

高等学校计算机专业规划教材

嵌入式系统开发

基于ARM Cortex A8系统

刘小洋 李勇 编著

*E*mbedded System
Development for
Cortex A8 on GNU/Linux



机械工业出版社
China Machine Press

高等学校计算机专业规划教材

嵌入式系统开发

基于ARM Cortex A8系统

刘小洋 李勇 编著



*Embedded System
Development for
Cortex A8 on GNU/Linux*



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

嵌入式系统开发：基于 ARM Cortex A8 系统 / 刘小洋，李勇编著. —北京：机械工业出版社，2017.12

(高等学校计算机专业规划教材)

ISBN 978-7-111-58357-8

I. 嵌… II. ①刘… ②李… III. 微型计算机—系统开发—高等学校—教材 IV. TP360.21

中国版本图书馆 CIP 数据核字 (2017) 第 262891 号

本书基于 ARM Cortex A8 系统介绍嵌入式系统开发，共 8 章，分为三部分：第一部分（第 1 章和第 2 章）介绍嵌入式系统的基础知识，第二部分（第 3~6 章）介绍嵌入式系统开发环境、嵌入式引导系统、嵌入式操作系统内核、嵌入式文件系统，第三部分（第 7 章和第 8 章）通过嵌入式驱动开发与嵌入式系统项目来论述嵌入式开发的方法论和开发过程。

本书可作为工科类计算机、电子信息、通信工程、自动化等相关专业学生的教材，同时可供嵌入式技术开发人员参考。

出版发行：机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码：100037）

责任编辑：余 洁

责任校对：殷 虹

印 刷：三河市宏图印务有限公司

版 次：2018 年 1 月第 1 版第 1 次印刷

开 本：185mm×260mm 1/16

印 张：17

书 号：ISBN 978-7-111-58357-8

定 价：49.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

客服热线：(010) 88378991 88361066

投稿热线：(010) 88379604

购书热线：(010) 68326294 88379649 68995259

读者信箱：hzjsj@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问：北京大成律师事务所 韩光 / 邹晓东

前 言

嵌入式系统是一种专用的计算机系统，其作为装置或设备的一部分，是现在工业 4.0 架构中的基础设备。嵌入式系统开发是覆盖范围很广的综合性交叉学科，涉及计算机科学与技术、电子科学与技术、自动化、通信工程、电子工程、智能科学与技术等诸多领域，在科技民生、智慧城市、交通运输、物流配送等方面有着广泛的应用前景，是高校工科类学生的首选科目。

全书共 8 章。第一部分共两章：第 1 章对嵌入式系统进行概述，并介绍嵌入式系统组成、嵌入式开源系统相关知识点，给读者一个完整的嵌入式系统概念；第 2 章重点介绍嵌入式 Linux 操作系统的基础知识以及相关操作，为之后的嵌入式学习打下良好基础。第二部分共四章：第 3 章为嵌入式系统开发环境的准备与相关配套工作，其目的是使读者理解嵌入式开发与一般开发的区别所在；第 4 章介绍嵌入式引导系统，基于 Cortex A8 来讲述引导过程的特点与方法，引导系统是嵌入式系统的核心部分；第 5 章介绍嵌入式操作系统内核的移植与相关理论，是本书的重中之重；第 6 章介绍的文件系统是嵌入式系统与普通操作系统区别较大的地方，大家要认真理解。第三部分共两章，通过项目方式来论述嵌入式系统开发的方法论与开发过程。

本书特点

嵌入式系统是集电子、通信、操作系统等多项技术于一体的综合应用。本书在剖析嵌入式体系结构的同时，仔细梳理了嵌入式开发的相关知识点及内在因素。这是作者在近 10 年的教学与工作中得出的相关结论与“教训”。

- 结构清晰，知识完整

全书以嵌入式为主线，按照“从下层到上层，从具体技术到方法论”的思路进行编写，结构清晰，便于读者从宏观上把握嵌入式系统工程的知识内涵。

- 深入浅出，易于理解

本书内容由浅入深，围绕嵌入式所需要的知识点层层论述，同时结合具体操作，避免一切空谈。

- 案例面向实际应用，变抽象为具体

本书中所有的操作与应用都是作者从多年的工作中总结而来的，同时把项目的整个过程按教学要求分解实施，力图向读者展现一幅完整的嵌入式开发画卷。

- 从自然中来，到自然中去

本书的主要目的是将复杂问题用通俗易懂的语言和具体而形象的案例展现给读者，使读者能够从中体会到嵌入式系统开发的整个过程。

本书的编写得到机械工业出版社华章公司多位老师的大力支持与关怀，他们提出了诸多宝贵意见与建议，在此表示感谢。

同时，感谢各高校同行的鼓励与支持，特别是华中科技大学的罗杰老师、广西大学的香赵真老师、湖南大学的王卫平老师、华中科技大学文华学院信息学部的俞侃主任，还有两位研究生付出大量的校对时间，以及对相关数据多次验证并对文档进行整理。

