

嵌入式技术与应用丛书

EMBEDDED
SYSTEM

嵌入式Linux 开发技术与应用

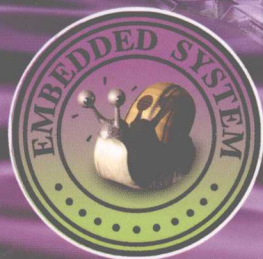
李超 肖建 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>



嵌入式技术与应用丛书

嵌入式 Linux 开发技术与应用

李超 肖建 编著

电子工业出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书以 ARM 微处理器为例,详细介绍了嵌入式系统基本原理和相关设计技术。书中内容均为作者多年从事嵌入式教学和科研经验之积累,内容翔实,阐述清晰,使读者能够深刻掌握嵌入式系统的基本原理和 Linux 应用程序的设计与开发。

全书共分 12 章,内容包括:第 1 章详细介绍了嵌入式系统的基本概念,给读者构建了一个嵌入式系统的轮廓;第 2 章重点介绍了 ARM 相关知识,包括 ARM 微处理器、ARM 编程模型、ARM 指令系统和 ARM 程序设计;第 3~5 章以 S3C2440 嵌入式微处理器为例,介绍了其体系结构、接口电路设计以及相关底层函数编写等内容;第 6~11 章介绍了嵌入式 Linux 相关概念,通过这些内容学习,读者可以了解到嵌入式 Linux 系统下程序开发基础知识、嵌入式 Linux 体系结构、内存管理、文件系统、进程管理和驱动程序设计等知识;第 12 章以嵌入式移动打印终端项目为例,详细介绍了嵌入式系统硬件设计及其软件开发,使读者对前面所学知识有更系统的掌握。

本书有完善的实验设备和教学课件与之配套,可作为高等院校通信类、电子类、信息类和理工类以及其他学科本科生教材使用,也可供研究生和嵌入式开发人员使用。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

嵌入式 Linux 开发技术与应用/李超,肖建编著. —北京:电子工业出版社,2008.5

(嵌入式技术与应用丛书)

ISBN 978-7-121-06374-9

I. 嵌… II. ①李…②肖… III. Linux 操作系统—程序设计 IV. TP316.89

中国版本图书馆 CIP 数据核字(2008)第 049886 号

责任编辑:高买花 特约编辑:陈宁辉

印 刷: 北京京科印刷有限公司

装 订:

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: 787×1 092 1/16 印张: 26.5 字数: 678 千字

印 次: 2008 年 5 月第 1 次印刷

定 价: 56.00 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。

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

服务热线:(010) 88258888。

出版说明

嵌入式技术是 21 世纪最具生命力的新技术之一，经过近几年的快速发展，已经成为电子信息产业中最具增长力的一个分支。随着手机、掌上电脑、GPS、机顶盒等新兴产品的大量应用，嵌入式系统的设计正成为软硬件工程师越来越关心的话题。面对不断涌现的技术需求和发展机遇，各大嵌入式系统开发商、各科研院所的研发人员都急需一套全方位、针对性强，且具有实际指导意义的嵌入式技术类书籍；各高等院校相关专业的本科生、研究生也迫切希望了解、掌握嵌入式系统的开发技巧，以推动嵌入式技术在各领域的广泛应用和快速发展。

《嵌入式技术与应用丛书》正是针对当前技术与市场需求，由国内站在 IT 业前沿并有实践开发经验的嵌入式系统专家，以实用技术为主线，理论联系实际，将他们在理论研究与实践工作中积累的大量经验和体会有机地融于一体，以丛书的形式奉献给广大读者！

本丛书由基础理论类、硬件设计类、软件开发类、综合应用类书籍组成，立足当前嵌入式技术的发展趋势、核心技术及其主要应用领域，将技术热点与实践应用紧密结合，以实际应用为主线，融合关键性嵌入式设计技术，围绕嵌入式设计理论、开发流程、嵌入式软件验证及测试、代码可重构以及代码优化等方面进行深入浅出的讲解和论述。

