

21世纪高等学校嵌入式系统专业规划教材

王黎明 夏立 卜乐平 闫晓玲 编著

# 嵌入式系统开发与应用 实验指导书： 基于Cortex-M3内核的 Stellaris微控制器

清华大学出版社



21世纪高等学校嵌入式系统专业规划教材

王黎明 夏立 卜乐平 闫晓玲 编著

**嵌入式系统开发与应用**  
**实验指导书：**  
**基于Cortex-M3内核的**  
**Stellaris微控制器**



清华大学出版社  
北京

## 内 容 简 介

本书从夯实基础、注重实用、力求可操作出发,首先引导读者对嵌入式系统有一个整体的印象和把握,从而引导读者进入嵌入式系统开发的大门;其次介绍了为程序的编译、测试提供了必要的基础、环境、工具支持的嵌入式开发的集成开发环境及驱动库;然后介绍了本书的实验对象 LM3S8962 实验平台硬件电路,相关外围接口的实验,包括硬件类型的定义位带操作、系统控制接口、系统节拍定时器、通用输入输出接口、Flash 内部总线读写、通用定时器控制、看门狗操作、中断控制、脉宽调制器的应用以及电压比较器、ADC 采样、互连 IC 总线、同步串行通信、CAN 总线接口、以太网等实验。

本书可作为大专、本科院校自动化、机电、仪器仪表、自动控制等专业工业控制网络等相关课程的教材或教学参考书,也可供从事工业控制网络系统设计和产品研发的工程技术人员参考。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

### 图书在版编目(CIP)数据

嵌入式系统开发与应用实验指导书:基于 Cortex-M3 内核的 Stellaris 微控制器/王黎明等编著.--北京:清华大学出版社,2013

21 世纪高等学校嵌入式系统专业规划教材

ISBN 978-7-302-32654-0

I. ①嵌… II. ①王… III. ①微型计算机—系统开发—高等学校—教材 IV. ①TP360.21

中国版本图书馆 CIP 数据核字(2013)第 122437 号

责任编辑:高买花 薛 阳

封面设计:常雪影

责任校对:时翠兰

责任印制:杨 艳

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质 量 反 馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

印 刷 者:北京富博印刷有限公司

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:28.5

字 数:691 千字

版 次:2013 年 9 月第 1 版

印 次:2013 年 9 月第 1 次印刷

印 数:1~2000

定 价:49.00 元

# 出版说明

嵌入式计算机技术是 21 世纪计算机技术两个重要发展方向之一,其应用领域相当广泛,包括工业控制、消费电子、网络通信、科学研究、军事国防、医疗卫生、航空航天等方方面面。我们今天所熟悉的电子产品几乎都可以找到嵌入式系统的影子,它从各个方面影响着我们的生活。

技术的发展和生产力的提高,离不开人才的培养。目前国内外各高等院校、职业学校和培训机构都涉足了嵌入式技术人才的培养工作,高校及其软件学院和专业的培训机构更是嵌入式领域高端人才培养的前沿阵地。国家有关部门针对专业人才需求大增的现状,也着手开发“国家级”嵌入式技术培训项目。2006 年 6 月底,国家信息技术紧缺人才培养工程(NITE)在北京正式启动,首批设定的 10 个紧缺专业中,嵌入式系统设计与软件开发、软件测试等 IT 课程一同名列其中。嵌入式开发因其广泛的应用领域和巨大的人才缺口,其培训也被列入国家商务部门实施服务外包人才培养“千百十工程”,并对符合条件的人才培训项目予以支持。

为了进一步提高国内嵌入式系统课程的教学水平和质量,培养适应社会经济发展需要的、兼具研究能力和工程能力的高质量专业技术人才。在教育部相关教学指导委员会专家的指导和建议下,清华大学出版社与国内多所重点大学共同对我国嵌入式系统软硬件开发人才培养的课程框架和知识体系,以及实践教学内容进行了深入的研究,并在该基础上形成了“嵌入式系统教学现状分析及核心课程体系研究”、“微型计算机原理与应用技术课程群的研究”、“嵌入式 Linux 课程群建设报告”等多项课程体系的研究报告。

本系列教材是在课程体系的研究基础上总结、完善而成,力求充分体现科学性、先进性、工程性,突出专业核心课程的教材,兼顾具有专业教学特点的相关基础课程教材,探索具有发展潜力的选修课程教材,满足高校多层次教学的需要。

