



10年口碑积累，成功培养50000多名研发工程师，铸就专业品牌形象

华清远见的企业理念是不仅要良心教育、做专业教育，更要做受人尊敬的职业教育。

《从实践中学 ARM 体系结构与接口技术》

作者：华清远见

专业始于专注 卓识源于远见

第 5 章 ARM RealView MDK 集成开发环境

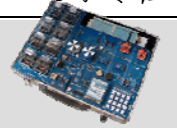
本章简介

ARM 公司在前一个开发环境 ADS 1.2（不再提供升级）后，推出了 RealView 系列开发环境，其中 RealView MDK 环境以其优越的性价比得到了快速的推广，本章主要介绍其使用、配置方法。本章主要内容：

- RealView MDK 环境介绍
- ULINK2 仿真器简介
- RealView MDK 的使用

专业始于专注 卓识源于远见

5.1 RealView MDK 环境介绍



MDK (Microcontroller Development Kit) 是 Keil 公司 (现已被 ARM 公司收购) 开发的 ARM 开发工具, 是用来开发基于 ARM 核的系列微控制器的嵌入式应用程序的开发工具。它适合不同层次的开发者使用, 包括专业的应用程序开发工程师和嵌入式软件开发的入门者。MDK 包含了工业标准的 Keil C 编译器、宏汇编器、调试器、实时内核等组件, 并支持所有基于 ARM 的设备, 能帮助工程师按照计划完成项目。

Keil ARM 开发工具集成了很多有用的工具 (如图 5-1 所示), 正确地使用它们, 有助于快速完成项目开发。

组件	Part Number	
	MDK-ARM ^{2,3}	DB-ARM
μVision IDE	✓	✓
RealView C/C++ Compiler	✓	
RealView Macro Assembler	✓	
RealView Utilities	✓	
RTL-ARM Real-Time Library	✓	
μVision Debugger	✓	✓
GNU GCC1	✓	✓

图 5-1 MDK 开发工具的组件

MDK 的最新版本是 μVision 3, 利用它可以开发基于 ARM7、ARM9、Cortex-M3 的微控制器应用程序。MDK 易学易用且功能强大, 其主要特性如下:

- (1) μVision 3 集成了一个能自动配置工具选项的设备数据库。
 - (2) 工业标准的 RealView C/C++ 编译器能产生代码容量最小、运行速度最快的高效应用程序, 同时它包含了一个支持 C++ STL 的 ISO 运行库。
 - (3) 集成在 μVision 3 中的在线帮助系统提供了大量有价值的信息, 利用它可加速应用程序的开发速度。
 - (4) 包含大量的例程, 帮助开发者快速配置 ARM 设备, 以及开始应用程序的开发。
 - (5) μVision 3 集成开发环境能帮助工程人员开发稳健、功能强大的嵌入式应用程序。
 - (6) μVision 3 调试器能够精确地仿真整个微控制器, 包括其片上外设, 使得在没有目标硬件的情况下也能测试开发程序。
 - (7) 包含标准的微控制器和外部 Flash 设备的 Flash 编程算法。
 - (8) ULINK USB-JTAG 仿真器可以实现 Flash 下载和片上调试。
 - (9) RealView RL-ARM 具有网络和通信的库文件及实时软件。
 - (10) 还可使用第三方工具扩展 μVision 3 的功能。
 - (11) μVision 3 还支持 GNU 的编译器。
- 本书的全部例程都是在 MDK 环境下开发编写的。

5.2 ULINK2 仿真器简介



ULINK 是 Keil 公司提供的 USB-JTAG 接口仿真器, 目前最新的版本是 2.0。它支持诸多芯片厂商的 8051、ARM7、ARM9、Cortex-M3、Infineon C16x、Infineon XC16x、Infineon XC8xx、STMicroelectronics μPSD 等多个系列的处理器。ULINK2 内部实物如图 5-2 所示, 由 PC 的 USB 接口提供电源。ULINK2 不仅包含了 ULINK USB-JTAG 适配器具有的所有特点, 还增加了串行线调试 (SWD) 支持, 以及返回时钟支持和实时代理功能。

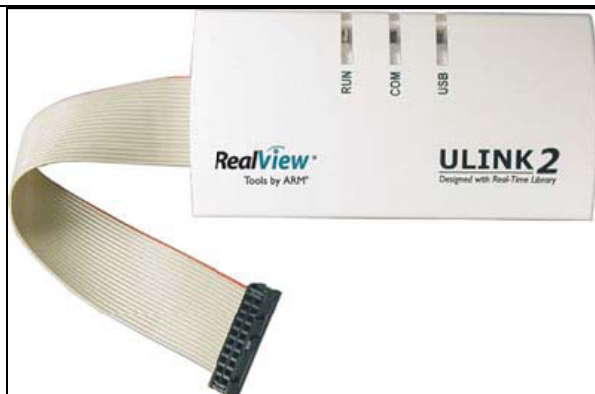


图 5-2 ULINK2 仿真器

ULINK2 的主要功能如下：

- (1) 下载目标程序。
- (2) 检查内存和寄存器。
- (3) 片上调试，整个程序的单步执行。
- (4) 插入多个断点。
- (5) 运行实时程序。
- (6) 对 Flash 存储器进行编程。

