



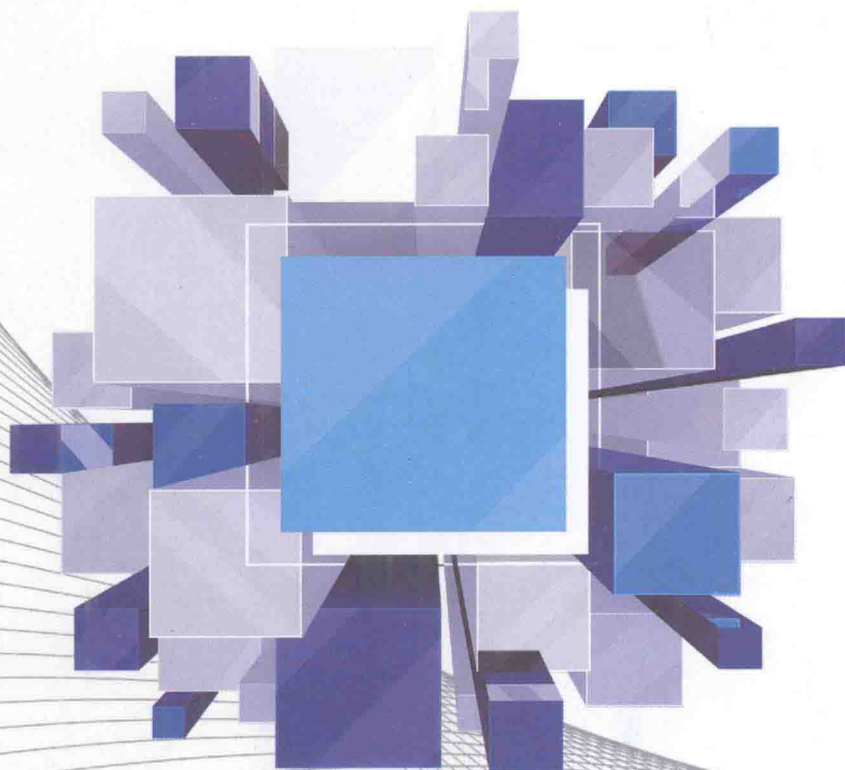
中国电子学会物联网专家委员会推荐

普通高等教育物联网工程专业“十二五”规划教材

嵌入式Linux开发技术及实践

Embedded Linux Technology Development and Practice

青岛东合信息技术有限公司 编著



西安电子科技大学出版社
<http://www.xduph.com>

中国电子学会物联网专家委员会推荐

普通高等教育物联网工程专业“十二五”规划教材

嵌入式 Linux 开发技术及实践

青岛东合信息技术有限公司 编著

西安电子科技大学出版社

内 容 简 介

本书从嵌入式系统的基本概念出发,以 ARM9 系列处理器 S3C2440 为基础,配合开发板硬件平台,深入浅出地讲解了基于 ARM 的嵌入式 Linux 开发的各个环节。

本书分为两篇——理论篇和实践篇。理论篇介绍了嵌入式系统的基本概念和组成、ARM 基础开发、ARM 进阶开发、系统构建、驱动移植、应用编程以及 GUI 程序设计。实践篇与理论篇相对应,以如何实现一个物联网网关为案例,将理论与实践相结合,使读者加深对嵌入式 Linux 开发技术的理解并掌握基于 ARM 的嵌入式 Linux 开发知识,全面提高动手能力。

本书重点突出,偏重应用,适用面广,可作为本科计算机科学与技术、通信电子,高职高专计算机软件、计算机网络专业的嵌入式 Linux 课程的教材。

图书在版编目(CIP)数据

嵌入式 Linux 开发技术及实践/青岛东合信息技术有限公司编著. —西安:西安电子科技大学出版社,2014.1
普通高等教育物联网工程专业“十二五”规划教材

ISBN 978-7-5606-3313-8

I. ① 嵌… II. ① 青… III. ① Linux 操作系统—程序设计—高等学校—教材 IV. ① TP316.89

中国版本图书馆 CIP 数据核字(2014)第 001537 号



策 划 毛红兵

责任编辑 毛红兵 张俊利

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 2014 年 1 月第 1 版 2014 年 1 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 30

