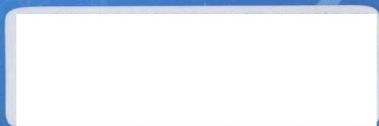


高等学校电子信息学科“十二五”规划教材

ARM Cortex

嵌入式系统开发教程

黄建华 宾辰忠 欧阳宁 编著



西安电子科技大学出版社
<http://www.xduph.com>

高等学校电子信息学科“十二五”规划教材

ARM Cortex 嵌入式系统开发教程

黄建华 宾辰忠 欧阳宁 编著

西安电子科技大学出版社

内 容 简 介

本书是嵌入式系统微控制器教材, 基于 ARM Cortex-M3 核的芯片 LPC1700 讲述了嵌入式系统基本概念以及 Cortex 体系结构和嵌入式系统开发设计方法。全书共 6 章, 主要内容包括嵌入式系统概述、ARM Cortex-M3 体系结构、LPC1700 系列处理器、LPC1700 系列处理器基本接口技术、LPC1700 系列处理器通信接口技术以及嵌入式实时操作系统。

本书可作为高等院校电子信息类、计算机类、自动控制类和机械电子类专业高年级本科生、研究生的教材, 也可作为嵌入式系统设计工程师的参考书。

图书在版编目(CIP)数据

ARM Cortex 嵌入式系统开发教程/黄建华, 宾辰忠, 欧阳宁编著. —西安: 西安电子科技大学出版社, 2012.12
高等学校电子信息类“十二五”规划教材
ISBN 978-7-5606-2903-2

I. ① A… II. ① 黄… ② 宾… ③ 欧… III. ① 微控制器—高等学校—教材 IV. ① TP332.3

中国版本图书馆 CIP 数据核字(2012)第 183751 号

策划编辑 马乐惠 毛红兵

责任编辑 毛红兵 杨 柳

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfb001@163.com

经 销 新华书店

印刷单位 陕西华沐印刷科技有限责任公司

版 次 2012 年 12 月第 1 版 2012 年 12 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 20.5

字 数 487 千字

印 数 1~3000 册

定 价 36.00 元

ISBN 978-7-5606-2903-2/TP·1368

XDUP 3195001-1

如有印装问题可调换

本社图书封面为激光防伪覆膜, 谨防盗版。

前 言

嵌入式系统是以应用为中心，以计算技术为基础，软、硬件皆可剪裁的专用计算机系统，可满足应用系统对功能、可靠性、成本、体积、功耗等严格约束的要求。当今的嵌入式系统已普遍应用于国防电子、数字家庭、工业自动化、汽车电子、医学科技、消费电子、无线通信、电力系统等国民经济的主要行业。随着嵌入式技术的发展，嵌入式系统将更广泛地应用于人类生活的方方面面，如基于嵌入式 Internet 网络的地球电子皮肤，可以嵌入到牙齿上的手机等都在研发之中。著名的嵌入式系统专家沈绪榜院士认为“计算机是认识世界的工具，而嵌入式系统则是改造世界的产物”。

面对嵌入式系统的发展形势，近年来嵌入式系统业界掀起了广泛学习嵌入式系统理论及其应用开发的热潮。各高等院校都陆续开设了相关课程，相关的出版物也在不断面市。

本书的编写是一种尝试，是以编者自身理论教学和工程实践的体会为基础，并尽可能地汲取近年来桂林电子科技大学嵌入式教学的成果和经验编写而成的。编者力图在教学内容和训练方法上有所突破。本书以综合性、设计性的实验作为训练手段，以素质教育为目标，强调培养和提高学生的动手能力，尽可能地带动学生的学习积极性。

本书共 6 章，第 1 章为嵌入式系统概述，介绍了嵌入式系统的基本概念、嵌入式处理器和常用的嵌入式操作系统；第 2 章详细讲述了 ARM Cortex-M3 内核的体系结构，包括编程模型、ARM Cortex-M3 寄存器组织、ARM Cortex-M3 存储器、ARM Cortex-M3 异常处理以及 NVIC 与中断控制等；第 3 章详细介绍了 LPC1700 系列处理器的核心结构，包括引脚配置、存储器管理、时钟和功率控制、系统控制模块和 LPC1700 系统例程等；第 4 章讲述了 LPC1700 系列处理器的接口技术，包括 GPIO 接口、定时器、看门狗、UART 串口通信、ADC/DAC、实时时钟及其它接口等；第 5 章为 LPC1700 系列处理器通信接口技术，详细介绍了 I²C 总线接口、以太网接口、SPI 接口与串口闪存，CAN 总线接口、USB 接口及 I²S 接口等；第 6 章为嵌入式实时操作系统，详细介绍了嵌入式实时操作系统基础、 $\mu\text{C}/\text{OS-II}$ 内核原理以及基于 $\mu\text{C}/\text{OS-II}$ 的嵌入式系统程序设计实例。

本书以当前流行的 ARM Cortex-M3 内核和业界著名的 NXP 公司出品的 LPC1700 系列最新处理器作为讲授对象。与本书配套的还有一套自行开发的、基于 LPC1700 系列处理器的实验平台，包括实验设备、相关软件和相应实验指导书，可以将理论教学与实践教学有机地结合起来，切实提高学生的实际动手能力，为培养、训练学生开展科研、产品研发的能力，乃至日后走上工作岗位打下较坚实的基础。