ULINK2 的新特点包括：

- (1) 标准 Windows USB 驱动支持，也就是 ULINK2 即插即用。
- (2) 支持基于 ARM Cortex-M3 的串行线调试。
- (3) 支持程序运行期间的存储器读写、终端仿真和串行调试输出。
- (4) 支持 10/20 针连接器。

本书使用的例程均使用 ULINK2 仿真器进行调试。

5.3

RealView MDK 的使用



RealView MDK 引入工程管理，使得基于 ARM 处理器的应用程序设计开发变得越来越方便。通常使用 RealView MDK 创建、完成一个新的工程只需要以下几个环节：选择工具集→创建工程并选择处理器→创建源文件→配置硬件选项→配置对应启动代码→编译链接→调试→生成镜像文件→镜像文件下载。

5.3.1 选择工具集

利用 μ Vision 3 创建一个基于处理器的应用程序，首先要选择开发工具集。选择 Project→Manage→Components, Environment and Books 命令，在弹出的图 5-3 所示的对话框中，可选择所使用的工具集。在 μ Vision 3 中既可以使用 ARM RealView 编译器、ARM ADS 编译器、GNU GCC 编译器，也可以使用 Keil CARM 编译器。当使用 GNU GCC 编译器或者 ARM ADS 编译器时，需要安装相应的工具集。在本例程中选择 RealView Compiler 编译器。

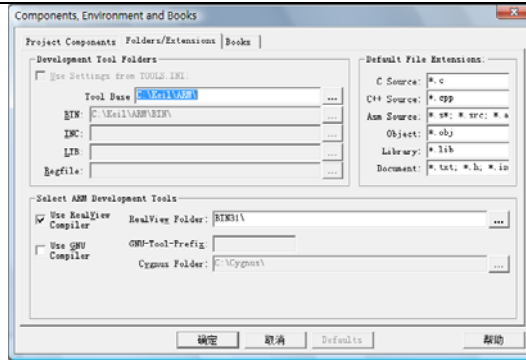


图 5-3 选择工具集

5.3.2 创建工程并选择处理器

选择 Project→New Project 命令，μVision 3 将打开一个标准对话框，输入工程名，即可创建一个新的工程。建议对每个新建工程都使用一个独立的文件夹。先在硬盘上创建一个新的文件夹 Hello，在前述对话框中输入 Hello，μVision 将会创建一个以 Hello.uv2 为名称的新工程文件，它包含了一个默认的目标（target）和文件名。这些内容在 Project→Workspace→Files 中可以看到。

创建一个新工程时，μVision 3 要求设计者为工程选择一款对应处理器，如图 5-4 所示，该对话框中列出了 μVision 3 所支持的处理器设备数据库，也可以通过选择 Project→Select Device 命令进入该对话框。当选择了某款处理器后，μVision 3 将会自动为工程设置相应的工具选项，这使得工具的配置过程简化很多。

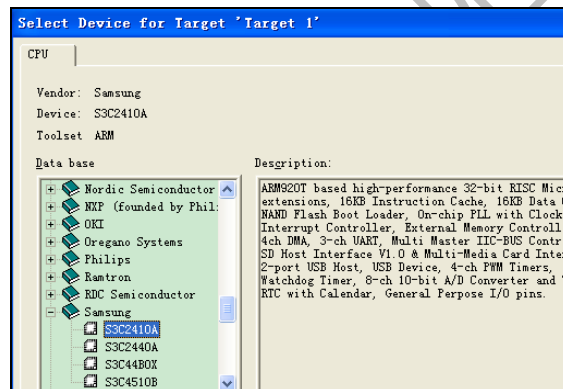


图 5-4 选择处理器

对于大部分处理器设备来说，μVision 3 会提示程序员是否在目标工程中加入 CPU 的相关启动代码（如图 5-5 所示）。启动代码是用来初始化目标设备的配置，完成运行时系统的初始化工作对于嵌入式系统开发而言是必不可少的。单击 OK 按钮便可将启动代码加入工程，这使得系统的启动代码编写工作量大大减少。图 5-5 给出了加入启动代码后的工程文件，其中 S3C2410A.s 就是系统自带的启动代码。这段代码是 CPU 复位后首先要执行的代码，不能忽视它的作用。

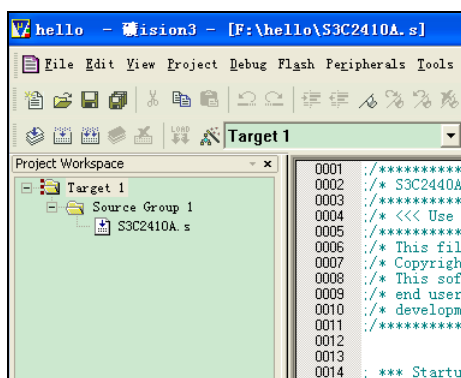


图 5-5 加入启动代码后的工程

5.3.3 建立一个新的源文件

工程创建完成以后,就可以开始编写程序。向工程中创建新文件的方法是选择 File→New 命令,μVision IDE 将会打开一个空的编辑窗口用以使程序员输入源程序。在输入完源程序后,选择 File→Save As 命令保存源程序。当以*.c 为扩展名保存源文件时,μVision IDE 将会根据语法以彩色高亮字体显示源程序。

