

高等学校计算机专业特色教材

嵌入式系统 原理与应用

*Principle and Application
of Embedded System*

王忠民 主编

 高等教育出版社
HIGHER EDUCATION PRESS

高等学校计算机专业特色教材

嵌入式系统原理与应用

Qianrushixitong Yuanli yu Yingyong

王忠民 主编



高等教育出版社·北京
HIGHER EDUCATION PRESS BEIJING

内容简介

本书以 ARM7 处理器和 $\mu\text{C}/\text{OS}-\text{II}$ 操作系统为主线, 结合大量实例, 从系统的角度详细介绍了嵌入式系统开发过程中所需要掌握的软、硬件入门基础知识。本书主要内容包括嵌入式系统基础、ARM 体系结构、ARM 指令系统、ARM 汇编语言程序设计基础、LPC2000 系列处理器原理、LPC2000 系列微处理器接口技术、嵌入式操作系统等。每章开始给出本章简介, 包括本章的主要内容、重点难点及其在整个课程中所处的地位, 最后给出本章小结和习题。本书体系合理, 条理清晰, 语言简练, 旨在使读者能够从系统角度掌握嵌入式系统的基本组成及工作原理, 并具有嵌入式应用系统开发的基本能力。

本书可作为普通高等学校工科各相关专业学生学习“嵌入式系统”课程的教材, 也可供研究生及广大从事嵌入式系统开发的工程技术人员参考。

图书在版编目 (C I P) 数据

嵌入式系统原理与应用 / 王忠民主编. -- 北京 : 高等教育出版社, 2011. 12
ISBN 978-7-04-033840-9

I. ①嵌… II. ①王… 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(2011)第232849号

策划编辑 刘 艳 责任编辑 李善亮 封面设计 于 涛 版式设计 马敬茹
插图绘制 尹 莉 责任校对 金 辉 责任印制 张泽业

出版发行	高等教育出版社	咨询电话	400-810-0598
社 址	北京市西城区德外大街4号	网 址	http://www.hep.edu.cn
邮政编码	100120		http://www.hep.com.cn
印 刷	三河市华润印刷有限公司	网上订购	http://www.landaco.com
开 本	787mm×1092mm 1/16		http://www.landaco.com.cn
印 张	15.5	版 次	2011年12月第1版
字 数	340千字	印 次	2011年12月第1次印刷
购书热线	010-58581118	定 价	25.00元

本书如有缺页、倒页、脱页等质量问题, 请到所购图书销售部门联系调换
版权所有 侵权必究
物料号 33840-00

前 言

随着后 PC 时代的到来，嵌入式技术已经渗透到人们生活的各个方面，正在成为当前发展最为迅速的一个应用领域，对于工科各相关专业本科生来说，学习和掌握嵌入式技术具有重要的现实意义。

嵌入式系统课程内容庞杂，涉及嵌入式微处理器的体系结构与工作原理、汇编语言指令系统及其编程、嵌入式操作系统原理及其移植与扩展、基于嵌入式操作系统的应用程序开发等技术。本书力求将如此众多的内容很好地组织到一起，形成一个结构合理、层次清晰的嵌入式系统开发入门基础教材，以满足目前由于课时限制，无法单独开设嵌入式系统系列课程的普通本科学校工科各相关专业学生学习嵌入式系统的需要。

“嵌入式系统”可以追溯到 20 世纪 70 年代，在以 Zilog 公司的 Z80 处理器为核心的单板机，以 Intel 公司的 51 系列单片机为核心的监测控制系统等中都得到了广泛应用，但由于当时系统硬件资源的限制，没有操作系统，系统上电后在一个循环运行的监控程序的控制下完成单任务工作。近年来，嵌入式技术得到了迅猛发展，尤其是由于后 PC 时代的到来，一些传统的嵌入式技术公司又从 PC 领域重新回到嵌入式领域，致使嵌入式市场出现了前所未有的竞争格局。嵌入式技术本身的特点决定了嵌入式市场不会出现像 PC 市场那样由少数几家公司垄断的局面，没有哪一个嵌入式系列的处理器和操作系统能够垄断全部市场。激烈的市场竞争必将推动嵌入式技术的快速发展，但也为嵌入式系统课程教材内容的选取提出了挑战。从市场占有、技术发展前景以及目前市场上实验教学仪器等情况综合考虑，本书选择 ARM7TDMI 核、NXP（原属于飞利浦）公司的 LPC2000 系列微处理器芯片和 $\mu\text{C}/\text{OS-II}$ 操作系统为主线，以合理的体系、清晰的条理、简练的语言将嵌入式系统涉及的方方面面的知识在有限的篇幅内简明扼要地交代清楚，从系统角度使读者掌握嵌入式系统的基本组成及工作原理，具有一定的嵌入式应用系统开发的基本技能，为后续的进一步学习和从事嵌入式系统开发打下较为系统全面的基础。

