



SD-FSL-KL25-EVB

用 户 手 册

Rev3.5, KDS 开发环境, 2014.11.10

苏州大学飞思卡尔嵌入式中心

<http://sumcu.suda.edu.cn>

目 录

第 1 章 概述.....	1
1.1 ARM Cortex-M0+及飞思卡尔 Kinetis L 系列概述	1
1.2 Kinetis L 系列 MCU 突出应用特点	1
1.3 SD-FSL-KL25-EVB 简介	1
1.3.1 SD-FSL-KL25-EVB 功能概述.....	1
1.3.2 SD-FSL-KL25-EVB 硬件测试方法.....	2
1.3.3 配套书籍.....	2
1.3.4 网上光盘下载地址与网上光盘目录结构	3
1.3.5 慕课—中电网	3
第 2 章 SD-FSL-KL25-EVB 硬件资源.....	4
2.1 SD-FSL-KL25-EVB 套件硬件实物图.....	4
2.2 SD-FSL-KL25-EVB 板上引出接口.....	5
第 3 章 开发工具的安装与配置.....	6
3.1 所需的工具软件清单.....	6
3.2 软件安装过程.....	6
3.3 KDS 开发环境的配置.....	8
3.3.1 在 KDS 中配置 USBDM 写入调试器.....	8
3.3.2 增加或更改文件的查看方式	8
3.3.3 更改字体属性.....	10
3.3.4 更改鼠标悬停提醒设置	10
第 4 章 KDS 简明使用方法.....	11
4.1 导入现有工程到开发环境中.....	11
4.2 编译工程.....	12
4.3 程序的运行与调试.....	12
4.3.1 使用 KDS 进行烧录与调试.....	12
4.3.2 使用 USBDM 独立写入软件进行烧录.....	13
4.4 常用操作.....	15
第 5 章 Q&A.....	17
附录 1 《SD-FSL-KL25-EVB》简明测试方法.....	21
附录 2 《SD-FSL-KL25-实验箱》简明测试方法.....	23

第1章 概述

1.1 ARM Cortex-M0+及飞思卡尔Kinetis L 系列概述

- 2012 年 3 月 14 日，ARM 公司于中国上海发布了一款拥有全球最低功耗的微处理器 ARM Cortex-M0+。M0+内核的性能是最接近 8 位或 16 位竞争产品 CoreMark/mA 的 2 倍。
- 与 Cortex-M4 等系列处理器具有良好的软件兼容性。以 Cortex-M4 为基础的 Kinetis 系列 MCU，选择丰富，为软件复用提供基础。如 E 系列面向 5V 应用、W 系列面向传感网应用、V 系列面向电机控制、M 系列面向仪表（有 24 位快速 A/D）等，若用过其中一种，积累了构件，软件复用变得容易。
- 飞思卡尔的 Kinetis L 系列 MCU 于 2012 年 6 月提供样片，2013 年正式上市。该系列 MCU 是业内首款基于 ARM Cortex-M0+内核的 MCU，具有超低功耗、应用设计方便、扩展性好、系列品种齐全等特点。目标市场是传统 8 位 MCU 应用领域的 32 位升级换代。Kinetis L 系列 MCU 和基于 ARM Cortex-M4 内核的 Kinetis K 系列 MCU 完全兼容。
- 飞思卡尔的 Kinetis L 系列 MCU 内核单周期访问内存速度可达 1.77 CoreMark/MHz，CPU 工作频率最大可支持 48MHz。
- 飞思卡尔的 Kinetis L 系列工作电压为 1.71V~3.6V；运行温度范围为-40℃~105℃；具有 64B 的 Cache；具有 USB OTG、定时器、DMA、UART、SPI、IIC、TSI、16 位 ADC、12 位 DAC 等模块。

1.2 Kinetis L系列MCU突出应用特点

Kinetis L 系列 MCU 的突出应用特点主要有：

- 电源冗余与平衡分布，提高了稳定性；引脚冗余分布，有利于硬件电路设计；
- 内含温度传感器用于测定芯片温度；
- 内含 16 位 A/D 转换器与 12 位 D/A 转换器；
- 内部 Flash 的在线编程解决了过去需要移入 RAM 问题，同时允许中断，应用上可以取代 EEPROM 存储参数；
- 内部 RTC 及独立电池供电引脚，可以取代外接时钟芯片；
- 超低功耗：内核可达 nA 级别的超低功耗；
- 可通过编程使得部分引脚具有 18mA 的驱动能力（一般为 5mA 驱动能力）；
- 宽电压范围：1.71V~3.6V；运行温度范围：-40℃~105℃
- 与 KW01-Zigbee 芯片为同一内核（ARM Cortex-M0+），有利于程序复用；
- 可支持轻量级 MQX 实时操作系统
- Kinetis L 系列 MCU 内核中统一的 SysTick，为基本定时与操作系统应用提供了便利。

1.3 SD-FSL-KL25-EVB简介

1.3.1 SD-FSL-KL25-EVB功能概述

SD-FSL-KL25-EVB 是飞思卡尔 Kinetis 微控制器 KL25(MKL25Z128VLK4)的评估板。

板载 USBDM 写入调试器，与《嵌入式技术基础与实践—ARM Cortex-M0+ Kinetis L 系列微控制器》一书配套。可完成 GPIO、串口、Systick、定时器、16 位 A/D、12 位 D/A、I2C、SPI、Flash、USB 等模块评估。与配合 SD-ExtBoard-D 扩展板，还可完成 LED、LCD、键盘、传感器、WSN 等实验，提供了丰富的软件构件，是学习 ARM Cortex-M0+ 的理想工具。

与《嵌入式实时操作系统 MQX 应用开发技术—ARM Cortex-M 微处理器》配合，可学习轻量级 MQX。

1.3.2 SD-FSL-KL25-EVB 硬件测试方法

附录 1 给出了《SD-FSL-KL25-EVB》简明测试方法；附录 2 给出了《SD-FSL-KL25-实验箱》简明测试方法。

1.3.3 配套书籍

配套书籍：王宜怀等著，嵌入式技术基础与实践（第 3 版）-ARM Cortex-M0+ Kinetis L 系列微控制器，清华大学出版社，2013 年 8 月。（普通高等教育“十一五”国家级规划教材），以下简称该书为“王宜怀 M0+书”。

1. 线条

通用知识—模块功能概要—芯片编程结构—基本编程方法—构件封装及实现—测试实例

2. 体系

- （1）符合软件工程可复用、可移植的原则，按照构件化封装要求编制底层驱动程序
- （2）实现工程结构的统一
- （3）直接可测试的工程例

3. 封面图片

封面图片

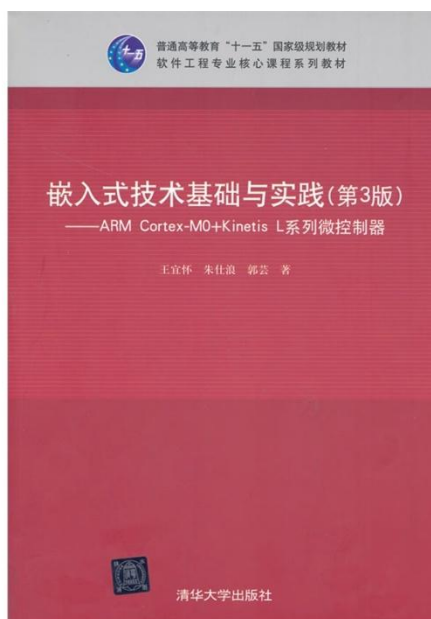


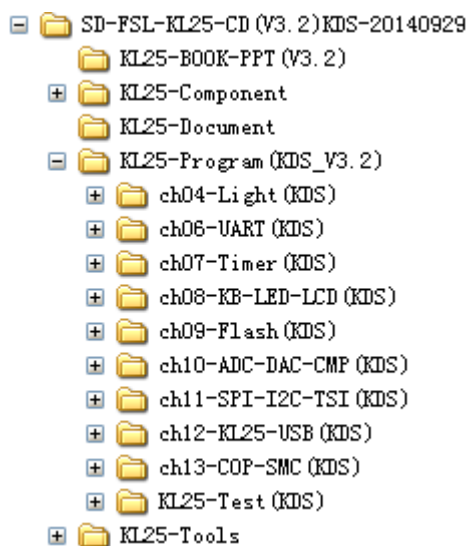
图 1-1 “王宜怀 M0+书”封面

1.3.4 网上光盘下载地址与网上光盘目录结构

网上光盘下载地址:

<http://sumcu.suda.edu.cn> “ **教学与培训** ” → “ ARM Cortex-M0+ ” → “SD-FSL-KL25-EVB” 下载。其内容会适时更新, 敬请关注。

网上光盘目录结构:



