

物联网工程专业系列教材

嵌入式Linux 开发技术基础

主 编 伍德雁

副主编 李显宁 欧义发 陈 军 陈新锐



中国水利水电出版社
www.waterpub.com.cn

物联



嵌入式 Linux 开发技术基础

主 编 伍德雁

副主编 李显宁 欧义发 陈 军 陈新锐



中国水利水电出版社
www.waterpub.com.cn

• 北京 •

内 容 提 要

本书是嵌入式 Linux 开发的基础教程,介绍了嵌入式 Linux 的概念和相关的开发技术。在编写上注重基础性和可操作性,以应用为驱动,以案例为依托,将理论知识与技能培养相结合。全书共分为 15 章,包括嵌入式系统和嵌入式 Linux 的概念、搭建开发环境、开发基础流程、使用自定义函数、使用多文件、使用 make 管理项目、文件读写、网络通信、驱动程序、控制嵌入式系统的硬件、QT 基础等内容。

本书由从事嵌入式 Linux 开发教学工作多年的教师编写,在组织内容时对广度和深度都做了充分的考虑,尽量做到对嵌入式 Linux 开发既有一个比较全面的介绍,又没有太大的难度。行文风格近似课堂教学,对概念的讲解深入浅出,同时具备较强的可操作性,可用作高等院校物联网专业以及相关专业的教材,也可作为嵌入式 Linux 开发初学者的入门教材。

图书在版编目(CIP)数据

嵌入式Linux开发技术基础 / 伍德雁主编. — 北京 :
中国水利水电出版社, 2017.6
物联网工程专业系列教材
ISBN 978-7-5170-5514-3

I. ①嵌… II. ①伍… III. ①Linux操作系统—程序
设计—高等学校—教材 IV. ①TP316.85

中国版本图书馆CIP数据核字(2017)第139511号

策划编辑:周益丹 责任编辑:李 炎 加工编辑:赵佳琦 封面设计:李 佳

书 名	物联网工程专业系列教材 嵌入式 Linux 开发技术基础 QIANRUSHI Linux KAIFA JISHU JICHU
作 者	主 编 伍德雁 副主编 李显宁 欧义发 陈 军 陈新锐
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路 1 号 D 座 100038) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 68367658 (营销中心)、82562819 (万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	三河市鑫金马印装有限公司
规 格	184mm×260mm 16 开本 12.25 印张 303 千字
版 次	2017 年 6 月第 1 版 2017 年 6 月第 1 次印刷
印 数	0001—3000 册
定 价	28.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换
版权所有·侵权必究

前言

近年来,随着计算机技术的发展、电子制造技术的微型化以及控制理论的完善,嵌入式技术得到了快速发展,并在各行各业得到了广泛的应用。越来越多的学校开设了与嵌入式系统相关的课程,有些学校甚至开设了嵌入式技术专业。

编者在高等院校从事计算机类课程的教学工作十几年,对嵌入式 Linux 开发课程的教学工作也深有体会。众所周知,IT 行业是一个知识更新换代非常快的行业,编者在教学过程中也需要不断地学习新知识与新技术。在教学和自学的过程中,编者逐渐地发现,“螺旋式学习模型”是掌握一门课程较有效的学习方法之一。

所谓的螺旋式学习模型,是编者借用了软件工程开发模型中知识自定义的学习模型,是指由浅入深,逐渐迭代的学习方法。首先学习课程最基础的知识,对课程有了初步的概念后,再适当增加难度和广度,向前迈进进一步,然后如此多次反复,最后对课程的掌握就能达到一个很好的高度。所以,编者每自学一个新的知识,一般都会选择从最基础的学起,再逐渐深入。

每一种学习方法都需要有相应的教材作为配套。所以,编者就想结合自己多年的教学经验和心得,按照“螺旋式学习模型”的思想,编写一本嵌入式 Linux 开发的入门教材。这就是本书编写的初衷。

作为入门教材,本书具有以下特点:

1. 简单。本书在组织内容时对广度和深度都做了充分的考虑,尽量做到让初学者对嵌入式 Linux 开发既有一个比较全面的了解,又没有太大的难度。因此,选取了一些相对比较简单必要的知识,舍弃了烧写系统、进程控制等一些初级阶段较少用或者不用的知识,并在讲解驱动程序时使用了虚拟的硬件,使得初学者既能够理解驱动程序的相关知识,又避开了较难的底层机器指令的操作。

2. 行文风格类似课堂讲课,通俗易懂。每一个行业都有大量概念和术语,这些概念和术语对于一个资深从业者而言耳熟能详,但对于初学者来说可能会显得不是那么友好。针对这个问题,本书以类似课堂讲课的风格展开,以通俗易懂的语言解释相关的概念和术语,尽量避免以概念解释概念,以术语解释术语。遇到一些相对比较抽象的概念和术语,书中辅以生活中的现象和例子作类比,以帮助读者理解。

3. 具有较强的可操作性。嵌入式 Linux 开发是一门实践性很强的课程,边学习边操作才会有好的效果,对于加强记忆、加深理解和保持兴趣都有很大的帮助。鉴于此,除了纯概念章节外,每章都提供了经典的例子以供读者实践。同时,考虑到部分读者为自学者,除讲解例子本身外,还讲解了必要的操作过程,文字不容易理解的地方还辅以直观的插图,以避免自学者在编辑、编译、运行中的某个环节被卡住而无计可施。