字 数 715 千字

印 数 1~3000 册

定 价 68.00 元

ISBN 978-7-5606-3313-8/TP

XDUP 3605001-1

如有印装问题可调换

普通高等教育物联网工程专业 “十二五”规划教材编委会

主 编： 韩敬海

副主编： 倪建成

编 委： 崔文善 王成端 薛庆文

孔繁之 吴明君 李洪杰

刘继才 吴海峰 张 磊

孔祥和 陈龙猛 窦相华

王海峰 张 伟 王 蕊

前 言

随着物联网产业的迅猛发展，企业对物联网工程应用型人才的需求越来越大。“全面贴近企业需求，无缝打造专业实用人才”是目前高校物联网专业教育的革新方向。

本系列教材是面向高等院校物联网专业方向的标准化教材，教材内容注重理论且突出实践，强调理论讲解和实践应用的结合，覆盖了物联网的感知识别、网络通信及应用支撑等物联网架构所包含的关键技术。教材研发充分结合物联网企业的用人需求，经过了广泛的调研和论证，并参照多所高校一线专家的意见，具有系统性、实用性等特点，旨在使读者在系统掌握物联网开发知识的同时，具备综合应用能力和解决问题的能力。

该系列教材具有如下几个特色。

1. 以培养应用型人才为目标

本系列教材以应用型物联网人才为培养目标，在原有体制教育的基础上对课程进行深层次改革，强化“应用型技术”动手能力，使读者在经过系统、完整的学习后能够达到如下要求：

- 掌握物联网相关开发所需的理论和技术体系以及开发过程规范体系；
- 能够熟练地进行设计和开发工作，并具备良好的自学能力；
- 具备一定的项目经验，包括嵌入式系统设计、程序编写、文档编写、软硬件测试等内容；
- 达到物联网企业的用人标准，实现学校学习与企业工作的无缝对接。

2. 以新颖的教材架构来引导学习

本系列教材分为四个层次：知识普及、基础理论、应用开发、综合拓展，这四个层面的知识讲解和能力训练分布于系列教材之间，同时又体现在单本教材之中。具体内容在组织上划分为理论篇和实践篇：理论篇涵盖知识普及、基础理论和应用开发；实践篇包括企业应用案例和综合知识拓展等。

■ **理论篇：**最小学习集。学习内容的选取遵循“二八原则”，即重点内容占企业中常用技术的20%，以“任务驱动”方式引导80%的知识点的学习，以章节为单位进行组织，章节的结构如下：

- ✓ 本章目标：明确本章的学习重点和难点；
- ✓ 学习导航：以流程图的形式指明本章在整本教材中的位置和学习顺序；
- ✓ 任务描述：以“案例教学”驱动本章教学的任务，所选任务典型、实用；
- ✓ 章节内容：通过小节迭代组成本章的学习内容，以任务描述贯穿始终。

■ **实践篇：**以任务驱动，多点连成一线。以接近工程实践的应用案例贯穿始终，力求使学生在动手实践的过程中，加深对课程内容的理解，培养学生独立分析和解决问题的能力，并配备相关知识的拓展讲解和拓展练习，拓宽学生的知识面。

本系列教材借鉴了软件开发中“低耦合、高内聚”的设计理念，组织架构上遵循软件开发中的 MVC 理念，即在保证最小教学集的前提下可根据自身的实际情况对整个课程体系进行横向或纵向裁剪。

3. 以完备的教辅体系和教学服务来保证教学

为充分体现“实境耦合”的教学模式，方便教学实施，保障教学质量和学习效果，本系列教材均配备可配套使用的实验设备和全套教辅产品，可供各院校选购：

■ **实验设备：**与培养模式、教材体系紧密结合。实验设备提供全套的电路原理图、实验例程源程序等。

■ **立体配套：**为适应教学模式和教学方法的改革，本系列教材提供完备的教辅产品，包括教学指导、实验指导、视频资料、电子课件、习题集、题库资源、项目案例等内容，并配以相应的网络教学资源。

■ **教学服务：**教学实施方面，提供全方位的解决方案(在线课堂解决方案、专业建设解决方案、实训体系解决方案、教师培训解决方案和就业指导解决方案等)，以适应物联网专业教学的特殊性。

本系列教材由青岛东合信息技术有限公司编写，参与本书编写工作的有韩敬海、李红霞、卢玉强、张玉星、李瑞改、孙锡亮、刘晓红、袁文明等。参与本书编写工作的还有青岛农业大学、潍坊学院、曲阜师范大学、济宁学院、济宁医学院等高校的教师。本系列教材在编写期间还得到了各合作院校专家及一线教师的大力支持和协作。在本系列教材出版之际要特别感谢给予我们开发团队大力支持和帮助的领导及同事，感谢合作院校的师生给予我们的支持和鼓励，更要感谢开发团队每一位成员所付出的艰辛劳动。

由于水平有限，书中难免有不当之处，读者在阅读过程中如有发现，请通过公司网站(<http://www.dong-he.cn>)或公司教材服务邮箱(dh_iTeacher@126.com)联系我们。