1.3.5 慕课一中电网

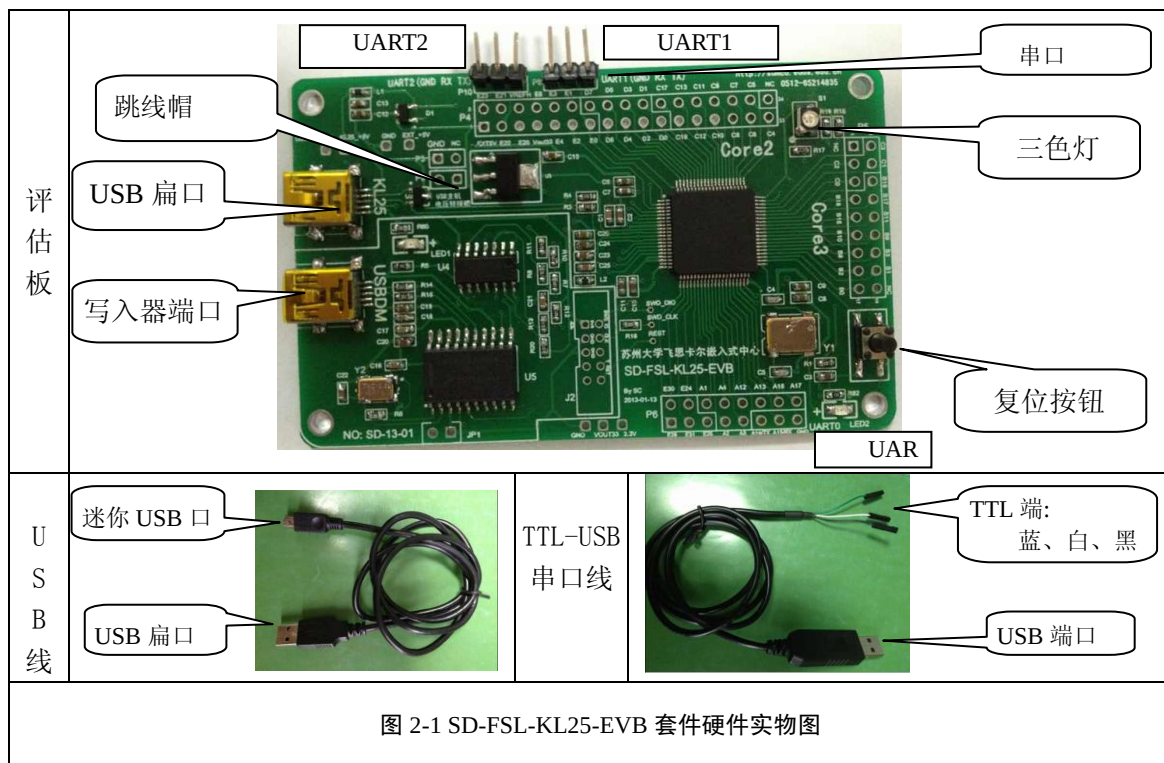
在中电网上录制了“慕课”。

http://support.eccn.com/training-manu_102.html

第2章 SD-FSL-KL25-EVB硬件资源

2.1 SD-FSL-KL25-EVB套件硬件实物图

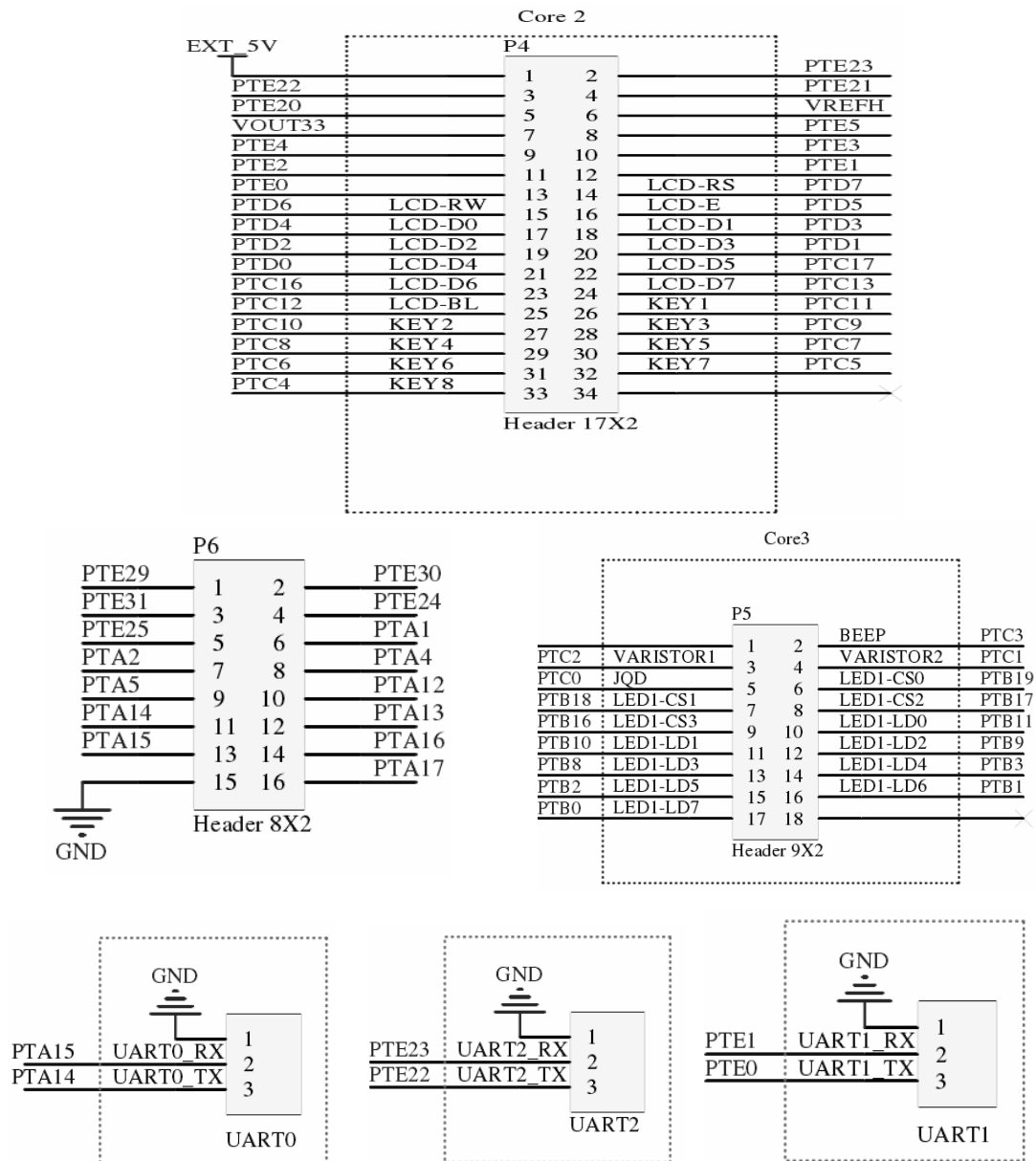
SD-FSL-KL25-EVB 开发板采用 ARM Cortex-M0+架构的 Kinetis L 系列 80 引脚 LQFP 封装的 MKL25Z128VLK4 芯片(简称 KL25)。SD-FSL-KL25-EVB 套件硬件实物图见图 2-1。



KL25 硬件最小系统引脚, 参见“王宜怀 M0+书”的 3.4 节, 引脚复用功能见该书附录 A, 原理图见该书附录 B。

2.2 SD-FSL-KL25-EVB板上引出接口

SD-FSL-KL25-EVB 板上引出接口见图 2-2。



另：三色灯，红灯：PTB19；绿灯：PTB18；蓝灯：PTB9，全为低电平点亮

图 2-2 SD-FSL-KL25-EVB 板上引出接口

第3章 开发工具的安装与配置

3.1 所需的工具软件清单

对 KL25 评估板进行测试评估所需的软件工具包括：

(1) 集成开发环境 KDS (Kinetis Design Studio)：KDS1.1.1.exe，具有编辑、编译、下载程序、调试等功能；

(2) USBDM 写入调试器驱动程序：USBDM_Drivers_1_2_0_Win_x32.msi (32 位操作系统下使用) 或 USBDM_Drivers_1_2_0_Win_x64.msi (64 位操作系统下使用)

(3) USBDM 写入调试器的独立写入软件：USBDM_4_10_6_190_Win.msi，具有独立写入功能，且可以挂接到 KDS 环境；

(4) TTL-USB 串口的驱动程序：PL2303_Prolific_DriverInstaller_v1.8.0.exe，用于 PC 机进行串行通信实验使用，硬件对应是“TTL-USB 串口线”。

(5) USB 驱动：SoochowUniversity-USBDevice，用于 PC 机与 KL25 开发板（作为从机）进行 USB 通信实验使用；将“USB 线”的“迷你 USB 口”端接入开发板的“USB 扁口”，WINDOWS 系统提示发现新硬件，指定驱动程序的安装目录 SD-FSL-KL25-CD (网上光盘)，具体安装过程在此不再赘述，详见 KL25 的 USB 程序中“SD-FSL-KL25-USB 使用说明.pdf”。

集成开发环境 KDS1.1.1.exe 可从飞思卡尔官网下载，其他四个软件在本板配套网上光盘的“..\KL25-Tools”文件夹下。

KDS 是飞思卡尔于 2014 年开始推出的面向 ARM Cortex-M 内核的 Kinetis 系列微控制器的嵌入式集成开发环境。在飞思卡尔官方网站上有专门的介绍页面，(http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=KDS_IDE&uc=true&lang_cd=en)，内含有 KDS 简介、最新资讯信息和下载链接。KDS v1.1.1 安装包大小为 500MB，支持操作系统有：Windows 7/8 (32 and 64-bit) 及 Linux(Ubuntu、Redhat、Centos)，在 WindowsXP 系统下也能安装，但可能会出现意想不到的问题，一般推荐使用 Windows 7 系统安装 KDS。

