



21世纪高等院校规划教材

嵌入式Linux开发基础

BASICS OF EMBEDDED LINUX DEVELOPMENT

谢 川 主编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

21 世纪高等院校规划教材

嵌入式 Linux 开发基础

主编 谢 川

参编 贺玲玲 谈 学 江朝元 朱超平 陈旭东

内 容 简 介

嵌入式系统是近年来的热门话题,已经遍及人们生活中的每个角落。本书针对在嵌入式市场上颇具竞争力的 S3C2410X 处理器和开放源码的 Linux 2.6 操作系统,介绍嵌入式 Linux 开发技术的基础知识。主要内容包括:嵌入式系统概述、ARM 处理器概述、Linux 系统概述、Linux 命令、Linux 编程基础、构建嵌入式 Linux 开发平台、嵌入式 Linux 设备驱动开发、文件 I/O 编程、多进程编程、多线程编程、网络编程、嵌入式图形用户界面设计和电子菜单设计等,在讲述各个操作和编程环节时配备大量的实例程序以增进读者对所学知识的理解和应用能力。

本书的特点是集嵌入式系统开发理论知识和实践环节于一体,并结合广州致远电子有限公司生产的 MagicARM2410 开发板给出大量实例,所有的实例程序和操作过程都经过验证可行。

本书语言通俗易懂,内容丰富,知识涵盖面广,非常适合作为高等院校嵌入式 Linux 技术类课程的教材,也可供从事嵌入式 Linux 系统开发的初级工程师、Linux 程序开发人员阅读和学习。

图书在版编目(CIP)数据

嵌入式 Linux 开发基础 / 谢川主编. — 北京: 中国铁道出版社, 2011. 8
21 世纪高等院校规划教材
ISBN 978-7-113-12950-7

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

中国版本图书馆 CIP 数据核字 (2011) 第 120924 号

书 名: 嵌入式 Linux 开发基础
作 者: 谢 川 主编

策划编辑: 杨 勇

责任编辑: 吴宏伟 徐盼欣

封面设计: 付 巍

版式设计: 刘 颖

读者热线: 400-668-0820

封面制作: 白 雪

责任印制: 李 佳

出版发行: 中国铁道出版社(北京市宣武区右安门西街 8 号 邮政编码: 100054)

印 刷: 三河市华丰印刷厂

版 次: 2011 年 8 月第 1 版 2011 年 8 月第 1 次印刷

开 本: 787mm×1092mm 1/16 印张: 21.75 字数: 526 千

印 数: 3 000 册

书 号: ISBN 978-7-113-12950-7

定 价: 33.00 元

版权所有 侵权必究

凡购买铁道版图书,如有印制质量问题,请与本社教材研究开发中心批销部联系调换。

嵌入式系统的热潮已经遍及神州大地，作为嵌入式系统的重头戏——ARM 处理器和嵌入式 Linux 操作系统自然成为广大开发人员关注的重点。Samsung 公司生产的基于 ARM920T 核的 S3C2410X 处理器因其高性能、低价格、低功耗、片内集成 MMU（内存管理单元）而广泛应用于手持设备、POS 机、多媒体设备中；开源软件 Linux 2.6 操作系统是目前版本最新、最稳定、功能最完善的 Linux 操作系统。对于大多数嵌入式开发人员而言，开发工作应有相应的侧重点，如偏向硬件设计或软件设计，本书重点介绍基于 S3C2410X 平台上的 Linux 2.6 应用软件设计。

由于 Linux 操作系统是开源软件，开发者不能从中获利，因此造成大多数开源项目缺少文档，即缺少“官方”文档。对于其他非官方写作的文档，由于其作者本身已经是高手，他们编写的资料对已入门者具有重要的参考价值，而对于初学者而言，只可作为“手册”来使用，而且这些大多是基于 Linux 2.4 的，与 Linux 2.6 内核存在较大差异。

对于 Linux 2.6 的应用来说，主要有三方面：移植 Linux 操作系统内核到嵌入式目标板、编写嵌入式 Linux 设备驱动程序、编写 Linux 应用软件。笔者针对嵌入式 Linux 2.6 的应用情况，通过阅读大量的资料和源码，删去大量的“细枝繁叶”，使其脉络清晰，并设计多个应用程序实例，这些实例包括 Linux 内核移植、设备驱动程序编写、多进程与多线程程序编写、嵌入式 Linux 网络编程、嵌入式图形界面设计等方面。希望读者在阅读本书后，由衷地说：“《嵌入式 Linux 开发基础》一书成功地将我引入到了嵌入式 Linux 世界。”

