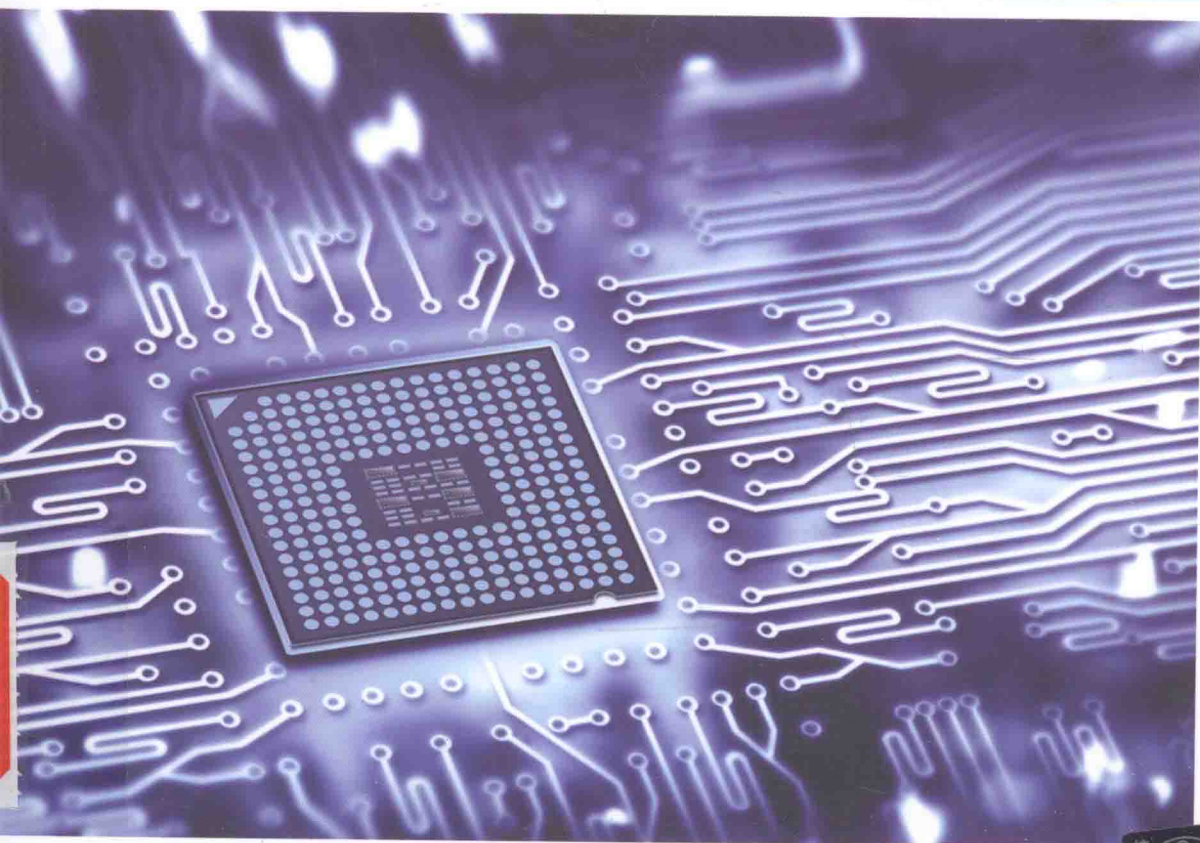


高等院校电子信息科学与工程规划教材

嵌入式系统原理、 应用与实践教程

孟利民 宋秀兰 徐志江 彭 宏 应颂翔 ● 编著



清华大学出版社



内 容 简 介

本书以目前流行的基于 ARM 架构的嵌入式微处理器及 Linux 操作系统为核心,结合作者多年的教学与科研工作经验,循序渐进地介绍了嵌入式系统原理、嵌入式系统开发环境的搭建、嵌入式系统开发和调试工具以及相关的实验案例,最后介绍了基于嵌入式系统的视频通信原理、方法及实验。

本书共分为 9 章,第 1~2 章介绍了嵌入式系统的基本概念及嵌入式处理器架构;第 3~5 章介绍了嵌入式系统实验平台及开发环境的搭建;第 6~9 章是本书的核心内容,介绍了如何利用 Linux 系统在 ARM 架构的嵌入式微处理器上进行相应的实验开发过程,包括 Linux 系统使用基础、调试工具、Linux 系统开发环境、内核移植、驱动程序开发等,并配有 Linux 虚拟机实验、挂载实验、交叉编译实验、Linux 内核移植实验、硬盘存储实验、GPIO 驱动实验、I2C 驱动实验、SD 卡驱动实验、RTC 时钟驱动实验等,最后通过多任务之间的同步和通信以及 TCP/IP 网络编程方法,介绍了网络视频通信实验案例,包括图像合成实验、视频采集及 RTP 发送实验、远程控制视频实验及多 Client 单 Server 实验。

本书适合作为通信工程和电子信息类相关专业本科生和研究生教材,同时也可供工程技术人员参考。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

嵌入式系统原理、应用与实践教程/孟利民等编著. —北京:清华大学出版社,2016

高等院校电子信息科学与工程规划教材

ISBN 978-7-302-41016-4

I. ①嵌… II. ①孟… III. ①微型计算机-系统开发-高等学校-教材 IV. ①TP360.21

中国版本图书馆 CIP 数据核字(2015)第 169510 号

责任编辑:贾小红

封面设计:刘超

版式设计:刘艳庆

责任校对:赵丽杰

责任印制:沈露

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772045, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62788951-223