读者群定位于高等院校相关领域的高年级学生，科研、开发人员，嵌入式相关领域设计人员等，本丛书可作为嵌入式领域学习、开发人员的参考资料，也可作为高等院校相关专业师生的教学参考书。

本丛书的出版得到了业界许多专家、学者的鼎力相助，对此表示衷心的感谢！同时，热切欢迎广大读者提出宝贵意见，或者推荐更多优秀选题（gmholife@hotmail.com），共同为嵌入式技术的发展添砖加瓦！

电子工业出版社通信分社

2007 年 6 月

前 言

嵌入式系统的概念早在 20 世纪 70 年代就已经出现,最近十几年随着半导体技术和信息技术的发展,嵌入式成为当前最热门的技术之一。现在,嵌入式技术已经渗透到科学研究、军事、航空航天、信息家电等各个领域,极大地改变着人们的生活。随着国内外各种各样嵌入式产品的设计和流行,嵌入式技术的重要性日益显现出来,这方面的人才需求也日益增多,但是嵌入式人才的培养在高校教学中却还是空白,因此高等院校如何根据社会需求培养出合格的嵌入式人才是一个非常紧迫的任务。在此背景下,作者于 2004 年编写了“嵌入式系统原理与设计”讲义,在南京邮电大学计算机学院试用,并在多次社会培训中使用。本书在原讲义的基础之上,结合广大师生的反馈意见修改而成,可以作为高等院校通信类、电子类、信息类和理工类以及其他学科本科生教材使用,也可供研究生和嵌入式开发人员使用。

嵌入式系统是软、硬件相结合的应用系统,涉及多门科学知识。在这些知识体系结构中,有两个核心:嵌入式微处理器和嵌入式操作系统,要想培养出合格的嵌入式人才,必须紧紧围绕这两个核心因材施教。本书以 S3C2440 微处理器为例,结合嵌入式 Linux 操作系统,详细讨论了其硬件结构和软件架构,从嵌入式理论和应用的角度阐述嵌入式系统的基本原理和实际应用开发。

本书内容丰富,包含 5 大部分内容:

1. ARM 体系架构及编程

首先介绍了 ARM 微处理器发展历史,然后重点讲解 ARM 编程模型、ARM 指令系统以及 ARM 汇编程序设计。读者在学习这部分内容时,应该把重点放在汇编和 C 的混合编程以及 ARM 体系架构上,而不必一条一条来学习汇编指令。从目前嵌入式应用来说,很少使用汇编语言来实现某项功能。

2. 嵌入式 S3C2440 微处理器及其接口技术

该部分以 S3C2440 微处理器为例,详细介绍了 ARM 微处理器的内部组织,并详细讲解看门狗、RTC、串口和中断等多个接口的硬件设计及软件开发。

在讲述完接口部件的基础上,重点讲解了嵌入式 BootLoader——vivi 的作用和使用说明,此外还对 BootLoader 的体系架构和源码组织进行了较为深入的解析,并最终对 vivi 的启动过程进行了详细分析。

3. 嵌入式 Linux 编程基础和应用程序开发

基于 Linux 开发应用程序必须熟练掌握嵌入式交叉编译套件,本书详细介绍了嵌入式 gcc 套件中每个工具的详细使用。在介绍这部分内容时,着重讲解了库和远程调试,特别是远程调试知识的掌握,对于开发应用程序至关重要。

在讲述应用程序设计时，重点介绍了进程、线程的创建，以及它们之间的同步互斥机制及通信机制。另外也重点讲解了嵌入式图形设计，在讲解这部分内容时，为了使读者更好地掌握这部分内容，这里采取了图解实例的方式来讲述。

