

# 嵌入式

## Linux C语言应用开发教程

### 移动学习版 | 第2版

华清远见嵌入式学院 刘洪涛 苗德行 高明旭 刘宗鑫 编著



扫一扫 立即观看配套视频



#### 图书学习 + 在线学习

25个详解视频配合图书同步讲解，  
辅助读者轻松掌握每一个知识点



#### 学习拓展

创客学院拥有海量视频教程 +VIP  
精品课，帮助用户提升开发技能



#### 在线答疑

学习遇到难题，在线提问，辅导  
老师 7 × 12 小时随时答疑

# 嵌入式

## Linux C语言应用开发教程

### 移动学习版 | 第2版

华清远见嵌入式学院 刘洪涛 苗德行 高明旭 刘宗鑫 编著



人民邮电出版社  
北京

## 图书在版编目 (CIP) 数据

嵌入式Linux C语言应用开发教程：移动学习版 / 华清远见嵌入式学院等编著. -- 2版. -- 北京：人民邮电出版社，2018.4  
高等院校嵌入式人才培养规划教材  
ISBN 978-7-115-47226-7

I. ①嵌… II. ①华… III. ①Linux操作系统—程序设计—高等学校—教材②C语言—程序设计—高等学校—教材 IV. ①TP316.89②TP312.8

中国版本图书馆CIP数据核字(2017)第284050号

## 内 容 提 要

本书重点介绍嵌入式Linux应用开发的基本概念和核心理论。全书分为10章，包括嵌入式系统基础、嵌入式Linux C语言程序开发工具、嵌入式Linux C语言基础、嵌入式Linux开发环境的搭建、嵌入式Linux文件I/O编程、嵌入式Linux多任务编程、嵌入式Linux网络编程、嵌入式Linux设备驱动编程、Qt图形编程和综合实例—仓库信息处理系统。本书在讲解中给出了翔实的案例，在部分章节后详细设计并分析了实验内容。

本书可作为高等院校计算机类、电子类、电气类、控制类专业高年级本科生、研究生学习嵌入式Linux应用开发的教材，也可供希望转入嵌入式领域的科研和工程技术人员参考使用。

- 
- ◆ 编 著 华清远见嵌入式学院  
刘洪涛 苗德行 高明旭 刘宗鑫
  - 责任编辑 武恩玉
  - 执行编辑 刘 尉
  - 责任印制 沈 蓉 彭志环
  - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号  
邮编 100164 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
三河市潮河印业有限公司印刷
  - ◆ 开本：787×1092 1/16  
印张：18.5 2018年4月第2版  
字数：487千字 2018年4月河北第1次印刷
- 

定价：55.00元

读者服务热线：(010)81055256 印装质量热线：(010)81055316

反盗版热线：(010)81055315

在社会日益信息化的今天，计算机和网络的应用已经全面渗透到日常生活中。应用嵌入式系统的电子产品随处可见，如人们平常使用的手机、摄像机、医疗仪器、汽车。在发达国家，每个家庭平均拥有 255 个嵌入式系统，如每辆汽车平均装有 35 个嵌入式系统。嵌入式系统的应用广泛进入工业、军事、宇宙、通信、运输、金融、医疗、气象、农业等众多领域。

本书以 ARM Cortex-A9 处理器作为硬件平台，以嵌入式 Linux 为软件平台、以 C 语言为开发语言，介绍了嵌入式系统开发的主要环节。本书重点讲解了嵌入式 Linux 应用开发的基本概念和核心理论，同时结合大量代码实例帮助读者理解和应用相关的概念和理论。突出理论重点，重视实践应用是贯穿本书的理念。

本书三大特色如下。

- (1) 提供丰富案例，搭配优质实验内容。
- (2) 赠送全套教学辅助资源，包括 PPT 课件、源代码、相关文档等。读者可登录人邮教育社区（[www.ryjiaoyu.com](http://www.ryjiaoyu.com)）免费下载。
- (3) 全新升级移动学习版，读者可扫描书中二维码，观看微课视频。

本书共 10 章，下表为本书教学建议学时表：

章节详情	授课学时/共计 64 学时
第 1 章 嵌入式系统基础	2 课时
第 2 章 嵌入式 Linux C 语言程序开发工具	2 课时
第 3 章 嵌入式 Linux C 语言基础	6 课时
第 4 章 嵌入式 Linux 开发环境的搭建	10 课时
第 5 章 嵌入式 Linux 文件 I/O 编程	8 课时
第 6 章 嵌入式 Linux 多任务编程	8 课时
第 7 章 嵌入式 Linux 网络编程	8 课时
第 8 章 嵌入式 Linux 设备驱动编程	12 课时
第 9 章 Qt 图形编程	4 课时
第 10 章 综合实例——仓库信息处理系统	4 课时

本书的出版要感谢华清远见嵌入式培训中心的无私帮助。本书的前期组织和后期审校工作都凝聚了培训中心多位老师的心血，他们认真阅读了书稿，提出了大量中肯的建议，并帮助纠正了书稿中的很多错误。

