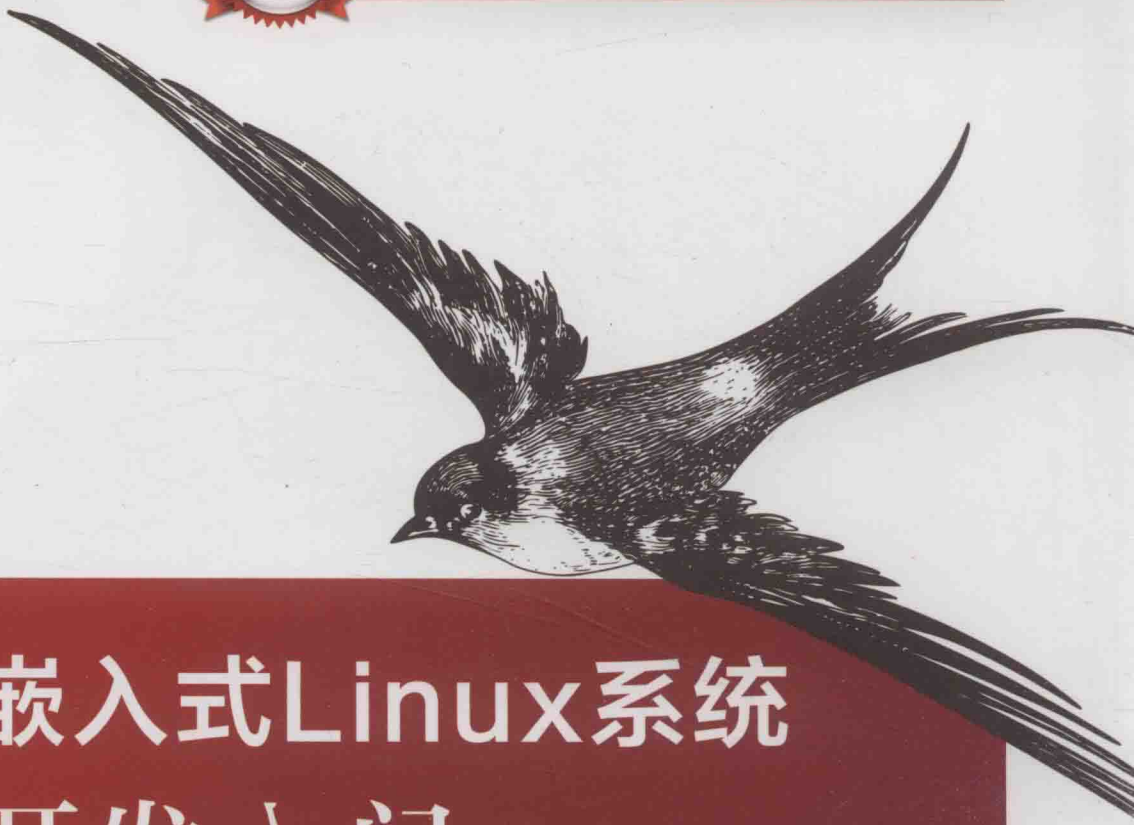
普通高等教育“十三五”规划教材



嵌入式Linux系统开发入门

方 元 编著

An Introduction to the Development of Embedded Linux System

中国工信出版集团
電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

普通高等教育“十三五”规划教材

嵌入式 Linux 系统开发入门

方　元　编著

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

内 容 简 介

本书基于嵌入式 Linux 系统，介绍其软件开发方法，重点介绍多任务、网络和设备驱动的编程。本书分为两个部分。第 1 部分是基础篇（第 1～8 章），内容包括 Linux 系统的基本使用方法、Linux 系统的主要开发工具、文件读写、多任务机制、网络套接字编程、模块与设备驱动、嵌入式 Linux 系统开发、GUI 程序设计初步。第 2 部分是实验篇（第 9～21 章），内容包括实验系统介绍、嵌入式系统开发实验、引导加载器、内核配置和编译、根文件系统的构建、图形用户接口、音频接口程序设计、嵌入式系统中的 I/O 接口驱动、触摸屏移植、Qt/Embedded 移植、MPlayer 移植、GTK+移植、实时操作系统 RTEMS。

