

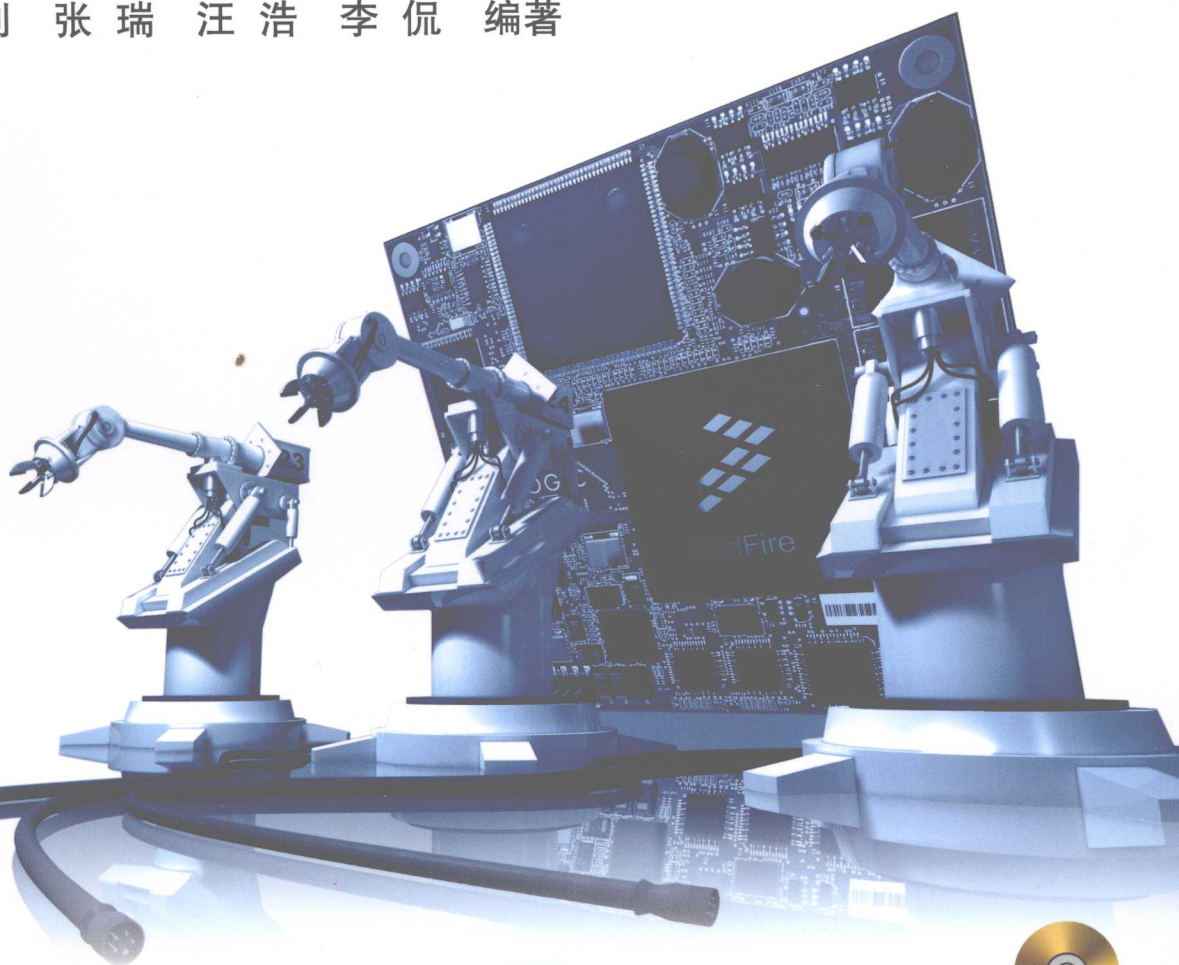


深入浅出

ColdFire系列

32位嵌入式微处理器

湛利 张瑞 汪浩 李侃 编著



北京航空航天大学出版社



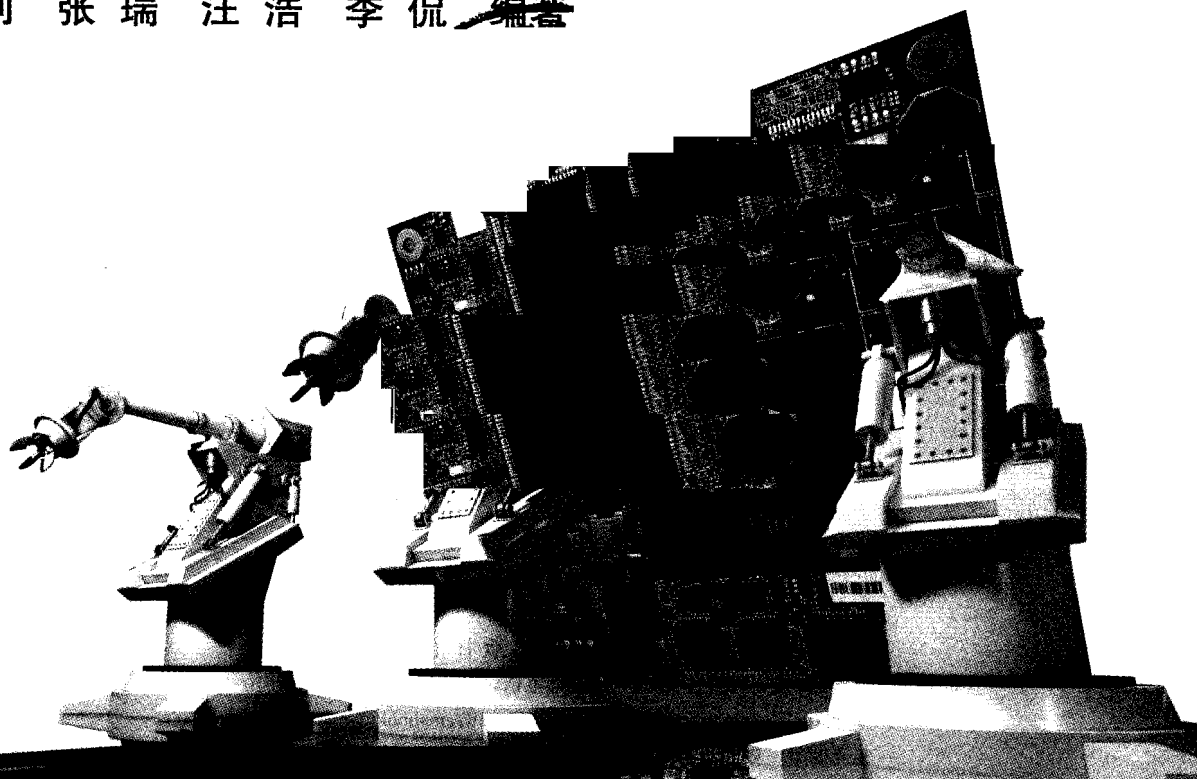


深入浅出

ColdFire系列

32位嵌入式微处理器

湛利 张瑞 汪浩 李侃 编著



内 容 简 介

本书是针对飞思卡尔 32 位 ColdFire 系列嵌入式微处理器的应用和开发,主要是针对该系列 V2 与 V3 内核,即代表该系列大部分的中低端产品的应用。通过对每个模块的介绍和详细的应用实例,使读者更方便更容易地理解各个功能模块的应用,从而达到快速上手的目的。本书基于一个具体的芯片 MCF5225x 进行举例和应用,该芯片为最新的 ColdFire 家族成员,其特点是集成了几乎所有 ColdFire 家族的中低端功能模块,包括使用比较多的以太网,USB OTG,CAN,DMA, FlexBus/MiniBus,SDR/DDR Controller 等模块,并且可以适用于其他各类 ColdFire 系列芯片。本书附光盘 1 张,内含飞思卡尔半导体公司授权的 ColdFire 系列资料和代码例程。

读者对象主要是目前 ColdFire 处理器的使用者和 32 位嵌入式处理器的开发者与爱好者,也可作为大学相关课程的实验教材。

图书在版编目(CIP)数据

深入浅出 ColdFire 系列 32 位嵌入式微处理器/谌利等

编著. — 北京:北京航空航天大学出版社,2009.9

ISBN 978-7-81124-903-3

I. 深… II. 谌… III. 微处理器,ColdFire IV. TP332

中国版本图书馆 CIP 数据核字(2009)第 153208 号

© 2009,北京航空航天大学出版社,版权所有。

未经本书出版者书面许可,任何单位和个人不得以任何形式或手段复制或传播本书及其所附光盘内容。

侵权必究。

深入浅出 ColdFire 系列 32 位嵌入式微处理器

谌 利 张 瑞 汪 浩 李 侃 编著

责任编辑 刘 星

*

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(100191) 发行部电话:010-82317024 传真:010-82328026

<http://www.buaapress.com.cn> E-mail:emsbook@gmail.com

涿州市新华印刷有限公司印装 各地书店经销

*

开本:787×1092 1/16 印张:23.5 字数:602 千字

2009 年 9 月第 1 版 2009 年 9 月第 1 次印刷 印数:5 000 册

ISBN 978-7-81124-903-3 定价:42.00 元(含光盘 1 张)

