

KEIL ARM-MDK系列丛书

ARM MCU开发工具 MDK使用入门



李宁 编著



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

KEIL ARM - MDK 系列丛书

ARM MCU 开发工具 MDK 使用入门

李 宁 编著

北京航空航天大学出版社

内 容 简 介

本书介绍 ARM MCU 开发工具 MDK 4. x 的基本使用方法、编译工具、调试工具以及一个简单开发实例。全书的内容分 14 章,可以分为 4 个部分。第 1 部分包括第 1~4 章,是 MDK 的基本入门部分。该部分首先介绍 MDK 的特点、功能和组成,在指导读者完成 MDK 的安装及注册过程之后,全面介绍 μ Vision IDE 环境的基本使用方法。最后给出一个简单工程开发示例。第 2 部分包括第 6~10 章,介绍 ARM 编译工具 RVCT 4. 1。第 3 部分包括第 11~13 章,介绍 μ Vision IDE 的各种工具、调试方法和 Flash 编程器。第 4 部分是第 14 章,介绍一个采用 MDK 实现 STM32 处理器应用开发的例程:RTX_ADC_Blinky。

本书内容丰富,既可作为嵌入式软件开发入门者的教材,又可作为使用 MDK 进行嵌入式软件开发设计人员的参考手册。

图书在版编目(CIP)数据

ARM MCU 开发工具 MDK 使用入门 / 李宁编著. — 北京
: 北京航空航天大学出版社, 2012. 1
ISBN 978 - 7 - 5124 - 0634 - 6

I. ①A… II. ①李… III. ①微处理器, ARM IV.
①TP332

中国版本图书馆 CIP 数据核字(2011)第 231920 号

版权所有,侵权必究。

ARM MCU 开发工具 MDK 使用入门

李 宁 编 著

责任编辑 苗长江 王 彤

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:emsbook@gmail.com 邮购电话:(010)82316936

北京时代华都印刷有限公司印装 各地书店经销

*

开本:787×960 1/16 印张:28 字数:627 千字

2012 年 1 月第 1 版 2012 年 1 月第 1 次印刷 印数:4 000 册

ISBN 978 - 7 - 5124 - 0634 - 6 定价:49.00 元

若本书有倒页、脱页、缺页等印装质量问题,请与本社发行部联系调换。联系电话:(010)82317024

前 言

2007年初,ARM 中国公司委托我来撰写一本关于 MDK 的工具书,当时流行的 ARM 开发工具是 ADS,而 μ Vision 在国内一直是开发 8051 单片机应用的主流工具。此时我们拿到的 MDK 是 μ Vision 3.0 β 版,其中可以选择 Keil C 编译器或 ARM 编译器,由于我个人没有能领会到 ARM 在 MCU 上的发展远景,因此选择以 Keil C 编译器为基础来写作初稿。当出版社即将完成排版工作之际,却发现 μ Vision 3.0 正式版只支持 ARM 编译器 RVCT;在与 ARM 中国前总裁谭军博士沟通之后,我个人开始对 ARM 公司在 MCU 方向的宏伟目标有了初识,因此放弃了前稿,以介绍 ARM 编译器 RVCT 为主体来重新开始写作,终于在 2008 年初完成了新的初稿。由于时间非常仓促,第 1 版中确实存在着较多的文字错误,在一些文字的翻译和表述上也过于生硬。

《ARM 开发工具 RealView MDK 使用入门》第 1 版出版之后,我收到了许多读者的来信,很多是讨论技术细节的。我和我的研究生与这些读者讨论了非常多关于 μ Vision 及 RVCT 的技术细节。这些读者的真知灼见让我们受益匪浅,我的研究生也为此写了很多技术短文发表在各种技术网站上为读者解惑。也有一些读者觉得此书真的不是一本“入门书”,其实本书确实不是一本入门书,而是一本技术指南。谭军博士也希望此书能作为 ARM 编译工具 RVCT 的指南,因为当时还没有一本关于 RVCT 的中文书或者手册。之所以使用“入门”作为名称,当时是考虑了出版社的建议,为了保持一致性,新版依然沿用“入门”作为书名。在此,我要对所有的读者表示衷心的感谢!对为第 1 版做出大量辛勤工作的研究生们表示感谢!

