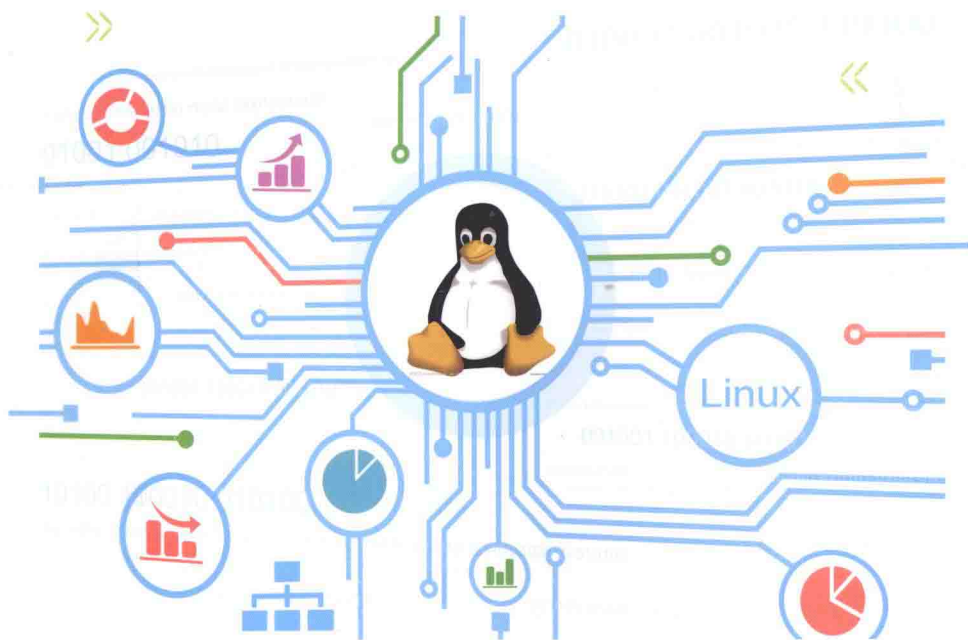


嵌入式Linux系统 开发教程

/ 关注理论知识结构完整性，重视实践前瞻性环节 /
/ 内容基于流行的ARM Cortex-A9处理器+Uboot系统+Linux系统 /
/ 丰富的系统移植和系统构建案例 / 所有实验均可通过FS4412硬件平台验证 /



华清远见嵌入式学院
姜先刚 袁祖刚 编著

嵌入式人才培养规划教材

嵌入式Linux系统 开发教程

华清远见嵌入式学院
姜先刚 袁祖刚 编著



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书结合大量实例,在基于 ARM Cortex-A9 四核处理器的硬件教学平台上,全面详细地讲解了 ARM 处理器及其上的嵌入式 Linux 系统开发。本书主要内容包括 ARM 处理器及常用指令介绍, Linux 编程环境,交叉开发环境, Boot Loader 介绍 U-Boot 启动过程及移植, Linux 内核配置、编译及 Qt 移植等。其中移植相关的内容不仅给出了移植的方法和步骤,还讲解了为什么要这么做。重视实践,实用是本书的最大特点,同时,本书配合大量的习题,读者能快速地掌握嵌入式 Linux 系统开发的要点。

本书可作为大学院校电子、通信、计算机、自动化等专业的“嵌入式 Linux 系统开发”课程的教材,也可作为嵌入式开发人员的参考用书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

嵌入式 Linux 系统开发教程 / 华清远见嵌入式学院, 姜先刚, 袁祖刚编著. —北京: 电子工业出版社, 2016.8
高等院校嵌入式人才培养规划教材
ISBN 978-7-121-29373-3

I. ①嵌… II. ①华… ②姜… ③袁… 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(2016)第 159484 号

责任编辑: 孙学瑛

印 刷: 北京京科印刷有限公司

装 订: 北京京科印刷有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

开 本: 787×1092 1/16 印张: 21 字数: 532.8 千字

版 次: 2016 年 8 月第 1 版

印 次: 2016 年 8 月第 1 次印刷

定 价: 59.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888, 88258888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式: 010-51260888-819 faq@phei.com.cn。