本书可作为电子信息、通信、自动化等专业相关课程的教材，也可供相关领域的工程技术人员学习、参考。

图书在版编目(CIP)数据

嵌入式 Linux 系统开发入门 / 方元编著. — 北京：电子工业出版社，2018.5
ISBN 978-7-121-33534-1

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

中国版本图书馆 CIP 数据核字(2018)第 013650 号

策划编辑：王晓庆
责任编辑：郝黎明　　特约编辑：张燕虹
印　　刷：三河市鑫金马印装有限公司
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：16.25　字数：416 千字
版　　次：2018 年 5 月第 1 版
印　　次：2018 年 5 月第 1 次印刷
定　　价：48.00 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 zlts@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)88254113，wangxq@phei.com.cn。

前　言

嵌入式系统几乎是伴随着微处理器同时发展的。根据“维基百科”介绍，诞生于20世纪60年代的阿波罗制导计算机被认为是最早的嵌入式系统之一。自进入21世纪以来，“嵌入式”在计算机领域已成为持续热门的话题。与通用计算机类似，嵌入式系统由软件和硬件组成。随着嵌入式处理器性能的不断提高，许多应用系统的实时性已经不成问题，越来越多的嵌入式系统开始直接使用通用计算机系统的软件。

例如，英国的树莓派基金会采用博通SoC处理器，开发出一系列的树莓派产品。Pi Zero是主频为700MHz的MHz ARM1176jzf-s核，价格定位在5美元；而在2017年年初发布的Pi 3 B+版本，更是具有4核64位CORTEX-A53（ARMv8指令集）的处理器，主频高达1.2GHz，与普通的笔记本电脑的性能相差无几，价格也不过三十几美元。它们都具有高性能的VC-4的图像处理单元（Graphics Processing Unit，GPU），可以流畅地运行一些图形桌面系统，播放高清视频。

在这样的背景下，采用通用计算机系统软件开发嵌入式系统，不仅大大缩短了开发周期、提高了开发效率，系统的可靠性也得到了提升。

在众多的软件中，以Linux为核心的操作系统以及大量的开源软件成为许多嵌入式系统的首选。Linux世界提供大量的自由软件，为开发人员提供了广泛的选择空间，同时也能得到许多无私的帮助。

本书基于嵌入式Linux系统，介绍其软件开发方法，重点介绍多任务、网络和设备驱动的编程。

本书分为基础篇和实验篇两个部分。

第1部分　基础篇

第1章介绍Linux系统的基本使用方法，重点介绍与嵌入式系统开发相关的命令。

第2章介绍Linux系统的主要开发工具，包括编译工具、GNU Make和版本控制系统的使用。

本章最后一节介绍了交叉编译工具的制作过程，供有兴趣的读者参考。

第3章介绍文件读写，重点介绍面向文件描述符的基本系统功能调用，它们是下面几章的基础。

第4章介绍多任务机制，重点介绍在Linux系统中实现多任务的两种主要形式（进程和多线程），以及在多任务程序设计中的一些基础问题。

第5章介绍网络套接字编程，重点介绍以套接字为基础的网络通信程序的基本编程方法。

第6章介绍模块与设备驱动，以个人计算机系统上的一个简单设备为模型，比较系统地介绍了Linux系统中字符设备驱动程序的开发方法。虽然研究对象是通用计算机系统中的一个设备，但其研究方法同样适用于嵌入式Linux中的设备。

第7章介绍嵌入式Linux系统的软件结构，概括地讨论了嵌入式系统的BootLoader、内核布局、文件系统和图形接口几个方面的问题。

第8章简要介绍基于GTK+库的图形接口应用编程基础，通过介绍一些常用组件的功能

和界面设计方法，帮助读者了解 Linux 系统中图形界面的编程风格。在移植了图形库的嵌入式 Linux 系统中，可以比较方便地开发图形化应用程序。

第 2 部分　实验篇

实验篇以美国德州仪器公司的卡片式计算机 Beagle Bone 为实验对象，在此基础上进行嵌入式 Linux 开发，按照从底层基本系统建立到上层应用软件的移植和编写的顺序加以组织。