印 刷 者:北京富博印刷有限公司

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:21.75 字 数:543 千字

版 次:2016 年 2 月第 1 版 印 次:2016 年 2 月第 1 次印刷

印 数:1~3000

定 价:46.00 元

产品编号:063551-01

前 言

随着网络化、高清化、智能化的各类高性能设备芯片的不断涌现,开发者需要根据不同的设计方案,选择合适的开发技术实现嵌入式开发。硬件软件化和软件硬件化是未来智能化设备开发的重点发展方向,ARM 微处理器在智能化、定制化和网络化设备的普及以及 Linux 操作系统的日益完善,为智能设备的硬件软件化和软件硬件化提供了非常好的实现路径。本书结合作者在浙江工业大学通信工程专业嵌入式系统课程教学中的体会,总结将嵌入式技术应用于 IPCAM、智能交通、智慧社区、物联网等方面的科研和经验呈现给读者。

本书起源来自于 2008 年承担的无线网络摄像机科研项目,作者系统地研究了无线多媒体终端设备和相应的嵌入式软件,发现嵌入式设备的巨大潜力,因为它将硬件软件化和软件硬件化的设计理念真正融入到了智能设备的设计中。之后作者就着手制作了 PN-ARM9 嵌入式系统教学实验箱,配合实验讲义,在浙江工业大学通信工程专业的本科教学中加以实施。嵌入式系统教学是实践性很强的一门课程,学生除了学习一整套完整的理论体系和一系列的操作函数外,还必须针对某种应用进行实验,没有实验环节的嵌入式系统教学很难让学生融会贯通。为此,作者从基本知识、基础技能、综合应用等方面配以相应的实验案例,加强学生感性认识,培养学生的综合设计和应用技能,多年来收到的反馈效果较好。结合近几年作者完成的网络视频通信方面的科研课题,本书介绍了网络视频通信实验案例,包括图像合成实验、视频采集及 RTP 发送实验、远程控制视频实验及多 Client 单 Server 实验等。本书所附的大量实验案例可以帮助读者更好地理解嵌入式系统基本原理和应用开发技术。

随着信息与通信技术的飞速发展,嵌入式系统技术在网络通信、电子消费、移动互联、工业控制等领域得到了广泛应用,同时它也是智能设备设计领域最为热门的技术之一。学习和应用嵌入式技术已经成为通信、电子、计算机和自动化领域工程师感兴趣的话题,衷心地希望本书能对高等院校相关专业的教师和学生、从事嵌入式系统设计和开发的研究人员及企业工程师有所帮助。

本书由孟利民、宋秀兰、徐志江、彭宏和应颂翔共同完成,孟利民提供了多年来嵌入式教学实验讲义及相关科研资料,宋秀兰精心编写了本书的大部分章节,徐志江、彭宏和应颂翔编写了本书部分章节并核对了书中的实验教程。感谢清华大学出版社贾小红编辑的大力支持,使本书得以与读者见面。感谢浙江工业大学通信与信息系统研究所老师的支持和帮助,感谢浙江省通信网技术应用研究重点实验室的研究生们对实验案例做了测试和修改,同时感谢浙江工业大学教务处为本书的出版提供帮助,获得“浙江工业大学重点教材

建设项目”支持。

对本书配套的嵌入式系统教学平台感兴趣的读者，可以联系浙江工业大学孟利民老师，电话：0571-85290593，E-mail：105669876@qq.com。

由于知识所限，书中不足之处在所难免，恳请各位专家和读者指正。

编 者

目 录

第 1 章 嵌入式系统概述	1
1.1 嵌入式系统简介	1
1.2 嵌入式系统的特点	1
1.3 嵌入式系统的组成	3
1.4 嵌入式系统的分类	4
1.5 嵌入式系统的发展	6
第 2 章 嵌入式微处理器概述	7
2.1 处理器原理	7
2.1.1 CPU 的指令系统	7
2.1.2 CPU 的分类	8
2.1.3 CPU 的结构	9
2.1.4 大端和小端	10
2.2 嵌入式处理器的分类	10
2.3 嵌入式处理器的特点	12
2.4 熟悉 ARM 处理器	12
2.4.1 ARM 技术的发展	12
2.4.2 ARM 处理器核	13
2.4.3 ARM 处理器的片内总线	14
2.5 处理器模式	16
2.6 寄存器组织	17
2.7 ARM 的选型原则	18
第 3 章 熟悉 ARM 嵌入式实验系统	21
3.1 ARM 嵌入式教学实验系统	21
3.2 嵌入式实验系统的总体设计方案	22
3.3 系统开发板硬件架构	24
3.4 视频采集	25
3.5 存储	28
3.6 以太网传输	30
第 4 章 ARM 指令集及汇编	32
4.1 ARM 指令集概述	32
4.2 ARM 指令的寻址方式	35
4.3 ARM 存储器访问指令	37