5.3.4 工程中文件的加入

创建完源文件后便可以在工程中加入此源文件,μVision 提供了多种方法加入源文件到工程中。例如,在 Project→Workspace→Files 菜单项中选择文件组,并单击鼠标右键,弹出如图 5-6 所示的快捷菜单,选择 Add Files to Group...命令打开一个标准文件对话框,将已创建好的源文件加入到工程中。

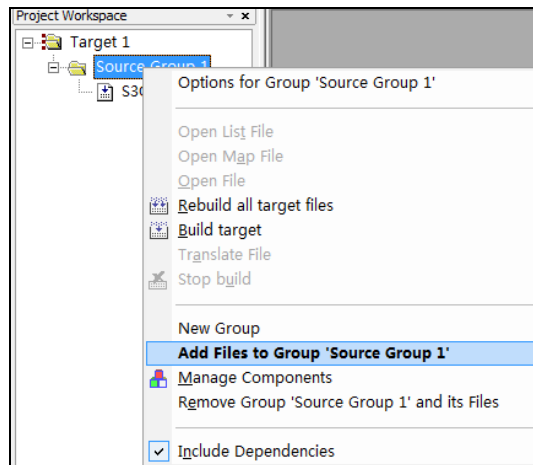


图 5-6 加入源文件到工程中

5.3.5 工程基本配置

1. 硬件选项配置

μVision 3 可根据目标硬件的实际情况对工程进行配置。通过单击目标工具栏图标或者选择 Project→Options for Target 命令,在弹出的对话框的 Target 选项卡中可指定目标硬件和所选择设备片内组件的相关参数,如图 5-7 所示。

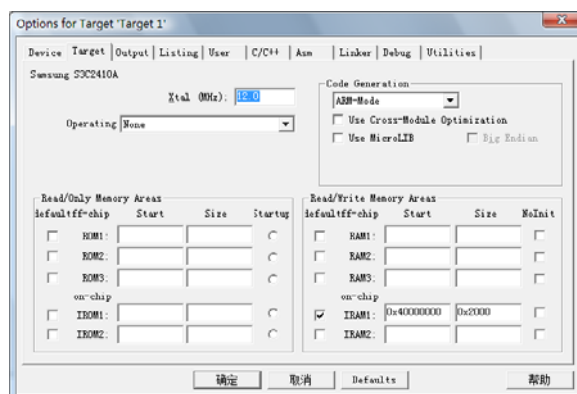


图 5-7 处理器配置对话框

所填写的内容包括 RAM、ROM 的起始地址及大小。同时还可以选择是否使用 Thumb 模式、是否使用 RTX 内核等。如果我们没有配置 ROM 等信息,在编译过程中会提示“ROM1 没有定义”之类的错误。

2. 处理器启动代码配置

通常情况下，ARM 程序都需要初始化代码用来配置所对应的目标硬件。如 4.3.2 节所述，当创建一个应用程序时， μ Vision 3 会提示使用者自动加入相应设备的启动代码。 μ Vision 3 提供了丰富的启动代码文件，可在相应文件夹中获得。例如，针对 Keil 开发工具的启动代码放在..\ARM\Startup 文件夹下，针对 GNU 开发工具的启动代码放在..\ARM\GNU\Startup 文件夹下，针对 ADS 开发工具的启动代码放在..\ARM\ADS\Startup 文件夹下。

在使用 ULINK 仿真器时，首先为仿真器选择合适的驱动，并为下载应用程序和可执行文件进行配置。关于仿真的设置如图 5-8 和图 5-9 所示。

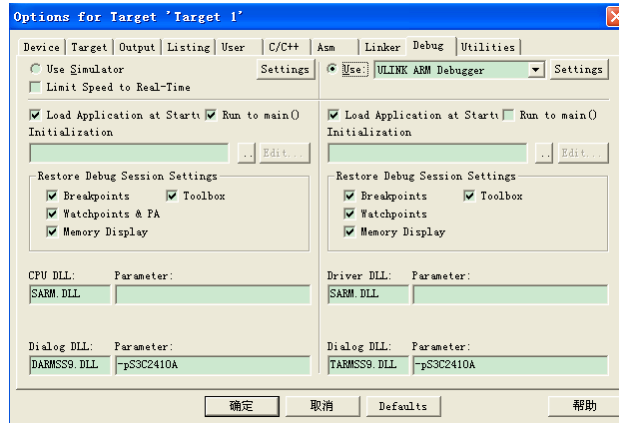


图 5-8 仿真器驱动配置图

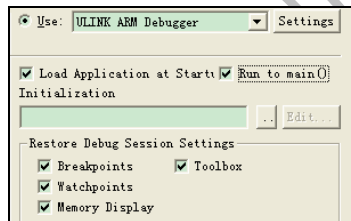


图 5-9 仿真器下载应用程序配置图

3. 工具配置

工具选项（Utilities）主要设置 Flash 的下载选项，如图 5-10 所示。

在图 5-10 所示的对话框中选择 Use Target Driver for Flash Programming 单选按钮，在下拉列表中选择 ULINK ARM Debugger 选项，同时勾选 Update Target before Debugging 复选框。这时还没有完成设置，还需要选择编程算法，单击 Settings 按钮，将弹出如图 5-11 所示的对话框。