高校物联网专业项目组
2013 年 11 月

目 录

理 论 篇

第 1 章 概述.....	2	2.5.1 概述.....	45
1.1 嵌入式系统.....	2	2.5.2 存储器控制器寄存器.....	46
1.1.1 嵌入式系统概述.....	2	2.5.3 存储器控制器编程.....	48
1.1.2 嵌入式系统发展.....	3	2.6 异常系统.....	53
1.2 嵌入式处理器.....	4	2.6.1 异常.....	53
1.2.1 处理器分类.....	4	2.6.2 中断机制.....	55
1.2.2 ARM 处理器.....	5	2.6.3 中断控制器寄存器.....	58
1.2.3 ARM 处理器系列.....	7	2.6.4 中断编程.....	61
1.3 嵌入式操作系统.....	9	2.7 定时器.....	66
1.4 嵌入式系统开发.....	10	2.7.1 概述.....	66
1.4.1 嵌入式系统开发流程.....	10	2.7.2 定时器寄存器.....	67
1.4.2 嵌入式系统开发环境.....	11	2.7.3 定时器编程.....	69
小结.....	14	2.8 ADC 和触摸屏.....	70
习题.....	15	2.8.1 ADC 概述.....	71
第 2 章 ARM 基础开发.....	16	2.8.2 触摸屏.....	72
2.1 S3C2440 概述.....	17	2.8.3 ADC 寄存器.....	72
2.2 编程基础.....	19	2.8.4 ADC 编程.....	74
2.2.1 S3C2440 体系结构.....	19	小结.....	77
2.2.2 汇编指令集.....	24	习题.....	77
2.2.3 汇编与 C 语言的交互编程.....	31	第 3 章 ARM 进阶开发.....	79
2.3 时钟与电源.....	35	3.1 Flash 编程.....	80
2.3.1 时钟体系.....	35	3.1.1 概述.....	80
2.3.2 电源模式.....	39	3.1.2 NAND Flash 控制器.....	82
2.4 GPIO.....	40	3.1.3 NAND Flash 控制器寄存器.....	83
2.4.1 概述.....	40	3.1.4 NAND Flash 实例.....	85
2.4.2 GPIO 寄存器.....	40	3.2 UART.....	98
2.4.3 GPIO 编程.....	42	3.2.1 概述.....	98
2.5 存储器控制器.....	45	3.2.2 S3C2440 UART.....	99

3.2.3	UART 编程	104	4.5.2	内核映像	187
3.3	USB	110	4.6	构建根文件系统	188
3.3.1	概述	110	4.6.1	文件系统概述	188
3.3.2	USB 主机控制器	117	4.6.2	根文件系统	190
3.3.3	USB 设备控制器	128	4.6.3	Busybox	191
3.3.4	USB 设备编程	129	4.6.4	构建根文件系统	195
3.4	DMA	138	4.7	Linux 内核启动分析	198
3.4.1	概述	138	小结		205
3.4.2	DMA 控制器	139	习题		206
3.4.3	DMA 控制器寄存器	140			
3.4.4	DMA 编程	143	第 5 章 驱动移植		207
3.5	LCD	144	5.1	概述	208
3.5.1	概述	144	5.1.1	设备驱动程序	208
3.5.2	LCD 控制器	146	5.1.2	设备类型与设备号	208
3.5.3	LCD 控制器寄存器	149	5.1.3	驱动模块	209
3.5.4	LCD 编程	150	5.1.4	驱动程序结构	211
3.6	MMU	159	5.1.5	驱动程序开发	211
3.6.1	概述	159	5.2	字符设备驱动程序	211
3.6.2	MMU 内存访问机制	164	5.2.1	字符设备特点	211
3.6.3	Cache	165	5.2.2	相关数据结构与函数	212
3.6.4	ARM 协处理器	166	5.2.3	驱动程序模型	217
3.6.5	MMU 编程	168	5.2.4	驱动程序实例	218
小结		170	5.3	块设备驱动程序	224
习题		170	5.3.1	块设备特点	224
			5.3.2	相关的数据结构和函数	225
第 4 章 系统构建		172	5.4	网络设备驱动程序	228
4.1	概述	173	5.4.1	网络设备特点	228
4.2	交叉编译环境	173	5.4.2	相关数据结构和函数	230
4.3	Linux 内核	174	5.4.3	DM9000 模块	234
4.3.1	Linux 内核特点	175	5.4.4	DM9000 驱动程序	235
4.3.2	内核结构	176	5.4.5	移植实例	248
4.4	内核编译配置	177	小结		248
4.4.1	配置工具	177	习题		249
4.4.2	内核 Makefile	177			
4.4.3	配置文件 Kconfig	179	第 6 章 应用编程		250
4.4.4	配置选项	180	6.1	概述	251
4.5	内核移植	182	6.1.1	应用程序	251
4.5.1	移植示例	184	6.1.2	API	251