4.4	ARM 数据处理指令	42
4.5	ARM 跳转指令	49
4.6	ARM 协处理指令	50
4.7	ARM 杂项指令	52
4.8	ARM 伪指令	54
第 5 章	ADS 1.2 集成开发环境和 ARM 仿真器介绍	57
5.1	ADS 集成开发环境	57
5.1.1	CodeWarrior 集成开发环境	58
5.1.2	ADS 调试器	60
5.2	ADS 使用入门	61
5.2.1	使用 ADS 1.2 软件	61
5.2.2	ADS 1.2 环境下工程的仿真、调试及配置方法	70
5.3	JTAG 介绍	73
5.4	Multi-ICE 仿真器	74
	实验 1: ADS 1.2 集成开发环境练习	75
	实验 2: 汇编指令实验 1	79
	实验 3: 汇编指令实验 2	81
	实验 4: 汇编指令实验 3	83
	实验 5: ARM 微控制器工作模式实验	87
	实验 6: C 语言程序实验	90
	实验 7: C 语言调用汇编程序实验	93
	实验 8: ADS 环境下的开发板汇编指令实验	96
	实验 9: C 语言程序设计与串口通信实验	108
	实验 10: I/O 接口控制实验	117
第 6 章	Linux 使用基础	126
6.1	Linux 简介	126
6.1.1	Linux 操作系统及其特点	126
6.1.2	Linux 系统的发展过程	127
6.1.3	Linux 系统的组成部分	127
6.1.4	常用 Linux 系统介绍	127
6.1.5	Red Hat Enterprise Linux 5 简介	127
6.1.6	VMware Workstation 简介	128
6.2	Linux 基本命令	128
6.2.1	常用命令	128
6.2.2	文件操作命令	131
6.2.3	文件编辑命令	138
6.2.4	系统关闭命令	141
6.2.5	用户管理相关命令	142
6.2.6	信息系统相关命令	144
6.2.7	网络相关命令	147
6.3	文本编辑工具 gedit	150

6.4	VI 编辑器.....	152
6.5	远程管理工具.....	153
6.5.1	Telnet 服务.....	154
6.5.2	Telnet 服务的安装.....	154
6.5.3	Telnet 服务的启动.....	155
6.5.4	Telnet 服务的配置.....	157
6.5.5	Telnet 客户端的使用.....	158
6.6	SSH 服务.....	159
6.6.1	OpenSSH 服务器的安装.....	160
6.6.2	OpenSSH 服务的配置.....	161
6.6.3	SSH 客户端的使用.....	162
6.7	Windows 下常用远程登录客户端.....	162
6.7.1	PuTTY.....	163
6.7.2	SSH Secure Shell Client.....	164
6.7.3	SecureCRT.....	165
6.8	NFS 服务器的配置及管理.....	167
6.8.1	NFS 概述.....	167
6.8.2	NFS 安装与配置.....	167
	实验 11: 建立 Linux 虚拟机及熟悉常用命令.....	171
	实验 12: mount 挂载实验.....	194
第 7 章	Linux 系统开发环境.....	198
7.1	Linux 交叉编译.....	198
7.2	GCC 和 GDB.....	199
7.2.1	基于 GNU 及 Linux 内核的编程风格.....	199
7.2.2	GCC 编译器.....	199
7.2.3	GDB 调试器.....	201
7.3	BootLoader.....	204
7.3.1	BootLoader 简介.....	204
7.3.2	常见 BootLoader 介绍.....	206
7.4	Linux 内核移植.....	207
7.4.1	Linux 内核结构.....	207
7.4.2	Linux 系统的可加载内核模块机制.....	208
7.4.3	用户空间和内核空间.....	208
	实验 13: Linux 交叉编译平台实验.....	209
	实验 14: ARMboot 下载运行实验.....	211
	实验 15: 程序下载烧写实验.....	215
	实验 16: Linux 内核编译实验.....	223
	实验 17: 硬盘存储实验.....	229
第 8 章	Linux 环境下驱动程序开发.....	237
8.1	Linux 设备驱动.....	237
8.1.1	设备驱动的分类.....	237

8.1.2 驱动程序中的基本要素	238
8.2 设备驱动开发	240
8.2.1 驱动开发的一般流程	240
8.2.2 驱动程序的框架	240
8.2.3 测试程序框架	242
8.3 Linux 驱动的部分技术简介	242
8.3.1 同步机制	242
8.3.2 阻塞与非阻塞	244
8.3.3 时间	244
8.4 Linux 驱动程序实例分析	246
8.5 Linux 系统下的具体实现	250
8.6 make 和 Makefile 文件	251
实验 18: Linux 下 GPIO 驱动程序编写实验	253
实验 19: I2C 驱动程序编写实验	267
实验 20: MMC/SD 驱动实验	269
实验 21: RTC 时钟驱动实验	274
第 9 章 Linux 环境下应用程序开发	285
9.1 多任务之间的同步和通信	285
9.1.1 System V 共享内存	286
9.1.2 消息队列	290
9.1.3 POSIX 信号量	294
9.1.4 System V 信号量	297
9.1.5 互斥锁和条件变量	302
9.2 TCP/IP 网络编程	307
9.2.1 概述	307
9.2.2 TCP 通信过程	309
9.2.3 基本 Socket 函数	310
9.2.4 Socket 编程的其他函数	313
9.2.5 TCP 编程举例	315
9.2.6 UDP 通信过程	317
9.2.7 UDP 通信服务器端实例	318
9.3 视频压缩与传输技术简介	320
9.3.1 H.264 视频编解码	320
9.3.2 RTP/RTCP 流媒体实时传输协议	322
实验 22: 图像合成实验	322
实验 23: 视频采集及 RTP 发送实验	326
实验 24: 远程控制视频实验	330
实验 25: 多 Client 单 Server 实验	334
参考文献	336