自从 2008 年末,32 位 MCU 的出货量首次全面超过了 8 位和 16 位处理器以来,32 位 MCU 就一直保持着高速增长的势头,这个增长与 ARM 推出的低值 MCU 有着密切的联系。从本书第 1 版开始写作的 2007 年到现在,随着 Cortex - M3、Cortex - M0 和 Cortex - M4 产品的相继发布,ARM MCU 取得惊人的业绩,其 MCU 的出货量以每年 2.4 倍的速度增长。ARM 公司预测,2009 ~ 2013 年间,ARM MCU 的复合增长率将达到 160%。过去一直采用专有内核的日系 MCU 也纷纷开发出基于 ARM 内核的 MCU。因此可以预见,未来的 MCU 市场可能会进入“ARM 标准”时代。ARM MCU 的发展,也带来了 ARM 工具的快速发展,新工具层出不穷,例如 ARM 与 NXP 联合推出的 mbed 快速



原型开发工具、免费的 Coocox Tools。MDK 也从 3.0 升级到了 4. x,除了编译器升级到了 RVCT 4.1,还新增加了一些更强大的功能,已经与第 1 版所介绍的 μ Vision 3.0 有了很大的变化。

2010 年夏天,在一次 ARM 研讨会上巧遇了北京航空航天大学出版社的胡晓柏主任,他建议我根据 MDK 最新的版本撰写修订版。尽管写书是件苦差事,但是第 1 版的很多缺憾还是让我接受了这个建议。

新版将以 μ Vision 4.1 为对象来介绍 MDK,其基本结构仍与第 1 版相同。其中增加了对 CMSIS 的介绍、用最新的 RVCT 4.1 替代了 RVCT 3.0 的介绍、增加了实时跟踪调试的介绍。考虑到实时内核 RTX 的相对独立性,第 2 版删除了对 RTX 的详细介绍,仅在实例中给出了 RTX 的简单使用方法。为了让更多 ARM MCU 用户能立即入门,在最后一章的简单应用举例中,使用当前国内最流行的 STM32 处理器作为对象介绍,替换了过去的 ARM9 处理器。

在本书的写作过程中得到多方支持和帮助。首先是深圳市英蓓特信息技术有限公司的支持,作为 ARM 工具关键合作伙伴,近年来他们总是在第一时间为作者提供最新的 MDK 和开发板,并在技术上给作者提供大量的无私帮助。在此要对英蓓特公司的刘炽、廖武、周麒、刘鑫、张斌等资深工程师表示感谢。其次,要感谢武汉理工大学计算机科学与技术学院 UP 团队的硕士研究生:王冲、段义鹏、宋薇、冯义力、张国琛、张孟东、范云龙、王博、李明、成虎超、卢涛、姚金波。他们完成了大量而繁杂的资料收集、翻译和整理工作,本书是他们汗水的结晶。最后要感谢北京航空航天大学出版社的董立娟编辑,她在本书的内容安排及出版等方面给了作者大量有益的建议和帮助。本书的第 1~11 章由武汉理工大学计算机学院李宁撰写,第 12~14 章由武汉理工大学网络信息中心的何汉青撰写。另外,本书大量使用和借鉴了 ARM 公司网站的内容、MDK 软件和 RVCT 手册的内容,这些已经得到了 ARM 公司的授权。

李 宁 博士

武汉理工大学 计算机科学与技术学院

2011 年 8 月

目 录

第 1 章 MDK 概述	1
1.1 μ Vision IDE	1
1.2 RealView 编译工具集	2
1.2.1 armcc	2
1.2.2 armasm	3
1.2.3 armlink	3
1.2.4 armar	3
1.3 实时库	3
1.4 调试器	4
1.5 软件开发流程	5
第 2 章 μVision 4 的安装与配置	6
2.1 安装的最小系统要求	6
2.2 MDK 的安装	6
2.3 MDK 目录结构	9
2.4 注册与帮助	9
第 3 章 μVision IDE	13
3.1 菜单栏、工具栏、快捷键	14
3.1.1 File 菜单	15
3.1.2 Edit 菜单	15
3.1.3 View 菜单	17
3.1.4 Project 菜单	19
3.1.5 Flash 菜单	20
3.1.6 Debug 菜单	20
3.1.7 Peripherals 菜单	21