3.2 软件安装过程

1. 集成开发环境KDS的安装

运行 KDS 安装文件“KDS-v1.1.1.exe”，安装界面见图 3-1 所示：

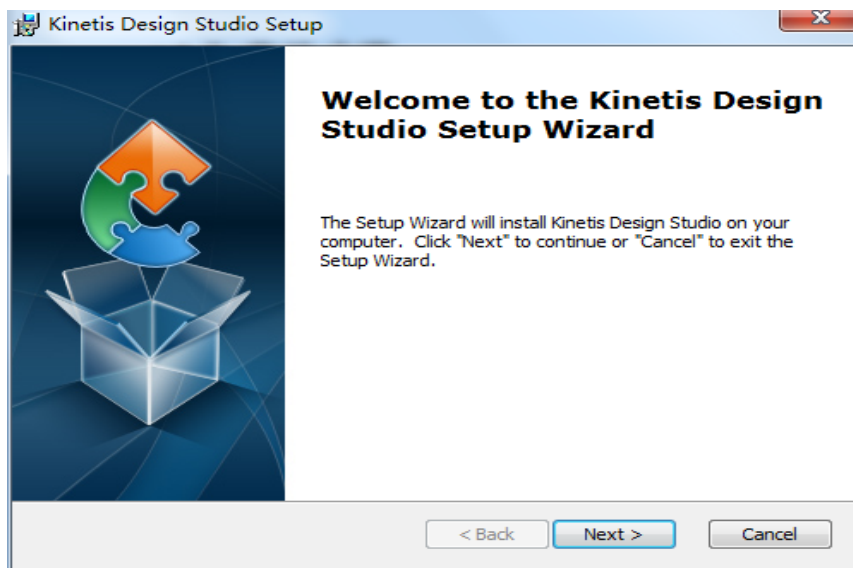


图 3-1 KDS 安装界面

选择 KDS 安装路径及调试端口：调试接口选择全部安装（默认设置），更改 KDS 目录点击 Browse。后续的安装只要根据提示进行操作即可。

KDS 安装路径及调试端口选择见图 3-2 所示：

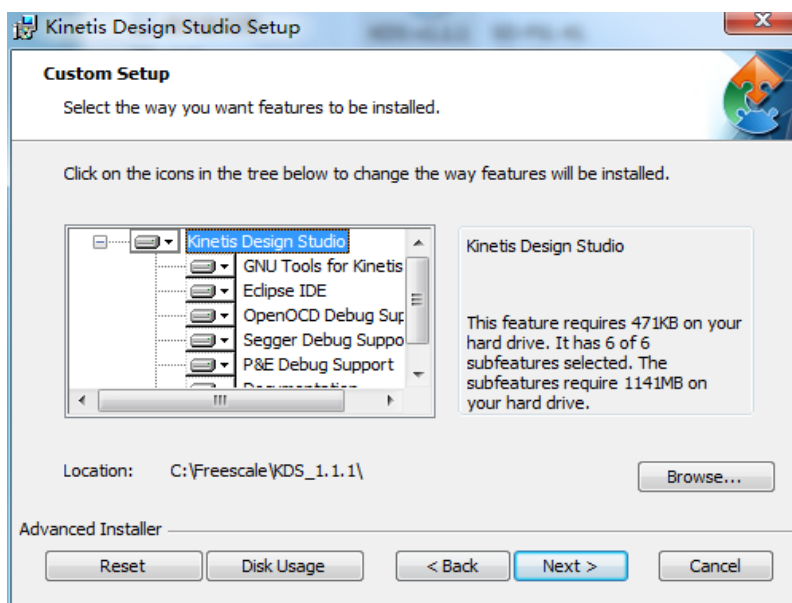


图 3-2 KDS 安装路径及调试端口选择

安装好后需要继续安装 USBDM 写入调试器驱动程序及 USBDM 写入调试器的独立写入软件。

2. USBDM驱动程序及USBDM独立写入软件的安装

在成功安装了集成开发环境后，再进行 USBDM 驱动程序及 USBDM 的独立写入软件安装，若之前已经安装过，需先卸载再安装。

USBDM 不是 KDS 默认支持的调试工具，但安装 USBDM_4_10_6_190 会向 KDS 添加 USBDM 的调试功能。USBDM 的下载地址为 <http://sourceforge.net/projects/usbdm/>。注意此驱动对应 32 位操作系统和 64 位操作系统，选择对应的进行安装。

此软件包安装后，最好重启计算机（依据不同的操作系统，有的可以不需重启）。正确的情况是：打开 KDS 环境后，菜单栏有“**USBDM**”菜单。

对安装驱动的备注说明：安装驱动程序前，外部设备还未连接到 PC 机上，手动点击安装驱动程序，此阶段一般是把安装的驱动文件信息拷贝到“C:\Program Files\”目录下，在 WIN7 系统拷贝到“C:\Program Files (X86)\”目录下。例如，对 USBDM 写入调试器驱动程序的安装，首先做的是点击驱动程序 USBDM_Drivers_1_2_0_Win_x32.msi（或 USBDM_Drivers_1_2_0_Win_x64.msi，64 位操作系统下使用），再安装 USBDM 编程器的独立写入软件 USBDM_4_10_6_190_Win.msi，之后，请重启电脑。其实这些安装都把相关安装信息装到 C 盘的“C:\Program Files\pgo\USBDM Drivers 1.2.0\Drivers\BDM_Driver”目录下（XP 系统），或“C:\Program Files (X86)\pgo\USBDM Drivers 1.0.1\Drivers\BDM_Driver”目录下（WIN7 系统）。

3. TTL-USB 串口驱动程序的安装

点击 PL2303_Prolific_DriverInstaller_v1.8.0.exe 驱动安装程序（此安装包为 32 位/64 位通用版本），安装过程不需要选择安装路径，点击下一步直到提示安装完成即可。安装完成后连接 TTL-USB 线，便可以在设备管理器的端口中看到 **Prolific USB-to-Serial Comm Port (COM3)**，右击该端口查看属性，在驱动程序选项卡下的驱动程序详细信息中可以看到 PL2303 的两个驱动程序位置。驱动程序文件位置见图 3-3 所示：

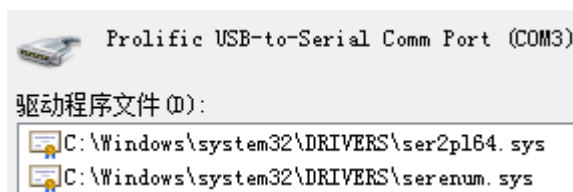


图 3-3 驱动程序文件位置图

3.3 KDS 开发环境的配置

3.3.1 在 KDS 中配置 USBDM 写入调试器

选择 KDS 菜单栏中的 USBDM->Configure，点击 C/C++ USBDM，在右方的 ARM Ltd GNU Tools for ARM 栏中，点击 Path 右侧的 Browse，找到 KDS 安装目录下 ../KDS_1.1.1/toolchain/bin 文件夹中的“arm-none-eabi-gcc.exe”文件，点击打开。

点开 C/C++ USBDM->ARM GDB Server，在 Target Device 中选择所要烧录的芯片类型（如：MKL25Z128M4）。将 USBDM 写入调试器与目标板连接，并使芯片上电，点击界面中的 Refresh，选择 USBDM-JS16-SWD-0001。其它选项保留默认值，点击 OK。

3.3.2 增加或更改文件的查看方式

KDS 在编译过程中默认建立 .elf 文件，如果需要增加 .hex 和 .map 文件，选中工程，点击菜单栏中 Project->Properties 进入工程属性设置，进入 C/C++ Build-->Settings-->Toolchains 选项卡，勾选 Create flash image 及 Create extended listing，点击 OK。此时工程在编译过程中就会在 DEBUG 中出现 .hex 和 .map 文件。具体操作见下列图所示：