本书的各章节安排如下：

第 1 章是嵌入式系统概述。本章主要介绍嵌入式系统简介，嵌入式系统的组成结构，嵌入式系统开发流程，以及嵌入式系统的发展趋势，以便读者对嵌入式系统有一个初步认识。

第 2 章是 ARM 处理器概述。本章主要介绍 ARM 的概念，ARM 体系结构的分类版本及扩展技术，ARM 微处理器系列，ARM 处理器的工作状态、存储格式、工作模式、应用领域和选型，因后面各章节介绍的嵌入式 Linux 开发环境、设备驱动程序等都是基于 S3C2410X 处理器来开展的，因此本章重点介绍 Samsung 公司生产的 S3C2410X 处理器。

第 3 章是 Linux 系统概述。本章主要介绍 Linux 系统的历史与现状、安装、文件及文件系统、启动过程，让读者全面认识 Linux 操作系统。

第 4 章是 Linux 命令。本章主要介绍 shell 的概念和一些常用的 shell 命令，通过本章的学习，读者能够熟练地使用 Linux，为基于 Linux 的程序设计打下基础。

第 5 章是 Linux 编程基础。本章主要介绍嵌入式 Linux 应用开发过程中所用到的各种 GNU 工具，包括 vi 编辑器、GCC 编译器、GDB 调试器、make 工程管理等。为了提高 Linux

程序开发和调试的工作效率，本章还介绍了 Linux shell 编程的基础知识。

第 6 章是构建嵌入式 Linux 开发平台。本章主要介绍嵌入式交叉编译环境搭建、U-Boot 移植、Linux 内核移植、根文件系统移植。

第 7 章是嵌入式 Linux 设备驱动开发。嵌入式设备根据其特征可分为 3 种类型：字符型设备、块设备和网络设备，各有不同的设备驱动开发过程。本章分别介绍了这 3 种设备驱动程序的开发内容，并以 S3C2410X 处理器的四线电阻式触摸屏设备驱动的开发作为实例。

第 8 章是文件 I/O 编程。在 Linux 环境下，一切皆文件，各种设备也作为文件来看待，因此在嵌入式 Linux 环境中，文件 I/O 操作是其他应用程序的基础，包括文件打开、文件读写、文件关闭等。

第 9 章是多进程编程。操作系统最主要的功能就是实现多任务管理，多进程编程也即成为 Linux 环境下的必须要求。本章主要介绍 Linux 进程概述，Linux 进程的基本操作，Linux 进程间通信。

第 10 章是多线程编程。由于线程的高效性和可操作性，在大型应用程序开发中使用得非常广泛，本章将介绍轻量级进程——线程的开发，具体包括 Linux 线程概述，Linux 线程编程实现。

第 11 章是网络编程。网络在嵌入式系统中的应用非常广泛，常见的应用都会与网络有关，因此掌握 Linux 网络编程对嵌入式开发显得非常重要。本章主要介绍 TCP/IP 协议的基础知识和嵌入式 Linux 网络编程的基本方法。

第 12 章是嵌入式图形界面设计。本章主要介绍 Qt4 编程及 Qt4/Embedded 图形界面设计，旨在为嵌入式产品提供美观的图形化用户界面接口。

第 13 章是电子菜单设计。本章介绍一个基于蓝牙技术的嵌入式电子菜单系统，让读者从系统层面了解嵌入式 Linux 的开发技术。

本书是面向初学者的入门书籍，不可能也没有必要介绍大型的 Linux 应用程序以及设备驱动程序的各个方面。建议本课程教学理论知识占 40 学时，实验占 22 学时，如果读者需要深入学习，可参考本书参考文献中列出的部分资料及其他相关书籍和源码，也可从 <http://www.edusources.net> 下载其他教学资源。

本书主要由重庆工商大学计算机科学与信息工程学院谢川编写，贺玲玲、谈学、江朝元、朱超平、陈旭东等参与了部分具体工作，如规划、统稿、审稿等。

本书介绍的所有程序实例都在广州致远电子有限公司生产的 MagicARM2410 开发平台上验证通过，确保其正确性。但限于编者的理论水平和经验，书中难免有疏漏之处，希望读者谅解，并反馈宝贵意见，让我们共同做到更好。

