



高职高专“十二五”规划教材

陈长顺 主编

嵌入式

Linux应用开发精解



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS



高职高专“十二五”规划教材

嵌入式 Linux 应用开发精解

陈长顺 主编



北京航空航天大学出版社

内 容 简 介

本书以项目为主线,全面介绍了嵌入式 Linux 系统开发技术的主要应用领域,包括 9 个项目,分别是构建嵌入式 Linux 开发环境、开发简单应用程序、开发设备驱动程序、实现图形用户界面程序、开发多线程程序、开发串口通信程序、开发多媒体程序、开发数据库程序和开发网络应用程序。每个项目以企业实战为主线,包含项目需求、项目设计、项目实施和项目小结等主要环节,并以知识背景作为项目基础,设置工程实训和拓展提高环节,用以巩固实训成果,强化能力养成,激发创新思维。内容编排由浅入深,通俗易懂,注重整体,兼顾一般,利于读者理解。

本书既可作为高职院校计算机、物联网、电子工程和机电一体化等相关专业“嵌入式 Linux 应用开发”课程的教材,也可用作各类培训机构的培训教材,还可作为嵌入式 Linux 系统开发专业人员和业余爱好者的参考书和工具书。书中提供的项目源代码稍加移植、修改、扩充和组合,即可构建实用的嵌入式 Linux 系统。

图书在版编目(CIP)数据

嵌入式 Linux 应用开发精解/陈长顺主编. --北京
:北京航空航天大学出版社, 2013. 1
ISBN 978-7-5124-1051-0

I. ①嵌… II. ①陈… III. ①Linux 操作系统—程序
设计 IV. ①TP316.89

中国版本图书馆 CIP 数据核字(2013)第 019714 号

版权所有,侵权必究。

嵌入式 Linux 应用开发精解

陈长顺 主编

责任编辑 何 献 王国兴

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:emsbook@gmail.com 邮购电话:(010)82316936

涿州市新华印刷有限公司印装 各地书店经销

*

开本:710×1 000 1/16 印张:15.25 字数:325 千字

2013 年 1 月第 1 版 2013 年 1 月第 1 次印刷 印数:3 000 册

ISBN 978-7-5124-1051-0 定价:34.00 元

若本书有倒页、脱页、缺页等印装质量问题,请与本社发行部联系调换。联系电话:(010)82317024

目 录

项目 1 构建嵌入式 Linux 开发环境	1
1.1 知识背景	1
1.1.1 嵌入式系统的组成	1
1.1.2 嵌入式系统开发板	2
1.1.3 交叉编译工具链	3
1.1.4 交叉编译环境的组成	4
1.1.5 Linux 服务	5
1.2 项目需求	6
1.3 项目设计	6
1.3.1 开发板的选定	6
1.3.2 开发方案的确定	7
1.4 项目实施	7
任务一:组建开发平台	7
任务二:配置超级终端	8
任务三:安装与配置 DNW	9
任务四:安装虚拟机	11
任务五:安装 Linux 操作系统	13
任务六:实现 Windows 共享	17
任务七:配置 NFS 服务	19
任务八:配置 FTP 服务	20
任务九:安装与配置交叉编译工具链	22
1.5 项目小结	23
1.6 工程实训	24
实训目的	24
实训环境	24

实训内容	24
实训步骤	24
1.7 拓展提高	25
思 考	25
操 作	25
项目2 开发简单应用程序	26
2.1 知识背景	26
2.1.1 嵌入式软件系统的组成	26
2.1.2 BootLoader 的功能与使用	27
2.1.3 内核移植与下载	31
2.1.4 根文件系统的建立	32
2.1.5 嵌入式应用程序开发	35
2.2 项目需求	39
2.3 项目设计	39
2.4 项目实施	41
任务一：下载 BootLoader	41
任务二：移植内核	43
任务三：移植根文件系统	44
任务四：编写“Hello World”应用程序	45
任务五：实现开机自启动“Hello World”	46
2.5 项目小结	47
2.6 项目实训	48
实训目的	48
实训环境	48
实训内容	48
实训步骤	48
2.7 拓展提高	49
思 考	49
操 作	49
项目3 开发设备驱动程序	50
3.1 知识背景	50
3.1.1 设备驱动程序的概念	50
3.1.2 设备驱动程序的结构	55
3.1.3 设备驱动开发的 API 函数	59