荆楚理工学院的李勇老师负责本教材的校对工作，并多次进行相关内容的调整。

教学建议

本书可作为工科类计算机、电子信息、通信工程、自动化等相关专业学生的教材。

本书安排 48 学时或更多（其中 32 学时为授课学时，16 学时为实验学时），在结束之后可以依据实际情况安排嵌入式系统课程设计课程。

章 节	授 课 学 时	实 验 学 时
第 1 章	2	
第 2 章	4	2
第 3 章	4	2
第 4 章	5	4
第 5 章	5	4
第 6 章	4	2
第 7 章	4	2
第 8 章	4	

授课教师可根据教学计划，灵活调整授课学时。为方便教学，本书提供全部课件。

由于作者水平有限，书中难免存在疏漏之处，敬请读者谅解。如果读者有问题需要与作者讨论，请发送电子邮件到 lxy535@163.com。

刘小洋
于华中科技大学文华园

目 录

前言

第 1 章 嵌入式系统概述 1

1.1 嵌入式系统组成 1

1.1.1 硬件层 2

1.1.2 中间层 4

1.1.3 系统软件层 5

1.2 嵌入式开源系统 6

1.2.1 开源计算项目 6

1.2.2 开源嵌入式开发平台 6

第 2 章 嵌入式 Linux 操作系统 8

2.1 主流的嵌入式操作系统 8

2.1.1 VxWorks 8

2.1.2 Windows Embedded 8

2.1.3 嵌入式 Linux 9

2.1.4 嵌入式实时内核 $\mu\text{C}/\text{OS}$ 9

2.2 嵌入式 Linux 操作系统简介 10

2.3 Linux 操作系统实践 12

2.3.1 Linux 系统 12

2.3.2 基于 VMware 安装 RedHat Linux 系统 13

2.3.3 全屏编辑器与 vi 29

2.3.4 与网络相关的命令 32

2.3.5 软件包的安装与管理 37

第 3 章 嵌入式系统开发环境 47

3.1 Linux 程序设计 47

3.1.1 GNUC 编译器 47

3.1.2 GCC 编译器 48

3.1.3 Makefile 52

3.1.4 用 GDB 调试程序 59

3.2 Linux shell 编程 61

3.2.1 shell 的种类和特点 62

3.2.2 shell 程序与 C 语言 63

3.2.3 shell 脚本的编写 66

3.2.4 shell 与 C 语言的调用 66

3.3 嵌入式开发环境 67

3.3.1 嵌入式 Linux 开发环境搭建 68

3.3.2 交叉编译 70

3.3.3 交叉编译工具的分类和说明 71

3.3.4 宿主机交叉环境建立 71

3.4 基于非操作系统的实践 72

3.4.1 S5PV210 硬件介绍 72

3.4.2 启动方式 73

3.4.3 S5PV210 裸板启动 78

3.4.4 非操作系统的驱动 79

第 4 章 嵌入式引导系统 96

4.1 概述 96

4.1.1 BootLoader 的种类 96

4.1.2 不同平台的开源项目 97

4.2 Linux 系统引导过程与嵌入式引导过程的差别 99

4.2.1 Linux 系统引导过程 99

4.2.2 嵌入式引导过程 103

4.2.3 引导系统启动方式 105

4.2.4 NOR Flash 和 NAND Flash 启动过程的差别 106

4.3 U-Boot 系统的实践 107

4.3.1 U-Boot 的组成 107

4.3.2 定制 S5PV210 配置 110

4.3.3 编译 U-Boot 110

4.3.4 编译过程分析	111	7.1.6 有操作系统时的设备驱动	198
第 5 章 嵌入式操作系统内核	114	7.1.7 内核模块化编程	199
5.1 概述	114	7.2 嵌入式驱动开发实践	207
5.2 嵌入式 Linux 内核实践	123	7.2.1 嵌入式字符设备的驱动程序 结构	207
5.2.1 内核编程	123	7.2.2 设备号的申请和字符设备的 注册	208
5.2.2 嵌入式 Linux 内核移植实践	138	7.2.3 字符设备驱动程序重要的 数据结构	209
5.3 嵌入式 Android 内核移植实践	152	7.2.4 字符设备驱动程序设计	211
5.4 基于 Android 网关的驱动开发	161	7.3 嵌入式驱动开发案例	217
5.4.1 LED 灯控制的 Android 驱动 开发	161	7.3.1 LED 的驱动	217
5.4.2 步进电机实验	163	7.3.2 LED 驱动程序	218
5.4.3 三路继电器实验	166	7.3.3 ADC 转换驱动	223
第 6 章 嵌入式文件系统	168	7.4 嵌入式 Qt 驱动开发案例	228
6.1 概述	168	7.4.1 Qt Creator 简介	228
6.1.1 文件存储结构	168	7.4.2 Qt Creator 的安装和搭建	228
6.1.2 inode 示例	169	7.4.3 驱动程序分析	233
6.1.3 Linux 文件类型	171	7.4.4 LED 蜂鸣器控制驱动案例	234
6.2 嵌入式根文件系统	171	7.4.5 步进电机控制驱动案例	238
6.2.1 基于 Flash 的文件系统	172	7.4.6 继电器控制驱动案例	241
6.2.2 基于 RAM 的文件系统	174	7.4.7 8×7 矩阵键盘驱动案例	244
6.3 嵌入式文件系统实践	175	7.4.8 16×24 点阵屏驱动案例	246
6.3.1 BusyBox 简化嵌入式 Linux 文件系统	175	第 8 章 嵌入式综合项目案例	249
6.3.2 BusyBox 源码分析	175	8.1 开源硬件 pcDuino3 的开发基础	249
6.3.3 基于 S5PV210 内核文件系统 移植	176	8.1.1 通过 VNC 访问 pcDuino3 桌面	249
第 7 章 嵌入式驱动开发	187	8.1.2 基于 pcDuino 的编程	251
7.1 概述	187	8.1.3 pcDuino BSP 的开发	252
7.1.1 嵌入式 Linux 的内核空间与 用户空间	187	8.2 基于 S5PV210 的嵌入式无线路灯 控制系统	259
7.1.2 嵌入式 Linux 的设备管理	188	8.2.1 项目背景	259
7.1.3 嵌入式 Linux 的驱动程序	190	8.2.2 方案介绍	260
7.1.4 嵌入式 Linux 驱动程序的 加载方式	196	8.2.3 功能实现	261
7.1.5 无操作系统时的设备驱动	196	8.2.4 后台控制系统	262
		参考文献	264