序

—

这是一本关于飞思卡尔 32 位 ColdFire 处理器开发和应用的书籍。很多人在看这本书前可能会问这样两个问题：为什么要 32 位？为什么是 ColdFire？

这是一个不争的事实：嵌入式系统中的核心部件微控制器(MCU)或微处理器(MPU)正从早些年主流 8 位系统逐渐向全新的 32 位系统快速升级换代。特别是近年来半导体设计和制造技术的突飞猛进，大大降低了芯片的价格，使得 32 位系统在总体成本上已经可以被包括消费类产品在内的众多应用领域所接受；另外由于日趋复杂的功能需求，特别是呈爆炸性扩展的网络通信互联的需要，正促使 32 位系统以异乎寻常的速度进入各类产品和系统的设计领域，32 位的设计方案也被大量摆放在广大电子设计工程师的面前。因此，如何深入理解一款 32 位处理器的性能和资源及其所需的开发环境，以便在较短时间内设计出一款高性价比和高可靠性的产品，是摆在很多硬件和软件设计工程师面前的一个巨大挑战。

和传统的 8 位系统发展道路类似，现在的 32 位系统设计也正处于群雄逐鹿、各显神通的阶段，国内比较常见的 32 位微控制器有 ARM、ColdFire 和 MIPS 等系列。纵观国内嵌入式系统设计领域，目前 ARM 架构无疑处于 32 位系统设计的主导地位，这点和当初的 51 系列在 8 位单片机中的地位是何其相似。各种原因当然有很多，但其中一个非常重要的原因是关于 ARM 架构的设计开发资料和相关书籍比较多，工程师们比较容易从公开的渠道获取一些基本资料，以支撑自己的产品开发。但恰恰也像 8 位单片机的发展历程一样，面对形形色色各类差异化的产品设计，绝非一个架构就可以包揽一切。对于广大设计工程师来说，有机会能了解和掌握不同架构的 32 位处理器系统，对于优化自己的设计方案，扩展设计思路将会提供非常大的帮助。

飞思卡尔的 ColdFire 系列 32 位处理器目前似乎不为广大的国内工程师所熟知。但若提及早些年 Motorola 的 68K 系列，很多有些年纪的工程师应该还记忆犹新。这款早年的 32 位处理器绝对统领了当时的 32 位嵌入式系统设计。国内的许多高端嵌入式设计也采用了 68K 系列，很多产品至今还在批量生产，其可靠性和稳定性被一致公认。ColdFire 系列正是脱胎于 68K，在原 68K 的基础上极大地提高了芯片的运行速度，增强了指令系统和数据运算能力，扩充了大量不同的外围功能模块。针对不同级别应用的实际需求，提供了由低端 V1 到高端 V4 的不同级别的内核。各内核级别之间在体现功能和性能差异化的同时，又保证了很好的兼容性和可移植性，为不同的产品设计提供了丰富灵活的选择。特别要强调的是，飞思卡尔充分考虑到现有 8 位系统直接向 32 位系统迁移的这么一个趋势，专门设计了引脚和外围功能模块完全兼容的 8 位单片机(9S08 系列)和 32 位 ColdFire 单片机(V1 系列)，这使得一个 8 位系统设计几乎不用改动任何硬件和软件，就可以直接升级到 32 位高性能系统成为一种可能。工程师们完全可依据现有的 8 位系统的设计经验，学习和掌握 32 位系统设计的基本要素，在此基础

上进一步提升到更高级别和更复杂的 32 位系统设计。

毋庸置疑,32 位系统的设计复杂程度远非简单的 8 位系统可以比拟。32 位系统往往涉及大量的接口与数据处理、复杂的系统控制过程和高端的人机界面等。这样,一个 32 位系统的设计,一般都牵涉到大量的数据存储管理、高效的嵌入式实时操作系统、众多可选的网络和通信接口、灵活可靠的数据互联技术等不同方面的设计要求。对于一个电子设计工程师,特别是刚接触 32 位系统的工程师而言,要同时掌握如此庞杂繁复的硬件和软件系统,在短时间内绝非易事。本人在日常现场技术支持的过程中,深刻体会到国内广大设计工程师迫切所需的是更多富含实际应用经验和开发心得的资料,以便他们在设计时可以参考和借鉴,减少设计弯路,缩短设计时间。于是,就极力邀请飞思卡尔的几个经验丰富的应用工程师们挤出他们宝贵的业余时间,写一些有关 ColdFire 处理器的应用文章,和广大设计工程师们做些技术交流并分享一些心得体会。很高兴他们能热心投入到这么一件对于他们来讲完全是工作职责范围之外的事情中,并在很短的时间内就成就了这本书。

