

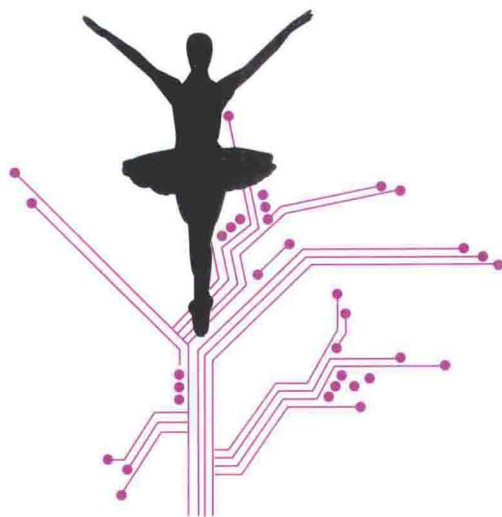
ARM公司微控制器系统级设计专家Joseph Yiu享誉全球的代表作品!

ARM公司微控制器工具总监Reinhard Keil作序!

全新改版,系统论述ARM Cortex-M3与Cortex-M4的内核、体系结构、指令集、编译器、程序设计及软件移植的经典著作!

清华

开发者书库



The Definitive Guide to ARM Cortex-M3 and Cortex-M4 Processors
(Third Edition)

ARM Cortex-M3 与 Cortex-M4

权威指南

(第3版)

Joseph Yiu◎著
吴常玉 曹孟娟 王丽红◎译

清华大学出版社

清华开发者书库

**The Definitive Guide to
ARM Cortex-M3 and Cortex-M4
Processors (Third Edition)**
**ARM Cortex-M3 与
Cortex-M4 权威指南 (第 3 版)**

Joseph Yiu 著

吴常玉 曹孟娟 王丽红 译



清华大学出版社
北京

The Definitive Guide to ARM Cortex-M3 and Cortex-M4 Processors, 3rd Edition

Joseph Yiu

ISBN: 9780124080829

Copyright © 2014 by Elsevier. All rights reserved.

Authorized Simplified Chinese translation edition published by the Proprietor.

Copyright © 2014 by Elsevier (Singapore) Pte Ltd. and Tsinghua University Press. All rights reserved.

Published in China by Tsinghua University Press under special arrangement with Elsevier (Singapore) Pte Ltd. This edition is authorized for sale in China only, excluding Hong Kong SAR and Taiwan. Unauthorized export of this edition is a violation of the Copyright Act. Violation of this Law is subject to Civil and Criminal Penalties.

本书简体中文版由 Elsevier (Singapore) Pte Ltd. 授予清华大学出版社在中国大陆地区(不包括香港、澳门特别行政区以及台湾地区)出版与发行。未经许可之出口,视为违反著作权法,将受法律之制裁。

北京市版权局著作权合同登记号 图字:01-2014-3940

本书封底贴有 Elsevier 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

ARM Cortex-M3 与 Cortex-M4 权威指南:第3版/(英)姚文祥著;吴常玉,曹孟娟,王丽红译.—北京:清华大学出版社,2015

(清华开发者书库)

书名原文: The Definitive Guide to ARM Cortex-M3 and Cortex-M4 Processors,3rd Edition

ISBN 978-7-302-40292-3

I. ①A… II. ①姚… ②吴… ③曹… ④王… III. ①微处理器—指南 IV. ①TP332-62

中国版本图书馆 CIP 数据核字(2015)第 106068 号

责任编辑:盛东亮

封面设计:李召霞

责任校对:梁毅

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

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>,010-62795954

印装者:清华大学印刷厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:35.75

字 数:890千字

版 次:2015年11月第1版

印 次:2015年11月第1次印刷

印 数:1~2500

定 价:89.00元

产品编号:054434-01

译者序

FOREWORD

从 2008 年开始,基于 Cortex-M3 的单片机以其高性能、低成本及易于使用等诸多优势,已经取代 ARM7,成长为 32 位微控制器的主流。Cortex-M4 在 Cortex-M3 的基础上增加了浮点单元及一些 DSP 指令,可以极大地提高数学运算的效率。由于其诸多特性,之前 ARM9+OS(如 Linux 等)的多种方案,目前可以由 Cortex-M3 或 Cortex-M4+嵌入式 OS 的方式取代。

