

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

嵌入式系统设计 大学教程

Embedded System Design
for University Students

刘艺 许大琴 万福 编著

- 采用嵌入式教学架构ARM + Linux
- 全面阐述嵌入式系统设计原理与过程
- 以丰富实例详细讲述嵌入式系统设计



精品系列

 人民邮电出版社
POSTS & TELECOM PRESS

TP360
2008
1

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

嵌入式系统设计 大学教程

Embedded System Design
for University Students

刘艺 许大琴 万福 编著



精品系列

人民邮电出版社

北京

图书在版编目 (CIP) 数据

嵌入式系统设计大学教程 / 刘艺, 许大琴, 万福编著.
北京: 人民邮电出版社, 2008. 11
21世纪高等学校计算机规划教材
ISBN 978-7-115-18827-4

I. 嵌… II. ①刘…②许…③万… III. 微型计算机—系
统设计—高等学校—教材 IV. TP360. 21

中国版本图书馆CIP数据核字 (2008) 第144722号

内 容 提 要

本书以嵌入式系统开发为主线, 以 Linux 操作系统为软件平台, 系统介绍了嵌入式系统开发的基本知识、基本流程和基本方法。全书共分 9 章, 分别对嵌入式系统基础知识、嵌入式硬件开发技术、嵌入式系统软件程序设计内容进行详细介绍, 并附以一个完整的嵌入式系统设计案例。

本书内容丰富, 结构合理, 概念清晰, 既可作为高等院校计算机及相关专业嵌入式课程的教材, 也可供工程技术人员自学参考。

21 世纪高等学校计算机规划教材

嵌入式系统设计大学教程

-
- ◆ 编 著 刘 艺 许大琴 万 福
责任编辑 滑 玉
执行编辑 武恩玉
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市海波印务有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 17
字数: 444 千字 2008 年 11 月第 1 版
印数: 1—3 000 册 2008 年 11 月河北第 1 次印刷

ISBN 978-7-115-18827-4/TP

定价: 29.80 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223

反盗版热线: (010)67171154

出版者的话

计算机应用能力已经成为社会各行业从业人员最重要的工作要求之一，而计算机教材质量的好坏会直接影响人才素质的培养。目前，计算机教材出版市场百花争艳，品种急剧增多，要从林林总总的教材中挑选一本适合课程设置要求、满足教学实际需要的教材，难度越来越大。

人民邮电出版社作为一家以计算机、通信、电子信息类图书与教材出版为主的科技教育类出版社，在计算机教材领域已经出版了多套计算机系列教材。在各套系列教材中涌现出了一批被广大一线授课教师选用、深受广大师生好评的优秀教材。老师们希望我社能有更多的优秀教材集中地呈现在老师和读者面前，为此我社组织了这套“21世纪高等学校计算机规划教材——精品系列”。

本套教材具有下列特点。

(1) 前期调研充分，适合实际教学需要。本套教材主要面向普通本科院校的学生编写，在内容深度、系统结构、案例选择、编写方法等方面进行了深入细致的调研，目的是在教材编写之前充分了解实际教学的需要。

(2) 编写目标明确，读者对象针对性强。每一本教材在编写之前都明确了该教材的读者对象和适用范围，即明确面向的读者是计算机专业、非计算机理工类专业还是文科专业的学生，尽量符合目前普通高等教育计算机课程的教学计划、教学大纲以及发展趋势。

(3) 精选作者，保证质量。本套教材的作者，既有来自院校的一线授课老师，也有来自IT企业、科研机构等单位的资深技术人员。通过他们的合作使老师丰富的实际教学经验与技术人员丰富的实践工程经验相融合，为广大师生编写出适合目前教学实际需求、满足学校新时期人才培养模式的高质量教材。

(4) 一纲多本，适应面宽。在本套教材中，我们根据目前教学的实际情况，做到“一纲多本”，即根据院校已学课程和后续课程的不同开设情况，为同一科目提供不同类型的教材。

(5) 突出能力培养，适应人才市场要求。本套教材贴近市场对于计算机人才的能力要求，注重理论知识与实际应用的结合，注重实际操作和实践动手能力的培养，为学生快速适应企业实际需求做好准备。

(6) 配套服务完善。对于每一本教材，我们在教材出版的同时，都将提供完备的PPT课件，并根据需要提供书中的源程序代码、习题答案、教学大纲等内容，部分教材还将在作者的配合下，提供疑难解答、教学交流等服务。