衷心希望此书能给从事嵌入式系统设计的工程师们带来一些帮助。

张明峰

飞思卡尔半导体公司现场应用工程师经理

2009 年 2 月 22 日于上海

序 二

ColdFire 系列微处理器作为 Motorola M68000 系列的延伸,从诞生至今已有 10 多年了,其间经过了多次的完善和性能的提升,并通过广大的专业工作者在汽车、工业、医疗、消费类等领域中得到广泛的应用。但介绍 ColdFire 微处理器的中文原版书籍,在此之前还没有。针对这种情况,作为飞思卡尔半导体的应用工程师,决定写一本介绍可变长 RISC 架构系统内核和外围模块的书籍,让更多国内读者了解 ColdFire,同时给应用开发人员的实际工作提供帮助。如果能为 ColdFire 系列微处理的普及尽一些绵薄之力,那就更好了。

作为一本面向嵌入式初学者和软硬件工程师的书籍,本书还包含了基于飞思卡尔半导体最新推出的 MCF52259、MCF5445 上的一些程序实例,希望能让读者深入浅出地了解芯片之外,对一些标准的工业接口和规范也能有一定的认识。

飞思卡尔半导体公司为广大的客户提供了基于 ColdFire 内核的大量衍生产品,它们集成了不同大小的存储和不同的外设,相信用户一定可以从中找到适合自己方案的处理器。

郭 雷

飞思卡尔半导体公司系统与应用工程师经理

2009 年 2 月 18 日

前言

ColdFire 系列家族最早起源于摩托罗拉的 68K 系列处理器,至今与 68K 系列处理器有着非常好的继承性和连贯性,其全系列的产品针对嵌入式领域的各方面及高低端的应用非常的完备,并且各外设模块都有很好的一致性,使得研发人员只要学会一种产品,就可以很容易地开发其家族成员的其他产品,这与 ARM 系列产品完全不同。由于 ARM 系列产品各厂商都有,其外设也是不尽相同,所以为跨厂商平台的升级移植工作带来一定难度。68K/ColdFire 产品在欧美地区应用一直非常广泛,而国内公司对 68K/ColdFire 产品并不熟悉,大多是跨国大型企业在国内的分公司才会熟悉。

飞思卡尔半导体的前身是摩托罗拉半导体事业部,是摩托罗拉非常重要的一部分,2004 年作为独立的上市公司从摩托罗拉中独立出来。2006 年飞思卡尔半导体公司为了拓展在中国的业务,在中国成立了针对工业与消费类电子的核心团队——系统与应用工程师团队,主要负责 8/16 位单片机和 32 位 ColdFire 处理器新产品在定义、验证、应用开发等各方面的研发工作以及为亚太区的市场提供技术服务。湛利在 2006 年之前一直在摩托罗拉半导体及后来的飞思卡尔半导体公司负责手机平台的软件开发和市场支持,2006 年底转入系统与应用工程师核心团队成为负责 32 位微处理器 ColdFire 的第一位工程师。后来应用工程师团队逐渐扩大,汪浩、李侃和居颖轶加入进来。通过现场应用工程师团队的努力,ColdFire 在中国的业务也是日益增大。

中国工科工程师普遍来说在英文上处于劣势,一般的阅读没有问题,但是要大量地精通阅读专业英文资料,则可能受到一定限制,会导致研发效率的下降。尤其是 ColdFire 产品中文的资料较少,导致工程师入门门槛比较高,这也使我们日益感觉到有必要为中国的嵌入式系统研发人员更有效地介绍 ColdFire 系列产品。此时恰逢现场应用工程师经理张明峰先生邀请应用工程师与现场应用工程师联手编写一本关于 ColdFire 系列产品实际应用的书,以便中国的工程师可以快速入门,熟练掌握该产品的开发和应用。在本书编写过程中,我们欣喜地看到我们的美国同事 Rudan Bettelheim 编写 *ColdFire Microprocessors and Microcontrollers* 一书的中文译本也出版发行,这本书主要从内部功能来介绍 ColdFire 系列产品,与本书的应用实例形成较好的搭配,读者可以把两本书结合起来阅读。

经过团队不懈的努力,历时半年多,终于完成此书的写作。在本书初稿完成的时候,正值全球经济危机蔓延到包括电子行业的实体经济的严冬,中国电子产业也受到波及。我们希望这本书正如 ColdFire 的名字一样,能够为中国电子产业的严冬带来一缕火焰。

本书的 4 位作者来自飞思卡尔半导体应用工程师和现场工程师团队,其中,湛利、汪浩和李侃均为高级应用工程师,张瑞为现场应用工程师,4 人均均为本书的编写花费了大量的时间和精力。此外,高级应用工程师居颖轶也为本书的编写和校对工作作出了重要贡献。本