本书由刘洪涛审定写作提纲，苗德行、高明旭和刘宗鑫编写。另外本书的编写得到了华清远见房焯明、董鹏杰等多名员工的帮助，是他们的帮助使本书得以出版，我们向他们致以诚挚的谢意。由于作者水平所限，书中不妥之处在所难免，恳请读者批评指正。

编 者

2017 年 10 月

创客学院 (www.makeru.com.cn) 是一家 IT 职业在线教育平台, 由国内高端 IT 培训领导品牌华清远见教育集团鼎力打造。学院依托于华清远见教育集团在高端 IT 培训行业积累的十多年教学及研发经验, 以及上百位优秀讲师资源, 专注于为用户提供高端、前沿的 IT 开发技术培训课程。以就业为导向, 以提高开发能力为目标, 努力让每一位用户在这里学到真本领, 为用户成为嵌入式、物联网、智能硬件时代的技术专家助力!

## 一、我们致力于这样的发展理念

我们有一种情怀: 为中国、为世界智能化变革的发展培养更多的优秀人才。

我们有一种坚持: 坚持做专业教育、做良心教育、做受人尊敬的职业教育。

我们有一种变革: 在互联网高速发展的时代, 打造“互联网+教育”模式下的 IT 人才终身学习教学体系。



## 二、我们致力于提供这样的学习方式

### 1. 多元化的课程学习体系

(1) 学习模式的多元化。您可以根据自身的实际情况选择 3 种学习模式，即在线学习、线下报班学习、线上线下结合式学习。每一种模式都有专业的学习路线指导，并有辅导老师悉心答疑，对于学完整套课程的同学有高薪就业职位推荐。

(2) 学习内容的多元化。我们提供基础知识课程、会员提升课程、流行技术精品套餐课程、就业直通车课程、职业成长课程等丰富的课程体系。不管您是职场“小白”，还是 IT 从业人员，都可以在这里找到您的学习路线。

(3) 直播课程的多元化。包括基础类、技术问答类、IT 人的职业素养类、IT 企业的面试技巧类、IT 人的职业发展规划类、智能硬件产品解析类。

### 2. 大数据支撑下的过程化学习模式

(1) 自主学习课程。我们提供习题练习模式支持您的学习，每章学习完成后都有配套的练习题助您检验学习成果，整个课程学习完成后，系统会自动根据您的答题情况，分析出您对课程的整体掌握程度，帮助您随时掌握自身学习情况。

(2) 报班模式下的学习课程。系统会根据您选取的班级，为您制定详细的阶段化学习路线，学习路线采用游戏通关模式，课程章节有考核测验、课程有综合检验、每阶段有项目开发任务。学习过程全程通过大数据进行数据分析，帮助您与班主任随时了解您的课程学习掌握程度，班主任会定期根据您的学习情况开放直播课程，为您的薄弱环节进行细致讲解，考核不合格则无法通过关卡进入下一个环节。

## 三、我们致力于提供这样的服务保障

### 1. 与企业岗位的无缝对接

(1) 在线课程经过企业实体培训检验。华清远见是国内最早的高端 IT 定制培训服务机构，在业界享有盛誉。每年我们都会为不同的企业“量身订制”满足企业需求的高端企业内训课程，曾先后为 Intel、松下、通用电器、摩托罗拉、ST 意法半导体、三星、华为、大唐电信等众多知名企业进行员工内训。

(2) 拥有独立的自主研发中心。研发中心为开发和培训提供技术和产品支持，已经研发多款智能硬件产品、实验平台、实验箱等设备，并与中南大学、中国科学技术大学等高校共建嵌入式、物联网实验室。目前已经公开出版 80 多本教材，深受读者的欢迎。

(3) 平台提供企业招聘通道。学员可在线将自己的学习成果全部展现给企业 HR，增加进入大型企业的机会。众多合作企业定期发布人才需求，还有企业上门招聘，全国 11 大城市就业推荐。

### 2. 丰富的课程资源

创客学院紧跟市场需求，全新录制高质量课程，深入讲解当下热门的开发技术，包括 8 大 IT 职业课程：嵌入式、物联网、Java EE、WEB 全栈、VR/AR、Android、iOS、UI 设计；希望我们的课程能帮您抓住智能硬件时代的发展机遇，打开更广阔的职业发展空间。



### 3. 强大的师资队伍

由华清远见金牌讲师团队+技术开发“大牛”组成的上百人讲师团队，有着丰富的开发与培训经验，其中不乏行业专家和企业项目核心开发者。

### 4. 便捷的学习方式

下载学院 APP 学习，不论您是在学校、家里还是外面，都可以随时随地学习。与教材配套使用，利用碎片时间学习，提升求职就业竞争力！

### 5. 超值的会员福利

会员可免费观看学院 70% 的课程，还可优先参加直播课程、新课程上线抢先试学、学习积分翻倍等活动，并有机会免费参加线下体验课。

## 四、我们期待您的加入

欢迎关注创客学院官网 [www.makeru.com.cn](http://www.makeru.com.cn)，见证我们的成长。期待您的加入，愿与您一起打造未来 IT 人的终身化学习体系。加入创客学院读者 QQ 群 201030910，获得更多资源与服务。

