



普通高等教育“十二五”规划教材

◎ 电子信息科学与工程类专业 规划教材

嵌入式系统原理 与应用设计

◎ 王光学 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

普通高等教育“十二五”规划教材
电子信息科学与工程类专业规划教材

嵌入式系统 原理与应用设计

王光学 编著

电子工业出版社
Publishing House of Electronics Industry
北京·BEIJING

内 容 简 介

本书主要内容包括嵌入式系统基本概念、ARM 编程模型、ARM 指令集、ARM 程序设计基础、嵌入式系统硬件与软件结构、嵌入式系统硬件与底层驱动程序设计及嵌入式系统应用程序设计。本书通过一典型设计实例阐述嵌入式系统组成原理与设计方法。采取“自顶向下+模块化”方式讲授设计实例，首先给出实例架构，然后划分为模块，再设计硬件与底层驱动程序，最后设计应用程序。本书集作者多年教学探索所成，重点突出、语言简练，内容全面。本书配有电子课件等教学资源。

本书适合高校计算机科学与技术、电子信息科学与技术、电子信息工程、软件工程与自动化等专业嵌入式系统及其相关课程的教材，也适合具备上述专业背景的工程技术人员自学或参考。

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

嵌入式系统原理与应用设计 / 王光学编著. —北京: 电子工业出版社, 2013.1

电子信息科学与工程类专业规划教材

ISBN 978-7-121-19130-5

I. ①嵌… II. ①王… III. ①微型计算机—系统设计—高等学校—教材 IV. ①TP360.21

中国版本图书馆 CIP 数据核字(2012)第 286215 号

策划编辑: 索蓉霞 刁伟兴

责任编辑: 史鹏举

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: 21.5 字数: 621 千字

印 次: 2013 年 1 月第 1 次印刷

定 价: 39.00 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。

前 言

嵌入式系统是研究手机、数码相机、微波炉、医院 B 超、小汽车及工业自动化等设备中专用计算机系统的一门学科。它是随着电子集成电路技术的发展和 32 位嵌入式处理器的广泛使用而带来的产物。嵌入式系统(Embedded System)在国外出现的历史要长些,但在国内,直到 2002 年左右这一名词才开始大量出现在媒体上,此后迅速成为 IT 技术的一个研究热点,国内很多高校相继开设了相关课程或专业方向。笔者于 2007 年开始组织和编写本书,历时 5 年完成,本书力图体现如下特色。

(1) 突出重点、难点,抓住“抓手”。

例如异常中断,既是重点,又是难点,本书不但在内容上不惜笔墨,而且在讲授方式上也煞费苦心,设法将它以一种最易接受的方式呈现给读者;又如嵌入式系统硬件设计,本书抓住总线结构及设备控制器这个“抓手”,使硬件设计变得简单容易;再如 $\mu\text{C}/\text{OS-II}$ 功能函数,笔者着力于源码的分析,通过源码理解其功能及工作原理。

(2) 通过典型实例阐述嵌入式系统原理及设计方法。

嵌入式系统涉及面广、复杂、抽象,若没有实例物化、深化、固化,学生不容易掌握。故在讲授完 ARM 架构与 $\mu\text{C}/\text{OS-II}$ 操作系统后,本书虚构了一个应用需求,给出了能完成这一需求的嵌入式系统组成架构,随后各章的讲授都围绕这一架构的实现来展开。

(3) 采取“自顶向下+模块化”方式讲授设计实例。

嵌入式系统设计实例能否被学生掌握是决定本门课程成败的关键,但实例本身太复杂,初学者不易掌握。常听学生如此抱怨:“前面汇编感觉还不错,但到后面的设计实例就晕了。”故从承担这门课起,笔者一直在思考如何讲授设计实例,并做了不少尝试,逐渐摸索出了讲授实例最有效的方式:自顶向下+模块化。

所谓“自顶向下”就是先讲授嵌入式系统架构,再讲授实现细节;“模块化”就是将复杂的嵌入式系统划分为模块,一个模块一个模块地进行设计介绍。“自顶”使学生登高望远,一览众山小,以掌握嵌入式系统整体结构;“向下”使学生深入其中,掌握嵌入式系统技术细节。“模块化”又将复杂的嵌入式系统做了简化,便于学习掌握。故自从采取“自顶向下+模块化”方式讲授设计实例后,讲课感觉轻松多了,也听不到先前的抱怨了。所以,本书在结构设计上有一定程度的创新,即在紧随 ARM 架构及 $\mu\text{C}/\text{OS-II}$ 操作系统介绍后,增加一章专门分析研究嵌入式系统架构,接着进行模块划分,将复杂的嵌入式系统设计实例划分为一个个模块,逐一讲授每个模块的硬件与底层驱动设计,最后一章则利用前面提出的架构与设计的模块,给出了完成相同事务处理的嵌入式系统应用程序的无核(单任务)及有核(多任务)实现。

目前,国内开设嵌入式系统及其相关课程的专业通常有计算机科学与技术、电子信息科学与技术、电子信息工程、软件工程与自动化等。但无论是在哪个专业开设,不外乎两种情况:一种情况是学分少,作为导论课开设;另一种情况是学分较多,作为专业方向课开设。本书作为嵌入式系统系列课程中最基础的入门课,两种情况均适宜。对于第一种情况,由于开设的是嵌入式系统导论课,建议讲授完本书的全部内容,但第 7 章可以删掉一些模块。对于第二种情况,由于专业方向课程均会由几门课组成,建议重点讲授本书除第 5 章以外的所有章节,第 5 章所涉内容可以另设一门课来讲授。