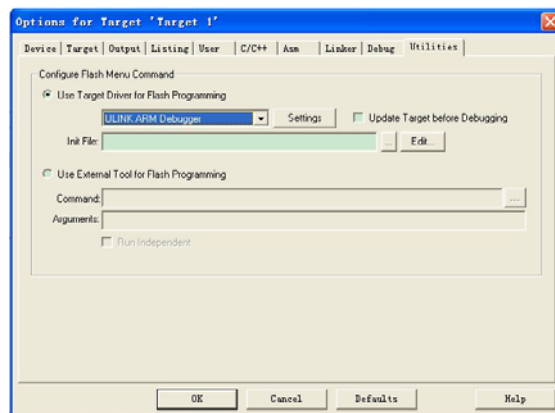


图 5-10 Utilities 配置对话框

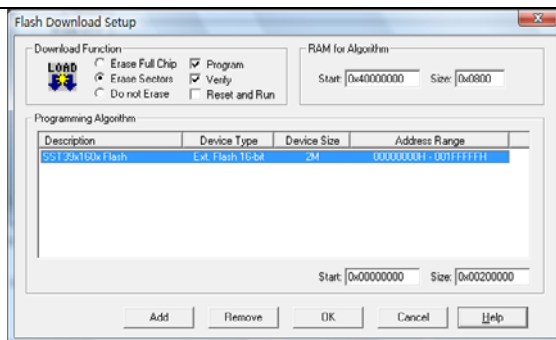


图 5-11 Flash 下载选项设置

单击对话框中的 Add 按钮，将弹出如图 5-12 所示的对话框，在该对话框中选择需要的 Flash 编程算法。例如，对 SST39x160x 芯片，由于其 Flash 为 2M，所以需要选择图 5-12 所标注的 Flash 编程算法。这一步主要是为了镜像下载用，参考本书 5.5 节的内容。

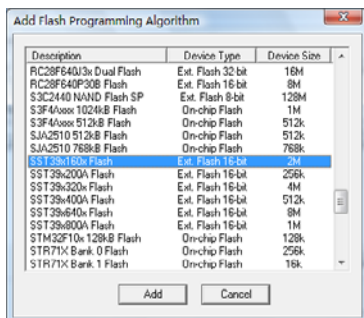


图 5-12 选择 Flash 编程算法

4. 调试设置

μVision 3 调试器提供了两种调试模式，可以选择 Project→Options for Target 命令，在弹出的对话框的 Debug 选项卡中进行选择，如图 5-13 所示。

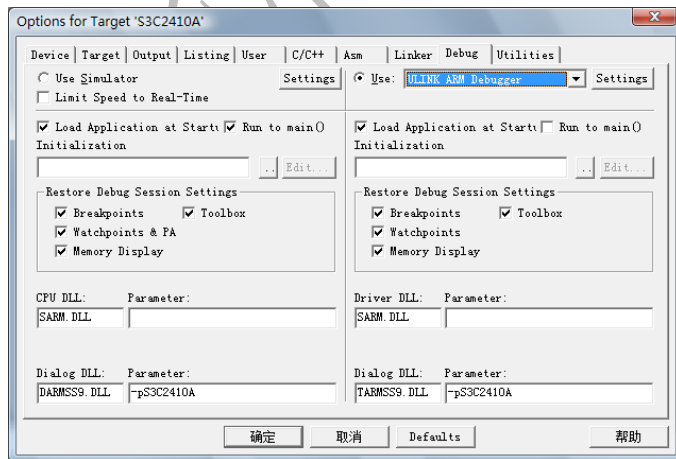


图 5-13 调试器的选择

各选项说明如下。

- Use Simulator: 表示使用 μVision 仿真器调试程序。
- Use: 选择调试的目标驱动器。用户可以通过使用不同的软件和硬件工具来测试程序。例如，选择 ULINK ARM Debugger 支持 ULINK2 仿真调试硬件。
- Load Application at Startup: 在调试器启动时装载在 Output 对话框中指定的可执行文件。
- Run to main(): 启动 μVision 调试器后，执行程序运行到主函数 main。
- Initialization: 指定一个初始化文件，在这个文件中可包含在调试器启动时要执行的调试命令，或者包含在调试会话阶段要使用到的调试函数。单击 Edit 或 Create 按钮可以打开一个编辑窗口来编