本书配套课程视频观看方法：注册创客学院，手机扫描二维码即可观看课程视频；或在计算机上搜索书名，查找配套课程视频。

# 使用说明

## 第一步：注册创客学院学习账号

使用微信扫描如下二维码，注册（登录）创客学院账号。



## 第二步：“教、学、练”一体化

注册（登录）账号后，即可开始学习。用户可享受 10000+海量视频，VIP 精品课，配套视频教程观看等服务。还可以下载课程相关资料辅助学习，在评论区提问互动，讲师会为你 7×12 小时答疑解惑。



在线视频



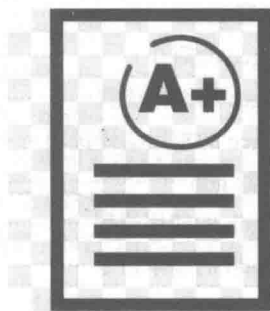
资料下载



讲师答疑

## 第三步：课后习题训练

每个小章节的最后都配备了丰富的课后练习题让读者进行自测，以验证学习效果。自测完成后可扫码加入学习群查看习题答案，还能与其他同学交流沟通，预约讲师 1 对 1 辅导。





<b>第 1 章 嵌入式系统基础</b> .....1	2.5.3 makefile 的规则.....43
1.1 嵌入式系统概述.....1	2.5.4 make 管理器的使用.....44
1.1.1 嵌入式系统的基本概念.....1	2.6 实验内容.....45
1.1.2 嵌入式系统的体系结构.....2	2.6.1 vim 使用练习.....45
1.1.3 几种常用的嵌入式操作系统.....2	2.6.2 用 gdb 调试程序的 bug.....45
1.1.4 嵌入式系统发展趋势.....4	2.6.3 编写包含多文件的 makefile.....46
1.2 ARM 处理器硬件开发平台.....5	思考与练习.....47
1.2.1 嵌入式处理器简介.....5	<b>第 3 章 嵌入式 Linux C 语言基础</b> ...48
1.2.2 ARM 处理器简介.....6	3.1 预处理.....48
1.2.3 Exynos4412 处理器简介.....8	3.1.1 预定义.....48
1.3 嵌入式软件开发流程.....11	3.1.2 文件包含.....53
1.3.1 嵌入式系统开发概述.....11	3.1.3 条件编译.....54
1.3.2 嵌入式软件开发概述.....12	3.2 C 语言中的内存分配.....56
1.4 实验内容：使用 SD-CARD 烧写 EMMC.....16	3.2.1 C 语言程序所占内存分类.....56
思考与练习.....19	3.2.2 堆和栈的区别.....56
<b>第 2 章 嵌入式 Linux C 语言程序 开发工具</b> .....20	3.3 程序的可移植性考虑.....57
2.1 嵌入式 Linux 下 C 语言概述.....20	3.3.1 字长和数据类型.....57
2.2 编辑器 vim.....21	3.3.2 数据对齐.....58
2.2.1 vim 的基本模式.....21	3.3.3 字节顺序.....58
2.2.2 vim 的基本操作.....22	3.4 C 语言和汇编的接口.....59
2.3 编译器 gcc.....24	3.4.1 内嵌汇编的语法.....59
2.3.1 gcc 的简介.....24	3.4.2 编译器优化.....62
2.3.2 gcc 的编译流程.....25	3.4.3 C 语言关键字 volatile.....62
2.3.3 gcc 的常用编译选项.....27	3.5 ARM Linux 内核常见数据结构.....63
2.4 调试器 gdb.....31	3.5.1 链表.....63
2.4.1 gdb 的使用流程.....32	3.5.2 树、二叉树、平衡树.....69
2.4.2 gdb 的基本命令.....34	3.5.3 哈希表.....75
2.4.3 gdbserver 的远程调试.....37	思考与练习.....80
2.5 make 工程管理器.....40	<b>第 4 章 嵌入式 Linux 开发环境的 搭建</b> .....81
2.5.1 makefile 的基本结构.....40	4.1 构建嵌入式 Linux 开发环境.....81
2.5.2 makefile 的变量.....41	4.1.1 嵌入式交叉编译环境搭建.....82
	4.1.2 主机交叉开发环境配置.....83