全书共分为 8 章。第 1 章的主要内容为嵌入式系统基础，在介绍了嵌入式系统的基本概念、发展历史及其特点的基础上，重点介绍计算机体系结构及指令集、ARM 体系结构版本、ARM 处理器 IP 核及其命名方法、ARM 系列微处理器芯片主要制造商等入门知识，引导读者进入 ARM 技术的殿堂。第 2 章的主要内容为 ARM 体系结构，围绕 ARM7TDMI 核，从应用系统开发人员的角度，介绍了 ARM 的内核结构、工作状态、工作模式以及应用程序可访问的寄存器结构、存储器组织、异常响应与处理等内容。第 3 章和第 4 章的主要内容为软件部分，详细介绍了 ARM 指令系统、ARM 汇编语言程序设计方法，以及汇编语言与 C 语言的混合编程技术等，无论是从事底层开发还是从事顶层应用软件开发读者都需要很好地掌握这两章的内容。第 5 章和第

II 前言

6 章的主要内容为硬件部分，在对 LPC2000 系列微处理器芯片的基本结构及性能参数进行对比的基础上，以较常用的 LPC2131 芯片为例，介绍了 LPC2000 系列微处理器芯片的内部结构、基本工作原理及引脚配置，介绍了 LPC2000 系列微处理器芯片的常用片内 I/O 及其初始化编程技术。第 7 章的主要内容为嵌入式操作系统，介绍了嵌入式操作系统的基本概念，几种主流嵌入式操作系统，重点介绍开放源代码的嵌入式实时操作系统 $\mu\text{C}/\text{OS}-\text{II}$ ，包括 $\mu\text{C}/\text{OS}-\text{II}$ 的任务管理、任务调度、中断处理、时钟管理、任务间同步与通信以及内存管理等内容。第 8 章的主要内容为基于 $\mu\text{C}/\text{OS}-\text{II}$ 的嵌入式应用程序开发，在介绍了嵌入式应用程序开发的特点基础上，介绍了 $\mu\text{C}/\text{OS}-\text{II}$ 在 ARM 微处理器上移植的基本思路，详细介绍了嵌入式应用程序开发所涉及的一些关键问题。

本书由王忠民担任主编，负责本书大纲拟定和统稿工作，并编写了第 1 章、第 2 章和第 5 章；贺炎编写第 3 章；范琳编写第 4 章；胡滨编写第 6 章和附录；梁琛编写第 7 章；王文浪编写第 8 章。

由于作者水平所限，书中难免存在一些不足与疏漏之处，恳请读者不吝指正（wzm_678@163.com）。

编者

2011 年 1 月

目 录

第 1 章 嵌入式系统基础	1	2.5.3 存储格式	24
1.1 引言	1	2.6 异常	24
1.2 嵌入式系统的发展历史	2	2.6.1 ARM 异常类型	24
1.3 嵌入式系统的特点	3	2.6.2 ARM 异常的处理	25
1.4 嵌入式系统的组成	4	2.6.3 ARM 异常的优先级	28
1.4.1 嵌入式系统硬件组成	5	本章小结	28
1.4.2 嵌入式系统软件组成	6	习题	29
1.5 ARM 微处理器	6	第 3 章 ARM 指令系统	30
1.5.1 计算机体系结构与指令集	7	3.1 ARM 处理器的指令格式	30
1.5.2 ARM 体系结构版本	8	3.1.1 ARM 指令集的特点	30
1.5.3 ARM 处理器核	9	3.1.2 ARM 指令的格式	31
1.5.4 主要 ARM 芯片供应商	11	3.2 ARM 指令的寻址方式	32
本章小结	12	3.2.1 立即寻址	33
习题	13	3.2.2 寄存器寻址	33
第 2 章 ARM 体系结构	14	3.2.3 寄存器移位寻址	34
2.1 ARM 核及基于 ARM 核的 芯片结构	14	3.2.4 直接寻址	35
2.1.1 ARM 内核	15	3.2.5 寄存器间接寻址	35
2.1.2 基于 ARM 核的芯片结构	15	3.2.6 基址变址寻址	36
2.2 ARM 工作状态和工作模式	16	3.2.7 多寄存器寻址	37
2.2.1 ARM 处理器工作状态	16	3.2.8 堆栈寻址	38
2.2.2 ARM 处理器工作模式	17	3.2.9 相对寻址	40
2.3 ARM 状态下寄存器组织	17	3.3 ARM 指令集	40
2.3.1 通用寄存器	18	3.3.1 加载/存储指令	40
2.3.2 程序状态寄存器	19	3.3.2 数据处理指令	44
2.4 Thumb 状态下寄存器组织	21	3.3.3 分支指令	47
2.5 ARM 存储器组织	23	3.3.4 程序状态寄存器访问指令	48
2.5.1 编址方式与寻址空间	23	3.3.5 异常中断产生指令	49
2.5.2 数据类型	23	3.4 Thumb 指令集	50
		本章小结	52