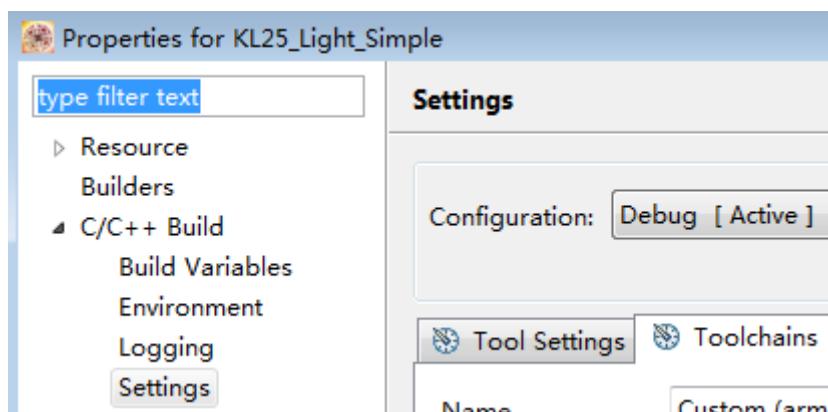


图 3-4 Toolchains 所在位置图

设置 Create flash image 及 Create extended listing，见图 3-5 所示：

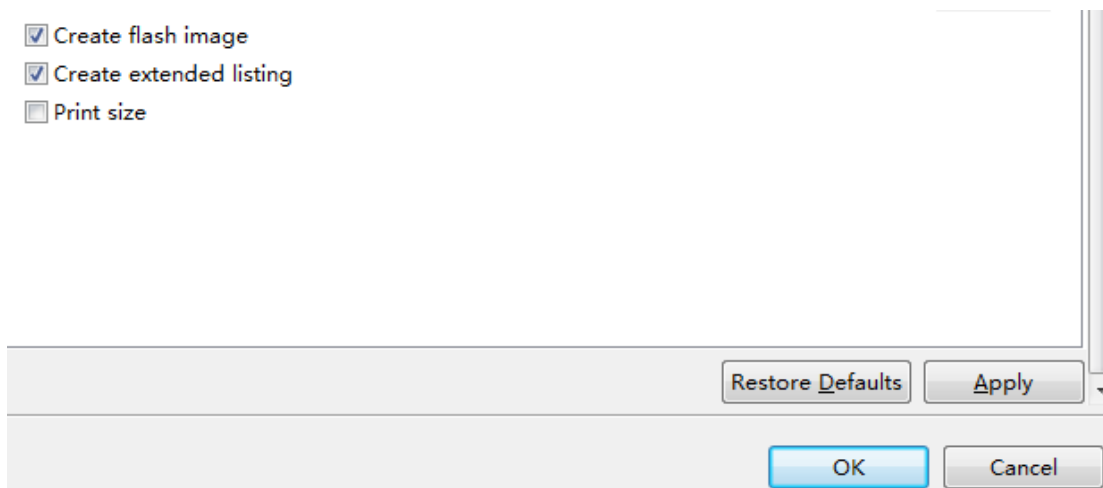


图 3-5 增加.hex 和.map 文件设置图

.hex 文件是由符合 Intel HEX 文件格式的文本构成的 ASCII 文本文件。具体知识见《嵌入式技术基础与实践（第三版）—ARM Cortex-M0+ Kinetis L 系列微控制器》中第 87 页。

.map 文件提供了查看程序、堆栈设置、全局变量、常量等存放的地址信息。具体知识见《嵌入式技术基础与实践（第三版）—ARM Cortex-M0+ Kinetis L 系列微控制器》中第 89 页。

KDS 中默认的文件打开方式为 C/C++ Editor，只能识别.c/h/s 等文件，对于工程中的.ld 等文件不能识别，因此需要更改 KDS 中对某种文件的打开方式，可以点击菜单栏中的 Window->Preferences，点击 General->Editors->File Associations，在右侧选项卡中的 File types 中选择需要更改的文件类型（若没有，可点击右侧的 Add 按钮手动添加），在下方 Associated editors 中会显示此类文件当前的打开方式，点击右侧的 Add 按钮添加新的打开方式，选中它后点击右侧的 Default 按钮可将其设置为默认打开方式，点击 OK 即可完成修改。

例如：需打开链接文件（.ld 文件），但是无法查看文件中代码，可以点击 Window->Preferences->General->Editors->File Associations，在右侧选项卡中的 File types 中选择*.ld 文件类型，在下方 Associated editors 中点击 Add，选择“C/C++ Editor”打开方式，点击 OK 即可查看文件中代码。

3.3.3 更改字体属性

若需改变文本编辑区文字的字体属性，可以点击菜单栏中的 Window->Preferences，点击 General->Appearance->Colors and Fonts，在右侧选项卡中的 Colors and Fonts 里选择 Basic->Text Font，点击 Edit 按钮即可进入字体属性界面。

设置字体属性见图 3-6 所示：

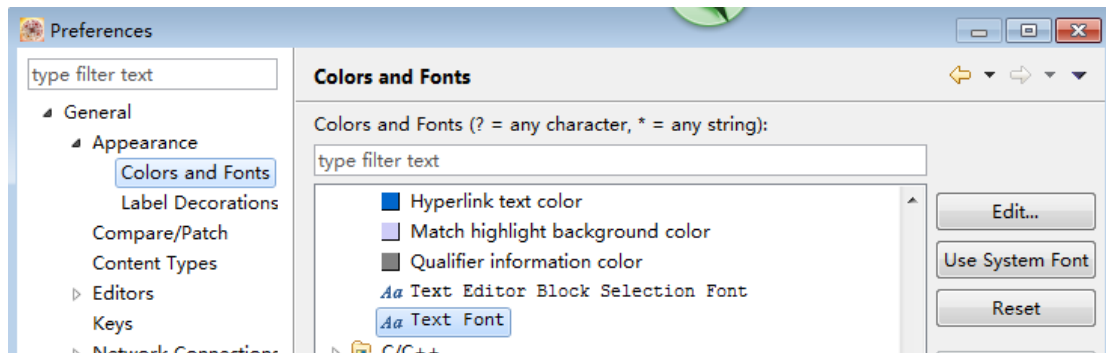


图 3-6：字体属性设置图

点击  即可修改文字属性。

3.3.4 更改鼠标悬停提醒设置

若选择或取消鼠标悬停提醒，可以点击菜单栏中的 Window->Preferences，点击 C/C++->Editors->Hovers，勾选或取消勾选右侧选项卡中 Combined Hover，点击 OK 即可完成更改。

注意：以上设置在不同的工程中可能需要重新设置，也就是说以上设置只存在于当前工程中，并不是设置了这些属性后，所有工程都存在了。

第4章 KDS简明使用方法

4.1 导入现有工程到开发环境中

(1) 什么是 KDS 工程目录？

如图 3-1 “KL25_Light_Component” 就是一个 KDS 工程目录。注：Debug 文件夹是工程进行构建后自动生成的文件夹。工程框架见图 4-1 所示：

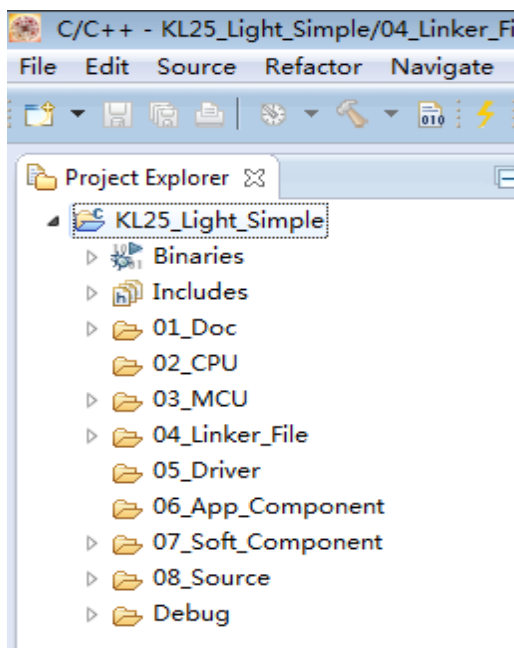


图 4-1 KDS 工程框架

(2) 什么是 KDS 工程面板

如图 3-2 所示，是 KDS 的工程面板。若没有，可以通过点击菜单栏中 Window->Show View->Project Explorer，显示出 KDS 的工程面板。具体操作见图 4-2 所示：

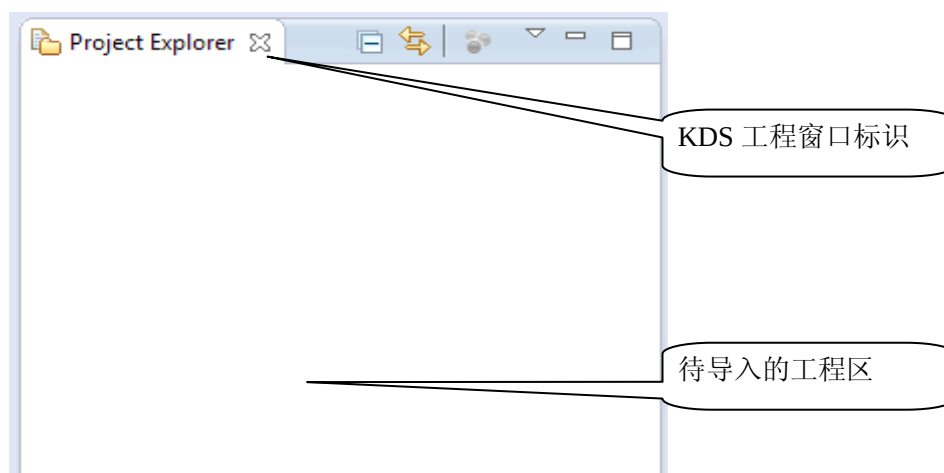


图 4-2 KDS 的工程面板

(3) 工程导入方法

在 KDS 中需要打开工程时，选择菜单栏中 **File->Import**，在弹出的对话框中选择 **General->Existing Projects into Workspace**，点击 **Next**，接着点选 **Select root directory**，点击右侧的 **Browse**，选择需要打开的工程文件夹，点击 **Finish** 完成，导入过程如图 4-3 所示：