为方便教学，本书配有电子课件，任课教师可以登录华信教育资源网(www.hxedu.com.cn)免费注册下载。

本书得以顺利出版，感谢家人的支持，感谢学校特色教材项目的资助，感谢电子工业出版社索蓉霞编辑的有益建议和史鹏举编辑付出的努力。

在多年的教学摸索中，笔者从教学实际需求的角度观察、思考，并试图尝试新的表述方式展现嵌入式系统的全貌，但由于水平有限，书中难免有不妥之处，恳请指正。

作者

目 录

| | |
|-------------------------------------|--|
| 第 1 章 嵌入式系统概述1 | |
| 1.1 嵌入式系统的基本概念.....1 | |
| 1.1.1 嵌入式系统定义.....1 | |
| 1.1.2 嵌入式系统的特点.....1 | |
| 1.2 嵌入式系统发展.....2 | |
| 1.3 嵌入式系统的应用.....3 | |
| 1.4 嵌入式系统的组成.....4 | |
| 1.4.1 嵌入式处理器.....5 | |
| 1.4.2 嵌入式操作系统.....9 | |
| 1.4.3 嵌入式应用程序.....10 | |
| 1.5 本章小结.....11 | |
| 习题与思考题.....12 | |
| 第 2 章 ARM 编程模型13 | |
| 2.1 ARM 发展历程及其技术特征.....13 | |
| 2.1.1 ARM 发展历程.....13 | |
| 2.1.2 ARM 技术特征.....13 | |
| 2.2 ARM 体系结构版本概述.....14 | |
| 2.2.1 ARM 体系结构版本.....14 | |
| 2.2.2 ARM 体系结构版本的变种.....16 | |
| 2.3 ARM 核概述.....18 | |
| 2.3.1 ARM 核命名规则.....18 | |
| 2.3.2 ARM7 系列微处理器核.....19 | |
| 2.3.3 ARM9.....19 | |
| 2.3.4 ARM9E.....19 | |
| 2.3.5 ARM10E.....20 | |
| 2.3.6 SecurCore.....20 | |
| 2.3.7 StrongARM.....20 | |
| 2.3.8 XScale.....20 | |
| 2.3.9 ARM11 系列核.....21 | |
| 2.3.10 ARM Cortex 系列核.....21 | |
| 2.4 ARM 微处理器核的工作状态.....21 | |
| 2.5 ARM 处理器核的工作模式.....22 | |
| 2.6 ARM 核的内部寄存器.....22 | |
| 2.6.1 ARM 状态下的寄存器.....22 | |
| 2.6.2 Thumb 状态下的寄存器.....25 | |
| 2.7 ARM 核的异常中断.....26 | |
| 2.7.1 ARM 核异常中断概述.....26 | |
| 2.7.2 ARM 核异常中断响应过程.....27 | |
| 2.7.3 ARM 核异常中断的返回.....28 | |
| 2.7.4 ARM 核异常中断处理程序结构.....29 | |
| 2.7.5 ARM 核异常中断向量表.....30 | |
| 2.7.6 ARM 核异常中断的优先级.....30 | |
| 2.8 ARM 核流水线.....31 | |
| 2.8.1 三级流水线.....31 | |
| 2.8.2 五级流水线.....34 | |
| 2.9 ARM 协处理器.....34 | |
| 2.10 ARM AMBA 接口.....35 | |
| 2.11 ARM 存储器结构.....36 | |
| 2.11.1 ARM 存储器的数据类型和 存储格式.....36 | |
| 2.11.2 ARM 存储体系.....36 | |
| 2.12 本章小结.....37 | |
| 习题与思考题.....38 | |
| 第 3 章 ARM 指令系统39 | |
| 3.1 ARM 指令系统概述.....39 | |
| 3.1.1 ARM 指令的基本汇编格式.....39 | |
| 3.1.2 ARM 指令的机器码格式.....39 | |
| 3.1.3 ARM 指令可选后缀.....40 | |
| 3.2 ARM 指令的寻址方式.....43 | |
| 3.2.1 立即寻址.....43 | |
| 3.2.2 寄存器寻址.....43 | |
| 3.2.3 寄存器间接寻址.....45 | |
| 3.2.4 基址变址寻址.....45 | |
| 3.2.5 堆栈寻址.....46 | |
| 3.2.6 多寄存器寻址.....47 | |
| 3.2.7 相对寻址.....48 | |
| 3.3 ARM 指令集.....48 | |
| 3.3.1 数据传送指令.....48 | |

| | | | | | |
|--------------|---------------------------------------|------------|--------------|---------------------------------------|------------|
| 3.3.2 | 数据处理指令 | 54 | 5.6.1 | $\mu\text{C}/\text{OS-II}$ 时钟节拍中断服务程序 | 141 |
| 3.3.3 | 程序状态寄存器访问指令 MRS 和 MSR | 63 | 5.6.2 | $\mu\text{C}/\text{OS-II}$ 的时间管理 | 142 |
| 3.3.4 | 跳转指令 | 64 | 5.7 | $\mu\text{C}/\text{OS-II}$ 的同步与通信 | 143 |
| 3.3.5 | 异常产生指令 | 66 | 5.7.1 | 同步与通信的基本概念 | 143 |
| 3.3.6 | 协处理器指令 | 69 | 5.7.2 | 事件控制块 ECB | 143 |
| 3.4 | Thumb 指令集 | 71 | 5.7.3 | 信号量 | 145 |
| 3.4.1 | Thumb 指令集概述 | 71 | 5.7.4 | 消息邮箱 | 148 |
| 3.4.2 | Thumb 指令分类介绍 | 71 | 5.7.5 | 消息队列 | 149 |
| 3.5 | 本章小结 | 80 | 5.8 | $\mu\text{C}/\text{OS-II}$ 的内存管理 | 149 |
| | 习题与思考题 | 80 | 5.9 | $\mu\text{C}/\text{OS-II}$ 的移植 | 151 |
| 第 4 章 | ARM 汇编程序设计 | 81 | 5.10 | 本章小结 | 157 |
| 4.1 | ARM 伪操作与伪指令 | 81 | | 习题与思考题 | 158 |
| 4.1.1 | GNU 编译环境下的伪操作 | 81 | 第 6 章 | 嵌入式系统架构 | 160 |
| 4.1.2 | ADS 环境下的伪操作 | 86 | 6.1 | S3C44B0X 处理器 | 160 |
| 4.1.3 | 两种编译环境下的常用伪操作 汇总 | 96 | 6.1.1 | 功能模块与总线结构 | 160 |
| 4.1.4 | 伪指令 | 96 | 6.1.2 | 设备控制器 | 161 |
| 4.2 | ARM 汇编程序设计 | 98 | 6.1.3 | 设备控制器设备侧接口 | 162 |
| 4.2.1 | ARM 汇编语言中的文件格式 | 98 | 6.1.4 | 设备控制器寄存器 | 166 |
| 4.2.2 | ARM 汇编语言语句格式 | 99 | 6.2 | 嵌入式系统硬件结构 | 170 |
| 4.2.3 | 汇编程序设计示例 | 103 | 6.2.1 | 单总线结构 | 170 |
| 4.2.4 | C 语言与汇编语言混合编程 | 105 | 6.2.2 | 多总线结构 | 171 |
| 4.3 | 本章小结 | 113 | 6.3 | 嵌入式系统软件结构 | 173 |
| | 习题与思考题 | 114 | 6.3.1 | 单任务结构 | 174 |
| 第 5 章 | 嵌入式操作系统 | 115 | 6.3.2 | 多任务结构 | 177 |
| 5.1 | 嵌入式操作系统 $\mu\text{C}/\text{OS-II}$ 简介 | 115 | 6.4 | 本章小结 | 178 |
| 5.2 | $\mu\text{C}/\text{OS-II}$ 的任务管理 | 116 | | 习题与思考题 | 178 |
| 5.2.1 | $\mu\text{C}/\text{OS-II}$ 任务概述 | 116 | 第 7 章 | 嵌入式系统硬件与底层驱动 程序设计 | 179 |
| 5.2.2 | 任务调度 | 118 | 7.1 | 功能模块划分 | 179 |
| 5.2.3 | 任务创建 | 123 | 7.2 | 电源电路模块 | 180 |
| 5.2.4 | 任务的挂起和恢复 | 128 | 7.3 | 复位电路模块 | 181 |
| 5.2.5 | 任务的删除 | 129 | 7.4 | JTAG 接口模块 | 182 |
| 5.3 | $\mu\text{C}/\text{OS-II}$ 的初始化 | 129 | 7.5 | 时钟与电源管理模块 | 183 |
| 5.4 | $\mu\text{C}/\text{OS-II}$ 的启动 | 137 | 7.5.1 | 时钟产生电路 | 183 |
| 5.5 | $\mu\text{C}/\text{OS-II}$ 的中断 | 138 | 7.5.2 | 电源管理 | 186 |
| 5.5.1 | $\mu\text{C}/\text{OS-II}$ 的中断过程 | 139 | 7.5.3 | 控制寄存器 | 188 |
| 5.5.2 | 中断级任务切换 | 140 | 7.5.4 | 驱动程序 | 189 |
| 5.6 | $\mu\text{C}/\text{OS-II}$ 的时钟 | 141 | 7.5.5 | 时钟电路电磁兼容设计 | 189 |
| | | | 7.6 | 存储器模块 | 190 |