II 目录

习题	52	本章小结	105
第4章 ARM 汇编语言程序设计基础	55	习题	106
4.1 ARM 汇编语言的程序结构	55	第6章 LPC2000 系列微处理器	
4.1.1 一个简单的 ARM 汇编程序例子	55	接口技术	107
4.1.2 汇编器伪指令	57	6.1 GPIO	107
4.1.3 汇编语言的规范	59	6.1.1 概述	107
4.2 ARM 汇编语言程序设计	60	6.1.2 引脚描述	108
4.2.1 顺序程序	61	6.1.3 寄存器描述	108
4.2.2 分支程序	64	6.1.4 应用举例	109
4.2.3 循环程序	66	6.2 定时器	110
4.2.4 子程序	69	6.2.1 概述	110
4.2.5 基于查表法的程序设计	72	6.2.2 引脚描述	111
4.3 C 语言与汇编混合编程	75	6.2.3 结构	112
4.3.1 C 语言内嵌汇编代码	75	6.2.4 寄存器描述	113
4.3.2 C 语言与汇编文件变量互访	77	6.2.5 应用举例	114
4.3.3 ARM 过程调用标准 APCS	79	6.3 UART 串口通信	117
4.3.4 C 语言程序调用汇编子程序	80	6.3.1 概述	117
4.3.5 汇编程序调用 C 语言函数	81	6.3.2 引脚描述	117
本章小结	82	6.3.3 结构	117
习题	82	6.3.4 寄存器描述	118
第5章 LPC2000 系列处理器原理	84	6.3.5 应用举例	120
5.1 LPC2000 系列处理器简介	84	6.4 其他外设接口	122
5.1.1 LPC2000 系列处理器特性	84	6.4.1 看门狗	122
5.1.2 LPC2000 系列处理器结构	85	6.4.2 实时时钟	124
5.2 处理器引脚配置	86	6.4.3 模数转换器	125
5.3 存储器管理	93	6.4.4 I ² C 接口	127
5.3.1 存储器映射	93	6.4.5 SPI 接口	128
5.3.2 存储器重映射及系统的 引导与启动	95	6.4.6 脉宽调制器 (PWM) 接口	130
5.4 系统控制模块	98	本章小结	133
5.4.1 概述	98	习题	133
5.4.2 时钟部件	100	第7章 嵌入式操作系统	134
5.4.3 系统复位	101	7.1 嵌入式操作系统概述	134
5.4.4 功率控制	101	7.1.1 什么是嵌入式操作系统	135
5.4.5 ARM 中断系统	102	7.1.2 嵌入式操作系统的特点	136
		7.1.3 嵌入式操作系统的分类	136

7.2 主流嵌入式操作系统简介·····	137	8.3.4 任务间的数据通信方法·····	186
7.3 $\mu\text{C}/\text{OS}-\text{II}$ 操作系统·····	139	8.4 $\mu\text{C}/\text{OS}-\text{II}$ 在 ARM 微处 理器上的移植·····	188
7.3.1 $\mu\text{C}/\text{OS}-\text{II}$ 简介·····	140	8.4.1 $\mu\text{C}/\text{OS}-\text{II}$ 对处理器的要求·····	188
7.3.2 $\mu\text{C}/\text{OS}-\text{II}$ 内核结构·····	141	8.4.2 $\mu\text{C}/\text{OS}-\text{II}$ 移植所涉及的文件·····	189
7.3.3 任务·····	142	8.4.3 $\mu\text{C}/\text{OS}-\text{II}$ 的移植过程及内容·····	190
7.3.4 任务调度·····	148	8.4.4 $\mu\text{C}/\text{OS}-\text{II}$ 的裁剪·····	196
7.3.5 $\mu\text{C}/\text{OS}-\text{II}$ 的初始化与启动·····	152	8.5 应用程序设计及实例·····	198
7.3.6 任务管理·····	153	8.5.1 硬件及其运行环境初始化·····	198
7.3.7 中断处理和时钟管理·····	153	8.5.2 应用程序设计·····	211
7.3.8 任务间同步与通信·····	156	本章小结·····	218
7.3.9 内存管理·····	162	习题·····	219
本章小结·····	165	附录 A ARM 开发工具 ADS·····	220
习题·····	165	A.1 ADS 简介·····	220
第 8 章 基于 $\mu\text{C}/\text{OS}-\text{II}$ 的嵌入式 应用程序开发 ·····	167	A.1.1 嵌入式系统开发流程·····	220
8.1 嵌入式应用程序开发的特点·····	167	A.1.2 ADS 环境下 ARM 程序 开发方式·····	221
8.1.1 开发调试环境的建立·····	167	A.1.3 ADS 工具介绍·····	222
8.1.2 基于嵌入式处理器的直接 编程技术·····	168	A.2 ADS 及 AXD 使用简述·····	222
8.1.3 基于嵌入式操作系统的 编程技术·····	169	A.2.1 创建工程项目·····	222
8.2 $\mu\text{C}/\text{OS}-\text{II}$ 应用程序结构分析·····	170	A.2.2 工程项目管理窗口·····	224
8.3 $\mu\text{C}/\text{OS}-\text{II}$ 程序设计技术·····	173	A.2.3 ADS 编译器·····	226
8.3.1 任务的划分与设计·····	173	A.2.4 ADS 链接器·····	227
8.3.2 任务间的行为同步方法·····	176	A.2.5 AXD 使用步骤·····	228
8.3.3 共享资源的同步方法·····	183	A.3 ADS 环境汇编程序设计实例·····	228
		参考文献·····	236