支持 Cortex-M3/M4 的芯片厂家也在日益增多,包括 ST、TI、Atmel 等在内的芯片巨头都有多款基于 Cortex-M3/M4 的微控制器产品,而且具有多种 Flash 及内存大小、外设以及运行频率等,这也使得我们的选择更加广泛。利用一定的程序架构,可以开发出基于多种硬件平台的程序,为产品提供了更多的保障。

在平常工作中,接触最多的就是 Cortex-M3/M4 了,从最开始的 STM32,到后来 TI、Atmel 及 Fujitsu 等多家的 Cortex-M3/M4 单片机,使用过程中体会最深的就是芯片的高性能和易用性。

在之前编写 8 位机的代码时,为了保证任务的正确执行,可能还要考虑代码的执行速度能否满足任务的需求,而对于 Cortex-M3/M4,由于芯片本身的性能及编译器的效率,一般情况下,无须为执行时间而优化代码,编写单片机的代码就如同计算机程序一样方便。

另外,尽管这些单片机都具有诸多外设以及众多控制和状态寄存器,要了解它们,需要花费一定的时间,但由于厂家一般都提供了丰富的底层驱动库,结合开发工具提供的工程示例,无须在底层代码上花费太多时间,从而可以专注于应用功能的实现。由于这些单片机都具有相同的内核架构,因此熟悉了其中一个,其他产品也能很快上手。

在所有介绍 Cortex-M3/M4 的书籍中,本书无疑是最经典的一本,一方面,作者本身就是 ARM 公司的专家,了解 Cortex-M3/M4 架构的设计;另一方面,作者选取的角度非常合适,既有架构设计的细节,也有程序代码实现示例,而且对容易出现问题的地方进行了说明。

本书虽然名为《ARM Cortex-M3 与 Cortex-M4 权威指南》,实际上也是《ARM Cortex-M3 权威指南》的第 3 版,其中不仅增加了 Cortex-M4 相关的内容,而且由于 Cortex-M3 处理器版本的更新,本书也对上一版的内容进行了修改和补充。

由于本书中的丰富内容,相信无论你是新手还是熟练的开发人员,都可以从中找到有用的信息。限于译者水平,疏漏之处敬请批评指正,最后希望这本书能给读者带来帮助。

译者

2015 年 9 月

推荐序

FOREWORD

嵌入式市场发生了革命性的变化：当前多数的微控制器都基于 ARM 架构，特别是常见的 Cortex-M3 和 Cortex-M4 处理器。近期还出现了一些新的处理器，在低端应用领域，许多之前 8 位和 16 位微控制器占主导的应用也可以使用 Cortex-M0+ 处理器。新的 64 位 Cortex-A50 则面向服务器等高端应用。除了标准化的系统和注重能耗效率的计算性能外，物联网(IoT)也推动了本次变革。分析人员预计，到 2020 年，连接到 IoT 的设备将会达到 500 亿，ARM 处理器的应用领域将会覆盖从传感器到服务器的所有领域。许多设备都会基于 Cortex-M3 和 Cortex-M4 微控制器，而且可能只使用一个小的电池甚至通过能量采集供电。

当前，由于多种开发工具、调试工具和丰富的工程实例的存在，使用基于 ARM Cortex-M3 和 Cortex-M4 处理器的设备是非常简单的。不过，要想提高应用代码的效率，则需要深入了解硬件架构和软件模型。本书提供的信息对系统架构师和软件工程师都非常重要：既有常见的开发工具，也提供了基于 Cortex 微控制器软件接口标准(CMSIS)的多个编程实例。另外，本书还涵盖了 Cortex-M4 处理器的数字信号处理(DSP)特性以及面向模拟应用的 CMSIS-DSP 库。随着许多嵌入式应用变得越来越复杂以及微控制器能力的提高，实时操作系统的使用也就更加普遍了。对于这些内容，本书都提供了易于理解的应用实例。

本书适合所有类型的用户：从开始学习小的 Cortex-M 微控制器项目的学生到需要深入了解处理器特性的系统专家。

Reinhard Keil
ARM 公司 MCU 工具总监

前言

PREFACE

前几年,我们见证了 ARM Cortex-M3 处理器不断扩大自己的应用领域,而且 Cortex-M4 也获得了迅速发展。同时,围绕着 Cortex-M 处理器的软件开发工具和多种技术也在不断进步。例如,目前基本上所有的 Cortex-M 设备驱动库都用上了 CMSIS-Core,而且 CMSIS 项目也扩展为 DSP 库软件等多个方面。