4. 章节结构统一。除第 1 章是纯概念的章节外,其余章节基本上按照“总体概念→例子→程序解读”的模式展开,在例子前后穿插必要的操作步骤,在程序解读过程中穿插解释一些必要的术语和概念。

本书精心编写了每一个例子,确保每个例子都能正确无误地运行。并且,从头到尾都在

向读者灌输一种好的编程习惯和思想。

为了方便读者上机实验，在没有涉及到嵌入式系统的专用硬件时，本书的例子直接以 PC Linux 为实验平台；只有在涉及到嵌入式系统的专用硬件时，才把程序移植到嵌入式 Linux 上运行。

本书是编者按照自定义的“螺旋式学习模型”，针对初学者和自学者的一次尝试，适合高等院校物联网相关专业、计算机相关专业的在校学生，以及其他自学者、初学者学习使用。

本书在编写过程中参考了作者手头和图书馆能借到的所有相关教材，以及网上大量的相关资料。在此衷心感谢这些教材的编者和出版社、网上资料的创作者、转载者以及相关的网站。

由于编者水平有限，加之时间仓促，书中难免存在不妥和错误之处，敬请广大读者批评指正。

编者

2017 年 4 月

目 录

前言

第1章 认识嵌入式系统	1	第6章 使用多个源文件编程	51
1.1 什么是嵌入式系统	1	6.1 使用多个源文件编程	51
1.2 嵌入式系统的发展历史	2	6.2 多源文件的项目编译	55
1.3 嵌入式系统的应用	3	6.3 【项目 6.1】的另外版本	56
1.4 常见的嵌入式操作系统	5	小结	58
1.5 嵌入式 Linux 的优点	6	思考与练习	58
1.6 本书的内容以及结构	7	第7章 认识和使用 make 工具	59
小结	8	7.1 认识 make 工具	59
思考与练习	8	7.2 改进版 Makefile	63
第2章 在虚拟机上安装 Linux	9	7.3 Makefile 的进阶用法	65
2.1 虚拟机软件简介	9	7.3.1 伪目标	65
2.2 下载安装 VMware Workstation Player	10	7.3.2 使用变量	66
2.3 Linux 发行版本简介	12	小结	67
2.4 在虚拟机上安装 CentOS 6.7	14	思考与练习	67
小结	23	第8章 文件读写	68
思考与练习	23	8.1 基于句柄的文件读操作	68
第3章 搭建 C 语言开发环境	24	8.2 基于句柄的文件写操作	71
3.1 IDE 和分步开发	24	8.3 基于流的文件读写操作	73
3.2 安装 C 语言开发工具集	24	小结	76
3.3 优化 gedit 编码环境	29	思考与练习	76
小结	35	第9章 网络通信编程	78
思考与练习	35	9.1 网络通信基本原理	78
第4章 第一个 Linux C 程序 Hello world!	36	9.2 网络编程基础	81
4.1 准备工作	36	9.2.1 套接字	81
4.2 编写代码	36	9.2.2 数据存储字节序	81
4.3 编译、链接程序	39	9.2.3 套接字数据类型	82
小结	43	9.2.4 地址格式转换	83
思考与练习	43	9.2.5 网络编程流程	83
第5章 使用自定义函数	44	9.3 TCP 协议网络编程实例	84
5.1 在程序中使用自定义函数	44	9.4 UDP 协议网络编程实例	89
5.2 使用多个自定义函数	48	小结	94
小结	50	思考与练习	95
思考与练习	50	第10章 驱动程序初步	96

10.1 驱动程序基本概念	96	13.3 跑马灯 LED 控制程序	131
10.2 最简单的字符设备驱动程序	98	13.4 按键读取程序	133
10.3 驱动程序进阶 1	103	13.5 按键控制 LED	136
10.4 测试程序	106	小结	139
小结	108	思考与练习	139
思考与练习	108	第 14 章 串口通信和 GPIO 的使用	140
第 11 章 驱动程序进阶 2	109	14.1 串口通信基础	140
11.1 具备读写接口的驱动程序	109	14.2 串口通信程序	143
11.2 测试程序	113	14.3 GPIO 基础	149
小结	115	14.4 GPIO 编程	150
思考与练习	115	小结	153
第 12 章 嵌入式 Linux 开发基础	116	思考与练习	154
12.1 嵌入式系统结构特点	116	第 15 章 基于 Qt 的嵌入式 Linux GUI 程序	
12.2 嵌入式系统所采用的处理器 (CPU)	117	设计基础	155
12.3 凌阳实验箱简介	119	15.1 Qt 相关概念	155
12.4 交叉编译	120	15.2 最简单的 Qt 应用程序 Hello world!	157
12.5 交叉编译工具 arm-linux-gcc 的		15.3 人工编码的 Hello world! 程序	164
安装和配置	122	15.4 图形界面控制 LED	166
12.6 在嵌入式 Linux 上运行第 1 个程序		小结	176
Hello world!	123	思考与练习	176
小结	126	附录 1 常用标准输入输出函数	177
思考与练习	126	附录 2 基于句柄的文件读写函数	179
第 13 章 控制板载 LED	127	附录 3 基于流的文件读写函数	181
13.1 控制板载 LED 的方法和相关概念	127	附录 4 基于套接字的网络通信函数	185
13.2 爆闪式 LED 控制程序	129		