嵌入式系统概述

1.1 嵌入式系统简介

嵌入式系统（Embedded System）也称嵌入式计算机系统。顾名思义，嵌入式系统是计算机的一种特殊形式，是计算机技术、通信技术、半导体技术、微电子技术、语音图像数据传输技术，甚至传感器等先进技术和具体应用对象相结合后的更新换代产品。嵌入式系统不仅和一般的 PC 机上的应用系统不同，而且针对不同的具体应用而设计的嵌入式系统之间的差别也很大。嵌入式系统强调硬件和软件的协同性与整合性，软件和硬件可剪裁的，适用于应用系统对功能、可靠性、成本、体积、功耗和应用环境等有严格要求的专用计算机系统。

1.2 嵌入式系统的特点

嵌入式系统特别强调“量身定做”的原则，开发人员往往需要针对某一种特殊用途开发出一个截然不同的嵌入式系统，其特点如下。

（1）嵌入式系统具有应用针对性

应用针对性是嵌入式系统的一个基本特征，体现这种应用针对性的首先是软件，软件实现特定应用所需要的功能，所以嵌入式系统应用中必定配置了专用的应用程序；其次是硬件，大多数嵌入式系统的硬件是针对应用专门设计的，但也有一些标准化的嵌入式硬件模块，采用标准模块可降低开发的技术难度和风险，缩短开发时间，但灵活性不足。

（2）嵌入式系统硬件扩展能力要求不高

硬件上，嵌入式系统作为一种专用的计算机系统，其功能、机械结构、安装要求比较固定，所以一般没有或仅有较少的扩展能力；软件上，嵌入式系统往往是一个设备固定组成部分，其软件功能由设备的需求决定，在相对较长的生命周期里，一般不需要对软件进行改动。但也有一些特例，比如现在的手机，尤其是安装有嵌入式操作系统的智能手机，软件安装、升级比较灵活，但相对于桌面计算机，其软件扩展能力还是相当弱。

（3）嵌入式系统操作系统精简

在现代的通用计算机中，没有操作系统是无法想象的，而在嵌入式计算机中情况则大

不相同。在一个功能简单的嵌入式系统中,可能根本不需要操作系统,直接在硬件平台上运行应用程序;而一些功能复杂的嵌入式系统,可能需要支持有线/无线网络、文件系统、实现灵活的多媒体功能、支持实时多任务处理,此时,在硬件平台和应用软件之间增加一个操作系统层,可使应用软件的设计变得简单,而且便于实现更高的可靠性,缩短系统开发时间,使系统的研发工作变得可控。

目前存在很多种嵌入式操作系统,如 VxWorks、pSOS、嵌入式 Linux、WinCE 等,这些操作系统功能日益完善,以前只在桌面通用操作系统具备的功能,如网络浏览器、HTTP 服务器、Word 文档阅读与编辑等,也可以在嵌入式系统中实现。但为适应嵌入式系统的需要,嵌入式操作系统相对通用操作系统,具有模块化、结构精练、定制能力强、可靠性高、实时性好、便于写入非易失性存储器(固化)等特点。

(4) 嵌入式系统一般有实时性要求

设备中的嵌入式系统常用于实现数据采集、信息处理、实时控制等功能,而采集、处理、控制往往是一个连续的过程。一个过程要求必须在一定长的时间内完成,这就是系统实时性的要求。在如图 1-1 所示的语音处理系统中,可实现实时的数据采集、编码,并通过网络传输的功能,按照 8kHz 采样率、精度 8bits 的工作模式进行单通道语音采样,这时系统会以每秒 8KB 的速率连续产生数据,计算机需要“及时”地进行语音数据采集、数据压缩编码、通过网络发送数据等处理,任何一个环节处理不及时,都会导致语音数据丢失。

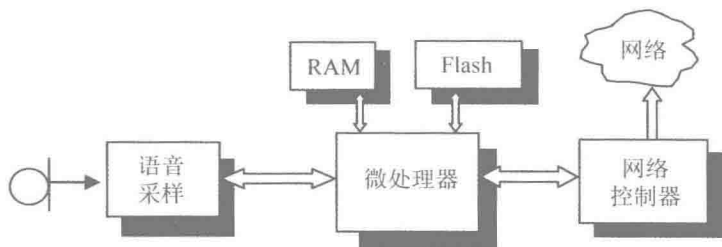


图 1-1 语音处理系统结构图

实时性和处理器速度并不相同,速度快的系统不一定实时性好,速度慢的系统实时性未必不能满足要求。计算机运行速度高,当然更有条件实现实时性,但不是实时性的充要条件。嵌入式系统的设计要求精练,因此在运算速度上不会留太多余量,为了保证实时性要求,更需要对硬件、软件精心设计。

(5) 嵌入式系统一般有较强的成本控制要求

在满足需求的前提下,在嵌入式系统开发中,要求高效率的设计,减少硬件、软件冗余,恰到好处的设计可以最大限度地降低系统成本,并有利于提高系统的可靠性。通用计算机则追求更快的计算速度、更大的存储容量、更丰富的配置、更大的显示器。强大的硬件平台才能满足日益复杂的桌面操作系统及各种类型软件的需要,这样的计算机“通用性”才最强。

