

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ARM 嵌入式方案技术总监Wayne Lyons、ARM产品市场总监Richard York联袂作序!

全新改版, 系统论述Cortex-M3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译作!



清华

开发者书库



The Definitive Guide to the ARM Cortex-M3  
Second Edition

# ARM Cortex-M3

## 权威指南

(第2版)

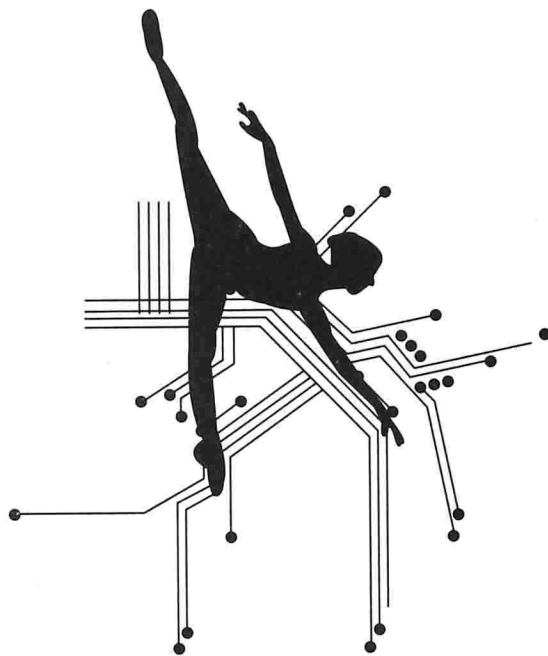
Joseph Yiu◎著  
吴常玉 程凯◎译

清华大学出版社



清华

开发者书库



The Definitive Guide to the ARM Cortex-M3  
Second Edition

# ARM Cortex-M3

## 权威指南

(第2版)

Joseph Yiu◎著

吴常玉 程凯◎译

清华大学出版社

**The Definitive Guide to the ARM Cortex-M3, 2nd Edition**

**Joseph Yiu**

**ISBN: 9781856179638**

Copyright © 2010 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: 01-2013-6748

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 权威指南(第2版)/(英)姚文祥著;吴常玉,程凯译.--北京:清华大学出版社,2014  
(清华开发者书库)

书名原文: The Definitive Guide to the ARM Cortex-M3, 2nd Edition

ISBN 978-7-302-36180-0

I. ①A… II. ①姚… ②吴… ③程… III. ①微处理器—系统设计—指南 IV. ①TP332-62

中国版本图书馆 CIP 数据核字(2014)第 189826 号



责任编辑: 盛东亮

封面设计: 李召霞

责任校对: 白蕾

责任印制: 王静怡

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)

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 刷 者: 北京鑫丰华彩印有限公司

装 订 者: 三河市新茂装订有限公司

经 销: 全国新华书店

开 本: 186mm×240mm 印 张: 26.75 字 数: 590 千字

版 次: 2014 年 8 月第 1 版 印 次: 2014 年 8 月第 1 次印刷

印 数: 1~3000

定 价: 69.00 元

# 译者序

## PREFACE

---

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。而且由于其诸多特性,之前“ARM9+OS”(如“ARM9+Linux”等)的多种方案,目前也可以由“Cortex-M3+嵌入式 OS”的方式取代。

支持 Cortex-M3 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 的微控制器产品,而且具有 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,我们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,我接触最多的就是 Cortex-M3 了,从最开始的 STM32,到后来 TI、Atmel 以及 Fujitsu 等多家的 Cortex-M3 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。

在以往编写 8 位机的代码时,为了保证任务的正确执行,我们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,由于芯片本身的性能及编译器的效率,一般情况下,我们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,要透彻了解它们,需要花费一定的时间。幸运的是,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实例,我们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,因此只要熟悉了其中的一个产品,其他产品也就能很快上手。

在所有介绍 Cortex-M3 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 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,也有程序代码实现的实例,而且对容易出现问题的地方进行了说明。另外,由于 Cortex-M3 处理器版本的更新,本书作为第 2 版,也对上一版的内容进行了大量修改和补充。

由于本书内容丰富,无论你是新手还是熟练开发人员,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。

吴常玉

2014 年 5 月

# 推荐序(一)

PREFACE

---

从本书第 1 版出版开始,ARM 微控制器大家庭的成员不断增加,这已经极大地超出了我们的预计,而且毫不夸张地说,这也是微控制器单元(MCU)领域的革命。成千上万的终端用户都在使用基于 ARM 的 MCU,它也成为市场上增长最快的 MCU 技术。Joseph 编写的《ARM Cortex-M3 权威指南(第 2 版)》非常及时,可以提供 MCU 技术的最新信息。

整体来看,我们在许多方面都取得了巨大的进步,其中包括生产基于 Cortex™-M3 处理器器件的合作伙伴的数量(目前已超过 30 家)。Cortex 微控制器软件接口标准(CMSIS)的发展简化了 Cortex 处理器和芯片供应商之间的产品的代码移植,而且 Cortex-M0 处理器的发布也进一步降低了 ARM MCU 的成本。

由于目前的巨大变化,现在开发基于 Cortex-M3 处理器的嵌入式方案非常令人振奋!