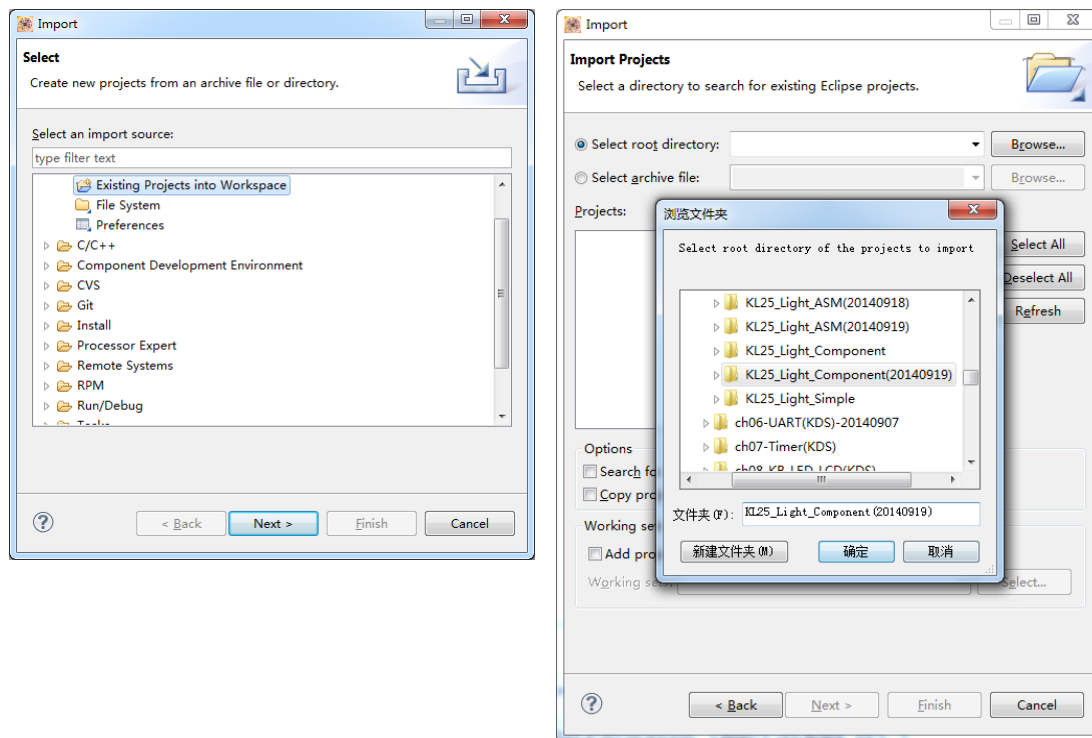


图 4-3 导入工程流程

4.2 编译工程

编译前，需在左侧的工程目录视图中点选需要编译的工程，然后在菜单栏中选择 **Project->Build Project** 或点击工具栏上的 图标，即可编译所选工程。

若需清理工程，则在菜单栏中选择 **Project->Clean** 即可。若需在清理工程后自动编译工程，则先需要将 **Project->Build Automatically** 的勾去掉，此时在清理工程后会自动对工程进行重新编译。建议使用此种方式对工程进行编译。

具体的编译过程信息可以查看下方的 **Console** 窗口。编译过程结束后可以在下方的 **Problems** 窗口中查看编译警告及错误信息。

在左侧工程栏的 **DEBUG** 中，工程默认编译生成“.elf”文件，如“ KL25_Light_Simple.elf - [arm/le] ”。若需要生成 hex 文件及 lst 文件，可以选中工程，点击菜单栏中 **Project->Properties** 进入工程属性设置，进入 **C/C++Build-->Settings-->Toolchains** 选项卡，勾选 **Create flash image** 及 **Create extended listing**，点击 **OK**。具体过程见目录“3.3.2 增加或更改文件的查看方式”。

若 **Problems** 中没有 **ERRORS**，则编译通过。可以进行程序的运行或调试。

4.3 程序的运行与调试

4.3.1 使用KDS进行烧录与调试

目前版本的 USBDM 插件只支持在 KDS 中进行调试，没有单独的写入选项。由于调试

过程也会把程序烧录入芯片中。故可以用调试代替烧录。

第一次调试：

点击工具栏中的“”调试图标的下拉箭头，选择“Debug Configurations”，双击“USBDM Hardware Debugging”新建一个以当前工程命名的 USBDM 写入调试器连接。若在 USBDM Hardware Debugging 下面已经有工程存在，需删除该工程，重新双击 USBDM Hardware Debugging，不然，写入芯片的仍然为原存在的调试程序。

以后内容一般情况在选择好调试工程后会默认，可以直接烧录。（在烧录配置界面中选择 main 选项卡，在 Project 中选择烧录文件所在工程，在 C/C++ Application 中选择需要烧录的文件（elf 或 hex），可以点击 Search Project（相对路径，只能选择 elf 文件）或 Browse（绝对路径，elf、hex 均可）来选取烧录文件。然后，进入烧录配置界面中的 Debugger 选项卡，在 Target Device 中选择目标芯片类型，在 BDM Selection 中选择所连接的 USBDM。）

在 Startup 选项卡中的 Runtime Options 中，可勾选 Set breakpoint at:来在程序中设置默认断点（默认断点位置为 main，即 main 函数的第一条语句）。

设置好后就可以点击 Debug 开始调试。首次启动调试时，KDS 会提示需要进入调试界面，勾选 Remember，点击 Yes 会进入调试界面。调试器会先将程序烧录入芯片中，然后运行程序并停止在所设置的默认断点处。若只想烧录程序而不进行调试，可点击菜单栏上的终止调试按钮来停止调试。此时可以点击屏幕右上角的按钮回到代码编辑界面。

程序调试界面如图 4-4 所示：

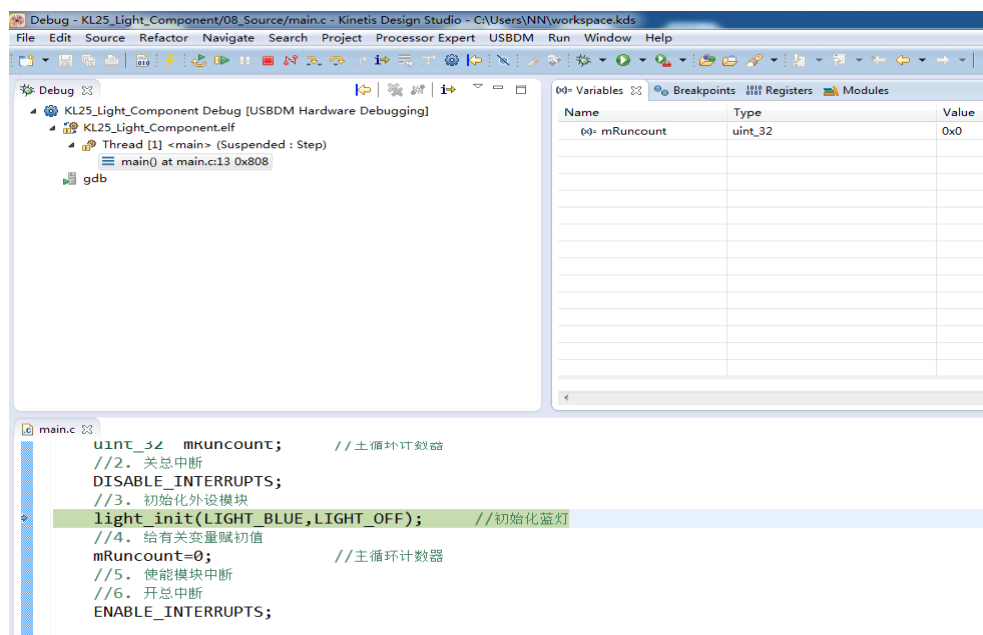
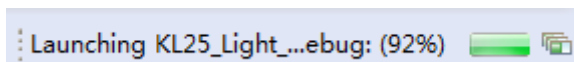


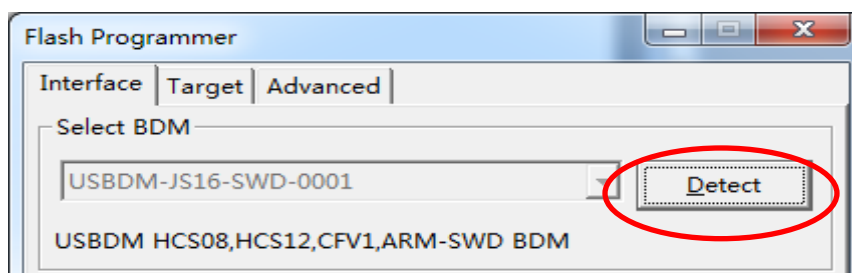
图 4-4 程序调试界面

注：启动调试时，界面右下角有进度条，可以查看启动调试的进度。

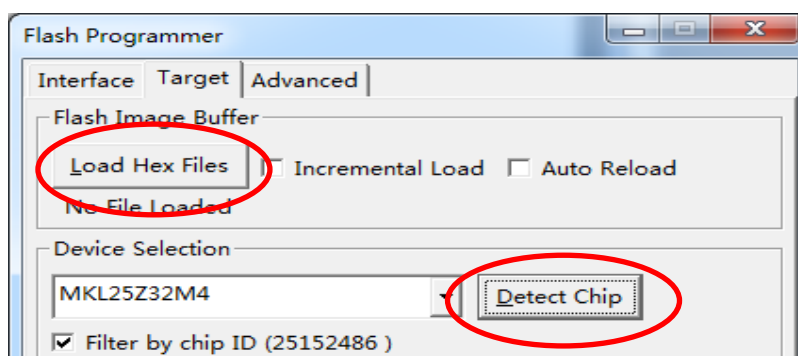


4.3.2 使用USBDM独立写入软件进行烧录

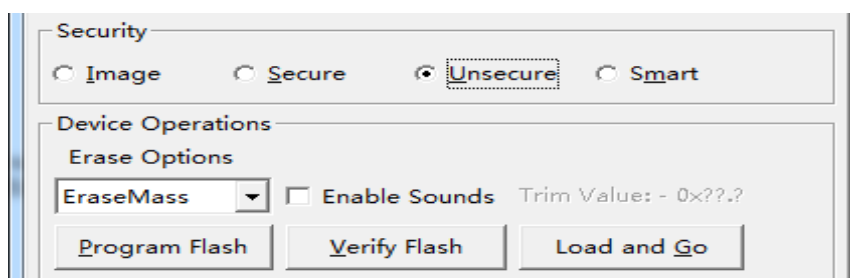
打开 ARM Programmer 后，在 Interface 选项卡中点击“Detect”按钮检测当前已连接 KL25 的 BDM 选项。



