Broadview
www.broadview.com.cn

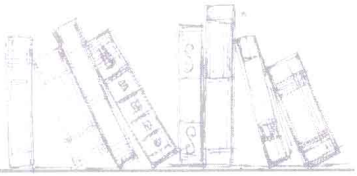
华清远见 FARSIGHT

高等院校嵌入式人才培养规划教材

从实践中学

嵌入式Linux C编程

华清远见嵌入式学院 曾宏安 编著



提供PPT等教学相关素材以及专业视频免费下载

丛书特色：

凝聚业内著名讲师一线教学经验
汇总百家知名企业最新人才标准
精选实用案例直击真实项目需求
结合学习思路提供配套技术资料

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

高等院校嵌入式人才培养规划教材

从实践中学

嵌入式Linux C编程

华清远见嵌入式学院　曾宏安　编著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

本书详细介绍了开发工具和 Linux C 语言基础、嵌入式 Linux C 语言高级编程、内核常见数据结构的解析与应用、嵌入式 Linux 编程基础、文件 I/O 操作相关的 C 语言应用及网络通信相关的 C 语言应用等，并设置了嵌入式 Linux C 函数参考附录。

本书是大学院校嵌入式技术专业、电子信息类其他专业的专业课程教材，也可供高等及中等职业技术院校使用。

图书在版编目（CIP）数据

从实践中学嵌入式 Linux C 编程 / 华清远见嵌入式学院编著. —北京：电子工业出版社，2012.3
高等院校嵌入式人才培养规划教材
ISBN 978-7-121-15883-4

Ⅰ. ①从… 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

中国版本图书馆 CIP 数据核字（2012）第 023558 号

策划编辑：胡辛征
责任编辑：贾　莉
特约编辑：赵树刚
印　　刷：
装　　订：三河市鑫金马印装有限公司
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：19.75　字数：506 千字
印　　次：2012 年 3 月第 1 次印刷
印　　数：3000 册　　定价：39.80 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。

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

服务热线：（010）88258888。

推荐序

移动与云计算的发展推动了越来越多的新技术、新应用和新产品的涌现，推动了嵌入式电子产品世界的不断更新和快速发展。作为嵌入式行业最著名的厂商之一，20多年来ARM除了不断地加大研发投资，开发最新的微处理器、图形技术、物理IP和开发工具，为产业升级搭建了最佳的开发架构；同时，也一直致力于建设一个开放的、具有强大生命力和发展前景的ARM嵌入式生态系统，使得每个存在于这个生态系统的成员都能发挥各自的特长，通过有效的产业分工和协作开发出高性能、低功耗、人性化的嵌入式产品服务于广大的消费者。

在这个生态系统中，嵌入式操作系统是必不可少的重要环节，是“链接”底层硬件和上层应用软件的纽带。其中，Linux 作为开源的嵌入式操作系统，多年来一直受到广大工程师朋友的喜爱，特别是在基于Linux内核的Android操作系统发布以来，Linux的应用和发展到了一个崭新的高度。ARM作为应用最广泛的嵌入式处理器，对Linux操作系统的发展也做出了大量的支持与贡献。

吴雄昂
ARM中国区总经理

前　言

在今天所处的大时代背景下，嵌入式、3G、物联网、云计算俨然已经成为信息产业的主旋律，不管是从政府大力扶持还是从产业变革来说，这股潮流早已势不可挡。而嵌入式系统正是这些产业应用技术中最核心的部分。随着智能化电子行业的迅猛发展，嵌入式行业更是凭借其“应用领域广、人才需求大、就业薪资高、行业前景好”等众多优势，成为当前最热门、最有发展前途的行业之一，与此同时，嵌入式研发工程师更是成为IT职场的紧缺人才。因此，近几年来，各大学、高职高专院校开始纷纷开设嵌入式专业。但是，各院校在嵌入式专业教学建设的过程中几乎都面临教材难觅、内容更新迟缓的困境。虽然目前市场上嵌入式开发相关书籍比较多，但几乎都是针对有一定基础的行业内研发人员而编写的，并不完全符合高校的教学要求。