第1章 嵌入式系统概述

嵌入式系统一般指的是非 PC 系统, 即有计算机功能但又不能称之为计算机的设备或器材。它是以应用为中心、软硬件可裁减的, 能够适应应用系统对功能、可靠性、成本、体积、功耗等综合性严格要求的专用计算机系统。简单地说, 嵌入式系统集系统的应用软件与硬件于一体, 类似于 PC 中 BIOS 的工作方式, 具有软件代码小、高度自动化、响应速度快等特点, 适合于要求实时性和多任务的体系。

嵌入式系统几乎包括了生活中的所有电器设备, 如掌上 PDA、移动计算设备、电视机顶盒、手机、数字电视、汽车、微波炉、数字相机、家庭自动化系统、电梯、空调、自动售货机、蜂窝式电话、工业自动化仪表与医疗仪器等。随着人们对嵌入式系统的依赖性越来越高, 嵌入式产品将发挥更大的作用。

嵌入式系统一般包含嵌入式微处理器、外围硬件设备、嵌入式操作系统和应用程序四部分。嵌入式领域已经有丰富的软硬资源可以选择, 涵盖了通信、网络、工业控制、消费电子、汽车电子等各种行业。

与通用计算机系统相比, 嵌入式计算机系统具有以下特点:

- 1) 嵌入式系统面向特定的系统应用。嵌入式处理器大多数是专门特定设计的, 具有低功耗、体积小、集成度高等特点。其一般是包含多种外围设备接口的片上系统。
- 2) 嵌入式系统涉及计算机技术、微电子技术、电子技术、通信和软件技术。它是一个技术密集、资金密集、高度分散、不断创新的知识集成系统。
- 3) 嵌入式系统的硬件和软件都必须具备高度可定制性。只有这样才能适合嵌入式系统应用的需要, 在产品性能等诸多方面具有竞争力。
- 4) 嵌入式系统工程的生命周期很长, 当嵌入式系统应用到产品以后, 还需要进行软件升级, 它的生命周期与产品的生命周期几乎一样长。
- 5) 嵌入式系统不具备本地系统开发能力, 通常需要有专门的开发工具和环境。

美国著名的未来学家尼葛洛庞帝在 1997 年访华时曾预言: “4 至 5 年后嵌入式系统将是继 PC 和 Internet 之后的最伟大的发明。”这个预言今天已经成为现实, 现在的嵌入式系统正处于高速发展阶段, 有力证明了它在现实生活中并不可或缺。

1.1 嵌入式系统组成

嵌入式系统装置一般由嵌入式计算机系统和执行装置组成。嵌入式计算机系统是整个嵌入式系统的核心。它由硬件层、中间层、系统软件层和应用软件层组成。执行装置也称为被控对象。它可以接收嵌入式计算机系统发出的控制命令, 执行所规定的操作或任务。执行装置很简单, 可以执行各种复杂的动作并感受各种状态信息。如图 1-1 所示。

嵌入式系统 (Embedded System) 是一种“完全嵌入受控器件内部, 为特定应用而设计的”专用计算机系统。根据英国电气工程师协会 (Institution of Electrical Engineer) 的定义, 嵌入式系统是用于控制、监视或辅助设备、机器或用于工厂运作的设备。与个人计算机这样的通用计算机系统不同, 嵌入式系统通常执行的是带有特定要求的、预先定义的任务。由于嵌入式系统只针对一项特殊的任务, 所以设计人员能够对它进行优化, 减小尺寸, 降低成本。嵌入式系统通常要大量生产, 由于单个成本减小, 所以总体的成本会随着产量成倍下降。