在本套教材的策划组织过程中，我们获得了来自清华大学、北京大学、人民大学、浙江大学、吉林大学、武汉大学、哈尔滨工业大学、东南大学、四川大学、上海交通大学、西安交通大学、电子科技大学、西安电子科技大学、北京邮电大学、北京林业大学等院校老师的大力支持和帮助，同时获得了来自信息产业部电信研究院、联想、华为、中兴、同方、爱立信、摩托罗拉等企业和科研单位的领导和技术人员的积极配合。在此，向他们表示衷心的感谢。

我们相信，“21世纪高等学校计算机规划教材-精品系列”一定能够为我国高等院校计算机教学做出应有的贡献。同时，对于工作欠缺和不妥之处，欢迎老师和读者提出宝贵的意见和建议。

前言

欢迎学习嵌入式系统！通过嵌入式系统的学习，你将开始探索一个美妙而富有挑战的领域。嵌入式系统已经广泛地渗透到科学研究、工程设计、军事技术、各类产业以及人们的日常生活中。随着国内外各种嵌入式产品的进一步开发和推广，嵌入式技术和人们的生活越来越紧密地结合起来。

嵌入式技术是 IT 领域中发展最快的技术，并将保持高速的技术发展和巨大的商业应用机会。其应用领域非常广泛，而且技术高度分散，谁也无力独霸这一市场，这正是后 PC 时代中国 IT 技术发展面临的难得机遇。加快嵌入式技术的开发应用，掌握嵌入式开发核心技术，对于未能赶上前几次世界范围计算机技术发展机遇的中国来说非常重要。

本书的结构

本书是面向 21 世纪教育改革的高等院校计算机基础教材，针对各高校相关专业的本科生教学而设计，也可作为嵌入式系统开发的初中级设计人员的参考用书。本书以嵌入式系统的基本开发技术为主线，以 ARM 处理器以及 TI 公司应用最为广泛的 DSP 内核与 ARM926EJ-S 微处理器组成的双核应用处理器 OMAP5912 为硬件平台，以免费的、源代码完全开放的 Linux 操作系统为软件平台，系统地讲述了嵌入式系统开发的基本知识、基本流程和基本方法。全书分为 9 章，共 4 大部分：嵌入式系统、硬件开发技术、软件程序设计和开发应用实例，如图 P-1 所示。

