

ARM Cortex-M4 微控制器原理与应用

——基于Atmel SAM4系列

毕 盛 钟汉如 董 敏 编著



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

ARM Cortex-M4微控制器原理与应用

——基于 Atmel SAM4 系列

毕盛 钟汉如 董敏 编著

北京航空航天大学出版社

本书详细讲解了 ARM Cortex-M4 微控制器的内部结构、工作原理、寄存器配置、外设接口、中断系统、低功耗模式、应用案例等。全书共分 14 章，内容包括 ARM Cortex-M4 内核、系统架构、外设接口、中断系统、低功耗模式、应用案例等。全书共分 14 章，内容包括 ARM Cortex-M4 内核、系统架构、外设接口、中断系统、低功耗模式、应用案例等。

内 容 简 介

本书以具有 ARM Cortex-M4 内核的 Atmel 公司 SAM4E 微控制器为蓝本讲述嵌入式开发技术。内容包括 ARM Cortex-M4 内核、系统架构、电路设计、程序设计入门、标准外设库应用、通用输入输出(GPIO)、通用异步/同步串行通信(UART/USART)、通用定时器/计数器(Timer/Counter)、实时定时器(RTT)、实时时钟(RTC)、看门狗定时器(WDT)、增强安全看门狗定时器(RSWDT)、PWM 模块、同步串行通信接口(SPI)、TWI 总线(I²C)、控制器局域网络(CAN)、以太网通信接口(GMAC)、USB 全速串行通信模块(UDP)、模拟前端控制器(AFEC)模块、数字/模拟转换控制器(DACC)模块、模拟比较控制器(ACC)模块、DMA、外设 DMA(PDC)、总线矩阵(MA-TRIX)、高速多媒体存储卡接口(HSMCI)、加密模块(AES)、SysTick 定时器、FPU 单元及浮点数运算和 DSP 指令与 DSP 库接口及应用。

本书中所有实例源代码可在北京航空航天大学出版社网站的“下载专区”免费下载。

本书可作为高等院校电子工程、自动化、计算机科学及技术和电气工程等专业的教材和参考书,也可供相关工程技术人员参考。

图书在版编目(CIP)数据

ARM Cortex-M4 微控制器原理与应用:基于 Atmel
SAM4 系列 / 毕盛,钟汉如,董敏编著. -- 北京:北京
航空航天大学出版社,2014.10

ISBN 978-7-5124-1395-5

I. ①A… II. ①毕… ②钟… ③董… III. ①微处理
器—系统设计 IV. ①TP332

中国版本图书馆 CIP 数据核字(2014)第 205380 号

版权所有,侵权必究。

ARM Cortex-M4 微控制器原理与应用——基于 Atmel SAM4 系列

毕 盛 钟汉如 董 敏 编著
责任编辑 张军香 朱江芳 宋显民

*

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)82316524

涿州市新华印刷有限公司印装 各地书店经销

*

开本:710×1 000 1/16 印张:30.25 字数:645 千字

2014 年 10 月第 1 版 2014 年 10 月第 1 次印刷 印数:3 000 册

ISBN 978-7-5124-1395-5 定价:69.00 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

ARM Cortex-M4 处理器是由 ARM 公司开发的最新嵌入式处理器,用于有效且易于使用的控制和信号处理功能混合的数字信号控制场合,与 Cortex-M3 相比,增加了快速数字信号处理模块。

Atmel 公司 SAM4S/SAM4L/SAM4E 系列微处理器是家族成员式的高性能 32 位 ARM Cortex-M4 的 RISC 处理器。其最大工作频率为 120 MHz,并具有多达 2 048 KB 的闪存、可选双组实施和快取记忆体及高达 160 KB 的 SRAM。其外设集成了全速 USB 设备端口和高速 SDIO/SD/MMC 接口,提供连接到 SRAM、PSRAM 和 NOR FLASH 静态内存控制器的外部总线接口,具有液晶显示模块和 NAND 闪存、USART 接口、UART 接口、TWI 接口、SPI 接口、CAN 接口、GMAC 以太网接口、PWM 定时器、通用 16 位定时器(支持步进电机及正交解码器逻辑)、RTC、12 位 ADC、12 位 DAC 和模拟比较器等。

