

嵌入式 Linux

系统开发技术详解 ——

基于 ARM

孙纪坤 张小全 编著

北京华清远见嵌入式培训中心 审校

嵌入式 Linux 系统开发技术详解—— 基于ARM

孙纪坤 张小全 编著

北京华清远见嵌入式培训中心 审校

 人民邮电出版社
POSTS & TELECOM PRESS

图书在版编目 (CIP) 数据

嵌入式 Linux 系统开发技术详解: 基于 ARM / 孙纪坤, 张小全编著.
—北京: 人民邮电出版社, 2006.8

ISBN 7-115-15014-1

I. 嵌... II. ①孙... ②张... III. ①Linux 操作系统 ②微处理器, ARM—系统设计
IV. ①TP316.89 ②TP332

中国版本图书馆 CIP 数据核字 (2006) 第 078284 号

内 容 提 要

本书以嵌入式 Linux 系统开发流程为主线, 剖析了嵌入式 Linux 系统构建的各个环节。本书从嵌入式系统基础知识和 Linux 编程技术讲起, 接下来说明了嵌入式 Linux 交叉开发环境的建立, 然后分析了嵌入式 Linux 系统的引导程序、内核和文件系统三大组成部分, 最后介绍了嵌入式 Linux 系统集成和部署的方法。本书使用的嵌入式 Linux 系统软件全部来源于开源站点, 文中提供了具体的链接地址。本书主要以 ARM 平台为例, 对 U-Boot 和 Linux 内核启动过程做了详细分析, 为学习嵌入式 Linux 系统开发奠定基础。本书从概念上阐述了嵌入式 Linux 系统开发流程, 实践上提供了具体的操作步骤, 使读者能够深入理解嵌入式 Linux 系统构建。

本书可作为高等院校电子类、电气类、控制类专业高年级本科生、研究生学习嵌入式 Linux 的教材, 也可供广大希望转入嵌入式领域的科研和工程技术人员参考使用, 还可作为广大嵌入式培训班的教材和教辅材料。

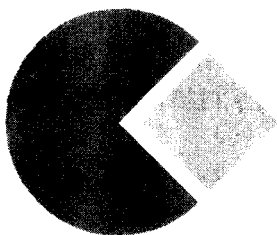
嵌入式 Linux 系统开发技术详解——基于 ARM

- ◆ 编 著 孙纪坤 张小全
- ◆ 审 校 北京华清远见嵌入式培训中心
责任编辑 屈艳莲
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
人民邮电出版社河北印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 24
字数: 582 千字 2006 年 8 月第 1 版
印数: 1—5 000 册 2006 年 8 月河北第 1 次印刷

ISBN 7-115-15014-1/TP · 5564

定价: 39.00 元

读者服务热线: (010) 67132692 印装质量热线: (010) 67129223



前 言

随着 Linux 操作系统的发展，特别是 Linux 2.6 内核的迅速发展，嵌入式 Linux 在嵌入式领域的应用越来越广泛。Linux 具备源码开放、内核稳定高效、软件丰富等优势，而且还具备支持广泛的处理器结构和硬件平台、可定制性好、可靠性高等特点。据 IDC 的报告显示，嵌入式 Linux 在未来两年将占嵌入式操作系统市场份额的 50%，约 3.5 亿美元，由此产生的应用市场前景更是不可估量。

正是由于市场的需求，嵌入式领域也需要大量的嵌入式 Linux 开发者。目前国内 Linux 程序员的素质和数量还不能满足企业的需要。

编写目的

大学计算机相关专业课程都已经包含计算机组成原理、计算机编程语言、计算机体系结构、计算机操作系统，甚至还包括电子技术和半导体技术。尽管已经具备这些嵌入式 Linux 系统的基础知识，但是多数大学毕业生不清楚到底该如何开发嵌入式 Linux 系统。

编写本书的目的就是阐述嵌入式 Linux 系统的各组成部分，从概念上和实践上说明嵌入式 Linux 系统开发的基本过程。这本书可以帮助具备计算机基础知识的开发者迅速进入嵌入式系统开发领域。