4. 嵌入式系统驱动设计和编程

驱动是嵌入式设计中比较难的一块内容，在介绍这部分内容时，作者结合本科学生的实际水平，讲解了较为简单的一些字符设备驱动和网络设备驱动。

5. 实例解剖

在本书的最后章节中，以嵌入式移动打印终端平台为例，讲解了嵌入式开发的过程，包括项目整体设计、驱动编程、应用程序设计以及调试等过程。

嵌入式系统软中有硬，硬中含软，软硬结合。本书在描述软、硬件时，都会从软、硬两个角度去探讨，力求让读者能够深刻掌握嵌入式系统的精华。一名合格的嵌入式系统设计人员往往要求同时具备软件和硬件两方面的知识。另外，本书特别重视理论和实践相结合，力求既面向教学，又面向实际开发，因此本书既可作为高等院校通信类、电子类、信息类和理工类以及其他学科本科生教材使用，又可供广大嵌入式开发人员在科学研究和产品开发时使用。

本书涉及内容广泛，加之时间仓促，作者水平有限，书中难免出现错误和不妥之处，恳请广大读者和同行批评指正。如果在学习开发中发现问题，或有好的建议，欢迎来函。电子邮件：lichao@njupt.edu.cn。

编 者

2008年3月

目 录

第1章 引言	(1)
1.1 嵌入式系统概述	(1)
1.1.1 嵌入式系统历史	(1)
1.1.2 嵌入式系统定义	(1)
1.1.3 嵌入式系统的应用	(2)
1.1.4 嵌入式系统的特点	(3)
1.2 嵌入式处理器	(4)
1.2.1 嵌入式微控制器	(4)
1.2.2 嵌入式微处理器	(5)
1.2.3 嵌入式 DSP 处理器	(7)
1.2.4 嵌入式片上系统 (System On Chip)	(7)
1.3 嵌入式操作系统	(8)
1.3.1 VxWorks 嵌入式操作系统	(8)
1.3.2 WinCE	(9)
1.3.3 RT-Linux	(11)
1.3.4 uCLinux	(11)
1.3.5 Palm	(11)
1.4 嵌入式系统设计	(12)
1.4.1 嵌入式系统开发流程	(12)
1.4.2 嵌入式系统开发模式	(13)
第2章 ARM 体系架构及汇编程序设计	(14)
2.1 ARM 微处理器介绍	(14)
2.1.1 ARM 微处理器的历史发展	(14)
2.1.2 ARM 微处理器的特点和应用	(15)
2.1.3 ARM 微处理器家族	(16)
2.2 ARM 编程模型	(17)
2.2.1 ARM 微处理器模式	(17)
2.2.2 ARM 寄存器	(18)
2.2.3 异常处理	(19)
2.2.4 ARM 的存储器组织	(23)
2.3 ARM 指令系统	(25)
2.3.1 ARM 指令格式	(25)
2.3.2 ARM 指令的寻址方式	(26)
2.3.3 ARM 最常用指令和条件后缀	(28)

2.4	ARM 汇编语言程序设计	(30)
2.4.1	ARM 汇编语言中的程序结构	(30)
2.4.2	ARM 汇编器所支持的伪指令	(31)
2.4.3	ARM 汇编语言与 C/C++ 的混合编程	(35)
2.5	汇编程序综合实验	(40)
2.5.1	流程图设计	(40)
2.5.2	详细代码设计	(40)
第 3 章	嵌入式 S3C2440 处理器	(43)
3.1	S3C2440 芯片概述	(43)
3.1.1	特性	(43)
3.1.2	引脚信息	(44)
3.2	S3C2440 微处理器功能模块	(45)
3.2.1	处理器核体系架构	(45)
3.2.2	S3C2440 存储系统	(46)
3.2.3	S3C2440 启动加载模式	(48)
3.2.4	LCD 屏/触摸屏支持模块	(49)
3.2.5	时钟和电源管理	(51)
第 4 章	接口技术	(54)
4.1	看门狗接口设计	(54)
4.1.1	S3C2440 处理器看门狗	(54)
4.1.2	看门狗使用实例	(56)
4.2	S3C2440 微处理器 GPIO 接口设计	(58)
4.2.1	GPIO 口概述	(58)
4.2.2	端口控制	(59)
4.2.3	基于 GPIO 口的 LED 灯设计	(60)
4.3	UART 设计	(62)
4.3.1	S3C2440 串口概述	(63)
4.3.2	UART 操作	(64)
4.3.3	UART 寄存器	(65)
4.3.4	串行接口电路设计及编程	(70)
4.4	S3C2440 中断管理及设计	(73)
4.4.1	中断控制器	(73)
4.4.2	中断源	(73)
4.4.3	中断优先级模块	(75)
4.4.4	中断控制器中端口寄存器	(77)
4.4.5	中断程序架构设计	(82)
第 5 章	嵌入式 BootLoader 设计	(91)
5.1	嵌入式 BootLoader 简介	(91)