第 1 章

嵌入式系统基础

嵌入式系统应用非常广泛，正在成为当前发展最为迅速的一个应用领域。对于工科各专 业本科生来说，学习和掌握嵌入式系统具有非常重要的现实意义。本章首先介绍嵌入式系统的基本概念及其发展历史，嵌入式系统的软硬件组成，然后重点介绍计算机体系结构及指令集、ARM 体系结构版本、ARM 处理器 IP 核（Intellectual Property Core）及其命名方法、ARM 系列微处理器芯片主要制造商等入门知识，引导读者进入 ARM 技术的殿堂。

1.1 引言

计算机作为 20 世纪人类社会最伟大的发明之一，在期待着第 5 代出现的同时，也迈入了其另一个新阶段——后 PC 时代。后 PC 时代的到来，使得人们开始越来越多地接触到一个新的概念——嵌入式产品，如手机、PDA、DVD 和机顶盒等。

20 世纪 90 年代后期，人类在经历了桌面系统的空前繁荣之后，由于信息家电、移动通信、手持信息设备以及汽车电子等的发展需要，嵌入式系统得到了迅猛发展，在工业、军事、通信、金融、医疗、农业、气象等领域得到了广泛应用，嵌入式系统已经变得无所不在。

关于“嵌入式系统”的定义有很多种说法。根据 IEEE（国际电气和电子工程师协会）的定义，嵌入式系统是控制、监视或者辅助装置、机器和设备运行的装置，这主要是从产品的应用角度加以定义的。按照目前企业界和学术界对嵌入式系统的普遍看法，嵌入式系统被定义为：以应用为中心，以计算机技术为基础，软硬件可裁剪，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。也有学者给出了更为通俗的定义：嵌入到对象体系中的专用计算机系统，由嵌入式微处理器、外围硬件设备、嵌入式操作系统以及用户的应用程序4部分组成。建议读者没有必要对嵌入式系统的定义做过多的毫无意义的追究。

“嵌入式系统原理及应用”这门课程涉及的知识点很多，内容包括嵌入式系统基本概念、ARM 处理器内部结构及工作原理、ARM 指令及伪指令、ARM 汇编/C 程序设计、嵌入式操作系统及其裁剪移植、基于操作系统的嵌入式应用程序开发，以及嵌入式系统设计技术等。不仅涉及硬件原理、指令系统及编程，还涉及操作系统及其裁剪移植，基于操作系统的应用程序开发等内容。本书以合理的体系、清晰的条理、简练的语言将嵌入式系统涉及的方方面面的知识在有限的篇幅内简明扼要地交代清楚，从系统角度使读者掌握嵌入式系统的基本组成及工作原理，具有一定的嵌入式应用系统开发的基本技能。

注：IP 核（Intellectual Property Core）是一段具有特定电路功能的硬件描述语言程序，该程序与集成电路工艺无关，可以移植到不同的半导体工艺中去生产集成电路芯片。

1.2 嵌入式系统的发展历史

虽然嵌入式系统是近年风靡起来的，但这个概念并非新近才出现。事实上，嵌入式系统是伴随着 20 世纪 70 年代微型计算机的诞生而出现的。微型计算机由于其具有体积小、价格低、可靠性高等特点，一经推出很快就得到了广泛的使用，而微型计算机所表现出来的智能化促使人们将其嵌入到一个对象体系中，实现对象体系的智能化控制，从而导致了嵌入式计算机系统的出现。如果说微型计算机的出现，使计算机进入到现代计算机发展阶段，那么嵌入式计算机系统的诞生，则标志着计算机进入了通用计算机系统与嵌入式计算机系统两大分支并行发展的时代。通用计算机系统正朝着高速、海量存储、多媒体信息处理等方向发展，以满足人们办公、科学研究以及娱乐的需要。而嵌入式计算机系统则朝着小型化、专用化方向发展，以满足各类不同智能产品的控制需要。

从 20 世纪 70 年代单片机的出现到今天各式各样的嵌入式微处理器、微控制器的大规模应用，嵌入式系统已经有了 40 多年的发展历史，其硬件以嵌入式微处理器为代表经历了 4 位、8 位、16 位、32 位等几个阶段，其性能日益提高，资源不断丰富；软件规模和复杂度也呈爆炸式增长，嵌入式操作系统成为嵌入式应用开发的核心支持系统。纵观嵌入式技术的发展过程，粗略可分为以下 3 个发展阶段。

第一阶段（20 世纪 70 年代—20 世纪 80 年代末），是基于单片机的嵌入式应用系统。20 世纪 70 年代单片机的出现，使得汽车、家电、通信装置等可以通过内嵌电子装置来获得更佳的使用性能。这些装置已经初步具备了嵌入式的应用特点，但这时的系统使用 8 位的芯片（有代表性的为 Intel 公司的 8048、8051；Zilog 公司的 Z80；Motorola 公司的 68HC05），一般没有操作系统，使用汇编语言编程，只能执行一些单线程的程序，还谈不上“系统”的概念。