第 9 章是实验系统基本介绍。

第 10 章介绍嵌入式系统开发实验环境搭建。

第 11 章介绍 BootLoader 的编译和启动设备的制作。

第 12 章、第 13 章分别完成 Linux 内核的编译和根文件系统的制作。至此，一个完全由源码打造的基本 Linux 系统已经建立，它是后续实验的基础。

第 14 章、第 15 章学习嵌入式 Linux 环境下的程序开发方法，其中包括图形界面程序设计和音频接口程序设计。在实验过程中应建立软件层次的概念。

第 16 章学习 Linux 系统中简单设备驱动程序的编写。

第 17～20 章安排了一些软件移植实验，从简单的触摸屏库到较为复杂的 GTK+图形库。通过以上逐层递进的实验，可以掌握在嵌入式平台上实现一种应用系统的方法。

鉴于实时操作系统在嵌入式应用中的地位，第 21 章介绍一款实时操作系统 RTEMS 在嵌入式平台上移植的过程。

实验篇的前 3 章为建立嵌入式 Linux 实验环境做初步准备。后续内容均可在此基础上以具体的应用形式实现，例如多进程的数据采集与回放、多线程的图形应用等。

本书假定读者系统地学习过 C 语言，并对 Linux 操作系统有初步的认识。针对当前嵌入式系统的应用特点，本书重点选取了多任务程序设计、网络通信、设备驱动等几个开发方面进行介绍。书中没有使用过多的 C 语言编程技巧，而着重于功能的实现。本书强调各种工具的使用。一些工具并不仅限于软件开发，在其他场合也能起到极大的帮助作用。例如版本控制系统，在撰写文稿、项目协作等工作中都是非常方便的工具。希望这些工具的介绍能对读者有所帮助。

本书内容作为“嵌入式系统及实验”课程的教材，已在南京大学电子信息类本科教学中实践多年。就笔者的经验来说，基础部分和实验部分每周各用 3 个课时是一个可行的方案。本书为任课教师免费提供配套电子课件、习题参考答案、程序代码等教学资源，请登录华信教育资源网（http://www.hxedu.com.cn）注册下载，也可联系编辑（wangxq@phei.com.cn，010-88254113）索取。

限于笔者的知识水平和认知能力，书中肯定存在不少错误及不当之处，恳请同行专家及读者批评指正。

编　著　者

目　录

第 1 部分　基　础　篇

第 2 部分 实 验 篇

第1部分 基 础 篇

INFORMATION
TECHNOLOGY

第 1 章　Linux 系统的基本使用方法

早期由于处理器硬件性能和资源的限制，嵌入式系统通过专用实时操作系统和精心设计的应用软件保证其性能。随着硬件性能的不断提高，很多民用产品的实时性已经不是问题，而更多关注的是产品的功能和扩展性。由于桌面系统有丰富的软件支持和便捷的开发手段，越来越多的应用开始将桌面系统移植到嵌入式产品中。

本章介绍 Linux 系统的基本使用方法。重点介绍与嵌入式开发关系比较密切的功能。

嵌入式系统发展的历史几乎和微处理器历史一样长。近十几年来，随着处理器性能的不断提高和计算机应用领域的不断扩展，嵌入式成为一个持续热门的词汇。所谓嵌入式系统，是一种由机械和电子系统构成的专用计算机系统，它的研究对象包括嵌入式硬件和软件。目前，98%的微处理器产品是嵌入式系统的组成部分。软件研究对象包括嵌入式操作系统和应用软件。