3.1.4 设备驱动程序的调试	67
3.2 项目需求	68
3.3 项目设计	69
3.3.1 理解驱动开发的本质	69
3.3.2 驱动开发的一般流程	69
3.3.3 内核态 Hello World 驱动程序设计	70
3.3.4 LED 驱动程序设计	70
3.3.5 按键驱动程序设计	72
3.4 项目实施	73
任务一：实现内核态的驱动程序	73
任务二：实现 LED 驱动程序	76
任务三：实现键盘驱动程序	81
3.5 项目小结	88
3.6 工程实训	89
实训目的	89
实训环境	89
实训内容	89
实训步骤	89
3.7 拓展提高	90
思 考	90
操 作	91
项目 4 实现图形用户界面应用程序	98
4.1 知识背景	98
4.1.1 Qt 简介	98
4.1.2 Qt 开发环境	100
4.1.3 Qt 编程机制	101
4.2 项目需求	106
4.3 项目设计	107
4.3.1 构建 Qt 集成开发环境	107
4.3.2 Qt 应用程序开发流程	108
4.3.3 Qt 应用程序开发方法	108
4.3.4 关闭系统的实现	108
4.3.5 菜单命令的实现	108
4.3.6 系统标准对话框的使用	109
4.3.7 部署 Qt 应用程序	110

目 录

4.3.8 实现 Qt 应用程序的开机自启动	110
4.4 项目实施	110
任务一：建立 Qt 开发环境	110
任务二：建立 Qt 运行环境	113
任务三：编写 Qt 下的“Hello World”程序	114
任务四：实现开机自启动 Hello_Qt4	123
4.5 项目小结	123
4.6 工程实训	124
实训目的	124
实训环境	124
实训内容	124
实训步骤	125
4.7 拓展提高	125
思 考	125
操 作	125
项目 5 开发多线程程序	126
5.1 知识背景	126
5.1.1 进程的概念	126
5.1.2 线程的概念	127
5.1.3 Qt 中的线程类	127
5.1.4 用户自定义事件在多线程编程中的应用	128
5.1.5 利用定时器机制实现多线程编程	130
5.1.6 利用 QProcess 实现多线程编程	131
5.2 项目需求	132
5.3 项目设计	132
5.3.1 LED 控制原理	132
5.3.2 LED 开发控制	133
5.3.3 按钮控制的灵活性	133
5.3.4 不同线程之间通信的实现	133
5.4 项目实施	134
任务一：建立项目文件	134
任务二：设计程序界面	134
任务三：修改系统主界面类文件	135
任务四：添加 LED 控制子线程类定义文件	135
任务五：添加 LED 控制子线程类实现源文件	136

任务六：编写主线程类实现源文件	137
任务七：实现系统主函数	139
任务八：在 PC 中编译调试	140
任务九：交叉编译成 ARM 可执行文件	140
任务十：下载到目标机	140
5.5 项目小结	142
5.6 工程实训	142
实训目的	142
实训环境	142
实训内容	142
实训步骤	143
5.7 拓展提高	143
思考	143
操作	144
项目 6 开发串口通信应用程序	145
6.1 知识背景	145
6.1.1 串行通信原理	145
6.1.2 串行通信协议	148
6.1.3 Qt 串行通信架构	150
6.2 项目需求	153
6.3 项目设计	154
6.3.1 串口设备的打开	154
6.3.2 串口设备的读/写	154
6.4 项目实施	156
任务一：建立项目文件	156
任务二：设计程序界面	156
任务三：编写头文件	157
任务四：编写源文件	158
任务五：修改系统主函数文件	160
任务六：编译、下载与调试	160
6.5 项目小结	163
6.6 工程实训	163
实训目的	163
实训环境	163
实训内容	163