4.2 Bootloader.....85	6.2.2 Linux 守护进程.....139
4.2.1 Bootloader 的种类.....85	6.3 进程间通信.....143
4.2.2 U-Boot 编译与使用.....86	6.3.1 Linux 下进程间通信概述.....143
4.2.3 U-Boot 移植.....91	6.3.2 管道通信.....144
4.3 Linux 内核与移植.....92	6.3.3 信号通信.....148
4.3.1 Linux 内核结构.....92	6.3.4 信号量.....154
4.3.2 Linux 内核配置与编译.....93	6.3.5 共享内存.....157
4.3.3 设备树文件.....95	6.3.6 消息队列.....159
4.3.4 Linux 内核移植.....96	6.4 多线程编程.....163
4.4 嵌入式文件系统构建.....99	6.4.1 线程基本编程.....163
思考与练习.....101	6.4.2 线程之间的同步与互斥.....166
<b>第5章 嵌入式 Linux 文件</b>	6.4.3 线程属性.....169
<b>I/O 编程.....102</b>	6.4.4 线程私有数据.....170
5.1 Linux 系统调用及用户编程接口.....102	6.5 实验内容.....171
5.1.1 系统调用.....102	6.5.1 编写多进程程序.....171
5.1.2 用户编程接口.....103	6.5.2 编写守护进程.....175
5.1.3 系统命令.....103	6.5.3 有名管道通信实验.....177
5.2 Linux 文件 I/O 系统概述.....103	6.5.4 共享内存实验.....180
5.2.1 虚拟文件系统.....103	6.5.5 线程池实验.....184
5.2.2 通用文件模型.....104	思考与练习.....187
5.2.3 Linux 中文件及文件描述符.....105	<b>第7章 嵌入式 Linux 网络编程.....188</b>
5.3 底层文件 I/O 操作.....106	7.1 TCP/IP 概述.....188
5.3.1 基本文件操作.....106	7.1.1 TCP/IP 的分层模型.....188
5.3.2 文件锁.....108	7.1.2 TCP/IP 分层模型特点.....189
5.3.3 多路复用.....111	7.1.3 TCP/IP 核心协议.....190
5.4 标准 I/O 编程.....116	7.2 网络编程基本知识.....192
5.4.1 基本操作.....117	7.2.1 套接字概述.....192
5.4.2 其他操作.....119	7.2.2 地址及顺序处理.....193
5.4.3 目录操作.....122	7.2.3 套接字编程.....198
5.5 实验内容.....123	7.2.4 编程实例.....202
思考与练习.....128	7.3 网络高级编程.....205
<b>第6章 嵌入式 Linux 多任务编程.....129</b>	7.3.1 非阻塞和异步 I/O.....205
6.1 Linux 下多任务概述.....129	7.3.2 使用多路复用.....209
6.1.1 任务.....129	7.4 实验内容: NTP 的客户端实现.....211
6.1.2 进程.....129	思考与练习.....216
6.1.3 线程.....134	<b>第8章 嵌入式 Linux 设备驱动</b>
6.2 进程控制编程.....135	<b>编程.....217</b>
6.2.1 进程编程基础.....135	8.1 设备驱动编程基础.....217

8.1.1	Linux 设备驱动概述	217
8.1.2	Linux 内核模块编程	219
8.2	字符设备驱动编程	226
8.2.1	字符设备驱动编写流程	226
8.2.2	重要数据结构	227
8.2.3	设备驱动程序主要组成	229
8.2.4	字符设备驱动程序框架	235
8.3	基于设备树的字符驱动程序实例	238
8.4	GPIO 驱动程序实例	242
8.4.1	GPIO 工作原理	242
8.4.2	GPIO 驱动程序	244
8.5	按键驱动程序实例	247
8.5.1	中断编程	247
8.5.2	按键工作原理	248
8.5.3	按键驱动程序	249
8.5.4	中断信息的编写	250
	思考与练习	251
<b>第9章 Qt 图形编程</b>		252
9.1	嵌入式 GUI 简介	252
9.1.1	Qt/Embedded	252
9.1.2	其他嵌入式图形用户界面开发环境	253
9.2	Qt/Embedded 开发入门	254
9.2.1	Qt/Embedded 介绍	254
9.2.2	Qt/Embedded 信号和插槽机制	255

9.2.3	搭建 Qt/Embedded-5.8.0 开发环境	258
9.2.4	Qt/Embedded 窗口部件	262
9.2.5	Qt/Embedded 图形界面编程	265
9.2.6	Qt/Embedded 对话框设计	267
9.3	实验内容: 使用 Qt 编写“Hello, World”程序	270
	思考与练习	275

## 第10章 综合实例——仓库信息处理系统

10.1	仓库信息处理系统概述	276
10.1.1	系统组成	276
10.1.2	前端数据中心 (Cortex-A9)	277
10.1.3	显示中心	277
10.2	基本数据结构	278
10.3	功能实现	280
10.3.1	数据接收模块	280
10.3.2	数据处理模块	281
10.3.3	共享内存刷新模块	283
10.3.4	显示中心	283
10.3.5	线程相关	285
	思考与练习	285

<b>参考文献</b>	286
-------------	-----

# 第 1 章

## 嵌入式系统基础

本章主要介绍嵌入式系统开发的基本知识，学习完本章内容，读者会对嵌入式系统的基础知识、软硬件开发平台以及嵌入式系统开发有一个整体性的理解。

本章主要内容：

- 嵌入式系统概述；
- ARM 处理器开发平台；
- 嵌入式软件开发流程。

### 1.1 嵌入式系统概述

#### 1.1.1 嵌入式系统的基本概念

在社会日益信息化的今天，计算机和网络的应用已经全面渗透到日常生活中。应用嵌入式系统的电子产品随处可见，如日常使用的手机、摄像机、医疗仪器、汽车，乃至工业控制、航天、航空等设备都要用到嵌入式系统。在一些发达国家，平均每个家庭拥有 255 个嵌入式系统，如每辆汽车平均装有 35 个嵌入式系统。嵌入式系统的应用已涉及工业、军事、宇宙、通信、运输、金融、医疗、气象、农业等众多领域。