在这一版中,我将书的内容进行了一定的调整,以便初学者可以快速理解 M3&M4 处理器架构,并提高它们在软件应用中的开发效率。应许多用户的要求,还会介绍几个前面的版本未涉及的高级话题,而且它们在其他书或者 ARM 的文档中也没有出现过。在这一版中,还加入了 Cortex-M4 处理器的许多新的信息,比如浮点单元和 DSP 指令的应用细节,并对一些内容进行了更加深入的介绍。例如,与上一版相比,本书介绍的微控制器软件开发组件更多,其中包括基于 CMSIS-RTOS API 的实时操作系统的一章内容以及多个高级话题的其他信息。

本版还增加了 DSP Concepts 的 CEO Paul Beckmann 写的两章内容,DSP Concepts 是为 ARM 开发 CMSIS-DSP 库的公司。我非常高兴能够得到他的帮助,因为他对 DSP 应用及 CMSIS-DSP 库的深入理解,使得本书对于任何 ARM 嵌入式软件开发人员都极具价值。

本书既面向嵌入式硬件系统设计人员,也面向软件工程师。由于书中的内容涵盖了从入门知识到许多详细的高级信息,它也适合多种读者使用,其中包括程序员、嵌入式产品设计人员、电子爱好者、研究人员及片上系统(SoC)工程师。若用户想从包括经典的 ARM 处理器 ARM7TDMI 在内的其他架构移植到 Cortex-M 微控制器,则可以参考介绍软件移植的一章。

真心希望读者能从本书中找到有用的东西。

我想感谢下面的这些人,他们对本书的第 3 版提出了建议和反馈:

首先,非常感谢 Paul Beckmann 博士,他提供了 DSP 方面的两章内容。DSP 运算能力是 Cortex-M4 处理器的一个重要特性,而 DSP 库则可为开发 DSP 应用的用户提供非常大的帮助。有了这两章,本书才称得上完整。

其次,我要感谢 ARM 公司的同事提供的支持,Joey Ye、Stephen Theobald、Graham Cunningham、Edmund Player、Drew Barbier、Chris Shore、Simon Craske 和 Robert Boys 反馈了很多有用的信息。还非常感激 ARM 嵌入式市场团队的支持,他们是 Richard York、Andrew Frame、Neil Werdmuller 和 Ian Johnson。

我要感谢 Keil 公司为我解答了许多 CMSIS 方面问题的 Reinhard Keil、Robert Rostohar 和 Martin Günther,检查 EWARM 相关内容的 IAR Systems 的 Anders Lundgren,以及检查了 Atollic TrueStudio 相关内容的 Magnus Unemyr。

我还要感谢下面的这些人,他们在我写本书第 1 版和第 2 版时提供了帮助,他们是: Dominic Pajak、Alan Tringham、Nick Sampays、Dan Brook、David Brash、Haydn Povey、Gary

Campbell、Kevin McDermott、Richard Earnshaw、Shyam Sadasivan、Simon Axford、Takashi Ugajin、Wayne Lyons、Samin Ishtiaq、Dev Banerjee、Simon Smith、Ian Bell、Jamie Brettle、Carlos O'Donell、Brian Barrera 和 Daniel Jacobowitz。

当然,还得感谢我之前写的书的读者,他们给我提供了很多有用的反馈信息。

另外,感谢 Elsevier 的各位同人,有了他们专业的工作,本书才得以出版。

最后,特别感谢所有的朋友在我写这本书时给予的支持和理解。

Joseph Yiu

关于本书

本书是《ARM Cortex-M3 权威指南》的第 3 版,为了体现出增加的 ARM Cortex-M4 处理器方面的信息,将书名也进行了修改。第 3 版做了全面的修订和更新,包含了 ARM Cortex-M4 处理器的更多信息,且对 Cortex-M3 和 Cortex-M4 处理器增加的新内容做了详细的介绍,这些信息对于各种处理器架构到强大的 Cortex-M3 和 M4 的移植非常有用。

本书介绍了指令集和中断处理等 ARM 架构的基本情况,以及浮点单元等高级特性的编程和使用方法。

在初学者开始编写程序代码时,可以参考 Keil MDK-ARM 入门、IAR EWARM、gcc 和 CooCox CoIDE 工具等章节。本书还会涉及软件开发的一些重要内容,如输入/输出信息、使用嵌入式 OS(CMSIS-RTOS)及汇编和 C 的混合语言工程。