前 言

随着嵌入式及物联网技术的快速发展，ARM 处理器已经广泛地应用到了工业控制、智能仪表、汽车电子、医疗电子、军工电子、网络设备、消费类电子、智能终端等领域。而较新的 ARM Cortex-A9 架构的四核处理器，更是由于其优越的性能被广泛应用在中高端的电子产品市场。比如基于 ARM Cortex-A9 的三星 Exynos 4412 处理器就被应用在三星 GALAXY Note II 智能手机上。

另一方面，Linux 内核由于其高度的稳定性和可裁剪性等特点，被广泛地应用到嵌入式系统中，其中 Android 系统就是一个典型的例子。这样，ARM 处理器和 Linux 操作系统紧密地联系在一起。所以，基于 ARM 和 Linux 的嵌入式系统得到了快速的发展。

目前，针对上述两方面完整涵盖的书籍较少，或者就是相对较老的 ARM 体系结构及较低版本的 Linux 内核。为了能够跟上嵌入式 Linux 技术发展的步伐，促进嵌入式技术的推广，华清远见研发中心自主研发了一套基于 Exynos 4412 处理器的开发板 FS4412，并组织编写了本书。本书注重实践、实用，本着从解决问题的角度出发，既给出了解决问题的方法，又给出了如何逐步解决问题的过程。

本书将 ARM 处理器和嵌入式 Linux 系统融为一体，形成了一套较完整的嵌入式 Linux 系统开发教程。全书共 11 章，循序渐进地讲解了嵌入式 Linux 系统开发所涉及的核心技术和一些经验、方法。本书主要分四个部分，第一部分（第 1 章和第 2 章）介绍了嵌入式系统和 ARM 处理器的概况；第二部分（第 3 章和第 4 章）介绍 Linux 下的软件开发；第三部分（第 5 章）介绍 U-Boot 的移植；第四部分（第 6~11 章）介绍 Linux 内核、驱动的移植，根文件系统的制作及 Qt 的移植。各章节的主要内容如下。

第 1 章对嵌入式系统有一个整体的概述，主要介绍当前比较流行的嵌入式操作系统，并着重介绍嵌入式 Linux 系统。

第 2 章介绍 ARM 处理器的整体情况，主要介绍 ARM 处理器的家族系列、常用的指令、寻址方式和常见的基于 ARM 的 SoC，重点介绍 Exynos 4412 处理器及 FS4412 开发板。

第 3 章讲解常用的 Linux 命令、Shell 脚本、正则表达式、Makefile、GNU 工具集和 Linux 编程库。

第 4 章讲解交叉开发环境，包括交叉编译工具链的安装、串口终端的安装和使用、TFTP 和 NFS 服务器的安装和配置、根文件系统挂载、交叉调试和 FS4412 开发板的开发

环境的搭建实例。

第 5 章讲解常见的 Bootloader、U-Boot 常用命令、U-Boot 的启动过程及 U-Boot 的移植过程。

第 6 章讲解 Linux 内核的配置及编译，包括 Linux 内核源码下载、配置、Makefile、Kconfig、配置选项、编译等内容。

第 7 章讲解 Linux 内核的移植基础内容，包括内核移植的基本工作、Linux 设备树、Linux 启动过程的详细讲解。

第 8 章讲解 FS4412 开发板上 Linux 内核的移植实例，包括内核的基础移植、网卡驱动移植、SD/eMMC 驱动移植、USB 主机控制器驱动移植和 LCD 驱动移植的详细讲解。

第 9 章讲解 Linux 内核的调试技术，包括常用调试方法的介绍、调试相关的配置选项、内核打印函数分析、系统请求键、proc 和 sys 接口、oops 及 panic 信息分析、KGDB 源码级内核调试等。