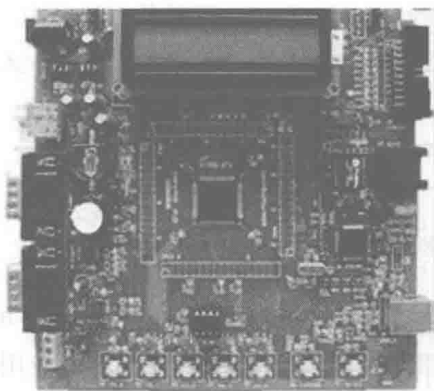


图 1-1 基本的嵌入式系统

通常, 嵌入式系统是一个控制程序存储在 ROM 中的嵌入式处理器控制板。事实上, 所有带有数字接口的设备, 如手表、微波炉、录像机、汽车等, 都是嵌入式系统。有些嵌入式系统还包含操作系统, 但大多数嵌入式系统都是由单个程序实现整个控制逻辑的。

嵌入式系统的核心是由一个或几个预先编程好、用来执行少数几项任务的微处理器或者单片机组成。与通用计算机能够运行用户选择的软件不一样, 嵌入式系统上的软件通常是暂时不变的, 所以经常称为“固件”。

1.1.1 硬件层

嵌入式系统中的硬件层包含了嵌入式微处理器、存储器 (如 SDRAM、ROM、Flash 等)、通用设备接口和 I/O 接口。在一片嵌入式处理器的基础上添加电源电路、时钟电路和存储器电路, 就构成了一个嵌入式核心控制模块。其中操作系统和应用程序都可以固化在 ROM 中。

1. 嵌入式微处理器

嵌入式系统硬件层的核心是嵌入式微处理器, 嵌入式微处理器与通用 CPU 最大的不同在于嵌入式微处理器大多工作在为特定用户群所专门设计的系统中。它将通用 CPU 许多由板卡完成的任务集成在芯片的内部, 从而使得嵌入式系统在设计时趋于小型化, 同时还具有很高的效率和可靠性。

嵌入式微处理器的体系结构可以采用冯·诺依曼或哈佛体系结构, 指令系统可以选用精简指令集系统 (Reduced Instruction Set Computer, RISC) 和复杂指令集系统 (Complex Instruction Set Computer, CISC)。RISC 在通道中只包含最有用的指令, 确保数据通道快速执行每一条指令, 从而提高了执行效率并使 CPU 硬件结构设计变得更为简单。

嵌入式微处理器有各种不同的体系, 即使在同一体系中也具有不同的时钟频率和数据总线宽度, 或者集成了不同的外设和接口。据不完全统计, 目前全世界嵌入式微处理器已经超过 1000 多种, 体系结构有 30 多个系列, 其中主流的体系有 ARM、MIPS、PowerPC、x86 和 SH 等。但与全球 PC 市场不同的是, 没有一种嵌入式微处理器可以主导市场。仅以 32 位的产品而言, 就有 100 种以上的嵌入式微处理器。嵌入式微处理器的选择

是根据具体的应用而决定的。

本书以三星公司生产的、以 ARM 核为主的 S5PV210 CPU 为例来讲述嵌入式系统知识。

2. 存储器

嵌入式系统需要存储器来存放和执行代码。嵌入式系统的存储器包含 Cache、主存储器和辅助存储器。

(1) Cache

Cache 是一种容量小、速度快的存储器阵列。它位于主存和嵌入式微处理器内核之间,存放的是最近一段时间微处理器使用最多的程序代码和数据。在需要进行数据读取操作时,微处理器尽可能地从 Cache 中读取数据,而不是从主存中读取,这大大改善了系统的性能,提高了微处理器和主存之间的数据传输速率。Cache 的主要目标就是:减小存储器(如主存和辅助存储器)给微处理器内核造成的存储器访问瓶颈,使处理速度更快,实时性更强。

在嵌入式系统中,Cache 全部集成在嵌入式微处理器内,可分为数据 Cache、指令 Cache 和混合 Cache,Cache 的大小依不同处理器而定。一般中高档的嵌入式微处理器才会把 Cache 集成进去。

(2) 主存储器

主存储器是嵌入式微处理器能直接访问的寄存器,用来存放系统和用户的程序及数据。它可以位于微处理器的内部或外部,其容量为 256KB~1GB,根据具体的应用而定,一般片内存储器容量小、速度快,片外存储器容量大。

常用的主存储器有:

- ROM 类: PROM、EPROM、NOR Flash、NAND Flash 等。
- RAM 类: SRAM、DRAM 和 SDRAM 等。

其中 NOR Flash 凭借其可擦写次数多、存储速度快、存储容量大、价格便宜等优点,在嵌入式领域内得到了广泛应用。

(3) 辅助存储器

辅助存储器用来存放大数据量的程序代码或信息,它的容量大,但读取速度与主存相比就慢得多,用来长期保存用户的信息。嵌入式系统中常用的辅助存储器有:硬盘、NAND Flash、CF 卡、MMC 和 SD 卡等。

3. 通用设备接口和 I/O 接口

嵌入式系统与外界交互需要一定形式的通用设备接口,如 A/D、D/A、I/O 等。外设通过与片外其他设备进行连接来实现微处理器的输入输出功能。每个外设通常都只有单一的功能,它可以在芯片外也可以内置于芯片内。外设的种类很多,可从一个简单的串行通信设备到非常复杂的 802.11 无线设备。