5.1.1	嵌入式系统软件体系架构	(91)
5.1.2	各式各样 BootLoader	(92)
5.2	典型嵌入式 BootLoader——vivi	(93)
5.2.1	vivi 简介	(93)
5.2.2	vivi 的配置与编译	(93)
5.2.3	vivi 的使用	(95)
5.2.4	vivi 代码体系架构分析	(96)
5.2.5	vivi 系统配置浅析	(105)
5.2.6	vivi 代码启动分析	(111)
第 6 章	嵌入式 Linux 编程基础	(132)
6.1	嵌入式集成交叉编译环境	(132)
6.1.1	构建嵌入式交叉编译环境	(132)
6.1.2	嵌入式交叉编译环境构成	(133)
6.1.3	嵌入式常用交叉编译工具的使用	(136)
6.2	Makefile 原理与使用	(148)
6.2.1	make 工具和 Makefile 文件	(148)
6.2.2	Makefile 基本结构	(150)
6.2.3	Makefile 变量	(153)
6.2.4	Makefile 高级规则的使用	(155)
6.2.5	Makefile 中函数的使用	(157)
6.2.6	make 工具的使用	(158)
6.2.7	Makefile 举例	(159)
6.2.8	使用 automake 生成 Makefile	(163)
6.3	应用程序调试	(170)
6.3.1	ARM 调试方法简介	(170)
6.3.2	GDB 本地调试	(172)
6.3.3	DDD 图形调试工具	(172)
6.3.4	嵌入式开发板远程调试	(175)
6.4	嵌入式 Linux 库的使用	(180)
6.4.1	库的概述	(180)
6.4.2	库的命名与分类	(181)
6.4.3	创建静态库	(182)
6.4.4	创建共享库	(183)
6.4.5	静态库 VS 共享库实验	(186)
第 7 章	嵌入式 Linux 操作系统	(187)
7.1	嵌入式 Linux 发展	(187)
7.2	嵌入式 Linux 内核结构	(188)
7.3	进程暨进程调度	(189)

7.3.1	进程概述	(189)
7.3.2	Linux 下对进程的描述	(190)
7.3.3	进程调度	(194)
7.4	内存管理	(194)
7.4.1	内存管理的本质特征	(194)
7.4.2	地址映射模型	(197)
7.5	中断管理系统	(198)
7.5.1	构建 ARM-Linux 中断系统骨架	(198)
7.5.2	ARM-Linux 系统下中断响应	(207)
7.6	系统调用	(208)
7.6.1	系统调用原理	(208)
7.6.2	嵌入式系统调用实例	(208)
7.7	文件系统	(211)
7.8	进程间通信机制	(212)
7.8.1	管道及有名管道	(213)
7.8.2	信号	(214)
7.8.3	消息队列	(215)
7.8.4	共享内存	(215)
7.8.5	信号量	(216)
7.9	嵌入式 Linux 内核源码组织	(217)
7.9.1	嵌入式 Linux 内核目录结构	(217)
7.9.2	嵌入式 Linux 内核配置系统	(218)
第 8 章	构建嵌入式 Linux 系统	(220)
8.1	嵌入式 Linux 内核裁减	(220)
8.2	构建根文件系统	(221)
8.2.1	根文件系统架构	(221)
8.2.2	定制根文件系统	(223)
8.3	文件系统实验	(234)
第 9 章	嵌入式 Linux 应用编程	(241)
9.1	进程控制及编程	(241)
9.1.1	进程控制函数	(241)
9.1.2	守护进程设计	(254)
9.2	进程间通信	(262)
9.2.1	管道通信	(262)
9.2.2	命名管道	(268)
9.2.3	消息队列	(272)
9.2.4	共享内存	(279)