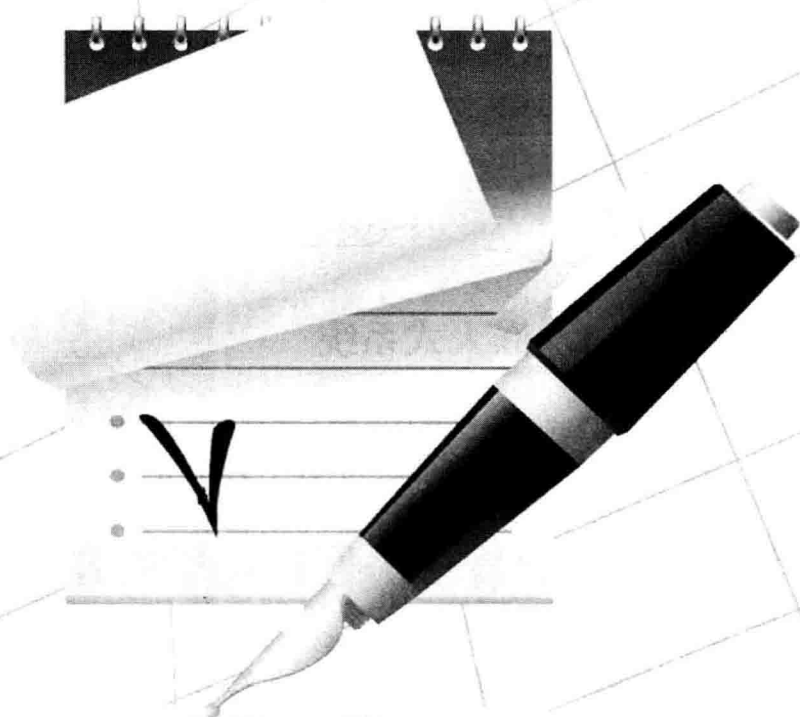
6.1.3 系统调用	252	7.1.3 几种流行的 GUI	329
6.1.4 API 与系统调用	253	7.1.4 Qt 及 Qt/Embedded	330
6.2 文件 I/O 编程	254	7.2 Qt/Embedded 开发平台的搭建	331
6.2.1 概述	254	7.2.1 Qt Creator 集成开发环境的搭建	331
6.2.2 基本 I/O 函数	255	7.2.2 编译安装触摸屏校验库 tslib	332
6.2.3 标准 I/O 函数	265	7.2.3 搭建 Qt/Embedded 开发环境	334
6.3 进程	269	7.3 Qt Creator 程序开发流程	335
6.3.1 概述	269	7.3.1 Qt Creator 启动界面简介	335
6.3.2 基本函数	271	7.3.2 创建“Hello World” GUI 应用程序	337
6.3.3 信号	276	7.3.3 纯代码编写程序	340
6.3.4 管道	282	7.4 程序框架类	341
6.3.5 信号量	288	7.4.1 QApplication 类	342
6.3.6 共享内存	293	7.4.2 QMainWindow 类	343
6.3.7 消息队列	298	7.4.3 QWidget 类	346
6.4 线程	304	7.4.4 QDialog 类	349
6.4.1 概述	304	7.4.5 QObject 类	351
6.4.2 基本函数	305	7.5 信号与槽机制	352
6.4.3 同步与互斥	310	7.5.1 信号与槽机制概述	352
6.5 网络编程	315	7.5.2 信号与槽的关联	353
6.5.1 TCP/IP 协议	315	7.5.3 元对象工具	355
6.5.2 socket 套接字	317	7.5.4 信号/槽使用示例	355
6.5.3 socket 编程实例	323	7.6 常用窗口部件	357
小结	326	7.6.1 QFrame 类族	357
习题	327	7.6.2 按钮部件	358
第 7 章 GUI 程序设计	328	7.6.3 行编辑控件	359
7.1 图形用户界面	329	7.6.4 滑块部件	361
7.1.1 图形用户界面的特征	329	小结	361
7.1.2 嵌入式系统图形用户界面特点	329	习题	362

实 践 篇

实践 1 概述	364	实践 2 ARM 基础开发	377
实践指导	364	实践指导	377
实践 1.G.1.	364	实践 2.G.1.	377
实践 1.G.2.	367	实践 2.G.2.	381
实践 1.G.3.	369	实践 3 ARM 进阶开发	394
实践 1.G.4.	371		