第二阶段（20 世纪 80 年代末—20 世纪 90 年代末），是以嵌入式 CPU 为基础，以嵌入式操

作系统为核心的嵌入式系统。可以认为在这个阶段“嵌入式系统”真正出现了。在这一阶段，嵌入式系统的软硬件技术加速发展，应用领域不断扩大。

第三阶段（20 世纪 90 年代末开始至今），是以 Internet 为标志的嵌入式系统。这是一个正在迅速发展的阶段。目前大多数嵌入式系统还孤立于 Internet 之外，但随着 Internet 的发展以及 Internet 技术与信息家电、通信技术、工业控制技术结合的日益密切，嵌入式设备与 Internet 的结合将代表嵌入式系统的发展方向。

经过 40 多年的发展，嵌入式系统技术正日益完善，32 位嵌入式微处理器已成为系统主流，嵌入式操作系统也从简单走向成熟，嵌入式系统与 Internet 的结合日益密切，其应用领域将不断扩大。

1.3 嵌入式系统的特点

为了满足不同的工作需要，计算机技术形成了两个相互独立的发展方向：通用桌面计算机系统 and 嵌入式计算机系统。与通用桌面计算机系统相比，嵌入式计算机系统有着自己独有的特点。

1. 交叉开发环境

与通用桌面计算机系统不同，嵌入式计算机系统的开发环境与运行环境是不一致的，需要通过交叉开发环境来完成嵌入式系统的开发调试工作，如图 1.1 所示。通常嵌入式计算机的资源有限，需要借助功能更强大的宿主机（通常为桌面 PC）以及基于宿主机的仿真开发环境来完成程序的开发调试工作，调试无误后可下载到目标机（嵌入式计算机系统）运行。

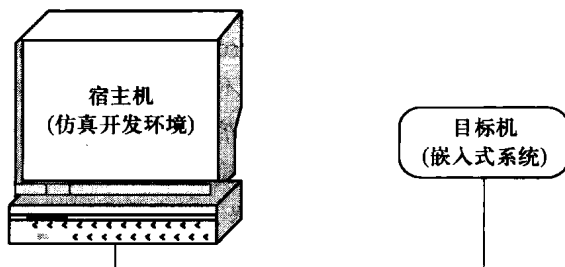


图 1.1 嵌入式交叉开发环境

2. 面向特定应用的技术密集型专用产品

嵌入式系统是计算机技术、微电子技术和各行业技术相结合的产物，是一个技术密集、面向特定应用被嵌入到专用产品中的计算机系统。嵌入式处理器的功耗、体积、成本、可靠性、速度、处理能力和电磁兼容性等方面均受到应用要求的制约，这些也是各个半导体厂商之间竞争的热点。和通用计算机不同，嵌入式系统的硬件和软件都必须高效率地设计，量体裁衣、去除冗余，力争在同等资源环境中实现更高的性能。嵌入式系统和具体应用有机地结合在一起，它的升级换代也是和具体产品同步进行，因此嵌入式系统产品一旦进入市场，具有较长的生命周期。

3. 嵌入式系统的实时性要求

大多数嵌入式系统工作在对实时性要求较高的场合，主要对仪器设备的动作进行检测控制，这种动作具有严格的、机械的时序；而一般的桌面计算机系统基本上是人通过键盘或鼠标发出的命令进行工作，对实时性通常没有太高要求。

为了提高执行速度，嵌入式系统中的软件一般都固化在存储器芯片或单片机中，要求程序编写和编译工具的质量要高，以减少程序二进制代码长度，满足实时性要求。在实际应用环境中，嵌入式系统处理的外部事件往往不是单一的，而且这些事件随机发生，可能同时出现。因此嵌入式系统还具有多任务分布和并发的特点。在这种多任务嵌入式系统中，对重要性各不相同的任务必须进行合理调度，以确保每个任务在规定的时间内被执行。

4. 嵌入式系统的安全可靠性要求

系统的安全可靠是不容忽视的问题，也是嵌入式系统的一个重要特征和发展方向。一般来说，希望设备中的嵌入式系统可以不出错地连续运行，或者出现错误也可以自行修复，而不需要也不大可能进行人工干预，因此，嵌入式系统通常有着极高的可靠性要求。

现在的智能设备几乎都提供了网络功能，随之而来的就是如何确保嵌入式智能设备的网络信息安全问题。目前，各种病毒与黑客程序正在虎视眈眈地盯着人们使用的各类嵌入式智能设备，嵌入式系统的安全已经受到极大的挑战，也开始逐步成为一个广受关注的研究课题。

5. 嵌入式产业具有不可垄断性

从某种意义上来说，通用计算机行业的技术是垄断的。硬件上有 80% 以上采用 Intel 公司的 x86 体系结构，芯片基本上来自 Intel、AMD、Cyrix 等几家公司。软件方面，通用计算机所必备的操作系统和办公软件，Microsoft 公司的 Windows 及 Office 占到 80%~90% 的份额。因此，当代通用计算机工业的基础被认为是由 Wintel (Microsoft 公司和 Intel 公司于 20 世纪 90 年代初建立的联盟) 垄断的工业。而嵌入式计算机系统则不同，它是一个高度分散、充满竞争与机遇的领域，没有哪一个系列的处理器和操作系统能够垄断全部市场。即便在体系结构上存在着主流系列，但各不相同的应用领域决定了不可能由少数公司、少数产品垄断全部市场。另外，随着嵌入式应用领域的不断扩大，对嵌入式微处理器、嵌入式操作系统及应用软件提出更高的要求，构成了推动嵌入式工业发展的强大动力，为众多的中小型嵌入式企业提供了广阔的发展空间。