辑初始化文件。在本书后面章节的很多例程中都会用到初始化文件。

如果目标板已上电，并且与 ULINK 连接好，单击图 5-13 右侧的 Settings 按钮，将弹出如图 5-14 所示的对话框，如果正常，则可读取目标板芯片 ID 号。如果读不出 ID 号，则需要检查 ULINK 与 PC 或目标板的连接是否正确。

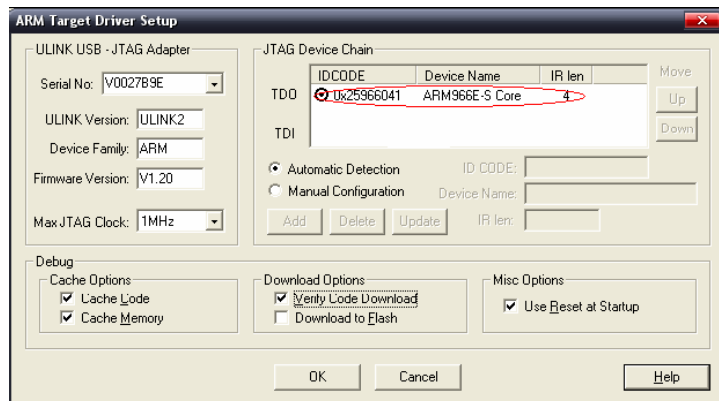


图 5-14 读取设备 ID

5. 编译配置

μVision IDE 目前支持 RealView、Keil CARM 和 GNU 这 3 种编译器，选择 Project→Manage→Component, Environment and Books 命令，或者直接单击工具栏中的图标，在弹出的对话框的 Folders/Extensions 选项卡中选择编辑器类型。我们使用 RealView 编译器，如图 5-15 所示。

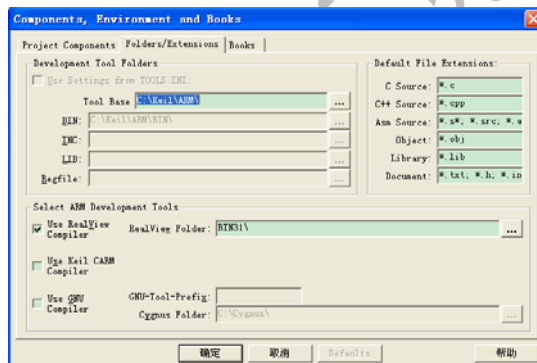


图 5-15 选择编译器

选择好编译器后，打开 Option for Target 对话框的 C/C++选项卡，出现如图 5-16 所示的编译属性配置页面（这里主要说明 RealView 编译器的编译配置）。

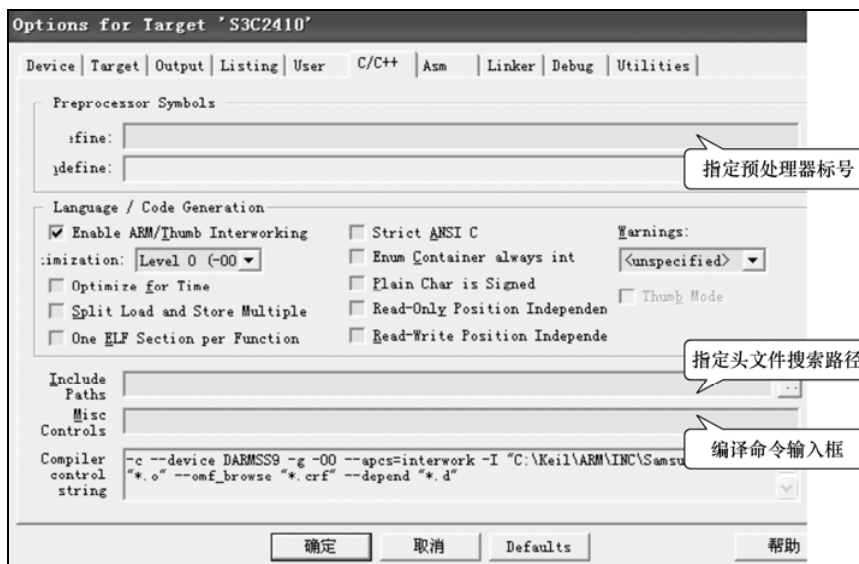


图 5-16 编译器配置页

各个编译选项说明如下。

- Enable ARM/Thumb Interworking: 生成 ARM/Thumb 指令集的目标代码, 支持两种指令之间的函数调用。
- Optimization: 优化等级选项, 分 4 个档次。
- Optimize for Time: 时间优化。
- Split Load and Store Multiple: 非对齐数据采用多次访问方式。
- One ELF Section per Function: 每个函数设置一个 ELF 段。
- Strict ANSI C: 编译标准 ANSI C 格式的源文件。
- Enum Container always int: 枚举值用整型数表示。
- Plain Char is Signed: Plain Char 类型用有符号字符表示。
- Read-Only Position Independent: 段中代码和只读数据的地址在运行时候可以改变。
- Read-Write Position Independent: 段中的可读/写的数据地址在运行期间可以改变。
- Warnings: 编译源文件时, 警告信息输出提示选项。