本系列教材在规划过程中体现了如下一些基本组织原则和特点。

(1) 反映嵌入式系统学科的发展和专业教育的改革,适应社会对嵌入式人才的培养需求,教材内容坚持基本理论的扎实和清晰,反映基本理论和原理的综合应用,在其基础上强调工程实践环节,并及时反映教学体系的调整和教学内容的更新。

(2) 反映教学需要,促进教学发展。教材要适应多样化的教学需要,正确把握教学内容和课程体系的改革方向,在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养,为学生知识、能力、素质协调发展创造条件。

(3) 实施精品战略,突出重点。规划教材建设把重点放在专业核心(基础)课程的教材建设上;特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版,逐步形成精品教材;提倡并鼓励编写体现工程型和应用型的专业教学内容和课程体系改革成果的教材。

(4) 支持一纲多本,合理配套。专业核心课和相关基础课的教材要配套,同一门课程可以有多个具有各自内容特点的教材。处理好教材统一性与多样化,基本教材与辅助教材、教

学参考书,文字教材与软件教材的关系,实现教材系列资源的配套。

(5) 依靠专家,择优落实。在制定教材规划时依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时,要引入竞争机制,通过申报、评审确定主编。书稿完成后认真实行审稿程序,确保出书质量。

繁荣教材出版事业,提高教材质量的关键是教师。建立一支高水平的、以老带新的教材编写队伍才能保证教材的编写质量,希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21 世纪高等学校嵌入式系统专业规划教材  
联系人: 魏江江 weijj@tup.tsinghua.edu.cn

# 前 言

随着计算机网络技术和芯片技术的进步,控制芯片的性能大幅提高,成本不断降低,网络几乎深入到人们生活的每一个角落,谁是最节能,最擅长把好钢用在刀刃上的人?要让我说,我一定得表一表单片机的开发者。他们使出浑身解数,写出精妙玲珑的代码,把单片机点点滴滴的力量汇集起来,让它如同涌泉一般尽情地迸发,灌溉滋养着各行各业。是什么灵丹妙药赐予了他们这么神奇的力量?除了好的处理器之外,还要配合好的开发环境和工具链。也正出于此,在设计 ARM7TDMI 处理器时,ARM 的工具链工程师们和 CPU 设计师们强强联手,为了让它的内部结构更优化、更精练、更到位而并肩奋战了很多日日夜夜,终于有了 ARM7TDMI 的无限辉煌,并且久经岁月的洗礼依旧光芒绽放。

珠联璧合的最新果实,是破茧而出的 ARM Cortex-M3 处理器。这个小尤物,处处闪耀着 ARM 体系结构激动人心的新突破。它基于最新最好的 32 位 ARMv7 架构——这个架构支持高度成功的 Thumb-2 指令集,还有很多时尚、前卫甚至崭新的特性,充满了新生代的气息。它在很好、很强大的同时,编程模型却变得更加清新爽洁了。不管你是祖国的花朵、是人民教师、还是精明的商人,也无所谓是新手还是骨灰级玩家,Cortex-M3 都将尽情展现它的秀外慧中,带给你喜出望外的收获和“激活”!

## 本书特点

- (1) 起点低,实验例程分析透彻。
- (2) 既注重基础理论,更面向应用。
- (3) 实例可操作性强。

## 本书的组织结构

第 1 章为基础部分,重点介绍了 Cortex-M3 内核的寄存器组、中断向量表、存储器映射、存储器保护单元。本章主要是帮助读者对 Cortex-M3 内核有一个整体的印象把握,从而引导读者进入嵌入式系统开发的大门。

第 2 章介绍了嵌入式开发的集成开发环境及驱动库安装。介绍了本书主要使用的开发工具链 IAR,主要包括 IAR EWARM 的安装、驱动库的安装、EWARM 中新建新项目、编译运行程序,以及将 IAR 的工程移植到 Keil 工程的方法。本章为后面章节中程序的编译、测试提供了必要的基础、环境、工具支持。

第 3 章主要介绍了以 LM3S8962 处理器为核心的实验平台的硬件电路,包括:滤波电路、复位电路、晶振时钟、以太网接口、RS232、RS485、CAN 总线、电源模块、ADC 采样、LED 及按键、蜂鸣器、IIC 扩展开关量输入、IIC 扩展开关量输出、继电器输出等电路设计。硬件电路是整个嵌入系统的躯体骨架。

第 4 章为处理器主要接口实验,包括硬件类型的定义位带操作、系统控制接口、系统节拍定时器、通用输入输出接口、Flash 内部总线读写、通用定时器控制、看门狗操作、中断控