在嵌入式系统行业内有一个被普遍接受的定义：嵌入式系统是以应用为中心，以计算机控制系统为基础，并且软硬件可裁剪，适用于应用系统对功能、可靠性、成本、体积、功耗有严格要求的专用计算机系统。笔者认为，将一套计算机控制系统嵌入已具有某种完整的特定功能的系统内，以实现对原有系统的计算机控制，此时将这个计算机控制系统叫作嵌入式系统。简单地说，嵌入式系统就是被嵌入电子设备中的专用计算系统。

嵌入式系统通常由特定功能模块和计算机控制模块组成，主要由嵌入式微处理器、外围硬件设备、嵌入式操作系统以及用户应用软件等部分组成。它具有“嵌入性”“专用性”与“计算机系统”三个基本要素。嵌入式系统的特点如下。

(1) 面向特定应用。嵌入式系统与通用型系统的最大区别就在于嵌入式系统大多工作在为特定用户群设计的系统中，因此它通常都具有功耗低、体积小、集成度高等特点，并且可以满足不同应用的特定需求。

(2) 嵌入式系统的硬件和软件都必须进行高效的设计，量体裁衣、去除冗余，力争在同样的硅片面积上实现更高的性能，这样才能在具体应用中对处理器的选择更具有竞争力。

(3) 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后的产物。这一点就决定了它必然是一个技术密集、资金密集、不断创新的知识集成系统,从事嵌入式系统开发的人才也必须是复合型人才。

(4) 为了提高执行速度和系统可靠性,嵌入式系统中的软件一般都固化在存储器芯片中或单片机本身,而不是存储于磁盘中。

(5) 嵌入式开发的软件代码尤其要求高质量、高可靠性,由于嵌入式设备往往是处在无人值守或条件恶劣的环境中,因此,对其代码必须有更高的要求。

(6) 嵌入式系统本身不具备二次开发功能,即设计完成后,用户通常不能在该平台上直接修改程序,必须有一套开发工具和环境才能进行再次开发。

## 1.1.2 嵌入式系统的体系结构

嵌入式系统是一类特殊的计算机系统,一般包括硬件设备、嵌入式操作系统和应用软件。它们之间的关系如图 1.1 所示。

硬件设备包括嵌入式处理器和外围设备。其中的嵌入式处理器(CPU)是嵌入式系统的核心部分,它与通用处理器最大的区别在于,嵌入式处理器大多工作在为特定用户群专门设计的系统中,它将通用处理器中许多由板卡完成的任务集成到芯片内部,从而有利于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。如今,全世界的嵌入式处理器已经超过 1 000 多种,流行的体系结构有 30 多个系列,其中以 ARM、PowerPC、MC 68000、MIPS 等使用得最为广泛。

外围设备是指嵌入式系统中用于完成存储、通信、调试、显示等辅助功能的其他部件。目前常用的嵌入式外围设备按功能可以分为存储设备(如 RAM、SRAM、Flash 等)、通信设备(如 RS-232 接口、SPI 接口、以太网接口、USB 接口、无线通信等)和显示设备(如显示屏等)3 类。

嵌入式操作系统不仅具有通用操作系统的一般功能,如向上提供对用户的接口(如图形界面、库函数 API 等),向下提供与硬件设备交互的接口(硬件驱动程序等),管理复杂的系统资源,同时,它还在系统实时性、硬件依赖性、软件固化性以及应用专用性等方面,具有更加鲜明的特点。

应用软件是针对特定应用领域,基于某一固定的硬件平台,用来达到用户预期目标的计算机软件。嵌入式系统自身的特点,决定了嵌入式应用软件不仅要求满足准确性、实时性、安全性和稳定性等方面需要,而且要尽可能地优化代码,以减少对系统资源的消耗,降低硬件成本。

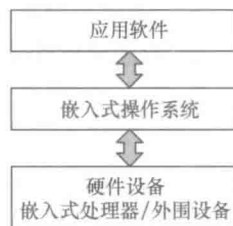


图 1.1 嵌入式系统的组成

## 1.1.3 几种常用的嵌入式操作系统

### 1. 嵌入式 Linux

嵌入式 Linux (Embedded Linux) 是指对标准 Linux 经过小型化裁剪处理之后,能够固化在容量只有几 KB 或者几 MB 的存储器芯片或者单片机中,适合于特定嵌入式应用场合的专用 Linux 操作系统。在目前已经开发成功的嵌入式系统中,大约有一半使用 Linux。这与它自身的优良特性是分不开的。

嵌入式 Linux 同 Linux 一样,具有低成本、多种硬件平台支持、优异的性能和良好的网络支持等优点。另外,为了更好地适应嵌入式领域的开发,嵌入式 Linux 还在 Linux 基础上做了部分改进,介绍如下。