本书由桂林电子科技大学信息与通信学院的教师编写，其中黄建华编写了第 2、3 章以及第 6 章的内容，宾辰忠编写了第 4 章、第 5 章的内容，欧阳宁编写了第 1 章的内容。熊娅、林乐平、李天松对本书的内容进行了审阅。

特别感谢桂林电子科技大学郑继禹教授的支持和鼓励。本书在编写的过程中，得到了桂林电子科技大学信息与通信学院及西安电子科技大学出版社的大力支持，在这里一并表示感谢，同时还要感谢欧阳悠悠、宾一朵对本书的支持。

由于编者能力有限，书中难免存在不妥之处，恳请读者批评指正。

编者
2012年3月

目 录

第 1 章 嵌入式系统概述	1	2.4.3 向量表	29
1.1 嵌入式系统简介	1	2.5 NVIC 与中断控制	31
1.1.1 嵌入式系统的定义	1	2.5.1 NVIC 概述	31
1.1.2 嵌入式系统的组成	2	2.5.2 中断配置基础	31
1.1.3 嵌入式系统的特点	3	2.5.3 中断的使能与禁止	31
1.2 嵌入式处理器	3	2.5.4 中断的挂起与解挂	32
1.2.1 嵌入式处理器的分类	4	2.5.5 中断建立全过程	33
1.2.2 ARM 处理器	6	2.5.6 中断/异常的响应序列	33
1.3 嵌入式操作系统	11	2.5.7 异常返回	34
1.3.1 操作系统的概念	11	2.5.8 SysTick 定时器	34
1.3.2 嵌入式 Linux	12	习题	37
1.3.3 μ C/OS-II	12	第 3 章 LPC1700 系列处理器	38
1.3.4 Windows CE	12	3.1 LPC1700 系列处理器简介	38
1.3.5 VxWorks	13	3.1.1 LPC1700 系列处理器特性	38
习题	13	3.1.2 LPC1700 系列处理器结构	40
第 2 章 ARM Cortex-M3 体系结构	14	3.2 处理器引脚配置	42
2.1 ARM Cortex-M3 编程模型	14	3.2.1 引脚配置	42
2.1.1 ARM Cortex-M3 处理器的 编程模型	14	3.2.2 引脚连接模块	52
2.1.2 Cortex-M3 处理器的工作状态和 工作模式	16	3.2.3 引脚连接模块的使用举例	54
2.2 ARM Cortex-M3 寄存器组织	17	3.3 存储器管理	55
2.2.1 通用寄存器	18	3.4 时钟和功率控制	58
2.2.2 程序状态寄存器	19	3.4.1 晶体振荡器	58
2.2.3 控制寄存器	20	3.4.2 PLL0 锁相环	60
2.2.4 中断屏蔽寄存器	21	3.4.3 时钟分频	65
2.3 ARM Cortex-M3 存储器	21	3.4.4 功率控制	68
2.3.1 Cortex-M3 存储器格式	21	3.4.5 外部时钟输出引脚	72
2.3.2 Cortex-M3 存储器映射	22	3.5 系统控制模块	74
2.3.3 存储器访问属性	24	3.5.1 复位	74
2.3.4 位带(bit-band)操作	25	3.5.2 掉电检测	75
2.4 ARM Cortex-M3 异常处理	27	3.5.3 外部中断	76
2.4.1 异常类型	27	3.5.4 系统控制和状态标志	79
2.4.2 异常优先级	29	3.6 LPC1700 系统例程	79
		3.6.1 CMSIS 的系统启动代码	79
		3.6.2 外部中断例程	88