制、脉宽调制器的应用以及电压比较器的实验。其中，Cortex-M3 内核的系统节拍定时器、中断控制、位带操作是其处理器与其他处理器很大的不同点，读者应认真体会分析，掌握其应用。软件是嵌入式系统的灵魂，只有软件牢牢地依附于硬件躯体，才会有其精气神。

第 5 章为嵌入式技术中模拟量的采集变换实验，主要包括 ADC 单通道触发采样、多通道采样、内部温度传感器采样、定时器触发采样、外部触发采样等实验，最后介绍了 ADC 的硬件过采样和软件过采样实验。

第 6 章为 UART 异步串口通信实验，主要包括：简单收发、发送 FIFO 工作原理、发送 FIFO 中断原理、以 FIFO 中断方式发送、以 FIFO 中断方式接收等实验。通过本章实验的学习，读者可以实现异步通信。

第 7 章为互连 IC 总线实验，主要包括：IIC 的 I/O 扩展芯片 PCF8574T、I/O 扩展芯片 PCA9554、IIC 存储器器件 CAT24C02 的读取等应用实验。

第 8 章为同步串行通信的实验，主要为 SSI 驱动静态 LED 和 SSI 驱动动态 LED 实验的例程。

第 9 章为 CAN 总线接口实验，包括 CAN 总线数据简单发送数据、CAN 总线发送 FIFO 数据、CAN 总线发送 ADC 采样数据、CAN 总线接收数据实验。

第 10 章为物联网的基础——以太网接口通信实验，包括 UDP 服务器通信、UDP 客户端通信、Web 服务器、TCP 客户端通信等实验。

第 11 章为  $\mu\text{C}/\text{OS}-\text{II}$  实时多任务操作系统在处理器上的移植及应用实验，包括任务操作实验、信号量及邮箱通信实验。

第 12 章为 BootLoader 实验，帮助读者从处理器内部的原理上学习 BootLoader 的驱动操作过程。

第 13 章为综合应用实验，包括 TFT-LCD 驱动、传感器 DS1820 测温、数字传感器 LM75A 测量温度、直流电机控制、综合显示等实验。

本书为每个例程都配有部分思考题，从而帮助读者更好地熟悉嵌入式系统接口开发方法。

### 读者对象

本书是嵌入式系统开发与应用——基于 Cortex-M3 内核的 Stellaris 处理器的配套实验指导书，当然本书也可以自成一册参考书。目标读者包括：一线程序员，嵌入式产品设计师，片上系统(SoC)工程师，嵌入式系统发烧友，学院研究员，还包括所有涉猎过单片机和微处理器领域，慧眼识珠看中了 Cortex-M3 的读者。同时也适合下列人员阅读。

- (1) 想学习或刚刚进入 Cortex-M3 内核 Stellaris 处理器的开发人员。
- (2) 想学习嵌入式技术的开发人员。
- (3) 对嵌入式技术开发感兴趣的人员。
- (4) 使用 Cortex-M3 进行快速开发产品的开发人员。