### (1) 改善内核结构

Linux 内核采用的是整体式结构 (Monolithic), 整个内核是一个单独的、非常大的程序, 这样虽然能够使系统的各个部分直接沟通, 提高系统响应速度, 但与嵌入式系统存储容量小、资源有限的特点不相符。因此, 嵌入式系统经常采用的是另一种称为微内核 (Microkernel) 的体系结构, 即内核本身只提供最基本的操作系统功能, 如任务调度、内存管理、中断处理等, 而类似于设备驱动、文件系统和网络协议等附加功能则可以根据实际需要进行取舍。这样就大大减小了内核的体积, 便于维护和移植。

### (2) 提高系统实时性

由于现有的 Linux 是一个通用的操作系统, 虽然它也采用了许多技术来加快系统的运行和响应速度, 但从本质上来说并不是一个嵌入式实时操作系统。因此, 利用 Linux 作为底层操作系统, 在其上进行实时化改造, 从而构建出一个具有实时处理能力的嵌入式系统, 如 RT-Linux 已经成功地应用于航天飞机的空间数据采集、科学仪器测控和电影特技图像处理等各种领域。

嵌入式 Linux 同 Linux 一样, 也有众多的版本, 其中不同的版本分别针对不同的需要在内核等方面加入了特定的机制。嵌入式 Linux 的主要版本如表 1.1 所示。

表 1.1 嵌入式 Linux 的主要版本

版本	简单介绍
$\mu$ CLinux	开放源码的嵌入式 Linux 的典范之作。 $\mu$ CLinux 主要是针对目标处理器没有存储管理单元 MMU (Memory Management Unit) 的嵌入式系统而设计的。由于没有 MMU, 其多任务的实现需要一定技巧。它秉承了标准 Linux 的优良特性, 经过各方面的小型化改造, 形成了一个高度优化的、代码紧凑的嵌入式 Linux。虽然它的体积很小, 却仍然保留了 Linux 的大多数优点: 稳定、良好的移植性、优秀的网络功能、对各种文件系统完备的支持和标准丰富的 API。它专为嵌入式系统做了许多小型化的工作, 目前已支持多款 CPU。其编译后, 目标文件可控制在几百 KB 数量级, 并已经被成功地移植到很多平台上
RT-Linux	由美国墨西哥理工学院开发的嵌入式 Linux 硬实时操作系统。到目前为止, RT-Linux 已经成功地应用于航天飞机的空间数据采集、科学仪器测控和电影特技图像处理等领域。RT-Linux 开发者并没有针对实时操作系统的特性而重写 Linux 的内核, 因为这样做的工作量非常大, 而且要保证兼容性也非常困难。为此, RT-Linux 提出了精巧的内核, 并把标准的 Linux 核心作为实时核心的一个进程, 同用户的实时进程一起调度。这样对 Linux 内核的改动非常小, 并且充分利用了 Linux 下现有的丰富的软件资源
Embedix	根据嵌入式应用系统的特点重新设计的 Linux 发行版本。它提供了超过 25 种的 Linux 系统服务, 包括 Web 服务器等。此外还推出了 Embedix 的开发调试工具包、基于图形界面的浏览器等。可以说, Embedix 是一种完整的嵌入式 Linux 解决方案
XLinux	号称是世界上最小的嵌入式 Linux 系统。采用了“超字节集”专利技术, 使 Linux 内核不仅能与标准字符集相容, 还涵盖了 12 个国家和地区的字符集。因此, XLinux 在推广 Linux 的国际应用方面有独特的优势
红旗嵌入式 Linux	由中科红旗软件技术有限公司推出的嵌入式 Linux, 它是国内做得较好的一款嵌入式操作系统。目前, 中国科学院计算机技术研究所自行开发的开放源码的嵌入式操作系统——Easy Embedded OS (EEOS) 也已经开始进入实用阶段了

为了不失一般性, 本书所用的嵌入式 Linux 是标准内核裁减的 Linux, 而不是表 1.1 中的任何一种。

## 2. $\mu$ C/OS-II

$\mu$ C/OS-II 是一种免费公开源代码、结构小巧、基于优先级的可抢先的硬实时内核。自从 1992

年问世以来,在世界各地应用广泛,它是一种专门为嵌入式设备设计的内核,目前已经被移植到 40 多种不同结构的 CPU 上,运行在 8~64 位的各种系统之上。尤其值得一提的是,该系统自从 2.51 版本之后,就通过了美国 FAA 认证,可以运行在诸如航天器等对安全要求极为苛刻的系统之上。鉴于  $\mu\text{C}/\text{OS-II}$  可以免费获得代码,对于嵌入式 RTOS 而言,选择  $\mu\text{C}/\text{OS}$  无疑是最经济的。

$\mu\text{C}/\text{OS-II}$  主要适合小型实时控制系统,具有执行效率高、占用空间小、实时性能优良和可扩展性强等优点。最小内核可编译至 2KB,如果包含内核的全部功能,编译之后的  $\mu\text{C}/\text{OS-II}$  内核仅有 6KB~10KB。