3.6.3 SysTick 定时器例程.....	89	4.5.6 ADC 引脚描述.....	139
习题.....	91	4.5.7 ADC 寄存器描述.....	140
第 4 章 LPC1700 系列处理器基本		4.5.8 ADC 基本操作.....	144
接口技术	92	4.5.9 应用举例.....	144
4.1 GPIO 接口.....	92	4.6 实时时钟.....	145
4.1.1 特性.....	92	4.6.1 功能描述.....	145
4.1.2 应用场合.....	93	4.6.2 结构及引脚.....	146
4.1.3 引脚描述.....	93	4.6.3 寄存器功能描述.....	147
4.1.4 寄存器描述.....	93	4.6.4 RTC 使用注意事项.....	154
4.1.5 使用注意事项.....	103	4.6.5 应用举例.....	154
4.1.6 应用举例.....	104	4.7 其它接口.....	156
4.2 定时器.....	106	4.7.1 GPDMA 控制器.....	156
4.2.1 特性.....	106	4.7.2 PWM 接口.....	164
4.2.2 应用场合.....	107	4.7.3 QEI 接口.....	169
4.2.3 定时器结构.....	107	习题.....	172
4.2.4 引脚功能描述.....	108	第 5 章 LPC1700 系列处理器	
4.2.5 寄存器功能描述.....	108	通信接口技术	173
4.2.6 应用举例.....	114	5.1 I ² C 总线接口.....	173
4.2.7 重复中断定时器(RIT)概述.....	117	5.1.1 I ² C 接口特性.....	173
4.2.8 RIT 寄存器描述.....	117	5.1.2 I ² C 总线引脚及应用.....	173
4.2.9 RIT 操作.....	119	5.1.3 I ² C 总线基本原理.....	174
4.3 看门狗.....	119	5.1.4 I ² C 操作模式.....	175
4.3.1 功能描述.....	119	5.1.5 I ² C 接口寄存器描述.....	176
4.3.2 看门狗结构.....	120	5.1.6 应用举例.....	186
4.3.3 寄存器功能描述.....	120	5.2 以太网接口.....	190
4.3.4 操作举例.....	123	5.2.1 以太网接口概述.....	190
4.4 UART 串口通信.....	124	5.2.2 以太网接口特性.....	190
4.4.1 概述.....	124	5.2.3 以太网接口结构及引脚描述.....	191
4.4.2 UART 结构.....	125	5.2.4 以太网接口操作概述.....	192
4.4.3 寄存器功能描述.....	126	5.2.5 帧描述符与状态字.....	194
4.4.4 基本操作.....	134	5.2.6 以太网帧操作举例.....	201
4.4.5 应用举例.....	135	5.2.7 寄存器描述.....	204
4.5 ADC/DAC.....	137	5.2.8 以太网接口驱动程序举例.....	223
4.5.1 LPC1700 DAC 特性.....	137	5.3 SPI 接口与串口闪存.....	232
4.5.2 DAC 引脚描述.....	137	5.3.1 SPI 接口概述.....	232
4.5.3 DAC 寄存器描述.....	137	5.3.2 SPI 接口引脚.....	232
4.5.4 DAC 基本操作.....	139	5.3.3 SPI 接口寄存器描述.....	233
4.5.5 LPC1700 ADC 特性.....	139	5.3.4 SPI 接口结构框图.....	237

5.3.5	SPI 接口操作.....	238
5.3.6	串口闪存操作举例	239
5.4	CAN 总线接口.....	245
5.4.1	CAN 总线接口概述	245
5.4.2	CAN 模块内存映射表	247
5.4.3	CAN 控制器寄存器描述	247
5.4.4	CAN 控制器操作	249
5.5	USB 接口	250
5.5.1	USB 总线概述.....	250
5.5.2	USB 设备接口结构描述.....	251
5.5.3	固定的端点配置	252
5.5.4	USB 设备接口操作概述.....	253
5.5.5	USB 设备接口寄存器描述.....	254
5.5.6	USB 设备控制器的初始化.....	255
5.5.7	串行接口引擎命令描述	256
5.6	I ² S 接口	258
5.6.1	I ² S 接口概述.....	258

5.6.2	引脚描述	259
5.6.3	I ² S 接口寄存器描述.....	260
	习题	261

第 6 章	嵌入式实时操作系统.....	262
6.1	嵌入式实时操作系统基础	262
6.1.1	嵌入式实时操作系统简介	262
6.1.2	嵌入式实时操作系统基本概念	263
6.2	μC/OS-II 内核原理.....	267
6.2.1	μC/OS-II 任务管理	269
6.2.2	μC/OS-II 的 API 函数.....	280
6.2.3	μC/OS-II 的文件结构和移植	294
6.3	基于 μC/OS-II 的嵌入式系统 程序设计实例	308
	习题	319
	参考文献.....	320

第1章 嵌入式系统概述

1.1 嵌入式系统简介

1.1.1 嵌入式系统的定义

经过30多年的发展,嵌入式系统已经广泛应用于科学研究、工程设计、军事技术、各类产业、商业文化艺术、娱乐业及人们的日常生活等方面,从厨房中的电饭煲、微波炉、电冰箱到客厅里的家庭媒体中心,各种智能化设备已经遍布在我们的周围。随着数字信息技术和网络技术的飞速发展,计算机、通信、消费电子的一体化趋势日益明显,这必将培养出一个庞大的嵌入式应用市场。现在,嵌入式系统带来的工业年产值已超过了1万亿美元,它已经成为信息技术(IT)产业争夺的重点。嵌入式系统技术也成为当前被关注、学习和研究的热点。

所谓嵌入式系统(Embedded System),实际上是“嵌入式计算机系统”的简称,它是相对于通用计算机而言的。有些系统中也有计算机,但它是作为某个专用系统中的一个部件而存在的。像这样嵌入到更大、更专用的系统中的计算机系统,就称为嵌入式计算机、嵌入式计算机系统或嵌入式系统。

日常生活中早已存在许多嵌入式系统的应用,如人们使用的移动电话、烹调用微波炉、戴在手腕上的电子表、家用的洗衣机、办公室里的打印机,以及数码相机、MP3、平板电脑等手持式设备,都可以认为是嵌入式系统。

按照美国电气和电子工程师协会(IEEE)的定义,嵌入式系统是“用于监视、控制或者辅助操作机器和设备的装置”。这个定义是从应用上考虑的,可以看出,嵌入式系统是软件和硬件的综合体,还可以涵盖机电等附属装置。

目前我们最常见、最通用的一个定义是:嵌入式系统是以应用为中心,以计算机技术为基础,其软、硬件可配置,对功能、可靠性、成本、体积、功耗有严格约束的一种专用计算机系统。这个定义是从技术角度来进行的。它不仅指明了嵌入式系统是一种专用的计算机系统(非PC的智能电子设备),而且说明了嵌入式系统的几个基本要素,即面向应用,以计算机技术为基础,软、硬件可剪裁以及在功能、可靠性、成本、体积和功耗上有严格约束。而“嵌入式系统”中“嵌入”一词,即指其软、硬件可剪裁的特性,它表示该系统通常是更大系统中的一个完整的部分。嵌入的系统中共存多个嵌入式系统。

嵌入式系统几乎应用于所有电器设备中,如手机、机顶盒、个人数字助理(PDA)、汽车控制系统、微波炉控制器、电梯控制器、安全系统、医疗仪器、立体音响、自动售货机

控制器、自动取款机等。即使是一台通用的 PC，也包括嵌入式系统，其外部设备都包含了嵌入式微处理器的成分，如硬盘、软驱、显示器、键盘、鼠标、声卡、网卡、Modem 和打印机、扫描仪等都是由嵌入式处理器控制的。

嵌入式系统是面向用户、产品和应用的，如果独立于应用而自行发展，则会失去市场。因此，大多数嵌入式系统的开发者并不是计算机专业的人员，而是各个行业的技术人员，例如数字医疗设备的开发，往往是由生物医学工程技术人员和计算机专业的技术人员一起参与完成的。

嵌入式系统是一种专用的计算机系统，它和通用计算机系统使用的技术是一样的，都包含了硬件部分和软件部分，但对二者的性能评价指标是不同的。嵌入式系统往往只是一个大系统中的组成部分，控制大系统的工作，它的价值在于其所控制的大系统。例如，智能洗衣机的评价指标往往是洗净度、耗水量、耗电量、洗衣速度等，而不是控制它的处理器的速度、存储容量等。而通用计算机不同，其更关注计算能力、处理速度及存储数据的能力等指标。

1.1.2 嵌入式系统的组成

嵌入式系统基于计算机技术，其组成也跟计算机组成类似，主要包括两个部分：嵌入式硬件系统和嵌入式软件系统。

(1) 嵌入式硬件系统主要包括嵌入式处理器、存储器、模拟电路、电源、接口控制器及接插件等几部分。

嵌入式处理器是嵌入式系统的核心部件。嵌入式处理器与通用处理器的最大区别在于嵌入式 CPU 大多工作在为特定用户群设计的系统中。它通常把通用 CPU 中许多由板卡完成的任务集成在芯片内部，从而有利于嵌入式系统设计趋于小型化。嵌入式处理器具有高效率、高可靠性等特征。

嵌入式系统中常用的存储器有静态易失型存储器(RAM、SRAM)、动态存储器(DRAM、SDRAM)、非易失型存储器(ROM、EPROM、EEPROM、Flash)。其中，Flash(闪存)具有可擦写次数多、存储速度快、容量大以及价格低等优点，在嵌入式领域得到了广泛的应用。

目前针对嵌入式系统的外围硬件设备扩展很多，常用的有串口、以太网接口(网络设备)、USB 接口(USB 设备，如优盘、数码相机、移动硬盘等外部存储设备)、音频接口(如 MP3)、液晶显示屏(如数码相机、数码摄像机、MP4 播放器、PDA 等)、摄像头(拍照手机)等。可以看到，不同的嵌入式系统的设计可能会用到不同的外围硬件设备。以数码相机为例，它需要使用到摄像头、液晶显示屏、USB 接口、SD 或 MS 卡，如果该数码相机有 MP3 功能，就还需要加上音频解码设备。没有外围设备的支持，嵌入式系统是不完整的。

(2) 嵌入式软件系统主要包括低层驱动、操作系统软件(嵌入式操作系统)和应用程序(应用软件)三个部分。低层驱动实现嵌入式系统硬件和软件之间的接口；操作系统实现系统的进程调度、任务处理；应用程序实现系统功能的应用。由于嵌入式系统的应用领域十分广泛，应用程序(应用软件)的表现形式也千差万别。有时设计人员会把操作系统和应用软件两部分组合在一起，应用软件控制系统的运作和行为，操作系统控制应用程序编程与硬件的交互。

在嵌入式系统的组成中，嵌入式系统的核心是嵌入式处理器。因此嵌入式处理器的技

术指标如功耗、体积、成本、可靠性、速度、处理能力、电磁兼容性等均受到应用要求的制约, 这些也是半导体厂商之间竞争的热点。嵌入式处理器的应用软件是实现嵌入式系统功能的关键。一般来说, 软件需要固化存储, 有时称为固件(Firmware)。

1.1.3 嵌入式系统的特点

与常见的通用计算机系统相比, 嵌入式系统一般具有以下特点:

(1) 面向特定的应用。与通用 CPU 相比, 嵌入式 CPU 是为特定用户群设计的。如 ARM 系列多用于手机中, Motorola 公司的 PowerPC 系统多用于网络服务器、工作站中。应用需求决定了嵌入式系统的设计。决定嵌入式 CPU 应用环境的主要因素在于其提供的接口功能和处理速度。

(2) 专用性强, 可根据需要灵活定制。嵌入式系统的个性化很强, 其中软件系统和硬件的结合非常紧密, 一般要针对硬件进行系统的移植。

(3) 系统内核小。嵌入式系统一般应用于小型电子装置中, 系统资源相对有限, 所以内核较传统的操作系统要小得多。例如, $\mu\text{C}/\text{OS}$ 系统的内核只有 5 KB, 而 Windows 的内核则要大得多。

(4) 体积小、功耗低、成本低、效率高。由于嵌入式系统集成度高、体积小, 所以对其系统软件和应用软件一般有一些特殊要求, 如软件固化在 ROM 中, 要求具有高质量高可靠性的软件代码, 并具有实时处理能力等。同时, 由于嵌入式系统往往没有充足的电能(如电池供电), 所以多为低功耗系统。系统功耗越低, 温度越低, 其可靠性和稳定性也就越高。

(5) 具有较长的生命周期。嵌入式系统与具体应用有机结合在一起, 其升级和具体产品同步进行。

(6) 通常有实时性要求, 因此一般都要求有高实时性操作系统(Real-Time Operating System, RTOS)。这是嵌入式软件的基本要求, 用以实现任务调度、资源分配等功能。按照实时性的不同, 嵌入式系统可以分为软实时系统和硬实时系统。软实时系统对实时要求不高, 通常用于人机交互较多的领域, 而硬实时系统主要应用于工控、航天、军事等领域。

(7) 需要专门的软、硬件开发工具和环境。由于嵌入式系统的运行平台与开发平台是不同的, 嵌入式系统本身不具备自主开发能力, 需要专门的软、硬件开发工具和环境, 因此开发较为困难。通常嵌入式系统的开发采用交叉开发环境: 开发平台称为宿主机, 有丰富的软、硬件资源; 运行嵌入式软件的平台称为目标机, 资源相对有限。用户在宿主主机上进行软件的编辑、编译, 然后下载到目标机上调试、运行。

1.2 嵌入式处理器

嵌入式系统的核心部件是嵌入式处理器, 据不完全统计, 截至 2000 年全球嵌入式处理器的品种总量已经超过 1000 种, 流行的体系结构有 30 多个系列, 其中 8051 体系的占了多半。生产 8051 单片机的半导体厂家有 20 多个, 共 350 多种衍生产品, 仅 NXP 就有近百种。现在几乎每个半导体制造商都生产嵌入式处理器, 而且越来越多的公司拥有自己的处理器设计部门。嵌入式处理器的寻址空间一般从 64 KB 到 4 GB, 处理速度从 0.1 MIPS 到 2000 MIPS,

常用封装从 8 个引脚到 208 个引脚。

1.2.1 嵌入式处理器的分类

从应用的角度来划分，嵌入式处理器包含了下面几种类型。

1. 嵌入式微控制器

嵌入式微控制器(Microcontroller Unit, MCU)又称单片机，它将整个计算机系统集成到一块芯片中。MCU 一般以某一种微处理器内核为核心，芯片内部集成 ROM、RAM、总线逻辑、定时器等各种必要的功能模块。与微处理器相比，微控制器的最大特点是单片化，体积大大缩小，使功耗和成本下降，可靠性提高。

MCU 是目前嵌入式系统应用的主流。由于 MCU 的片上资源一般比较丰富，适于控制，因此称微控制器。为适应不同的应用需求，一般一个系列的单片机具有多种衍生产品，每种衍生产品的处理器内核都是一样的，不同的是存储器 and 外设的配置及封装。这样可以最大限度地与应用需求相匹配，从而减小功耗和成本。

目前 MCU 的品种和数量繁多，比较有代表性的通用系列包括 8051、P51XA、MCS-251、MCS-96/196/296、C166/167、MC68HC05/11/12/16、68300 等。另外，还有许多半通用系列，如支持 USB 接口的 MCU 8XC930/931、C540、C541 等。

2. 嵌入式微处理器

嵌入式微处理器(Microprocessor Unit, MPU)的基础是通用计算机中的 CPU。为了满足嵌入式应用的特殊要求，MPU 虽然在功能上和标准微处理器基本一样，但在工作温度、抗电磁干扰、可靠性等方面一般都做了各种增强。

目前 MPU 主要有 Aml86/88、386EX、SC-400、PowerPC、68000、MIPS、ARM 系列等。

MPU 可分为复杂指令集计算机 CISC 和精减指令集计算机 RISC 两类。大多数台式计算机都使用 CISC 微处理器，如 Intel 的 X86。RISC 结构体系有两大主流：Silicon Graphics 公司(硅谷图形公司)的 MIPS 技术及 ARM 公司的 Advanced RISC Machines 技术。

RISC 和 CISC 是目前设计制造微处理器的两种典型技术，为达到高效的目的，采用的方法不同。它们的差异主要有以下几点：

(1) 指令系统。RISC 设计者把主要精力放在那些经常使用的指令上，对不常用的功能，常通过组合指令来实现；CISC 计算机的指令系统比较丰富，有专用指令来完成特定的功能。

(2) 存储器操作。RISC 对存储器操作有限制，使控制简单化；CISC 机器的存储器操作指令多，操作直接。

(3) 程序。RISC 汇编语言程序一般需要较大的内存空间，实现特殊功能时程序复杂，不易设计；CISC 汇编语言程序编程相对简单，科学计算及复杂操作的程序设计相对容易，效率较高。