全书共 14 章,内容包括 ARM Cortex-M4 内核、系统架构、电路设计、程序设计入门、标准外设库应用、通用输入/输出(GPIO)、通用异步/同步串行通信(UART/USART)、通用定时器/计数器(Timer/Counter)、实时定时器(RTT)、实时时钟(RTC)、看门狗定时器(WDT)、增强安全看门狗定时器(RSWDT)、PWM 模块、同步串行通信接口(SPI)、TWI 总线(I²C)、控制器局域网络(CAN)、以太网通信接口(GMAC)、USB 全速串行通信模块(UDP)、模拟前端控制器(AFEC)、数字/模拟转换控制器(DACC)模块、模拟比较控制器(ACC)模块、DMA、外设 DMA(PDC)、总线矩阵(MATRIX)、高速多媒体存储卡接口(HSMCI)、加密模块(AES)、SysTick 定时器、FPU 单元及浮点数运算、DSP 指令、DSP 库接口及应用。

ARM 实例程序的编译环境是 Atmel Studio 6,配套资料包括书中所有的实例代码,读者可直接从北京航空航天大学出版社网站(www.buaapress.com.cn)“下载专区”免费下载使用。

本书主要由毕盛、董敏和钟汉如编写,其中第 1、2 章由钟汉如编写,第 3~14 章由毕盛和董敏编写。黄铨雍、邱荣财、韦锐平、韦如明、曾潇、金泽豪、陈奇石、曹丹、黄

鑫龙、钟建兴、陈德智等同学参与了翻译 Atmel 原著及实例验证工作。在编写本书过程中,还得到 Atmel 公司的支持和 Atmel 中国区微处理器应用支持团队的帮助,在此表示衷心的感谢。

由于编写时间比较仓促,编者水平有限,书中难免存在错误和疏漏之处,恳请读者批评指正!

编者

2014 年 4 月

目 录

第 1 章 ARM 系列及 Atmel Cortex-M4 芯片	1
1.1 ARM 芯片类别及体系结构	1
1.1.1 ARM 芯片主要类别	1
1.1.2 ARM 处理器体系结构	6
1.1.3 安全内核 SecurCore 处理器与 FPGA 可编程逻辑门阵列	10
1.2 ARM Cortex-M4	12
1.2.1 Cortex-M4 功能说明	12
1.2.2 Cortex-M4 模式	16
1.2.3 内存映射	19
1.2.4 电源管理的睡眠模式	23
1.2.5 Cortex-M4 核心外设	24
1.3 Atmel 公司的 SAM4S/SAM4L/SAM4E 系列 ARM 芯片配置	25
1.3.1 SAM4S 系列配置	25
1.3.2 SAM4L 系列配置	26
1.3.3 SAM4E 系列配置	28
第 2 章 SAM4E 系列 MCU 芯片及 Atmel SAM4E - EK 开发板	30
2.1 SAM4E 系列 MCU 引脚	30
2.1.1 SAM4E MCU 内部结构图	30
2.1.2 SAM4E MCU 引脚说明和功能	32
2.2 SAM4E 开发板说明	33
2.2.1 SAM4E - EK 开发板主要功能及特点	33
2.2.2 开发板电路介绍	34
2.2.3 I/O 口复用引脚	48
2.3 SAM4E 总线 APB/AHB 桥	53
2.3.1 总线 APB/AHB 桥	53
2.3.2 总线矩阵	54
2.4 SAM4E 主要模块	55
2.4.1 DMA 控制器概述	55

2.4.2	外设 DMA 控制器概述	56
2.4.3	SAM4E 处理器核心模块概述	57
第 3 章	SAM4 GPIO 及程序开发	59
3.1	SAM4 GPIO	59
3.1.1	GPIO 结构与特点	59
3.1.2	GPIO 功能描述	63
3.2	Atmel Studio 开发环境介绍	72
3.2.1	Atmel Studio 6 开发环境安装	73
3.2.2	Atmel Studio 环境下第一个 SAM4E 程序	74
3.3	Atmel 软件框架	82
3.3.1	ASF 架构	82
3.3.2	ASF 功能描述	84
3.3.3	利用 ASF 实现第一个 SAM4E 程序	87
3.4	Cortex 微控制器软件接口标准	89
3.4.1	CMSIS 架构	89
3.4.2	Atmel Studio 6 利用 CMSIS 应用实例	91
第 4 章	SAM4 供电和时钟管理	92
4.1	SAM4 电源管理	92
4.1.1	内部供电结构	92
4.1.2	供电管理模式	93
4.1.3	供电控制器	96
4.2	功耗管理控制器及时钟配置	102
4.2.1	时钟发生器	103
4.2.2	时钟管理	108
4.2.3	主控时钟设置说明	115
第 5 章	SAM4 中断/DMAC/PDC/总线矩阵	121
5.1	SAM4 嵌套向量中断控制器	121
5.1.1	NVIC 功能描述	121
5.1.2	NVIC 程序说明及应用实例	122
5.2	DMA 控制器	128
5.2.1	DMAC 功能描述	128
5.2.2	DMAC 应用实例	131
5.3	外设 DMA 控制器	135
5.3.1	PDC 功能概述	135