$\mu\text{C}/\text{OS-II}$  的源代码结构合理、清晰易读,不仅成功应用在众多的商业项目中,而且被很多大学采纳,作为教学的范例,同时也是嵌入式工程师学习和提高的绝好材料。

### 3. VxWorks

VxWorks 操作系统是美国 Wind River 公司于 1983 年设计开发的一种嵌入式实时操作系统 (RTOS),它是当前市场占有率很高的嵌入式操作系统之一。VxWorks 的实时性做得非常好,其系统本身的开销很小,进程调度、进程间通信、中断处理等系统公用程序精练而有效,这使得它们造成的延迟很短。另外 VxWorks 提供的多任务机制,对任务的控制采用了优先级抢占 (Linux 2.6 内核也采用了优先级抢占的机制) 和轮转调度机制,这充分保证了可靠的实时性,并使同样的硬件配置能满足更强的实时性要求。另外 VxWorks 具有高度的可靠性,从而保证了用户工作环境的稳定。同时, VxWorks 还有完备强大的集成开发环境,这也大大方便了用户的使用。

但是, VxWorks 的开发和使用都需要交纳高额的专利费,大大增加了用户的开发成本。同时, VxWorks 的源码不公开造成它部分功能的更新 (如网络功能模块) 滞后。

### 4. Windows CE

Windows CE 是微软开发的一个开放的、可升级的 32 位嵌入式操作系统,是基于掌上型计算机类的电子设备操作系统。Windows CE 的图形用户界面相当出色。Windows CE 具有模块化、结构化和基于 Win32 应用程序接口以及与处理器无关等特点。它不仅继承了传统的 Windows 图形界面,并且用户在 Windows CE 平台上可以使用 Windows 上的编程工具 (如 Visual Studio 等),也可以使用同样的函数和同样的界面风格,这使大多数 Windows 上的应用软件只需简单地修改和移植,就可以在 Windows CE 平台上继续使用。

## 1.1.4 嵌入式系统发展趋势

### 1. 提供强大的网络服务

为适应嵌入式分布处理结构和应用上网需求,面向 21 世纪的嵌入式系统要求配备标准的一种或多种网络通信接口。针对外部联网要求,嵌入设备必须配有通信接口,相应需要 TCP/IP 协议簇软件支持;为满足家用电器相互关联 (如防盗报警、灯光能源控制、影视设备和信息终端交换信息等) 及实验现场仪器的协调工作等要求,新一代嵌入式设备还需具备 IEEE 1394、USB、CAN、Bluetooth 或 IrDA 通信接口,同时也需要提供相应的通信组网协议软件和物理层驱动软件。为了支持应用软件的特定编程模式,如 Web 或无线 Web 编程模式,还需要相应的浏览器,如 HTML 浏览器、WML 浏览器等。

### 2. 小型化、低成本、低功耗

为满足这种特性,要求嵌入式产品设计者相应降低处理器的性能,限制内存容量和复用接口芯片。这就相应提高了对嵌入式软件设计技术的要求,如选用最佳的编程模型和不断改进算法,采用 Java 编程模式,优化编译器性能等。因此,既需要软件人员具有丰富的开发经验,也需要发



展先进的嵌入式软件技术,如 Java、Web 和 WAP 等。

### 3. 人性化的人机界面

用户之所以乐于接受嵌入式设备,其重要因素之一是它们与使用者之间的亲和力。它具有自然的人机交互界面,如司机操纵高度自动化的汽车主要还是通过已使用习惯的方向盘、脚踏板和操纵杆。人们与信息终端交互要求以 GUI 屏幕为主的多媒体界面。手写文字输入、语音拨号上网、收发电子邮件及彩色图形、图像已取得成效。目前一些 PDA 在显示屏幕上已实现汉字写入、短消息语音发布,但离掌式语言同声翻译还有很大距离。

### 4. 完善的开发平台

随着 Internet 技术的成熟、带宽的提高,互联网内容提供商 (Internet Content Provider, ICP) 和应用服务提供商 (Application Service Provider, ASP) 在网上提供的信息内容日趋丰富、应用项目多种多样,像移动电话、固定电话及电冰箱、微波炉等嵌入式电子设备的功能不再单一,电气结构也更为复杂。为了满足应用功能的升级,设计者一方面采用更强大的嵌入式处理器,如 32 位、64 位 RISC 芯片或数字信号处理器 (Digital Signal Processor, DSP) 增强处理能力;同时还采用实时多任务编程技术和交叉开发工具技术来控制功能复杂性,简化应用程序设计、保障软件质量和缩短开发周期。

## 1.2 ARM 处理器硬件开发平台

### 1.2.1 嵌入式处理器简介

嵌入式系统的核心部件是各种类型的嵌入式处理器,据不完全统计,全世界的嵌入式处理器已经超过 1 000 种,流行的体系结构有 30 多个系列,数据总线宽度从 8 位到 32 位,处理速度从 0.1 到 2 000MIPS (MIPS 指每秒执行的百万条指令数)。按功能和内部结构等因素,嵌入式系统硬件平台可以分成如下两类。