尽管本书面向初级 Cortex-M3 内核 Stellaris 处理器的开发人员，但读者需要熟悉相关的硬件知识以及 C 语言，至少能读懂书中提到的代码。

## 致谢

在本书的编写过程中,得到了很多人的支持和热心关注,在这里表示衷心的感谢。

其次感谢周立功单片机发展有限公司,为本书的完成提供了非常完美的测试环境,并提供了相应的硬件测试平台,同时本书部分内容及代码是由该公司提供。

因作者水平和编写书稿时间的限制,书中难免有疏漏和不妥之处,恳请广大读者批评指正。作者联系方式为 icesoar@163.com。

王黎明

2013年6月7日晚于武汉



# 目 录

<b>第 1 章 ARM 微处理器概述</b> .....	1
1.1 ARM 简介 .....	1
1.2 ARM 微处理器的应用领域及特点 .....	1
1.3 Cortex-M3 结构处理器概览 .....	4
<b>第 2 章 集成开发环境及驱动库</b> .....	11
2.1 嵌入式软件开发过程 .....	11
2.2 IAR 集成开发环境使用 .....	13
2.3 安装驱动库 .....	15
2.3.1 下载最新库文件 .....	15
2.3.2 复制连接器命令文件 .....	16
2.3.3 复制驱动库头文件 .....	17
2.3.4 复制底层驱动函数库 .....	19
2.4 EWARM 新建新项目 .....	20
2.4.1 建立一个项目文件目录 .....	20
2.4.2 新建工作区 .....	20
2.4.3 生成新项目 .....	21
2.4.4 添加/新建文件 .....	23
2.4.5 项目选项设置 .....	26
2.4.6 通用选项设置 .....	27
2.4.7 C/C++ 编译器选项设置 .....	27
2.4.8 Assembler 选项设置 .....	29
2.4.9 Linker 选项设置 .....	29
2.4.10 Debugger 选项设置 .....	31
2.5 编译和运行应用程序 .....	34
2.5.1 编译连接处理 .....	34
2.5.2 查看 MAP 文件 .....	35
2.5.3 加载应用程序 .....	35
2.5.4 应用程序的相关调试 .....	36
2.5.5 生成 hex 文件 .....	39
2.5.6 探究编译连接 .....	40
2.6 将 IAR 的工程移植到 Keil 工程 .....	43

<b>第 3 章 实验平台电路结构</b> .....	46
3.1 LM3S8962 接口特性 .....	46
3.2 LM3S8962 结构图 .....	50
3.3 LM3S8962 处理器引脚描述 .....	52
3.4 LM3S8962 最小系统电路图 .....	55
3.5 实验开发平台电路分析 .....	58
3.5.1 主处理器电路 .....	58
3.5.2 处理器滤波电路 .....	59
3.5.3 复位接口电路 .....	59
3.5.4 主处理器时钟接口 .....	59
3.5.5 以太网接口 .....	59
3.5.6 实时时钟及供电电路 .....	59
3.5.7 IIC 存储器 EEPROM 电路 .....	60
3.5.8 IIC 数字温度采集接口 .....	61
3.5.9 IIC 扩展开关量输入电路 .....	61
3.5.10 开关量隔离输入电路 .....	61
3.5.11 IIC 扩展开关量输出电路 .....	61
3.5.12 开关量隔离输出电路 .....	61
3.5.13 继电器输出电路组 1 .....	63
3.5.14 继电器输出电路组 2 .....	63
3.5.15 IIC 扩展输入输出电路 .....	63
3.5.16 UART 串行 RS232 电路 .....	63
3.5.17 UART 串行 RS485 及 CAN 总线接口电路 .....	63
3.5.18 红外线串口电路 .....	65
3.5.19 ADC 采样电路 .....	65
3.5.20 LED 及按键电路 .....	65
3.5.21 SPI 驱动 7LED 接口电路 .....	67
3.5.22 SPI 驱动 EEPROM 接口电路 .....	68
3.5.23 SPI 驱动 TFT LCD 接口电路 .....	68
3.5.24 SPI 的 ADC 采样电路 .....	69
3.5.25 PWM 驱动电机电路 .....	69
3.5.26 JTAG 接口电路 .....	71
3.5.27 外部接口电路 .....	71
3.5.28 扩展接口电路 .....	71
3.6 实验平台概述 .....	72
3.6.1 实验平台外观 .....	72
3.6.2 实验平台主要接口描述 .....	72
3.6.3 主要功能资源 .....	73

第 4 章 接口功能单元及开发实验 .....	75
4.1 嵌入式编程 .....	75
实验 4.1.1 GPIO 驱动 LED 闪烁发光实例 .....	75
实验 4.1.2 输出 hello world 实验 .....	82
实验 4.1.3 UART 控制台实验 .....	93
4.2 系统控制实验 .....	102
实验 4.2.1 电源结构与 LDO 控制 .....	102
实验 4.2.2 处理器主时钟控制 .....	107
实验 4.2.3 处理器复位控制 .....	114
实验 4.2.4 硬件寄存器访问 .....	118
实验 4.2.5 位带操作 .....	120
4.3 系统节拍实验 .....	126
实验 4.3.1 系统节拍定时实验 .....	126
实验 4.3.2 系统节拍中断实验 .....	130
实验 4.3.3 模拟 PC 按键重复特性实验 .....	133
4.4 通用 GPIO 输入输出 .....	136
实验 4.4.1 两只 LED 交替闪烁控制实验 .....	136
实验 4.4.2 KEY 控制 LED 实验 .....	143
4.5 中断控制实验 .....	147
实验 4.5.1 GPIO 中断控制实验 .....	147
实验 4.5.2 中断优先级控制实验 .....	151
4.6 Flash 操作控制实验 .....	155
实验 4.6.1 Flash 读、写操作实验 .....	155
实验 4.6.2 Flash 当作 EEPROM 操作实验 .....	159
4.7 定时器 Timer 实验 .....	163
实验 4.7.1 32 位单次触发定时实验 .....	163
实验 4.7.2 32 位周期定时实验 .....	171
实验 4.7.3 16 位单次触发定时实验 .....	173
实验 4.7.4 16 位周期触发定时实验 .....	176
实验 4.7.5 16 位输入边沿计数捕获实验 .....	179
实验 4.7.6 16 位输入边沿定时捕获实验 .....	183
实验 4.7.7 定时器 16 位 PWM 实验 .....	191
实验 4.7.8 Timer PWM 应用蜂鸣器发声实验 .....	193
实验 4.7.9 Timer 模块 16 位 PWM 演奏音乐实验 .....	196
4.8 看门狗实验 .....	201
实验 4.8.1 看门狗复位实验 .....	201
实验 4.8.2 看门狗作为普通定时器实验 .....	206
4.9 PWM 操作控制实验 .....	209