5.3.2 PDC 实例说明	138
5.4 总线矩阵	138
5.4.1 Matrix 功能描述	138
5.4.2 Matrix 实例说明	142
第 6 章 SAM4 串行通信	143
6.1 异步串行通信	144
6.1.1 UART 概述	144
6.1.2 UART 功能描述	145
6.1.3 UART 应用实例	149
6.2 同步/异步串行通信	156
6.2.1 USART 概述	156
6.2.2 USART 功能描述	158
6.2.3 USART 扩展工作模式	177
6.2.4 USART 应用实例	188
6.3 同步串行通信接口	196
6.3.1 SPI 概述	196
6.3.2 SPI 功能描述	198
6.3.3 SPI 应用实例	205
第 7 章 SAM4 串行通信总线	211
7.1 TWI 总线	211
7.1.1 TWI 概述	211
7.1.2 TWI 功能描述	213
7.1.3 TWI 应用实例	227
7.2 控制器局域网	232
7.2.1 CAN 总线概述	232
7.2.2 CAN 控制器特性	235
7.2.3 CAN 功能描述	243
7.2.4 CAN 总线应用实例	252
第 8 章 SAM4 定时器/计数器相关模块	259
8.1 系统定时器	259
8.2 通用定时器/计数器	262
8.2.1 TC 概述	262
8.2.2 TC 功能描述	264
8.2.3 TC 应用实例	276

8.3	脉宽调制控制器	278
8.3.1	PWM 概述	278
8.3.2	PWM 功能描述	280
8.3.3	PWM 应用实例	301
8.4	实时定时器	305
8.4.1	RTT 概述	305
8.4.2	RTT 功能描述	305
8.4.3	RTT 应用实例	306
8.5	实时时钟	309
8.5.1	RTC 概述	309
8.5.2	RTC 功能描述	310
第 9 章	SAM4 模拟电压相关模块	314
9.1	模拟前端控制器	314
9.1.1	AFEC 概述	314
9.1.2	AFEC 功能描述	317
9.1.3	AFEC 应用实例	328
9.2	数字/模拟转换控制器	334
9.2.1	DACC 概述	334
9.2.2	DACC 功能描述	336
9.3	模拟比较控制器	338
9.3.1	ACC 概述	338
9.3.2	ACC 功能描述	340
第 10 章	SAM4 高级通信模块	342
10.1	以太网 MAC	342
10.1.1	GMAC 概述	342
10.1.2	GMAC 功能描述	344
10.1.3	GMAC 编程接口	366
10.1.4	GMAC 应用实例	370
10.2	USB 设备端口	374
10.2.1	UDP 概述	374
10.2.2	UDP 功能描述	377
10.2.3	UDP 应用实例	388
第 11 章	SAM4 数字信号处理模块	390
11.1	SAM4 FPU 单元及浮点数运算	390

11.1.1 FPU 模块介绍	390
11.1.2 SAM4 FPU 应用实例	393
11.2 SAM4 DSP 指令及 DSP 库	397
11.2.1 DSP 模块介绍	397
11.2.2 DSP 模块功能	398
11.2.3 SAM4 DSP 应用实例	400
第 12 章 SAM4 存储模块及接口	404
12.1 SAM4 内嵌存储器及控制器	404
12.1.1 SAM4 内嵌存储器概述	404
12.1.2 内嵌 FLASH 及控制器	404
12.1.3 快速 FLASH 编程接口	413
12.1.4 SAM4 启动与引导装载程序	416
12.2 Cortex-M 缓存控制器	421
12.3 静态存储控制器	423
12.3.1 静态存储控制器 SMC 概述	423
12.3.2 SMC 存储器连接	425
12.3.3 SMC 控制 LCD 实例	427
12.4 高速多媒体存储卡接口	430
12.4.1 高速多媒体存储卡接口 HSMCI 概述	430
12.4.2 HSMCI 功能描述	433
第 13 章 SAM4 RESET 及其他模块	445
13.1 RESET 控制器	445
13.2 看门狗定时器	449
13.3 增强安全的看门狗定时器	451
13.4 高级加密标准	453
13.4.1 AES 概述	453
13.4.2 AES 功能描述	454
13.5 芯片标识符	457
第 14 章 SAM4 综合应用实例	459
14.1 综合实例介绍	459
14.2 硬件说明	460
14.3 软件说明	460
参考文献	470