书主要基于飞思卡尔半导体公司 ColdFire 系列芯片手册、应用手册等资料以及各位工程师的使用和开发经验。在编写本书的过程中,我们得到了系统与应用工程师经理郭雷先生和中国区现场应用工程师经理张明峰先生的大力支持,同时还得到飞思卡尔公司全球产品经理 Mr. Jeff Bock、亚太市场经理曾劲涛先生和全球系统应用工程师经理 Mr. Clay Merritt 的大力支持,在此对他们以及所有关心本书编写和出版的朋友们表示诚挚的谢意。

由于时间与水平所限,本书难免会有错误,望谅解,并与编者联系指出错误,本人邮箱 shenli77@hotmail.com;也可以发送电子邮件到 emsbook@gmail.com,与本书策划编辑进行交流。

谌 利

飞思卡尔半导体公司应用工程师

2009 年 2 月于上海



第 1 章 ColdFire 基本介绍

1.1 ColdFire 的历史和概述	2
1.2 ColdFire 应用领域	5
1.2.1 工业控制领域	6
1.2.2 消费类电子领域	6
1.2.3 医疗电子领域	7
1.2.4 测试与测量	8
1.2.5 家庭及楼宇自动化	9
1.3 本书内容	9

第 2 章 ColdFire 内核及处理器架构介绍

2.1 ColdFire 内核基本介绍	11
2.2 ColdFire 内核结构	11
2.2.1 V2 内核架构	12
2.2.2 V3 内核架构	13
2.2.3 V4 内核架构	14
2.2.4 V4e 内核架构	17
2.3 内核主要寄存器	18
2.3.1 数据寄存器	18
2.3.2 地址寄存器	18
2.3.3 堆栈指针	18
2.3.4 程序指针	19
2.3.5 条件寄存器	19
2.3.6 异常中断向量基地址寄存器	19
2.3.7 状态寄存器	19
2.4 MAC 和 EMAC	19
2.4.1 MAC	19
2.4.2 EMAC	21
2.4.3 应用实例	21
2.5 高速缓存	22
2.5.1 ColdFire 缓存工作原理	22
2.5.2 主要寄存器	25
2.6 内部 SRAM 和内部 Flash	26

2.6.1 内部 SRAM	26
2.6.2 内部 Flash	26
2.7 ColdFire 处理器架构	31
2.7.1 CF5210 平台	31
2.7.2 标准产品平台	33
2.7.3 系统访问控制	35
2.8 基本指令集介绍	35
2.8.1 寻址模式	37
2.8.2 指令集	39
2.9 μ COS-II 在 ColdFire 上的移植	46
2.9.1 μ COS-II 移植的关键代码	46
2.9.2 OS_CPU.H	47
2.9.3 OS_CPU_C.C	48
2.9.4 OS_CPU_A.ASM	50
2.9.5 OS_CPU_I.ASM	55
第 3 章 编程开发工具	
3.1 开发工具概况	56
3.2 CodeWarrior for ColdFire	56
3.2.1 CodeWarrior 基本使用	57
3.2.2 项目配置	64
3.2.3 Link 文件语法	69
3.2.4 ColdWarrior 的默认库文件	71
3.2.5 烧写编程	73
3.2.6 调 试	76
3.3 Linux/ μ Clinux 开发环境——BSP	76
3.3.1 Linux/ μ Clinux for ColdFire 基本介绍	76
3.3.2 LTIB 使用	77
3.3.3 内核与文件系统的下载	80
3.3.4 调 试	81
3.4 IAR for ColdFire 基本介绍	83
3.4.1 IDE 环境介绍	83
3.4.2 编译器	84
3.4.3 调试器 C-SPY	86
第 4 章 内核异常与中断控制器	
4.1 内核异常与中断控制器的基本介绍	88
4.2 内核异常处理	88
4.2.1 异常中断处理的工作原理	88
4.2.2 中断向量表与异常介绍	91
4.3 中断控制器的介绍	94

4.3.1 中断优先级和中断级别	94
4.3.2 寄存器基本介绍	98
4.4 应用实例	99
4.4.1 中断控制器的初始化	99
4.4.2 中断向量表的初始化	100
4.4.3 中断服务程序的例程	103
第5章 Flex 总线和 Mini-Flex 总线	
5.1 Flex 总线基本介绍	105
5.2 硬件信号	106
5.3 寄存器介绍	109
5.4 工作模式	110
5.4.1 总线状态机和突发模式	110
5.4.2 时序分析	112
5.4.3 数据对齐和非对齐	119
5.5 应用实例	120
5.5.1 连接通用总线设备	120
5.5.2 Flex 总线与 EIM 的区别	123
第6章 SDRAM 控制器	
6.1 SDRAM 外部功能引脚支持	126
6.1.1 统一架构	126
6.1.2 伪分裂架构	127
6.1.3 全分裂架构	128
6.1.4 SDRAM 控制器的信号	129
6.2 SDRAM 控制寄存器简介	130
6.2.1 SDRAM 模式/扩展模式寄存器	130
6.2.2 SDRAM 控制寄存器	130
6.2.3 SDRAM 配置寄存器 1/2	131
6.3 SDR/DDR/DDR2 的功能比较	131
6.3.1 外部引脚功能比较	131
6.3.2 性能差异分析	132
6.4 应用案例	132
6.4.1 MCF5445x SDRAM 接口应用向导	132
6.4.2 硬件设计样例	133
6.4.3 DDR2 RAM 初始化样例	134
6.4.4 DDR2 硬件设计的布局参考	136
6.4.5 PCB 布线指导	138
第7章 USB 控制器	
7.1 USB 基本概述	141
7.2 MCU USB 模块介绍	145

7.2.1	MCU USB 模块概述	145
7.2.2	主机实现	150
7.2.3	设备类实现	151
7.2.4	人机接口设备类介绍	153
7.2.5	存储设备类实现	158
7.3	MPU USB 模块介绍	164
7.3.1	MPU USB 模块概述	164
7.3.2	USB 设备类的工作原理	166
7.3.3	USB 设备类例程	168
7.3.4	USB 主机类原理	174
7.3.5	USB 主机类例程	178
第 8 章 快速以太网控制器		
8.1	快速以太网控制器概述	186
8.2	以太网控制寄存器简介	188
8.3	以太网控制器外部功能引脚	190
8.3.1	功能引脚简介	190
8.3.2	MII 接口原理图	190
8.4	以太网控制器的中断控制	191
8.4.1	中断源简介	191
8.4.2	中断初始化样例	192
8.5	以太网控制器应用简介	194
8.5.1	缓冲区描述符	194
8.5.2	初始化启动流程	195
8.5.3	发送数据流程	196
8.5.4	接收数据流程	197
8.5.5	以太网控制器简单测试实例	198
8.6	应用案例——ColdFire_TCP/IP_Lite	200
8.6.1	简介	200
8.6.2	协议栈启动过程	202
8.6.3	NicheTask 实时操作系统	205
8.6.4	Mini Socket TCP API 简介	206
8.6.5	协议的流程分析样例	207
第 9 章 串行外设接口模块		
9.1	队列串行外设模块	210
9.1.1	QSPI 概述	210
9.1.2	QSPI 寄存器介绍	211
9.1.3	QSPI 工作原理与数据传输流程	212
9.1.4	QSPI 使用实例	217
9.2	DMA 串行外设接口模块	220

9.2.1 DSPI 概述	221
9.2.2 DSPI 寄存器介绍	222
9.2.3 DSPI 工作原理	225
9.2.4 DSPI 使用实例	228
9.3 EZPORT 模块	232
9.3.1 EZPORT 概述	232
9.3.2 EZPORT 命令集	233
9.3.3 EZPORT 使用实例	236
第 10 章 I2C 模块介绍与应用	
10.1 I2C 协议简介	243
10.2 I2C 模块框图和寄存器介绍	245
10.3 I2C 模块初始化流程	247
10.4 I2C 模块中断处理流程	249
10.5 I2C 模块应用实例——基于 NicheTask 的 LCD 驱动	257
第 11 章 FlexCAN 控制器	
11.1 FlexCAN 控制器寄存器简介	261
11.1.1 FlexCAN 模式寄存器	261
11.1.2 FlexCAN 控制寄存器	261
11.1.3 自由计时器	262
11.1.4 接收屏蔽寄存器	262
11.1.5 错误计数器	262
11.1.6 错误和状态寄存器	262
11.1.7 消息缓冲中断屏蔽寄存器	262
11.1.8 消息缓冲中断标志寄存器	262
11.1.9 消息缓冲	262
11.2 CAN 外部功能引脚简介	264
11.3 CAN 的中断控制	265
11.4 FlexCAN 应用向导	265
11.4.1 CAN 总线位时序的计算	265
11.4.2 FlexCAN 模块的振荡器容许公差	268
11.5 CAN 底层驱动简介	272
11.5.1 软件架构	272
11.5.2 API 函数简介	273
11.5.3 API 函数样例	284
第 12 章 DMA 与 EDMA 控制器介绍与应用	
12.1 DMA 控制器	286
12.1.1 DMA 控制器概述	286
12.1.2 DMA 寄存器介绍	287
12.1.3 DMA 控制器原理	289