实践指导	394	实践 5.G.1	440
实践 3.G.1	394	知识拓展	444
实践 3.G.2	395		
实践 3.G.3	396	实践 6 应用编程	448
实践 3.G.4	398	实践指导	448
		实践 6.G.1	448
实践 4 系统构建	428	实践 6.G.2	451
实践指导	428		
实践 4.G.1	428	实践 7 GUI 程序设计	462
实践 4.G.2	430	实践指导	462
实践 4.G.3	433	实践 7.G.1	462
实践 4.G.4	437	实践 7.G.2	464
		实践 7.G.3	468
实践 5 驱动移植	440	实践 7.G.4	469
实践指导	440		

理论篇



第1章 概述

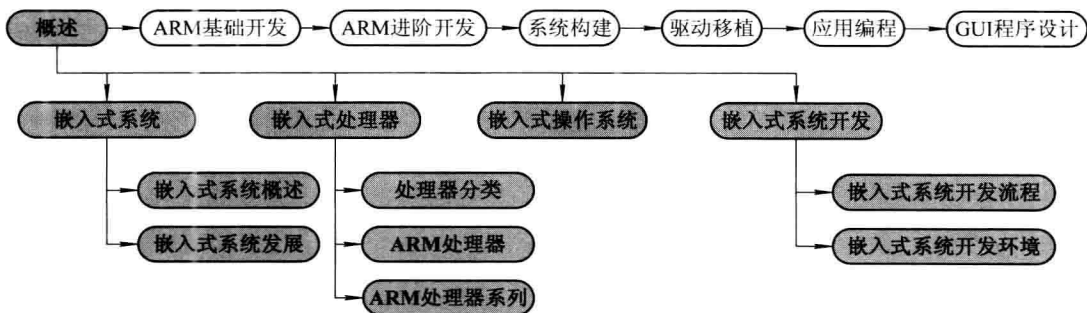


本章目标

- ◆ 了解嵌入式系统的定义和特点。
- ◆ 了解嵌入式系统的发展。
- ◆ 了解 ARM 处理器系列。
- ◆ 了解嵌入式 Linux 操作系统的特点。
- ◆ 了解嵌入式开发的不同。
- ◆ 了解嵌入式开发在不同阶段开发环境的使用。



学习导航



1.1 嵌入式系统

目前,嵌入式系统已经全面融入我们的日常生活中,小到 MP3、手机等微型数字化产品,大到智能家电、数字机床、机器人等都离不开嵌入式系统的应用。美国著名的未来学家尼葛洛庞帝在 1999 年曾预言,4~5 年后嵌入式系统是继 PC 和 Internet 之后最伟大的发明,这个预言已经成为现实。现在的嵌入式系统正处于高速发展的阶段。

1.1.1 嵌入式系统概述

嵌入式系统(Embedded System)的定义是:以应用为中心,以计算机技术为基础,软硬

件可裁剪,适用于应用系统,对功能、可靠性、成本、体积、功耗等方面有着特殊要求的专用计算机系统。它的主要特点是嵌入、专用。

嵌入式系统也是一种计算机系统,但它与通用的计算机系统的不同之处在于,嵌入式系统是一种“完全嵌入受控器件内部,为特定应用环境而设计的专用计算机系统”,具体体现在以下几个方面:

✧ 嵌入式系统是面向特定应用的。嵌入式 CPU 大多是为特定应用而设计的,具有功耗低、体积小、集成度高等特点,一般是包括多个外围设备接口的片上系统。

✧ 嵌入式系统是将先进的计算机技术、半导体技术和电子技术与各个行业的具体应用相结合的产物,这就决定了它必然是一个技术密集、资金密集、高度分散、不断创新的知识集成系统。

✧ 嵌入式系统的硬件和软件的设计都必须效率高、量体裁衣、去除冗余,这样系统才能体现更高的性价比,在选择上更有竞争力。

✧ 嵌入式系统和具体应用有机地结合在一起,它的升级换代和产品同步进行,因此嵌入式系统产品一旦进入市场,将具有较长的生命周期。

✧ 为了提高系统的执行速度和可靠性,嵌入式系统的软件一般都固化在存储器芯片中。

✧ 嵌入式系统本身并不具备自主开发的能力,因此必须有配套的开发工具和环境才能开发。

嵌入式系统是面向用户、面向应用的系统,这就决定了嵌入式系统的组成。一般来讲,一个嵌入式系统由应用软件、嵌入式操作系统和硬件设备组成,如图 1-1 所示。

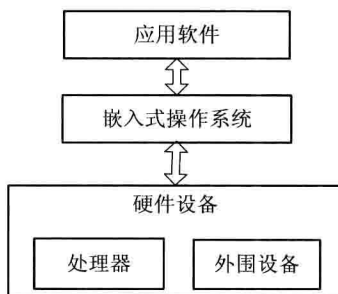


图 1-1 嵌入式系统组成

嵌入式系统各部分的功能如下:

✧ 应用软件可以具体实现用户的需求,是开发者在操作系统上基于硬件接口所开发的软件,其功能是完成嵌入式系统的功能应用;

✧ 嵌入式操作系统是整个系统的核心,它负责整个系统的软、硬资源的分配,任务的调度和控制等,可实现系统所要求的功能;

✧ 硬件设备包括嵌入式处理器及其外围设备,其中嵌入式处理器是硬件设备中的核心,外围设备是嵌入式系统用于实现存储、通信、显示等功能的辅助部件。

1.1.2 嵌入式系统发展

从 20 世纪 70 年代单片机的出现到各式各样的嵌入式微处理器、微控制器的大规模应用,嵌入式系统已经有了 30 多年的发展历史。

1. 过去

嵌入式系统的出现最初基于单片机(SCM, Single Chip Microcomputer)。利用单片机实现的设备已经初步具备了嵌入式的应用特点,但是这时的应用只是使用 8 位芯片,执行一些单线程的程序,还谈不上“系统”的概念。

最早的单片机是 Intel 公司的 8048,它出现在 1976 年。之后在 20 世纪 80 年代初,Intel

公司又进一步完善了 8048, 在它的基础上研制成功了 8051, 这在单片机的历史上是值得纪念的一页, 迄今为止, 51 系列的单片机仍然是最为成功的单片机, 在各种产品中有着非常广泛的应用。

20 世纪 80 年代早期, “嵌入式操作系统”出现了, 它是在微控制器(MCU, Micro Controller Unit)的基础上开发的。设计者们可以根据要求进行外围接口电路的扩展, 可以在专用操作系统上进行开发, 缩短了开发周期, 提高了开发效率, 但是 CPU 与系统的兼容性很差, 功能相对单一。

2. 现在

随着新技术、新工艺的飞速发展, 嵌入式系统技术也获得了广阔的发展。具体表现在:

- ✧ 嵌入式操作系统能够运行于各种不同类型的处理器上, 兼容性好、操作系统内核小、效率高, 并且有高度的模块化和扩展性。

- ✧ 同时具备文件和目录管理, 支持多任务, 支持网络应用, 具备图形窗口和用户界面, 具有大量的应用程序接口 API, 使得开发应用程序比较简单。

更重要的是, 设计者们还可以根据目标系统的要求进行软件、硬件的定制。

3. 未来

未来的时代是信息化、智能化和网络化的, 嵌入式系统还具有极其宽广的发展空间。根据目前用户对嵌入式设备的需求来看, 未来嵌入式系统应具有以下特点:

- ✧ 系统开发需要强大的开发工具和操作系统的支持。
- ✧ Internet 联网成为必然趋势。
- ✧ 精简系统内核、算法, 降低功耗和软硬件成本。
- ✧ 提供良好的人机交互界面。



1.2 嵌入式处理器

嵌入式系统硬件层的核心是嵌入式处理器, 它是控制、辅助系统运行的硬件单元。因此, 处理器的性能直接关系到整个嵌入式系统的性能。

1.2.1 处理器分类

目前, 嵌入式处理器已经超过 1000 种, 流行的架构体系有 30 多个系列。从最初的 4 位处理器以及目前仍然大规模使用的 8 位单片机, 到最新的受到广泛青睐的 32 位/64 位嵌入式 CPU, 处理器的速度越来越快、性能越来越强、封装引脚越来越多。根据功能以及应用场合的不同, 处理器可分为以下几类:

1. 嵌入式微处理器

嵌入式微处理器(MPU, Micro Processor Unit)是由通用计算机中的 CPU 演变而来的, 具有 32 位以上的处理器, 但它只保留了和嵌入式功能相关的硬件。与工业控制计算机相比, 嵌入式 MPU 具有体积小、重量轻、成本低、可靠性高等优点。目前主要的嵌入式 MPU 有 Am186/88、386EX、ARM/StrongARM 系列, 在市场上占有很大的优势。

2. 嵌入式微控制器

嵌入式微控制器(MCU, Micro Controller Unit)的片上外设资源比较丰富, 适合于控制, 因此称为微控制器, 其典型代表是单片机。单片机芯片内部集成了 RAM、ROM、I/O、A/D、TIMER 等基本功能外设, 实用性较强, 开发较容易, 价格也比较低廉, 因此 8/16 位单片机在嵌入式设备上仍有广泛的应用。比较有代表性的单片机有大家比较熟知的 51 核的单片机系列。近来 Atmel 公司推出的 AVR 单片机具有很高的性价比, 推动了单片机的发展。