针对高校专业教材缺乏的现状，我们以多年来在嵌入式工程技术领域内人才培养、项目研发的经验为基础，汇总了近几年积累的数百家企业对嵌入式研发相关岗位的真实需求，并结合行业应用技术的最新状况及未来发展趋势，调研了数十所开设“嵌入式工程技术”专业的院校的课程设置情况、学生特点和教学用书现状。通过细致的整理和分析，对专业技能和基本知识进行合理划分，编写了这套高等院校嵌入式人才培养规划教材，包括：

- 《从实践中学ARM嵌入式体系结构与接口技术》
- 《从实践中学嵌入式Linux操作系统》
- 《从实践中学嵌入式Linux C编程》
- 《从实践中学嵌入式Linux应用程序开发》

本套教材按照专业整体教学要求组织编写，各自对应的主干课程之间既相对独立又有机衔接，整套教材具有系统性。《从实践中学ARM嵌入式体系结构与接口技术》侧重介绍接口技术；在操作系统教材方面，根据各院校的教学重点和行业实际应用情况，编写了《从实践中学嵌入式Linux操作系统》；考虑到嵌入式专业对学生C语言能力要求较高，编写了《从实践中学嵌入式Linux C编程》，可作为“C语言基础”课程的后续提高课程使用；《从实践中学嵌入式Linux应用程序开发》则重点突出了贯穿前面所学知识的实训内容，供“嵌入式Linux应用开发”课程使用。

作为嵌入式Linux开发的主要编程语言，C语言是嵌入式开发工程师的必备基础。本书从嵌入式Linux环境下C语言的开发工具入手，通过大量的代码和实例分析，引领读者逐步掌握嵌入式Linux平台上C语言编程的核心知识和技能。

全书共7章，第1章介绍了嵌入式Linux下常用的C语言开发工具，为后面的学

习打下基础。

第 2 章和第 3 章讲解了嵌入式 Linux C 语言的基础和高级编程的经验技巧。

第 4 章介绍了嵌入式 Linux 内核中常见的数据结构。

第 5 章为文件操作，主要讲述了 Linux 系统调用、Linux 文件 I/O 系统、底层文件 I/O 操作、嵌入式 Linux 串口应用编程、标准 I/O 编程等内容。

第 6 章为进程/线程编程，主要讲解了 Linux 系统下进程的基本概念与进程管理相关的系统调用、进程间通信的方法和多线程编程的知识。

第 7 章为网络通信相关的 C 语言应用，主要讲解了 Linux 环境下网络编程方法。涉及网络的非阻塞访问、异步处理、多路复用等具体实现。

本书由华清远见嵌入式学院金牌讲师曾宏安编著。本书的完成还要感谢华清远见嵌入式学院，教材内容参考了学院与嵌入式企业需求无缝对接的、科学的专业人才培养体系。同时，嵌入式学院从业或执教多年的行业专家团队也对教材的编写工作做出了贡献，刘洪涛、冯利美、曹忠明、程姚根、季久峰、温尚书、贾燕枫、方琳琳、沈静、冯瑜、杨曼、王利丽、李媛媛、张丹、刘晶晶、王丽丽、谭翠君、关晓强、王彦红等老师在书稿的编写过程中认真阅读了所有章节，提供了大量在实际教学中积累的重要素材，对教材结构、内容提出了中肯的建议，并在后期审校工作中提供了很多帮助，在此表示衷心的感谢。

由于编者水平所限，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，可以发到 www.embedu.org 技术论坛。

编著者

2012 年 1 月

目 录

从实践中学嵌入式 Linux C 编程

第 1 章 嵌入式 Linux C 语言开发工具

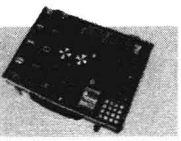
任何应用程序的开发都离不开编辑器、编译器及调试器，嵌入式 Linux 的 C 语言开发也一样，需要拥有一套优秀的编辑、编译及调试工具。

掌握这些工具的使用是至关重要的，它直接影响到程序开发的效率。希望读者通过自己的实践，熟练掌握这些工具的使用。

本章主要内容

- C 语言产生的历史背景
- 嵌入式 Linux 下 C 语言的开发环境
- 嵌入式 Linux 下的编辑器 vi
- 嵌入式 Linux 下的编译器 GCC
- 嵌入式 Linux 下的调试器 GDB
- 嵌入式 Linux 下的工程管理器 make
- 如何使用 autotools 来生成 Makefile
- 嵌入式 Linux 下的综合编辑器 Emacs