实验 4.9.1 产生两路 PWM 波实验 .....	209
实验 4.9.2 PWM 发生器中断实验 .....	215
实验 4.9.3 产生两路带死区的 PWM 实验 .....	218
4.10 模拟比较器操作实验 .....	221
实验 4.10.1 模拟比较器内部参考电源 .....	221
实验 4.10.2 外部参考电源输出触发中断实验 .....	226
<b>第 5 章 ADC 采集实验 .....</b>	<b>230</b>
5.1 处理器触发 ADC 采样实验 .....	230
5.2 ADC 内置位温度传感器实验 .....	236
5.3 定时器溢出触发 ADC 采样实验 .....	242
5.4 模拟比较器触发 ADC 采样 .....	247
5.5 差分输入 ADC 采样实验 .....	251
5.6 ADC 硬件过采样实验 .....	258
5.7 ADC 软件过采样实验 .....	263
<b>第 6 章 UART 通信实验 .....</b>	<b>268</b>
6.1 UART 简单收发实验 .....	268
6.2 发送 FIFO 实验 .....	277
6.3 发送 FIFO 中断实验 .....	280
6.4 以 FIFO 中断方式发送实验 .....	285
6.5 以 FIFO 中断方式接收实验 .....	289
<b>第 7 章 互连 IC 总线实验 .....</b>	<b>294</b>
7.1 I/O 扩展芯片 PCA9554 控制实验 .....	294
7.2 IIC 总线器件 EEPROM 存储器 CAT24C02 读写 .....	298
7.3 I/O 扩展芯片 PCF8574T 输出控制实验 .....	303
7.4 I/O 扩展芯片 PCF8574T 输入控制实验 .....	309
<b>第 8 章 SSI 通信实验 .....</b>	<b>316</b>
8.1 SSI 驱动静态 LED 实验 .....	316
8.2 SSI 驱动动态 LED 实验 .....	320
<b>第 9 章 CAN 现场总线收发实验 .....</b>	<b>325</b>
9.1 CAN 总线简单发送数据实验 .....	325
9.2 CAN 总线发送 FIFO 数据实验 .....	337
9.3 CAN 总线发送 ADC 采样数据实验 .....	341
9.4 CAN 总线接收数据实验 .....	347

---

<b>第 10 章 以太网接口实验</b> .....	352
10.1 UDP 服务器通信实验 .....	352
10.2 UDP 客户端通信实验 .....	356
10.3 Web 服务器实验 .....	360
10.4 TCP 客户端通信实验 .....	364
<b>第 11 章 <math>\mu</math>C/OS 嵌入式实时操作系统</b> .....	370
11.1 $\mu$ C/OS-II 模板使用——点亮 LED 实验 .....	370
11.2 $\mu$ C/OS-II 模板使用——中断控制蜂鸣器 .....	376
11.3 信号量通信实验 .....	381
11.4 邮箱通信实验 .....	385
<b>第 12 章 BootLoader 实验</b> .....	391
12.1 实验目的 .....	391
12.2 实验内容 .....	391
12.2.1 BootLoader 的启动 .....	391
12.2.2 BootLoader 的配置 .....	395
12.2.3 Flash 底层操作 .....	398
12.2.4 使用 UART 的 BootLoader 下载 .....	398
12.3 思考题 .....	403
<b>第 13 章 应用相关实验</b> .....	404
13.1 TFT-LCD 驱动实验 .....	404
13.2 传感器 DS18B20 测温实验 .....	416
13.3 数字传感器 LM75A 测量温度实验 .....	424
13.4 直流电机控制实验 .....	431
13.5 综合显示实验 .....	436
<b>参考文献</b> .....	440