| | | | | | |
|--------|---------------------|-----|--------|-------------------|-----|
| 7.6.1 | RAM | 190 | 7.14 | LCD 模块 | 255 |
| 7.6.2 | ROM | 192 | 7.14.1 | LCD 工作原理 | 255 |
| 7.6.3 | 存储器芯片 | 194 | 7.14.2 | 液晶显示器驱动电路 | 256 |
| 7.6.4 | 存储空间规划 | 195 | 7.14.3 | S3C44B0X LCD 控制器 | 256 |
| 7.6.5 | 存储器电路设计 | 197 | 7.14.4 | 驱动程序 | 264 |
| 7.6.6 | SDRAM 驱动程序设计 | 198 | 7.15 | A/D 转换与触摸屏模块 | 265 |
| 7.6.7 | Flash 驱动程序 | 205 | 7.15.1 | S3C44B0X A/D 转换器 | 266 |
| 7.7 | RS-232 接口模块 | 207 | 7.15.2 | 触摸屏工作原理 | 267 |
| 7.7.1 | RS-232 接口介绍 | 207 | 7.15.3 | 触摸屏检测电路 | 268 |
| 7.7.2 | UART 接口介绍 | 208 | 7.15.4 | 驱动程序 | 269 |
| 7.7.3 | RS-232 接口电路设计 | 211 | 7.16 | 以太网接口模块 | 271 |
| 7.7.4 | RS-232 接口驱动程序设计 | 211 | 7.16.1 | 以太网 MAC 与 PHY | 271 |
| 7.8 | LED 与 GPIO 模块 | 217 | 7.16.2 | 以太网中的 TCP/IP 协议 | 273 |
| 7.8.1 | GPIO 控制寄存器 | 217 | 7.16.3 | 以太网接口电路设计 | 275 |
| 7.8.2 | GPIO 初始化 | 220 | 7.16.4 | 以太网接口驱动程序设计 | 276 |
| 7.8.3 | LED 设计 | 221 | 7.17 | USB 接口模块 | 289 |
| 7.9 | 中断控制器模块 | 222 | 7.18 | IIS 接口模块 | 293 |
| 7.9.1 | 外部中断接口分配 | 223 | 7.18.1 | IIS 总线 | 293 |
| 7.9.2 | 中断控制器的功能 | 223 | 7.18.2 | S3C44B0X IIS 总线接口 | 295 |
| 7.9.3 | IRQ 中断模式 | 231 | 7.18.3 | IIS 总线接口电路 | 297 |
| 7.9.4 | 中断处理程序 | 233 | 7.18.4 | 驱动程序 | 300 |
| 7.10 | 定时器模块 | 234 | 7.19 | 本章小结 | 308 |
| 7.10.1 | PWM 定时器 | 234 | 习题与思考题 | 311 | |
| 7.10.2 | 看门狗定时器 | 237 | 第 8 章 | 嵌入式系统应用程序设计 | 313 |
| 7.10.3 | 实时时钟 | 239 | 8.1 | 可执行文件的产生 | 313 |
| 7.11 | 键盘模块 | 240 | 8.2 | 链接与装入程序 | 314 |
| 7.12 | 8 段数码管模块 | 245 | 8.3 | 启动程序 Bootloader | 318 |
| 7.13 | EEPROM 与 IIC 总线接口模块 | 247 | 8.4 | 单任务应用程序 | 327 |
| 7.13.1 | IIC 总线接口协议 | 247 | 8.5 | 多任务应用程序 | 329 |
| 7.13.2 | AT24LC04 | 249 | 8.6 | 本章小结 | 332 |
| 7.13.3 | IIC 总线控制器 | 250 | 习题与思考题 | 332 | |
| 7.13.4 | IIC 接口电路 | 250 | 参考文献 | 333 | |
| 7.13.5 | IIC 总线控制寄存器 | 251 | | | |
| 7.13.6 | 驱动程序设计 | 252 | | | |

第1章 嵌入式系统概述

什么是嵌入式系统？有何特点？与传统的单片机系统以及 PC 系统有何不同？由哪些部分组成？如何开发？本章试图对这些问题做出回答。第 1 节阐述了嵌入式系统的基本概念，包括嵌入式系统定义、特征、与 PC 系统及单片机系统的区别；第 2 节介绍了嵌入式系统的发展；第 3 节介绍了嵌入式系统的应用；第 4 节介绍了嵌入式系统的组成，主要介绍其中的嵌入式处理器、嵌入式操作系统及嵌入式应用软件。