Richard York

(英)ARM 产品市场总监

# 推荐序(二)

PREFACE

---

微控制器编程人员本来就是很灵活的一个群体,从一个确定设计方案出发,他们可以通过某种特定的方式创造出令人惊异的新产品。一般来说,他们需要精确计算,最大限度地节省系统设计。要施展这种魔法,所需要的最主要的元素为工具链环境,也正是因为这个原因,ARM 自己的工具链部门人员才会同 CPU 设计者一道,设计出合理、简化以及优化的 ARM7TDMI 处理器。

作为这种组合的产物,ARM Cortex™-M3 的出现,标志着 ARM 架构取得了令人兴奋的发展。该设备在融合了 32 位 ARM 架构最优的特性以及非常成功的 Thumb-2 指令集设计的同时,还增加了许多新的功能。尽管进行了这些修改,Cortex-M3 还是保留了为人熟知的简单编程模型。

Wayne Lyons

(英)ARM 嵌入式方案技术总监

# 前言

## FOREWORD

---

本书适合对 ARM Cortex™-M3 处理器感兴趣的软件和硬件工程师阅读,《Cortex-M3 技术参考手册》(Cortex-M3 Technical Reference Manual, TRM)和《ARMv7-M 架构应用层参考手册》(ARMv7-M Architecture Application Level Reference Manual)已经提供了本处理器的许多信息,但它们涉及了太多细节,可能不太适合新手。

本书适合编程人员、嵌入式产品设计人员、片上系统(SoC)工程师、电子爱好者、研究人员以及对 Cortex-M3 感兴趣且有微控制器或微处理器使用经验的其他人员。文中系统介绍了架构、指令集描述、指令实例、硬件特性以及处理器高级调试系统,还提供了大量应用程序实例,包括使用 ARM 工具以及 GNU 工具链开发 Cortex-M3 软件的基本步骤。若工程师要将软件从 ARM7TDMI 移植到 Cortex-M3 处理器上,本书也非常适合,因其涉及两个处理器间的差异以及从 ARM7TDMI 到 Cortex-M3 的应用程序软件移植。

Joseph Yiu

# 致谢

## ACKNOWLEDGEMENTS

---

感谢以下朋友,他们给我提供了帮助、指导以及对本书第 2 版如何改进的宝贵建议:

Richard York、Andrew Frame、Reinhard Keil、Nick Sampays、Dev Banerjee、Robert Boys、Dominic Pajak、Alan Tringham、Stephen Theobald、Dan Brook、David Brash、Haydn Povey、Gary Campbell、Kevin McDermott、Richard Earnshaw、Shyam Sadasivan、Simon Craske、Simon Axford、Takashi Ugajin、Wayne Lyons、Samin Ishtiaq 以及 Simon Smith。

感谢 National Instruments(美国国家仪器公司)的 Ian Bell 和 Jamie Brettle 的支持,他们帮我检查了涉及 NI LabVIEW 的材料。还要感谢 CodeSourcery 的 Carlos O'Donell、Brian Barrera 和 Daniel Jacobowitz 的支持,他们帮我检查了涉及 CodeSourcery 工具链软件开发的材料。当然,还要感谢 Elsevier 的工具人员,有了他们专业的工作,本书才得以顺利出版。

最后,感谢 Peter Cole 和 Ivan Yardley 在本书编写过程中给予的建议和支持。

Joseph Yiu



# 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) 伪代码：

```
if (a > b) { ...
```

(5) 数据：

① 4'hC 和 0x123 都是 16 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。

(6) 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 位到 12 位。

(7) 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。

# 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
ARM TRM	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
CS3	CodeSourcery 通用启动代码序列
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
FSR	错误状态寄存器
HTM	CoreSight AHB 跟踪宏单元
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
MCU	微控制器单元
MDK-ARM	Keil ARM 微控制器开发套件
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
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
SCB	系统控制块
SCS	系统控制空间
SIMD	单指令,多数据
SoC	片上系统
SP	栈指针
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
UART	通用异步接收/发送装置
WIC	唤醒中断控制器