# 第 1 章 ARM 微处理器概述

嵌入式处理器有多种内核结构,包括 ARM、PowerPC、MIPS 等,其中 ARM 内核的嵌入式处理器占 32 位处理器市场 75% 的份额。本节简介 ARM 微处理器的一些基本概念、应用领域及其特点,以引导读者进入 ARM 技术的殿堂。

## 1.1 ARM 简介

目前,采用 ARM 技术知识产权核的微处理器,即人们通常所说的 ARM 微处理器,已遍及工业控制、消费类电子产品、通信系统、网络系统、无线系统等各类产品市场,基于 ARM 技术的微处理器应用约占据了 32 位 RISC 微处理器 75% 以上的市场份额,ARM 技术正在逐步渗入人们生活的各个方面。

ARM 公司是专门从事基于 RISC 技术芯片设计开发的公司,作为知识产权供应商,它本身并不直接从事芯片生产,而是靠转让设计许可由合作公司生产各具特色的芯片。世界各大半导体生产商从 ARM 公司购买其设计的 ARM 微处理器 IP 核,根据各自不同的应用领域,加入适当的外围电路,从而形成自己的 ARM 微处理器芯片进入市场。目前,全世界有几十家大的半导体公司都在使用 ARM 公司的授权,因此使得 ARM 技术更容易获得更多的第三方工具、制造、软件的支持,又使整个系统成本降低,使产品更容易进入市场被消费者所接受,更具有竞争力。

首先回顾一下 ARM 的进化史,以了解为何会有品种如此之多的 ARM 处理器和 ARM 架构。

ARM 公司于 1990 年成立,当初的名字是 Advanced RISC Machines Ltd,当时它由三家公司合资创建——苹果、Acorn 以及 VLSI 技术(公司)。1991 年,ARM 推出了 ARM6 处理器家族,VLSI 则是第一个吃螃蟹的人。后来,陆续有其他巨头:包括 TI、NEC、Sharp、ST 等都获取了 ARM 授权,它们真正把 ARM 处理器大面积地铺开,使得 ARM 处理器在手机、硬盘控制器、PDA、家庭娱乐系统以及其他消费电子市场中都大展雄才。

如今的 ARM(Advanced RISC Machines)既可以认为是一个公司的名字,也可以认为是对一类微处理器的通称,更可以认为是一种技术的名字,是强有力的“胳膊”,也做为“武器”向 8 位/16 位微处理器提出了挑战。

## 1.2 ARM 微处理器的应用领域及特点

### 1. ARM 微处理器的应用领域

到目前为止,ARM 微处理器及技术的应用几乎已经深入到各个领域。

(1) 工业控制领域。作为 32 位的 RISC 架构,基于 ARM 核的微控制器芯片不但占据

了高端微控制器市场的大部分份额,同时也逐渐向低端微控制器应用领域不断扩展,ARM 微控制器的低功耗、高性价比,向传统的 8 位/16 位微控制器提出了挑战。

(2) 无线通信领域。目前已有超过 85% 的无线通信设备采用了 ARM 技术,ARM 以其高性能和低成本的特点,在该领域的地位日益巩固。

(3) 网络应用。随着宽带技术的推广,采用 ARM 技术的 ADSL 芯片正逐步获得竞争优势。此外,ARM 在语音及视频处理上也进行了优化,并获得广泛支持,这也对 DSP 的应用领域提出了挑战。

(4) 消费类电子产品。ARM 技术在目前流行的数字音频播放器、数字机顶盒和游戏机中得到广泛应用。

(5) 成像和安全产品。现在流行的数码相机和打印机中绝大部分采用 ARM 技术。手机中的 32 位 SIM 智能卡也采用了 ARM 技术。

除此以外,ARM 微处理器及技术还应用到许多不同的领域,并会在将来取得更加广泛的应用。

## 2. ARM 微处理器的特点及各种架构版本

采用 RISC 架构的 ARM 微处理器一般具有如下特点。

- (1) 体积小、低功耗、低成本、高性能。
- (2) 支持 Thumb(16 位)/ARM(32 位)双指令集,能很好地兼容 8 位/16 位器件。
- (3) 大量使用寄存器,指令执行速度更快。
- (4) 大多数数据操作都在寄存器中完成。
- (5) 寻址方式灵活简单,执行效率高。
- (6) 指令长度固定。