(6) 嵌入式系统软件一般有固化的要求

在现代的通用计算机中,硬盘是操作系统和应用软件的载体,对于几 GB,甚至几十 GB、几百 GB 的软件及数据,硬盘是最好的记录媒介。嵌入式系统软件一般把操作系统和

应用软件直接固化在非易失性存储器（如 Flash 存储器）中。首先，嵌入式系统一般没有硬盘，就算有硬盘或存储卡之类的外部存储器，也很少用于存储系统软件，多用于存储数据或用户扩展的软件；其次，无论是操作系统还是应用软件都很精练，所占容量相对通用计算机要小得多，所以有固化的条件；再次，嵌入式系统不同于通用计算机那么容易安装和升级软件，而且也很少需要改动，所以要求软件存储可靠性高，因此有必要把软件固化；最后，软件固化有利于提高嵌入式系统的启动速度。

（7）嵌入式系统软件一般采用交叉开发的模式

目前软件设计工作大多采用集成开发环境，将代码编辑、编译、链接、仿真、调试等软件开发工具集成在一起。嵌入式系统针对具体的应用进行设计，其硬件、软件的配置往往不便于或不可能支持应用软件开发。实际开发中，一般用通用计算机（主要是 PC 机）作为开发机，进行嵌入式软件的编辑、编译、链接，在开发机上进行仿真，或下载到嵌入式目标系统中运行测试，最终的目标代码固化到目的系统的存储器中运行，这就是交叉开发的软件设计模式。

（8）嵌入式系统在体积、功耗、可靠性、环境适应性上一般有特殊要求

嵌入式系统作为一个固定的组成部分“嵌入”在设备中，因受装配、供电、散热等条件的约束，其体积、功耗必然有一定的限制。例如，现在的手机功能日益强大，但体积越做越小，集成度和装配密度非常高，在这种应用环境里，嵌入式计算机部分的芯片封装、电路板设计、系统装配等都要求紧凑、小巧。在功耗方面也有严格的要求，一方面密封在手机里，没有良好的散热条件，功耗控制不好会导致手机温度过高；另一方面，电路的功耗直接决定了手机一次充电后持续工作的时间。嵌入式系统作为设备的核心，其可靠性直接决定了设备可靠性，因此在这方面有严格的要求。

1.3 嵌入式系统的组成

嵌入式系统是具有应用针对性的专用计算机系统，应用时作为一个固定的组成部分“嵌入”在应用对象中。每个嵌入式系统都是针对特定应用定制的，所以彼此间在功能、性能、体系结构、外观等方面可能存在很大的差异，但从计算机原理的角度看，嵌入式系统包括硬件和软件两个组成部分。

图 1-2 给出的是一个典型的嵌入式系统组成，实际系统中可能并不包括所有的组成部分。嵌入式系统硬件部分以嵌入式处理器为核心，扩展存储器及外部设备控制器。在某些应用中，为提高系统性能，还可能为处理器扩展 DSP 或 FPGA 等作为协处理器，实现视频编码、语音编码及其他数字信号处理等功能。在一些 SOC（System on Chip）中，将 DSP 或 FPGA 与处理器集成在一个芯片内，降低系统成本，缩小电路板面积，提高系统可靠性。嵌入式系统软件部分，驱动层向下管理硬件资源，向上为操作系统提供一个抽象的虚拟硬件平台，是操作系统支持多硬件平台的关键。在嵌入式系统软件开发过程中，用户的主要精力一般在用户应用程序和设备驱动程序开发上。

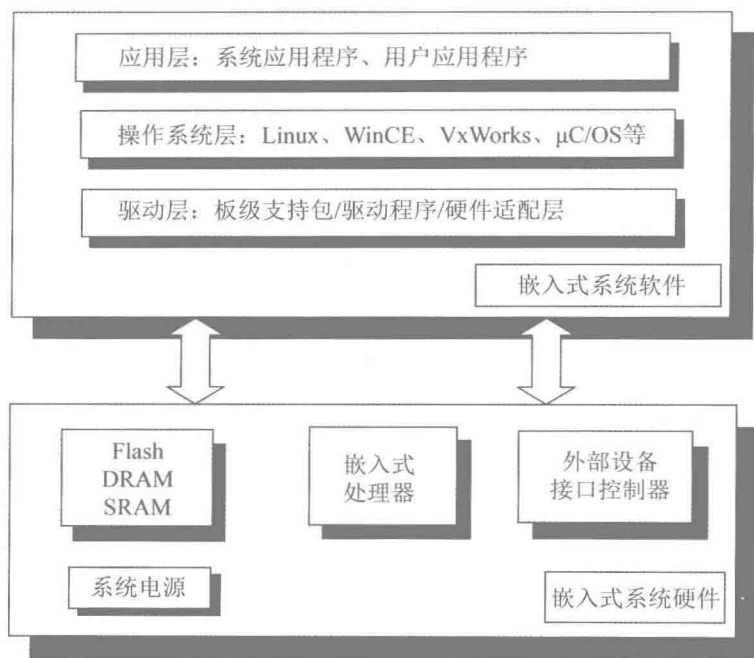


图 1-2 嵌入式系统组成

1.4 嵌入式系统的分类

根据不同标准，嵌入式系统有不同的分类方法。

1. 按处理器位宽分类

按处理器位宽可将嵌入式系统分为 4 位、8 位、16 位、32 位和 64 位系统，一般情况下，位宽越大，性能越强。对于通用计算机处理器，因为要追求尽可能高的性能，在发展历程中总是高位宽处理器取代、淘汰低位宽处理器。而嵌入式处理器不同，千差万别的应用对处理器要求也大不相同，因此不同性能的处理器都有各自的用武之地。