1.4 嵌入式系统的组成

与通用计算机系统一样，嵌入式计算机系统也是由硬件和软件两大部分组成。前者是整个系统的物理基础，提供软件运行平台和通信（包括人机交互）接口；后者实际控制系统的运行。整个系统的体系结构如图 1.2 所示。

由于嵌入式系统应用环境的特殊性，在嵌入式系统设计时必须考虑操作系统及其他系统

软件模块与硬件之间的接口形式。经过多年的发展，随着通用嵌入式操作系统技术的日趋成熟和应用的不断扩大，一种统一的接口形式得到广泛的认可和应用，这就是通常所说的板级支持包，即 BSP (Board Support Package)，如图 1.2 所示，相当于通用计算机系统 BIOS。BSP 是一个介于操作系统和底层硬件之间的软件层次，包括了系统中大部分与硬件相关的软件模块。在功能上包含两部分：系统初始化及与硬件相关的设备驱动。由于引入 BSP，屏蔽了底层硬件的多样性，操作系统不再直接面对具体的硬件环境，大大推动了嵌入式实时操作系统的通用化，从而为嵌入式系统的广泛应用提供了可能。

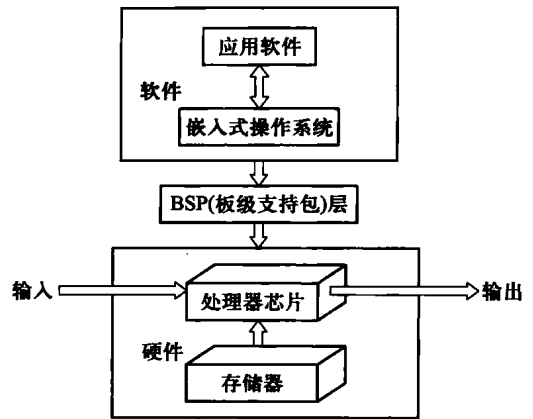


图 1.2 嵌入式系统的组成

1.4.1 嵌入式系统硬件组成

图 1.3 所示，嵌入式系统的硬件部分由嵌入式 ARM 芯片（包括 ARM 微处理器 IP 核、片内存储器（SRAM/Flash）以及各种片内 I/O）、片外存储器、各种人机交互接口以及电源与时钟等组成。其中 ARM 微处理器 IP 核是由英国集成电路设计公司 ARM (Advanced RISC Machines) 开发的，ARM 公司本身并不生产嵌入式处理器芯片，只向半导体厂家提供其设计的功能稳定、功耗很低的微处理器 IP 核，半导体厂家根据其芯片的市场定位，在购买来的 ARM 微处理器 IP 核的基础，集成应用领域所需要的各种 I/O 电路及存储器系统，投片生产面向某些应用领域的嵌入式 ARM 芯片。

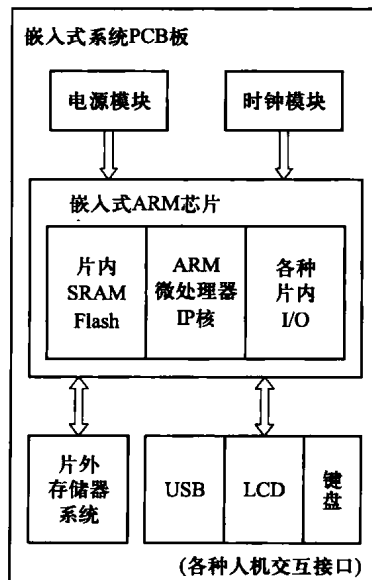


图 1.3 嵌入式系统硬件组成

1.4.2 嵌入式系统软件组成

图 1.4 所示为嵌入式软件由板级支持包 BSP (Board Support Packet)、嵌入式实时操作系统 RTOS (Real-Time Operating System)、应用程序编程接口 API (Application Programming Interface) 以及嵌入式应用程序 4 个不同层次组成。

① 板级支持包 (BSP): 板级支持包是软件模块的最底层。主要目的是屏蔽下层硬件。该层具有两种功能: 一是系统引导, 包括嵌入式微处理器和基本芯片的初始化; 二是提供设备的驱动接口, 实现嵌入式系统与外设的信息交换。其功能相当于 PC 类型的 BIOS。

② 嵌入式实时操作系统 (RTOS): 操作系统是对多任务嵌入式系统进行有效管理的核心部分, 可以分为基本内核和扩展内核两部分。前者提供操作系统的核心功能, 负责整个系统的任务调度、存储器分配、时钟管理、中断管理, 也可提供文件、GUI、网络等通用服务; 后者则是根据应用领域的需要, 为用户提供面向领域或面向具体行业的操作系统扩展功能。