DSP 特性和 CMSIS-DSP 库这两章内容由 DSP Concepts 的创始人 Paul Beckmann 博士提供,DSP Concepts 是为 ARM 开发 CMSIS-DSP 库的公司。这两章介绍了 DSP 的基本内容以及如何编写 Cortex-M4 处理器用的 DSP 软件,其中包括 CMSIS-DSP 库示例以及 Cortex-M4 处理器 DSP 功能的有用信息。

本书中的许多章节都涉及多种调试技术,以及从其他架构进行软件移植方面的问题。其对 ARM Cortex-M3 和 Cortex-M4 处理器进行了详细介绍,由参与开发内核的一名 ARM 工程师编写。书中还包括许多易于理解的例子、图表以及指令集和 CMSIS-Core API 等快速参考附录。

ARM、CORTEX、CORESIGHT、CORELINK、THUMB、AMBA、AHB、APB、Keil、ARM7TDMI、ARM7、ARM9、ARM1156T2(F)-S、Mali、DS-5、Embedded Trace Macrocell 和 PrimeCell 是 ARM 在欧盟和/或其他地方注册的商标,且保留所有权利。其他名字则可能是它们各自拥有者的商标。

注:本书附录及书中示例的源代码可以从 Elsevier 网站下载:<http://booksite.elsevier.com/9780124080829>。

术语和缩写

缩写	含义
ADK	AMBA 设计套件
AHB	高级高性能总线
AHB-AP	AHB 访问端口
AMBA	高级微控制器总线架构
APB	高级外设总线
API	应用编程接口
ARM ARM	ARM 架构参考手册
ASIC	专用集成电路
ATB	高级跟踪总线
BE8	字节不变大端模式
CMSIS	Cortex 微控制器软件接口标准
CPI	周期指令比
CPU	中央处理单元
DAP	调试访问端口
DSP	数字信号处理器/数字信号处理
DWT	数据监视点和跟踪单元
EABI/ABI	嵌入式应用程序二进制接口
ETM	嵌入式跟踪宏单元
FPB	Flash 补丁和断点单元
FPGA	现场可编程门阵列
FPU	浮点单元
FSR	错误状态寄存器
ICE	在线仿真器
IDE	集成开发环境
IRQ	中断请求(一般指外部中断)
ISA	指令集架构
ISR	中断服务程序
ITM	指令跟踪宏单元
JTAG	联合测试行动小组(一种测试/调试接口标准)
JTAG-DP	JTAG 调试端口
LR	链接寄存器
LSB	最低位
LSU	加载/存储单元
MAC	乘累加

MCU	微控制器单元
MMU	存储器管理单元
MPU	存储器保护单元
MSB	最高位
MSP	主栈指针
NaN	非数字(浮点数表示方法)
NMI	不可屏蔽中断
NVIC	嵌套向量中断控制器
OS	操作系统
PC	程序计数器
PMU	电源管理单元
PSP	进程栈指针
PPB	私有外设总线
PSR	程序状态寄存器
RTOS	实时操作系统
SCB	系统控制块
SCS	系统控制空间
SIMD	单指令多数据
SP、MSP、PSP	栈指针、主栈指针、进程栈指针
SoC	片上系统
SRPG	状态保持功率门
SW	串行线
SW-DP	串行线调试端口
SWJ-DP	串行线 JTAG 调试端口
SWV	串行线查看(TPIU 的一种操作模式)
TCM	紧密耦合存储器(Cortex-M1 特性)
TPA	跟踪端口分析仪
TPIU	跟踪端口接口单元
TRM	技术参考手册
UAL	统一汇编语言
WIC	唤醒中断控制器

本书约定

本书在印刷时遵循如下的诸多约定。

(1) 普通汇编程序代码

`MOV R0, R1`; 将寄存器 R1 中的数据送到 R0 中

(2) 汇编代码语法中, <> 中的内容要用实际的寄存器名代替:

`MRS <reg>, <special_reg>`

(3) C 程序代码:

```
for (i = 0; i < 3; i++) { func1(); }
```

(4) 数据:

① `4'hC` 和 `0x123` 都是十六进制数值。

② `#3` 表示 3 号项目(如 `IRQ#3` 表示编号为 3 的 `IRQ`)。

③ `#immed_12` 表示 12 位立即数。

(5) 寄存器位:

一般表示基于位所在位置的部分数据值, `bit[15:12]` 表示 15 位~12 位。

(6) 寄存器访问类型如下:

① R 为只读。

② W 为只写。

③ R/W 为可读可写。

④ R/Wc 为可读,且可被写访问清除。

目录

CONTENTS

译者序	1
推荐序	3
前言	5
关于本书	7
术语和缩写	9
本书约定	11
第 1 章 ARM Cortex-M 处理器简介	1
1.1 什么是 ARM Cortex-M 处理器	1
1.1.1 Cortex-M3 和 Cortex-M4 处理器	1
1.1.2 Cortex-M 处理器家族	2
1.1.3 处理器和微控制器的区别	3
1.1.4 ARM 和微控制器供应商	4
1.1.5 选择 Cortex-M3 和 Cortex-M4 微控制器	4
1.2 Cortex-M 处理器的优势	6
1.2.1 低功耗	6
1.2.2 性能	6
1.2.3 能耗效率	6
1.2.4 代码密度	6
1.2.5 中断	6
1.2.6 易于使用	7
1.2.7 可扩展性	7
1.2.8 调试特性	7
1.2.9 OS 支持	7
1.2.10 多种系统特性	7
1.2.11 软件可移植性和可重用性	7
1.2.12 选择(设备、工具和 OS 等)	7
1.3 ARM Cortex-M 处理器应用	8
1.4 ARM 处理器和 ARM 微控制器的资源	8
1.4.1 ARM 网站上有什么	8

1.4.2	微控制器供应商提供的文档	10
1.4.3	工具供应商提供的文档	10
1.4.4	其他资源	10
1.5	背景和历史	11
1.5.1	ARM 简史	11
1.5.2	ARM 处理器的发展	11
1.5.3	Thumb ISA 的架构版本	13
1.5.4	处理器命名	15
1.5.5	关于 ARM 生态系统	16
第 2 章	嵌入式软件开发简介	18
2.1	ARM 微控制器是怎样构成的	18
2.2	开始时需要准备什么	19
2.2.1	开发组件	19
2.2.2	开发板	19
2.2.3	调试适配器	20
2.2.4	软件设备驱动	21
2.2.5	例子	21
2.2.6	文档和其他资源	21
2.2.7	其他设备	21
2.3	软件开发流程	22
2.4	编译应用程序	23
2.5	软件流程	24
2.5.1	轮询	24
2.5.2	中断驱动	25
2.5.3	多任务系统	27
2.6	C 程序中的数据类型	27
2.7	输入、输出和外设访问	28
2.8	微控制器接口	31
2.9	Cortex 微控制器软件接口标准 (CMSIS)	32
2.9.1	CMSIS 简介	32
2.9.2	CMSIS-Core 所做的标准化	34
2.9.3	CMSIS-Core 的组织结构	34
2.9.4	如何使用 CMSIS-Core	35
2.9.5	CMSIS 的优势	36
2.9.6	CMSIS 的多个版本	38

第 3 章 技术综述	40
3.1 Cortex-M3 和 Cortex-M4 处理器的一般信息	40
3.1.1 处理器类型	40
3.1.2 处理器架构	40
3.1.3 指令集	41
3.1.4 模块框图	43
3.1.5 存储器系统	44
3.1.6 中断和异常支持	45
3.2 Cortex-M3 和 Cortex-M4 处理器的特性	45
3.2.1 性能	45
3.2.2 代码密度	46
3.2.3 低功耗	46
3.2.4 存储器系统	47
3.2.5 存储器保护单元	47
3.2.6 中断处理	47
3.2.7 OS 支持和系统级特性	48
3.2.8 Cortex-M4 的特殊特性	48
3.2.9 易于使用	49
3.2.10 调试支持	49
3.2.11 可扩展性	50
3.2.12 兼容性	51
第 4 章 架构	52
4.1 架构简介	52
4.2 编程模型	52
4.2.1 操作模式和状态	52
4.2.2 寄存器	54
4.2.3 特殊寄存器	56
4.2.4 浮点寄存器	62
4.3 应用程序状态寄存器	65
4.3.1 整数状态标志	65
4.3.2 Q 状态标志	66
4.3.3 GE 位	66
4.4 存储器系统	67
4.4.1 存储器系统特性	67
4.4.2 存储器映射	68