# 目录

## CONTENTS

---

译者序 .....	1
推荐序(一) .....	3
推荐序(二) .....	5
前言 .....	7
致谢 .....	9
本书约定 .....	11
术语和缩写 .....	13
<b>第 1 章 介绍</b> .....	<b>1</b>
1.1 ARM Cortex-M3 处理器是什么 .....	1
1.2 ARM 和 ARM 架构的背景 .....	2
1.2.1 历史简介 .....	2
1.2.2 架构版本 .....	3
1.2.3 处理器命名 .....	5
1.3 指令集开发 .....	6
1.4 Thumb-2 技术和指令集架构 .....	7
1.5 Cortex-M3 处理器应用 .....	7
1.6 本书的组织结构 .....	8
1.7 深入阅读 .....	8
<b>第 2 章 Cortex-M3 综述</b> .....	<b>10</b>
2.1 基础 .....	10
2.2 寄存器 .....	11
2.2.1 R0~R12: 通用目的寄存器 .....	11
2.2.2 R13: 栈指针 .....	11
2.2.3 R14: 链接寄存器 .....	12
2.2.4 R15: 程序计数器 .....	12
2.2.5 特殊寄存器 .....	12

2.3	操作模式	13
2.4	内置的嵌套向量中断控制器	13
2.4.1	支持嵌套中断	14
2.4.2	支持向量中断	14
2.4.3	支持动态修改优先级	14
2.4.4	中断等待减小	14
2.4.5	中断屏蔽	14
2.5	存储器映射	14
2.6	总线接口	15
2.7	MPU	16
2.8	指令集	16
2.9	中断和异常	17
2.10	调试支持	19
2.11	特点概括	19
2.11.1	高性能	20
2.11.2	中断处理的高级特性	20
2.11.3	低功耗	21
2.11.4	系统特性	21
2.11.5	调试支持	21
<b>第3章</b>	<b>Cortex-M3 基础</b>	<b>22</b>
3.1	寄存器	22
3.1.1	通用目的寄存器 R0~R7	22
3.1.2	通用目的寄存器 R8~R12	22
3.1.3	栈指针 R13	22
3.1.4	链接寄存器 R14	24
3.1.5	程序计数器 R15	25
3.2	特殊寄存器	25
3.2.1	程序状态寄存器	26
3.2.2	PRIMASK、FAULTMASK 和 BASEPRI 寄存器	27
3.2.3	控制寄存器	28
3.3	操作模式	29
3.4	异常和中断	31
3.5	向量表	32
3.6	栈存储操作	32
3.6.1	栈的基本操作	33

3.6.2	Cortex-M3 栈的应用 .....	34
3.6.3	Cortex-M3 的双栈模型 .....	35
3.7	复位流程 .....	36
<b>第 4 章</b>	<b>指令集 .....</b>	<b>38</b>
4.1	汇编基础 .....	38
4.1.1	汇编语言：基本语法 .....	38
4.1.2	汇编语言：后缀的使用 .....	39
4.1.3	汇编语言：统一汇编语言 .....	40
4.2	指令列表 .....	41
4.3	指令描述 .....	46
4.3.1	汇编语言：传送数据 .....	47
4.3.2	LDR 和 ADR 伪指令 .....	50
4.3.3	汇编语言：处理数据 .....	51
4.3.4	汇编语言：调用和无条件跳转 .....	54
4.3.5	汇编语言：决断和条件跳转 .....	56
4.3.6	汇编语言：组合比较和条件跳转 .....	59
4.3.7	汇编语言：指令屏障和存储器屏障指令 .....	60
4.3.8	汇编语言：饱和运算 .....	61
4.4	Cortex-M3 上一些有用的指令 .....	63
4.4.1	MSR 和 MRS .....	63
4.4.2	进一步了解 IF-THEN 指令块 .....	64
4.4.3	SDIV 和 UDIV .....	66
4.4.4	REV、REVH 和 REVSH .....	66
4.4.5	位反转 .....	66
4.4.6	SXTB、SXTH、UXTB 和 UXTH .....	67
4.4.7	位域清除和位域插入 .....	67
4.4.8	UBFX 和 SBFX .....	68
4.4.9	LDRD 和 STRD .....	68
4.4.10	表格跳转字节和表格跳转半字 .....	69
<b>第 5 章</b>	<b>存储器系统 .....</b>	<b>71</b>
5.1	存储器系统特性概述 .....	71
5.2	存储器映射 .....	71
5.3	存储器访问属性 .....	73
5.4	默认的存储器访问权限 .....	75