感谢中国铁道出版社的大力支持，他们为规划本书付出了很多心血，没有他们的努力，本书很难及时完成和出版。

编 者

2011 年 5 月

第 1 章 嵌入式系统概述	1
1.1 嵌入式系统简介	1
1.1.1 嵌入式系统发展简史	1
1.1.2 嵌入式系统的定义	3
1.1.3 嵌入式系统的特点	3
1.1.4 嵌入式系统的分类	4
1.1.5 嵌入式系统的应用范围	4
1.2 嵌入式系统的组成结构	7
1.2.1 嵌入式系统的硬件组成结构	8
1.2.2 嵌入式系统的软件层次结构	11
1.2.3 启动程序 BootLoader	13
1.2.4 嵌入式操作系统简介	14
1.3 嵌入式系统开发概述	17
1.3.1 嵌入式开发考虑的要素	17
1.3.2 软硬件协同设计	17
1.3.3 嵌入式系统开发的基本流程	18
1.4 嵌入式系统的发展趋势	21
思考与练习	22
第 2 章 ARM 处理器概述	23
2.1 ARM 的概念	23
2.2 ARM 体系结构的分类版本及扩展技术	24
2.2.1 ARM 处理器的体系结构分类版本	24
2.2.2 ARM 体系结构的扩展	25
2.3 ARM 微处理器系列	27
2.3.1 ARM7 微处理器系列	27
2.3.2 ARM9 微处理器系列	28
2.3.3 ARM9E 微处理器系列	28
2.3.4 ARM10E 微处理器系列	29
2.3.5 ARM11 微处理器系列	30
2.3.6 Cortex 微处理器系列	31
2.3.7 SecurCore 系列	33
2.3.8 Intel 的 Xscale 和 StrongARM	34
2.4 ARM 处理器的工作状态	34

2.5	ARM 处理器的存储格式	35
2.6	ARM 处理器的工作模式	35
2.7	ARM 处理器的应用领域和选型	36
2.7.1	应用领域	36
2.7.2	应用选型	37
2.8	S3C2410A 处理器简介	38
2.8.1	S3C2410A 处理器概述	38
2.8.2	Magic ARM2410 开发板	43
	思考与练习	46
第 3 章	Linux 系统概述	47
3.1	Linux 的历史与现状	47
3.1.1	Linux 的发展概述	47
3.1.2	Linux 作为嵌入式操作系统的优势	51
3.1.3	Linux 发行版本	52
3.1.4	内核版本号与发行版本号	54
3.2	Linux 安装	54
3.2.1	Linux 安装基础知识	54
3.2.2	安装前的准备工作	56
3.2.3	RHEL 5.5 安装过程	57
3.3	Linux 文件及文件系统	67
3.3.1	文件类型及文件属性	67
3.3.2	文件系统类型介绍	69
3.3.3	Linux 目录简介	70
3.4	Linux 启动过程	71
3.4.1	概述	71
3.4.2	内核引导阶段	72
3.4.3	init 阶段	72
	思考与练习	73
第 4 章	Linux 命令	74
4.1	初识 shell	74
4.2	Linux 常用命令	75
4.2.1	用户系统相关命令	75
4.2.2	文件目录相关命令	80
4.2.3	常用的网络命令	88
4.2.4	比较与合并文件相关命令	94
4.2.5	压缩打包相关命令	97
	思考与练习	100

第 5 章 Linux 编程基础	101
5.1 vi 编辑器	101
5.1.1 vi 简介	101
5.1.2 vi 基本编辑命令	103
5.2 GCC 编译器	105
5.2.1 GCC 编译过程	105
5.2.2 GCC 编译选项介绍	108
5.3 GDB 调试器	112
5.3.1 GDB 使用流程	112
5.3.2 GDB 基本命令	115
5.4 make 工程管理器	118
5.4.1 Makefile 文件格式	119
5.4.2 Makefile 文件组成	120
5.4.3 make 管理器选项	125
5.4.4 使用 autoTools	125
5.5 Linux shell 编程	129
5.5.1 shell 变量	129
5.5.2 shell 测试命令	131
5.5.3 shell 流控制结构	133
5.5.4 shell 函数	137
思考与练习	138
第 6 章 构建嵌入式 Linux 开发平台	140
6.1 嵌入式交叉编译环境搭建	140
6.2 U-Boot 移植	142
6.2.1 引导系统简介	142
6.2.2 U-Boot 概述	143
6.2.3 U-Boot 移植过程	145
6.3 Linux 内核移植	155
6.3.1 概述	155
6.3.2 Linux 内核移植过程	156
6.4 根文件系统移植	161
6.4.1 根文件系统概述	162
6.4.2 BusyBox 根文件系统移植过程	163
思考与练习	168
第 7 章 嵌入式 Linux 设备驱动开发	169
7.1 设备驱动概述	169
7.1.1 驱动模块	169
7.1.2 设备文件分类	170
7.1.3 设备号	171