目前嵌入式系统中常用的通用设备接口有 A/D(模/数转换)接口、D/A(数/模转换)接口,I/O 接口有 RS-232 接口(串行通信接口)、Ethernet 接口(以太网接口)、USB 接口(通用串行总线接口)、音频接口、VGA 视频输出接口、SPI 接口(串行外围设备接口)、IRDA(红

外线接口)和I2C总线(集成电路总线)等。

1.1.2 中间层

硬件层与软件层之间称为中间层,也称为硬件抽象层(Hardware Abstract Layer, HAL)或板级支持包(Board Support Package, BSP)。它将系统上层软件与底层硬件分离开来,使系统的底层驱动程序与硬件无关,上层软件开发人员无需关心底层硬件的具体情况,根据BSP提供的接口即可进行开发。该层一般包含相关底层硬件的初始化、数据的输入输出操作和硬件设备的配置功能。

BSP具有以下两个特点:

- 硬件相关性:因为嵌入式实时系统的硬件环境具有应用相关性,而作为上层软件与硬件平台之间的接口,BSP需要为操作系统提供操作和控制具体硬件的方法。
- 操作系统相关性:不同的操作系统具有各自的软件层次结构,因此,不同的操作系统具有特定的硬件接口形式。

实际上,BSP是一个介于操作系统和底层硬件之间的软件层次,包括了系统中大部分与硬件联系紧密的软件模块。设计一个完整的BSP需要完成两部分工作:一是嵌入式系统硬件初始化以及BSP功能开发,二是设计相关设备驱动过程。

1. 嵌入式系统硬件初始化

系统初始化过程可以分为3个主要环节。按照自底向上、从硬件到软件的次序依次为:片级初始化、板级初始化和系统级初始化。

(1) 片级初始化

完成嵌入式微处理器的初始化,包括设置嵌入式微处理器的核心寄存器和控制寄存器、嵌入式微处理器核心工作模式和嵌入式微处理器的局部总线模式等。片级初始化把嵌入式微处理器从上电时的默认状态逐步设置成系统所要求的工作状态。这是一个纯硬件的初始化过程。

(2) 板级初始化

完成嵌入式微处理器以外的其他硬件设备的初始化。另外,还需设置某些软件的数据结构和参数,为随后的系统级初始化和应用程序的运行建立硬件和软件环境。这是一个同时包含软硬件两部分的初始化过程。

(3) 系统级初始化

该初始化过程以软件初始化为主,主要进行操作系统的初始化。BSP将对嵌入式微处理器的控制权转交给嵌入式操作系统,由操作系统完成余下的初始化操作。这包含加载和初始化与硬件无关的设备驱动程序,建立系统内存区,加载并初始化其他系统软件模块,如网络系统、文件系统等。最后,操作系统创建应用程序环境,并将控制权交给应用程序的入口。

2. 与硬件相关的设备驱动程序

BSP的另一个主要功能是与硬件相关的设备驱动。硬件相关的设备驱动程序的初始化通常是一个从高到低的过程。尽管BSP中包含硬件相关的设备驱动程序,但是这些设备驱动

程序通常不直接被 BSP 使用,而是在系统初始化过程中由 BSP 将它们与操作系统中通用的设备驱动程序关联起来,并在随后的应用中被通用的设备驱动程序调用,实现对硬件设备的操作。与硬件相关的驱动程序是 BSP 设计与开发中另一个非常关键的环节。

1.1.3 系统软件层

系统软件层由实时多任务操作系统 (Real-time Operation System, RTOS)、文件系统、图形用户接口 (Graphic User Interface, GUI)、网络系统及通用组件模块组成。RTOS 是嵌入式应用软件的基础和开发平台。

1. 嵌入式操作系统

嵌入式操作系统 (Embedded Operation System, EOS) 是一种用途广泛的系统软件,过去它主要应用于工业控制和国防系统领域。EOS 负责嵌入系统的全部软硬件资源的分配、任务调度、控制和协调并发活动。它必须体现其所在系统的特征,能够通过装卸某些模块来达到系统所要求的功能。目前已推出一些应用比较成功的 EOS 产品系列。随着 Internet 技术的发展、信息家电的普及应用及 EOS 的微型化和专业化, EOS 开始从单一的弱功能向高专业化的强功能方向发展。嵌入式操作系统在系统实时高效性、硬件的相关依赖性、应用的专用性及软件固化等方面具有较为突出的特点。EOS 是相对于一般操作系统而言的,它除具备了一般操作系统最基本的功能(如任务调度、同步机制、中断处理、文件功能等)外,还有以下特点:

- 1) 可装卸性。具有开放性、可伸缩性的体系结构。
- 2) 强实时性。EOS 实时性一般较强,可用于各种设备的控制当中。
- 3) 统一的接口。提供各种设备驱动接口。
- 4) 操作方便、简单,提供友好的图形界面,追求易学易用。
- 5) 提供强大的网络功能。支持 TCP/IP 协议及其他协议,提供 TCP/UDP/IP/PPP 协议支持及统一的 MAC 访问层接口,为各种移动计算设备预留接口。
- 6) 强稳定性与交互性。嵌入式系统一旦开始运行就不需要用户过多的干预,这就要负责系统管理的 EOS 具有较强的稳定性。嵌入式操作系统的用户接口一般不提供操作命令,它通过系统调用命令向用户程序提供服务。
- 7) 固化代码。在嵌入式系统中,嵌入式操作系统和应用软件被固化在嵌入式系统计算机的 ROM 中。辅助存储器在嵌入式系统中很少使用,因此,嵌入式操作系统的文件管理功能应该能够很容易地拆卸,而用于各种内存文件系统。
- 8) 更好的硬件适应性,也就是良好的移植性。