5.5	位段操作	75
5.5.1	位段操作的优势	78
5.5.2	不同数据宽度的位段操作	81
5.5.3	C程序实现位段操作	81
5.6	非对齐传输	82
5.7	排他访问	84
5.8	端模式	85
<b>第6章</b>	<b>Cortex-M3 设计综述</b>	<b>88</b>
6.1	流水线	88
6.2	详细框图	89
6.3	Cortex-M3 上的总线接口	92
6.3.1	I-CODE 总线	92
6.3.2	D-CODE 总线	92
6.3.3	系统总线	92
6.3.4	外部 PPB	93
6.3.5	DAP 总线	93
6.4	Cortex-M3 上的其他接口	93
6.5	外部 PPB	94
6.6	典型连接	94
6.7	复位类型和复位信号	95
<b>第7章</b>	<b>异常</b>	<b>97</b>
7.1	异常类型	97
7.2	优先级定义	98
7.3	向量表	103
7.4	中断输入和挂起行为	104
7.5	错误异常	106
7.5.1	总线错误	106
7.5.2	存储器管理错误	107
7.5.3	使用错误	108
7.5.4	硬件错误	109
7.5.5	处理错误	110
7.6	请求管理调用和可挂起的服务调用	111

<b>第 8 章 嵌套向量中断控制器和中断控制</b> .....	114
8.1 嵌套向量中断控制器概述 .....	114
8.2 基本的中断配置 .....	114
8.2.1 中断使能和清除使能 .....	115
8.2.2 中断设置挂起和清除挂起 .....	116
8.2.3 优先级 .....	117
8.2.4 活跃状态 .....	117
8.2.5 PRIMASK 和 FAULTMASK 特殊寄存器 .....	118
8.2.6 BASEPRI 特殊寄存器 .....	118
8.2.7 其他异常的配置寄存器 .....	119
8.3 设置中断的步骤实例 .....	121
8.4 软件中断 .....	123
8.5 SYSTICK 定时器 .....	123
<b>第 9 章 中断行为</b> .....	126
9.1 中断/异常流程 .....	126
9.1.1 压栈 .....	126
9.1.2 取向量 .....	127
9.1.3 寄存器更新 .....	127
9.2 异常退出 .....	128
9.3 嵌套中断 .....	128
9.4 末尾连锁中断 .....	128
9.5 延迟到达 .....	129
9.6 进一步了解异常返回值 .....	130
9.7 中断等待 .....	131
9.8 中断相关的错误 .....	132
9.8.1 压栈 .....	132
9.8.2 出栈 .....	132
9.8.3 取向量 .....	132
9.8.4 非法返回 .....	132
<b>第 10 章 Cortex-M3 编程</b> .....	133
10.1 概述 .....	133
10.2 典型的开发流程 .....	133
10.3 使用 C .....	134



10.3.1	使用 RealView 开发组件的简单 C 程序实例 .....	134
10.3.2	使用 Keil MDK-ARM 编译相同的例程 .....	137
10.3.3	用 C 访问存储器映射的寄存器 .....	138
10.3.4	内在函数 .....	140
10.3.5	嵌入式汇编和内联汇编 .....	141
10.4	CMSIS .....	142
10.4.1	CMSIS 背景 .....	142
10.4.2	标准化的问题 .....	142
10.4.3	CMSIS 的组织结构 .....	143
10.4.4	使用 CMSIS .....	143
10.4.5	CMSIS 的优势 .....	145
10.5	使用汇编 .....	146
10.5.1	汇编和 C 的接口 .....	147
10.5.2	汇编编程的第一步 .....	147
10.5.3	生成输出 .....	148
10.5.4	“Hello World”实例 .....	149
10.5.5	使用数据存储器 .....	153
10.6	在信号量中使用排他访问 .....	153
10.7	在信号量中使用位段 .....	156
10.8	使用位域提取和表格跳转 .....	157
<b>第 11 章</b>	<b>异常编程 .....</b>	<b>159</b>
11.1	使用中断 .....	159
11.1.1	设置栈 .....	159
11.1.2	设置向量表 .....	160
11.1.3	设置中断优先级 .....	160
11.1.4	使能中断 .....	161
11.2	异常/中断处理 .....	163
11.3	软件中断 .....	165
11.4	向量表重定位实例 .....	166
11.5	使用 SVC .....	168
11.6	SVC 实例：在文字消息输出函数中的应用 .....	170
11.7	用 C 实现 SVC .....	172
<b>第 12 章</b>	<b>高级编程特性和系统行为 .....</b>	<b>175</b>
12.1	运行具有两个独立栈的系统 .....	175