7.1.4	设备驱动层次结构	171
7.1.5	设备驱动程序与外界接口	171
7.1.6	设备驱动特点	172
7.2	Linux 字符设备驱动	172
7.2.1	cdev 结构体	173
7.2.2	分配和释放设备号	174
7.2.3	file_operations 结构体	174
7.2.4	字符设备驱动程序组成	178
7.3	触摸屏设备驱动实例介绍	181
7.3.1	S3C2410X 触摸屏工作原理	181
7.3.2	触摸屏设备驱动中的数据结构	183
7.3.3	触摸屏设备驱动中的硬件控制	184
7.3.4	触摸屏驱动模块加载和卸载函数	184
7.3.5	触摸屏驱动中断、定时器处理程序	185
7.3.6	触摸屏驱动的打开、释放函数	187
7.3.7	触摸屏驱动的读函数	188
7.3.8	触摸屏驱动的轮询与异步通知函数	188
7.4	块设备驱动	189
7.4.1	字符设备与块设备 I/O 操作的区别	189
7.4.2	block_device_operations 结构体	189
7.4.3	gendisk 结构体	190
7.4.4	request 和 bio 结构体	191
7.4.5	块设备驱动的注册与注销	197
7.4.6	块设备驱动模块的加载与卸载	197
7.4.7	块设备驱动的打开与释放	199
7.4.8	块设备驱动的 I/O 控制	199
7.4.9	块设备驱动的 I/O 请求处理	200
7.5	网络设备驱动	203
7.5.1	Linux 网络设备驱动的结构	203
7.5.2	网络设备驱动的注册与注销	211
7.5.3	网络设备的初始化	212
7.5.4	网络设备的打开与释放	213
7.5.5	数据发送流程	214
7.5.6	数据接收流程	215
7.5.7	网络连接状态	217
7.5.8	参数设置和数据统计	219
思考与练习		221
第 8 章	文件 I/O 编程	222
8.1	Linux 系统调用及用户编程接口	222

8.2	Linux 文件及文件描述符	223
8.3	不带缓存的文件 I/O 操作	224
8.3.1	open()和 close()函数	224
8.3.2	read()、write()和 lseek()函数	226
8.3.3	fcntl()函数	228
8.4	嵌入式 Linux 设备文件 I/O 操作实例	231
8.4.1	串口设备概述	231
8.4.2	串口设备设置	232
8.4.3	串口设备的使用	235
	思考与练习	237
第 9 章	多进程编程	238
9.1	Linux 进程概述	238
9.1.1	Linux 进程相关概念	238
9.1.2	Linux 进程结构与模式	240
9.1.3	Linux 进程控制	241
9.2	Linux 进程的基本操作	241
9.2.1	创建进程 fork()函数	241
9.2.2	系统调用 exec()函数族	243
9.2.3	系统调用 exit()和 _exit()函数	245
9.2.4	系统调用 wait()和 waitpid()函数	246
9.2.5	系统调用 sleep()函数	248
9.3	Linux 进程间通信	248
9.3.1	管道通信	249
9.3.2	消息队列	255
9.3.3	信号量	260
9.3.4	共享内存	264
	思考与练习	268
第 10 章	多线程编程	269
10.1	Linux 线程概述	269
10.2	Linux 线程编程实现	270
10.2.1	Linux 基本线程相关函数	270
10.2.2	Linux 线程互斥锁	273
10.2.3	Linux 线程信号量	276
	思考与练习	279
第 11 章	网络编程	280
11.1	TCP/IP 协议概述	280
11.1.1	TCP/IP 协议簇	281
11.1.2	UDP 与 TCP 简介	282

11.2	socket	287
11.3	Linux 网络通信辅助函数介绍	288
11.4	socket 基础编程	290
11.4.1	socket()函数	290
11.4.2	bind()函数	291
11.4.3	listen()函数	291
11.4.4	accept()函数	291
11.4.5	connect()函数	292
11.4.6	send()和 recv()函数	292
11.4.7	sendto()和 recvfrom()函数	293
11.4.8	网络通信编程实例	294
	思考与练习	297
第 12 章	嵌入式图形用户界面设计	298
12.1	嵌入式 GUI 简介	298
12.1.1	嵌入式 GUI 的特点	298
12.1.2	嵌入式 GUI 的种类	298
12.2	Qt 介绍	299
12.2.1	Qt/Embedded 简介	300
12.2.2	建立 Qt 开发环境	301
12.2.3	Qt designer 主界面	303
12.3	Qt4 编程	304
12.3.1	信号和槽	304
12.3.2	hello 窗口实例	306
12.3.3	控件的几何排列实例	308
12.3.4	对话框使用实例	310
12.3.5	主窗口使用实例	313
12.3.6	Qt designer 图形设计器使用实例	319
12.4	Qt4/Embedded 的嵌入式应用	322
12.4.1	建立 Qt4/Embedded 开发环境	322
12.4.2	Qt4/Embedded 程序实例	324
	思考与练习	325
第 13 章	电子菜单设计	326
13.1	功能描述	326
13.2	系统设计	327
13.2.1	开发环境	328
13.2.2	系统功能模块设计	328
13.2.3	项目小结	335
参考文献		336