1.1 嵌入式系统的基本概念

1.1.1 嵌入式系统定义

嵌入式系统的定义要从计算机工业的分类说起。传统上，按照计算机的体系结构、运算速度、结构规模分为大型计算机、中型机、小型机和微型计算机，并以此来组织学科和产业分工，这种分类沿袭了约 40 年。但很明显，这种分类已不适合现代计算机工业了，现代微型计算机的速度、结构复杂度都直抵原来的大、中型机，于是现代分类按计算机的嵌入式应用和非嵌入式应用将其分为通用计算机和嵌入式计算机。通用计算机即通常所说的个人计算机(PC)，嵌入式计算机即指嵌入式系统。因此，我们可以给嵌入式系统下这样一个定义：嵌入式系统是嵌入到对象体系中的专用计算机系统。如微波炉、空调、小汽车、数码相机与手机等产品中都需要计算机系统来进行控制与管理，这些产品中的计算机系统就是嵌入式系统。但习惯上，也称这些产品为嵌入式系统。

IEEE(国际电气和电子工程师协会)给嵌入式系统下了这样一个定义：嵌入式系统是“用于控制、监视或者辅助操作机器和设备的装置”。

可以看出此定义是从应用上考虑的，嵌入式系统是软件和硬件的综合体，还可以涵盖机电等附属装置。

综上所述，嵌入式系统有狭义与广义两类定义。狭义上看，嵌入式系统是指嵌入微波炉、空调、小汽车及数码相机等产品中用于控制与管理的专用计算机系统；广义上看，微波炉、空调、小汽车及数码相机等内部包含有专用计算机的产品都属嵌入式系统。

嵌入式系统的另一个较通用的定义是：嵌入式系统是以应用为中心，以计算机技术为基础，软件硬件可裁剪，适应应用系统对功能、可靠性、成本、体积、功耗等严格要求的专用计算机系统。

1.1.2 嵌入式系统的特点

PC 与单片机系统是我们较熟悉的，下面从嵌入式系统与 PC、单片机系统的区别来介绍嵌入式系统的特点。

1. 嵌入式系统与 PC 的区别

嵌入式系统与个人计算机系统(PC)的区别主要体现在以下方面：

(1) 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，力争在较少的资源上实现更高的性能；PC 的软/硬件都很庞大、臃肿。

(2) 嵌入式系统的目标代码通常是固化在非易失性存储器 (ROM, EPROM, EEPROM, Flash) 芯片中的；PC 存放在硬盘中。

(3) 嵌入式系统使用的操作系统一般是实时操作系统 (RTOS)，系统有实时性约束；PC 无此限制。

(4) 嵌入式系统需要专用开发工具和方法进行设计——交叉开发；PC 直接开发。

(5) 通用 PC 软、硬件技术高度垄断；嵌入式系统技术不容易被垄断。

2. 嵌入式系统与单片机系统的区别

嵌入式系统与单片机系统的区别主要体现在以下方面：

(1) 嵌入式系统通常指基于 32 位微处理器设计的系统 (往往带操作系统)；单片机系统指基于 4 位、8 位与 16 位微处理器设计 (不使用操作系统) 的系统。

(2) 嵌入式系统设计的核心是软件设计，约占 70% 的工作量，硬件只占 30%；单片机系统软/硬件设计工作所占比例基本相同，约为 1:1。

(3) 嵌入式软件职位与硬件职位的需求比为 7:3，单片机系统约为 1:1，甚至软/硬件职位不分。

但根据嵌入式系统的定义，单片机系统显然也属于嵌入式系统，满足嵌入式系统的定义。为将单片机系统与通常所指的嵌入式系统区别开来，我们可将单片机系统看做低端嵌入式系统，而将以 32 位嵌入式处理器为代表的嵌入式系统看做中、高端嵌入式系统。

3. 嵌入式系统的特征

嵌入式系统的特征主要体现在以下方面：

(1) 嵌入式系统是一个专用计算机系统。

(2) 嵌入式系统软、硬件根据需要进行定制，一般有功耗低、体积小、集成度高、成本低、可靠性高、实时性强等要求。具体要求随应用环境而异。

(3) 嵌入式系统软件采取交叉方式进行开发。

1.2 嵌入式系统发展

嵌入式系统的发展历程，大致经历了以下 4 个阶段。

(1) 无操作系统阶段

单片机是最早应用的嵌入式系统，单片机作为各类工业控制和飞机、导弹等武器装备中的微控制器，用来执行一些单线程的程序，完成监测、伺服和设备指示等多种功能，一般没有操作系统的支持，程序设计采用汇编语言。由单片机构成的这种嵌入式系统使用简便，价格低廉，在工业控制领域中得到了非常广泛的应用。

(2) 简单操作系统阶段

20 世纪 80 年代，出现了大量具有高可靠性、低功耗的嵌入式 CPU (如 PowerPC 等)，芯片上集成有 I/O 接口、串行接口及 RAM、ROM 等部件。一些简单的嵌入式操作系统开始出现并得到迅速发展，程序设计人员也开始基于一些简单的“操作系统”开发嵌入式应用软件。此时的嵌入式操作系统虽然还比较简单，但已经初步具有了一定的兼容性和扩展性，内核精巧且效率高，大大缩短了开发周期，提高了开发效率。

(3) 实时操作系统阶段

20 世纪 90 年代，面对分布式控制、柔性制造、数字化通信和信息家电等巨大市场的需求，嵌入

式系统飞速发展。随着硬件实时性要求的提高,嵌入式系统的软件规模也不断扩大,实时多任务操作系统(Real-Time Operation System, RTOS)逐渐形成,系统能够运行在各种不同类型的微处理器上,具备了文件和目录管理、设备管理、多任务、网络、图形用户界面等功能,并提供了大量的应用程序接口(Application Programming Interface, API),从而使应用软件的开发变得更加简单。

(4) 面向 Internet 阶段

进入 21 世纪,Internet 技术与信息家电、工业控制技术等的结合日益紧密,嵌入式技术与 Internet 技术的结合正在推动着嵌入式系统的飞速发展。面对嵌入式技术与 Internet 技术的结合,嵌入式系统技术和应用在飞速发展,主要体现在以下方面:

① 新的微处理器不断出现,主频越来越高,从单核向多核发展,功耗和成本不断下降。提供更加友好的多媒体人机交互界面。

② Linux、Windows CE、Palm OS 与 Android 等嵌入式操作系统迅速发展。嵌入式操作系统自身结构的设计更便于移植,具有源代码开放、系统内核小、执行效率高、网络结构完整等特点,能够在短时间内支持更多的微处理器。计算机的新技术、新观念开始逐步移植到嵌入式系统中,嵌入式软件平台得到进一步完善。