9.2.5 信号通信	(284)
9.3 线程控制及编程	(291)
9.3.1 线程概述	(291)
9.3.2 线程编程函数	(293)
9.3.3 线程同步与互斥	(296)
第 10 章 嵌入式 Linux 设备驱动	(306)
10.1 设备驱动基本概念	(306)
10.1.1 设备驱动在 Linux 系统中的地位	(306)
10.1.2 设备驱动完成的工作	(307)
10.1.3 Linux 对设备驱动的管理机制	(307)
10.2 内核模块及模块编程	(309)
10.2.1 模块简介	(309)
10.2.2 模块编程设计	(309)
10.3 Linux 下简单设备驱动程序设计	(314)
10.3.1 Linux 设备驱动层次结构	(314)
10.3.2 典型字符设备驱动编写框架	(316)
10.3.3 高级字符设备驱动程序设计	(324)
10.3.4 设备文件系统	(333)
10.3.5 misc 设备设计	(335)
10.4 网络设备及驱动程序	(337)
10.4.1 基本概念	(337)
10.4.2 网络数据包处理流程	(340)
10.4.3 以太网控制器 CS8900A	(343)
10.4.4 CS8900A 驱动程序设计	(344)
第 11 章 嵌入式 Linux 图形设计	(359)
11.1 Qt/Embedded 图形开发基础	(359)
11.1.1 Qt 简介	(359)
11.1.2 Qt/Embedded 组件	(360)
11.2 Qt/Embedded 关键技术	(361)
11.2.1 Qt/Embedded 对象模型	(361)
11.2.2 信号与插槽机制	(362)
11.2.3 信号/槽编程实例	(363)
11.2.4 Qt/Embedded 关键类	(366)
11.3 Qt/Embedded 实例设计	(368)
第 12 章 嵌入式移动打印终端开发实例剖析	(378)
12.1 项目概述	(378)
12.2 嵌入式系统移动打印终端总体架构构建	(378)
12.2.1 总体架构	(378)

12.2.2 软/硬件选型	(380)
12.3 软/硬件设计	(384)
12.3.1 设计流程	(384)
12.3.2 硬件设计	(384)
12.3.3 打印机驱动的设计	(388)
12.3.4 扫描仪驱动设计	(397)
12.3.5 网络通信模块及服务器端的开发	(401)
12.3.6 系统应用程序设计	(404)
12.4 系统调试	(406)
参考文献	(410)

第1章 引言

随着社会信息化的发展,嵌入式产品已经走进人们的日常生活中,比如,小到MP3、PDA、手机等微型数字产品,大到智能家电、车载设备、航空航天等领域,都可以看到嵌入式设备的影子。那么,什么是嵌入式?它有哪些特点?本章将简单介绍嵌入式系统的一些基本概念、应用领域及特点,引导读者进入嵌入式技术的殿堂。

1.1 嵌入式系统概述

1.1.1 嵌入式系统历史

随着信息技术的飞速发展,嵌入式技术已经成为微电子技术和计算机技术一个重要的分支,同时也使得计算机的分类从以前的巨型机、大型机、小型机以及微机之分变为通用计算机和嵌入式系统之分。

当今时代,制造技术的集成化和小型化发展趋势越来越明显,能否设计和生产出个性化需求的电子产品也将决定一个企业能否生存和继续发展。所有这些,都决定了嵌入式技术必将成为后 PC 时代的主宰者。

嵌入式系统并不是一个新的概念,其实早在 1970 年左右就有嵌入式系统应用的例子。当时的嵌入式系统并没有专门的操作系统,它们只是为了实现某个控制功能,使用简单的循环控制语句来完成。当应用系统设计越来越复杂,功能越来越强大时,传统的设计方法已经无法胜任,这时随着通用操作系统的迅速发展,人们将目光汇聚到设计专用的嵌入式操作系统来解决上述问题。因此,从 20 世纪 80 年代开始,出现了各种各样专用的嵌入式操作系统,比较著名的有 VxWorks、pSOS、WinCE、Linux 等。