但是 MCU 因资源的限制, 例如总线宽度一般为 4 位、8 位或 16 位, 使得 MCU 处理速度有限, 进行一些复杂的应用很困难, 运行操作系统更难。

3. 嵌入式 DSP 处理器

嵌入式 DSP 处理器(EDSP, Embedded Digital Signal Processor)是专门用于信号处理方面的处理器, 在系统结构和指令算法方面进行了特殊设计, 具有很高的编译效率和执行速度, 在数字滤波、光谱分析等仪器上已获得大规模的使用。但是 DSP 是运算密集处理器, 一般用于执行特定算法, 实现控制比较困难, 为了追求执行效率, 一般不使用操作系统。目前, 使用较为广泛的 DSP 是 TI 公司的 TMS320C2000/5000 系列。

4. 嵌入式片上系统

嵌入式片上系统(SOC, System On Chip), 顾名思义, 是直接在处理器片内嵌入操作系统。SOC 从整个系统的功能和性能出发, 利用 IP 核(Intellectual Property Core)复用和深亚微米技术, 采用软件和硬件结合的设计和验证方法, 综合考虑软硬件资源的使用成本, 设计满足性能要求的高效率、低成本的软硬件体系结构。目前高集成半导体工艺技术的飞速发展, 使得 SOC 成为替代集成电路的主要解决方案, 也已经成为当前微电子芯片发展的必然趋势。

1.2.2 ARM 处理器

ARM 处理器具有一个 32 位精简指令集(RISC)处理器架构, 其丰富的硬件资源和软件资源广泛地使用在嵌入式系统设计中, 常作为嵌入式系统处理器的首选。

1. ARM 简介

ARM 是 Advanced RISC Machines 的简写, 可以理解为以下三个含义:

- ◇ 一个生产高级 RISC(精简指令集)处理器的公司。
- ◇ 一种高级 RISC 的技术。
- ◇ 一类采用高级 RISC 的处理器。

也就是说, ARM 是一类嵌入式微处理器, 同时也是一个公司的名字。

ARM 公司成立于 1990 年, 总部位于英国剑桥, 它是一家专门从事 16 位/32 位 RISC 微处理器知识产权设计的供应商, 并不生产和销售实际的半导体芯片, 只是向生产和销售半导体的公司和厂商授权 ARM 内核, 并提供基于 ARM 架构的开发设计技术。

对于每一个授权公司和厂商来说, 他们获得的授权都是独一无二的。他们可以根据不同的应用领域和自身的技术优势, 适当加入外围电路, 形成自己的 ARM 微处理器芯片进入市场, 从而缩短了开发周期, 提升了产品竞争力。例如本书采用的 ARM9 系列的 ARM S3C2440, 即为三星公司在 ARM 公司向其授权的 ARM9 内核及开发技术基础上开发研制

而成的半导体芯片。

到目前为止，ARM 公司向 200 多家公司出售了 600 多个处理器许可证，全球已经销售了超过 150 亿枚基于 ARM 的芯片。

2. RISC

RISC 的英文全称是 Reduced Instruction Set Computing，即“精简指令集”。RISC 与 CISC(Complex Instruction Set Computing，复杂指令集)相对应，这是指令系统两个截然不同的优化方向。