12.1.4	DMA 使用实例	292
12.2	EDMA 控制器	296
12.2.1	EDMA 控制器概述	297
12.2.2	EDMA 寄存器介绍	297
12.2.3	EDMA 控制器原理	299
12.2.4	EDMA 应用实例	300
第 13 章 ColdFire 内置定时器		
13.1	ColdFire 定时器基本介绍	308
13.2	通用定时器	308
13.2.1	通用定时器的输入捕捉模式	308
13.2.2	通用定时器的输出比较模式	311
13.2.3	通用定时器的脉冲计数模式	313
13.2.4	通用定时器的 PWM 功能	314
13.3	可编程中断定时器	315
13.3.1	可编程中断定时器概述	315
13.3.2	应用实例	315
13.4	DMA 定时器	316
13.4.1	DMA 定时器概述	316
13.4.2	应用实例	318
13.5	实时时钟模块 RTC	319
第 14 章 脉宽调制模块		
14.1	简介	322
14.2	PWM 寄存器介绍	323
14.2.1	PWM 使能寄存器	323
14.2.2	PWM 极性控制寄存器	323
14.2.3	PWM 时钟源选择寄存器	323
14.2.4	PWM 时钟预分频选择寄存器	323
14.2.5	PWM 中央对齐使能寄存器	324
14.2.6	PWM 控制寄存器	324
14.2.7	PWM 比例寄存器 A 和 PWM 比例寄存器 B	324
14.2.8	PWM 通道计数器	324
14.2.9	PWM 通道周期寄存器	325
14.2.10	PWM 通道占空比寄存器	325
14.2.11	PWM 关闭寄存器	325
14.3	功能介绍	326
14.3.1	PWM 时钟源选择	326
14.3.2	PWM 定时器	327
14.4	PWM 使用实例	332

第 15 章 通用异步收发器

15.1 UART 模块概述	344
15.2 UART 工作简介	345
15.2.1 异步通信的数据格式	345
15.2.2 UART 的通道工作模式	345
15.2.3 UART 的中断	347
15.2.4 波特率计算	348
15.2.5 DMA 操作 UART 收发	348
15.2.6 UART 多点通信	349
15.3 UART 的寄存器	350
15.4 UART 的应用	352
15.4.1 UART 配置流程	352
15.4.2 例 程	352
15.4.3 UART 外围硬件设计	357

参考文献

第 1 章

ColdFire 基本介绍

为什么要使用 32 位处理器？

在嵌入式领域,对于早期的应用来说主要是基于 8 位和 16 位单片机,这些应用一般是任务单一、简单可靠的系统。随着应用领域对系统的功能和性能等方面提出更高的要求,8 位和 16 位单片机系统已经无法胜任,这时就需要基于 32 位系统架构的微处理器。一般来说,如果一个嵌入式系统有以下几个方面的特点,就需要采用 32 位微处理器:

- 当系统寻址范围大于 64 KB 时,所需的地址线位宽是 16 位,16 位单片机勉强能胜任,此时应该考虑 32 位系统;当寻址范围大于 1 MB 时,则地址位宽为 20 位,此时需要使用 32 位系统。
- 当需要在一个 8 位的系统上实现大于 20 MIPS 的性能时,或者在 16 位系统上实现大于 40 MIPS 的性能时,需要考虑使用 32 位系统。
- 当需要采用 Linux 操作系统时,处理器需要采用 32 位带内存管理单元 MMU;而 μ CLinux 则用于没有内存管理单元的 32 位微处理器。
- 当需要使用多层次的通信协议栈时,如 TCP/IP,采用 32 位处理器可以达到很好的通信效果和性能支持。
- 需要采用浮点运算或者高精度定点运算时,使用 32 位微架构可以达到更好的运算处理能力。

Freescape 为业界提供了广泛的 32 位微处理器产品线,从低端与 8 位单片机完全兼容的 V1 ColdFire 系列到最高端的 PowerPC 架构,无论用户有何种应用,总能够从整个产品线中找到适合的产品。图 1-1 显示的是 Freescape 32 位微处理器的产品线。