打开 Target 选项卡，首先点击“Detect Chip”按钮，检测 KL25 芯片类型，检测成功后，芯片类型下方原先是灰色的标签变成黑色可勾选状态，接着载入 elf 文件，点击“Load Hex Files”，选择要写入程序的 elf 文件，点击打开。



“Security”中选择“Unsecure”，“Erase Options”中选择“EraseMass”，最后点击“Program Flash”即可完成程序烧录，需要注意的是，本独立写入软件仅支持程序烧录，而不支持程序在线调试，但在不需要调试的情况下推荐使用本方法进行程序烧录。



4.4 常用操作

1. 变量与函数声明的定位

在文本编辑区里，将鼠标指针悬停在某个宏常量、变量或是函数上时，会弹出文本框显示对应的宏展开、变量或函数声明。

若想跳转到相应的声明处，可以右键单击相应宏常量、变量或是函数，在弹出的菜单中选择 **Open Declaration**（或左键单击文本，按 F3 快捷键；也可以按住 Ctrl 键不放，左键单击相应文本）追踪该变量或函数、宏常量的上层定义位置，继续该操作直到最早定义位置（有些定义是存放到 .ld 文件中，需按照前面目录“3.3.2 增加或更改文件的查看方式”设置关联 .ld 文件或其他文件。此外，可以用 Alt+<- 及 Alt+>- 快捷键来后退或前进到前一个或后一个光标所在处。由此可以查看函数或变量的属性，用于更好的理解程序。

有以下特殊情况无法索引到变量或函数声明，可使用“搜索与替换”中“在所有打开的工程文件中搜索/替换关键字”的方法进行搜索定位。

- (1) 变量或函数声明所在的文件不属于当前工程。
- (2) 变量或函数声明所在的文件是链接文件或汇编文件。

2. 搜索与替换

若只需在单个文件中搜索/替换关键字，可以点击菜单栏中的 **Edit->Find/Replace**（快捷键 Ctrl+F）。

若需要在所有打开的工程文件中搜索/替换关键字，可以点击菜单栏中的 **Search->File**（也可使用快捷键 Ctrl+H，然后点击 **File Search** 选项卡），在 **Containing text** 里填入搜索关键字，在 **File name patterns** 里填入需要搜索的文件类型（一般填写 *.*），点击 **Search** 或 **Replace** 来进行搜索与替换。搜索结果会在屏幕下方的 **Search** 窗口中显示。

3. 添加文件/文件夹

在工程中添加软件构件时采用本方法。

方法一：点击工程菜单右键选择“**Import**”，在对话框中选择“**File System**”，点击“**Next**”，选择需要添加的文件/文件夹路径，确定后选中需要选中的文件/文件夹，如果添加的是文件夹，下方 **Options** 选项卡中要勾选“**Create top-level folder**”，点击 **Finish** 即可；如果添加的是文件，直接点击 **Finish** 即可。

方法二：将要添加的文件复制到要放入的工程目录下，然后在 KDS 下，右击工程名点击“**Refresh**”，文件便会自动加进工程。推荐使用第二种方法添加一个文件/文件夹。

4. 添加文件夹引用

虽然添加文件夹将文件夹加进了工程，KDS 并未将文件夹“引用”，需要添加工程应用，这样头文件才能将其包含至工程。在 KDS 下右击工程名点击“**Properties**”，在左边栏点击“**C/C++ Build**”下选择“**Settings**”。在“**Setting**”右边的选项卡内选择“**Tool Settings**”下的“**Cross ARM C Compiler**”中的“**Includes**”项，然后在“**Include Paths (-I)**”单击  添加工程路径单击“**OK**”便可。单击  可以删除某一路径引用。

5. KDS组件更新

如果 KDS 不支持当前连接的目标板，则需要对 KDS 开发环境进行组件更新，以获得更多的芯片型号支持。

在菜单栏下单击“**Help**”的“**Check for Updates**”，如果有新的更新，则选择 **Help** 菜单下的“**Install New Software**”，在对话框中选择要升级文件即可。注：组件更新不等于 KDS

软件版本更新，如果飞思卡尔官网有新的 KDS 版本，仍需在官网上进行下载安装，同时可以卸载旧版本。

第5章 Q&A

(1) 在编译MQX工程时会提示“writing to APSR without specifying a bitmask is deprecated”错误。

答：这个提示在 Console 窗口中为警告，但在 Problems 窗口中变为了错误，这是 KDS 的 bug，可以忽略，不影响最终生成 elf 及 hex 文件。

(2) 在打开工程时，选择好工程所在文件夹后无法点击Finish按钮导入工程，工程导入向导中的Projects栏里虽然出现了工程所在路径，但却是灰色的无法勾选。

答：这是因为在 KDS 中已经打开了同名工程。可以在工程浏览器里将打开的同名工程关闭，或是将已打开的工程重命名（右键点击工程，选择 Rename），就可以将另外一个同名工程导入了。

(3) 将核心板通过写入器端口用USB线连接到电脑后没有任何反应，电脑右下方也没有弹出“发现新硬件”的提示。

答：右击“我的电脑->管理”，显示“计算机管理”视图，找到“系统工具->设备管理器”，其中有带问号的疑问设备，如图 5-1 所示。



图 5-1 待安装驱动界面

右击选择“更新驱动程序”，系统弹出“发现新硬件”的提示，并弹出“找到新的硬件向导”对话框，选择“从列表或指定位置安装（高级）”选项。如图 5-2 所示，

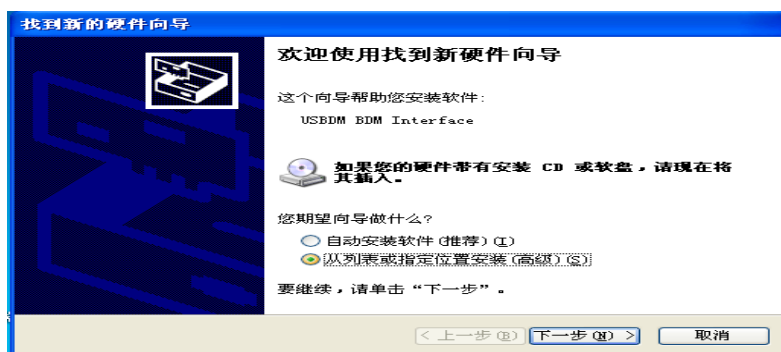
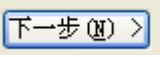


图5-2 发现USBDM

单击“”，如图 5-3 所示。

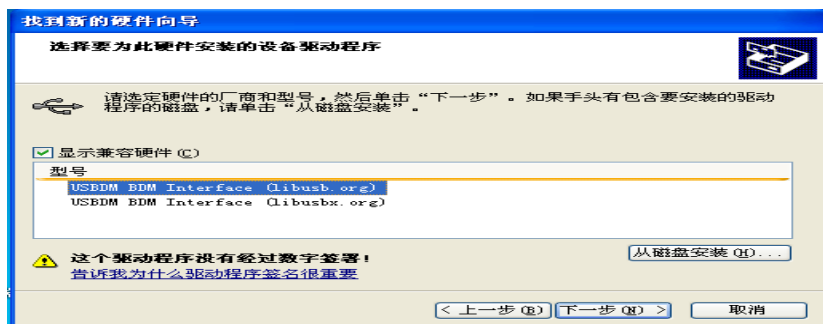


图 5-3 选择要安装的驱动

选择“从磁盘安装 (D)...”，然后选择驱动程序的路径，即“C:\Program Files\pgo\USBDM Drivers 1.0.1\Drivers\BDM_Driver”文件夹（XP 系统）。“C:\Program Files (X86)\pgo\USBDM Drivers 1.0.1\Drivers\BDM_Driver”文件夹（WIN7 系统）。如图 5-4 所示。

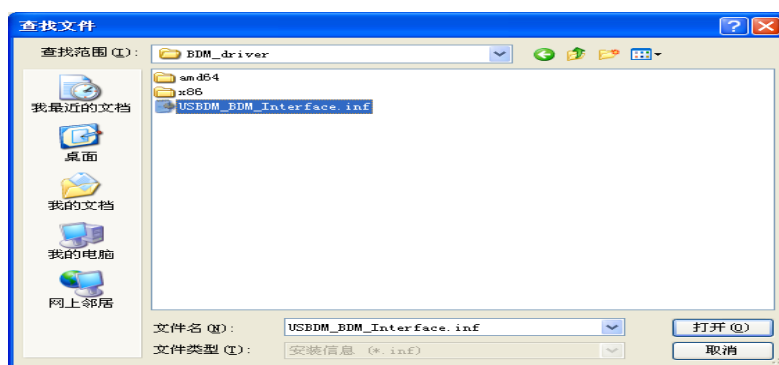


图 5-4 驱动程序路径

单击“打开 (O)”按钮，进行驱动安装，最后单击“完成”将完成 USB 驱动的安装。驱动程序是 USBDM_BDM_Interface.inf。

若在 64 位 WIN7 系统下，安装 USBDM_Drivers_1_0_1_Win_x64.msi 后，将核心板的写入器端口与 PC 机的 USB 口相连，系统会自动安装好驱动。视不同系统而定。