1.1 嵌入式 Linux C 语言概述



在嵌入式系统中，应用程序的主体是在宿主机中开发完成的。就嵌入式 Linux 而言，此过程通常在安装有 Linux 的宿主机中完成。

本章中介绍的是嵌入式 Linux 下 C 语言的开发工具，用户在开发时先在 Linux 宿主机中对程序进行调试，然后再进行交叉编译。

1.1.1 C 语言简史

C 语言于 20 世纪 70 年代诞生于美国的贝尔实验室。在此之前，人们编写系统软件时主要使用汇编语言。

汇编语言编写的程序依赖于计算机硬件，其可读性和可移植性都比较差。而高级语言的可读性和可移植性虽然较汇编语言好，但一般高级语言不具备低级语言能够直观地对硬件实现控制和操作而且执行速度快等特点。

在这种情况下，人们迫切需要一种既具有一般高级语言特性，又具有低级语言特性的语言，于是 C 语言就应运而生了。

由于 C 语言既具有高级语言的特点，又具有低级语言的特点，因此迅速普及，成为当今最有发展前途的计算机高级语言之一。C 语言既可以用来编写系统软件，也可以用来编写应用软件。现在，C 语言已经被广泛地应用于计算机、机械、建筑、电子等各个行业。

C 语言的发展历程如下：

- C 语言最初是美国贝尔实验室的 D.M.Ritchie 在 B 语言的基础上设计出来的，此时的 C 语言只是为了描述和实现 UNIX 操作系统的一种工作语言。在一段时间里，C 语言还只在贝尔实验室内部使用。
- 1975 年，UNIX 第 6 版公布后，C 语言突出的优点引起人们的普遍注意。
- 1977 年出现了可移植的 C 语言。
- 1978 年 UNIX 第 7 版的 C 语言成为后来被广泛使用的 C 语言版本的基础，被称为标准 C 语言。
- 1983 年，美国国家标准化协会（ANSI）根据 C 语言问世以来的各种版本，对 C 语言进行发展和扩充，并制定了新的标准，称为 ANSI C。
- 1990 年，国际标准化组织 ISO 制定了 ISO C 标准，目前流行的 C 语言编译系统都是以它为标准的。

1.1.2 C 语言特点

C 语言兼有汇编语言和高级语言的优点，既适合于开发系统软件，也适合于编写应用程序，被广泛应用于事务处理、科学计算、工业控制、数据库技术等领域。

C 语言之所以能存在和发展，并具有强大的生命力，要归功于其鲜明的特点。这些特点是多方面的，归纳如下。

1. C 语言是结构化的语言

C 语言采用代码及数据分隔的方式，使程序的各个部分除了必要的信息交流外彼此独立。这种结构化方式可使程序层次清晰，便于使用、维护以及调试。

C 语言是以函数形式提供给用户的，这些函数可方便地调用，并具有多种循环、条件语句控制程序流向，从而使程序完全结构化。

2. C 语言是模块化的语言

C 语言主要用于编写系统软件和应用软件。一个系统软件的开发需要很多人经过几年的时间才能完成。一般来说，一个较大的系统程序往往被分为若干个模块，每一个模块用来实现特定的功能。

在 C 语言中，用函数作为程序的模块单位，便于实现程序的模块化。在程序设计时，将一些常用的功能模块编写成函数，放在函数库中供其他函数调用。模块化的特点可以大大减少重复编程。程序设计时，只要善于利用函数，就可减少劳动量、提高编程效率。

3. 程序可移植性好

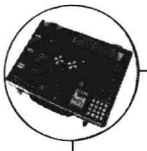
C 语言程序便于移植，目前 C 语言在许多计算机上的实现大都是由 C 语言编译移植得到的，不同机器上的编译程序大约有 80%的代码是公共的。程序不做任何修改就可用于各种型号的计算机和各种操作系统。因此，特别适合在嵌入式开发中使用。