第 10 章讲解根文件系统的制作和固化，包括根文件系统目录结构、init 系统初始化过程、利用 Busybox 制作根文件系统的过程和根文件系统的固化等。

第 11 章讲解 Qt 的移植，包括 Qt 的下载、配置、编译、安装和在根文件系统中的添加，Qt 集成开发环境的安装，在集成开发环境中加入 ARM 平台的构建环境，Qt 应用程序的编译和在开发板上的运行测试等。

本书由华清远见成都中心的姜先刚和袁祖刚编写，其中第 1~5 章由袁祖刚编写，第 6~11 章由姜先刚编写，北京中心的刘洪涛完成本书的统稿及审校工作。本书的内容是华清远见嵌入式培训中心所有老师心血的结晶，是多年教学成果的积累。他们认真阅读了书稿，提出了大量的建议，并纠正了书稿中的很多错误，在此表示感谢。

由于作者水平有限，书中不妥之处在所难免，恳请读者批评、指正。对于本书的批评和建议，可以发表到 www.farsight.com.cn 技术论坛。

编 者

2016 年 1 月

目 录

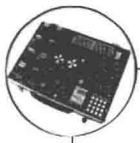
第 1 章 嵌入式系统概述	1
1.1 嵌入式系统	2
1.2 嵌入式操作系统.....	3
1.3 嵌入式 Linux 历史.....	5
1.4 嵌入式 Linux 开发环境.....	6
1.5 嵌入式 Linux 系统开发要点.....	8
第 2 章 ARM 处理	9
2.1 ARM 处理器简介	10
2.1.1 ARM 公司简介.....	11
2.1.2 ARM 处理器体系结构.....	12
2.1.3 Linux 与 ARM 处理器	17
2.2 ARM 指令集	18
2.2.1 ARM 微处理器的指令集概述.....	18
2.2.2 ARM 指令寻址方式.....	19
2.2.3 Thumb 指令	21
2.2.4 Thumb-2 指令.....	22
2.2.5 ThumbEE 指令	22
2.3 典型 ARM 处理器简介	22
2.3.1 Atmel AT91RM9200	22
2.3.2 Samsung S3C2410	23
2.3.3 TI OMAP5 系列.....	25
2.3.4 Freescale i.Max6	26
2.3.5 Intel Xscale PXA 系列.....	27
2.3.6 Cortex-A 系列的 Exynos4412.....	29
2.4 华清远见 FS4412 开发板.....	30
2.4.1 华清远见 FS4412 开发板介绍.....	30

2.4.2 众多的开发板供应商	34
2.5 习题	34
第 3 章 Linux 编程环境	35
3.1 Linux 常用工具	36
3.1.1 Shell 简介	36
3.1.2 常用的 Shell 命令	38
3.1.3 编写 Shell 脚本	46
3.1.4 正则表达式	50
3.1.5 程序编辑器	53
3.2 Makefile	57
3.2.1 GNU make	57
3.2.2 Makefile 规则语法	59
3.2.3 Makefile 文件中变量的使用	60
3.3 二进制代码工具的使用	62
3.3.1 GNU Binutils 工具介绍	62
3.3.2 Binutils 工具软件使用	63
3.4 GCC 编译器的使用	64
3.4.1 GCC 编译器简介	64
3.4.2 GCC 编译选项解析	66
3.5 调试器 GDB 的使用技巧	70
3.5.1 GDB 调试器介绍	70
3.5.2 GDB 调试命令	71
3.6 Linux 编程库	75
3.6.1 Linux 编程库介绍	75
3.6.2 Linux 系统调用	76
3.6.3 Linux 线程库	78
3.7 习题	81
第 4 章 交叉开发环境	82
4.1 交叉开发环境介绍	83
4.1.1 交叉开发概念模型	83
4.1.2 目标板与主机之间的连接	84
4.1.3 文件传输	86