3.1.8	Tool 菜单	22
3.1.9	SVCS 菜单	22
3.1.10	Window 菜单	22
3.1.11	Help 菜单	22
3.1.12	Status 栏	23
3.2	工程窗口	24
3.2.1	工程页	24
3.2.2	Regs 页	25
3.2.3	Books 页	26
3.2.4	Functions 页	27
3.2.5	Templates 页	27
3.3	编辑窗口	28
3.4	输出窗口	29
3.4.1	Build 输出窗口	29
3.4.2	Command 窗口	30
3.4.3	Find In Files 输出窗口	30
3.4.4	内存窗口	30
3.4.5	观测窗口	31
3.4.6	调用栈窗口	33
3.4.7	外设对话框	33
3.4.8	分析输出窗口	34
第 4 章	第一个 MDK 工程	36
4.1	CMSIS 标准	36
4.1.1	基于 CMSIS 标准的软件架构	37
4.1.2	CMSIS 代码规范	38
4.1.3	CMSIS 文件结构	40
4.2	工程创建	47
4.2.1	选择工具集	47
4.2.2	创建工程并选择处理器	48
4.2.3	配置处理器启动代码	49
4.2.4	硬件选项配置	50
4.2.5	创建源文件及文件组	52
4.2.6	编译链接工程	54

4.3 工程调试	55
4.3.1 选择调试方式及调试器	55
4.3.2 配置 ULINK2 调试适配器	56
4.3.3 开始调试	56
4.4 工程下载	58
4.4.1 生成 HEX 文件	58
4.4.2 配置 Flash 编程工具及算法	59
4.4.3 工程下载	61
4.5 建立多工程	61
4.5.1 创建多工程文件	62
4.5.2 多工程文件的管理	62
4.5.3 多工程文件的编译	63
4.6 常用技巧	63
4.6.1 编译后启用外部工具	64
4.6.2 为目标和列表文件指定单独的文件夹	64
4.6.3 管理处理器数据库	65
4.6.4 创建一个库文件	67
4.6.5 文件和文件组属性	67
4.6.6 文件扩展名	69
4.6.7 选择始终包含的库模块	70
4.6.8 使用其他编译器	70
第 5 章 μVision 4 工具	72
5.1 配置向导	72
5.1.1 配置菜单	72
5.1.2 配置向导示例	73
5.2 文本搜索	75
5.3 源浏览器	76
5.4 定制菜单运行外部程序	79
5.5 版本控制	80
5.6 语法检查工具 PC-Lint	82
5.7 键码序列	83



第 6 章 RealView 汇编器 armasm	85
6.1 ARM 汇编语言基础	85
6.1.1 ARM 处理器架构版本	85
6.1.2 指令集简介	86
6.1.3 处理器模式	87
6.1.4 寄存器	87
6.2 汇编程序的基本格式与结构	88
6.2.1 汇编源语句行格式	88
6.2.2 汇编语言程序的基本结构	88
6.2.3 子程序	89
6.2.4 宏	90
6.3 源程序汇编过程	92
6.3.1 环境设置	92
6.3.2 命令与提示	92
6.3.3 汇编输出文件	93
6.4 汇编命令选项	96
6.5 符号、运算符与表达式	105
6.5.1 符号	105
6.5.2 运算符	108
6.5.3 表达式	112
第 7 章 RealView 编译器 armcc	114
7.1 编译源程序	114
7.1.1 环境设置	114
7.1.2 命令与提示	115
7.1.3 文件命名约定和搜索约定	116
7.1.4 编译输出文件	117
7.2 编译命令选项	120
7.3 armcc 编译器特性	151
7.3.1 关键字	151
7.3.2 __declspec 属性	162
7.3.3 函数、类型、变量的属性	164
7.3.4 pragmas	171