（4）系统中已经存在某些驱动文件，导致冲突，以上处理方法无效，USBDM 驱动安装不成功。

答：首先，确定驱动安装在系统中的文件名。找一台已经安装好 USBDM 驱动的机器，在设备管理器中找到 USBDM 的设备条目，右击选择“属性”，选择“驱动程序→驱动程序详细信息”，如图 5-5 所示。



图 5-5 USBDM 驱动文件信息框

从 USBDM 驱动文件信息框可知道驱动程序在系统中的文件为 WdCoInstaller01011.dll，在操作系统的“Windows\system32”目录下搜索该文件，得到结果如图 5-6 所示。

名称	所在文件夹	大小	类型	
WdCoInstaller01011.dll	C:\WINDOWS\system32	1,571 KB	应用程序扩展	2
WdCoInstaller01011.dll	C:\WINDOWS\system32\DRVST...	1,571 KB	应用程序扩展	2
WdCoInstaller01011.dll	C:\WINDOWS\system32\DRVST...	1,571 KB	应用程序扩展	2

图 5-6 USBDM 驱动文件的位置

删除这些文件（但最好从 WINDOWS 系统的“开始→控制面板→卸载程序”，打开卸载程序窗口，找到 USBM Driver 驱动程序，将其卸载。有的系统这些文件是难以直接删除的。），重新安装 USBDM 的驱动：USBDM_Drivers_1_0_1_Win_x32.msi（32 位操作系统下使用）或 USBDM_Drivers_1_0_1_Win_x64.msi（64 位操作系统下使用）。驱动安装完成后，重启计算机，问题得以解决。

（5）有时 USB 接头插入计算机 USB 接口时无反应。

答：可能原因是笔记本电脑或台式机的前置 USB 口供电不足，换其他 USB 口重试。

（6）编程下载对话框中的选项出现异常，无法下载程序。

答：断开 BDM 与 PC 机的连接，将当前工程关闭。再重新连接 BDM，打开工程，一般可以消除异常执行正常的下载操作。注意此步骤是在写入器固件程序升级的基础上执行的。

（7）工程无误，但编译时提示错误找不到路径包含头文件路径

答：可能是工程使用了中文目录，应保证工程的路径为全英文字符。

（8）导入现有工程后直接编译不过。

答：可以清除当前工程之前的生成文件，重新创建生成文件即可。

（9）打开工程后看不到工程目录结构，或者误删了工程框栏。

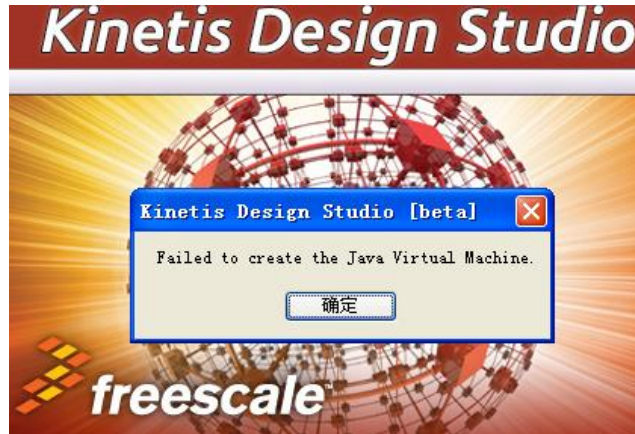
答：选择“Window”->“Show View”->“Project Explorer”，可在左侧面板显示工程目录结构。

(10) 编程下载对话框中的选项出现异常，无法下载程序。

断开 BDM 与 PC 机的连接，将当前工程关闭。再重新连接 BDM，打开工程，一般可以消除异常执行正常的下载操作。注意此步骤是在写入器固件程序升级的基础上执行的。

(11) 打开KDS环境时出现“Failed to create the Java Virtual Machine”或者“Out of memory”问题。

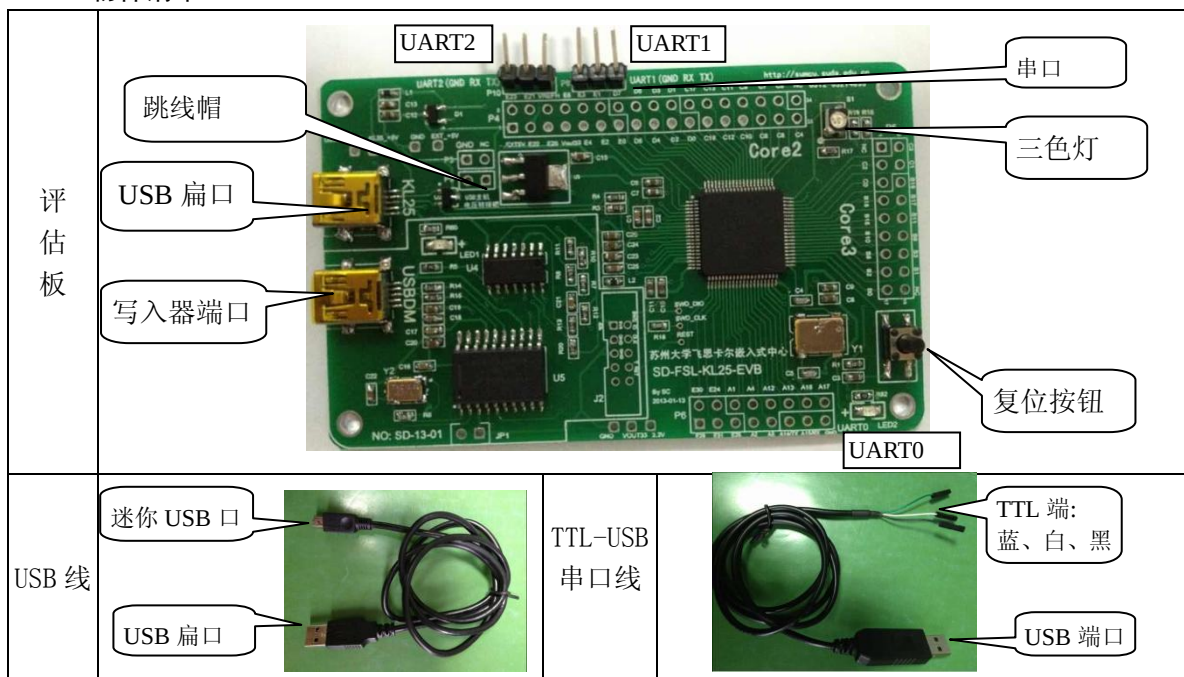
答：



首先关闭如 360 等杀毒软件，再进行尝试。如果还是出现上述问题，则需要重新设置开发环境的最高堆栈大小，找到 KDS 安装目录\ eclipse\kinetis-designstudio.ini，打开 ini 文件，默认是-Xmx512m，意味着为 KDS 分配的最高堆栈大小为 512MB，若出现“Failed to create the Java Virtual Machine”问题，则降低该值（如 256MB），若出现“Out of memory”，则提高该值（如 1024MB）。

附录1 《SD-FSL-KL25-EVB》简明测试方法

1. 物件清单:



2. 测试方法: (出厂时, 已经将 SD-FSL-KL25-CD(网上光盘)中“..\KL25Test”文件夹下的程序写入 KL25 芯片)

第一步: 观察三色灯变化: (1) 将“USB 线”的“迷你 USB 口”端接入开发板的“USB 扁口”(注意不要接到评估板的“写入器端口”); (2) 将“USB 线”的 USB 扁口”端接入 PC 机的 USB 口 (给开发板供电)。现象: 开发板上的“三色灯”颜色不停变换, 则 KL25 芯片运行正常 (若不运行, 可按一下开发板上的“复位按钮”, 再试一下)。