6. 汇编选项设置

打开 Option for Target 对话框的 Asm 选项卡, 出现如图 5-17 所示的汇编属性配置界面。

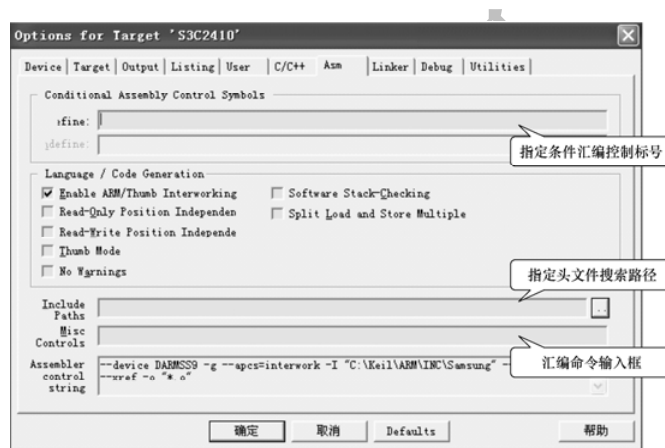


图 5-17 汇编配置界面

各个汇编选项说明如下。

- Enable ARM/Thumb Interworking: 生成 ARM/Thumb 指令集的目标代码, 支持两种指令之间的函数调用。
- Read-Only Position Independent: 段中代码和只读数据的地址在运行时候可以改变。
- Read-Write Position Independent: 段中的可读/写的数据地址在运行期间可以改变。
- Thumb Mode: 只编译 Thumb 指令集的汇编源文件。
- No Warnings: 不输出警告信息。
- Software Stack-Checking: 软件堆栈检查。
- Split Load and Store Multiple: 非对齐数据采用多次访问方式。

7. 链接选项设置

链接器/定位器用于将目标模块进行段合并, 并对其定位, 生成程序。既可通过命令行方式使用链接器, 也可在 μ Vision IDE 中使用链接器。单击图标, 打开 Option for Target 对话框的 Linker 选项卡, 出现如图 5-18 所示的链接属性配置页面。如果需要使用 Scatter File 文件, 则需要取消勾选 Use Memory Layout from Target Dialog 复选框。

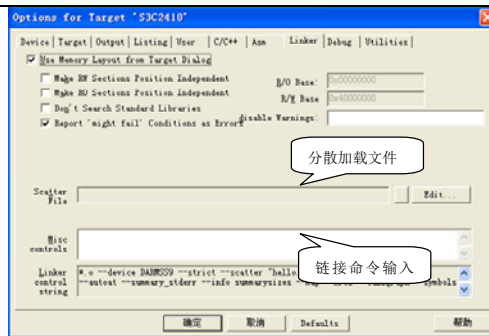


图 5-18 链接配置页

各个链接选项配置说明如下。

- Make RW Sections Position Independent: RW 段运行时可改变。
- Make RO Sections Position Independent: RO 段运行时可改变。
- Don't Search Standard Libraries: 链接时不搜索标准库。
- Report 'might fail' Conditions as Errors: 将 might fail 报告为错误提示输出。
- R/O Base: R/O 段起始地址输入框。
- R/W Base: R/W 段起始地址输入框。

8. 输出文件设置

选择 Project→Option for Target 命令，在弹出的对话框的 Output 选项卡中配置输出文件，如图 5-19 所示。



图 5-19 输出文件配置页

输出文件配置选项说明如下。

- Select Folder for Objects: 选择一个存储目标文件的文件夹。
- Name of Executable: 指定输出文件名。
- Create Executable: 为用户目标硬件或者仿真器生成一个可执行应用程序。
- Debug Information: 对于用户应用程序的符号测试的调试信息是必需的。如果用户没有勾选该复选框，在 μVision 调试器中，用户将不能查看函数名、变量名或者源代码。
- Browse Information: 集成的源码浏览所需的浏览信息，用于源文件的快速导航。
- Create HEX File: 允许用户创建一个 Intel HEX 文件。HEX 文件用于下载软件到一个 PROM 编程器中去或者其他的用途。
- Create Library: 一个库文件。库是目标模块的程序集合，能在以后为链接器所使用。用户可以将库当成其他工程文件的一部分来使用。
- Create Batch File: μVision 生成一个批处理文件 (*.BAT)，这个文件包含建立一个应用程序的所有命令。用户可以利用这个批处理文件来重复建立一个应用程序的过程，而无须利用 μVision IDE。
- Debug Information: 允许时，在可执行文件内存储符号的调试信息。
- Create HEX File: 允许时，使用外部程序生成一个 HEX 文件进行 Flash 编程。