7.3.5	内在函数(intrinsics)	173
7.3.6	预定义宏	178
7.4	内联汇编与内嵌汇编	182
7.4.1	内联汇编	182
7.4.2	内嵌汇编	188
7.4.3	访问旧内联汇编代码中 sp、lr 或 pc	194
7.4.4	内联汇编与内嵌汇编的区别	195
7.5	高级编程管理和技巧	196
7.5.1	预编译头文件	196
7.5.2	代码优化	199
7.5.3	内 联	204
7.5.4	函 数	205
7.5.5	程序复杂度度量	208
7.5.6	使用浮点运算	209
第 8 章	RealView 链接器 armlink	211
8.1	链接器 armlink 的使用	211
8.1.1	环境设置	211
8.1.2	链接模型	212
8.1.3	armlink 命令行及输入输出	212
8.2	链接命令	214
8.2.1	获取帮助和信息类命令选项	214
8.2.2	库控制类命令选项	214
8.2.3	控制链接器行为类选项	215
8.2.4	设置输出文件类选项	216
8.2.5	设置内存映射类选项	216
8.2.6	控制调试信息类选项	218
8.2.7	控制映像内容类选项	219
8.2.8	控制胶合代码类选项	224
8.2.9	字节寻址模式类选项	225
8.2.10	控制映像相关信息类选项	225
8.2.11	控制链接器诊断信息类选项	228
8.2.12	使用 via 文件选项	229
8.2.13	杂 项	230



8.3	链接器的基本功能	230
8.3.1	指定映像结构	230
8.3.2	节布局	233
8.3.3	优化和修正	234
8.3.4	使用命令行选项创建简单映像	242
8.3.5	使用命令行选项来处理 C++ 异常	244
8.3.6	链接器生成映像的信息	245
8.4	映像符号的定义与访问	246
8.4.1	ARM/Thumb 同义词	246
8.4.2	链接器定义的符号	247
8.4.3	访问其他映像中的符号	249
8.4.4	隐藏和重命名全局符号	250
8.4.5	用 \$ Super \$ 和 \$ Sub \$ 覆盖符号定义	253
8.5	分散加载描述文件	253
8.5.1	分散加载机制	254
8.5.2	分散加载描述文件的语法	257
第 9 章	RealView 库及库管理器	268
9.1	RealView 库	268
9.1.1	环境设置	268
9.1.2	ARM C、C++ 运行时库	269
9.1.3	Thumb C 库	269
9.1.4	浮点支持库	269
9.2	C 微库	270
9.2.1	与默认 C 库的差异	270
9.2.2	使用微库建立应用程序	271
9.3	库管理器 armar	271
9.3.1	库的搜索与扫描	272
9.3.2	库管理器 armar	273
第 10 章	映像文件转换器 fromELF	277
10.1	fromELF 命令	277
10.2	使用 μ Vision4 IDE 生成 HEX 文件	287

第 11 章	μVision IDE 工具	289
11.1	配置向导	289
11.1.1	配置菜单	290
11.1.2	配置向导示例	290
11.2	文本搜索	293
11.3	源浏览器	293
11.4	键码序列	296
11.5	定制用户菜单	297
11.6	版本控制	298
11.7	语法检查工具 PC-Lint	300
第 12 章	软件调试	303
12.1	调试器的使用	304
12.1.1	启动调试模式	304
12.1.2	应用程序的执行	304
12.1.3	CPU 模拟	304
12.2	调试窗口及对话框	305
12.2.1	断点(Breakpoint)对话框	305
12.2.2	代码覆盖(Code Coverage)窗口	308
12.2.3	寄存器(Regs)页	308
12.2.4	反汇编窗口	308
12.2.5	内存窗口及内存映射对话框	309
12.2.6	执行剖析器	311
12.2.7	性能分析仪	312
12.2.8	串行窗口	312
12.2.9	符号窗口	313
12.2.10	工具箱	314
12.2.11	观察窗口	314
12.2.12	调用栈窗口	314
12.2.13	命令窗口	315
12.2.14	指令跟踪窗口	316
12.2.15	系统观测窗口	316
12.3	逻辑分析仪	317