4.1.4	网络文件系统.....	87
4.2	安装交叉编译工具.....	87
4.2.1	获取交叉开发工具链.....	87
4.2.2	主机安装工具链.....	88
4.3	主机开发环境配置.....	88
4.3.1	主机环境配置.....	88
4.3.2	串口控制台工具.....	89
4.3.3	TFTP 服务.....	93
4.3.4	NFS 服务.....	93
4.4	启动目标板.....	95
4.4.1	系统引导过程.....	95
4.4.2	内核解压启动.....	96
4.4.3	挂接根文件系统.....	98
4.5	应用程序的远程交叉调试.....	99
4.5.1	交叉调试的模型.....	99
4.5.2	交叉调试程序实例.....	100
4.6	实例: FS4412 嵌入式开发环境搭建.....	102
4.6.1	开发主机安装.....	102
4.6.2	联调测试.....	110
4.7	习题.....	113
第 5 章	Bootloader.....	114
5.1	Bootloader 简介.....	115
5.1.1	Bootloader 介绍.....	115
5.1.2	Bootloader 的启动.....	115
5.1.3	Bootloader 的种类.....	118
5.2	U-Boot 简介.....	120
5.2.1	U-Boot 介绍.....	120
5.2.2	U-Boot 的常用命令.....	120
5.2.3	U-Boot 的环境变量.....	124
5.2.4	U-Boot 源码结构.....	126
5.3	U-Boot 源码分析.....	127
5.3.1	配置编译.....	127
5.3.2	U-Boot 启动过程.....	132

5.3.3	U-Boot 与内核的关系	136
5.4	U-Boot 移植实例	138
5.4.1	收集移植相关资源	138
5.4.2	选择 U-Boot 源码版本	139
5.4.3	u-Boot 配置编译	139
5.4.4	串口能输出信息	140
5.4.5	移植网卡实现 Ping 和 TFTP 的功能	145
5.5	习题	151
第 6 章	配置编译内核	153
6.1	Linux 内核特点	154
6.1.1	Linux 内核版本介绍	154
6.1.2	Linux 内核特点	155
6.2	配置编译内核源码	156
6.2.1	内核源码的下载方法和结构	157
6.2.2	内核配置系统	160
6.2.3	Kbuild Makefiles	167
6.2.4	内核编译	178
6.2.5	内核编译结果	185
6.3	内核配置选项	187
6.3.1	使用配置菜单	187
6.3.2	基本配置选项	189
6.3.3	驱动程序配置选项	191
6.4	习题	192
第 7 章	内核移植基础	193
7.1	移植内核源码	194
7.1.1	移植的基本工作	194
7.1.2	移植后的工作	196
7.2	Linux 设备树	197
7.2.1	Linux 设备树的由来	197
7.2.2	Linux 设备树的目的	197
7.2.3	Linux 设备树的使用	199
7.3	Linux 内核启动过程分析	206

7.3.1	内核启动流程介绍	206
7.3.2	内核启动加载程序	209
7.3.3	内核主体程序入口	215
7.3.4	Linux 系统初始化	218
7.3.5	初始化驱动模型	224
7.3.6	挂载根文件系统	225
7.4	习题	227
第 8 章	内核移植实例	229
8.1	基本内核移植	230
8.2	网卡驱动移植	234
8.2.1	网卡原理图分析	234
8.2.2	网卡驱动移植	235
8.2.3	以 NFS 挂载根文件系统	237
8.3	SD/eMMC 驱动移植	238
8.4	USB 主机控制器驱动移植	240
8.5	LCD 驱动移植	244
8.6	习题	249
第 9 章	内核调试技术	250
9.1	内核调试方法	251
9.1.1	内核调试介绍	251
9.1.2	学会分析内核源程序	252
9.1.3	调试方法介绍	252
9.2	内核打印函数	255
9.2.1	内核映像解压前的串口输出函数	255
9.2.2	内核映像解压后的串口输出函数	257
9.2.3	printk	258
9.3	获取内核信息	263
9.3.1	系统请求键	263
9.3.2	通过/proc 接口	264
9.3.3	通过/sys 接口	265
9.4	处理出错信息	268
9.4.1	oops 信息	268