- Big Endian: 输出文件采用大端对齐方式。
- Create Batch File: 创建批文件。

5.3.6 工程的编译链接

完成工程的设置后, 就可以对工程进行编译链接了。本书将利用 MDK 自带的一个例程来讲解工程的编译及 5.3.7 节的调试内容。

打开 MDK 的安装盘符:\Keil\ARM\Examples\Hello.Uv2, 如图 5-20 所示。

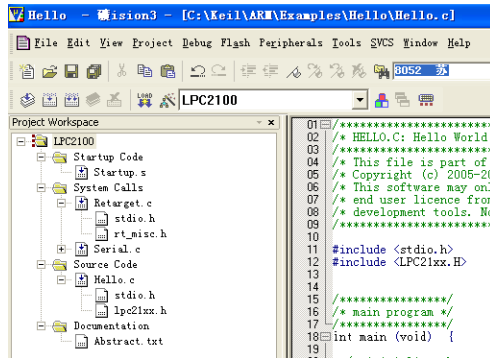


图 5-20 Hello 工程

用户可以通过选择 Project→Build target 命令, 或单击工具条按钮, 如图 5-21 所示, 编译相应的文件或工程, 同时在输出窗的 Build 子窗口中输出有关信息, 如图 5-22 所示。如果在编译链接过程中出现错误, 包括源文件语法错误和其他错误, 编译链接操作会立刻终止, 并在输出窗口的 Build 子窗口中提示错误。如果是语法错误, 用户可以通过双击错误提示行, 来定位引起错误的源文件行。

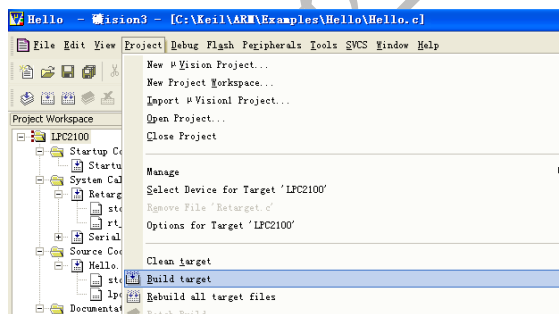


图 5-21 Hello 工程 Project 菜单和工具条

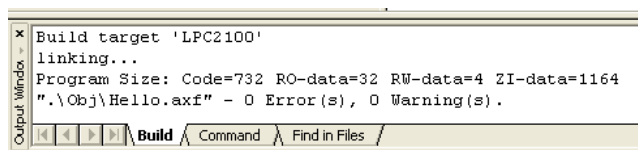


图 5-22 编译结果

5.3.7 工程的调试

在 5.3.5 节中介绍了关于调试器选择的配置, 本节将结合 5.3.6 节中的例程讲解如何进行调试。

选择 Debug→Start/Stop Debug Session 命令, 如图 5-23 所示, 进入调试状态, 如图 5-24 所示。

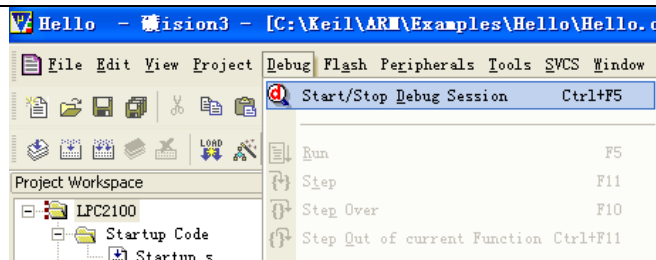


图 5-23 Debug 菜单

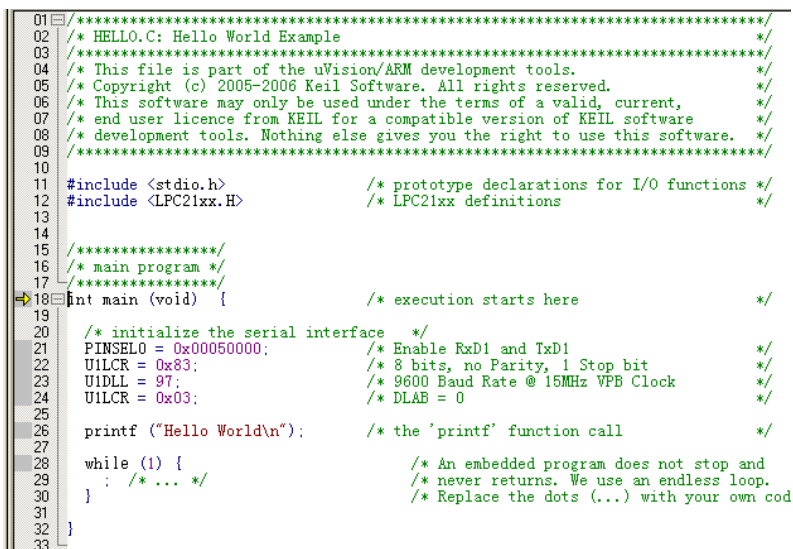


图 5-24 调试状态

此时可以通过图 5-25 的 Debug 调试工具来调试代码。



图 5-25 Debug 调试工具

在调试过程中设置断点的方法:

(1) 简单的程序断点, 可以通过在程序行前双击的方法来实现, 如图 5-26 所示。

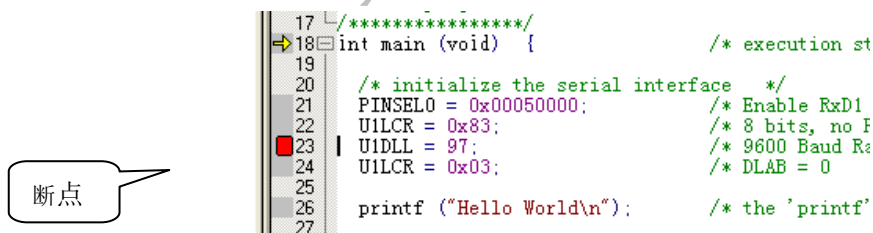


图 5-26 断点标识

(2) 复杂的断点, 可以通过选择 Debug→Breakpoints 命令, 会弹出如图 5-27 所示的对话框。

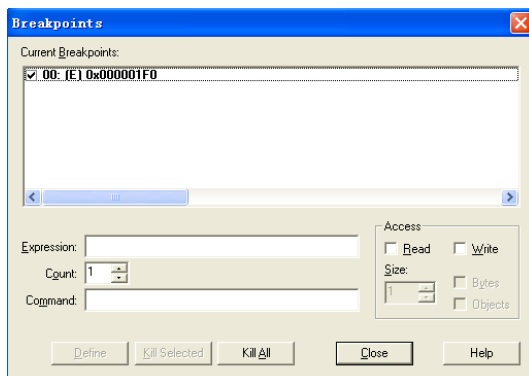


图 5-27 调试选项

其中各项含义如下。

- Expression: 指定断点表达式。
- Count: 当断点表达式的值累计达到 Count 时, 断点将会被触发。
- Command: 如果用户为断点指定一个命令, μ Vision 将会在该断点激活时执行相应的命令, 然后恢复执行目标程序。
- Define: 定义断点。
- Kill Selected: 清除在列表中被选择的断点。
- Kill All: 清除工程中定义的所有断点。
- Close: 关闭对话框。
- Read: 定义一个存取断点, 当读内存发生时这个断点将被激活。
- Write: 定义一个存取断点, 当写内存发生时这个断点将被激活。
- Size: 在存取断点处指定被检查的内存大小。
- Bytes: 指定内存大小以字节为单位。
- Objects: 指定内存大小以模块为单位。

灵活地配置调试选项, 可以大大缩减调试时间。

5.3.8 映像文件下载

在仿真环境下调试好程序后, 还需要将生成的文件镜像烧写到目标板的 Flash 中。

RealView MDK 结合 ULINK2 可以支持 Flash 烧写功能, 步骤如下:

- 按照 5.3.5 节中“工具配置”的要求选择正确的 Flash 算法。
- 连接好 ULINK2 仿真器。
- 编译成功后, 单击工具栏中的 load 按钮, 如图 5-28 所示, 实现镜像文件下载, 如图 5-29 所示。

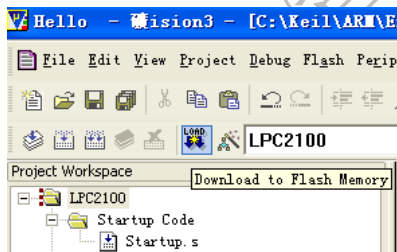


图 5-28 下载镜像

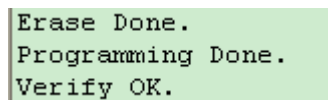


图 5-29 镜像下载成功

5.4

本章小结



本章主要介绍了 RealView MDK 环境的使用。本书后面章节的大部分实验都是基于这个环境的。工欲善其事, 必先利其器, 所以必须熟练掌握环境的使用。

5.5

本章习题



1. 熟悉 RealView MDK 开发环境。

2. 新建一个 RealView MDK 工程，编写一个汇编程序实现 $3+13=16$ 的操作。

联系方式

集团官网: www.hqyj.com

嵌入式学院: www.embedu.org

移动互联网学院: www.3g-edu.org

企业学院: www.farsight.com.cn

物联网学院: www.topsight.cn

研发中心: dev.hqyj.com

集团总部地址: 北京市海淀区西三旗悦秀路北京明园大学校内 华清远见教育集团

北京地址: 北京市海淀区西三旗悦秀路北京明园大学校区, 电话: 010-82600386/5

上海地址: 上海市徐汇区漕溪路银海大厦 A 座 8 层, 电话: 021-54485127

深圳地址: 深圳市龙华新区人民北路美丽 AAA 大厦 15 层, 电话: 0755-22193762

成都地址: 成都市武侯区科华北路 99 号科华大厦 6 层, 电话: 028-85405115

南京地址: 南京市白下区汉中路 185 号鸿运大厦 10 层, 电话: 025-86551900

武汉地址: 武汉市工程大学卓刀泉校区科技孵化器大楼 8 层, 电话: 027-87804688

西安地址: 西安市高新区高新一路 12 号创业大厦 D3 楼 5 层, 电话: 029-68785218

华清远见