12.3.1	逻辑分析仪的配置	319
12.3.2	信号配置举例	320
12.3.3	限制	321
12.4	调试命令	321
12.4.1	调试命令分类	321
12.4.2	调试命令详解	323
12.5	表达式	344
12.5.1	常量	345
12.5.2	系统变量	346
12.5.3	外围设备变量	346
12.5.4	程序变量	349
12.5.5	行号	351
12.5.6	位地址	352
12.5.7	类型说明	352
12.5.8	操作符	352
12.5.9	μ Vision IDE 和 C 中表达式的差异	352
12.5.10	表达式实例	353
12.6	调试函数	355
12.6.1	函数的建立及调用	355
12.6.2	预定义调试函数	356
12.6.3	用户函数	366
12.6.4	信号函数	367
12.6.5	调试函数与 C 之间的差异	369
12.7	信号函数模板	370
12.7.1	数字信号输入	370
12.7.2	模拟信号输入	374
12.7.3	UART 通信	380
12.7.4	CAN 总线	382
12.7.5	I ² C 总线	387
12.7.6	SPI 接口	391
第 13 章	FLASH 编程器	396
13.1	Flash 菜单配置	396
13.2	配置初始化文件	399

13.2.1 总线配置.....	399
13.2.2 加载附加程序.....	400
第 14 章 基于 STM32 的简单应用开发.....	401
14.1 EM-STM3210E 开发板	401
14.1.1 硬件配置.....	401
14.1.2 基本结构.....	403
14.1.3 连接器、跳线	404
14.1.4 外设及 I/O 接口	404
14.2 硬件电路.....	407
14.3 RTX 内核	408
14.4 创建应用程序 RTX_ADCBlinky	409
14.4.1 创建新工程 RTX_ADCBlink	409
14.4.2 添加用户源程序.....	412
14.5 模拟仿真调试 RTX_ADCBlinky	420
14.5.1 模拟外部输入信号.....	420
14.5.2 模拟 UART 通信	422
14.5.3 查看片上外围设备状态.....	422
14.5.4 逻辑分析仪查看输入输出信号时序.....	422
14.6 硬件调试 RTX_ADCBlinky	423
14.6.1 配置实时跟踪调试.....	423
14.6.2 实时跟踪调试.....	427
参考文献.....	434

第 1 章

MDK 概述

2006 年 ARM 公司收购了著名的 MCU 开发工具开发商 Keil, 随后推出了用于开发 ARM7、ARM9 和 Cortex - M 处理器的开发工具 MDK (Microcontroller Development Kit)。MDK 是 Keil 公司集成开发环境 μ Vision IDE 与 ARM 公司高效编译工具 RVCT (RealView Complies Tools) 的完美结合。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 主要包含以下 4 个核心组成部分:

- μ Vision IDE: 是一个集项目管理器、源代码编辑器、调试器于一体的强大集成开发环境。
- RVCT: ARM 公司提供的编译工具链, 包含编译器、汇编器、链接器和相关工具。
- RL - ARM: 实时库, 可将其作为工程的库来使用。
- ULINK USB - JTAG 仿真器: 用于连接目标系统的调试接口 (JTAG 或 SWD 方式), 帮助用户在目标硬件上调试程序。

本章将分别对这 4 个部分进行简要介绍。

1.1 μ Vision IDE

μ Vision IDE 是一个基于 Windows 操作系统的嵌入式软件开发平台, 集编辑器、调试器、项目管理器和一些 Make 工具等于一体。其主要特征如下:

- 项目管理器, 用于产生和维护项目。
- 处理器数据库, 集成了一个能自动配置选项的工具。
- 带有用于汇编、编译和链接的 Make 工具。
- 全功能的源码编辑器。
- 模板编辑器可用于在源码中插入通用文本序列和头部块。
- 源码浏览器用于快速寻找、定位和分析应用程序中的代码和数据。
- 函数浏览器用于在程序中对函数进行快速导航。
- 函数略图 (Function Outlining) 可形成某个源文件的函数视图。