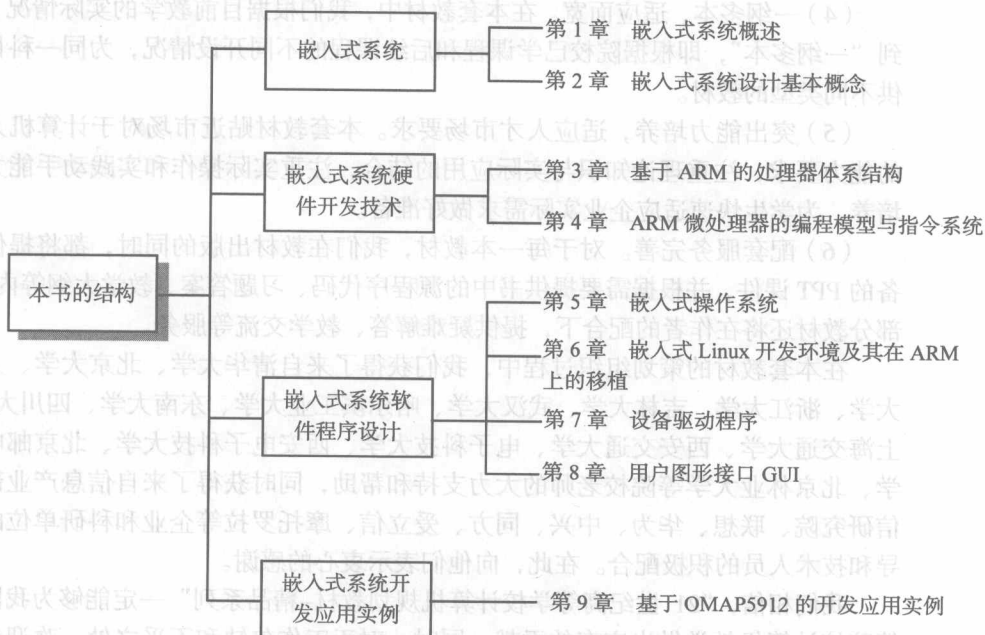


图 P-1 本书纲要

第1部分：嵌入式系统（第1~2章）主要介绍嵌入式系统的入门知识。

第1章：嵌入式系统概述 在这一章，我们介绍了嵌入式系统的定义、特点、体系结构、分类以及嵌入式系统的应用。

第2章：嵌入式系统设计基本概念 在这一章，我们讨论嵌入式硬件系统（处理器及外围硬件）、嵌入式操作系统以及嵌入式系统的基本设计过程。

第2部分：嵌入式系统硬件开发技术（第3~4章）主要介绍嵌入式系统的硬件开发技术。

第3章：基于ARM的处理器体系结构 在这一章，我们讨论ARM处理器的体系结构、处理模式和状态、存储器组织等，对比ARM7与ARM9处理器，介绍基于ARM9的嵌入式硬件平台OMAP5912。

第4章：ARM微处理器的编程模型与指令系统 在这一章，我们讨论ARM微处理器的工作状态、存储器格式、指令系统、指令集以及Thumb指令。

第3部分：嵌入式系统软件程序设计（第5~8章）主要关注嵌入式系统软件的各个方面。

第5章：嵌入式操作系统 在这一章，我们将从概念入手，详细讨论嵌入式操作系统的基本知识，包括嵌入式操作系统的进程、进程调度、进程间通信机制、嵌入式操作系统的中断和时钟管理、嵌入式操作系统的内存管理等，最后介绍了常用的 $\mu\text{C}/\text{OS-II}$ 和Linux操作系统。

第6章：嵌入式Linux开发环境及其在ARM上的移植 在这一章，我们将讨论嵌入式Linux开发环境及其在ARM上的移植。我们将说明嵌入式Linux开发环境是如何构建的，以及嵌入式Linux操作系统在ARM上的移植。

第7章：设备驱动程序 在这一章，我们讨论了嵌入式系统的设备驱动程序的管理、接口、文件操作以及中断处理，并从应用的角度出发，举出应用实例详细介绍了字符型设备、网络设备和CAN总线驱动程序的开发方法。

第8章：用户图形接口GUI 在这一章，我们讨论用户图形接口GUI在嵌入式系统中的发展需求和功能特点，分析当前主流的嵌入式GUI系统，介绍Qt/Embedded的特点、体系架构及其开发环境，并给出Qt/Embedded的开发实例。

第4部分：嵌入式系统开发应用实例（第9章）我们以开发平台OMAP5912为例，介绍嵌入式系统的开发应用实例。

第9章：基于OMAP5912的开发应用实例 在这一章，我们讨论了OMAP5912的结构特点，介绍了OMAP5912硬件平台设计，并详细介绍了基于OMAP5912平台的Bootloader的移植、MontaVista Linux内核的移植、文件系统的移植和Linux设备驱动程序的设计和使用。

本书的特点

本书的一些特色不仅使得本书出类拔萃，同时也特别有助于刚入门的学生学习。

理论与实践相结合

我们将理论与实践相结合贯穿本书，既强调嵌入式系统的基本知识、基本理论的学习，又结合实际应用提升嵌入式系统教学的系统性和实用性。

图文并茂

本书图文并茂。全书提供了100多幅图片，以增进读者对文字的理解。

免费提供教辅材料和 PPT

本书为授课教师提供教辅材料（包括 PPT 课件及思考与练习题答案）有需要者请登录人民邮电出版社教育服务与资源网（<http://www.ptpedu.com.cn>）免费下载。

致谢

除了作者的辛勤劳作，本书的顺利出版还得到了许多人的支持，在此对他们表示衷心的感谢。在本教材的编写过程中，得到了人民邮电出版社的大力支持。海军工程大学吴学智教授也给予了很大的支持和帮助。

感谢我们的研究室领导宋祥斌和谈亮，他们给我们提供了宽松的环境和良好的实验条件。

感谢我们的家人和朋友，他们为本书的付出是值得的。

参加本书编写的还包括谷秀莲、尹亚兰和蔡敏，他们不仅编写了部分章节，而且还认真阅读了全书，提出了大量中肯的建议，同时修订了书稿中的很多错误。

嵌入式系统设计是一门综合性很强的技术，涉及学科面宽、发展速度快。限于作者的能力和水平，本书有限的篇幅不可能完全覆盖嵌入式技术的方方面面，书中内容也难免出现各种错误和不足。敬请读者谅解，并真诚地欢迎读者提出宝贵的意见和建议。