嵌入式系统最大的特点是其具有专用性,也就是说,每一套嵌入式系统都有其特殊的应用场合与特定功能,它们都是为了特定的目的而设计的,而且往往还受环境、成本、体积、功耗等条件的制约,需要最大限度地在硬件和软件上“量身定做”,以提高资源的利用率,这也是嵌入式系统与通用计算机最大的一个区别。

1.1.2 嵌入式系统定义

1. IEEE 定义

嵌入式系统并没有一个标准的定义,IEEE 是这样定义嵌入式系统的:嵌入式系统是“控制、监视或者辅助设备、机器和车间运行的装置”(原文为 devices used to control, monitor, or assist the operation of equipment, machinery or plants)。从这个定义,我们可以看

出嵌入式系统功能比较广泛。

2. 国内定义

不过, IEEE 的定义并不能够反映出嵌入式系统的本质特征。目前, 国内普遍采用下列描述来定义嵌入式系统: 嵌入式系统以应用为中心, 以计算机技术为基础, 软、硬件可裁减, 适应应用系统对功能、可靠性、成本、体积、功耗严格要求的专用计算机系统。

可以从以下几个方面来理解上述嵌入式系统的定义:

- 嵌入式系统以应用为中心, 也就是说, 嵌入式系统必须与具体应用相结合才会有生命力, 这也是嵌入式技术诞生和发展的不竭动力的源泉。目前, 嵌入式应用的场合极其广泛, 金融、航天、通信、家电、医疗、工业控制以及军事等各个领域都可以看到嵌入式技术应用的例子。
- 嵌入式系统是一个技术高度密集型的综合体, 它包含了计算机技术、电子技术、半导体技术、通信技术、多媒体技术等, 可以说, 嵌入式系统是各种科学技术的集成系统。
- 嵌入式系统包括软件和硬件两个组成部分, 可以涵盖机械和其他的附属装置。其中, 硬件部分涉及芯片设计、存储设计以及各个接口部分的设计; 软件部分包括专用嵌入式操作系统、硬件驱动、通信协议、图形界面等各个方面。
- 嵌入式系统软、硬件必须可裁减。作为一个嵌入式产品, 必须考虑到可靠性、成本、体积和功耗等各个方面因素。另外, 很多用于工控的嵌入式产品, 还要求具有实时性的特点。所以, 在进行嵌入式设计时必须对多余的软件和硬件进行裁减, 使其最小化, 从而可以降低功耗、成本、体积, 提高整个系统的可靠性。

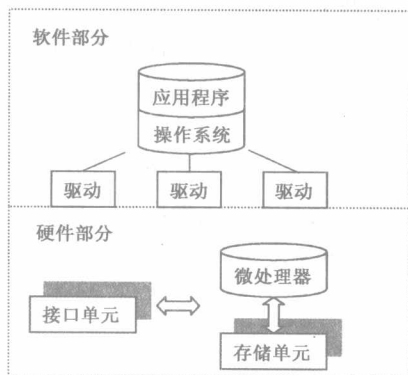


图 1-1 典型嵌入式系统架构

通常情况下, 嵌入式系统的架构包含微处理器、存储单元、接口单元和软件单元几个部分, 其中软件单元至少包含嵌入式操作系统和应用程序。当然嵌入式操作系统和底层硬件单元之间必须要有相应驱动程序作为接口。

典型嵌入式系统架构如图 1-1 所示。从图中可以看出, 嵌入式系统和普通微机在体系结构上并没有本质的区别, 两者非常类似, 因此普通微机上开发程序的那些知识照样可以用在嵌入式系统开发上。当然嵌入式系统的微处理器和操作系统具有专用性, 开发嵌入式系统应用软件时, 必须首先掌握这两个核心内容。