第1章 认识嵌入式系统

在今天这个互联网已经延伸到世界每个角落，各种物件越来越聪明的年代，嵌入式系统无处不在。正是因为有了嵌入式系统，世界才越来越智能化，越来越精彩。那么，什么是嵌入式系统？嵌入式系统有着什么样的发展历史？它们有哪些应用？有哪些类型？等等。下面将围绕这些问题来初步认识嵌入式系统。

1.1 什么是嵌入式系统

IEEE（电气和电子工程师协会）对嵌入式系统的定义是这样的：“嵌入式系统是控制、监视或辅助机器和设备运行的装置。”国内大多数资料都这样描述嵌入式系统：“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，适应应用系统对功能、可靠性、成本、体积、功耗等严格要求的专用计算机系统。”

要理解嵌入式系统，得先从普通计算机下手。我们日常使用的PC，以及机房里的服务器，都是完整的计算机系统，有独立的主机，有键盘、鼠标等常用输入设备，有显示器、打印机等常用输出设备。同时，它们干的也是通常意义上计算机该干的活，比如服务器对外提供Web服务、DNS服务，PC进行文档处理、玩游戏、上网等。

随着计算机技术的发展、电子制造技术的微型化以及控制理论的完善，人们越来越不满足于计算机只能干传统的活儿。人们在想能不能把计算机安装到其他设备或机器中，帮助控制或者部分控制设备或机器，通过采集设备或机器的运行数据和环境数据，借助计算机的高速运算，随时调整设备或机器的运行状态，实现设备或机器的自动化、智能化或效率最优化？今天，这个想法已经得到了实现，大家都熟知的汽车电子燃油喷射系统，如图1-1所示就是这么一个应用。传统的化油器燃油喷射系统完全依靠驾驶员踩油门打开化油器的进气阀门，空气流过化油器喉管时产生负压，将浮子室内的汽油吸到喉管并随同空气流雾化成可燃混合气。在化油器燃油喷射系统中，油门踩多深，燃油就喷射多少。电子燃油喷射系统则由车载计算机根据各个传感器（进气温度传感器、节气门位置传感器、曲轴位置传感器、水温传感器、氧传感器等）采集的数据来控制燃油喷射。由于发动机的工作状态不同，在不同的情况下，驾驶员把油门踩到同样的深度，电子燃油喷射系统喷射的燃油量是不同的。采用电子燃油喷射技术大大提高了燃油的燃烧效率，降低了发动机的燃油消耗，减少排气污染，同时大大改善了发动机的工况，提高了发动机的热效率，允许发动机采用更高的压缩比，可以减少发动机的爆震倾向。

像汽车的电子燃油喷射系统这样，把一个计算机系统植入到其他设备或机器中，帮助控制或部分控制设备或机器，实现设备或机器的自动化、智能化或效率最优化等目的，这样的—个计算机系统就是嵌入式系统。为了描述方便，我们姑且把被计算机系统植入的设备或机器称为母体。