实训步骤.....	164
6.7 拓展提高	164
思 考.....	164
操 作.....	164
项目 7 开发多媒体应用程序	165
7.1 知识背景	165
7.1.1 Qt 的画图机制	165
7.1.2 图像框架结构	166
7.1.3 音、视频播放机理	167
7.2 项目需求	169
7.3 项目设计	169
7.3.1 界面设计	169
7.3.2 图片浏览	169
7.3.3 动画播放	171
7.3.4 音、视频播放.....	173
7.4 项目实施	175
任务一：建立项目文件	175
任务二：设计系统用户界面	175
任务三：编写头文件	176
任务四：编写系统源文件	178
任务五：修改系统主函数文件	185
任务六：编译、下载与调试	185
7.5 项目小结	186
7.6 工程实训	187
实训目的.....	187
实训环境.....	187
实训内容.....	187
实训步骤.....	187
7.7 拓展提高	188
思 考.....	188
操 作.....	188
项目 8 开发数据库应用程序	189
8.1 知识背景	189
8.1.1 嵌入式数据库简介	189

8.1.2	SQLite 简介	190
8.1.3	SQLite 的 Shell 命令	190
8.1.4	Qt 下数据库编程	192
8.2	项目需求	195
8.3	项目设计	195
8.3.1	界面设计	195
8.3.2	数据库的建立	196
8.3.3	按钮功能的实现	196
8.4	项目实施	196
	任务一：建立项目文件	196
	任务二：设计系统用户界面	197
	任务三：编写头文件	198
	任务四：编写源文件	198
	任务五：修改系统主函数文件	201
	任务六：编译、下载与调试	202
8.5	项目小结	204
8.6	工程实训	204
	实训目的	204
	实训环境	204
	实训内容	204
	实训步骤	205
8.7	拓展提高	205
	思 考	205
	操 作	206
项目 9	开发网络应用程序	207
9.1	背景知识	207
9.1.1	OSI 网络互联参考模型	207
9.1.2	网络协议	209
9.1.3	QtNetwork 模块	212
9.2	项目需求	212
9.3	项目设计	213
9.3.1	界面设计	213
9.3.2	服务器端通信方式设计	214
9.3.3	客户端通信方案设计	215
9.4	项目实施	216

任务一：建立 TcpServer 项目文件	216
任务二：编写项目源文件	216
任务三：添加 TCP 服务器类	218
任务四：添加 tcpClientSocket 服务器类	220
任务五：编译、调试服务器	221
任务六：建立 TcpClient 项目文件	222
任务七：编写项目源文件	223
任务八：编译、调试客户端程序	227
9.5 项目小结	227
9.6 工程实训	228
实训目的	228
实训环境	228
实训内容	228
实训步骤	228
9.7 拓展提高	229
思 考	229
操 作	229
参考文献	230

构建嵌入式 Linux 开发环境

学习目标:

- ▶ 了解嵌入式 Linux 开发环境的组成;
- ▶ 熟悉嵌入式 Linux 开发环境的构建过程;
- ▶ 掌握嵌入式 Linux 开发环境的使用方法。

嵌入式 Linux 系统开发是一个软、硬件协同设计的过程,在开发嵌入式 Linux 系统之前,首先需要构建方便、适用、高效的开发环境。一个性能良好的开发环境,可使开发工作事半功倍。本项目实施的目标在于,理解嵌入式 Linux 开发环境的基本组成,熟悉嵌入式 Linux 开发环境的构建过程,掌握嵌入式 Linux 开发环境的使用方法。

1.1 知识背景

1.1.1 嵌入式系统的组成

嵌入式系统是以应用为中心,以计算机技术为基础,软、硬件可裁减,满足应用系统对功能、可靠性、成本、体积、功耗等严格要求的专用计算机系统。嵌入式系统一般由硬件和软件两部分组成,硬件通常包含嵌入式微处理器、存储器和外围接口电路,软件通常由引导程序、操作系统和应用程序组成。一个典型的嵌入式系统如图 1-1 所示。

随着芯片技术的不断发展,嵌入式微处理器品种已有数百种,主频也越来越高,通常主频都在 200 MHz 以上,有的甚至高达 1 GHz。多处理器、多核处理器平台也逐渐应用在嵌入式领域,不过现在大量使用的还是 32 位单处理器组成的平台。