第 1 章 | 嵌入式系统概述

本章介绍嵌入式系统开发的基础知识，从嵌入式计算机的历史由来、嵌入式系统的定义、嵌入式系统的基本特点、嵌入式系统分类及应用、嵌入式系统软硬件组成、嵌入式系统的开发流程以及嵌入式技术的发展趋势等方面进行介绍，涉及嵌入式系统开发的基本内容，可使读者系统地建立起嵌入式系统的整体概念。

1.1 嵌入式系统简介

嵌入式系统的应用日益广泛，可以说无处不在，嵌入式系统的快速发展也极大地丰富、延伸了嵌入式系统的概念。

1.1.1 嵌入式系统发展简史

实际上，嵌入式系统这一术语并不是最近出现的新名词，它已经存在了大约半个世纪之久。在早期的工业控制领域，计算机就已经嵌入到应用对象中。例如，20 世纪 60 年代，它被用于对电话交换进行控制，当时称为存储程序控制系统（stored program control system）。但由于那时的计算机无论是体积、功耗还是价格都难以满足各种设备尤其是小型设备的需求，因此严格意义上的嵌入式计算机应该从微处理器出现开始算起。

1. 微处理器的发展

从 20 世纪 70 年代起，VLSI 技术的运用可以实现将整个中央处理器集成在一个芯片上。1971 年，Intel 公司生产了世界上第一台 4 位微处理器 Intel 4004，它本身就是为嵌入式应用（即计算器）而设计的。它仅提供基本的算术运算功能，因此不能算作通用计算机。第二年，Intel 公司又研制成功了 8 位微处理器 Intel 8008。同 4004 一样，8008 也是为专门用途而设计的嵌入式微处理器。它们都属于第一代微处理器，其典型应用是计算器、打字机、交通灯控制等。

1974 年，第二代 8 位微处理器 Intel 8080 诞生。作为代替传统复杂电子逻辑电路的器件，8080 是诸如字处理机、导航系统以及巡航导弹这样具有可编程、体积小等特点的嵌入式应用的标准微处理器。同时期微处理器的代表性产品还有 Motorola 公司的 M6800、Zilog 公司的 Z80 以及 Intel 公司的 8085 等。

1978 年，出现了第三代微处理器，其典型代表为 Intel 公司的 8086、Zilog 公司的 Z8000 以及 Motorola 公司的 M68000。第三代微处理器的性能较第二代提高了近 10 倍，使得微处理器从专用目的微处理器发展成通用微型计算机系统的中央处理器。1981 年，IBM 公司推出了基

于 8088（8086 的变种产品，8 位外部总线）的个人计算机系统 IBM-PC，使得计算机进入了 PC 时代。

1983 年，第四代 32 位微处理器问世，其典型代表为美国国家半导体公司的 32032，Motorola 公司的 M68020，Intel 公司的 80386、i486 等。基于 32 位微处理器的微机系统在性能上可与 20 世纪 70 年代的大、中型计算机相媲美。

从 20 世纪 80 年代起，随着微电子技术的迅猛发展，半导体厂商致力于将微处理器、存储器件、I/O 接口、A/D 转换器、D/A 转换器等集成到一个芯片上，这就是单片机，或者称为微控制器（micro-controller unit，MCU）。作为面向 I/O 设计的微控制器，单片机在过去的 20 多年间，被广泛应用于仪器仪表、智能控制、消费电子、军事电子等各种领域。即使在今天，它依然占据着普及型嵌入式应用的大部分市场份额。

但是，单片机应用只是嵌入式应用的初级阶段，其控制逻辑主要是循环查询的前后台监控程序。随着应用复杂性的提高，如网络、GUI（graphical user interface）和多媒体技术的广泛使用，迫切需要更高性能的微处理器及操作系统的支持。于是，高性能的 32 位 RISC 微处理器、数字信号处理器（digital signal processor，DSP）、片上系统（system on chip，SoC）以及实时操作系统等成为高端嵌入式应用的主要组成部分。