嵌入式 Linux 系统开发教程

9.4.2	panic	269
9.4.3	通过 ioctl 方法	271
9.5	内核源代码调试	272
9.6	习题	275
第 10 章	制作 Linux 根文件系统	276
10.1	根文件系统目录结构	277
10.2	init 系统初始化过程	280
10.2.1	inittab 文件	281
10.2.2	System V init 启动过程	284
10.2.3	Busybox init 启动过程分析	285
10.3	制作根文件系统	287
10.3.1	配置并编译 Busybox	287
10.3.2	添加共享库文件和内核模块	289
10.3.3	添加其他目录和文件	290
10.4	固化根文件系统	293
10.4.1	制作 image 格式的 initrd 根文件系统镜像	293
10.4.2	固化 ext2 格式的根文件系统	295
10.5	习题	297
第 11 章	Qt 移植	299
11.1	Qt 源码配置、编译及安装	300
11.2	在根文件系统中添加 Qt	309
11.3	安装 Qt 集成开发环境	309
11.4	添加 ARM 平台的构建环境	313
11.5	编写并运行 Qt 测试程序	316
11.6	习题	320
参考文献	322
附录 习题答案	323

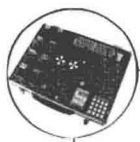
第 1 章 嵌入式系统概述

本章主要介绍嵌入式系统和嵌入式操作系统的概况，讲述嵌入式 Linux 的发展历史和开发环境，概括说明嵌入式 Linux 系统开发的特点。学完本章内容，读者可以对嵌入式 Linux 系统有整体的认识，了解嵌入式 Linux 开发的要点。

本章目标

- 嵌入式系统定义
- 嵌入式操作系统介绍
- 嵌入式 Linux 操作系统
- 嵌入式 Linux 开发环境
- 嵌入式 Linux 系统开发要点





1.1 嵌入式系统



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，适用于应用系统，对功能、可靠性、成本、体积、功耗等方面有特殊要求的专用计算机系统。

嵌入式系统与通用计算机系统的本质区别在于系统应用不同。嵌入式系统是将一个计算机系统嵌入到对象系统中。这个对象既可能是庞大的机器，也可能是小巧的手持设备，用户并不关心这个计算机系统的存在。

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 4 个部分。嵌入式领域已经有丰富的软硬件资源可以选择，涵盖了通信、网络、工业控制、消费电子、汽车电子等各种行业。

嵌入式计算机系统与通用计算机系统相比具有以下特点。

(1) 嵌入式系统是面向特定系统应用的。嵌入式处理器大多数是专门为特定应用设计的，具有低功耗、体积小、集成度高等特点，一般是包含各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、通信和软件等行业。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。只有这样才能适用嵌入式系统应用的需要，在产品价格性能等方面具备竞争力。

(4) 嵌入式系统的生命周期相当长。当嵌入式系统应用到产品以后，还可以进行软件升级，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，通常需要有一套专门的开发工具和环境。

在计算机后 PC 技术时代，嵌入式系统已拥有较大的市场。计算机和网络已经全面渗透到日常生活的每一个角落。各种各样的新型嵌入式系统设备在应用数量上已经远远超过通用计算机，任何一个普通人可能拥有各种使用嵌入式技术的电子产品，小到 MP3、PDA 等微型数字化产品，大到网络家电、智能家电、车载电子设备等。而在工业和服务领域中，使用嵌入式技术的数字机床、智能工具、工业机器人、服务机器人也将逐渐改变传统的工业和服务方式。

美国著名的未来学家尼葛洛庞帝在 1999 年访华时曾预言，四五年后嵌入式系统将是继 PC 和 Internet 之后最伟大的发明之一。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阶段。