在嵌入式系统中,存储器负责保存程序和数据。与 PC 机有所区别的是,为了保持嵌入式系统的微型化,存储器通常由半导体集成电路来实现。

嵌入式系统在一个应用系统中处于核心位置,负责检测外部输入信号,并根据预先存储在存储器中的处理方案对数据进行处理,最后根据处理结果或显示驱动相应的执

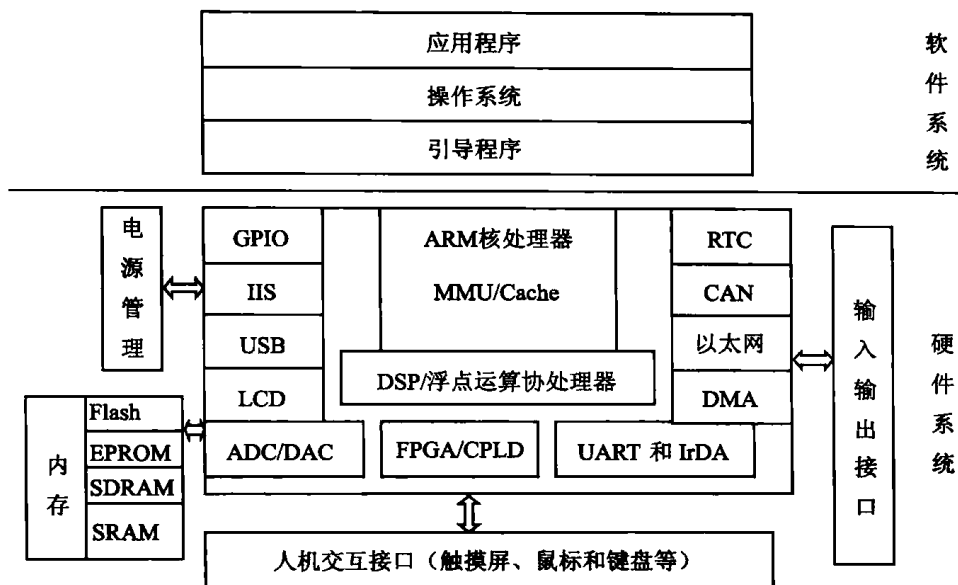


图 1-1 嵌入式系统的组成

行机构。这些对外部信息的检测和对外部机构的驱动，都是通过接口电路加以实现的。

仅有嵌入式硬件系统是无法实现智能化功能的。要使嵌入式系统在改造世界中发挥重要作用，嵌入式软件是必不可少的。

嵌入式软件系统包括引导程序、操作系统和应用程序 3 个层次。

促使嵌入式系统启动并进入正常工作状态的程序是引导程序。嵌入式引导程序类似于 PC 机中的 BIOS 程序，在嵌入式系统上电后，首先运行嵌入式引导程序，检查系统硬件的基本情况，并将控制权转交给操作系统。

操作系统是嵌入式系统软件的核心。基于源代码开放的 Linux 操作系统经过长期运行考验，日趋成熟，其相关的标准和软件开发方式已被用户普遍接受，同时积累了丰富的开发工具和应用软件资源，已成为目前嵌入式系统中的主要操作系统。

应用程序运行在嵌入式操作系统之上，一般情况下与操作系统是分开的。当处理器上带有 MMU(Memory Management Unit, 存储器管理单元)时，它可以从硬件上将应用程序和操作系统分开编译和管理。这样做的好处就是系统安全性更高，可维护性更强，更有利于各功能模块的划分。很多情况下在没有 MMU 的处理器时，如 ARM7TDMI，经常将应用程序和操作系统编译在一起运行。对于开发人员来说，操作系统更像一个函数库。

1.1.2 嵌入式系统开发板

嵌入式系统是嵌入式硬件和软件的有机结合体，嵌入式系统开发离不开具体的硬件平台。硬件平台通常结合具体应用进行设计，学习者一般通过购买开发板进行实验开发。本书选择广州友善之臂计算机科技有限公司基于 ARM S3C6410 微处理