第 1 章

ARM 系列及 Atmel Cortex-M4 芯片

本章主要讲解 ARM 系列,其中包括 ARM 芯片类别及 Cortex-M4 体系结构,并对 Atmel 公司的 Cortex-M4 芯片 3 种主要系列 SAM4S、SAM4L 和 SAM4E 芯片的配置作介绍。

1.1 ARM 芯片类别及体系结构

ARM 公司在 32 位精简指令集微处理器(RISC, Reduced Instruction Set Computer)领域不断取得突破,其结构已经从 V3 发展到 V8。ARM 公司自成立以来一直以 IP(Intelligence Property)提供者的身份向各大半导体制造商出售知识产权,从不介入芯片的生产销售,加上其设计的芯片内核具有低功耗、低成本等显著优点,因此获得众多的半导体厂家和整机厂商的大力支持,在 32 位嵌入式应用领域获得了巨大的成功,目前已经占有 75% 以上的 32 位 RISC 嵌入式产品市场,在低功耗、低成本的嵌入式应用领域确立了市场地位。使用 ARM 公司芯核包装外围接口电路设计,生产 ARM 芯片的国际大公司已经超过 70 多家,国内中兴通信股份有限公司和华为通信股份有限公司也已经购买 ARM 公司的芯核技术用于通信专用芯片的设计。

1.1.1 ARM 芯片主要类别

主流 ARM 芯片从十几年前进入中国,已从最初的 ARM7、ARM9、ARM11 等系列芯片发展到目前的 ARM Cortex 系列芯片。

在 ARM7~ARM11 阶段,ARM 芯片主要被用于嵌入式微处理的功能(MPU)。由于 ARM7 芯片结构中没有内存管理单元(MMU),所以当时人们在编写逻辑程序时,主要使用一些不需要做内存地址映射的操作系统,例如 μ Cos 和 μ Clinux 等操作系统,所以也较好地推动了当时这种类型操作系统在嵌入式芯片上的发展。ARM9 芯片有了内存管理单元,所以人们更喜欢在此系统上直接运行 Linux 系统,所以也推动了当时 ARM Linux 操作系统的发展。ARM11 对 ARM9 的功能进行了进一步的加强,添加了多媒体协处理器,所以有更好的人机交互功能。其中苹果 1 就是针对 ARM11 开发出来的一款嵌入式应用芯片。

从 ARM11 芯片继续发展进入 ARM Cortex 系列芯片时代,包括 Cortex-A、Cortex-R 和 Cortex-M 三个系列。Cortex-A 和 Cortex-R 系列芯片还保持嵌入式微处理的功能(MPU),通过扩展存储器可以运行 Linux、Android、IOS 等多种复杂的操作系统。ARM 公司为了占据单片机市场,把 Cortex-M 系列芯片用于一种 32 位单片机,所以,Cortex-M 系列芯片具有丰富的外设接口,但没有运行如 Linux、Android、IOS 等多种复杂操作系统的内存管理单元。

根据 ARM 系列单片机的发展,把 ARM 芯片分为三大类。第一类为经典 ARM 处理器,其中包括 ARM7~ARM11 系列处理器;第二类为嵌入式 Cortex 处理器,其中包括 ARM Cortex-M 系列处理器,如图 1-1 所示 ARM 处理器类别图中的 Cortex-M0、Cortex-M0+、Cortex-M1、Cortex-M3、Cortex-M4 和 Cortex-R4、Cortex-R5、Cortex-R6 等处理器;第三类为应用类的 Cortex 处理器,其中包括 Cortex-A5、Cortex-A7、Cortex-A8、Cortex-A9 和 Cortex-A15 系列处理器。