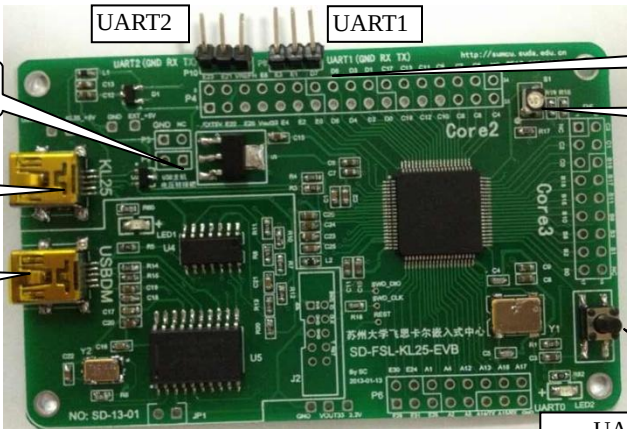
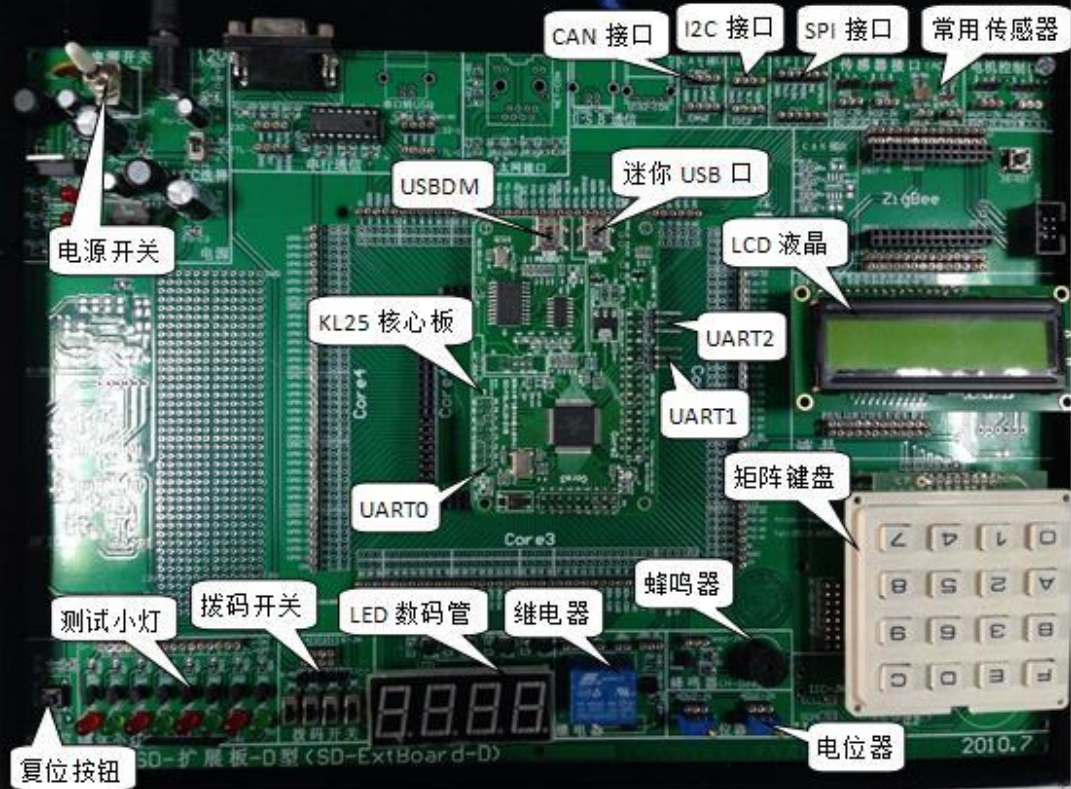
第二步: 测试串口通信: (a) 在 PC 机安装 TTL-USB 驱动 (网上光盘“..\Tools\TTL-USB 驱动”中) PL2303_Prolific_DriverInstaller_v110.exe; (b) 将“TTL-USB 串口线”的“USB 端口”接 PC 机的 USB 口, 串口线的串口接核心板上的串口 (3 根, 以 UART_1 为例, RX 接蓝线, TX 接白线, GND 接黑线; UART_2 同样对应字母对应不同线); (c) 在此, 以 UART_1 为例, 在 PC 机, 运行串口调试工具 (串口调试工具), (“9600、无校验”), 选择正确串口号, 测试串口不停换行显示“HELLO WORLD”。按主板上的“复位按钮”, 串口测试窗口显示“This is UART1 INT Test!”, 说明主板串口发送正常。单击“发送”按钮, 若出现回显字符串输入框内容“abcdefg1234567h”, 说明主板串口收发正常。

第三步: 测试 USB: 安装网上光盘中“..\KL25-Program\ch12-KL25-USB\USB-Device\USBTest(PC-C#)\SoochowUniversity-USB(Driver)”中 PC 方 USB 设备驱动程序“SoochowUniversity-USBDevice.inf”。(a) 将“USB 线”的“迷你 USB 口”端接入开发板的“USB 扁口”。(b) 双击“USBTest.exe”打开 USBTest

窗口，在发送框中写入任意数字或字母，点击“发送”，在接受框内显示每个数字或字母的下一位。

附录2 《SD-FSL-KL25-实验箱》简明测试方法

1. 实验箱器件:

核心板	 跳线帽 UART2 UART1 串口 三色灯 USB 扁口 写入器端口 复位按钮 UAR
U S B 线	 迷你 USB 口 USB 扁口 TTL-USB 串口线  TTL 端: 蓝、白、黑 USB 端口
主 板	 CAN 接口 I2C 接口 SPI 接口 常用传感器 电源开关 USBDM 迷你 USB 口 LCD 液晶 KL25 核心板 UART2 UART1 UART0 矩阵键盘 测试小灯 拨码开关 LED 数码管 继电器 蜂鸣器 电位器 复位按钮
电 源 适 配 器	 接交流220V 电源 接实验板插头

2. 测试方法: (默认已经将 SD-FSL-KL25-CD(网上光盘)中“..\KL25-Test(KDS)”文件夹下的程序写入 KL25 芯片)

第一步: 与软件无关测试:

(1) **测试电源模块:** 接通 12V 电源适配器, 观察有没有出现灯不正常闪烁或者有蜂鸣声。若有则马上关闭电源; 若没有则观察 LED12V, LED5V, LED3.3V 是否正常 (全亮是正常); 拨动 PW-Se1, 测量 VCC 是否为 5V 或 3.3V。

(2) **测试小灯:** 用导线的一端接+5V 或, 另一端依次点接 D-JK 的 1~16 脚 (每两个脚对应一盏小灯), 观察 1~8 盏小灯是否能够点亮。(粗略测试: 一手触摸+3.3V, 另一手触摸 D-JK 的 1~16 脚各小灯微亮, 说明小灯基本正常)

(3) **拨码开关:** 用导线将拨码开关接小灯, 测试开关是否正常。

(4) **继电器:** 用导线将“继电器的 JDQ-JK 插线孔”接+5V 或+3V, 能听到“喀嗒”声说明继电器正常。

(5) **电位器 1:** 用万用表测量“电位器的 ADV1-JK 插线孔”的电压 Vd, 旋转电位器旋钮, 正常情况 Vd=0~5V。电位器 2 同样测量。

第二步: 快速测试:

(1) 接上电源, 打开电源开关, 电源开关旁边三个红色指示灯亮

(2) 观察核心板上圆形小灯发“白色”光

(3) 几秒后, 圆形小灯变为其他颜色, 数码管显示“5678”, LCD 显示 Wait Receiving..Soochow 2013.01.

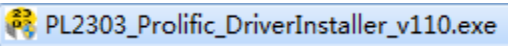
(4) 按下键盘任一键, LCD 屏的最后一位显示键的标识值, 并且蜂鸣器响一声

第三步: 观察 LCD 液晶: (1) 拔下 USB 线, 在确保实验箱上主板电源开关处于关闭状态下, 将开关电源适配器插上实验箱的主板。(2) 打开实验箱主板的电源开关。现象: 开发板上的“LCD 液晶屏”显示两行字符“Wait Receiving..Soochow 2013.01.”, 则 LCD 液晶运行正常 (若不运行, 可按一下开发板上的“复位按钮”再试一下)。

第四步: 测试键盘, LED 和蜂鸣器: (1) 继第三步操作; (2) 按下键盘上的 4*4 按钮中的任何一个。

现象: LCD 液晶显示字符“The keyboard you just input is X”(X 为键盘上对应字符), LED 数码管显示“888

8”, 蜂鸣器在键盘按下时发出“嘀”的一声, 三色灯显示黄色。则 LED 数码管, 键盘和蜂鸣器运行正常 (若不运行, 可按一下开发板上的“复位按钮”, 再试一下)。

第五步: 测试串口通信: (测试前按一下复位按钮) (a) 在 PC 机安装 TTL-USB 驱动 (网上光盘“..\Tools\TTL-USB 驱动”中) ; (b) 将“TTL-USB 串口线”的“USB 端口”接 PC 机的 USB 口, 串口线的串口接核心板上的串口 (3 根, 以 UART_1 为例, RX 接绿线, TX 接白线, GND 接黑线; UART_2 同样对应字母对应不同线); (c) 在此, 以 UART_1 为例, 在 PC 机, 运行串口调试工具 () (“9600、无校验”), 选择正确串口号, 测试串口不停换行显示“HELLO WORLD”。按核

心板上的“复位按钮”，串口测试窗口显示“This is UART1 INT Test!”，说明核心板串口发送正常。单击“发送”按钮，若出现回显字符串输入框内容“abcdefg1234567h”，说明核心板串口收发正常。

第 六 步 ： 测 试 USB ： 安 装 网 上 光 盘 中
“..\KL25-Program\ch12-KL25-USB\USB(Slave)\USBTest(PC-C#)\

SoochowUniversity-USB(Driver) ” 中 PC 方 USB 设备驱动程序
“SoochowUniversity-USBDevice.inf”。(a) 将“USB 线”的“迷你 USB 口”端接入核心板的“USB 扁口”。(b) 双击“USBTest.exe”打开 USBTest 窗口，在发送框中写入任意数字或字母，点击“发送”，在接受框内显示每个数字或字母的下一位。(如果 USB 读取失败，可以尝试拔下重新插入或者按复位按钮)

关于写入程序，见“..\KL25-Document”相关文档。特别注意：写入程序时，大板需供电。