2. 嵌入式系统的发展

真正意义上的嵌入式系统是从 20 世纪 70 年代随着微处理器的出现发展起来的。伴随着微处理器的发展，嵌入式系统至今已经有 30 多年的历史，它大致经历了以下 4 个发展阶段：

第一阶段是以 4 位到 8 位单片机为核心的可编程控制器系统，同时具有检测、伺服、指示设备相配合的功能。这一阶段系统的主要特点是：结构和功能相对单一、效率较低、存储容量较小、几乎没有人机交互接口，其应用范围主要局限于一些专业性极强的工业控制系统中，一般没有操作系统支持，通过汇编语言对系统进行直接控制。尽管这类系统使用简单方便、价格便宜，但是，对于需要大容量存储介质、友好的人机交互界面、远距离或无线通信的高性能现代化工业控制和后 PC 时代新兴的信息家电等领域而言，已经远远不能满足要求。

第二阶段是以 8 位到 16 位嵌入式中央处理器（CPU）为基础，以简单操作系统为核心的嵌入式系统。这一阶段系统的主要特点是：CPU 种类繁多、通用性较弱、系统开销小、操作系统兼容性和扩展性较低、应用软件较为专业、用户界面不够友好以及网络功能较弱。这种嵌入式系统的主要任务是提高应用对象的智能化水平，如智能化仪器仪表、智能化家电等。

第三阶段是以 32 位 RISC 嵌入式中央处理器加嵌入式操作系统为标志的嵌入式系统。这一阶段系统的主要特点是：嵌入式操作系统能够运行于各种不同类型的处理器之上，操作系统内核精简小巧、效率高、模块化程度高，具有文件和目录管理，支持多任务，支持网络操作，具有图形窗口和用户界面功能，具有大量的应用程序接口以及各种组件，开发程序简单、高效，能满足日益复杂的应用需求。这也是现在通常所说的典型的嵌入式系统。然而，它在通用性、兼容性和扩展性方面仍有待改进。

第四阶段是基于 Internet 接入为标志的嵌入式系统，这是一个正在迅速发展的阶段。随着网络在人们生活中的地位日益重要，越来越多的应用需要采用支持 Internet 接入功能的嵌入式系统，如手机、PDA，甚至电视机等传统家电都需要上网，所以在嵌入式系统中使用网络操作系统将成为今后的发展趋势。

1.1.2 嵌入式系统的定义

目前,对嵌入式系统的定义多种多样,但没有一种是全面的。下面给出两种比较流行的定义:

① 嵌入式系统是以应用为中心,以计算机技术为基础,软硬件可裁剪,适应应用系统对功能、可靠性、成本、功耗及体积严格要求的专用计算机系统。

② 嵌入式系统是设计完成复杂功能的硬件和软件,并使其紧密耦合在一起的计算机系统。术语“嵌入式”反映了这些系统通常是更大系统中的一个完整的部分,称为嵌入的系统。嵌入的系统中可以共存多个嵌入式系统。

两种定义的出发角度不同,前者是从技术的角度来定义的,后者是从系统的角度来定义的。事实上,在大多数情况下,嵌入式系统被真正地嵌入,即它们是“系统中的系统”,没有自身的功能。例如,数字机顶盒(digital set-top box, DST)可以在许多家庭娱乐场所中找到。数字音频/视频解码系统称为A/V解码器(A/V decoder),是DST的一个完整部分,是一个嵌入式系统,它接收单个的多媒体流,并且产生声音和视频帧作为输出。DST从卫星接收的信号中包含多个流或频道,因此,A/V解码器与传输流解码器链接工作。传输流解码器也是一个嵌入式系统。传输流解码器解调收到的多媒体流到分离的频道上,并且只将所选的频道送给A/V解码器。

某些情况下,嵌入式系统在功能上是独立的系统。例如,路由器是独立的嵌入式系统,它由特殊的通信处理器、内存、许多网络访问接口(称为网络端口)以及实现IP分组的算法的特殊软件组成。换句话说,路由器是一个独立的嵌入式系统,通过程序化的路由算法实现IP分组从一个端口到另一个端口。

1.1.3 嵌入式系统的特点

由于嵌入式系统是应用于特定环境下,针对特定用途来设计的系统,所以不同于通用计算机系统。嵌入式系统是针对具体应用设计的“专用系统”,它的软件和硬件都必须高效率地设计,“量体裁衣”,去除冗余,力争在较少的资源上实现更高的性能。与通用计算机系统相比,它具有以下显著特点:

① 与应用密切相关,执行特定功能。任何一个嵌入式系统都和特定应用相关,用途固定。嵌入式系统的硬件和软件都必须高效率地设计,要具备良好的软硬件可裁剪性,力争在满足应用目标的前提下使系统最精简。