器的嵌入式系统开发板 Tiny6410, 进行嵌入式 Linux 系统开发实验与实训。虽然如此, 许多其他开发板的设计理念和开发方法相似, 只是地址分配、接口方式和存储处理稍有区别而已。因此, 完成本书体系的实验与实训也可以选用诸如上海双实科技、北京博创科技、保定飞凌等相似的任何一款开发板, 当实例程序中与硬件相关的参数不一致时, 读者只需结合具体开发板的参数进行微调即可。

Tiny6410 由核心板和扩展板组成, 元器件分布如图 1-2 所示。

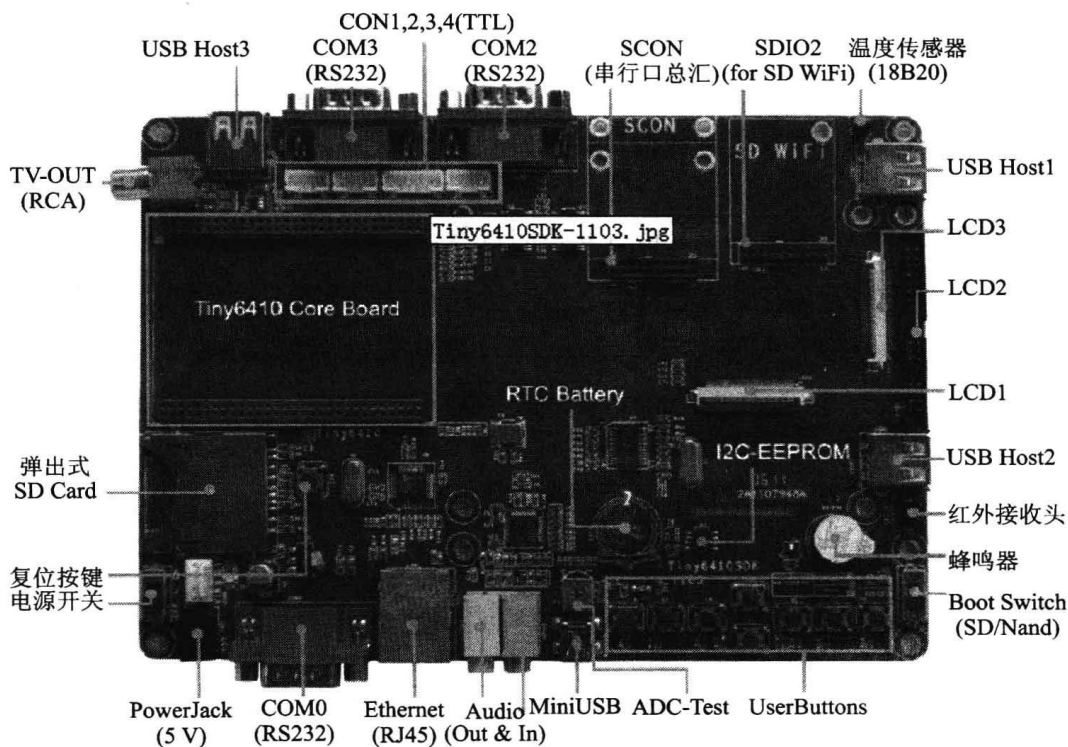


图 1-2 Tiny6410 元器件分布图

核心板采用高密度 6 层板设计, 尺寸为 64×50 mm, 它集成了 256 MB DDR RAM, 2 GB SLC Nand Flash 存储器, 采用 5 V 供电, 在板实现 CPU 必需的各种核心电压转换, 还带有专业复位芯片, 通过 2.0 mm 间距的排针, 引出各种常见的接口资源, 以供不打算自行设计 CPU 板的开发者进行快捷的二次开发使用。

扩展板具有 3 个 LCD 接口、4 线电阻触摸屏接口、100M 标准网络接口、标准 DB9 五线串口、Mini USB 2.0 接口、USB Host 1.1 接口、3.5 mm 音频输入/输出口、标准 TV-OUT 接口、SD 卡座、红外接收等常用接口; 在板的还有蜂鸣器、IIC-EEPROM、备份电池、AD 可调电阻、8 个中断式按键等。