③ 应用程序编程接口 (API): 也可称为应用程序编程中间件, 由为开发嵌入式应用程序而提供的各种编程接口库 (Lib) 或组件 (Component) 组成, 可以针对不同的应用领域、不同安全要求分别构建, 从而减轻开发人员的负担, 加快应用系统的开发进程。

④ 嵌入式应用程序: 最终运行在目标机上的应用程序。

实际在构建嵌入式系统时, 并不一定需要 RTOS 和应用程序编程接口 API, 即使需要使用, 也可以根据应用需求进行配置和裁剪。

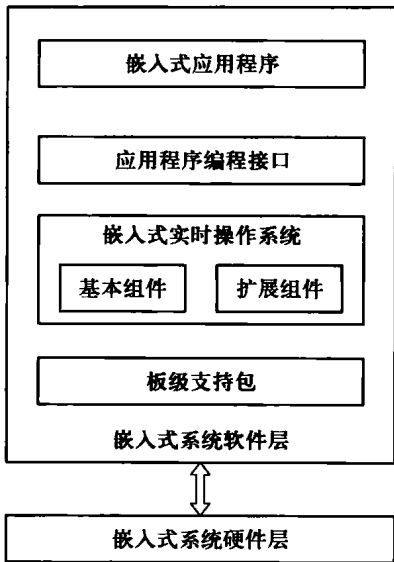


图 1.4 嵌入式系统软件组成

1.5 ARM 微处理器

ARM (Advanced RISC Machines) 公司 1991 年成立于英国剑桥, 是专门从事 RISC 技术芯片设计开发的公司, 主要出售芯片设计技术的授权, 作为知识产权供应商, ARM 公司本身不直接从事芯片生产, 靠转让设计许可由合作公司生产各具特色的芯片, 半导体生产商从 ARM 公司购买其设计的 ARM 微处理器核, 根据各自不同的应用领域, 加入适当的外围电路, 从而形成自己的 ARM 微处理器芯片进入市场。目前, 全世界有几十家大的半导体公司都使用 ARM 公司的授权, 使得 ARM 技术获得了更多的第三方工具、制造、软件的支持, 使产品更容易进入市场, 更具有竞争力。目前, ARM 微处理器几乎已经深入到工业控制、无线通信、网络应用、

消费类电子产品、成像和安全产品各个领域。

1.5.1 计算机体系结构与指令集

1. 冯·诺依曼体系结构与哈佛总线体系结构

众所周知，早期的微处理器内部大多采用冯·诺依曼体系结构，以 Intel 公司的 x86 系统微处理器为代表，如图 1.5 所示。采用冯·诺依曼结构的微处理器的程序空间和数据是合在一起的，取指令和取操作数利用同一条总线通过时分复用的方式进行。在高速运行时，不能达到同时取指令和取操作数的目的，从而形成了传输过程的瓶颈。冯·诺依曼体系结构被大多数微处理器所采用，ARM7 处理器也采用此体系结构。

随着微电子技术的发展，以 DSP 和 ARM 为应用代表的哈佛总线技术应运而生，如图 1.6 所示。在采用哈佛总线体系结构的芯片内部，程序空间和数据空间是分开的，这就允许同时取指令（来自程序空间）和取操作数（来自数据空间），从而使运算能力大大提高。

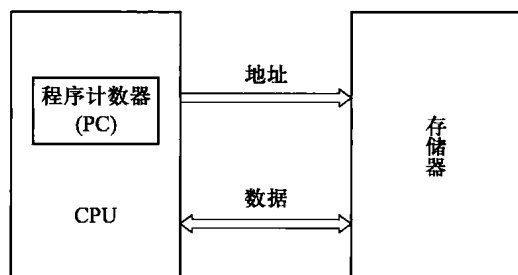


图 1.5 冯·诺依曼体系结构

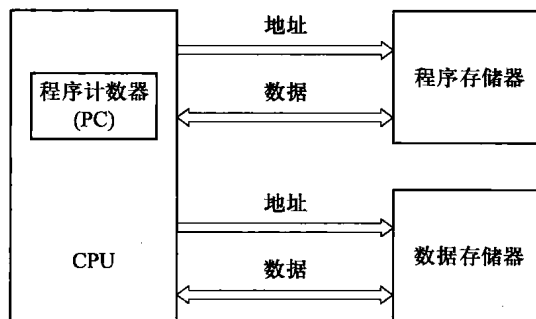


图 1.6 哈佛总线体系结构

哈佛总线体系结构主要有以下几个特点：

- ① 程序空间和数据空间分开。
- ② 提供较大的数据存储器带宽。
- ③ 适合于 DSP 数字信号处理。

目前，绝大多数的 DSP 以及 ARM9 以上系列 ARM 处理器内核都采用哈佛总线体系结构。

2. CISC 与 RISC 指令集

CISC (Complex Instruction Set Computer, 复杂指令集计算机) 和 RISC (Reduced Instruction Set Computer, 精简指令集计算机) 是当前 CPU 的两种架构，它们的区别在于 CPU 的设计理念和方式不同。

早期的计算机大都采用 CISC 指令集处理器。它的特点是：指令系统庞大，一般都有数百条指令；指令长度不固定，寻址方式复杂，增加了硬件电路的复杂程度；指令系统中绝大多数复杂指令在程序设计过程中使用频率较低，浪费严重。目前只有 Intel 及其兼容 CPU 还在使用 CISC 架构。