RISC 的主要特点如下：

✧ 简化指令集，原则是使常用指令简单高效，不常用功能通过流水线技术和超标量技术加以弥补，增强了处理效率。

✧ 采用大量寄存器，并且使寄存器操作简单化，使大部分指令操作可以在寄存器中进行，提高了处理速度。

3. ARM 体系架构版本

ARM 体系架构各版本从最初开发至今已经有了重大改进，在保持各版本更高的兼容性的基础上不断完善。其典型架构版本发展历史以及代表内核如图 1-2 所示。

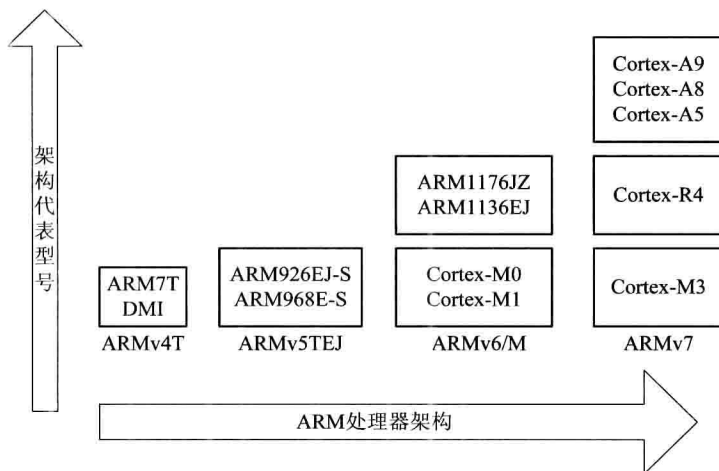


图 1-2 ARM 处理器架构进化

各个架构版本的特点如下所述：

✧ ARMv4T 架构：引进了 16 位 Thumb 指令集和 32 位 ARM 指令集，目的是在同一个架构中同时提供高性能和较高的代码密度。16 位 Thumb 指令集相对于 32 位 ARM 指令集可缩减高达 35% 的代码大小，同时保持 32 位架构的优点。采用此架构的内核如 ARM7TDMI，具体芯片有三星 S3C44B0x 系列等。

✧ ARMv5TEJ 架构：引进了数字信号处理(DSP)算法和 Jazelle Java 字节码引擎来启用 Java 字节码的硬件执行，从而改善了用 Java 编写的应用程序的性能。与非 Java 加速内核比较，Jazelle 将 Java 执行速度提高了 8 倍，并且减少了 80% 的功耗。许多基于 ARM 处理器的便携式设备中已经使用此架构，目的是在游戏和多媒体应用程序的性能方面提供显著改进的用户体验。采用此架构的内核如 ARM926EJ-S，具体芯片有 ATMEL 的 AT91SAM926x

系列等。

✧ ARMv6 架构: 引进了包括单指令多数据(SIMD)运算在内的一系列新功能。SIMD 扩展已针对多种软件应用程序(包括视频编解码器和音频编解码器)进行优化, 对于这些软件应用程序, SIMD 扩展最多可将性能提升 4 倍。此外, 还引进了作为 ARMv6 架构的变种的 Thumb-2 和 TrustZone 技术。采用此架构的内核如 ARM1176JZ, 具体芯片有三星 S3C6410x 系列等。

✧ ARMv7 架构: 此架构是目前 ARM 公司最新的架构, 所有 Cortex 处理器都实现了 ARMv7 架构(ARMv6/M 的 Cortex-M 系列处理器除外)。所有 ARMv7 架构都实现了 Thumb-2 技术(一个经过优化的 16 位/32 位混合指令集), 在保持与现有 ARM 解决方案的代码完全兼容的同时, 既具有 32 位 ARM 指令集的性能优势, 又具有 16 位 Thumb 指令集的代码大小优势。ARMv7 架构还包括 NEON 媒体加速技术, 该技术可将 DSP 和媒体处理吞吐量提升高达 400 个百分点, 并提供改进的浮点支持以满足下一代 3D 图形和游戏物理学以及传统嵌入式控制应用程序的需要。

1.2.3 ARM 处理器系列

目前, 基于 ARM 内核结构的处理器有以下系列: ARM7 系列、ARM9 系列、ARM9E 系列、ARM10 系列、ARM11 系列、SecurCore 系列以及 Intel StrongARM 系列等, 其中 ARM7、ARM9、ARM9E、ARM10 和 ARM11 系列为通用处理器系列。每个系列名字都有后缀字母, 例如 ARM7TDMI、ARM920T 等等, 这些后缀字母的含义如下:

- ✧ T: 表示支持 Thumb 指令集。
- ✧ D: 表示支持片上调试。
- ✧ M: 表示支持内嵌硬件乘法器。
- ✧ I: 表示支持片上断点和调试点。
- ✧ E: 表示支持增强型 DSP 功能。

下面分别介绍各个系列的特点以及应用领域。

1. ARM7

ARM7 系列处理器包括 ARM7TDMI、ARM7TDMI-S、ARM720T、ARM7EJ 等。其主要特点如下:

- ✧ 32 位 RISC 处理器。
- ✧ 具有嵌入式 ICE, 调试开发方便。
- ✧ 兼容 16 位的 Thumb 指令集, 代码密度高。
- ✧ 0.9 MIPS/MHz 的 3 级流水线结构。
- ✧ 支持小型操作系统。
- ✧ 主频最高可达 130 MHz。
- ✧ 指令与其他通用系列兼容, 产品容易升级。

主要应用领域: 工业控制、Internet 设备、网络和调制解调器设备以及移动电话等。

2. ARM9

ARM9 系列处理器主要包括 ARM920T、ARM922T 和 ARM940T。其主要特点如下: