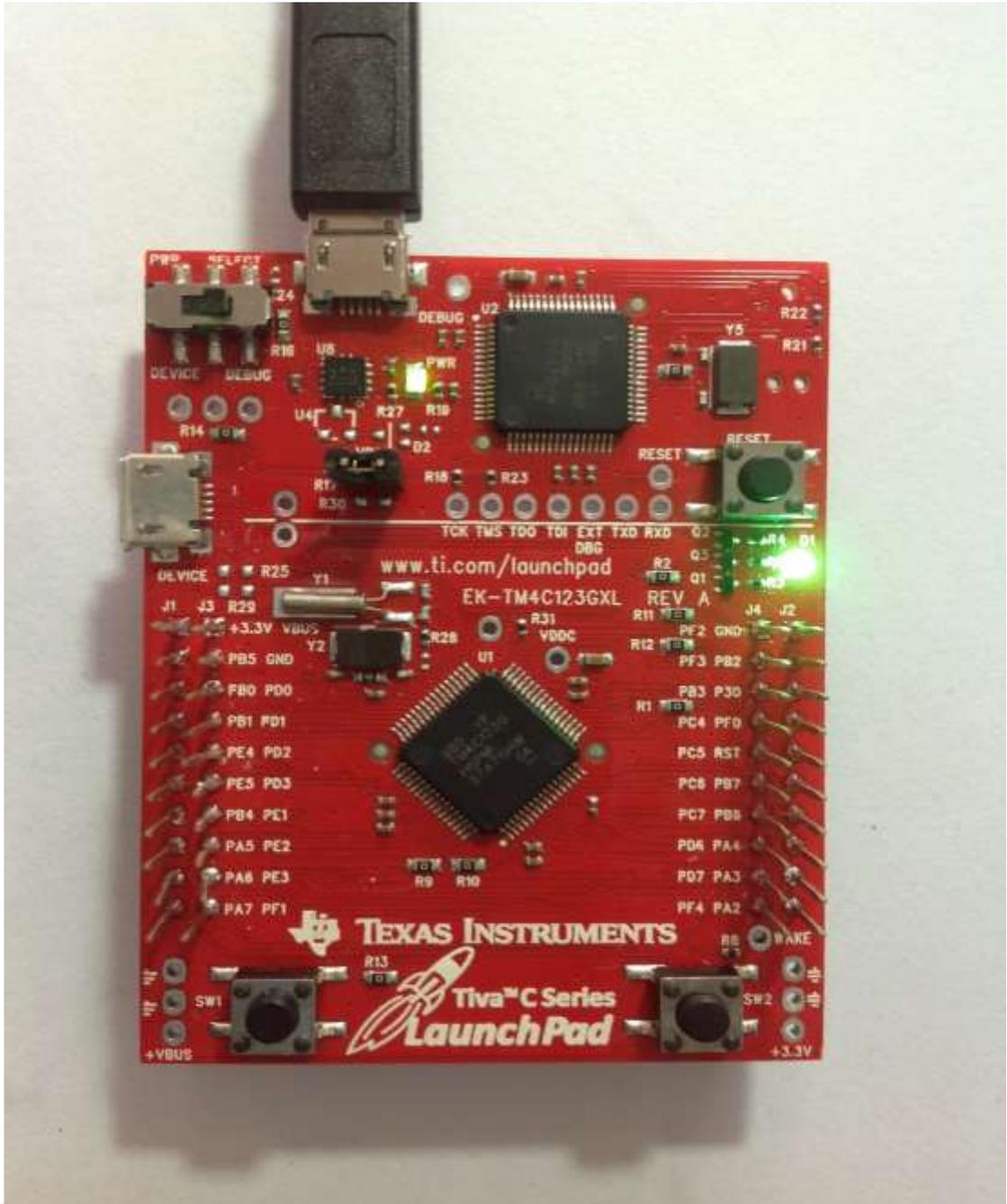


图 5.11 运行 blinky 工程