作为通用计算机操作系统的 Linux 系统，在嵌入式领域中表现同样出色。以手机市场为例，以 Linux 为内核的 Android 系统曾经与 Windows 和 iOS 成三足鼎立，如今在移动终端上则主要由 Android 和 iOS 统治。（据 https://www.netmarketshare.com 统计，2017 年 7 月，使用 Android 系统的手机/平板市场份额占 60%以上。）在大型计算机系统中，Linux 更是出类拔萃。据www.top500.org 的数据显示，在全世界性能最高的超级计算机 500 强中，Linux 系统自 1998 年 6 月开始进榜（此前一直是 UNIX 的天下）以来，占比持续上升，至 2015 年 6 月，使用 Linux 各种发行版的超级计算机达到 489 台。在最近几次的更新榜单中，使用非 Linux 系统的超级计算机仅占个位数。

在通用计算机系统中，Linux 系统和 UNIX 系统非常相似，而与 Windows 系列操作系统的特征则相差比较大。Linux 除了作为嵌入式系统的运行环境以外，也是理想的嵌入式开发环境。熟练掌握 Linux 系统是嵌入式 Linux 开发的基础。

1.1　Linux 系统的使用环境

软件是计算机系统的重要组成部分。软件包括程序、数据、文档，它们存储在计算机的存储设备上，存储设备通过文件系统组织。这种文件系统又称为“分区”，在 Windows 操作系统中，访问分区通过盘符“C:”、“D:”这样的符号。每个分区又有一些目录，Windows 称之为“文件夹”（Folders）。目录之下可以有文件和子目录，目录和目录之间用反斜线“\”分隔。

Linux 系统没有盘符的概念。操作系统内核启动的最后阶段，会挂载一个分区作为它的根文件系统，它可以是一个磁盘分区，也可以是一块内存，这块内存被组织成某种分区的格式。根文件系统中有操作系统赖以正常运行的基本程序。其他分区可以在系统启动后，根据需要，通过“mount”命令挂载到指定目录上。一经挂载，访问这个分区的方式和访问目录的形式是完全一样的。Linux 系统中用斜线“/”分隔目录层次。顶层的目录被称为“根目

录”。Linux 不使用反斜线分割目录。反斜线在命令行中的功能是转义符，这与它在许多编程语言及命令中的功能是一致的。

1.1.1　Linux 系统的目录结构

Linux 系统的目录结构遵循文件系统结构层次标准 FHS（Filesystem Hierarchy Standard）。该标准普遍被类 UNIX 系统采用，它定义了目录结构和内容。图 1.1 是 Linux 系统的主要目录。

图 1.1　Linux 系统的主要目录

Linux 依据不同程序在系统中的地位，采用一级目录结构、二级目录结构（/usr）、三级目录结构（/usr/local）管理。不同程序在系统中的地位不同，对系统的影响不同，限制着用户的使用权限。供普通用户使用的命令放在/bin（binary）和/usr/bin 目录里；供超级用户使用的命令放在各级结构的 sbin（system binary）目录里，普通用户要么无权使用这些命令，要么命令的功能会受到一定限制。

1.1.2　Linux 系统的用户

UNIX 类的操作系统对用户有比较严格的管理策略。超级用户（super user，用户名为“root”，用户 ID 为 0）可以运行系统中的所有程序，支配系统的所有资源。拥有这个权限

的通常是系统的管理员。管理员可以为其他使用者创建账户，这些账户通常拥有普通用户的权限。普通用户可以正常使用计算机，但不能对计算机的设置进行改变。根据使用者在系统中的关系，管理员还可以对他们进行分组。在以组为单位的项目开发中，这种分组关系既满足了资源共享要求，同时也隔离了与其他项目之间的交叉。

有些 Linux 的发行版还有为临时使用计算机的人创建“访客用户”的做法。访客用户只有有限的计算机支配资源，其数据也不会长期保存。

供个人使用的计算机中，尽管用户可以以超级用户身份登录使用，但出于安全性方面的考虑，仍然建议用户日常操作以普通用户的身份完成。超级用户的误操作会给系统造成较大的损害，此外超级用户的使用环境会给恶意软件带来方便。很少听说 Linux 操作系统有病毒泛滥，对用户权限的限制可能是一个重要的原因。如果发现有这样的软件，非超级用户无法正常使用（通常这是一个可能改变系统设置的软件）。这时可以通过临时切换身份（使用“sudo”命令）的方式运行这个软件。如果这个问题频繁出现，要么是用户权限设置有问题，要么是软件本身的设计问题。前者应由管理员调整用户权限，后者的问题可以向软件开发者反映。