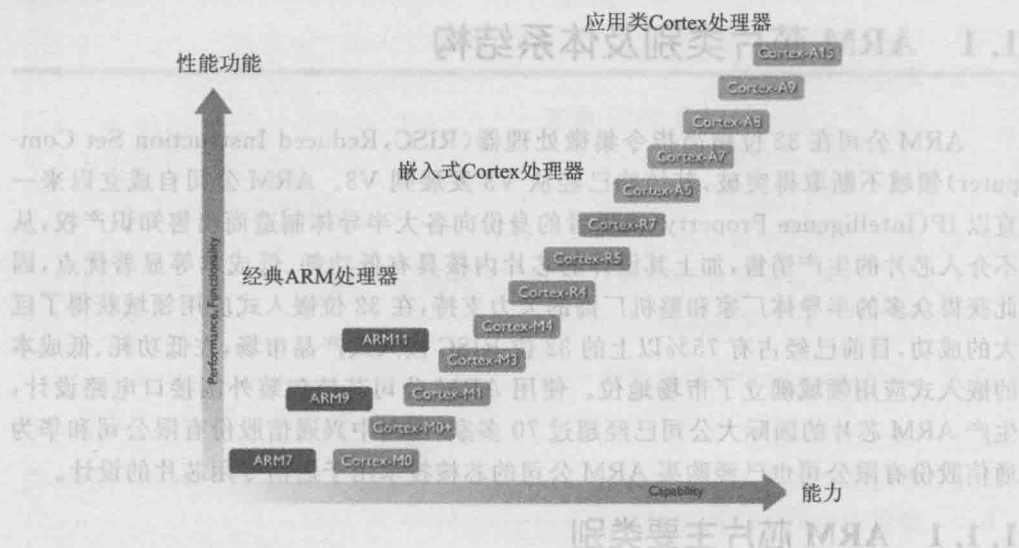


图 1-1 ARM 处理器类别

1. 经典 ARM 处理器

ARM 经典处理器包括 ARM7、ARM9、ARM11,在中国市场以韩国三星 ARM9 最为突出,飞利浦 ARM7 走低端应用。在新应用中应使用其技术经过市场验证的组织处理器,这些处理器提供了众多的特性、卓越的功效和范围广泛的适应能力,适用于低成本解决方案。其中:

ARM7 系列——基于 ARMv4T 架构面向普通应用的经典处理器,例如飞利浦 LPC21XX ARM7 系列芯片。

ARM9 系列——基于 ARMv5 架构的常用处理器,例如韩国三星的 S3C24XX ARM9 系列芯片。

ARM11 系列——基于 ARMv6 架构的高性能处理器,例如韩国三星的 S3C6410 芯片。

2. ARM 处理器 Cortex-M 系列

Cortex-M 系列面向成本敏感的微控制器解决方案。Cortex-M 系列处理器主要是针对微控制器领域开发的,在该领域中,既需进行快速且具有高确定性的中断管理,又需将门数和功耗控制在最低。应用包括:混合信号设备、智能传感器、汽车电子等广泛的微控制器方案应用领域。

Cortex-M0 处理器是市场上现有的体积最小、能耗最低、最节能的 ARM 处理器。Cortex-M0 基于 ARMv6M 架构,该处理器能耗非常低,门数量少,代码占用空间小,使得 MCU 开发人员能够以 8 位处理器的价位,获得 32 位处理器的性能。超低门数还使其能够用于模拟信号设备和混合信号设备及 MCU 应用中,可望明显降低系统成本。现在已有多家公司获得 Cortex-M0 处理器授权,如 STMicroelectronics STM32 F0 系列和 NXP LPC11xx、LPC12xx 系列等。

Cortex-M3 处理器具有较高的性能和较低的动态功耗,能够提供领先的能效。Cortex-M3 基于 ARMv7M 架构,将集成的睡眠模式与可选的状态保留功能相结合,确保在同时需要低能耗和出色性能的应用中不存在折衷。该处理器执行包括硬件除法、单周期乘法和位字段操作在内的 Thumb-2 指令集,可获取最佳性能和代码大小。Cortex-M3 NVIC 在设计时是高度可配置的,最多可提供 240 个具有单独优先级、动态重设优先级功能和集成系统时钟的系统中断。

Cortex-M4 处理器是由 ARM 专门开发的最新嵌入式处理器,用来满足需要有效且易于使用的控制和信号处理功能混合的数字信号控制市场,针对 Cortex-M3 添加了快速数字信号处理模块,具有高性能的数字信号控制。它采用扩展的单周期乘法累加(MAC)指令、优化的 SIMD 运算、饱和运算指令和一个可选的单精度浮点单元(FPU),具备最佳的数字信号控制操作所需的所有功能,还结合了深受市场认可的 Cortex-M 系列处理器的低功耗特点。常见的 Cortex-M4 系列,有 Atmel 公司推出的 SAM4 系列、ST 公司的 STM32F4 系列、飞思卡尔的 Kinetis 系列及 NXP 公司的 LPC4300 系列。

3. ARM 处理器 Cortex-A、Cortex-R 系列

ARM 处理器 Cortex-A 是以 Cortex-A5、Cortex-A7、Cortex-A8、Cortex-A9、Cortex-A15 为代表的应⽤类 Cortex 处理器。这些开放式处理器有高性能的兼容操作系统软件支持。应⽤类 Cortex 处理器在制作工艺中可实现高达 2 GHz+ 的标准工作频率,其性能卓越,可支持下一代移动 Internet 设备。这些处理器具有单核和多核种类,最多提供 4 个具有可选 NEON 多媒体处理模块和高级浮点执行单元的处理单元。其应⽤包括智能手机、智能本和上网本、电子书阅读器、数字电视、家用网关及各种其他产品。