(4) 中断。RISC 机器在一条指令执行的适当地方可以响应中断；CISC 机器是在一条指令执行结束后响应中断的。

(5) CPU。RISC CPU 包含较少的单元电路，面积小、功耗低；CISC CPU 包含丰富的电路单元，功能强、面积大、功耗高。

(6) 设计周期。RISC 微处理器结构简单，布局紧凑，设计周期短，且易于采用最新技术；CISC 微处理器结构复杂，设计周期长。

(7) 易用性。RISC 微处理器结构简单，指令规整，性能容易把握，易学易用；CISC 微处理器结构复杂，功能强大，实现特殊功能容易。

(8) 应用范围。RISC 机器更适合于嵌入式应用，CISC 机器则更适合于通用计算机。

3. 嵌入式 DSP

嵌入式 DSP(Digital Signal Processor)对系统结构和指令进行了特殊设计，使其适合于执行 DSP 算法，编译效率较高，指令执行速度也较高。在数字滤波、FFT、谱分析等方面，DSP 算法正在大量进入嵌入式领域。

推动嵌入式 DSP 发展的一个重要因素是嵌入式系统的智能化。例如，各种带有智能逻辑的消费类产品、生物信息识别终端、带有加/解密算法的键盘、ADSL 接入、实时语音压缩解压系统、虚拟现实显示等。这类智能化算法一般运算量都比较大，特别是向量运算、指针线性寻址等较多，而这些正是 DSP 的长处所在。

嵌入式 DSP 有两个发展来源，一是 DSP 经过单片化、EMC 改造、增加片上外设成为嵌入式 DSP，例如 TI 的 TMS320C2000/C5000；二是在通用单片机或片上系统(SoC)中增加 DSP 协处理器，例如 Intel 的 MCS-296。

嵌入式 DSP 比较有代表性的产品是 Texas Instruments 的 TMS320 系列和 Motorola 的 DSP56000 系列。TMS320 系列处理器包括用于控制的 C2000 系列、用于移动通信的 C5000 系列以及性能更高的 C6000 和 C8000 系列。DSP 的设计者们把重点放在了处理连续的数据流上。如果嵌入式应用中强调对连续数据流的处理及高精度复杂运算，则应该选用 DSP 器件。

4. 嵌入式片上系统

随着 VLSI 设计的普及和半导体工艺的迅速发展，可以在一块硅片上实现一个更为复杂的系统，这就是 SoC(System on-Chip)。各种通用处理器内核和其它外围设备都将成为 SoC 设计公司标准库中的器件，可用标准的 VHDL 等硬件描述语言描述。用户只需定义出整个应用系统，仿真通过后就可以将设计图交给半导体工厂制作芯片样品。这样，整个嵌入式系统大部分都可以集成到一块芯片中，应用系统的电路板将变得很简洁，这将有利于减小体积和功耗，从而提高系统的可靠性。

嵌入式 SoC 的设计关键是 IP 核的设计。IP 核分为硬核、软核和固核。IP 核设计是嵌入式技术的重要支持技术。在设计嵌入式系统时，可以通过使用 IP 核技术完成系统硬件的设计。在 IP 技术中把不同功能的电路模块称为 IP，这些 IP 都是经过实际制作并证明其是正确的。在 EDA 设计工具中把这些 IP 组织在一个 IP 元件库中，供用户使用。设计电子系统时，用户需要知道 IP 模块的功能和技术性能。把不同的 IP 模块嵌入在一个硅片上，可以形成完整的应用系统。IP 技术极大地简化了 SoC 的设计过程，缩短了设计时间，因此已经成为目前电子系统设计的重要的基本技术。

SoC 可以分为通用系列和专用系列两类。通用系列包括 Motorola 的 M-Core、某些 ARM 系列器件、Echelon 和 Motorola 联合研制的 Neuron 芯片等。专用 SoC 一般专用于某类系统中，不为一般用户所知。

1.2.2 ARM 处理器

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,还可以认为是一种技术的名字。1991年ARM公司成立于英国剑桥,主要出售芯片设计技术的授权。目前,采用ARM技术知识产权(IP)核的微处理器,即通常所说的ARM微处理器,已广泛应用于工业控制、消费类电子产品、通信系统、网络系统、无线系统等各个领域。

1. ARM 微处理器的应用领域

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市场份额,而且也逐渐向低端微控制器应用领域扩展,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,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进行了优化,并获得广泛支持,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。

2. ARM 微处理器的特点

ARM微处理器有如下特点:

- (1) 体积小、功耗低、成本低、性能高;
- (2) 支持Thumb(16位)/ARM(32位)双指令集,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。

3. ARM 微处理器系列

ARM微处理器的体系结构从最初开发到目前已有了很大的改进,并仍在完善和发展中。为了清楚地表达每个ARM应用实例所使用的指令集,ARM公司定义了7种主要的ARM指令集体系结构版本,以版本号V1~V7表示。

(1) V1版架构只在原型机ARM1中出现过,只有26位的寻址空间,没有用于商业产品。其寻址空间为64MB。

(2) V2 版架构对 V1 版进行了扩展, 例如 ARM2 和 ARM3(V2a)架构。V2 包含了对 32 位乘法指令和协处理器指令的支持。版本 V2a 是版本 V2 的变种, ARM3 芯片采用了版本 V2a, 是第一个采用片上 Cache 的 ARM 处理器。

(3) V3 版架构是 ARM 公司于 1990 年设计的第一个微处理器 ARM6。它是一款 IP 核独立的处理器, 具有片上高速缓存、MMU 和写缓冲等功能, 其变种版本有 V3G 和 V3M。它的寻址空间增至 32 位(4 GB), 当前程序状态信息从原来的 R15 寄存器移到当前程序状态寄存器 CPSR 中(Current Program Status Register)。V3 版架构增加了程序状态保存寄存器 SPSR(Saved Program Status Register)和两种异常模式, 使操作系统代码可方便地使用数据访问中止异常、指令预取中止异常和未定义指令异常。

(4) V4 版架构在 V3 版上作了进一步扩充, 它是目前应用最广的 ARM 体系结构, ARM7、ARM8、ARM9 和 StrongARM 都采用该架构。V4 不再强制要求与 26 位地址空间兼容, 而且还明确了哪些指令会引起未定义指令异常。其指令集增加了 T 变种, 处理器可工作在 Thumb 状态, 增加了 16 位 Thumb 指令集, 完善了软件中断 SWI 指令的功能。V4 版架构处理器系统模式引进特权方式时使用用户寄存器操作, 把一些未使用的指令空间捕获为未定义指令。

(5) V5 版架构在 V4 版基础上增加了一些新的指令, ARM10 和 XScale 都采用该版架构。其新增的指令有带有链接和交换的转移 BLX 指令、计数前导零 CLZ 指令、BRK 中断指令和数字信号处理指令(V5TE 版)。V5 还为协处理器增加了更多可选择的指令, 改进了 ARM/Thumb 状态之间的切换效率, 它的指令增加了 E 变种(增强型 DSP 指令集, 包括全部算法操作和 16 位乘法操作)和 J 变种(支持新的 JAVA, 提供字节代码执行的硬件和优化软件加速功能)。

(6) V6 版架构是 2001 年发布的, 并在 2002 年春季发布的 ARM11 处理器中使用。V6 在降低耗电量的同时, 还强化了图形处理性能。通过追加有效进行多媒体处理的 SIMD(Single Instruction, Multiple Data, 单指令多数据)功能, 将语音及图像的处理功能提高到了原型机的 4 倍。

(7) V7 架构是在 V6 架构的基础上诞生的。该架构采用了 Thumb-2 技术, 它是在 ARM 的 Thumb 代码压缩技术的基础上发展起来的, 并且保持了对现存 ARM 解决方案的完整的代码兼容性。Thumb-2 技术比纯 32 位代码少使用 31%的内存, 减小了系统开销, 同时能够提供比已有的基于 Thumb 技术的解决方案高出 38%的性能。V7 架构还采用了 NEON 技术, 将 DSP 和媒体处理能力提高了近 4 倍, 并支持改良的浮点运算, 满足下一代 3D 图形、游戏物理应用以及传统嵌入式控制应用的需求。此外, V7 架构还支持改良的运行环境, 以迎合不断增加的 JIT(Just In Time)和 DAC(Dynamic Adaptive Compilation)技术的使用。

目前市面上常用的 ARM 处理器有 ARM7 系列、ARM9 系列、ARM9E 系列、ARM10 系列、ARM11 系列、Cortex 系列和 SecurCore 系列, 分别用于开发不同的产品。如 ARM7 系列适用于工业控制、网络设备、移动电话等应用; ARM9、ARM9E 和 ARM10E 系列则更适合无线设备、消费类电子产品的设计; SecurCore 系列专门为安全要求较高的应用而设计。

常见的基于 ARM 核的产品有 Intel 公司的 XScale 系列, ST 公司的 STM32 系列, Freescale 公司龙珠系列 iMX 处理器, TI 公司 DSP+ARM 处理器 OMAP 和 Cortex 核的 LM3S 系列,

Cirrus Logic 公司的 ARM 系列, SamSung 公司的 ARM 系列, Atmel 公司的 AT91 系列微控制器, NXP 公司的微控制器系列。

4. ARM Cortex 处理器简介

ARM 公司在 ARM11 系列以后的产品改用 Cortex 命名, 并分成 A、R 和 M 三类, 旨在为各种不同的市场提供服务。

Cortex 系列属于 V7 架构, 它是 ARM 公司最新的指令集架构。V7 架构定义了三大分工明确的系列: A 系列面向尖端的基于虚拟内存的操作系统和用户应用, R 系列针对实时系统, M 系列针对微控制器。由于应用领域不同, 基于 V7 架构的 Cortex 处理器系列所采用的技术也不相同, 基于 V7A 的称为 Cortex-A 系列, 基于 V7R 的称为 Cortex-R 系列, 基于 V7M 的称为 Cortex-M 系列。

1) Cortex-A 处理器

Cortex-A 处理器适用于具有高计算要求、运行丰富操作系统以及提供交互媒体和图形体验的应用领域。从最新技术的移动 Internet 必备设备(如手机和超便携的上网笔记本或智能笔记本)到汽车信息娱乐系统和下一代数字电视系统, ARM Cortex-A 处理器可向托管丰富的操作系统平台的设备和用户应用提供全方位的解决方案, 包括超低成本的手机、智能手机、移动计算平台、数字电视、机顶盒、企业网络、打印机和服务器解决方案。高性能的 Cortex-A15、可伸缩的 Cortex-A9、经过市场验证的 Cortex-A8 处理器和高效的 Cortex-A5 处理器均共享同一体系结构, 因此具有完整的应用兼容性, 支持传统的 ARM、Thumb® 指令集和新增的高性能紧凑型 Thumb-2 指令集。