ARM 十几年如一日地开发新的处理器内核和系统功能块,包括流行的 ARM7TDMI 处理器,还有更新的高档产品 ARM1176TZ(F)-S、Cortex A8 处理器,后者能用于高档手机中。功能的不断进化,处理水平的持续提高,年深日久造就了一系列的 ARM 架构。要说明的是,架构版本号和名字中的数字的含义并不相同。比如,ARM7TDMI 是基于 ARMv4T 架构的(T 表示支持“Thumb 指令”);ARMv5TE 架构则是伴随着 ARM9E 处理器家族亮相的。ARM9E 家族成员包括 ARM926E-S 和 ARM946E-S。ARMv5TE 架构添加了“服务于多媒体应用增强的 DSP 指令”。后来又出了 ARM11,ARM11 是基于 ARMv6 架构建成的。基于 ARMv6 架构的处理器包括 ARM1136J(F)-S,ARM1156T2(F)-S,以及 ARM1176JZ(F)-S。ARMv6 是 ARM 进化史上的一个重要里程碑:从那时候起,许多突破性的新技术被引进,存储器系统加入了很多的崭新特性,单指令流多数据流(SIMD)指令也是从 v6 开始首次引入的。而最前卫的新技术,就是经过优化的 Thumb-2 指令集,它专为低成本的单片机及汽车组件市场设计。

ARMv6 的设计中还有另一个重大的决定:虽然这个架构要能上能下,从最低端的 MCU 到最高端的“应用处理器”都适用,但仍须定位准确,使处理器的架构能胜任每个应用领域。结果就是,要使 ARMv6 能够灵活地配置和剪裁。对于成本敏感市场,要设计一个低门数的架构,让它有极强的确定性;另一方面,在高端市场上,既要有丰富功能,还要具有高性能。

最近几年,基于从 ARMv6 开始的新设计理念,ARM 进一步扩展了它的 CPU 设计,成果就是 ARMv7 架构的闪亮登场。在这个版本中,内核架构首次从单一款式变成三种款式。

- (1) 款式 A: 设计用于高性能的“开放应用平台”。
- (2) 款式 R: 用于高端的嵌入式系统,尤其是那些带有实时要求的。
- (3) 款式 M: 用于深度嵌入的,单片机风格的系统中。

下面详细介绍这三种款式。

(1) 款式 A(ARMv7-A): 需要运行复杂应用程序的“应用处理器”。支持大型嵌入式操作系统(不一定实时),比如 Symbian(诺基亚智能手机用)、Linux、微软的 Windows CE、智能手机操作系统 Windows Mobile、流行的 Android 系统。这些应用需要强大的处理性能,并且需要硬件 MMU 实现的完整而强大的虚拟内存机制,还基本上会配有 Java 支持,有时还要求一个安全程序执行环境(用于电子商务)。典型的产品包括高端手机和手持仪器、电子钱包以及金融事务处理机。

(2) 款式 R(ARMv7-R): 硬实时且高性能的处理器,目标是高端实时市场。例如,高档轿车的组件、大型发电机控制器、机器人手臂控制器等,它们使用的处理器不但要很好很强大,还要极其可靠,对事件的反应也要极其敏捷。

(3) 款式 M(ARMv7-M): 根据旧世代单片机的应用而量身定制。在这些应用中,尤其是对于实时控制系统,低成本、低功耗、极速中断反应以及高处理效率,都是至关重要的。Cortex 系列是 v7 架构的第一次亮相,其中 Cortex-M3 就是按款式 M 设计的。

本书主要介绍 Cortex-M3。到目前为止,Cortex-M3 也是款式 M 中唯一成功的一例。其他 Cortex 家族的处理器包括款式 A 的 Cortex-A8(应用处理器)、款式 R 的 Cortex-R4(实时处理器)。如图 1-1 所示为一些 ARM 处理器架构进化史。