希望本书能够帮助读者更好地理解嵌入式 Linux 系统，并且参与到嵌入式 Linux 系统开发中来。

主要内容

本书以嵌入式 Linux 系统开发流程为主线，剖析了嵌入式 Linux 系统构建的各个环节。从嵌入式系统基础知识和 Linux 编程技术讲起，接下来说明了建立嵌入式 Linux 交叉开发环境，然后分析了嵌入式 Linux 系统的引导程序、内核和文件系统三大组成部分，最后介绍了嵌入式 Linux 系统集成和部署的方法。

第 1 章介绍了嵌入式系统和嵌入式操作系统的概况，讲述了嵌入式 Linux 发展历史和开发环境，概括说明了嵌入式 Linux 系统开发的特点。

第 2 章描述了 ARM 体系结构和 ARM Linux 的发展，介绍了几种应用 Linux 的典型 ARM 处理器和开发板。

第 3 章介绍了 Linux 编程常用的工具，Makefile 语法规则，还有 binutils、gcc 和 gdb 等工具的使用。

第 4 章介绍了嵌入式交叉开发环境的概念和配置，说明了应用程序交叉开发和调试的基本方法。

第 5 章介绍了编译生成 GNU 工具链的基本步骤。

第 6 章介绍了 Bootloader 的类型的特点，详细分析了 U-Boot 的使用、编译和移植。

第 7 章介绍了 Linux 2.6 内核的特点和 Kbuild 管理方式，说明了内核基本的配置选项的用法。

第 8 章以 ARM 平台为例介绍了内核移植的基本方法，并且详细分析了 Linux 内核启动过程。

第 9 章介绍了各种 Linux 内核调试方法，为内核移植提供了有效的调试手段。

第 10 章介绍了 Linux 根文件系统的组织结构，并且分析了 init 进程调用文件系统脚本初始化的过程。

第 11 章介绍了嵌入式 Linux 系统常用的开源软件，包括系统工具、图形库、网络和串口应用程序等。

第 12 章介绍了系统集成测试需要的各种工具，主要包括系统跟踪、性能测试和内存测试 3 个方面。

第 13 章介绍了 Linux 系统部署的基本方法，分析了文件系统和存储介质的特点。

第 14 章介绍了以 S3C2410 处理器的 GPS 手持设备开发过程为例，介绍了嵌入式 Linux 系统软硬件的设计与开发。

本书可作为高等院校电子类、电气类、控制类专业高年级本科生、研究生学习嵌入式 Linux 的教材，也可供广大希望转入嵌入式领域科研和工程技术人员参考使用，还可作为广大嵌入式 Linux 就业培训班的教材和教辅材料。

阅读建议

根据本书的指导，可以自己动手构建嵌入式 Linux 开发环境和嵌入式 Linux 系统。这对于深刻理解和掌握嵌入式 Linux 开发是非常重要的。

嵌入式的开发与具体的硬件环境紧密相关，本书的内容以常见的 ARM 9 S3C2410 平台为例来讲解。对于其他硬件平台可以触类旁通，通过分析具体的源代码学习。

感谢

感谢张小全的密切合作，他的努力使得本书得以及时完稿，他撰写了第 2 章、第 3 章、第 10 章、第 11 章。

感谢开放源码软件和开放文档的作者们。

感谢华清远见的季久峰老师。

感谢麦克泰公司和我的同事们，特别是支持我的龙中花和易松华。

另外还要感谢以下人员的支持：孙天泽、袁文菊、田彦、周明、黄昕、史宜彬、张秀丽、谭翠君等。

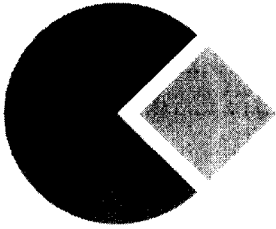
相关内容

本书内容来自北京华清远见科技信息有限公司 (<http://www.farsight.com.cn>) 的培训课程资料,有关本书的相关资料和嵌入式 Linux 更多的资料、公开课视频,请参见 <http://www.farsight.com.cn/download/>。

由于时间仓促,加之水平有限,书中的不足之处在所难免,敬请读者批评指正。本书责任编辑的联系方法是 quyanlian2@ptpress.com.cn, 欢迎来信交流。