2. 按有无操作系统分类

现代通用计算机中，操作系统是必不可少的系统软件。在嵌入式系统中则有两种情况：有操作系统的嵌入式系统和无操作系统（裸机）的嵌入式系统。在有操作系统支持的情况下，嵌入式系统的任务管理、内存管理、设备管理、文件管理等都由操作系统完成，并且操作系统为应用软件提供丰富的编程接口，用户应用软件开发可以把精力都放在具体的应用设计上，这与在 PC 机上开发软件相似。在一些功能单一的嵌入式系统中，如基于 8051 单片机嵌入式系统，硬件平台很简单，系统不需要支持复杂的显示、通信协议、文件系统、多任务的管理等，这种情况下可以不用操作系统。

3. 按实时性分类

根据实时性要求，可将嵌入式系统分为软实时系统和硬实时系统两类。在硬实时系统中，系统要确保在最坏情况下的服务时间，即对事件响应时间的截止期限必须得到满足。

在这样的系统里,如果一个事件在规定期限内不能得到及时处理,则会导致致命的系统错误。在软实时系统中,从统计的角度看,一个任务能够得到确保的处理时间,到达系统的时间也能够在截止期限前得到处理,但截止期限条件没得到满足时并不会带来致命的系统错误。

4. 按应用分类

嵌入式系统应用在各行各业,按照应用领域的不同可对嵌入式系统进行分类。

(1) 消费类电子产品

消费类电子产品是嵌入式系统需求最大的应用领域,日常生活中的各种电子产品都有嵌入式系统的身影,从传统的电视、冰箱、洗衣机、微波炉,到数字时代的影碟机、MP3、MP4、手机、数码相机、数码摄像机等,在可预见的将来,可穿戴计算机也将走入日常生活。现代社会里,人们被各种嵌入式系统的应用产品包围着,嵌入式系统已经在很大程度上改变了人们的生活方式。

(2) 过程控制类产品

过程控制类产品的应用有很多,如生产过程控制、数控机床、汽车电子、电梯控制等。过程控制引入嵌入式系统可显著提高效率和精确性。

(3) 信息、通信类产品

通信是信息社会的基础,其中最重要的是各种有线、无线网络,在这个领域大量应用嵌入式系统,如路由器、交换机、调制解调器、多媒体网关、计费等。很多与通信相关的信息终端也大量采用嵌入式技术,如POS机、ATM自动取款机等。使用嵌入式技术的信息类产品还包括键盘、显示器、打印机、扫描仪等计算机外部设备。

(4) 智能仪器、仪表产品

嵌入式系统在智能仪器、仪表中大量应用,采用计算机技术不仅提高仪器、仪表性能,还可以设计出传统模拟设备所不具备的功能。如传统的模拟示波器能显示波形,通过刻度人为计算频率、幅度等参数,而基于嵌入式计算机技术设计的数字示波器,除更稳定地显示波形外,还能自动测量频率、幅度,甚至可以将一段时间里的波形存储起来,供事后详细分析。

(5) 航空、航天设备与武器系统

航空、航天设备与武器系统一向是高精尖技术集中应用的领域,如飞机、宇宙飞船、卫星、军舰、坦克、火箭、雷达、导弹、智能炮弹等,嵌入式计算机系统是这些设备的关键组成部分。

(6) 公共管理与安全产品

这类应用包括智能交通、视频监控、安全检查、防火防盗设备等。现在常见的可视安全监控系统已基本实现数字化,在这种系统中,嵌入式系统常用于实现数字视频的压缩编码、硬盘存储、网络传输等,在更智能的视频监控系统中,嵌入式系统甚至能实现人脸识别、目标跟踪、动作识别、可疑行为判断等高级功能。

(7) 生物、医学微电子产品

这类应用包括生物特征(指纹、虹膜)识别产品、红外温度检测、电子血压计、电子化的医学化验设备、医学检查设备等。

1.5 嵌入式系统的发展

信息时代、数字时代使得嵌入式产品获得了巨大的发展契机，为嵌入式市场展现了美好的前景，同时也对嵌入式生产厂商提出了新的挑战，从中我们可以看出未来嵌入式系统有以下几大发展趋势。

(1) 嵌入式开发是一项系统工程，因此要求嵌入式系统厂商不仅要提供嵌入式软硬件系统本身，同时还需要提供强大的硬件开发工具和软件包支持。

(2) 网络化、信息化的要求随着因特网技术的成熟、带宽的提高日益提高，使得以往单一功能的设备，如电话、手机、冰箱、微波炉等功能不再单一，结构更加复杂。ARM 公司推出的 ARM Cortex-M 处理器的计算性能和 DSP 处理能力是现有产品的两倍，可让用户以低成本满足高性能嵌入式应用需求，其适用范围包括马达控制、工业自动化、高级音频、图像处理、联网车载应用、物联网和穿戴式设备等相关应用。