② 运行环境差异大。嵌入式系统无所不在,但运行环境差异也很大,可以运行在飞机上、冰天雪地的两极中、汽车里、要求温度湿度恒定的科学实验室等。特别是在恶劣的环境或突然断电的情况下,要求系统仍能够正常工作。

③ 具有实时性约束。嵌入式系统大多都是实时系统,有时限要求。若违反实时约束则可能使系统瘫痪或不可用。特别是对于一些强实时性嵌入式系统,如军事电子、飞行控制、核电控制、卫星发射等,如果违反实时约束有可能造成非常严重的后果。

④ 系统资源少。由于对成本、体积、功耗有严格要求,使得嵌入式系统的资源(如内存容量、I/O接口数量等)都非常紧缺,因此软硬件都需要仔细设计以充分利用有限的系统资源。

⑤ 系统可靠性要求高。嵌入式系统应用环境复杂,经常应用于无人值守环境中,要求嵌

入式系统具有较强的容错能力，因此需要有相应的可靠性保障机制，如看门狗定时器（watchdog timer, WDT）等。

⑥ 具有较长的生命周期。嵌入式系统是和实际具体应用有机结合的产物，它的升级换代也是和具体产品同步进行的。因此，一旦产品定型并进入市场，目标代码就固化在非易失性存储器芯片中，一般具有较长的生命周期。

⑦ 需要交叉开发环境和调试工具。嵌入式系统本身不具备自举开发能力，即使在设计完成以后，用户通常也不能对其中的程序功能进行修改，必须有一套开发工具和环境才能进行开发和测试。这些工具和环境一般是基于通用计算机上的硬件设备、逻辑分析仪、示波器、万用表等专门的软硬件开发和调试工具。

1.1.4 嵌入式系统的分类

嵌入式系统按表现形式及使用硬件种类可分为：

- 系统中使用含程序或算法的处理器器的嵌入式系统称为芯片级嵌入；
- 系统中使用某个核心模块的嵌入式系统称为模块级嵌入。

嵌入式系统按软件实时性可分为：

- 非实时系统（如掌上计算机）；
- 软实时系统（如消费类产品）；
- 硬实时系统（工业实时控制系统）。

1.1.5 嵌入式系统的应用范围

在工业和服务领域中，大量嵌入式技术也已经应用于工业控制、数控机床、智能工具、工业机器人、服务机器人等各个行业，正在逐渐改变着传统的工业生产和服务方式。例如，飞机的电子设备、城市地铁购票系统等都可以应用嵌入式系统来实现。嵌入式系统的应用领域如图 1-1 所示。

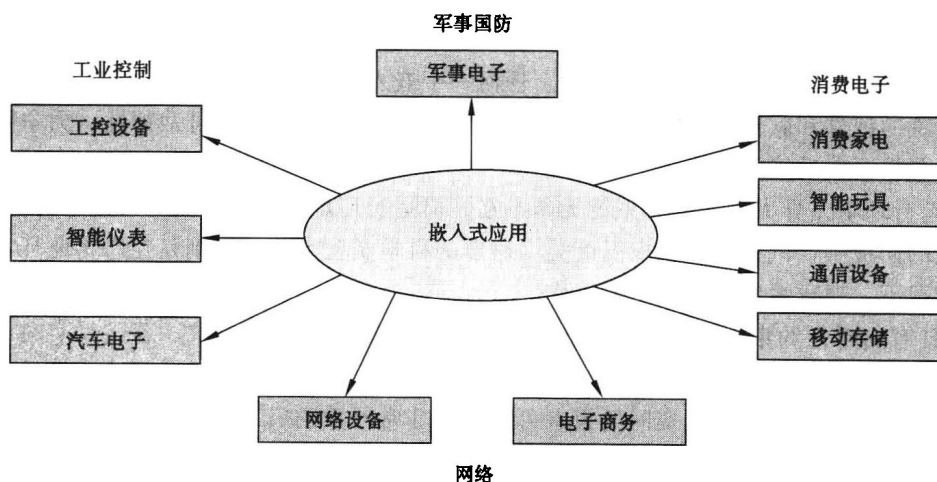


图 1-1 嵌入式系统应用领域

嵌入式技术可应用在工业控制、交通管理、信息家电、家庭智能化管理、网络及电子商务、环境监测和机器人等方面。ARM 技术的 IP（知识产权）在许多领域已经取得了巨大成功。在 PDA 之类的无线设备中，嵌入式微处理器针对视频流进行了优化，并获得了广泛的支持；在数字音频播放器、数字机顶盒和游戏机等设备中，嵌入式技术得到了广泛的应用；在汽车领域，包括驾驶、安全和车载娱乐等各种功能在内的设备仅用几个嵌入式微处理器就可将要求的功能统一实现。事实上，嵌入式技术无处不在，计算机技术也开始进入一个被称为后 PC 技术的春天。我们不仅拥有那种放在桌上处理文档、进行工作管理和生产控制的计算机“机器”，而且一个普通人也可能拥有从大到小的各种使用嵌入式技术的电子产品，如 MP3、PDA、手机、智能玩具、电子病历、智能血压仪、无线收费设备、超市物流设备、网络家电、智能车载电子设备、安全监控装置、GPS、倒车雷达等。