编者

2006年6月



目 录

第 1 章 概述	1
1.1 嵌入式系统	1
1.2 嵌入式操作系统	2
1.3 嵌入式 Linux 历史	4
1.4 嵌入式 Linux 开发环境	5
1.5 嵌入式 Linux 系统开发要点	7
第 2 章 ARM 处理器	8
2.1 ARM 处理器简介	8
2.1.1 ARM 公司简介	9
2.1.2 ARM 处理器体系结构	10
2.1.3 Linux 与 ARM 处理器	12
2.2 ARM 指令集	13
2.2.1 ARM 微处理器的指令集概述	13
2.2.2 ARM 指令寻址方式	15
2.2.3 Thumb 指令概述	17
2.3 典型 ARM 处理器简介	17
2.3.1 Atmel AT91RM9200	17
2.3.2 Samsung S3C2410	18
2.3.3 TI OMAP1510/1610 系列	19
2.3.4 Freescale i.Max21	22
2.3.5 Intel Xscale PXA 系列	23
2.4 三星 S3C2410 开发板	24
2.4.1 三星 S3C2410 开发板介绍	24
2.4.2 众多的开发板供应商	26
第 3 章 Linux 编程环境	28

3.1 Linux 常用工具	28
3.1.1 Shell 简介	28
3.1.2 常用 Shell 命令	30
3.1.3 编写 Shell 脚本	38
3.1.4 正则表达式	42
3.1.5 程序编辑器	44
3.2 Makefile	48
3.2.1 GNU make	48
3.2.2 Makefile 规则语法	49
3.2.3 Makefile 文件中变量的使用	51
3.3 二进制代码工具的使用	52
3.3.1 GNU Binutils 工具介绍	52
3.3.2 Binutils 工具软件使用	54
3.4 编译器 GCC 的使用	54
3.4.1 GCC 编译器介绍	54
3.4.2 GCC 编译选项解析	56
3.5 调试器 GDB 的使用技巧	60
3.5.1 GDB 调试器介绍	60
3.5.2 GDB 调试命令	61
3.6 Linux 编程库	66
3.6.1 Linux 编程库介绍	66
3.6.2 Linux 系统调用	67
3.6.3 Linux 线程库	70
第 4 章 交叉开发环境	73
4.1 交叉开发环境介绍	73
4.1.1 交叉开发概念模型	73
4.1.2 目标板与主机之间的连接	75
4.1.3 文件传输	76
4.1.4 网络文件系统	77
4.2 安装交叉编译工具	78
4.2.1 获取交叉开发工具链	78
4.2.2 主机安装工具链	79
4.3 主机开发环境配置	80
4.3.1 主机环境配置	80
4.3.2 串口控制台工具	81
4.3.3 DHCP 服务	84
4.3.4 TFTP 服务	85
4.3.5 NFS 服务	86

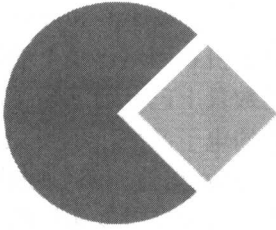
4.4	启动目标板	88
4.4.1	系统引导过程	88
4.4.2	内核解压启动	89
4.4.3	挂接根文件系统	90
4.5	应用程序的远程交叉调试	91
4.5.1	交叉调试的模型	91
4.5.2	交叉调试程序实例	92
第 5 章	交叉开发工具链	94
5.1	工具链软件	94
5.1.1	相关软件工程	94
5.1.2	软件版本的匹配	95
5.1.3	工具链制作流程	97
5.2	制作交叉编译器	98
5.2.1	准备编译环境	98
5.2.2	编译 binutils	99
5.2.3	编译 GCC 的辅助编译器	100
5.2.4	编译生成 glibc 库	101
5.2.5	编译生成完整的 GCC 编译器	105
5.3	制作交叉调试器	106
5.3.1	编译交叉调试器	106
5.3.2	编译 gdbserver	106
第 6 章	Bootloader	108
6.1	Bootloader	108
6.1.1	Bootloader 介绍	108
6.1.2	Bootloader 的启动	109
6.1.3	Bootloader 的种类	111
6.2	U-Boot 编程	113
6.2.1	U-Boot 工程简介	113
6.2.2	U-Boot 源码结构	113
6.2.3	U-Boot 的编译	114
6.2.4	U-Boot 的移植	117
6.2.5	添加 U-Boot 命令	118
6.3	U-Boot 的调试	121
6.3.1	硬件调试器	121
6.3.2	软件跟踪	121
6.3.3	U-Boot 启动过程	123
6.3.4	U-Boot 与内核的关系	128