2008年7月

本书特点

本书与同类教材相比，具有以下特点：

1. 注重理论与实践相结合

本书在编写过程中，力求做到理论与实践相结合，既注重基础知识的传授，又注重实践能力的培养。全书共分10章，前4章主要介绍嵌入式系统的基本概念、组成、开发方法和应用，后6章主要介绍嵌入式系统的硬件设计、软件开发和系统应用。

本书文图

本书在编写过程中，力求做到理论与实践相结合，既注重基础知识的传授，又注重实践能力的培养。全书共分10章，前4章主要介绍嵌入式系统的基本概念、组成、开发方法和应用，后6章主要介绍嵌入式系统的硬件设计、软件开发和系统应用。

目 录

第 1 部分 嵌入式系统

第 1 章 嵌入式系统概述.....2

- 1.1 什么是嵌入式系统.....2
- 1.2 嵌入式系统的特点.....3
- 1.3 嵌入式系统的体系结构.....4
 - 1.3.1 硬件平台.....5
 - 1.3.2 硬件抽象层.....6
 - 1.3.3 实时操作系统.....7
 - 1.3.4 实时应用程序.....7
- 1.4 嵌入式系统的分类.....7
- 1.5 嵌入式系统的应用.....9
- 本章小结.....10
- 思考与练习题.....11

第 2 章 嵌入式系统设计基本概念.....12

- 2.1 嵌入式硬件系统.....12
 - 2.1.1 嵌入式处理器.....13
 - 2.1.2 存储器.....18

- 2.1.3 外围设备.....22

2.2 嵌入式操作系统.....28

- 2.2.1 嵌入式操作系统的发展.....28
- 2.2.2 嵌入式实时操作系统的分类.....28
- 2.2.3 嵌入式实时操作系统的可裁剪性及其实现.....29
- 2.2.4 常用的实时操作系统.....30

2.3 嵌入式系统的基本设计过程.....33

- 2.3.1 需求分析.....34
- 2.3.2 规格说明.....36
- 2.3.3 体系结构设计.....36
- 2.3.4 设计硬件构件和软件构件.....40
- 2.3.5 系统调试与集成.....41

2.4 本教程选择的软硬件平台.....41

- 2.4.1 教学硬件平台——ARM.....41
- 2.4.2 教学软件平台——Linux.....41

本章小结.....42

思考与练习题.....42

第 2 部分 嵌入式系统硬件开发技术

第 3 章 基于 ARM 的处理器体系结构.....46

- 3.1 ARM 体系结构.....47
 - 3.1.1 ARM 微处理器结构.....47
 - 3.1.2 ARM 工作状态.....53
 - 3.1.3 ARM 微处理器的寄存器组织.....53
 - 3.1.4 ARM 微处理器的指令结构.....53
- 3.2 ARM9 处理器简介.....53
 - 3.2.1 与 ARM7 处理器比较.....54
 - 3.2.2 OMAP5912 处理器介绍.....58
- 本章小结.....61

思考与练习题.....62

第 4 章 ARM 微处理器的编程模型与指令系统.....63

- 4.1 ARM 微处理器的工作状态.....63
- 4.2 ARM 微处理器的运行模式.....64
- 4.3 ARM 体系结构的存储器格式.....64
 - 4.3.1 大端格式.....64
 - 4.3.2 小端格式.....65
- 4.4 寄存器组织.....65
 - 4.4.1 ARM 状态下的寄存器组织.....65

4.4.2 Thumb 状态下的寄存器组织	69	4.7.6 堆栈寻址	80
4.5 ARM 的异常	70	4.7.7 相对寻址	80
4.5.1 ARM 体系结构所支持的异常类型	70	4.8 ARM 指令集	81
4.5.2 异常向量 (Exception Vectors)	71	4.8.1 数据处理指令	81
4.5.3 对异常的响应	71	4.8.2 乘法指令与乘加指令	85
4.5.4 从异常返回	72	4.8.3 程序状态寄存器访问指令	87
4.5.5 各类异常的具体描述	73	4.8.4 加载/存储指令	88
4.5.6 异常的进入/退出	74	4.8.5 批量数据加载/存储指令	89
4.5.7 异常优先级 (Exception Priorities)	75	4.8.6 跳转指令	90
4.6 ARM 微处理器的指令系统基础	75	4.8.7 数据交换指令	91
4.6.1 指令长度及数据类型	75	4.8.8 移位指令 (操作)	92
4.6.2 ARM 微处理器的指令的分类与格式	76	4.8.9 协处理器指令	93
4.6.3 指令的条件域	76	4.8.10 异常产生指令	94
4.7 ARM 指令的寻址方式	77	4.9 Thumb 指令及应用	95
4.7.1 立即寻址	77	4.9.1 Thumb 编程模型	95
4.7.2 寄存器寻址	77	4.9.2 Thumb 状态切换	96
4.7.3 寄存器间接寻址	78	4.9.3 Thumb 指令集特点	96
4.7.4 基址变址寻址	78	本章小结	97
4.7.5 多寄存器寻址	79	思考与练习题	97