- 还带有一些内置工具,例如“Find in Files”等。
- 集模拟调试和目标硬件调试于一体。
- 配置向导可实现图形化地快速生成启动文件和配置文件。
- 可与多种第三方工具和软件版本控制系统接口。
- 带有 Flash 编程工具对话框。
- 丰富的工具设置对话框。
- 完善的在线帮助和用户指南。

1.2 RealView 编译工具集

RealView 编译工具集 RVCT 是 ARM 公司多年发展所积累的成果,RVCT 在业界被认为是面向 ARM 技术的编译器中能够提供最佳性能的编译工具。RVCT 的开发致力于高性能和高代码密度,以降低产品成本。RVCT 编译器能生成优化的 32 位 ARM 指令集、16 位的 Thumb 指令集以及最新的 Thumb-2 指令集代码,完全支持 ISO 标准 C 和 C++。

MDK 的 RealView 编译工具集用于将 C/C++ 源文件转换为可重定位的目标模块,并生成 μ Vision IDE 调试器可用的调试信息。其主要包括以下几个核心部分:

- armcc: ARM C/C++ 编译器。
- armasm: ARM 宏汇编器。
- armLink: ARM 链接器。
- 其他工具: 库管理器 armar、十六制文件产生器 FromELF。

下面将分别对这几部分做简要介绍。

1.2.1 armcc

ARM C/C++ 编译器的主要特征:

- 支持同一源文件中的 ARM 和 Thumb 混合模式。
- 工业领先的代码尺寸优化技术,可产生最小尺寸的编译代码。
- 工业领先的性能优化技术,在不增加时钟频率的情况下最大化处理器的性能。
- 具有“硬件支持”函数属性,为访问 ARM 硬件提供方便。
- 支持内嵌汇编,可用于快速 DSP 或其他信号处理算法。
- 函数内联特性,可加快被频繁调用函数的执行速度。
- 可自动通过 CPU 寄存器传递参数,甚至一些小的 C 结构也可通过 CPU 寄存器传递和返回,加快了执行速度。
- 可重入的运行时库,程序段多数可重入,既可从主程序中调用,也可在中断中调用。
- 遵从单精度、双精度数的标准 IEEE 754 标准,以用于高精度的浮点计算。

1.2.2 armasm

ARM 宏汇编器的主要特征：

- 标准的宏处理。
- 条件汇编,可从同一源文件得到不同的目标文件。
- 符号引用列表文件包含可选的符号交叉引用信息,以提供源文件的详细信息。

1.2.3 armlink

ARM 链接器的主要特征：

- 链接器产生的详细列表文件非常易于用户理解。它包含内存配置、输入模块、内存映像、符号表和交叉引用信息。
- 全局代码列表文件,包含由链接器产生的符号反汇编信息。
- 静态堆栈分析,帮助链接器在链接时处理堆栈请求。

1.2.4 armar

ARM 库管理器的主要特征：

- 对库文件进行模块管理,为链接器组合、引用多个模块提供方便。 μ Vision IDE 也可生成库文件。
- 变量和函数引用。可从库中抽取所需的模块,模块中的代码段如果未在应用中被使用,则它们不会被包含在最终的输出中。
- 安全、快速、磁盘空间占用小。库为分布在初始源代码中的大量函数和程序段提供了一种载体。

1.3 实时库

RealView 实时库(RTL, Real-Time Library)是为解决基于 ARM MCU 的嵌入式系统中实时及通信问题而设计的紧密耦合库集合。其主要特征如下：

- 实时内核 RTX,免版税使用,但源码需要付费。
- 带有 TCP/IP 网络协议族,完整的嵌入式网络协议族。
- 带有 Flash 文件系统,可在内存和存储系统中产生、修改文件。
- 带有 CAN 协议,通用 ARM MCU 设备的 CAN 驱动。
- 带有 USB 协议,适用于标准 Windows 设备类。
- 带有例子和模板,可帮助用户快速开始使用 RL-ARM 组件。
- 在 MDK 中集成了对 RTL 的配置以及一些工具。