要深刻理解嵌入式系统，需要注意以下几点：

（1）母体不是一个计算机系统，而是有专门用途的其他设备或机器，比如前面所说的汽车，它的用途是行驶；比如冰箱，它的用途是冷冻。

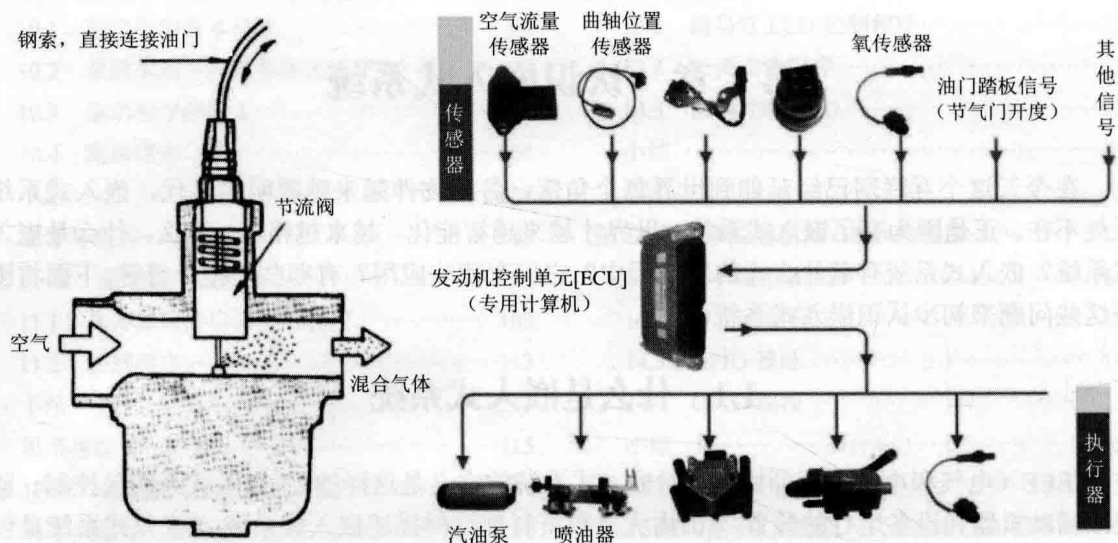


图 1-1 汽车化油器系统和电喷系统比较

(2) 一般而言，嵌入式系统是一个有专门用途的计算机系统，大多数时候，没有通用计算机系统常见的键盘鼠标显示器等输入输出设备。

(3) 由于嵌入式系统是植入到母体的计算机系统，为了不影响母体的结构和使用，必须要求嵌入式系统的体积足够小，因此，会把一些不必要的功能硬件（模块）裁剪掉。

(4) 微型化的嵌入式系统除了裁剪功能硬件外，还会在性能硬件上有所妥协，为了保证系统的正常运行，需要把不必要的功能软件（模块）裁剪掉。

(5) 由于嵌入式系统的专用性、植入性，因此，对体积、功耗都有严格要求。

(6) 对于一些高可靠性的母体，同样要求嵌入式系统具有高可靠性。

(7) 对于一些低价格的商品性母体，对嵌入式系统的成本有严格要求。

所以，IEEE 和国内资料对嵌入式系统的定义表面上看不一样，实际上是从不同的角度给出的定义。

1.2 嵌入式系统的发展历史

第一台电子数字计算机于 1946 年诞生。在其后漫长的历史进程中，计算机一直是供养在专门机房里的大型昂贵设备。

随着微电子技术和信息技术的进一步发展，微处理器的功能日益强大、集成的模块也越来越多，一个单独的芯片就可以完成以往一整套计算机的功能（SOC, System On Chip, 片上系统）。计算机技术的这些特点以及其智能化引起了控制专业人士的兴趣，研究将计算机系统嵌入到其他设备或机器中，实现设备或机器的智能化控制和自动化控制。例如，将微型化的计算机系统安装到大型舰船中以实现自动驾驶或轮机状态监测。这样一来，计算机系统便失去了原来的形态与通用的计算机功能。为了区别于原有的通用计算机系统，我们把嵌入到其他设备或机器中，实现设备或机器智能化控制和自动化控制的计算机，称作嵌入式计算机系统，简称嵌入式系统。因此，嵌入式系统是微型计算机继续发展和工业自动化进一步发展的产物。

第一个嵌入式系统可以追溯到 20 世纪 60 年代美国麻省理工学院仪器研究室开发的阿波罗导航计算机,在两次阿波罗登月中都使用了这种惯性导航系统。第一款大批量生产的嵌入式系统是 20 世纪 60 年代美国民兵 I 导弹上的 D-17 自动导航控制计算机。这些嵌入式系统的出现甚至早于个人计算机。但这些嵌入式系统有个共同的特点,就是体积庞大、耗电量大,与我们现在所理解的微型化的嵌入式系统有着很大的不同,更像是把一个通用的计算机系统嵌入到其他设备或机器中,只能算是现代嵌入式系统的雏形。

嵌入式系统的真正发展是在微处理器问世之后。1971 年 11 月,Intel 公司成功地把算术运算器和控制器电路集成在一起,推出了第一款微处理器 Intel 4004,其后各厂家陆续推出了多款 8 位、16 位的微处理器,包括 Intel 8080/8085、8086, Motorola 的 6800、68000,以及 Zilog 的 Z80、Z8000 等。以这些微处理器作为核心所构成的系统,广泛地应用于仪器仪表、医疗设备、机器人、家用电器等领域。70 年代中后期 IT 厂商研发出了集微处理器、存储器和 I/O 于一片超大规模集成电路上的单片机(比如 1976 年 Intel 公司推出的 MCS-48 单片机),正好符合嵌入式系统对体积、功耗和可靠性的追求,于是,嵌入式系统开始基于单片机发展。这一时期的嵌入式系统还没有操作系统的支持,只是使用汇编语言甚至机器语言的方式进行编程,对系统进行直接的控制,虽然已经初步具备了嵌入式的特点,但仅仅是通过 CPU 执行一些简单的代码,因此严格来说也不算是真正的嵌入式系统。

20 世纪 80 年代,随着微电子技术的提高,微处理器、RAM、ROM、IO 接口以及其他接口集成到一片超大规模集成电路中,制造出了面向 I/O 设计的微控制器。同时,软件技术也得到了同步发展,研发出了适应集成微控制器的简单的操作系统。由于操作系统的支持,应用软件的开发难度大大降低,开发周期大大缩短,开发效率大大提高,软件的功能也更加完善、复杂。

90 年代,分布控制、柔性制造以及数字化通信等需求迅速爆发,带领着嵌入式系统进一步飞速发展。面向实时信号处理的 DSP 产品向着高速度、高精度、低功耗的方向发展。另一个方面,随着硬件性能的大大提高,嵌入式系统的操作系统也向高效率、多任务方向发展,并逐渐成为嵌入式操作系统的主流。这一阶段的嵌入式系统已经具备了文件系统的概念,并具备多任务、网络以及图形用户界面(GUI)的功能,甚至提供了丰富的应用程序接口(API),使得应用软件的开发更加简单。

21 世纪,计算机技术迅速向着网络化的方向发展,嵌入式系统也自然而然地顺应着这一历史潮流,普遍地具有网络连接的功能,能够应用到 Internet 中去,实现远距离通信。特别是随着物联网时代的到来,嵌入式系统更是发挥了前所未有的巨大作用,设备或机器通过嵌入式系统连接到 Internet 中,实现智能化的机器对话、人机对话。

1.3 嵌入式系统的应用

嵌入式系统无论在生活、生产还是管理上都有着广泛的应用。典型的例子包括智能家电、家庭远程管理、交通管理、移动支付、环境监测、机器人等。

1. 生活方面的应用

如图 1-2 所示,嵌入式系统在生活方面的应用主要包括通信、智能家电、家庭远程管理、移动支付(POS 支付)等。我们现在每天都在使用的智能手机,不管是苹果还是安卓都是典