③ 嵌入式系统的开发成了一项系统工程,开发厂商不仅要提供嵌入式软、硬件系统本身,同时还要提供强大的硬件开发工具和软件支持包。

1.3 嵌入式系统的应用

嵌入式系统的应用非常广泛,在军事国防、消费电子、工业控制、商业领域、医疗设备、交通管理、环境监测及信息家电中都得到了应用。

在军事国防方面,嵌入式系统应用于各种武器控制(火炮控制、导弹控制、智能炸弹制导引爆装置)及坦克、舰艇、轰炸机等陆海空各种军用电子装备、雷达、电子对抗、军事通信装备、野战指挥作战用各种专用设备中。

在消费电子方面,嵌入式系统应用于各种信息家电产品中,如数字电视机、机顶盒、数码相机、VCD、DVD、音响设备、可视电话、家庭网络设备、洗衣机、电冰箱、智能玩具、手机等。

在工业控制方面,嵌入式系统应用于各种智能测量仪表、数控装置、可编程控制器、分布式控制系统、现场总线仪表及控制系统、工业机器人、机电一体化机械设备及汽车电子设备中。

在商用领域,嵌入式系统应用于各类收款机、POS 系统、电子秤、条形码阅读机、商用终端、银行点钞机、IC 卡输入设备、取款机、自动柜员机、自动服务终端、防盗系统及其他各种银行专业外围设备中。

在办公领域,嵌入式系统应用于复印机、打印机、传真机、扫描仪、激光照排系统、安全监控设备、个人数字助理(PDA)、变频空调设备、通信终端、程控交换机、网络设备、录音录像、电视会议设备、数字音频广播等各种办公系统中。

在医疗设备领域,嵌入式系统应用于 X 光机、超声诊断仪、计算机断层成像系统、心脏起搏器、监护仪、辅助诊断系统、专家系统等各种医疗电子仪器中。

在交通管理、环境监测方面,嵌入式系统应用于车辆导航、流量控制、信息监测、车载 GPS 设备、水文资料实时监测、防洪体系及水土质量监测、堤坝安全、地震监测网、实时气象信息网、水源和空气污染监测等各种系统中。

1.4 嵌入式系统的组成

嵌入式系统是一个专用计算机系统，由硬件与软件组成。如图 1.1 所示，软件可分为三层，即驱动层、OS 层与应用层。硬件可分为处理器核、片上功能模块与外部设备三部分。

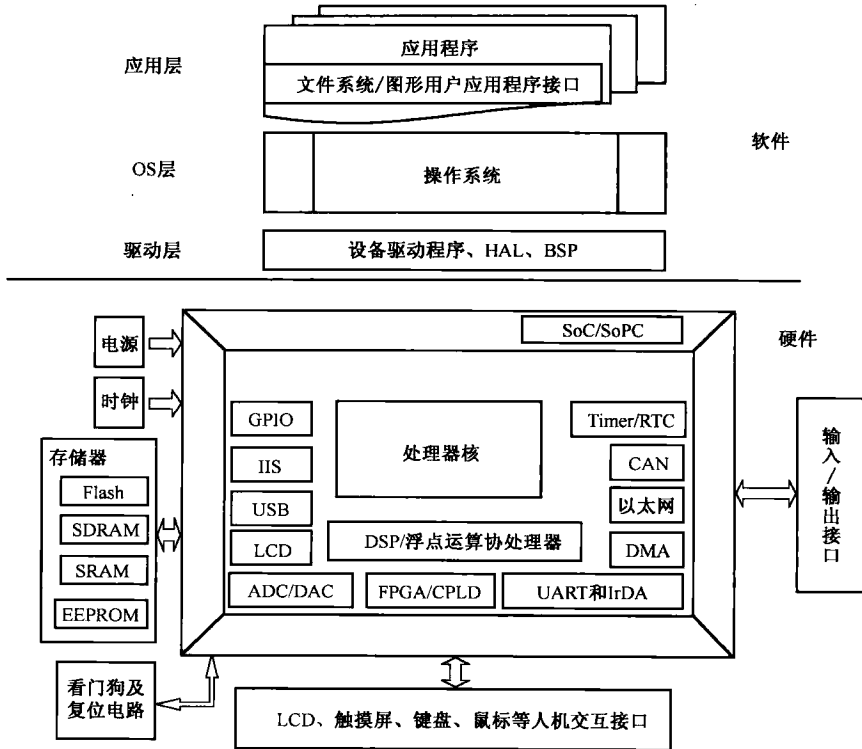


图 1.1 典型的嵌入式系统组成

驱动层指设备驱动程序、硬件抽象层 (HAL) 或板级支援包 (BSP)，可以包括在操作系统中，作为操作系统内核的组成部分。在驱动层中还应包括 Bootloader，是嵌入式系统上电复位后执行的第一个程序——开机程序，负责初始化硬件、装载代码到 RAM 等工作。PC 中类似工作由 BIOS (Basic Input /Output System) 来做，嵌入式系统中无通用 BIOS，用户必须自己编写 Bootloader 程序。嵌入式系统通常有两种启动方式，一种是直接从 Flash 启动，另一种是将压缩的内存映像文件从 Flash 中复制、解压到 RAM，再从 RAM 启动。Bootloader 完成基本软、硬件初始化后，若有操作系统，则将控制权转交操作系统；若没有操作系统，则直接执行应用程序或等待用户命令。

操作系统 (OS) 是嵌入式系统中的重要软件，管理嵌入式系统中的资源，向上向用户提供使用资源的接口，向下对硬件进行操作控制。但并非所有的嵌入式系统都需要操作系统。嵌入式操作系统种类很多，后面专辟一节介绍。

应用程序是向用户提供服务的程序，从功能上看，其与 PC 应用程序无多大区别，不同之处是其开发过程，通常采取交叉开发方式进行开发。

处理器核是嵌入式系统的大脑，种类很多，有 ARM、MIPS、PowerPC、SPARC、MCS 等，后面将专辟一节进行介绍。

片上功能模块是指 UART、IIS、ADC/DAC、LCD、DMA 与 Timer 等与处理器核集成在一起的独立功能模块或设备控制器。独立功能模块可单独提供某种使用功能,如 Timer 可定时、ADC/DAC 可进行模数/数模转换。设备控制器可用于挂接外部设备,如 LCD 控制器可挂 LCD, UART 控制器可挂接 RS-232 接口。