作为嵌入式系统的开发对象，使用 root 身份是正常的。因为系统运行伊始仅有这个权限，甚至还没建立用户。但作为开发环境使用的计算机，以上原则仍然有效。

1.2 命令行工作方式

在 Linux 上进行软硬件开发，命令行接口（Command-Line Interface，CLI）是主要的工作方式。虽然 Linux 也有集成开发环境，但命令行能调动的资源更多，命令行操作方式更加灵活。在网络环境下，使用图形界面可能会对服务器和客户端都有较高的要求，而命令行方式则基本上不受限制。开发嵌入式 Linux，初期的目标系统肯定不具备图形工作环境，甚至终生都不会为它移植图形界面，这时在它上面的操作只能通过命令行方式完成。命令行方式的唯一缺点是需要记住很多命令的用法。

1.2.1 终端

提供命令行工作环境的设备称为“shell”（有时也称为“终端”）。目前使用最多的是 bash（Bourne-Again SHell）。shell 可以运行系统的所有程序，包括图形界面程序。多数情况下，在终端上运行程序等效于在图形界面下用鼠标单击程序的图标。而在使用字符界面程序时，可以看到程序正常的运行过程，比如用 C 语言函数“printf( )”在标准设备上输出的信息，这在图形方式下有时是做不到的。此外，如果是可以带参数执行的命令，对于在图形操作方式下单击鼠标如何输入参数，目前也没有通用的解决方案。

在 shell 中执行命令或程序，就是直接在提示符下用键盘输入程序的名字，以回车确认。一般格式如下：

```
$ command [options] parameters
```

“command”是程序名，[①]绝大多数这样的程序在“/bin”、“/sbin”、“/usr/bin”、

① “$”表示普通用户使用 shell 的提示符，无须用键盘输入。超级用户的提示符是“#”，区别于普通用户，起到一定的警示作用。本书将以这两个符号区分命令的权限。

“/usr/sbin”目录中以可执行文件的形式存在。Linux 系统中一个重要的环境变量“PATH”包含了这些目录的列表。如果想执行的程序不在这个目录列表中，则必须明确指明这个程序所在目录。例如：

```
$ ./hello
```

表示执行当前目录下一个名为“hello”的程序。“./”表示当前目录。

命令可以带有一些选项，选项前面通常会使用“-”或“--”以便和命令的参数相区别，有的命令选项甚至不使用“-”。具体使用哪一种方式，取决于该程序作者的风格。

Linux 系统中，命令中的选项、参数，包括命令本身，都严格区分字母大小写，除非程序内部对字母大小写有专门的合并处理。[①]

1.2.2　目录操作

shell 中访问目录的命令主要有转移目录、创建和删除目录等操作。

- 列目录、文件清单“ls”。该命令不带选项时仅列出文件名。用“ls -l”可以看到列出某个目录下的文件较完整的属性：

```
$ ls -l
drwxr-xr-x 19 harry harry  4096    6 月 14 12:47 docs
drwxr-xr-x 23 harry harry  4096    10 月 20 2016 programs
drwx------  3 harry harry  4096    6 月 21 14:28 videos
-rw-rw-r--  1 harry harry  1830    11 月 30 2016 problem1_3.m
```