型的嵌入式系统，它们既是一个通信工具也是一台小型平板电脑。越来越多的家用电器由于安装了嵌入式系统而变得智能，比如智能冰箱、智能电视等。未来智能家电甚至能够接入到因特网中实现远程交互，实现家庭的智能管理和远程管理。此外，我们平时购物刷卡付款使用的POS机也是一个嵌入式系统的例子。



图 1-2 嵌入式系统在日常生活中的应用

2. 生产方面的应用

在生产方面的应用一直是嵌入式系统应用的重要领域，特别是近年工业 4.0 概念的提出，基于嵌入式系统的工业智能化设备获得了长足的发展，应用到各行各业的生产实践中，如工业过程控制、数字机床、电力系统、电网安全、电网设备监测、石油化工系统，提高了生产效率和产品质量，减少了人力资源的消耗，降低了人力成本。随着微电子技术的发展，嵌入式系统使用的微处理器已经从传统的 8 位单片机发展到 16 位、32 位，以及 64 位，甚至多核心。性能的提高，使得嵌入式系统在生产方面的应用更加广泛。如图 1-3 所示是嵌入式系统在生产方面应用的例子。

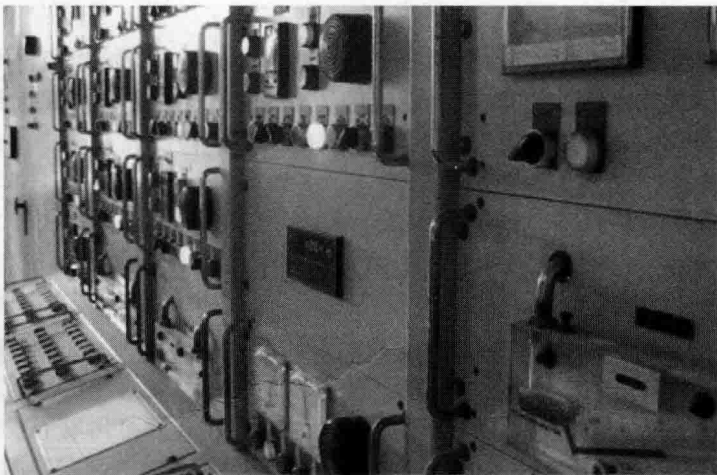


图 1-3 嵌入式系统在生产中的应用

3. 管理方面的应用

随着物联网技术的发展,很多国家都提出了智慧国家的概念,在此基础上,国外国内的很多城市提出了智慧城市的概念。智能交通(如图 1-4 所示)、智能医疗、智能图书馆就是智慧城市的典型应用。所有这些应用中,最基础的是智能设备,而智能设备的基础就是嵌入式系统。

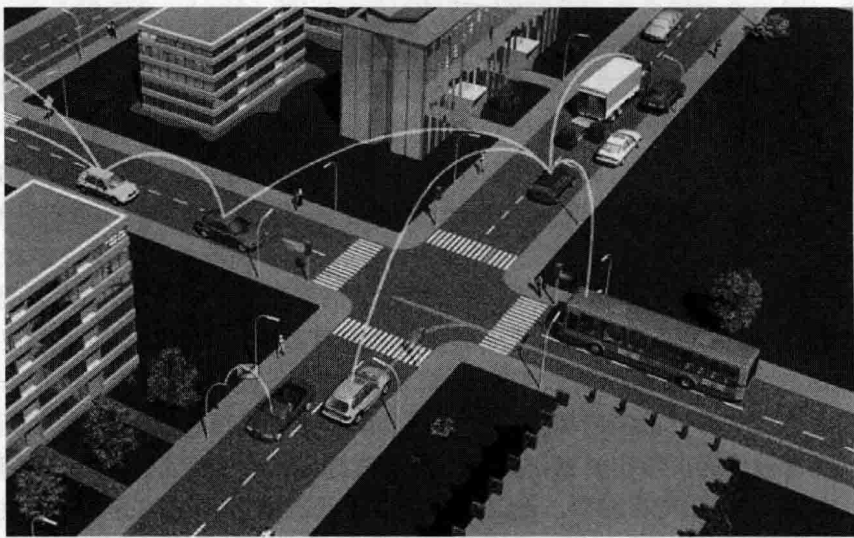


图 1-4 嵌入式系统在交通管理中的应用

随着生活、生产和管理的智能化,越来越多的嵌入式系统被应用到各行各业、各个领域。

1.4 常见的嵌入式操作系统

从上世纪 80 年代开始,嵌入式计算机系统就有了操作系统的支持,国际上一些 IT 组织和公司开始进行嵌入式操作系统的研发。经过多年的发展,涌现了一批著名的嵌入式操作系统。其中,现在应用中比较流行也比较有代表性的有 Windows CE、VxWorks、eCos、 μ C/OS-II、QNX、Linux 等。