1.1.3 交叉编译工具链

在 Linux 平台下, 要为开发板编译 BootLoader、内核、根文件系统、图形用户界

项目 1 构建嵌入式 Linux 开发环境

面、应用程序等,均需要交叉编译工具链。

交叉编译就是在一种平台上编译出能运行在体系结构不同的另一种平台上的程序,对嵌入式 Linux 系统开发而言,即在 PC 平台上编译出能在以 ARM 为内核的平台上可运行的程序。相对于交叉编译,平常所做的编译叫本地编译,也就是在当前平台编译,所得目标程序也是在本地上运行。用来编译这种跨平台程序的编译器称为交叉编译器。由于一般的嵌入式系统存储容量有限,通常都需要在性能良好的 PC 上建立一个用于目标机的交叉编译工具链,用该交叉编译工具链在 PC 上编译目标机上可运行的程序。

交叉编译工具链通常是一个由编译器、链接器和解释器组成的综合开发环境,主要由 Binutils、gcc 和 glibc 组成,有时出于减小 libc 库大小的考虑,也可以用 c 库去代替 glibc。

建立交叉编译工具链是一个相当复杂的过程。但因 Linux 是一个开源系统,所以通常在网上有许多交叉编译工具链可以下载。一般情况下,可以从网上下载一个基本的交叉编译工具包,然后解压到宿主机的 Linux 之中,再根据开发需要进行简单配置和编译,就可以快速建立适合于开发者的交叉编译工具链。

1.1.4 交叉编译环境的组成

基于 Linux 的开发环境在开发以 Linux 作为操作系统的嵌入式应用中具备得天独厚的优势,因而得到许多开发者的认同。开发嵌入式 Linux 软件系统有多种方案,其典型开发环境架构如图 1-3 所示。

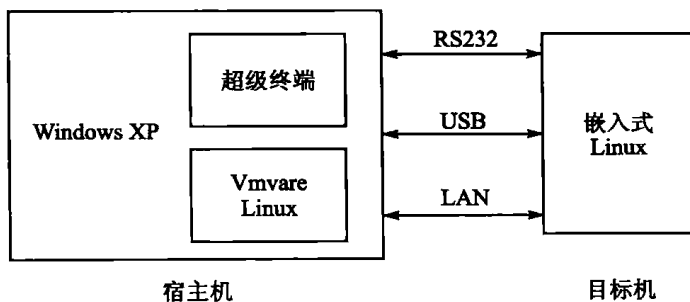


图 1-3 基于 Linux 的开发环境架构示意图

在这一开发环境中,应用程序的编写、编译和链接在虚拟机 Linux 系统中进行。由于嵌入式系统的操作系统是 Linux,应用程序的开发也在相同系统中,因而,处理问题的思路、方法最接近目标机。

为了能把在宿主机上编译好的应用程序下载到目标机进行调试,同时避免反复烧写目标机 Flash,在这一开发架构中,通过网络线把宿主机与目标机相连,使两者在同一网段中,进而通过网络配置,在宿主机上建立共享目录,使目标机可以共享宿主机上编译好的目标程序。这样,在目标机上调试程序时,就不必把宿主主机上的程序

下载到目标机上,其宿主机上的共享目录已经可以看成目标机的存储器了。

经调试通过的应用程序最终需要从宿主机烧写到目标机的 Nand Flash 之中,借助于开发板的 BootLoader 命令和宿主机中的 DNW 工具,通过 USB 线缆连接,可以方便地进行传输。利用这一方案,也可以把 Linux 内核和根文件系统经裁减后下载到开发板之中。

1.1.5 Linux 服务

为了实现宿主机中在 Windows 和 Linux 之间进行文件的共享与传输,同时实现宿主机和目标机 Linux 之间的文件共享与通信,需要借助 Linux 的网络服务功能。Linux 是一个网络操作系统,除了具有稳定性和安全性等优良特点外,还提供了许多网络服务,这给嵌入式系统开发带来很大的方便。在进行嵌入式 Linux 应用开发时,常用 Samba、NFS 和 FTP 服务实现宿主机、目标机之间的通信与共享。

1. Samba 服务

当局域网中既有安装 Windows 的计算机,又有安装 Linux 的计算机时,架构 Samba 服务器可实现不同类型计算机之间文件共享。在嵌入式系统开发中,常用 Samba 服务来实现在 Windows 和 Linux 虚拟机之间进行文件传输。