1.1.3 嵌入式系统的应用

嵌入式技术具有非常广泛的应用场合, 无论在航天、军事、信息家电等各个领域, 人们都会无时无刻地接触到嵌入式产品。

1. 航天领域

嵌入式系统在航空航天领域得到了非常广泛的应用,比如在美国宇航局的“极地登陆者”号、“深空二号”和火星气候轨道器等登陆火星探测器上,就采用了 VxWorks,负责火星探测器全部飞行控制,包括飞行纠正、载体自旋和降落时的高度控制等,而且还负责数据收集以及与地球的通信工作。

2. 军事领域

进入 20 世纪 90 年代以来,随着高新技术的迅猛发展及其在军事领域的广泛应用,武器装备在军事斗争和军队建设中的作用日益突出。各国在新军事革命的大潮中发展武器装备的一个明显共性是,利用现有信息技术的发展成果对现有的武器装备进行改造,最终实现武器装备的智能化,作战系统的网络化。例如,美国战略核武器民兵 (MinuteMan) 系列导弹的导航系统就采用了较为先进的嵌入式技术。

3. 信息家电及智能仪器

(1) 手机、PDA 类产品:图 1-2 给出的是采用 eCos 操作系统,基于 MiniGUI 开发的高端智能手机的应用程序界面。

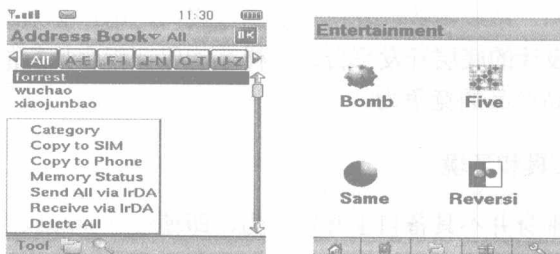


图 1-2 基于 MiniGUI 开发的智能手持设备

(2) 仪表及控制系统:图 1-3 给出的是基于 Linux 和 MiniGUI 操作系统开发的数控系统、工业仪表及医疗仪器的界面。

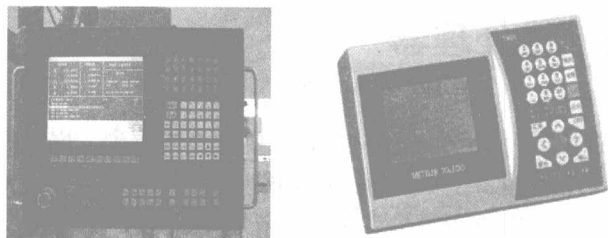


图 1-3 工业仪表及控制系统界面

1.1.4 嵌入式系统的特点

从嵌入式系统的定义可以看出,嵌入式系统具有下面几个非常明显的特征:

1. 微内核

由于嵌入式系统多用于手持设备和小型电子装置,系统资源比较少,所以内核较通用操作系统要小得多。比如经过裁减的嵌入式专用 Linux 内核一般只有 200KB 左右,而通用的 PC 上的 Linux 内核可能要达到几 MB。

2. 专用性强

嵌入式产品都是以特定应用为背景来进行设计的,不同的任务实现,往往设计人员会对原系统的软、硬件进行较大的修改,这和通用软件的“升级”是完全不同的概念。

3. 可裁减

嵌入式系统一般只是实现一个特定的功能,同时它对最后产品的功耗、体积、成本都有着严格的要求,这就需要能够进行软、硬件的裁减,把多余的软件和硬件部分裁减达到设计的要求。

4. 专用操作系统的支撑

嵌入式系统的应用程序可以没有操作系统而直接在芯片上运行;但是为了合理地调度多任务,利用系统资源、系统函数以及已有的函数接口,用户必须选择某一专用嵌入式操作系统作为应用软件设计的底层开发平台,这样才能保证程序执行的实时性、可靠性,缩短项目开发周期,提高产品的竞争力。

5. 专用的开发工具和环境

由于嵌入式系统本身并不具备自主开发能力,即使设计完成以后,用户通常也不能对其中的程序功能直接进行修改,因此必须有一套开发工具和环境才能进行开发,这些工具和环境一般是基于通用计算机上的软、硬件设备以及各种逻辑分析仪、信号示波器等。开发时由主机和目标机两个部分构成,主机用于程序的开发,目标机作为目标代码的执行载体,在进行开发时,往往需要在主机和目标机之间交替结合进行。