2. 文件系统

嵌入式文件系统指的是嵌入式系统所应用的文件系统。嵌入式文件系统与我们通常所用的文件系统有较大的区别:我们平时所用的文件系统大致都是相同的,但嵌入式文件系统要为嵌入式系统的设计目的而服务,不同用途的嵌入式操作系统下的文件系统在许多方面各不相同。目前大多数嵌入式系统采用的都是 Linux,而嵌入式 Linux 常用的文件系统有 Ext2fs 第二版扩展文件系统、JFFS 文件系统、YAFFS 文件系统等。

3. 图形用户接口

嵌入式图形用户接口 (Graphic User Interface) 作为嵌入式系统中的一大关键技术, 为用户提供设备的控制接口。当前嵌入式系统中的 GUI 实现方式主要有两种: 一是采用现有的 GUI 库, 二是开发商基于嵌入式操作系统设计的特有的 GUI 系统。采用第一种方式一般要对通用 GUI 库进行剪裁和个性化定制, 往往要支出额外的成本获得软件授权。相对而言, 第二种方式实现的 GUI 占用资源较小, 容易满足嵌入式系统的实时性和个性化需求。

4. 网络系统及通用组件模块

嵌入式技术得到广泛的发展, 已成为现代工业控制、通信类和消费类产品发展的方向。以太网在实时操作、可靠传输、标准统一等方面的卓越性能及其便于安装、维护简单、不受通信距离限制等优点, 已经被国内外很多监控、控制领域的研究人员广泛关注, 并在实际应用中展露出显著的优势。但嵌入式网络系统不同于计算机领域的网络系统, 它有着嵌入式的特点和功能。目前常用的嵌入式网络协议有 LWIP、UIP、TCP/IP 等。

1.2 嵌入式开源系统

开源硬件 (Open Source Hardware) 指与自由及开源软件相同方式设计的计算机和电子硬件。开源硬件是开源文化的一部分, 一般情况下开源硬件会公布详细的硬件设计信息, 包括机械图、电路图、BOM 清单、PCB 版图、HDL 源码、IC 版图和开源软件相关的驱动软件。其中三个比较有代表性的是 Raspberry Pi、BeagleBone 和 Arduino, 前两个可以归为微型计算机一类 (SOC), 而 Arduino 则是功能更弱小的单板机。

1.2.1 开源计算项目

2011 年 4 月, Facebook 建成首个性能最先进的数据服务中心, 同时向全球公开了其服务器和数据中心核心技术, 任何人可以在开源计算项目 (Open Compute Project, OCP) 的网站上看到服务器和数据中心的 CAD 设计图, OCP 由 Facebook 牵头, 与惠普、戴尔、AMD、Intel 公司共同合作。

OCP 已经形成了一个大规模计算处理的生态系统, 有更多的服务器厂商、系统级软件厂商以及各种用户的加入, OCP 可以称为有史以来最大规模的开源项目。

1.2.2 开源嵌入式开发平台

随着移动产品的发展以及 Android 系统的流行, 对于众多的开发者来说, 拥有一款开源硬件的嵌入式开发平台是一个不错的选择。世界各大公司都相继推出并支持多个开源硬件项目, 其中比较著名的有 BeagleBoard、PandBoard、OpenMoke 开源手机项目、ZYNQ-7000、PScC (片上可编程系统)、Arduino 等众多产品。本书以软硬件结合的扩展平台 pcDuino 为例讲述开源项目在嵌入式系统中的重要性。

pcDuino 是一种高性能、高性价比的迷你 PC 平台, 能够运行 PC 操作系统, 如 Ubuntu 和 Android 的 ICS 等。它可以通过内置 HDMI 接口输出视频到电视或显示器屏幕。pcDuino

专门针对开源社区快速增长的需求，即希望有一个平台可以运行完整的 PC 操作系统、容易使用的工具链和兼容流行的 Arduino 开放的生态系统，如 Arduino Shield 和开源项目等。pcDuino 就是 Mini PC+Arduino，最初的版本为 pcDuino V1，配置 1GB 内存、2GB NAND Flash、两个 USB Host 接口。作为 pcDuino 的原型版，该版本的插针在 PCB 的一侧，需要转接板才能连接 Arduino Shield。pcDuino V2 作为 V1 的改进版本，重新修改了 PCB，除了拥有 V1 中的配置以外，在板上继承了无线网卡（当然相对地也少了一个 USB Host 接口），并将扩展接口重新排布，使之能够兼容 Arduino 的接口尺寸，可以直接使用部分 Arduino 的扩展模块，同时这也改善了主板与扩展模块之间的机械连接结构。pcDuino V3 的硬件如图 1-2 所示。

pcDuino 的硬件性能指标远超“树莓派”，性能稳定，做工精良。pcDuino 采用的 CPU 是 1GHz ARM Cortex A8 内核，DRAM 为 1GB，板载存储达到 2GB Flash，完全兼容 Arduino 接口。另外，pcDuino 可以从 NAND 或者从 mini-SD 卡 (TF 卡) 启动，在 NAND Flash 内有出厂预装的 Ubuntu 系统，你完全可以将其当作手机来使用。