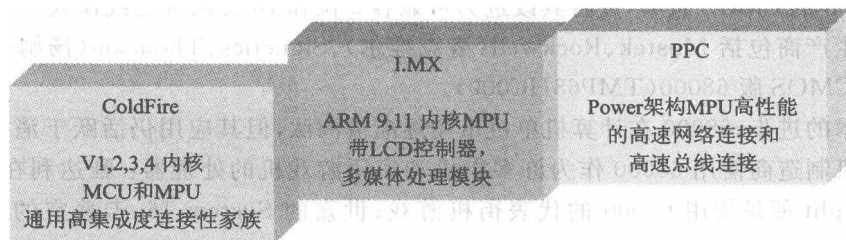


图 1-1 Freescape 32 位微处理器家族

1.1 ColdFire 的历史和概述

摩托罗拉公司于 1976 年启动 MACSS 项目(Motorola Advanced Computer System on Silicon, 摩托罗拉硅晶高级计算机系统), 打算开发一款与以前产品完全不兼容的全新微处理器。根据计划, 新 CPU 应该是对当时摩托罗拉主流 8 位 CPU 6800 的一个高端互补产品, 因此不会考虑两者间的兼容性。不过, 当 68000 设计出来后, 它还是被保留了一个可兼容 6800 外设的总线协议模式, 并且专门有 8 位数据总线的产品被生产出来。当然, 设计人员还是更在意于其向后兼容性, 这为 68000 在 32 位 CPU 领域确立领先优势奠定了基础。例如, 68000 使用 32 位寄存器和内部总线, 尽管其本身的结构很少直接操作长字。小型计算机诸如 PDP-11 和 VAX(二者采用了类似的微编码)对 68000 的设计有深刻的影响。

20 世纪 70 年代中期, 8 位微处理器生产商纷纷竞争导入下一代 16 位 CPU。国家半导体在 1973—1975 年间首先开发出了 IMP-16 和 PACE, 但其速度并不理想。英特尔公司于 1977 年推出 8086, 迅速受到欢迎。此时, 为确保竞争上的领先, 摩托罗拉认识到其 MACSS 项目必须跳过 16 位系统, 而直接推出 16/32 位混合型 CPU。到 1979 年, 摩托罗拉 68000, 即 MC68000, 才姗姗来迟。由于比 8086 晚两年, 其晶体管数目更多, 并因其易用性得到了好评。

最初的 MC68000 使用 $3.5\ \mu\text{m}$ HMOS 技术(即高性能 N 通道金属氧化物半导体, CMOS 的前身)制造; 1979 年发布了工程样品, 次年产品型面世, 速度有 4、6、8、10 MHz 多种; 最快的 16.67 MHz 版本到 20 世纪 80 年代末才面市。MC68000 在早期得到了很多高端产品的青睐。在 Sun 公司的 Sun workstation 等多种 Unix 工作站中, MC68000 一度占统治地位。市场领先的其他一些计算机, 包括 Amiga(阿米加)、Atari ST(雅达利 ST)、Apple Lisa(苹果 Lisa)和 Macintosh(麦金托什), 以及第一代激光打印机, 如苹果公司的 LaserWriter, 都使用 MC68000。1982 年, 摩托罗拉进一步更新了 MC68000 的指令集以支持虚拟内存, 并使其能够满足由 Popek 和 Goldberg 于 1974 年提出的虚拟化标准。为支持低成本系统和使用较少内存的应用, 摩托罗拉于 1982 年推出了面向 8 位外部数据总线的 MC68008。1982 年以后, 摩托罗拉开始把更多的注意力投向 68020 和 88000。

由日立公司设计, 与摩托罗拉公司于 1985 年联合推出了使用 CMOS 技术的 68HC000。68HC000 的速度有 8~20 MHz 多个版本。除了使用 CMOS 电路, 68HC000 与基于 HMOS 的 MC68000 完全一致, 但正是因此其能耗得到大幅下降。MC68000 在 25℃ 环境下能耗大约为 1.35 W, 而 8 MHz 的 68HC000 能耗为 0.13 W, 较高频率的版本能耗也相应提高(HMOS 技术则不同, 其在 CPU 空闲时仍会耗电, 因此功耗与频率基本无关)。1990 年摩托罗拉又推出了 MC68008 的 CMOS 版本, 并将其改进为可兼容 8 位和 16 位两种总线模式。其他 HMOS 版 68000 的生产商包括 Mostek、Rockwell(洛克维尔)、Signetics、Thomson(汤姆逊)和东芝。东芝也生产 CMOS 版 68000(TMP68HC000)。

随着技术的进步, 68000 在计算机单机市场逐渐被淘汰, 但其应用仍活跃于消费和嵌入式领域。游戏机制造商使用 68000 作为许多街机和家用游戏机的处理器: 雅达利在 1983 年推出的 Food Fight 便是使用 68000 的代表街机游戏; 世嘉的 System 16、卡普空的 CPS-1 和 CPS-2 以及 SNK 的 Neo Geo 也都使用 68000。到了 20 世纪 90 年代, 尽管街机游戏开始使用更加强大的 CPU, 68000 仍被用作声音控制器。20 世纪 80 年代末到 90 年代初, 一些游戏