Cortex-A5、Cortex-A7、Cortex-A8、Cortex-A9 和 Cortex-A15 处理器适用于各种不同的性能应用领域。Cortex-A 处理器比较如表 1-1 所列。

表 1-1 Cortex-A 处理器比较

内 核	Cortex-A5	Cortex-A5 MPCore	Cortex-A8	Cortex-A9	Cortex-A9 MPCore	Cortex-A9 硬件	Cortex-A15 MPCore	Cortex-A7 MPCore
架 构	ARMv7	ARMv7 +MP	ARMv7	ARMv7	ARMv7 +MP	ARMv7 +MP	ARMv7+ MP+LPAE	ARMv7+ MP+LPAE
中断控 制器	GIC-390	已集成 -GIC	GIC-390	GIC-390	已集成 -GIC	已集成 -GIC	已集成 -GIC	GIC-400
L2 高速 缓存控 制器	L2C-310	L2C-310	已集成	L2C-310	L2C-310	L2C-310	已集成	已集成
主频/ MHz	300~800	300~800	600~1 000	600~1 000	600~1 000	800~2 000	1 000~ 2 500	800~1 500
DMIPS/ MHz	1.6	1.6(每个 CPU)	2.0	2.5	2.5 (每个 CPU)	5.0 (双核)	TBC	1.9 (每个 CPU)

所有 Cortex-A 处理器都共享共同的架构和功能集,在开放式平台之间软件具有兼容性和可移植性。ARMv7-A 架构支持如下操作系统:Linux 全部分发版本——Android、Chrome、Ubuntu 和 Debian;Linux 第三方——Monta Vista、QNX、Wind River Vxworks 和 Symbian;Windows CE,以及包括需要使用内存管理单元的其他操作系统。其指令集支持 ARM、Thumb-2、Thumb、Jazelle、DSP。另外,还具有 TrustZone 安全扩展、高级单精度和双精度浮点支持及 NEON 媒体处理引擎等特点。

ARM 公司通过提供所需高性能特点和可伸缩性,提供所需能效和低成本,同时维持完整的软件兼容性,各种 Cortex-A 处理器可共同提供设计的灵活性。

Cortex-A5 处理器隶属于 Cortex-A 系列,基于 ARMv7-A 架构,是能效最高、成本最低的处理。Cortex-A5 处理器可为现有 ARM9 和 ARM11 处理器设计提供很有价值的迁移途径,可获得比 ARM1176JZ-S 更好的性能,比 ARM926EJ-S 更好的功效和能效。另外,Cortex-A5 处理器不仅在指令及功能方面与更高性能的 Cortex-A8、Cortex-A9 和 Cortex-A15 处理器完全兼容,同时还保持与经典 ARM 处理器(包括 ARM926EJ-S、ARM1176JZ-S 和 ARM7TDMI)的向后应用程序兼容性。

Cortex-A7 处理器隶属于 Cortex-A 系列,基于 ARMv7-A 架构,其特点是在保证性能的基础上提供出色的低功耗表现。Cortex-A7 处理器的体系结构和功能集与 Cortex-A15 处理器完全相同,不同之处在于 Cortex-A7 处理器的微体系结构侧重于

提供最佳能效,因此这两种处理器可在各种芯核处理器结构(big. LITTLE 处理设计)配置中协同工作,从而提供高性能与超低功耗的最终组合芯片。单个 Cortex-A7 处理器的能源效率是 ARM Cortex-A8 处理器的 5 倍,性能提升 50%,尺寸仅为后者的 1/5。作为独立处理器,Cortex-A7 可以使 2013—2014 年期间低于 100 美元价格点的入门级智能手机与 2010 年 500 美元的高端智能手机相媲美。这些入门级智能手机在发展中可重新定义和互联网的连接。

Cortex-A8 处理器属于 Cortex-A 系列,基于 ARMv7-A 架构,是目前使用的单核手机中最常见的产品。ARM Cortex-A8 处理器是首款基于 ARMv7 体系结构的产品,能够将速度从 600 MHz 提高到 1 GHz 以上。Cortex-A8 处理器可以满足功耗在 300 mW 以下运行的移动设备的功率优化要求;也可以满足需要 2 000 Dhrystone MIPS 的消费类应用领域的性能优化要求。Cortex-A8 高性能处理器目前已经非常成熟,从高端特色手机到上网本、DTV、打印机和汽车信息娱乐,Cortex-A8 处理器都提供了可靠的高性能解决方案。