1. Windows CE

Windows CE 是 Microsoft 公司专门为掌上电脑和嵌入式设备设计开发的、可升级的 32 bit 嵌入式操作系统。Windows CE 提供了与 Windows 桌面版类似的图形界面,容易上手。Windows CE 具有模块化、结构化和基于 Win32 应用程序接口的特点,能在多种处理器体系架构上运行。进行 Windows CE 的应用开发比较容易,可以使用桌面版 Windows 上的编程工具(如 Visual Basic、Visual C++等)、同样的函数、同样的界面风格。大多数的应用软件只需简单的修改和移植就可以在 Windows CE 平台上继续使用。

2. VxWorks

VxWorks 是美国 WindRiver 公司推出的嵌入式实时操作系统(RTOS)。系统内核具有高可靠性、高实时性、高性能、组件丰富、可裁剪的特点。由于其良好的可靠性和卓越的实时性,VxWorks 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,

如美国的 F-16、FA-18 战斗机、B-2 隐形轰炸机和爱国者导弹上就使用了 VxWorks。VxWorks 支持的 CPU 有 M68K、PowerPC、ColdFire、Intel X86、ARM、SPARC、MIPS 等几十种。WindRiver 公司为 VxWorks 推出了集成开发环境 Tornado，为用户提供支持 C、C++ 语言的交叉编译器和类库。VxWorks 性能优越，开发调试便利，虽然有强大的技术支持，但是昂贵的价格让开发者望而却步。

3. eCos

eCos (embedded Configuration operating system) 诞生于 1997 年，最早由 Cygnus 公司开发，后被 Red Hat 收购。eCos 是一个小型的嵌入式实时操作系统，它源代码是开放的，内核可灵活配置，编译出来的内核最小可至 10 KB 的级别，适合用于微小型系统。eCos 支持的 CPU 包括 ARM、Hitachi H8300、Intelx86、MIPS、Matsushita AM3x、Motorola 68k、PowerPC、SuperH、SPARC 和 NEC V8xx，可以方便对不同平台的移植。eCos 的开发编译工具可支持 GNU 开源的开发工具链，开发调试方便。

4. μ C/OS-II

μ C/OS-II 是 Micrium 公司的基于优先级抢占式多任务的小型实时操作系统，最早发布于 1992 年。 μ C/OS-II 包含了实时内核、任务管理、时间管理、任务间通信同步（信号量，邮箱，消息队列）和内存管理等功能。内核代码量小巧，其源码 99% 符合 MISRA C 编码标准，适合学习和研究，经常被作为大学实时操作系统的典型教材。 μ C/OS-II 支持多种技术架构的 CPU，包括 ARM、AVR、PowerPC、8051、80xC52、80x86 等。 μ C/OS-II 的开发环境往往依赖于具体的 CPU 环境，比如 ARM 用 ADS，x86 用 Borland C++，AVR 用 AVR Studio。

5. 嵌入式 Linux

得益于 Linux 的完全开放源码、遵循 POSIX 标准的特点和类 UNIX 的设计思想，以及网络社团推动的发展方式，近年来嵌入式 Linux 已经成为最重要的嵌入式操作系统，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。

与其他嵌入式操作系统相比，Linux 的源代码是完全开放的，全世界自由软件开发者和设备厂家共同推动嵌入式 Linux 的发展。嵌入式 Linux 继承了 Linux 的特点，内核稳定、体积小，可裁剪程度高。嵌入式 Linux 适用于多种 CPU 和硬件平台，同时，Linux 与生俱来的优秀网络血统更为其在物联网时代的发展铺平了一条宽广平坦的大路。嵌入式 Linux 正逐渐成为嵌入式操作系统中的新贵。近年来风头正劲的完全图形化人机接口的嵌入式系统 Android，实际上也算半个嵌入式 Linux。Google 以 Linux 内核为内核，增加图形接口、应用的运行库和运行环境等元素即形成了 Android。

1.5 嵌入式 Linux 的优点

目前主流的嵌入式系统中，大约有一半使用的操作系统是 Linux，或者以 Linux 为基础二次开发的操作系统（比如 Android）。Linux 之所以能在嵌入式系统市场上取得如此辉煌的成绩，与其自身的优良特性是分不开的。

1. 完全的开源

Linux 从一开始就在因特网上完全公开源代码，Linux 的发展依靠网络上的爱好者和热心人士共同推动。任何人、任何组织只要遵守 GPL 条款，就可以自由地使用 Linux 源代码。这

一点为嵌入式系统提供了最大限度的自由度，因为嵌入式系统千差万别，往往需要针对具体的应用进行修改和优化，因而获得源代码就变得至关重要。另外，遵守 GPL 条款的 Linux 软件资源十分丰富，每种通用程序的源码几乎都可以在网上找到，使得设计者在其基础之上进行二次开发变得非常容易。

2. 广泛的硬件支持

Linux 能够支持 x86、ARM、MIPS、ALPHA、PowerPC 等多种体系结构，目前已经成功移植到数十种硬件平台，几乎能够运行在所有流行的 CPU 上。Linux 有着异常丰富的驱动程序资源，支持各种主流硬件设备和最新硬件技术，甚至可以在没有存储管理单元（MMU）的处理器上运行，这些都进一步促进了 Linux 在嵌入式系统中的应用。

3. 高可定制性

由于不同的嵌入式应用对系统要求各不相同，这要求嵌入式 OS 具备高可定制性，能够根据需要方便地增加和减少各项功能模块。Linux 的内核设计非常精巧，分为进程调度、内存管理、进程间通信、虚拟文件系统和网络接口五大部分，其独特的模块机制可以根据用户的需要，实时地将某些模块插入到内核或从内核中移走。Linux 的驱动程序机制也很灵活，除了可以编译到内核外，还可以编译成独立的模块，根据需要随时加载或卸载。这些特性使得 Linux 系统内核可以裁剪得非常小巧，很适合于嵌入式系统的需要。