4. C 语言运算符丰富、代码效率高

C 语言共有 34 种运算符，使用各种运算符可以实现在其他高级语言中难以实现的运算。在代码质量上，C 语言可与汇编语言媲美，其代码效率仅比用汇编语言编写的程序的代码低 10%～20%。

1.1.3 嵌入式 Linux C 语言编程环境

嵌入式 Linux C 语言程序设计与在其他环境中的 C 程序设计很类似，也涉及编辑器、编译链接器、调试器及项目管理器的使用。现在我们先对这 4 种工具进行简单介绍，后面会对其一一进行讲解。



1．编辑器

嵌入式 Linux 下的编辑器就如 Windows 下的 Word、记事本等一样，完成对所录入字符的编辑功能，最常用的编辑器有 vi（vim）和 Emacs，它们功能强大，使用方便，本书重点介绍 vi。

2．编译链接器

编译过程包括词法、语法和语义的分析；中间代码的生成和优化；符号表的管理和出错处理等。在嵌入式 Linux 中，最常用的编译器是 GCC 编译器，它是 GNU 推出的功能强大、性能优越的多平台编译器。与其他编译器相比，GCC 编译的代码的执行效率平均要高 20%～30%。

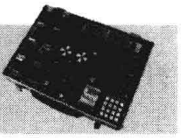
3．调试器

调试器可以方便程序员在程序运行时进行源代码级的调试，但不是代码执行的必备工具。在程序开发的过程当中，调试所消耗的时间远远大于编写代码的时间。因此，有一个功能强大、使用方便的调试器是必不可少的。GDB 可以方便地设置断点、单步跟踪等，足以满足开发人员的需要。

4．项目管理器

嵌入式 Linux 中的项目管理器“make”类似于 Windows 中 Visual C++里的“工程”管理，它是一种控制编译或者重复编译代码的工具。另外，它还能自动管理软件编译的内容、方式和时机，使程序员能够把精力集中在代码的编写上而不是在源代码的组织上。

1.2 嵌入式 Linux 编辑器 vi 的使用



vi 是 Linux 系统的第一个全屏幕交互式编辑工具。它从诞生至今一直得到广大用户的青睐，历经数十年后仍然是人们主要使用的文本编辑工具，足见其生命力之强，其强大的编辑功能可以同任何一款最新的编辑器相媲美。

虽然用惯了 Windows 中的 Word 等编辑器的读者在刚刚接触 vi 时或多或少会有些不适应，但使用过一段时间后，就能感受到它的方便与快捷。

小知识

Linux 系统提供了一个完整的编辑器家族系列，如 Ed、Ex、vi 和 Emacs 等，按功能它们可以分为两大类：行编辑器（Ed、Ex）和全屏幕编辑器（vi、Emacs）。行编辑器每次只能对一行进行操作，使用起来很不方便，而全屏幕编辑器可以对整个屏幕进行编辑，用户编辑的文件直接显示在屏幕上，从而克服了行编辑的那种不直观的操作方式，便于用户学习和使用，具有强大的功能。

1.2.1 vi 的基本模式

vi 编辑器具有 3 种工作模式，分别是命令行模式（Command Mode）、插入模式（Insert Mode）和底行模式（Last Line Mode），各模式的功能区分如下。

1. 命令行模式

在该模式下用户可以输入命令来控制屏幕光标的移动，字符、单词或行的删除，移动复制某区段，也可以进入到底行模式或者插入模式下。

2. 插入模式

用户只有在插入模式下才可以进行字符输入，用户按【Esc】键可回到命令行模式下。

3. 底行模式

在该模式下，用户可以将文件保存或退出 vi，也可以设置编辑环境，如寻找字符串、显示行号等。这一模式下的命令都以“:”开始。

不过在一般使用时，人们通常把 vi 简化成两个模式，即将底行模式也归入命令行模式中。

1.2.2 vi 的基本操作

1. 进入与离开 vi

进入 vi 可以直接在系统提示符下键入 vi ＜文档名称＞，vi 可以自动载入所要编辑的文档或是创建一个新的文档。例如，在 shell 中键入“vi hello.c”（新建文档）即可进入 vi 画面，如图 1.1 所示。

图 1.1 在 vi 中打开/新建文档