1979年,美国加州大学伯克利分校提出了RISC(Reduced Instruction Set Computer,精简指令集计算机)的概念。RISC指令集优先选取使用频率高的简单指令,避免复杂指令;将指令长度固定,指令格式和寻址方式种类减少;以控制逻辑为主,不用或少用微码控制等措施来达到上述目的。

ARM采用RISC结构,在简化处理器结构,减少复杂功能指令的同时,提高了处理器的速度。考虑到处理器与存储器打交道的指令执行时间远远大于在寄存器内操作的指令执行时间,RISC型处理器采用了Load/Store(加载/存储)结构,即只有Load/Store指令可以访问存储器,实现操作数的读写操作,其余指令都不允许进行存储器操作。同时,为了进一步提高指令和数据的存取速度,RISC型处理器增加了指令高速缓冲I-Cache和数据高速缓冲D-Cache及多寄存器结构,使指令的操作尽可能在寄存器之间进行。表1.1给出CISC和RISC两种指令系统的特点。

表 1.1 CISC 和 RISC 两种指令系统比较

指令集类别	CISC	RISC
指令数目	指令数量很多	较少,通常少于100条
执行时间	有些指令的执行时间很长,如需要访问存储器来获得操作数的指令执行时间通常较长	所有运算所涉及的操作数都直接从寄存器获得,指令的执行时间短,且较固定
编码长度	指令的编码长度可变,1~15 B	指令的编码长度固定,通常为4 B
寻址方式	寻址方式复杂,增加了硬件实现的复杂度	寻址方式简单
操作	可对存储器操作数和寄存器操作数进行算术和逻辑操作	只能对寄存器操作数进行算术和逻辑操作,Load/Store体系结构
编译	难以用优化编译器生成高效的目标代码程序	采用优化编译技术,可生成高效的目标代码程序

1.5.2 ARM 体系结构版本

ARM 32位体系结构目前被认为是嵌入式应用领域领先的32位嵌入式RISC微处理器结构。ARM体系结构从最初开发到现在有了很大的改进,并仍在不断完善和发展中。为了清楚地表达每个ARM应用实例所使用的指令集,按照出现的先后,ARM公司定义了7种主要的ARM指令集体系结构版本,以版本号V1~V7表示。表1.2给出ARM体系结构各版本的特点。其中V3版本之前的ARM处理器核目前已不再使用。

表 1.2 ARM 体系结构版本

版本号	寻址空间	使用该版本的 ARM 核	备注
V1	26 位	ARM1	该版架构只在原型机 ARM1 中使用,未商业化(1985年推出)
V2	26 位	ARM2、ARM3 (V2a 架构)	目前已废弃

续表

版本号	寻址空间	使用该版本的 ARM 核	备注
V3	32 位	ARM6、ARM600、ARM610 ARM7、ARM700、ARM710	增加了中止 (Abort) 和未定义两种处理器模式, 目前已废弃
V4	32 位	ARM7、ARM9、ARM9E、 Strong ARM (Intel 公司)	是目前广泛使用的 ARM 体系架构
V5	32 位	ARM10, Xscale (Intel 公司)	
V6	32 位	ARM11	2001 年发布, 2002 年首先在 ARM11 中使用
V7	32 位	Cortex-M、Cortex-R 和 Cortex-A	大大增强了面向嵌入式应用的多媒体处理功能

1.5.3 ARM 处理器核

ARM 公司开发了多个系列的 ARM 处理器核, 目前主要包括 ARM7、ARM9、ARM9E、ARM10E、ARM11、SecurCore 和 Cortex 等系列处理器产品, 以及基于 ARM 体系结构的 Intel 公司推出的 StrongARM 和 XScale 处理器。其中, 每一系列又根据各自包含的功能模块而分成多个变种。对于支持同样 ARM 体系版本的处理器, 其应用程序软件是兼容的。表 1.3 给出部分 ARM 处理器核的性能参数。

表 1.3 ARM 处理器核性能参数

IP 核系列	ARM7	ARM9	ARM10	ARM11
流水线	3	5	6	8
典型频率/MHz	80	150	260	335
功耗/(mW/MHz)	0.06	0.19	0.5	0.4
速度/(MIPS/MHz)	0.97	1.1	1.3	1.2
架构	冯·诺依曼	哈佛	哈佛	哈佛

注: “ARM 核”并不是芯片, ARM 核与其他部件如 RAM、ROM、片内外设组合在一起才能构成现实的芯片。

1. ARM7 系列

ARM7TMDI 是目前使用最广泛的 32 位嵌入式 RISC 处理器, 属低端 ARM 处理器核。指令系统与 ARM9 系列、ARM9E 系列和 ARM10E 系列兼容, 便于用户产品的升级换代。该系列有 4 种类型: ARM7TDMI、ARM7TDMI-S、ARM720T 和 ARM7EJ。

ARM7 系列广泛应用于多媒体和嵌入式设备, 包括 Internet 设备、网络和调制解调器设备以及移动电话、PDA 等无线设备。