Cortex-A9 处理器属于 Cortex-A 系列,基于 ARMv7-A 架构,目前能见到的四核处理器大多属于 Cortex-A9 系列。Cortex-A9 处理器的设计旨在打造最先进、高效率、长度动态可变、多指令执行的超标量体系结构,提供采用乱序猜测方式执行的 8 段管道处理器。凭借范围广泛的消费类、网络、企业和移动应用中的前沿产品所需的功能,可以提供史无前例的高性能和高能效。Cortex-A9 微体系结构既可用于可伸缩的多核处理器(Cortex-A9 MPCore 多核处理器),也可用于更传统的处理器(Cortex-A9 单核处理器)。可伸缩的多核处理器和单核处理器支持 16 KB、32 KB 或 64 KB 4 路关联的 L1 高速缓存配置,对于可选的 L2 高速缓存控制器,最多支持 8 MB 的 L2 高速缓存配置,具有极高的灵活性,均适用于特定应用领域和市场。

Cortex-A15 处理器属于 Cortex-A 系列,基于 ARMv7-A 架构,是业界迄今为止性能最高且可授予生产许可的处理器。Cortex-A15 MPCore 处理器具有无序超标量管道,带有紧密耦合的低延迟 2 级高速缓存,该高速缓存的容量最高可达 4 MB。浮点和 NEON 媒体性能方面的其他改进使设备能够为消费者提供下一代用户体验,并为 Web 基础结构应用提供高性能计算。Cortex-A15 处理器可以应用在智能手机、平板电脑、移动计算、高端数字家电、服务器和无线基础结构等设备上。总之,Cortex-A15 MPCore 处理器的移动配置所能提供的性能是当前的高级智能手机性能的 5 倍还多。在高级基础结构应用中,Cortex-A15 的运行速度最高可达 2.5 GHz,这将支持在不断降低功耗、散热和成本预算方面实现高度可伸缩的解决方案。

ARM Cortex-R 实时处理器为高可靠性、高可用性,具容错功能、可维护性和实时响应的嵌入式系统提供高性能的计算解决方案。Cortex-R 系列具有满足许多应用的关键特性,即:高性能——与高时钟频率相结合的快速处理能力;实时性——处理能力在所有场合都符合硬实时限制;安全性——具有高容错能力的可靠且可信的系统;经济实惠——可实现最佳性能、功耗和面积的功能。Cortex-R 系列包括

Cortex-R4、Cortex-R5 和 Cortex-R7 处理器,其发展的基础是深层嵌入式市场和实时市场(如汽车安全或无线基带)所要求的主要性能,即高性能、实时性、安全性和经济实惠。

Cortex-R4 处理器是为实现高级芯片工艺而设计的,其设计重点是更高的能效、实时的响应速度、高级功能和简单的系统设计。该处理器提供高度灵活且有效的双周期本地内存接口,使 SoC 设计者可以最大限度地降低系统成本和功耗。

Cortex-R5 处理器集成了许多高级系统级功能,以帮助设计者进行软件开发,提高安全性和整个系统的可靠性。这些功能中包括低延迟外设端口(LLPP),这是一个一致性接口,用于实现并保持 Cortex-R5 Cache 与智能外设传输的数据完全同步,支持能够扩展至所有处理器接口的增强型错误检查和纠正(Error Checking and Correcting,ECC)。

Cortex-R7 处理器为范围广泛的深层嵌入式应用提供了高性能的双核、实时解决方案。Cortex-R7 处理器通过引入新技术(包括乱序指令执行和动态寄存器重命名),并与改进的分支预测、超标量执行功能和用于除法、DSP 和浮点函数的更快的硬件支持相结合,提供了比其他 Cortex-R 系列处理器高得多的性能级别。

1.1.2 ARM 处理器体系结构

1. ARM 处理器体系结构概述

ARM 处理器体系结构如图 1-2 所示,可划分为三大类别,分属于不同的颜色系列:

① 经典 ARM 处理器,属于蓝色芯核,包括 ARMv4T 的 SC100 和 ARM7TDM1,ARMv5TJ 的 ARM926、ARM968、ARM946,ARMv6 的 ARM11MP、ARM176JZ、ARM1136J 和 ARM1156T2。

② 应用类 Cortex 处理器,属于黄色芯核,包括 ARMv7A/R 的 Cortex-A15、Cortex-A9。

③ 嵌入式 Cortex 处理器,属于绿色芯核,包括 ARMv7A/R 的 Cortex-R7、Cortex-R5、Cortex-R4,ARMv7M/ME 的 SC300、Cortex-M3、Cortex-M4,ARMv6M 的 SC000、Cortex-M1、Cortex-M0+、Cortex-M0。