一些典型的嵌入式系统应用实例如图 1-2 所示。



图 1-2 典型的嵌入式系统应用实例

① 家用方面：数字电视、信息家电、智能玩具、手持通信设备等，如图 1-3 所示。

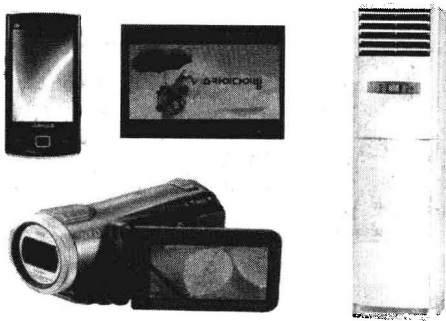


图 1-3 嵌入式在家电方面的应用

② 家庭智能管理系统：社区建筑的水、电、煤气表的远程自动抄表，安全防火、防盗系统，远程点菜器等，如图 1-4 所示。

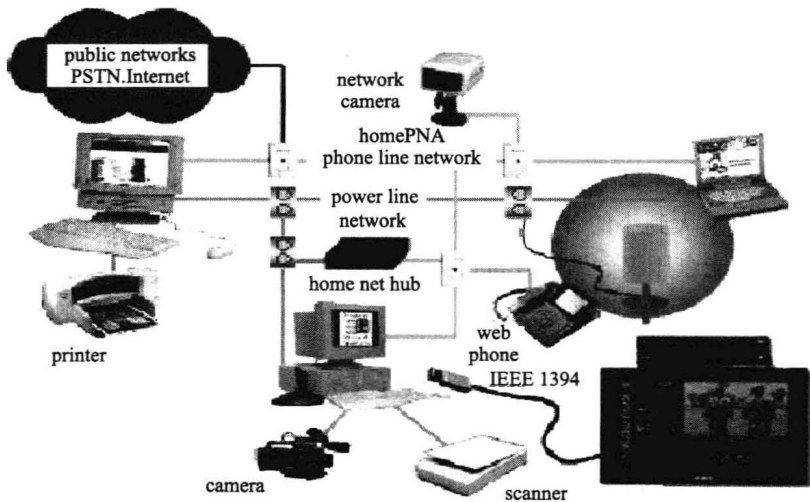


图 1-4 智能家庭管理系统

③ 军事侦察与反恐防暴, 如图 1-5 所示。

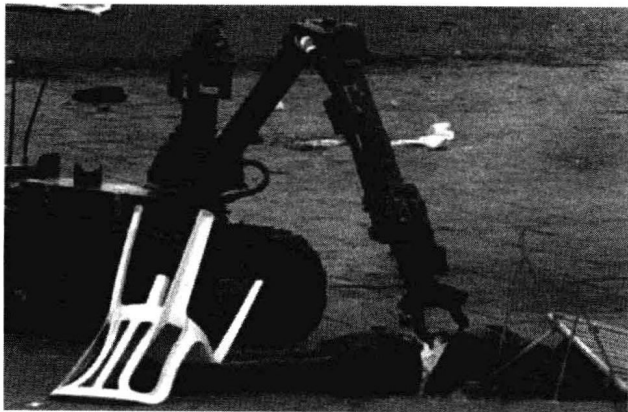


图 1-5 反恐防暴

④ 汽车电子方面：CAN 总线网络汽车电子产品，如图 1-6 所示。

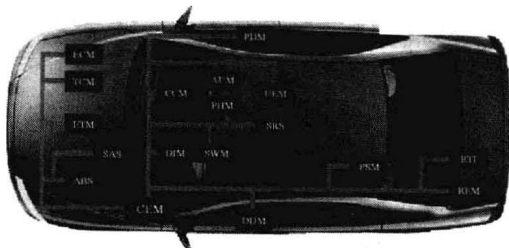


图 1-6 嵌入式汽车控制系统

⑤ 社会发展方面：嵌入式 Internet 应用，如图 1-7 所示。