上面列出的文件清单中，每一行描述了一个文件的主要特性。其中，第一个字段是文件的访问权限属性，它显示了该文件的性质和读写权限。第一个字母“d”表示它是一个目录文件（Directory），普通文件则用“-”表示。后面紧跟的 9 个字母中，每三个为一组，依次表示文件拥有者、文件属组以及其他人对该文件的访问权限。访问权限的三个字母 r、w、x 分别表示读（Read）、写（Write）和运行（eXecute）许可。Linux 系统的文件可执行属性就体现在“x”标志位上，而与其名称及后缀无关。目录的“x”标识表示该目录可以访问。对应位置的“-”表示其权限的缺失。属性字段后面的数字表示该文件的链接数。接下来的两个名字分别表示文件拥有者和属组。上面的清单显示出，docs 和 programs 目录允许所有人访问，但只有 harry 本人有修改权限，而 videos 只有 harry 本人有访问权限。文件 problem1_3.m 不是一个可执行文件，允许 harry 及 harry 组员读写，其他人只能读不能修改。

- 转移目录“cd”，这是一条 shell 的内部命令。从当前的工作目录转移到另一个目录的命令格式是[②]

```
$ cd pathname
```

表示路径的方式有两种：从根目录开始，将各级目录名按顺序用斜线分隔，这种形式称为“绝对路径”；以当前目录为起点，向上或向下达到目标目录的形式称为“相对路径”。

---

① 归根到底，这种大小写敏感的特性不是由操作系统决定的，而是由是由文件系统决定的。如果使用了文件名不区分字母大小写的文件系统（如 FATFS），命令行操作会在一定程度上失去大小写的区别意义。

② 目录（directory）有时又被称为“路径”（path）。

图 1.2 中，假设用户“harry”想从当前目录“docs”转移到“videos”目录下工作，他可以采用绝对路径操作方式“cd /home/harry/videos”，也可以采用相对路径操作方式“cd ../videos”。“../”表示上一级目录。采用后一种方式时，转移目录的操作方式与当前位置有关。

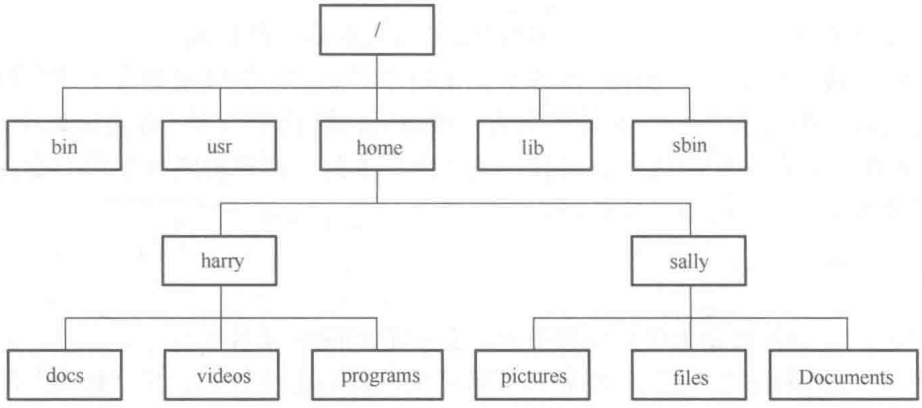


图 1.2 相对路径和绝对路径

“cd”命令不带参数时，将回到用户的主目录（环境变量 HOME 指向的目录）。

“cd -”表示回到上一次所在的工作目录。

- 创建目录 mkdir。用户 harry 想在 programs 下面创建一个新目录“proj1”，再在“proj1”下面创建一个子目录“new”。他可以先用“mkdir”创建“proj1”，再转移到这个目录下创建“new”；也可以：mkdir -p proj1/new 一次完成。选项“-p”表示：如果所创建目录的上层目录（父目录）还不存在，就同时创建。
- 删除一个空目录，使用“rmdir pathname”。非空目录不能使用 rmdir 删除。
- pwd 可以打印当前的绝对路径名。此命令在一般的操作过程中作用不大，因为通常 shell 的提示符中已经显示了路径名。它主要用在软件开发过程中不便于人工干预的场合。