1.2 嵌入式处理器

嵌入式系统由软件和硬件两个部分构成,从硬件角度来说,嵌入式处理器是嵌入式硬件中最核心的部分。目前,世界上具有嵌入式功能特点的处理器有 1000 多种,分为嵌入式微控制器(MCU)、嵌入式 DSP 处理器、嵌入式微处理器(MPU)、嵌入式片上系统(SOC)几类。

1.2.1 嵌入式微控制器

嵌入式微控制器实际上指的是单片机,顾名思义,就是将整个计算机系统集成到一块芯片中。嵌入式微控制器一般以某一种微处理器内核为核心,芯片内部集成 ROM/EPROM、

RAM、总线、总线逻辑、定时/计数器、WatchDog、I/O、串行口、脉宽调制输出、A/D、D/A、Flash RAM、EEPROM 等各种必要功能和外设。为适应不同的应用需求，一般一个系列的单片机具有多种衍生产品，每种衍生产品的处理器内核都是一样的，不同的是存储器和外设的配置及封装。这样可以使单片机最大限度地和应用需求相匹配，功能不多不少，从而减少功耗和成本。

和嵌入式微处理器相比，微控制器的最大特点是单片化，体积大大减小，从而使功耗和成本下降，可靠性提高。微控制器是目前嵌入式系统工业的主流之一。微控制器的片上外设资源一般比较丰富，适合于控制，因此被称作微控制器。

嵌入式微控制器目前的品种和数量最多，比较有代表性的通用系列包括 8051、P51XA、MCS-251、MCS-96/196/296、C166/167、MC68HC05/11/12/16、68300 等。另外还有许多半通用系列，如：支持 USB 接口的 MCU 8XC930/931、C540、C541；支持 I²C、CAN-Bus、LCD 及众多专用 MCU 和兼容系列。目前 MCU 占嵌入式系统约 70% 的市场份额。

特别值得注意的是，近年来提供 x86 微处理器的著名厂商 AMD 公司，将 Am186CC/CH/CU 等嵌入式处理器称为 Microcontroller，Motorola 公司把以 PowerPC 为基础的 PPC505 和 PPC555 也列入单片机行列，TI 公司亦将其 TMS320C2XXX 系列 DSP 作为 MCU 进行推广。

1.2.2 嵌入式微处理器

嵌入式微处理器由通用计算机 CPU 演变而来，通常情况下，嵌入式微处理器都是 32 位以上的处理器，具有体积小、重量轻、成本低、可靠性高等优点。

目前主要的嵌入式微处理器类型有如下几种：

1. ARM/StrongARM

ARM (Advanced RISC Machine) 公司是全球领先的 16/32 位 RISC 微处理器芯片设计商。ARM 公司并没有自己的芯片生产线，它专注于设计高性能、低成本、低功耗的嵌入式微处理器核，同时 ARM 公司在全球有很多合作伙伴，它们通过购买 ARM 公司的芯片设计技术、专利等方式，再配上自己的外围电路，生产出各具特色的嵌入式微处理器芯片。

在嵌入式微处理器市场中，ARM 微处理器芯片占据绝对的统治地位，目前全球有 75% 的市场份额被 ARM 微处理器所垄断。ARM 微处理器有三个主要特点：小体积、低功耗、低成本和高性能；16/32 位双指令集；全球合作伙伴众多。

2. MIPS

MIPS 是最早的、最成功的 RISC (Reduced Instruction Set Computer) 处理器之一，起源于 Stanford Univ 的电动机系。其创始人 John L. Hennessy 于 1984 年在硅谷创立了 MIPS INC. 公司。

MIPS 是高效精简指令集计算机 (RISC) 体系结构中最优雅的一种，这可以从 MIPS 对于后来研制的新型体系结构，比如 DEC 的 Alpha 和 HP 的 Precision 产生的强烈影响看出来。虽然自身的优雅设计并不能保证在充满竞争的市场上长盛不衰，但是 MIPS 微处理器