外部设备指输入/输出接口,人机交互接口与通信接口及电源、复位电路、存储器等处理器片外的设备。其中,输入/输出接口就是通常所说的 I/O 口,提供嵌入式处理器与外设之间连接所需的控制信号及数据通道,是嵌入式系统中最庞大的部分。电源电路提供嵌入式系统所需电源,一般为 $\pm 3/5V$ DC。时钟电路提供嵌入式系统所需时钟信号,可由一外部时钟源或一晶振电路产生。复位电路提供嵌入式系统复位信号。人机交互接口有 LCD、触摸屏、键盘与鼠标等。存储器种类也很多,通常有如下几类:

SRAM: 静态随机存储器,速度高,体积大,成本高,无需刷新;

DRAM: 动态随机存储器,速度低,体积小,成本低,需刷新;

SDRAM: 同步动态随机存储器,与 DRAM 同属一类存储器,但速度比 DRAM 快;

ROM: 只读存储器;

Flash: 只读存储器,也称闪存;

EEPROM: 电可擦除的只读存储器。

1.4.1 嵌入式处理器

嵌入式处理器通常分为嵌入式微控制器、嵌入式 DSP、嵌入式微处理器与嵌入式片上系统 4 类,下面分别进行介绍。

1. 嵌入式微控制器 MCU (Micro Control Unit)

嵌入式微控制器是主要用于控制领域的嵌入式处理器。在一块芯片内部集成了 ROM、RAM、总线、总线控制逻辑、定时/计数器、看门狗、I/O、串行口、脉宽调制输出、A/D、D/A 等各种适宜控制用的功能模块,因此称为微控制器或单片机。典型代表是 MCS-51 系列及 PIC 系列的 4 位、8 位及 16 单片机。

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,特别适合工业控制。

2. 嵌入式 DSP (Digital Signal Processor)

DSP 是专门用于数字信号处理的处理器。数字信号处理是指数字滤波、FFT、DCT、小波变换、谱分析及音视频编码解码等处理,这些处理通常涉及大量数据的传输、乘法及乘累加等操作。因此,为便于数字信号处理,DSP 在系统结构和指令算法方面进行了特殊设计,如采用便于大量数据传输的哈佛体系结构及硬件乘法器和硬件乘累加器设计,以加快数据的传输、乘法及乘累加操作的速度。

哈佛体系结构如图 1.2 所示,它与传统的冯·诺依曼体系结构(见图 1.3)不同,程序存储器与指令存储器是分开的,有两套地址总线与数据总线,在取指令的同时也可取数据,克服了冯·诺依曼体系结构取指令时不能取数据带来的数据传输瓶颈。

DSP 芯片厂商主要有 TI(德州仪器)、Motorloa、Lucent 等。

TI 占据 DSP 市场 80%以上份额,主要产品有 2000、5000 与 6000 三个系列,其中 2000 系列主要用于高速控制领域,5000 系列主要用于音频处理,6000 系列主要用于视频处理。

3. 嵌入式微处理器 MPU

MPU (Micro Processor Unit) 是由通用计算机中的 CPU 演变而来的,主要类型有 Am186/88、386EX、SC-400、PowerPC、68000、MIPS、SPARC 与 ARM 等系列。

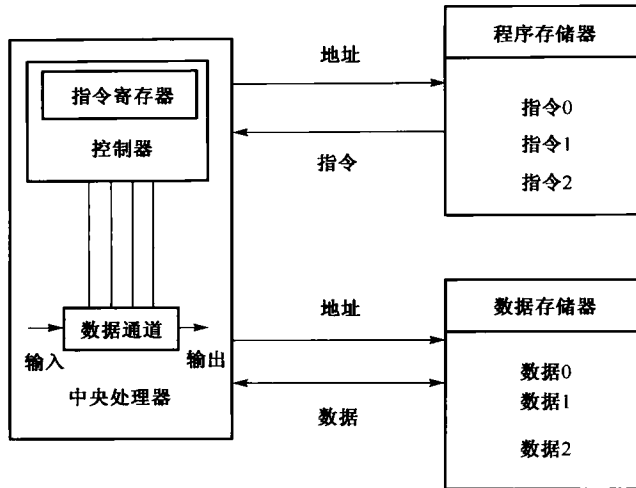


图 1.2 哈佛体系结构

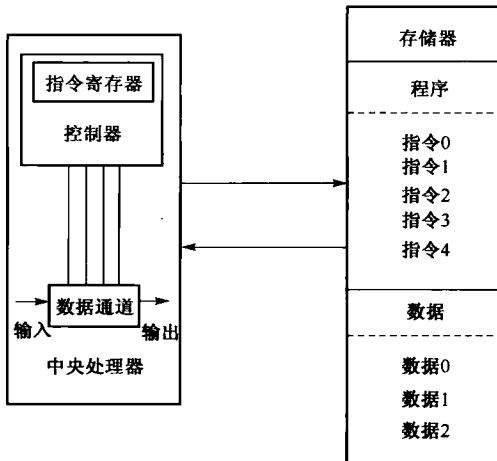


图 1.3 冯·诺依曼体系结构模型

(1) PowerPC

PowerPC 架构的处理器最初是由 IBM、Motorola 和 Apple 三家公司联合研制的产品，1994 年第一个 PowerPC 处理器 PowerPC 601 问世，到现在已有几十种 PowerPC 处理器投放市场，其主频范围从 32 MHz 到 1 GHz 不等。Apple 后来放弃了 PowerPC，但 IBM 与 Motorola 仍还在继续研发和生产 PowerPC 处理器。

PowerPC 处理器品种很多，既有通用的处理器，又有嵌入式控制器和内核，应用范围非常广泛，从高端的工作站、服务器到桌面计算机系统，从消费类电子产品到大型通信设备，无所不包。

PowerPC 主要型号是 PowerPC 750，它于 1997 年研制成功，最高的工作频率可以达到 500 MHz，

采用先进的铜线技术。该处理器有许多品种，以便

适合各种不同的系统。包括 IBM 小型机、苹果计算机和其他系统。嵌入式用的为 PowerPC 405(主频最高为 266 MHz)和 PowerPC 440(主频最高为 550 MHz)，其内核可以用于各种 SoC 设计上，在电信、金融和其他许多行业具有广泛的应用。

(2) MIPS

MIPS (Microprocessor without Interlocked Piped Stages)，为无内部互锁流水级的微处理器。

MIPS 微处理器最早是在 20 世纪 80 年代初期由斯坦福大学 Hennessy 教授领导的研究小组研制出来的。1984 年，MIPS 计算机公司成立，1992 年，SGI 收购了 MIPS 计算机公司，1998 年，MIPS 脱离 SGI，成为 MIPS 技术公司。MIPS 技术公司是一家设计与制造高性能嵌入式 32 位和 64 位处理器的厂商，在 RISIC 处理器方面占有重要地位。