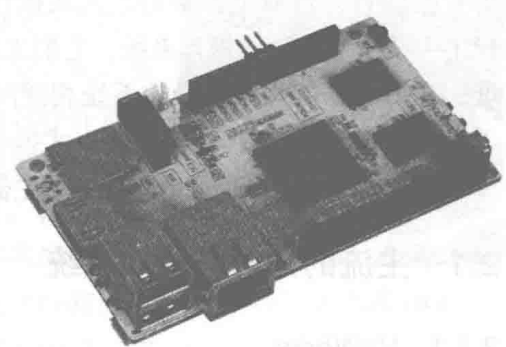


图 1-2 pcDuino V3

第2章 嵌入式Linux操作系统

据调查,目前全世界的嵌入式操作系统已经有200多种。从20世纪80年代开始,出现了一些商用嵌入式操作系统,它们大部分都是为专有系统而开发的。随着嵌入式领域的发展,各种各样的嵌入式操作系统相继问世,包括许多商用嵌入式操作系统和开放源码的嵌入式操作系统。其中著名的嵌入式操作系统有QNX、 μ C/OS、VxWorks、嵌入式Linux和Windows CE等。下面简单介绍一下主流的嵌入式操作系统。

2.1 主流的嵌入式操作系统

2.1.1 VxWorks

VxWorks操作系统是美国WindRiver公司于1983年设计开发的一种嵌入式实时操作系统(RTOS),具有良好的持续发展能力、高性能的内核以及友好的用户开发环境,在嵌入式实时操作系统领域牢牢占据着一席之地。如图2-1所示为VxWorks操作系统启动界面。

VxWorks所具有的显著特点是可靠性、实时性和可裁减性。它支持多种处理器,如x86、i960、Sun Sparc、Motorola MC68xxx、MIPS、POWERPC等。它以其良好的可靠性和卓越的实时性被广泛地应用在通信、军事、航空、航天等高精尖技术及实时性要求极高的领域中,如卫星通信、军事演习、弹道制导、飞机导航等。在美国的F-16/FA-18战斗机、B-2隐形轰炸机和爱国者导弹上,甚至连1997年4月在火星表面登陆的火星探测器、2008年5月登陆的凤凰号和2012年8月登陆的好奇号也都使用了VxWorks。



图2-1 VxWorks操作系统启动界面

2.1.2 Windows Embedded

Windows Embedded(见图2-2)是一种嵌入式操作系统,可以以组件化形式提供Windows操作系统功能。Windows Embedded与Windows一样基于二进制,包含10 000多个独立功能组件,因此开发人员在自定义设备映像中管理或降低内存占用量时可以选择并获得最佳功能。Windows Embedded基于Win32编程模型,由于采用常见开发工具Visual Studio.NET,使用商品化PC硬件,与桌面应用程序无缝集成,因此可以缩短上市时间。使用Windows Embedded构建操作系统的常见设备



图2-2 Windows Embedded图标

类别包括零售销售点终端、客户机和高级机顶盒。

2.1.3 嵌入式 Linux

嵌入式 Linux (见图 2-3) 是以 Linux 为基础的嵌入式操作系统, 它被广泛应用在移动电话、个人数字助理 (PDA)、媒体播放器、消费性电子产品以及航空航天等领域中。嵌入式 Linux 是将日益流行的 Linux 操作系统进行裁剪修改, 使之能在嵌入式计算机系统上运行的一种操作系统。嵌入式 Linux 既继承了 Internet 上无限的开放源代码资源, 又具有嵌入式操作系统的特性。嵌入式 Linux 的特点是: 版权免费, 购买费用、媒介成本、技术支持、全世界的自由软件开发者提供支持的网络特性免费, 而且性能优异, 软件移植容易, 代码开放, 有许多应用软件支持, 应用产品开发周期短, 新产品上市迅速。因为有许多公开的代码可以参考和移植, 如实时性 RT_Linux 等嵌入式 Linux 支持, 实时性能稳定性好, 安全性高。这是我们学习嵌入式重点所在, 在本书中我们会用大量的章节来描述 Linux 与嵌入式 Linux 系统。

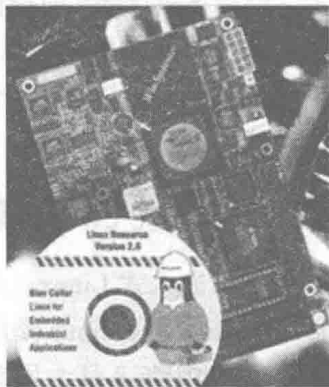


图 2-3 嵌入式 Linux

2.1.4 嵌入式实时内核 μ C/OS

μ C/OS 与嵌入式 Linux 一样, 是一款公开源代码的免费实时内核, 已在各个领域得到了广泛的应用。

μ C/OS 的特点:

- 具有 RTOS 所具有的基本性能。
- 代码尺寸小, 结构简明。
- 易学、易移植。

μ C/OS 提供完整的嵌入式实时内核的源代码, 并对该代码作详尽的解释。而商业上的实时操作系统软件不但价格昂贵 (一般都在 5 千到 2 万美元的价位), 而且其中很多都是所谓黑盒子 (即不提供源代码)。

该源代码的绝大部分是用 C 语言写的, 经过简单地编译就能在 PC 上运行。由于用汇编语言写的部分只有 200 行左右, 该实时内核可以方便地移植到几乎所有的嵌入式应用类 CPU 上。移植范例的源代码可以在因特网上下载。

从最早的实时内核 μ C/OS, 到新版本的 μ C/OS-II, 许多行业上都有成功应用该实时源代码、实时内核移植的实例, 这些应用的实践是该内核实用性和无误性的最好证据。

μ C/OS-II 读为 “microCOS2”, 意为 “微控制器操作系统版本 2”。世界上已有数千人各个领域使用 μ C/OS, 如照相机行业、医疗器械、音响设施、发动机控制、网络设备、高速公路电话系统、自动提款机、工业机器人等。很多高等院校将 μ C/OS 用于实时系统教学。

嵌入式操作系统的选择是前期设计过程的一项重要工作, 这将影响到工程后期的发布以及软件的维护。不管选用什么样的系统, 首先都应该考虑操作系统对硬件的支持, 否则这



图 2-4 嵌入式实时内核 μ C/OS 图标

个系统是不合适的；其次，要考虑的是开发调试时使用的工具，特别是对于开销敏感和技术水平不强的企业来说，开发工具往往在开发过程中起着决定性作用；第三，要考虑系统能否满足应用要求。如果一个操作系统提供的 API 很少，那么无论这个系统有多么稳定，应用层很难进行二次开发，这个显然不是开发人员希望看到的。由此可见，选择一个能满足应用需要、性价比最佳的实时操作系统，对开发工作的意义非常重大。

作为本教材而言，我们选择的是 Linux，一是因为 Linux 是开源的系统，二是 Linux 能够解决我们工作、学习和开发中等大量的相关问题，因此我们采用嵌入式 Linux 作为本教材的系统。

2.2 嵌入式 Linux 操作系统简介

所谓嵌入式 Linux，是指 Linux 在嵌入式系统中的应用，而不是嵌入式功能。实际上，嵌入式 Linux 和 Linux 是同一件事。

在 1991 年 8 月，网络上出现了一篇以“Hello everybody out there using minix”为开篇话语的帖子。这是一个芬兰的名为 Linus Torvalds 的大学生为自己开始写作一个类似 MINIX、可运行在 386 上的操作系统寻找志同道合的合作伙伴。

1991 年 10 月 5 日，Linus Torvalds 在 comp.os.minix 新闻组上发布了大约有一万行代码的 Linux 0.01 版本。

到了 1992 年，大约有 1000 人在使用 Linux。值得一提的是，他们基本上都属于真正意义上的 hacker。

1993 年，大约有 100 余名程序员参与了 Linux 内核代码编写 / 修改工作，其中核心组由 5 人组成，此时 Linux 0.99 的代码大约有十万行，用户大约有 10 万左右。

1994 年 3 月，Linux 1.0 发布，代码量 17 万行，当时是按照完全自由免费的协议发布，随后正式采用 GPL 协议。至此，Linux 的代码开发进入良性循环。很多系统管理员开始在自己的操作系统环境中尝试 Linux，并将修改的代码提交给核心小组。由于拥有了丰富的操作系统平台，Linux 的代码中充实了对不同硬件系统的支持，提高了跨平台移植性。

1995 年，此时的 Linux 可在 Intel、Digital 以及 Sun SPARC 处理器上运行了，用户量也超过了 50 万。

1996 年 6 月，Linux 2.0 内核发布，此内核有大约 40 万行代码，并可以支持多个处理器。此时的 Linux 已经进入了实用阶段，全球大约有 350 万人使用。

1997 年夏，影片《泰坦尼克号》在制作特效中使用的 160 台 Alpha 图形工作站中，有 105 台采用了 Linux 操作系统。

1998 年是 Linux 迅猛发展的一年。1 月，“小红帽”（RedHat）高级研发实验室成立，同年 RedHat 5.0 获得了 InfoWorld 科技网站的“操作系统”奖项。4 月，Mozilla 代码发布，成为 Linux 图形界面上的王牌浏览器。RedHat 宣布商业支持计划，网罗了多名优秀技术人员开始商业运作。“王牌”搜索引擎 Google 现身，采用的也是 Linux 服务器。值得一提的是，Oracle 和 Informix 两家数据库厂商明确表示不支持 Linux，这个决定给予了 MySQL 数据库充分的发展机会。同年 10 月，Intel 和 Netscape 宣布小额投资 RedHat 软件，这被业界视作 Linux 获得商业认同的信号。同月，微软在法国发布了反 Linux 公开信，这表明