嵌入式处理器简介

#### 1. 嵌入式 RISC 微处理器

精简指令集计算机 (Reduced Instruction Set Computer, RISC) 把着眼点放在如何使计算机的结构更加简单和如何使计算机的处理速度更加快速上。RISC 选取了使用频率最高的简单指令,抛弃复杂指令,固定指令长度,减少指令格式和寻址方式,不用或少用微码控制。这些特点使得 RISC 非常适合嵌入式处理器。嵌入式微控制器将整个计算机系统或者一部分集成到一块芯片中。嵌入式微控制器一般以某一种微处理器内核为核心,比如以 MIPS 或 ARM 核为核心,在芯片内部集成 ROM、RAM、内部总线、定时/计数器、看门狗、I/O 端口、串行端口等各种必要的功能和外设。与嵌入式微处理器相比,嵌入式微控制器的最大特点是单片化,实现同样功能时系统的体积大大减小。嵌入式微控制器的品种和数量较多,比较有代表性的通用系列包括 Atmel 公司的 AT91 系列、Samsung 公司的 S3C 系列、Marvell 公司的 PXA 系列等。

#### 2. 嵌入式 CISC 微处理器

CISC 是指复杂指令系统计算机 (Complex Instruction Set Computer),早期的 CPU 全部是 CISC 架构,它的设计目的是要用最少的机器语言指令来完成所需的计算任务。嵌入式 CISC 微处理器是指通用计算机中的 CPU 在不同应用中,将微处理器装配在专门设计的电路板上,只保留和嵌入式应用有关的功能的装置,这种装置可以大幅度减小系统体积和功耗。微处理器 (CPU) 厂商一



直在走 CISC 的发展道路，包括 Intel、AMD，还有其他一些现在已经更名的厂商，如 TI（德州仪器）、Cyrix 以及 VIA（威盛）等。桌面计算机流行的 x86 体系结构即使用的 CISC。RISC 和 CISC 之间的主要区别如表 1.2 所示。

表 1.2 RISC 和 CISC 的主要区别

指标	RISC	CISC
指令集	一个周期执行一条指令，通过简单指令的组合实现复杂操作；指令长度固定	指令长度不固定，执行需要多个周期
流水线	流水线每周期前进一步	指令的执行需要调用微代码的一个微程序
寄存器	更多通用寄存器	用于特定目的的专用寄存器
Load/Store 结构	独立的 Load 和 Store 指令完成数据在寄存器和外部存储器之间的传输	处理器能够直接处理存储器中的数据

## 1.2.2 ARM 处理器简介

ARM（Advanced RISC Machines）有 3 种含义，它是一个公司的名称，是一类微处理器的通称，还是一种技术的名称。

ARM 公司是微处理器行业的一家知名企业，其设计了大量高性能、廉价、低耗能的 RISC 芯片，并开发了相关技术和软件。ARM 处理器具有高性能、低成本和低功耗的特点，适用于嵌入式控制、消费/教育类多媒体、DSP 和移动式应用等领域。

ARM 公司本身不生产芯片，它转让设计许可，由合作伙伴公司来生产各具特色的芯片。ARM 这种商业模式的强大之处在于其价格合理，它在全世界范围的合作伙伴超过 100 个，其中包括许多著名的半导体公司。ARM 公司专注于设计，设计的芯片内核耗电少、成本低、功能强，特有 16/32 位双指令集。ARM 已成为移动通信、手持计算和多媒体数字消费等嵌入式解决方案的 RISC 实际标准。

ARM 处理器的产品分为多个系列，包括 ARM7、ARM9、ARM9E、ARM10E、ARM11 和 SecurCore、Cortex 等。每个系列提供一套特定的性能来满足设计者对功耗、性能、体积的需求。SecurCore 是独立的一个产品系列，是专门为安全设备而设计的。下面简单介绍 ARM 各个系列处理器的特点。

### 1. ARM9 系列处理器

ARM9 系列于 1997 年问世。由于采用了 5 级指令流水线，ARM9 处理器能够运行在比 ARM7 更高的时钟频率上，改善了处理器的整体性能；存储器系统根据哈佛体系结构（程序和数据空间独立的体系结构）重新设计，区分了数据总线和指令总线。

ARM9 系列的第一个处理器是 ARM920T，它包含独立的数据指令 Cache 和 MMU（Memory Management Unit，存储器管理单元）。此处理器能够被用在要求有虚拟存储器支持的操作系统上。该系列中的 ARM922T 是 ARM920T 的变种，只有一半大小的数据指令 Cache。

ARM940T 包含一个更小的数据指令 Cache 和一个 MPU（Micro Processor Unit，微处理器）。它是针对不要求运行操作系统的应用而设计的。ARM920T、ARM940T 都执行 v4T 架构指令。

ARM9 系列处理器主要有以下应用。

- （1）下一代无线设备，包括视频电话和 PDA 等。
- （2）数字消费品，包括机顶盒、家庭网关、MP3 播放器和 MPEG-4 播放器。
- （3）成像设备，包括打印机、数码照相机和数码摄像机。
- （4）汽车、通信和信息系统。