SMB(Server Message Block,服务信息块)协议是实现网络上不同类型计算机之间文件和打印机共享服务的协议。Samba 服务的工作原理是:在 TCP/IP 协议之上运行 SMB 和 NetBIOS 协议,利用 NetBIOS 名字解析功能让 Linux 计算机可以在 Windows 计算机的网上邻居中看到,从而实现 Linux 计算机与 Windows 计算机之间相互访问共享文件的功能。

2. NFS 服务

NFS(NetWork File System,网络文件系统)是由 SUN 开发的一种基于网络的文件共享协议,它使不同系统平台上的用户通过网络能够共享同一个文件系统。用户通过 NFS 访问其他系统平台的文件和访问本地文件一样便捷而不会感到任何区别。

NFS 基于 RPC 机制,分为 NFS 服务器端和客户端两部分,使用星形拓扑结构连接。NFS 服务器端提供文件系统共享,客户端能够挂载服务器文件系统并进行访问。

Linux 支持 NFS 文件系统,把宿主机端配置成 NFS 服务器,目标机配置成 NFS 客户端。通过 NFS 将宿主机端文件目录共享给目标机,在目标机 Linux 上就可以访问该目录下的文件。这样,在宿主机上进行应用程序的编写和编译,然后在目标机的 Linux 上通过 NFS 挂载就可以直接运行调试程序,可避免反复下载的麻烦,大大提高了应用程序的开发速度。

3. FTP 服务

虽然 SMB 和 NFS 都可以用来传送文件,但 FTP 凭借简单高效的特性,成为跨

项目 1 构建嵌入式 Linux 开发环境

平台直接传送文件的主要方式。

FTP 服务采用客户机/服务器模式,开发者利用客户机程序连接到 FTP 服务器程序,然后向服务器程序发送命令,而服务器程序执行用户发出的命令,并将执行结果返回给客户机。在此过程中,FTP 服务器与 FTP 客户机之间建立两个连接:控制连接和数据连接。控制连接用于传送 FTP 命令以及响应结果,而数据连接负责传送文件。

在安装 Linux 操作系统的宿主机和目标机的开发环境中,利用 FTP 服务可以从宿主机登录到目标机,借助 FTP 子命令可以方便地在目标机上查看文件目录,实现文件的上传和下载等操作。

1.2 项目需求

要开发嵌入式 Linux 应用系统,首先需要构建嵌入式 Linux 开发环境。这一环境应包含以下内容:

- ① 设计或选购一款满足应用需求的嵌入式系统目标机(开发板),其中 CPU、内存、Flash、显示器、键盘、接口等满足应用需求并有一定冗余。
- ② 选择一台 PC 作为宿主机,其配置能满足嵌入式 Linux 开发环境的需要。
- ③ 提供一个可编辑应用程序的编辑器,能够满足应用程序的编写与编辑。
- ④ 提供一组交叉编译工具,能将编写的应用系统源程序交叉编译成 ARM 嵌入式系统可执行的机器代码。
- ⑤ 提供一组调试工具,能方便地调试嵌入式应用程序。
- ⑥ 提供一组下载工具,能将宿主机中编译好的嵌入式系统程序通过串口、网口、USB 口或其他方式,从宿主机下载到目标机的 Flash 中。

1.3 项目设计

要实现项目需求目标,完成项目任务,需要重点解决开发板的选择和开发方案的确定两大主要问题。

1.3.1 开发板的选定

开发嵌入式 Linux 系统首先需要一块嵌入式系统开发板。开发板的产生最终可以自己设计并生产出来。开发板的基本生产流程是:首先根据应用需求设计出开发板的原理图,然后在诸如 Protel 印制板 CAD 系统中输入原理图,排版编制 PCB 印制板图和元件分布图,再交由印制板厂加工完成。初始学习或产品研发阶段,可以通过选购一块能满足目标需求并有一定冗余的开发板。本书将选用 Tiny6410 ARM11 开发板进行开发,读者也可以选用其他型号的开发板,如上海双实公司的 PA2440A、