6.4	使用 U-Boot	133
6.4.1	烧写 U-Boot 到 Flash	134
6.4.2	U-Boot 的常用命令	134
6.4.3	U-Boot 的环境变量	141
第 7 章	配置编译内核	143
7.1	Linux 内核特点	143
7.1.1	Linux 内核版本介绍	143
7.1.2	Linux 内核特点	143
7.1.3	Linux 2.6 内核新特性	144
7.2	配置编译内核源码	147
7.2.1	内核源码结构	148
7.2.2	内核配置系统	150
7.2.3	Kbuild Makefile	157
7.2.4	内核编译	169
7.2.5	内核编译结果	179
7.3	内核配置选项	180
7.3.1	使用配置菜单	180
7.3.2	基本配置选项	181
7.3.3	驱动程序配置选项	183
第 8 章	内核移植浅析	185
8.1	移植内核源码	185
8.1.1	移植前的准备工作	185
8.1.2	开发板内核移植	186
8.1.3	移植后的工作	194
8.2	Linux 内核启动过程分析	194
8.2.1	内核启动流程源代码分析	194
8.2.2	内核自引导程序	195
8.2.3	内核 vmlinux 入口	199
8.2.4	Linux 系统初始化	201
8.2.5	挂接根文件系统	206
8.2.6	初始化设备驱动	208
8.2.7	启动用户空间 init 进程	209
第 9 章	内核调试技术	212
9.1	内核调试方法	212
9.1.1	内核调试概述	212
9.1.2	学会分析内核源程序	213

9.1.3 调试方法介绍	213
9.2 内核打印函数	216
9.2.1 内核映像解压前的串口输出函数	216
9.2.2 内核错误报告子程序	218
9.2.3 内核打印函数	220
9.3 获取内核信息	227
9.3.1 系统请求键	227
9.3.2 通过/proc 接口	228
9.3.3 通过/sys 接口	229
9.3.4 通过 ioctl 方法	232
9.4 处理出错信息	233
9.4.1 oops 信息	233
9.4.2 panic	234
9.5 内核源码调试	236
9.5.1 KGDB 调试内核源代码	236
9.5.2 BDI2000 调试内核源代码	237
第 10 章 制作 Linux 根文件系统	242
10.1 根文件系统目录结构	242
10.1.1 FHS 目录结构	243
10.1.2 文件存放规则	246
10.2 添加系统文件	247
10.2.1 添加共享链接库	247
10.2.2 添加内核模块	249
10.2.3 添加设备文件	251
10.3 init 系统初始化过程	253
10.3.1 inittab 文件	253
10.3.2 System V init 启动过程	256
10.3.3 Busybox init 启动过程分析	258
10.4 定制文件系统	260
10.4.1 定制应用程序	260
10.4.2 配置应用程序自动启动	260
第 11 章 充分利用开源软件	262
11.1 开放源代码工程介绍	262
11.1.1 Linux 系统和开源软件	263
11.1.2 开源软件的特点	264
11.2 Busybox 使用	265
11.2.1 Busybox 工程介绍	265