4. 占有较少的硬件资源

多数嵌入式系统具有成本敏感性，处理器速度较低，存储器空间较少，这要求嵌入式 OS 体积小，速度快。Linux 体系结构比较灵活，易于裁减，可以小到 2MB Flash，4MB RAM。Linux 的 GUI 并不是内核的模块，可以把 GUI 部分彻底去除，以减轻系统对硬件的依赖程序和加快系统的运行速度。

5. 具备强大的网络功能

随着物联网时代的到来，要求越来越多的嵌入式设备具备网络功能。这就要求嵌入式操作系统支持常用的网络协议。Linux 的网络功能经过几次改进，其效率、功能都很突出，支持几乎所有常见的网络协议，具有众多的网络工具，并且，向上提供了标准的网络调用接口，使得程序开发人员开发网络应用程序变得容易。这些特性使得 Linux 在嵌入式系统中倍受青睐。

6. 其他优点

嵌入式 Linux 还有很多其他优点，比如高安全性。由于 Linux 源代码公开，用户不必担心有“后门”等安全隐患。比如具备完善的开发工具集，从编辑到编译，到链接，再到调试，一应俱全。比如拥有嵌入式 GUI QT/Embedded，嵌入式浏览器 Opera 等。

1.6 本书的内容以及结构

由于 Linux 使用 C 语言以及少量汇编语言写成，所以 Linux 和 C 密不可分，Linux 系统本身配备了 C 语言函数库和相关的编译、调试工具集，C 语言是编写 Linux 应用程序以及系统模块（比如驱动程序）的首选语言。本书讲述嵌入式 Linux 的 C 语言开发技术。内容包括 Linux 下 C 开发工具、编译工具、调试工具的安装、配置和使用、Linux C 开发一般方法、自动编译工具 make 的使用、文件读写、多线程编程、网络通信、嵌入式系统的输入输出等。

本书假定读者具有 Linux 操作系统以及 C 语言程序设计的基础知识，所以书中不再系统

地讲解 Linux 操作系统的使用和 C 语言的详细语法，只针对 C 语言在 Linux 下的开发展开论述。如果读者需要补充 Linux 和 C 的基础知识，请自行参考相关书籍和资料。

嵌入式系统由于对体积、功耗、成本等都有严格限制，导致没有足够的硬件资源在板上直接进行应用开发，因此，嵌入式系统的开发一般采用交叉编译的方式。先在个人计算机上进行软件的编写、编译，再将编译好的软件系统移植到嵌入式系统上运行。嵌入式系统的这一特点使得读者朋友对于嵌入式系统的开发学习变得困难。

但是，另一方面，嵌入式 Linux 在系统结构和使用上与 PC 版通用 Linux（后续章节简称为 PC Linux）并无二致，当应用不涉及嵌入式系统的专有硬件时，程序的开发过程和运行方式、运行结果在嵌入式 Linux 上和 PC Linux 上是一模一样的。因此，为了方便读者学习和实验，当不涉及嵌入式系统的专用硬件时，本书的内容直接在 PC Linux 上展开，程序的开发、编译和运行都直接在 PC Linux 上完成。当涉及到嵌入式系统的专用硬件时，比如读取嵌入式开发板的按钮、控制 LED、控制蜂鸣器等，则进行交叉编译，把程序移植到嵌入式系统上运行。

所以，本书的内容结构大致分为两大部分。第一部分是 C 语言在 PC Linux 下的开发；第二部分是嵌入式 Linux 下的 C 语言开发。这两大部分内容实际上都是围绕同一个主题，就是“嵌入式 Linux 开发”。

小结

本章介绍了嵌入式系统的基本概念，嵌入式系统的发展历史以及其在日常生活、生产以及管理等方面的应用。在万物越来越智能的今天，嵌入式系统无处不在。在众多的嵌入式操作系统中，由于嵌入式 Linux 具有比较突出的优势而被广泛使用。嵌入式 Linux 开发是软件开发行业的重要领域，是物联网相关专业的重要课程。

思考与练习

1. 什么是嵌入式系统？嵌入式系统有什么特点？举出你知道的嵌入式系统应用的例子。
2. 从各方面比较嵌入式系统与通用计算机的区别。
3. Linux 作为嵌入式系统的操作系统有哪些优势？
4. 查找相关资料，了解工业 4.0 的概念和内涵。

第2章 在虚拟机上安装 Linux

Windows 无疑是 PC 以及笔记本电脑上（以下统称桌面系统）最成功的操作系统。华丽的界面、人性化的布局、简单易用的人机接口和丰富的第三方软件都使得 Windows 成为桌面系统的不二选择。Linux 曾经雄心勃勃地想进军桌面系统，和 Windows 分一杯羹。但是，华丽、人性化和简单易用并不是 Linux 的长处，经过多年的努力，Linux 在桌面系统领域依然无法撼动 Windows 的霸主地位。Linux 的优势在于安全、稳定、高效、需要硬件资源少、高度可定制。和 Windows 相比，Linux 更多地应用在服务器和嵌入式系统上。