图中横轴芯片体系结构 ARMv4~ARMv7 包含了 ARM7~ARM11、Cortex-A、Cortex-R、Cortex-M 处理器芯片,这些嵌入式芯核采用 ARM 32-Bit ISA 和 Thumb 16-Bit ISA 指令集,不同的 ARM 芯核指令集向下兼容。有些芯核采用 VFP 浮点指令体系结构、NVIC 嵌套向量中断控制器指令结构,这些指令集提供 TrustZone 安全扩展、可信计算指令功能、WIC(Wake-Up Interrupt Controller)唤醒中断控制器指令功能和 NVIC 低功耗睡眠的模式。

ARM 处理器体系指令集功能结构解析如下:

ARM 32-Bit ISA:基于 RISC 原理的 32 位 ARM 指令集。

Thumb 16-Bit ISA:Thumb 技术是对 32 位 ARM 体系结构的扩展。Thumb

指令集是已压缩至 16 位宽操作码的、最常用 32 位 ARM 指令集的子集。在执行时, 这些 16 位指令实时、透明地解压缩为完整 32 位 ARM 指令, 且无性能损失。其卓越的代码密度, 可尽量减小系统内存大小和降低成本。

Thumb-2 指令集提供最佳代码大小和性能, 以 ARM Cortex 体系结构为基础, 提升了众多嵌入式应用的性能、能效和代码密度。以获得成功的 Thumb (ARM 微处理器内核的创新型高代码密度指令集) 为基础进行构建, 增强 ARM 微处理器内核的功能, 使开发人员能够开发出低成本且高性能的系统。

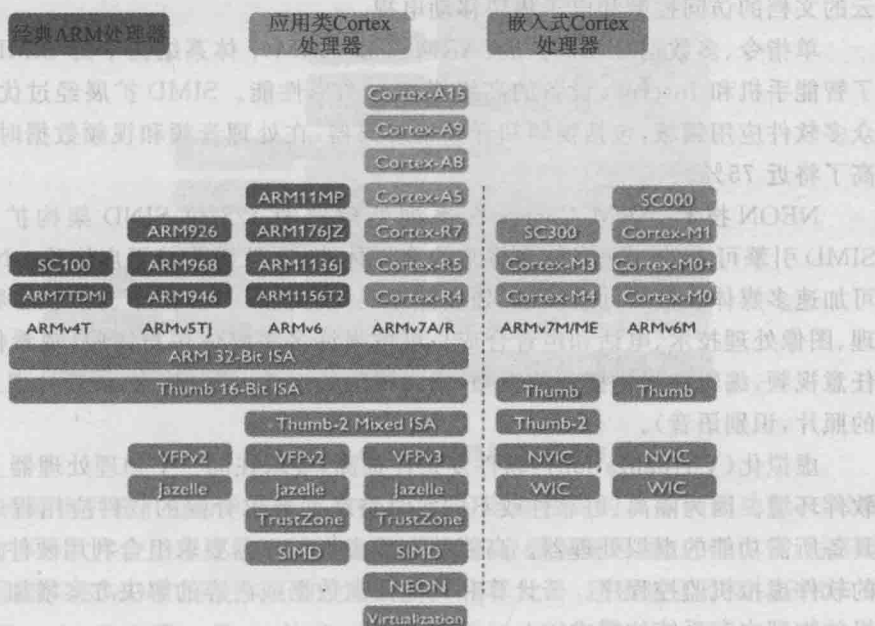


图 1-2 ARM 处理器体系结构分类

浮点体系结构 (VFP, Vector Floating Point): 为半精度、单精度和双精度浮点运算中的浮点操作提供硬件支持。为汽车动力系统、车身控制应用和图像应用 (如打印中的缩放、转换和字体生成及图形中的 3D 转换、FFT 和过滤) 中使用的浮点运算提供增强的性能。

Jazelle 技术: 提高执行环境 (如 Java、. Net、MSIL、Python 和 Perl) 运行速度的技术, 是 ARM 提供的组合型硬件和软件解决方案。ARM Jazelle 技术软件是功能丰富的多任务 Java 虚拟机 (JVM), 经过高度优化, 可利用许多 ARM 处理器内核中提供的 Jazelle 技术体系结构扩展, 还包括功能丰富的多任务虚拟机 (MVM)。领先的手机供应商和 Java 平台软件供应商提供的许多 Java 平台中均集成了此类虚拟机。通过利用基础 Jazelle 技术体系结构扩展, ARM MVM 软件解决方案可提供高性能应用程序和游戏, 快速启动和应用程序切换, 并且使用的内存和功耗预算非常低。