11.2.2	配置编译 Busybox	265
11.3	X11 图形系统	270
11.3.1	X Windows 介绍	270
11.3.2	Tiny-X 介绍	270
11.3.3	GTK 图形库	271
11.4	Qt 图形库	277
11.4.1	Qt 介绍	277
11.4.2	Qt/Embedded 介绍	278
11.4.3	Qt/Embedded 架构	280
11.4.4	Qt/Embedded 软件包与安装	281
11.5	MiniGUI 图形系统	283
11.5.1	MiniGUI 图形系统概述	283
11.5.2	MiniGUI 移植	285
11.6	MicroWindows 图形系统	290
11.7	Linux 下的网络应用	292
11.7.1	嵌入式设备的网络化	292
11.7.2	TCP/IP 协议概述	292
11.7.3	Linux 下的 Socket 编程	294
11.8	嵌入式 Linux 的串行通信	304
11.8.1	Linux 下的串口操作	304
11.8.2	Linux 串口编程实例	309
第 12 章	系统集成测试	314
12.1	系统集成测试	314
12.1.1	系统集成测试概述	314
12.1.2	系统集成测试要求	315
12.2	系统跟踪工具	315
12.2.1	为什么需要跟踪工具	315
12.2.2	Strace	316
12.2.3	Ltrace	316
12.2.4	LTT	317
12.3	系统性能测量工具	321
12.3.1	代码效率测量	321
12.3.2	LTP	324
12.3.3	LMbench	325
12.4	测量内存泄漏	326
12.4.1	mtrace	326
12.4.2	dmalloc	327
12.4.3	memwatch	328

12.4.4	YAMD	330
第 13 章	部署 Linux 系统	333
13.1	部署 Linux 系统概述	333
13.1.1	部署 Linux 系统的基本流程	333
13.1.2	部署 Linux 系统的关键问题	334
13.2	文件系统类型	335
13.2.1	EXT2/EXT3	335
13.2.2	JFS	337
13.2.3	cramfs	339
13.2.4	JFFS/JFFS2	340
13.2.5	YAFFS	341
13.3	存储设备	343
13.3.1	MTD 类型设备	343
13.3.2	磁盘类型设备	344
13.4	部署 Linux 系统	346
13.4.1	安装 MTD 工具	346
13.4.2	使用磁盘文件系统	347
13.4.3	使用 RAMDISK 设备	348
13.4.4	使用 MTD 设备和 JFFS2 文件系统	349
13.4.5	系统启动和升级	351
第 14 章	系统设计开发实例	352
14.1	需求分析	352
14.2	系统硬件设计	354
14.3	系统软件设计	364
14.4	系统集成与部署	366



第 1 章 概述

本章目标

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。读完本章内容，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。

- 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、通信和软件等各行各业。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, 嵌入式系统将拥有最大的市场。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从大到小的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。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, 4~5 年后嵌入式系统将是继 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阶段。

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。想像飞机正在空中飞行, 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。

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、Nucleus、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 是一个发展最快、应用最为广泛的操作系统。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的开发者们的想像力。图 1.1 所示是业内人士对国内 Linux 软件市场的预测。

根据 IDC 的报告, Linux 已经成为全球第二大操作系统。预计在服务器市场上, Linux 在未来几年内将以每年 25% 的速度增长, 中国的 Linux 市场更是保持 40% 左右的增长速度。而在 Linux 操作系统方面, IDC 对中国在 2001~2006 年的市场预测发现, 其市场占有率从 2001 年的 4.47% 平稳地上升到 2006 年的 26.77%。

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;

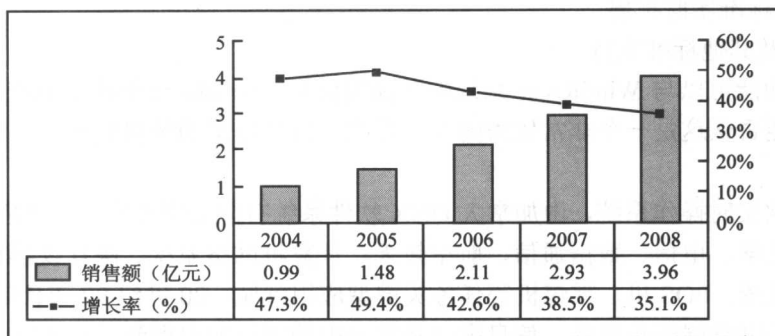


图 1.1 2004~2008 年国内 Linux 软件市场总量预测

RTAI、Kurt 和 Linux/RK 也提供了实时能力；还有 μ CLinux 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) μ C/OS

μ C/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 μ C/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《 μ C/OS-II: the Real Time Kernel》翻译后，在国内掀起 μ C/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价掌上电脑都采用了 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（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。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，那这个系统是不合适的；其次要考虑的是开发调试用的工具，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；第三要考虑的问题是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（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：USB，PC 卡（PCMCIA）的支持，内置的即插即用，等等。

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。