Cortex-A 处理器的应用非常广泛, 具体包括上网笔记本、智能笔记本、输入板、电子书阅读器、智能手机、特色手机、数字家电、机顶盒、数字电视、蓝光播放器、游戏控制台、导航、激光打印机、路由器、无线基站、VOIP 电话和设备、Web 2.0、无线基站、交换机、服务器等。Cortex-A 处理器的应用示例如图 1.1 所示。

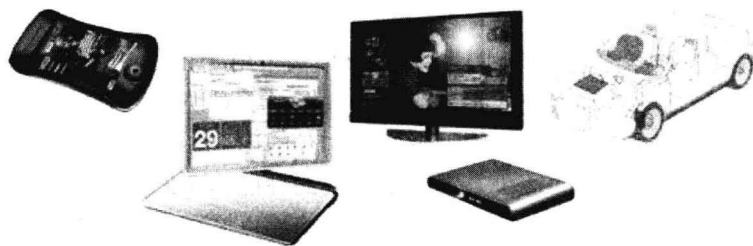


图 1.1 Cortex-A 处理器的应用示例

目前比较常见的 Cortex-A 处理器型号有 Cortex-A8、Cortex-A9 等。

2) Cortex-R 处理器

Cortex-R 处理器为具有严格的实时响应限制的深层嵌入式系统提供高性能计算解决方案, 为范围广泛的深层嵌入式半导体应用市场设置了行业标准, 提供了大约 20 个许可接收方、100 个设计和数百万的设备。

(1) Cortex-R 处理器的目标应用。Cortex-R 处理器的目标应用包括智能手机和基带调制解调器中的移动手机处理、企业系统、硬盘驱动器、联网和打印、家庭消费性电子产品、

机顶盒、数字电视、媒体播放器和数码相机以及应用于医疗行业、工业和汽车行业的可靠系统的嵌入式微控制器。

在这些应用中,采用的是对处理响应设置硬截止时间的系统,如果要避免数据丢失或机械损伤,则必须符合所设置的这些硬截止时间。因此 Cortex-R 处理器是专为高性能、高可靠性和容错能力而设计的,其行为具有高确定性,同时保持很高的能效和成本效益。

(2) Cortex-R 处理器的主要功能。

- ① 快速:以高时钟频率获得高处理性能;
- ② 确定性:处理在所有场合都必须符合硬实时限制;
- ③ 安全:系统必须可靠且可信(某些系统须是安全关键系统);
- ④ 成本效益:在处理器及其内存系统中都具有竞争力的成本和功耗。

此功能集将 Cortex-R、Cortex-M 和 Cortex-A 系列处理器区别开来。显而易见,Cortex-R 提供的性能比 Cortex-M 系列提供的性能高得多,而 Cortex-A 专用于具有复杂软件操作系统(使用虚拟内存管理)的面向用户的应用。

Cortex-R 处理器保持了与经典 ARM 处理器(如 ARM7TDMI-S、ARM946E-S、ARM968E-S 和 ARM1156T2-S)的二进制兼容性,因此可确保应用的可移植性。它对于经认证可用于汽车系统的代码很有用,当旧源代码不再可用时也非常有用。这些嵌入式系统处理器通常运行实时软件操作系统(RTOS)并且不需要虚拟内存管理单元(MMU)。但是,实时 ARM 处理器支持内存保护单元(MPU)和紧密耦合内存(TCM),它们使代码和数据随时可供处理器访问。

Cortex-R 系列处理器常用的有 Cortex-R4、Cortex-R5 和 Cortex-R7 等,主要满足深层嵌入式和实时市场(如汽车安全或无线基带)所要求的主要功能。Cortex-R 处理器的应用示例如图 1.2 所示。

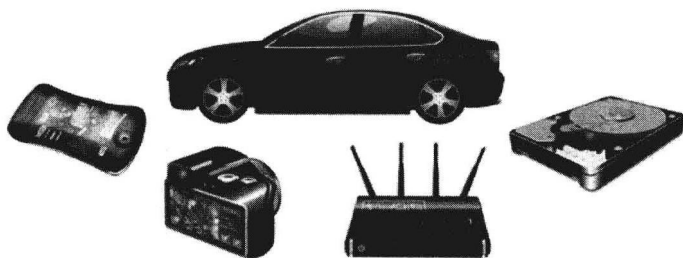


图 1.2 Cortex-R 处理器的应用示例

3) Cortex-M 处理器

Cortex-M 处理器是全球微控制器标准,已许可给 40 个以上的 ARM 合作伙伴,包括 NXP Semiconductors、STMicroelectronics、Texas Instruments 和 Toshiba 等领先供应商。使用标准处理器可以使 ARM 合作伙伴创建具有一致体系结构的设备,同时使它们可以专注于创建优秀的设备实现。

Cortex-M 处理器是一系列可向上兼容的高能效、易于使用的处理器,这些处理器旨在帮助开发人员满足将来的嵌入式应用的需要。这些需要包括以更低的成本提供更多功能、不断增加连接、改善代码重用和提高能效。Cortex-M 系列针对成本和功耗敏感的 MCU 和