目前 BAT（百度、阿里巴巴、腾讯）等在移动互联上激烈拼杀。现实中的穿戴设备、智能手机、物联网和云互联等也广泛运用起来，嵌入式设备、移动端 APP 等技术开发浪潮正汹涌而来。

1.2 嵌入式操作系统



嵌入式操作系统的一个重要特性是实时性。所谓实时性，就是在确定的时间范围内响应某个事件的特性。操作系统的实时性在某些领域是至关重要的，比如工业控制、航空航天等领域。想象飞机正在空中飞行，如果嵌入式系统不能及时响应飞行员的控制指令，那么极有可能导致空难事故。有些嵌入式系统应用并不需要绝对的实时性，比如 PDA 播放音乐，个别音频数据丢失并不影响效果。这可以使用软实时的概念来衡量。

随着越来越多的智能硬件产品的流行，各大企业也纷纷加入到智能产品的开发浪潮中，如阿里巴巴成立智能生活事业部，腾讯推出 TOS+开放平台，百度推出 BaiduInside 计划，京东、小米、360 等公司智能硬件相关业务也在不断完善。从穿戴设备到无人机的流行，再到无人汽车的发展，可以说嵌入式技术的发展迎来新的高潮，目前市场上对于嵌入式开发人才的需求十分火爆。

炙手可热的“智能硬件”是移动互联网与传统制造业的结合产品，目前应用主要集中在智能家居、可穿戴设备、医疗健康以及车联网四大类。对于这个市场，除了大佬之外，互联网创业者们更是趋之若鹜。一大批创客涌现出来，众多创业孵化基地纷纷成立，加上政策的利好，可以预见嵌入式开发前景将是一片光明，未来的生活将会越来越智能。

据调查，目前全世界的嵌入式操作系统已经有两百多种。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，各种各样嵌入式操作系统相继问世。有许多商业的嵌入式操作系统，也有大量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 $\mu\text{C}/\text{OS}$ 、VxWorks、Neculeus、Linux 和 Windows CE 等。下面介绍一些主流的嵌入式操作系统。

(1) Linux: 在所有的操作系统中，Linux 是一个发展快、应用广泛的操作系统。Linux 本身的种种特性使其成为嵌入式开发中的首选。在进入市场的头两年中，嵌入式 Linux 设计通过广泛应用获得了巨大的成功。随着嵌入式 Linux 的成熟，提供更小的尺寸和更多类型的处理器支持，并从早期的试用阶段迈进到嵌入式的主流，它抓住了电子消费类设备开发者们的想象力。随着芯片处理能力的逐步提升，智能手机、平板电脑等设备的广泛运用，Linux 系统成为了嵌入式开发的主流操作系统。

嵌入式 Linux 版本还有多种变体。例如：RTLinux 通过改造内核实现了实时的 Linux；RTAI、Kurt 和 Linux/RK 也提供了实时能力；还有 $\mu\text{C}Linux$ 去掉了 Linux 的 MMU（内存管理单元），能够支持没有 MMU 的处理器等。

(2) $\mu\text{C}/\text{OS}$: $\mu\text{C}/\text{OS}$ 是一个典型的实时操作系统。该系统从 1992 年开始发展，目前流行的是第 2 个版本，即 $\mu\text{C}/\text{OS II}$ 。它的特点是公开源代码、代码结构清晰、注释详尽、组织有条理、可移植性好；可裁剪、可固化；抢占式内核，最多可以管理 60 个任务。自



从清华大学邵贝贝教授将 Jean J. Labrosse 的 *μC/OS-II: the Real Time Kernel* 翻译后，在国内掀起 $\mu\text{C}/\text{OS II}$ 的热潮，特别是在教育研究领域。该系统短小精悍，是研究和学习实时操作系统的首选。