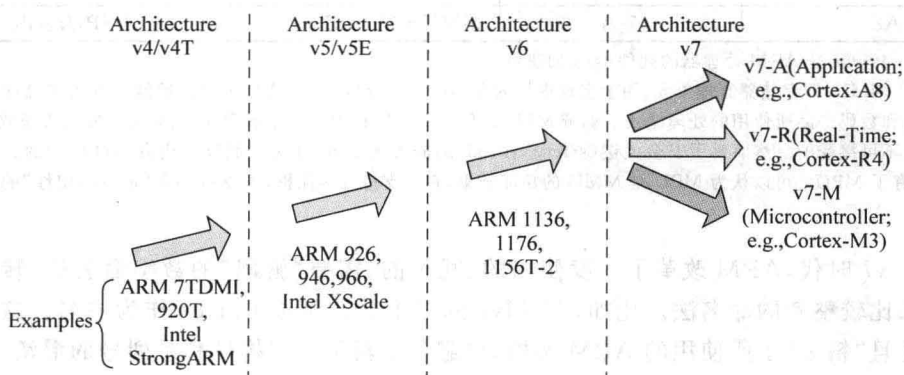


图 1-1 ARM 处理器架构进化史

### 3. 处理器命名法

以前,ARM 使用一种基于数字的命名法。在早期(20 世纪 90 年代),还在数字后面添加字母后缀,用来进一步明细该处理器支持的特性。以 ARM7TDMI 为例,T 代表 Thumb 指令集,D 表示支持 JTAG 调试(Debugging),M 指快速乘法器,I 则对应一个嵌入式 ICE 模块。后来,这 4 项基本功能成了任何新产品的标配,于是就不再使用这 4 个后缀了。但是新的后缀不断加入,包括定义存储器接口的,定义高速缓存的,以及定义“紧耦合存储器(TCM)”的,于是形成了新一套命名法,这套命名法也是一直在使用的。如表 1-1 所示为一些 ARM 处理器名字。



表 1-1 ARM 处理器名字

处理器名字	架构版本号	存储器管理特性	其他特征
ARM7TDMI	v4T		
ARM7TDMI-S	v4T		
ARM7EJ-S	v5E		DSP, Jazelle <sup>[1]</sup>
ARM920T	v4T	MMU	
ARM922T	v4T	MMU	
ARM926EJ-S	v5E	MMU	DSP, Jazelle
ARM946E-S	v5E	MMU	DSP
ARM966E-S	v5E		DSP
ARM968E-S	v5E		DMA, DSP
ARM966HS	v5E	MPU(可选)	DSP
ARM1020E	v5E	MMU	DSP
ARM1022E	v5E	MMU	DSP
ARM1026EJ-S	v5E	MMU 或 MPU <sup>[2]</sup>	DSP, Jazelle
ARM1136J(F)-S	v6	MMU	DSP, Jazelle
ARM1176JZ(F)-S	v6	MMU+TrustZone	DSP, Jazelle
ARM11MPCore	v6	MMU+多处理器缓存支持	DSP, Jazelle
ARM1156T2(F)-S	v6	MPU	DSP
Cortex-M3	v7-M	MPU(可选)	NVIC
Cortex-R4	v7-R	MPU	DSP
Cortex-R4F	v7-R	MPU	DSP+浮点运算
Cortex-A8	v7-A	MMU+TrustZone	DSP, Jazelle

[注 1]: Jazelle 是 ARM 处理器的硬件 Java 加速器。

[注 2]: MMU, 即存储器管理单元, 用于实现虚拟内存和内存的分区保护, 这是应用处理器与嵌入式处理器的分水岭。计算机和数码产品所使用的处理器几乎都带 MMU。但是 MMU 也引入了不确定性, 这有时是嵌入式领域——尤其是实时系统不可接受的。然而对于安全关键(safety critical)的嵌入式系统, 还是不能没有内存分区保护的。为解决矛盾, 于是就有了 MPU。可以认为 MPU 是 MMU 的功能子集, 它只支持分区保护, 不支持具有“定位决定性”的虚拟内存机制。

到了 v7 时代, ARM 改革了一度使用的、冗长的、需要“解码”的数字命名法, 转到另一种看起来比较整齐的命名法。比如, ARMv7 的三个款式都以 Cortex 作为主名。这不仅更加澄清并且“精装”了所使用的 ARM 架构, 也避免了新手对架构号和系列号的混淆。例如, ARM7TDMI 并不是一款 ARMv7 的产品, 而是辉煌起点——v4T 架构的产品。下面重点介绍 Cortex-M3。

Cortex-M3 处理器的舞台: 高性能+高代码密度+小硅片面积, 使得 CM3 大面积地成为理想的处理平台, 主要应用在以下领域: 低成本单片机、汽车电子、数据通信、工业控制、消费类电子产品等。

### 1.3 Cortex-M3 结构处理器概览

#### 1. 简介

如图 1-2 所示为基于 Cortex-M3 内核处理器的结构。