相信阅读本书的读者朋友都是使用 Windows 作为自己的桌面系统，不大可能为了学习嵌入式 Linux 开发而把 Windows 换成 Linux。因此，为了不影响大家的工作、生活和娱乐，同时方便学习本书内容，最好的解决办法是使用虚拟机，在虚拟机上安装 Linux 操作系统。

所谓虚拟机，简单地说就是在一台计算机系统上安装一个软件，通过软件模拟的方式产生一台具有完整硬件系统，运行在一个完全隔离环境中的“独立的”计算机系统。这个模拟软件称为虚拟机软件。通过虚拟机软件，可以在一台物理计算机上模拟出一台或多台虚拟的计算机，这些虚拟机可以像真正的计算机那样进行工作，可以安装操作系统、安装应用程序、访问网络资源等。对于使用者而言，虚拟机只是运行在物理计算机上的一个应用程序，但是对于在虚拟机中运行的应用程序而言，它就在真正的计算机中工作。

2.1 虚拟机软件简介

目前比较常用的虚拟机软件主要有三款：Virtual PC、VMware Workstation 和 VirtualBox。

Virtual PC 是 Microsoft 公司开发的免费软件，其占用资源少，运行速度较快，易学易用，但功能较少。作为 Microsoft 自己的产品，Virtual PC 完美支持 Windows 操作系统，但对其他操作系统兼容性不够好。Virtual PC 的下载地址是<https://www.microsoft.com/zh-cn/download/details.aspx?id=3702>。

VMware Workstation 是全球知名虚拟化解决方案领导厂商 VMware 的产品，功能强大。VMware Workstation 对网络的访问提供了多种解决方案，可以满足一般用户和高端用户的需求。同时，VMware Workstation 对各种操作系统的兼容性较好，能够完美支持 Windows 和 Linux。但是 VMware Workstation 的缺点是占用资源较多，运行速度较慢。而且，VMware Workstation 是商用版本，需要付费使用。VMware 公司的官方网站是 <http://www.vmware.com>。

VirtualBox 是 Sun Microsystems 公司的产品。2010 年 1 月，Oracle 收购 Sun，VirtualBox 归到 Oracle 旗下，更名为 Oracle VM VirtualBox。VirtualBox 是一款免费并且开源的轻量级产品，提供了能够运行在很多不同的主机操作系统的版本。同时 VirtualBox 支持很多不同的操作系统，包括 Windows、大多数的 Linux 发行版、Mac OS 以及 Solaris 等，但遗憾的是，目前 VirtualBox 只能支持 32 位的操作系统，不支持 64 位版本。VirtualBox 的官方网站是 <http://www.virtualbox.org>。

综上所述，三款虚拟机软件各有优缺点，Virtual PC 易学易用，运行速度快，但对 Linux 支持不完美；VMware Workstation 功能强大，但占用资源较多，运行速度慢，需要付费使用；VirtualBox 轻量、开源，支持多种操作系统，但遗憾的是不支持 64 位版本。

为了方便个人用户学习，VMware 提供了一个免费的版本 VMware Workstation Player。VMware Workstation Player 是 VMware Workstation 的功能删减版，但完全能够满足一般用户的日常使用需要，而且体积小巧，运行速度比 VMware WorkStation 快，好学好用，支持各种操作系统，还支持 32 位和 64 位版本。

综合比较，本书采用 VMware Workstation Player 作为 PCLinux 的虚拟机软件。

2.2 下载安装 VMware Workstation Player

打开 VMware 公司的主页，将鼠标指针移动到页面右上方的“下载”链接上，网站弹出 VMware 公司可供下载的产品列表对话框，如图 2-1 所示。

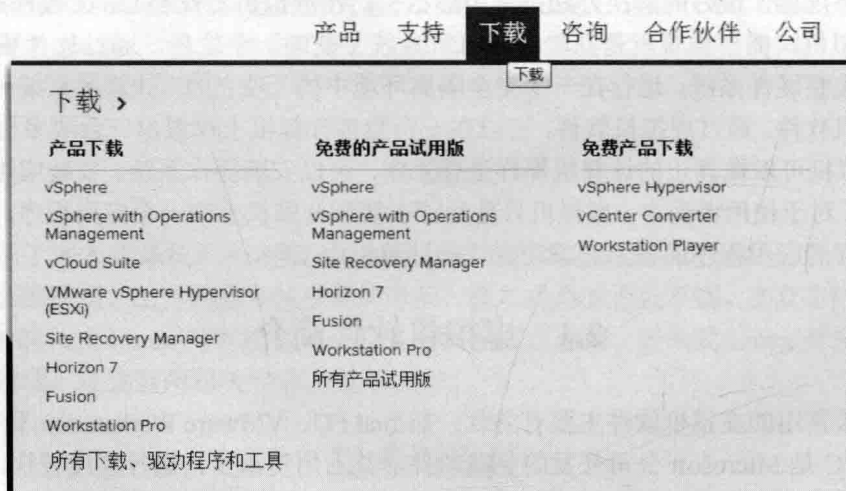


图 2-1 VMware Workstation Player 下载页面

在最右边的“免费产品下载”列里即可找到 Workstation Player，此即是 VMware Workstation Player 的下载链接。点击此链接，跳转到 VMware Workstation Player 下载页，如图 2-2 所示。



图 2-2 VMware Workstation Player 版本选择

页面显示的是 VMware Workstation Player 64 位最新版本，如果读者使用的是 32 位