(3) Windows CE: Windows CE 是微软的产品，它是从整体上为有限资源的平台设计的多线程、完整优先权、多任务的操作系统。Windows CE 采用模块化设计，并允许它对于从掌上电脑到专用的工控电子设备进行定制。操作系统的基本内核需要至少 200KB 的 ROM。从 SEGA 的 DreamCast 游戏机到现在大部分的高端掌上电脑都采用了 Windows CE。

随着嵌入式操作系统领域日益激烈的竞争，微软不得不应付来自 Linux 等免费系统的冲击。微软在 Windows CE.Net 4.2 版中，将增加一项授权价仅 3 美元的精简版本 WinCE.Net Core。WinCE.Net Core 具有基本的功能，包括实时 OS 核心 (Real Time OS Kernel)、档案系统；IPv4、IPv6、WLAN、蓝牙等联网功能；Windows Media Codec；.Net 开发框架以及 SQL Server.ce。微软推出低价版本 WinCE.Net，主要是看好语音电话、WLAN 的无线桥接器和个性化视听设备的成长潜力。

(4) VxWorks: VxWorks 是 WindRiver 公司专门为实时嵌入式系统设计开发的操作系统软件，为程序员提供了高效的实时任务调度、中断管理，实时的系统资源以及实时的任务间通信。应用程序员可以将尽可能多的精力放在应用程序本身，而不必再去关心系统资源的管理。该系统主要应用在单板机，数据网络（如以太网交换机、路由器）和通信方面等多方面。其核心功能主要有以下几个。

- 微内核 wind。
- 任务间通信机制。
- 网络支持。
- 文件系统和 I/O 管理。
- POSIX 标准实时扩展。
- C++ 以及其他标准支持。

这些核心功能可以与 WindRiver 系统的其他附件和 Tornado 合作伙伴的产品结合在一起使用。谁都不能否认这是一个非常优秀的实时系统，但其昂贵的价格使不少厂商望而却步。

(5) QNX: 这也是一款实时操作系统，由加拿大 QNX 软件系统有限公司开发，广泛应用于自动化、控制、机器人科学、电信、数据通信、航空航天、计算机网络系统、医疗仪器设备、交通运输、安全防卫系统、POS 机、零售机等任务关键型应用领域。20 世纪 90 年代后期，QNX 系统在高速增长的特网终端设备、信息家电及掌上电脑等领域也得到了广泛应用。

QNX 的体系结构决定了它具有非常好的伸缩性，用户可以把应用程序代码和 QNX 内核直接编译在一起，使之成为简单的嵌入式应用生成一个单一的多线程映像。它也是世界上第一个遵循 POSIX1003.1 标准从零设计的微内核，因此具有非常好的可移植性。

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。首先，不管选用什么样的系统，都应该考虑操作系统对硬件的支持。如果选择的系统不支持将来要使用的硬件平台，那么这个系统是不合适的。其次，考虑开发调试用的工具，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决定性作用；第三，考虑该系统能否满足应用需求。如果一个操作系统提供出来的 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，这显然也不是开发人员希望看到的。由此可见，选择一款既能满足应用需求，性价比又可达到最佳的实时操作系统，对开发工作的顺利开展意义非常重大。

1.3 嵌入式 Linux 历史



所谓嵌入式 Linux，是指 Linux 在嵌入式系统中应用，而不是什么嵌入式功能。实际上，嵌入式 Linux 和 Linux 是同一件事。

我们了解一下 Linux 的发展历史。

Linux 起源于 1991 年，由芬兰的 Linus Torvalds 开发，随后按照 GPL 原则发布。Linux 1.0 正式发行于 1994 年 3 月，仅支持 386 的单处理器系统。