MIPS 系列微处理器是目前仅次于 ARM 的用得最多的处理器之一(1999 年以前 MIPS 是世界上用得最多的处理器)，其应用领域覆盖游戏机、路由器、激光打印机、掌上电脑等各个方面。MIPS 的系统结构及设计理念比较先进，在设计理念上 MIPS 强调软/硬件协同提高性能，同时简化硬件设计。

MIPS 处理器的发展如表 1.1 所示。下面仅对其中的 MIPS32 24KE 和 MIPS32 74K 做一简单介绍。

表 1.1 MIPS 的发展

| | | | |
|--------|-----------|--------|-------------|
| 1986 年 | R2000 处理器 | 1996 年 | R10000 处理器 |
| 1988 年 | R3000 处理器 | 1997 年 | R12000 处理器 |
| 1991 年 | R4000 处理器 | 2000 年 | MIPS32 24KE |
| 1994 年 | R8000 处理器 | 2007 年 | MIPS32 74K |

MIPS32 24KE 是 MIPS 公司在 2000 年推出的高性能、低功耗的 32 位处理器内核系列。该内核系列采用高性能 24K 微架构，同时集成了 MIPS DSP 特定应用架构扩展(ASE)。24KE 内核系列包括 24KEc、24KEf、24KEc Pro 和 24KEf Pro。目标市场包括机顶盒、DTV、DVD 刻录机、调制解调器与住宅网关等。其功能框图如图 1.4 所示。

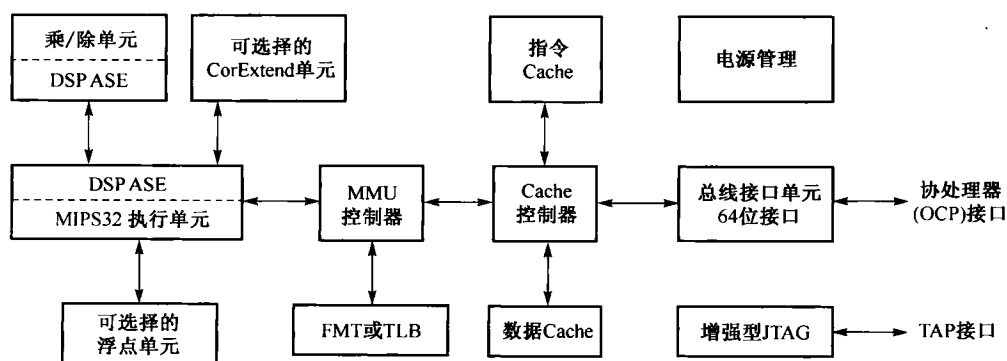


图 1.4 MIPS32 24KE 功能框图

MIPS32 74K(以下简称 74K)是 MIPS 公司在 2007 年推出的内核产品。该产品采用标准硅工艺，是目前嵌入式市场速度较快的可综合处理器内核，主频速度达到 1 GHz 以上。

MIPS32 74K 按照普通单元和 EDA 标准流程设计，采用 65 nm 制造工艺，内核面积为 1.7 mm²。内含自主研发的嵌入式微架构，在同类产品之中的性能/芯片面积比很高。具有如下特点：

- ① 具有 CorExtendTM 功能，该功能可供用户自定义指令。具有二进制兼容的特性，可以直接替代原有的 MIPS32 24K 系列内核，而不需要对应用代码进行任何修改；
- ② 74K 内核运行速度可达到 24K 内核的 1.5 倍到 1.6 倍。
- ③ 双流流水线架构，支持非对称双发。
- ④ 一条 6 级地址生成(AGEN)流水线可处理存储转移负载，并控制传输分支转移指令。另外一条 5 级 ALU 流水线处理所有的与算数、逻辑和计算相关的指令。提供加快 DSP 和媒体处理应用的增强型指令集 DSP ASE(第 2 版)。

(3) SPARC

SPARC(Scalable Processor Architecture)，即可扩展处理器架构，是 SUN 公司在 1985 年提出的体系结构标准，它基于 1980 年到 1982 年间加州大学伯克利分校关于 Berkeley RISC 的研究成果，并由一个独立、非盈利组织 SPARC International 负责 SPARC 架构标准的管理和开发认证，是国际上流行的 RISC(Reduced Instruction Set Computer)微处理器体系架构之一。SPARC 是开放的，任何机构或个人均可研究或开发基于 SPARC 架构的产品，如东芝、富士通、Aeroflex、ESA(Europen Space Agent)、北京时代、珠海欧比特等都在此架构上开发出了自己的 SPARC 微处理器。

1985年, SUN发布了世界上第一个32位可扩展处理器架构标准 SPARC V7。V7定义了 SPARC 体系结构的数据类型、寄存器、指令、存储器模型和异常处理, 处理器指令字长是32位。它采用独立的指令(SAVE, RESTORE)来进行寄存器管理, 用 LOAD 和 STORE 指令访问内存。

1987年, SUN发布了业界第一款有可扩展性功能的32位微处理器 SPARC。因为它采用了 SPARC 的首款架构 SPARC V7, 所以获得了更高的流水线硬件执行效率和更为优化的编译器, 并缩短了其开发周期, 满足了 Sun-4 计算机迅速投放市场的要求。

1990年, SPARC International 发布了32位 SPARC V8 架构标准。它在 SPARC V7 的基础上增加了乘法和除法指令, 加速乘除法的处理, 使得用户不必使用子程序完成相同操作。

1994年, SPARC International 发布了64位 SPARC V9 架构标准。相对于 SPARC V8, 这一版本的显著变化在于: 数据和地址的位宽由32位变到64位, 支持超标量微处理器的实现, 支持容错及多层嵌套陷阱, 具有超快速陷阱处理及上下文切换能力。

1995年以前, 基于 SPARC V7 或 V8 架构的微处理器种类不多, 而且基本上只有 SUN 一家公司在研制开发。从1995年以后, 基于 SPARC V9 架构的64位 SPARC 微处理器日渐丰富, 其面向高性能计算和服务器的微处理器得到了市场广泛的接受, 如 SUN 的 UltraSPARCT1/T2 系列及富士通的 SPARC64 系列等。

2003年, 随着基于 SPARC V8 架构的 LEON2 的发布, 面向高可靠嵌入式领域(如工业控制、军工电子、空间应用等)的 SPARC 微处理器的研制得到了众多公司的青睐。ESA 研制了基于 SPARC V7 架构的 ERC32 微处理器, ATMEL 制造了 SPARC V8 架构的 AT697 微处理器。