1.2.3 文件操作

文件操作包括文件复制、改名、移动、删除等。对文件的操作命令，多数也可以对目录操作。

- 修改属性 chmod。拥有文件操作权限的用户可以根据需要修改文件的权限属性。例如，harry 将自己辖下的 videos 目录开放给其他用户，可以使用命令“chmod 777 videos”。其中的数字是八进制，与属性字段除第一位以外的每一位相对应。
- 文件复制 cp。“cp oldfile newfile”，将 oldfile 复制一份到 newfile。如果最后一个参数是目录，则表示把前面几个复制到该目录下。当复制的源包含目录时，需要使用选项“-r”。
- 删除文件 rm。被删除的文件无法通过正常操作恢复。这与在桌面环境中将文件移除至垃圾桶的行为不同。删除目录同样也需要选项“-r”。
- 文件改名 mv。“mv oldfile newfile”，旧文件名被新文件名替代。如果需要一次性给多个文件改名，Linux 没有原生的批量重命名命令，此项功能可以通过一些命令的组合实现。1.4.3 节提供了一种方案。

当最后一个参数是目录时，mv 操作的结果是将前面所列的文件和目录移至该目录下。

- 链接命令 ln。链接的结果有点像文件复制，但本质上是不一样的。链接的文件和原始文件只有名字的差别，数据在存储设备上只存储了一份。带有选项“-s”创建的链接又称为软链接或符号链接，它比较像 Windows 系统的“快捷方式”，链接文件依赖于原始文件的存在。当原始文件被移动或删除后，软链接文件就成为“死链”。
  硬链接则是所有链接文件都指向相同的数据区，每个链接文件的地位是平等的，不存在依赖关系。只有当最后一个链接被删除后，存储空间才被真正释放。硬链接只能是同一个分区里的文件（不包括目录），软链接则无此限制。
  链接功能由文件系统支持，只有支持 inode 型或 vnode 型结构的文件系统才能创建链接。（inode 普遍见于 Linux 系统的文件系统，vnode 主要见于 BSD 系统。）
  链接在 Linux 系统中使用非常普遍。动态链接库通过链接的形式允许多个版本共存。Linux 有一个基本软件包 BusyBox，里面包含了上百个 Linux 的基本操作命令，这些命令都指向同一个文件 busybox。如果要升级 BusyBox，只需要更换这一个文件就可以了。
- 查找文件 find。find 查找文件方式很灵活，通过不同选项，可以根据文件名、文件大小、文件最后修改时间等多种属性查找。对于找到的文件，还可以使用命令直接对这些文件进行操作。例如，查找当前目录下（包括子目录）超过 10KB 的文件并复制到 harry 的 videos 目录：

```
$ find . -size +10K -exec cp {} /home/harry/videos \;
```

  解释：选项“-size”告诉 find 按文件大小查找，“-exec”告诉 find 找到的文件后执行后面的命令，“{}”匹配找到的每一个文件；命令行中的分号“;”是 shell 中顺序执行一系列命令的分割符，为避免被 shell 误解为是“find”命令的分隔符，这里用“\”转义。

对文件和目录的操作可以使用通配符。通配符是用于代替某一个或一串具体字符的特殊字符。常用的通配符有“*”、“?”和“[...]”。例如，图 1.2 中，harry 要把当前目录（假设是 videos）里所有“.c”和“.h”结尾的文件复制到 programs 目录中，他可以这样操作：

```
$ cp *.c *.h ../videos/
```

这里的“*”用于替代任意字符串，它省去了对逐个文件操作的烦琐过程。通配符“?”用于替代任意单个字符；“[...]”中的符号相当于一个清单，例如“[Cc]”表示“C”或“c”，“[a-z]”表示所有小写字母。

1.2.4　浏览文件

除了利用编辑器观察文件内容以外，还可以用一些轻量级工具直接在终端上查看文件内容：

- more。此命令将长文件分屏显示，每满一屏暂停一下，等待用户敲键以后再继续上滚一屏。
- head/tail。用于浏览文件头部或尾部。选项“-n”用于指定行数。默认的是 10 行；也可以用“-c”选项指定以字节为单位。tail 还有一个有用的选项“-f”（follow 的意思），当浏览一个不断变化着的文件（例如某个软件的日志文件）时，可以跟踪显示这个文件。
- cat。此命令可以将文件和标准输入设备合并，送到标准输出设备上。