(3) 网络互联成为必然趋势。未来的嵌入式设备为了适应网络发展的要求，必然要求硬件上提供各种网络通信接口。传统的单片机对于网络支持不足，而新一代的嵌入式处理器已经开始内嵌网络接口，除了支持 TCP/IP 协议，还支持 IEEE1394、USB、CAN、Bluetooth 或 IrDA 通信接口中的一种或者几种，同时也需要提供相应的通信组网协议软件和物理层驱动软件。软件方面系统内核支持网络模块，甚至可以在设备上嵌入 Web 浏览器，真正实现随时随地用各种设备上网。

(4) 精简系统内核、算法，降低功耗和软硬件成本。未来的嵌入式产品是软硬件紧密结合的设备，为了降低功耗和成本，需要设计者尽量精简系统内核，只保留和系统功能紧密相关的软硬件，利用最低的资源实现最适当的功能，这就要求设计者选用最佳的编程模型和不断改进算法，优化编译器性能。因此，既要软件人员有丰富的硬件知识，又需要发展先进嵌入式软件技术，如 Java、Web 和 WAP 等。

(5) 提供友好的多媒体人机界面。嵌入式设备能与用户亲密接触，最重要的因素就是能提供非常友好的用户界面，图像界面和灵活的控制方式，使得用户感觉嵌入式设备就像一个熟悉的老朋友。这方面的要求使得嵌入式软件设计者在图形界面、多媒体技术上多下苦功。手写文字输入、语音拨号上网、收发电子邮件以及彩色图形、图像都会使使用者获得自由的感受。

第 2 章

嵌入式微处理器概述

2.1 处理器原理

处理器，即中央处理器（CPU），是所有计算机的核心，在介绍嵌入式处理器前，先简单介绍一下处理器的原理。

2.1.1 CPU 的指令系统

指令是 CPU 能理解并执行的命令单元，一条完整的指令一般包括操作码和操作数两部分，操作码决定要完成的操作，而操作数则是操作过程中需要的数据或数据的地址。现代 CPU 都采用二进制表达方法，计算机硬件只识别 0 和 1 两个数字，所有的 CPU 指令都由这两个数字进行编码。例如，某个指令系统的指令长度为 32 位，操作码长为 8 位，地址码长度也为 8 位，当收到一个 00000010 00000100 00000001 00000110 指令时，先分析前面的 8 位操作码，判断为减法操作，后面是 3 个操作数地址，CPU 在 00000100 地址中取出被减数，在 00000001 地址取出减数，送到 CPU 的算术逻辑单元中进行减法运算，并将计算结果送到 00000110 地址中。一条指令只能完成一个简单的功能，如加/减运算、逻辑判断、读数据、写数据等，如果要完成复杂功能，就需要把很多指令组合起来协调执行，这些有机组合在一起的一串指令就是程序。

不同 CPU 支持的指令不同，CPU 支持的所有指令的集合就是该 CPU 的指令系统，如 x86 指令系统、ARM 指令系统、MCS-51 指令系统等。指令系统是 CPU 的基本属性，主要包括以下几种类型的指令。

- 算术运算指令：主要包括加、减、乘、除等数的计算。
- 逻辑运算指令：实现逻辑数的与、或、非、异或等逻辑运算。
- 数据传送指令：实现寄存器与寄存器、寄存器与存储单元以及存储单元与存储单元之间数据的传送。
- 移位操作指令：包括算术移位、逻辑移位和循环移位 3 种，实现对操作数左移、右移一位或若干位。
- 其他指令：除以上指令外还有一些其他指令，如堆栈操作指令、转移类指令、输入输出指令、多处理器控制指令、空操作指令等。

2.1.2 CPU 的分类

根据结构不同,可把处理器分为哈佛结构处理器和冯·诺依曼结构处理器。在冯·诺依曼结构处理器中,程序指令和数据采用统一的存储器,对数据和指令的寻址不能同时进行,只能交替完成。有别于冯·诺依曼结构,哈佛结构的处理器中数据和指令分开存储,通过不同的总线访问,具体特点体现在两个方面:

(1) 程序存储器和数据存储器分离,分开存储指令和数据,使用两套彼此独立的存储器总线,CPU 通过两套总线分别读、写程序存储器或数据存储器。

(2) 哈佛结构的处理器中,因为有两套相互独立的指令和数据存储器总线,因此可以同时进行指令和数据的访问,从而提高系统性能。而改进的哈佛结构处理器中,独立的存储总线可以不止两套,例如有多套数据存储器总线,可以进一步提高数据访问的速度。

根据指令格式的不同可以把 CPU 分为 CISC (Complex Instruction Set Computer, 复杂指令集计算机) 处理器和 RISC (Reduced Instruction Set Computer, 精简指令集计算机) 处理器。早期的 CPU 都采用 CISC 指令,最典型的就是 x86 系列处理器。CISC 的特点是有大量复杂的指令、指令长度可变、寻址方式多样。