SPARC 微处理器具备精简指令集、支持32位/64位数据精度, 架构运行稳定、可扩展性优良、体系标准开放等特点。此外, 寄存器窗口技术既是 SPARC 微处理器的显著特点, 也是 SPARC 架构不同于由斯坦福大学提出的 MIPS 微处理器架构的主要不同点之一。采用这项技术可以显著减少过程调用和返回执行时间、执行的指令条数和访问存储器的次数, 从而易于实现直接高效的编译。寄存器窗口技术是将工作寄存器组成若干窗口, 建立一个环形结构, 利用重叠寄存器窗口来加快程序的运转。每个过程分配一个寄存器窗口(含有一组寄存器), 当发生过程调用时, 可以把处理器转换到不同寄存器窗口使用, 无需保存和恢复操作。相邻寄存器窗口部分重叠, 便于调用参数传送。为每个过程提供有限数量的寄存器窗口, 各个过程的部分寄存器窗口重叠。

伴随 LEON2 的发布, SPARC 微处理器在嵌入式应用领域获得了巨大的发展空间, 全球大约已有3万多个成功的应用案例。比较著名的是国际空间站上的控制计算机 DMS-R 及空间自动转移器 ATV 中均使用了 SPARC 微处理器 ERC32, 而在太空观测台 JEM-EUSO 上则使用了 SPARC V8 架构的微处理器。国内研制的 SPARC 微处理器在军工电子领域已得到应用, 在民用领域正处于普及推广应用过程中。

经过20多年的发展, SPARC 微处理器凭借其持续的创新研发能力, 不断取得骄人成绩。在服务器等高端处理器领域, 以及在空间应用等高可靠嵌入式应用领域, SPARC 微处理器发挥着越来越重要的作用。如 ESA 已决定在2013年发射的水星探测任务中采用 SPARC 微处理器。

SPARC 架构标准的开放和最先进的多核心、多线程 SPARC 微处理器的设计代码开放, 促使世界上越来越多的公司、机构和大学加入到 SPARC 微处理器的研发中。到目前为止, 对于开源的 SPARC 微处理器设计代码, 已经有超过10,000个下载。而业界对研究 SPARC 微处理器的积极响应, 必将推动 SPARC 微处理器持续进步, 让它始终具有超强的竞争性。

(4) ARM

ARM 最初是由英国 Acorn Computer 公司于1983~1985年间设计的第一个商用 RISC 微处理器架构。

1990年, ARM(Advanced RISC Machines)公司成立。现 ARM 公司已成为一个全球领先的嵌入式

微处理器 IP (Intellectual Property) 核供应商, 提供一些高性能、低功耗、低成本和高可靠性的 RISC 处理器核、外围部件和系统级芯片设计方案, 自己既不生产芯片也不销售芯片。

ARM 微处理器核的特点是结构简单、低功耗、低成本、高性能, 适用于便携式通信工具、手持式计算机、多媒体数字消费类产品以及其他的嵌入式系统。

关于 ARM 的更多介绍请参见第 2 章。

4. 嵌入式片上系统 SoC

SoC(System on Chip), 意为“片上系统”, 即将一个“系统”所需的功能模块都做一个芯片上。如将 USB、TCP/IP、GPRS、GSM、IEEE1394、蓝牙等模块都集成在一块芯片上, 而这些单元以往都是一个独立的 IC。这样做有如下好处:

- (1) 实现了软、硬件无缝结合, 可直接在微处理器芯片内嵌入操作系统的代码模块。
- (2) 通过改变内部工作电压, 可降低芯片功耗。
- (3) 减少芯片对外引脚数, 简化了制造过程。
- (4) 减少了外围驱动接口单元及电路板之间的信号传递, 可以加快其数据处理的速度。
- (5) 内嵌的线路可以避免外部电路板在信号传递时所造成的系统杂讯干扰。

将 SoC 与 PLD/FPGA 相结合便产生另一种新的嵌入式处理器 SoPC(System On Programmable Chip), 即可编程片上系统, 其结合了 SoC 和 PLD/FPGA 各自的技术优点, 使得系统具有可编程的功能, 是可编程逻辑器件在嵌入式应用中的完美体现, 极大提高了系统的在线升级、换代能力。

嵌入式处理器的以上划分是历史上形成的, 并不一定合理, 实际上有时很难将一款嵌入式处理器归为以上的哪一类, 如三星公司的 S3C2440 嵌入式微处理器片上就集成了 USB、LCD、AC97、摄影机、串口、IIC 与 IIS 等控制器, 算得上是一款 SoC, 但通常仍将其当做一款微处理器。

1.4.2 嵌入式操作系统

嵌入式操作系统是嵌入式系统中的重要软件, 介于应用程序与硬件之间, 向上为应用程序提供使用硬件的接口, 向下管理控制硬件。

嵌入式操作系统一般仅指操作系统的内核, 通常只包括任务管理、存储管理、设备管理与存储器管理等内核模块, 窗口界面、文件以及通信协议等模块则不被包括, 根据需要选用。实际上, 在嵌入式系统中, 除了核外模块可以根据需要进行裁剪外, 内核也需要是可裁剪的, 一方面是因为嵌入式系统的需求千差万别, 另一方面嵌入式系统要求软、硬件必须精简, 不允许有多余的成分, 否则易于引起成本增加或可靠性下降。

大多数嵌入式系统应用在实时环境中, 因此嵌入式操作系统跟实时(Real-Time)操作系统往往关联在一起, 但嵌入式操作系统并非都是实时系统。实时又分为硬实时与软实时两类, 一些嵌入式系统, 如火箭发射、轮船控制等要求操作系统是硬实时的, 另一些嵌入式系统, 如手机、PDA 与订票系统等, 只需软实时甚至非实时都可以。

嵌入式操作系统有近百种, 典型的有 Linux、Windows CE、VxWorks、Psos、Palm OS、OS-9、LynxOS、QNX、LYNX 与 μ C/OS-II 等, 下面选取其中具有代表性的几种做一简单介绍。

(1) VxWork

VxWork 是美国 WindRiver 公司于 1983 年设计开发的一种嵌入式实时操作系统(RTOS), 具有良好的持续发展能力、高性能的内核以及友好的用户开发环境, 在嵌入式实时操作系统领域牢牢占据着一席之地。显著特点是: 可靠性高、实时性和可裁减性好, 支持多种处理器, 如 x86、i960、Sun Sparc、Motorola MC68xxx、MIPS、PowerPC 等。