Linux 1.2 发行于 1995 年 3 月，它是第一个包含多平台（如 Alpha、Sparc、Mips 等）支持的官方版本。

Linux 2.0 发行于 1996 年 6 月，包含很多新的平台支持。最重要的是，它是第一个支持 SMP（对称多处理器）体系的内核版本。

Linux 2.2 于 1999 年 1 月发布，它带来了 SMP 系统上性能的极大提升，同时支持更多的硬件。

Linux 2.4 于 2001 年 1 月发布，它进一步提升了 SMP 系统的扩展性，同时它也集成了很多用于支持桌面系统的特性，如 USB、PC 卡（PCMCIA）的支持，内置的即插即用等。

Linux 2.6 于 2003 年 12 月发布，它的多种内核机制都有了重大改进，无论对大系统还是小系统（PDA 等）的支持都有很大提高。

Linux 3.14 版本开始，ARM 正式引入了设备树机制，内核的易维护性得到很大提升。最新的 Linux 内核版本可以从官方网站获取。

<http://www.kernel.org>

Linux 是一种类 UNIX 操作系统。从绝对意义上讲，Linux 是 Linus Torvalds 维护的内核。现在的 Linux 操作系统已经包括内核和大量应用程序，这些软件大部分来源于 GNU 软件工程。因此，Linux 又叫做 GNU/Linux。

目前 Linux 操作系统的发行版已经有很多，例如：Redhat Linux、Suse Linux、Turbo

Linux 等台式机或者服务器版本, 还有各种嵌入式 Linux 版本。不同的 Linux 版本之间总会有些差异。鉴于 UNIX 技术历史的教训, LSB (Linux Standard Base) 为 Linux 系统制定了规范。LSB 规范定义了几种模块, 并且为应用程序定义系统接口和基本配置, 为大量的应用程序提供了统一的行业标准。读者可从以下站点可以获取 LSB 的文档。

<http://www.linuxbase.org>

ELC (Embedded Linux Consortium, 嵌入式 Linux 联盟) 是一个非营利性的、中立的行业协会。它的目标是在嵌入式应用和设备计算市场做 Linux 的改进、推广和标准化工作。ELC 成员贡献会费, 并且参与管理、推广、实现和平台规范工作组的维护, 谋求不断增长的市场机遇。ELC 成员为了 API 的互用性积极推广了一套平台标准, 消除分割并且发布更加具有竞争力的商业方案。

<http://www.embedded-linux.org>

OSDL (Open Source Development Labs, 开放源码开发实验室) 支持围绕 Linux 开发和指导的各种活动。它为 OSDL 协会免费提供硬件资源。OSDL 发起了电信 Linux (Carrier Grade Linux) 和数据中心 Linux (Data Center Linux) 工作组。这些工作组包含 OSDL 成员和有兴趣的个人, 他们致力于创建特点列表和规范, 并且参与开源工程为电信和数据中心进一步开发 Linux。OSDL 还积极参与内核测试, 提供了开放的测试环境 (Scalable Test Platform), 并且贡献给开发状态的内核测试。

<http://www.osdl.org>

CELF (Consumer Electronics Linux Forum, 消费电子 Linux 论坛) 是加州的一个非营利性公司, 它致力于把 Linux 改进成消费电子设备的开放平台。

<http://www.celinuxforum.org>

越来越多的个人、社团和公司已经或正在参与 Linux 社区的工作, 他们为 Linux 系统开发、测试以及应用作出了大量贡献。这使得嵌入式 Linux 系统成为标准化的操作系统, 功能日趋完善, 应用更加广泛。

1.4 嵌入式 Linux 开发环境



通用计算机可以直接安装发行版的 Linux 操作系统, 使用编辑器、编译器等工具为本机开发软件, 甚至可以完成整个 Linux 系统的升级。

嵌入式系统的硬件一般有很大的局限性, 或者处理器频率很低, 或者存储空间很小, 或者没有键盘、鼠标设备。这样的硬件平台无法胜任 (或者不便于) 庞大的 Linux 系统开发任务。因此, 开发者提出了交叉开发环境模型。

交叉开发环境是由开发主机和目标板两套计算机系统构成的。目标板 Linux 软件是