在计算机发展之初,CPU 指令系统包含很少的指令,一些复杂的操作通过简单指令的组合来实现,如两个数 a 和 b 相乘可以用 a 个 b 的加法来实现。随着集成电路技术的迅速发展,为了软件编程方便和提高程序运行速度,在 CPU 的设计中不断增加可实现复杂功能的指令,如乘法运算中直接使用乘法指令而不是多个数的累加。随着复杂指令增多,CPU 指令系统变得越来越复杂。而指令越来越多、越来越复杂带来另一个问题,因为指令系统的指令数量由指令操作码的位数决定,例如指令数量为 n ,指令码位数为 k ,则 $n=2^k$ 。CPU 指令编码宽度不可能随意增加,促使操作码扩展技术出现。假如操作码为 2 位,则正常情况可表示 4 个指令,分别是 00、01、10、11。为了增加指令长度,如果把编码 11 作为扩展码,并把操作码扩展到 4 位,则该指令系统就有 00、01、10、1100、1101、1110、1111 这 7 条指令,这就是长度可变的操作码编码方式。

上述这种具有大量复杂指令、指令长度可变且寻址方式多样的指令系统就是传统 CISC 指令系统。一方面,采用复杂指令系统的计算机有着较强的处理高级语言的能力,这有利于提高计算机的性能;但另一方面,复杂的指令、变长的编码、灵活的寻址方式大大增加了指令解码的难度,而随着硬件的高速发展,复杂指令所带来的速度提升已不及在解码上浪费的时间。

IBM 公司于 1975 年组织力量研究指令系统的合理性问题,发现 CISC 存在一些缺点:首先,在这种计算机中,各种指令的使用率差别很大,一个典型程序的运算过程所使用的 80% 指令,只占一个处理器指令系统的 20%,最频繁使用的是取、存和加这些最简单的指令,而占指令数 80% 的复杂指令却只有 20% 的机会用到。复杂的指令系统必然导致结构的复杂,从而增加了设计、制造的难度,尽管大规模集成电路技术已发展到很高的水平,但也很难把 CISC 的全部硬件集中在一个芯片上,妨碍了单片机的发展;另外,在 CISC 中,许多复杂指令需要完成复杂的操作,这类指令多数是某种高级语言的直接翻版,因而通用性差,采用二级的微码执行方式,降低了那些被频繁调用的简单指令系统的运行速度。针对 CISC 的弊端,业界提出了精简指令的设计思想,即指令系统应当只包含那些使用频率很高的少量指令,并提供一些必要的指令以支持操作系统和高级语言,按照这个原则发展

而成的计算机被称为精简指令集计算机结构,简称 RISC。RISC 的最大特点是指令长度固定,指令种类少,寻址方式种类少,大多数是简单指令且都能在一个时钟周期内完成,易于设计超标量与流水线,寄存器数量多,大量操作在寄存器之间进行。一般认为 RISC 处理器有以下几个方面的优点:

(1) 芯片面积小

实现精简的指令系统需要的晶体管少,芯片面积自然就小一些。节约的面积可以用于实现提高性能的功能部件,如高速缓存、存储器管理和浮点运算器等,也便于在单片上集成更多其他模块,如网络控制器、语音/视频编码器、SDRAM 控制器、PCI 总线控制器等。

(2) 开发时间短

开发一个结构简洁的处理器在人力、物力上的投入要更少,整个开发工作的时间更易于预测且可控制。

(3) 性能高

在 CISC 处理器中,一些复杂的操作有专用的指令,对于单个的操作使用专用指令可以提高处理效率,但复杂指令的使用降低了所有其他指令的执行效率。完成同样功能的程序时,RISC 处理器需要更多的指令,但 RISC 单个指令执行效率高,而且 RISC 处理器容易实现更高的工作频率,从而使整体性能得到提高。RISC 处理器性能上的优点在处理器发展的实践中得到验证。

目前,通用计算机,如 PC 机、服务器等大多采用 CISC 结构的 x86 处理器,随着技术的发展,新的 x86 处理器融合了 RISC 的特性。在嵌入式处理器中,RISC 技术则得到普遍的应用,如 MIPS 处理器、ARM 处理器等。

2.1.3 CPU 的结构

CPU 典型组成部分包括运算器、控制器、寄存器阵列及连接各个部分的内部总线,如图 2-1 所示。运算器包括算术逻辑单元、累加器、暂存器及标志寄存器等,完成加、减、乘、除四则运算及各种逻辑运算。

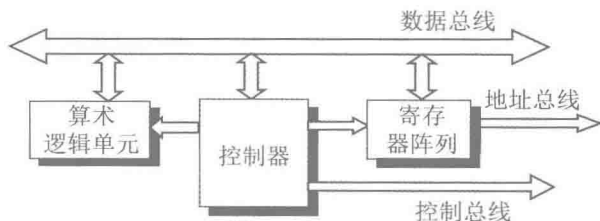


图 2-1 CPU 内部结构

控制器由指令寄存器、指令译码器、控制逻辑电路组成,完成 CPU 的全部控制功能。CPU 从存储器取出指令,通过数据总线存入指令寄存器,然后由指令译码器对指令进行译码。译码产生的结果与时钟信号配合,在控制逻辑电路中产生各种操作所必需的控制信号。控制信号经由控制总线送到微处理器系统的其他功能部件中,以便执行各种操作。CPU 在操作过程中需要获取数据并产生一些新数据,为了提高 CPU 的速度,在 CPU 内部设计一组临时存储器单元,用于操作数据及中间结果的存放与转移,这就是 CPU 的寄存器阵列。

如图 2-1 所示是 CPU 的最基本组成,现在一般意义的 CPU 内部包含的内容要丰富得