4.4.3	栈存储	69
4.4.4	存储器保护单元(MPU)	72
4.5	异常和中断	73
4.5.1	什么是异常	73
4.5.2	嵌套向量中断控制器(NVIC)	74
4.5.3	向量表	75
4.5.4	错误处理	76
4.6	系统控制块(SCB)	76
4.7	调试	76
4.8	复位和复位流程	78
第5章	指令集	80
5.1	ARM Cortex-M 处理器指令集的背景简介	80
5.2	ARM Cortex-M 处理器间的指令集比较	82
5.3	理解汇编语言语法	83
5.4	指令后缀的使用	87
5.5	统一汇编语言(UAL)	88
5.6	指令集	89
5.6.1	处理器内传送数据	90
5.6.2	存储器访问指令	92
5.6.3	算术运算	100
5.6.4	逻辑运算	101
5.6.5	移位和循环移位指令	102
5.6.6	数据转换运算(展开和反序)	103
5.6.7	位域处理指令	105
5.6.8	比较和测试	106
5.6.9	程序流控制	106
5.6.10	饱和运算	113
5.6.11	异常相关指令	115
5.6.12	休眠模式相关指令	116
5.6.13	存储器屏障指令	117
5.6.14	其他指令	118
5.6.15	不支持的指令	119
5.7	Cortex-M4 特有的指令	120
5.7.1	Cortex-M4 的增强 DSP 扩展简介	120
5.7.2	SIMD 和饱和指令	121
5.7.3	乘法和 MAC 指令	124

5.7.4	打包和解包	127
5.7.5	浮点指令	128
5.8	桶形移位器	130
5.9	在编程中访问特殊寄存器和特殊指令	131
5.9.1	简介	131
5.9.2	内在函数	132
5.9.3	内联汇编和嵌入汇编	132
5.9.4	使用其他的编译器相关的特性	132
5.9.5	访问特殊寄存器	132
第 6 章	存储器系统	134
6.1	存储器系统特性简介	134
6.2	存储器映射	134
6.3	连接处理器到存储器和外设	137
6.4	存储器需求	140
6.5	存储器的端	140
6.6	数据对齐和非对齐数据访问支持	142
6.7	位段操作	143
6.7.1	简介	143
6.7.2	位段操作的优势	146
6.7.3	不同数据大小的位段操作	148
6.7.4	C 程序实现的位段操作	148
6.8	默认的存储器访问权限	149
6.9	存储器访问属性	150
6.10	排他访问	152
6.11	存储器屏障	154
6.12	微控制器中的存储器系统	154
第 7 章	异常和中断	157
7.1	异常和中断简介	157
7.2	异常类型	158
7.3	中断管理简介	159
7.4	优先级定义	160
7.5	向量表和向量表重定位	165
7.6	中断输入和挂起行为	168
7.7	异常流程简介	171
7.7.1	接受异常请求	171

7.7.2	异常进入流程	171
7.7.3	执行异常处理	171
7.7.4	异常返回	172
7.8	中断控制用的 NVIC 寄存器细节	172
7.8.1	简介	172
7.8.2	中断使能寄存器	173
7.8.3	设置中断挂起和清除中断挂起	174
7.8.4	活跃状态	175
7.8.5	优先级	176
7.8.6	软件触发中断寄存器	176
7.8.7	中断控制器类型寄存器	177
7.9	用于异常和中断控制的 SCB 寄存器细节	177
7.9.1	SCB 寄存器简介	177
7.9.2	中断控制和状态寄存器(ICSR)	178
7.9.3	向量表偏移寄存器(VTOR)	179
7.9.4	应用中断和复位控制寄存器(AIRCR)	179
7.9.5	系统处理优先级寄存器(SCB->SHP[0~11])	180
7.9.6	系统处理控制和状态寄存器(SCB->SHCSR)	181
7.10	用于异常或中断屏蔽的特殊寄存器细节	182
7.10.1	PRIMASK	182
7.10.2	FAULTMASK	182
7.10.3	BASEPRI	183
7.11	设置中断的步骤示例	184
7.11.1	简单情况	184
7.11.2	向量表重定位时的情况	185
7.12	软件中断	185
7.13	要点和提示	186
第 8 章	深入了解异常处理	188
8.1	简介	188
8.1.1	关于本章	188
8.1.2	C 实现的异常处理	188
8.1.3	栈帧	189
8.1.4	EXC_RETURN	191
8.2	异常流程	192
8.2.1	异常进入和压栈	192
8.2.2	异常返回和出栈	193