第 3 部分 嵌入式系统软件程序设计

第 5 章 嵌入式操作系统 100

5.1 嵌入式系统的进程	100
5.1.1 进程的概念	101
5.1.2 上下文切换	102
5.1.3 进程状态	103
5.1.4 进程调度	104
5.1.5 进程间通信机制	110
5.2 嵌入式操作系统的中断和时钟管理	115
5.2.1 中断管理	115
5.2.2 时钟管理	118
5.3 嵌入式操作系统的内存管理	120
5.3.1 内存管理的主要功能	120
5.3.2 虚拟内存的概念	120
5.4 常用嵌入式操作系统	121
5.4.1 $\mu\text{C}/\text{OS-II}$	121
5.4.2 Linux 操作系统	130

本章小结	139
思考与练习题	140

第 6 章 嵌入式 Linux 开发环境 及其在 ARM 上的移植 141

6.1 嵌入式 Linux 开发环境	142
6.1.1 交叉编译工具介绍	143
6.1.2 交叉编译环境的建立	155
6.2 嵌入式 Linux 在 ARM 平台上的移植	161
6.2.1 Linux 内核源代码的基本组织情况	161
6.2.2 嵌入式 Linux 内核裁剪方法	162
6.2.3 嵌入式 Linux 内核定制过程	165
6.2.4 内核编译及装载	169
6.2.5 文件系统及其实现	169
本章小结	172
思考与练习题	173

第 7 章 设备驱动程序174	第 8 章 用户图形接口 GUI205
7.1 概述.....174	8.1 嵌入式系统中的 GUI.....205
7.1.1 设备驱动原理.....175	8.1.1 嵌入式 GUI 的发展需求.....206
7.1.2 模块化编程.....176	8.1.2 嵌入式 GUI 的功能特点.....206
7.1.3 设备类型.....177	8.1.3 目前流行的嵌入式 GUI 系统.....207
7.1.4 设备号.....181	8.2 Qt/Embedded 基础.....211
7.2 设备文件接口.....182	8.2.1 Qt/Embedded 简介.....212
7.2.1 用户访问接口.....182	8.2.2 Qt/Embedded 特点.....212
7.2.2 一些重要数据结构.....183	8.2.3 Qt/Embedded 体系架构.....213
7.2.3 I/O 操作.....187	8.3 Qt/Embedded 开发环境.....219
7.3 中 断 处 理.....189	8.3.1 Qt/E 2.x 系列.....220
7.3.1 注册中断处理程序.....189	8.3.2 Qt/E 3.x 系列.....222
7.3.2 中断处理程序实现.....191	8.4 Qt/Embedded 开发实例.....223
7.4 应用实例.....191	8.4.1 Qt/Embedded 应用系统基本 开发流程.....223
7.4.1 字符设备——按键驱动程序.....192	8.4.2 Qt/Embedded 下触摸屏驱动的 设计.....226
7.4.2 网络设备——CS8900A 芯片 驱动程序.....193	本章小结.....231
7.4.3 CAN 总线驱动开发.....200	思考与练习题.....231
本章小结.....203	
思考与练习题.....203	

第 4 部分 嵌入式系统开发应用实例

第 9 章 基于 OMAP5912 的 开发应用实例234	9.2.4 外围接口.....242
9.1 MAP5912 的结构和特点.....234	9.3 基于 OMAP5912 的软件系统设计.....243
9.1.1 ARM926EJ-S 内核.....235	9.3.1 OMAP5912 系统的软件架构.....243
9.1.2 TMS320C55x 内核.....235	9.3.2 嵌入式 Linux 系统的启动流程.....244
9.1.3 存储器管理.....235	9.3.3 Bootloader 及其移植.....244
9.1.4 直接存储器访问控制器 (DMA).....236	9.3.4 MontaVista Linux 内核的移植.....251
9.1.5 时钟和电源管理.....237	9.3.5 文件系统的移植.....254
9.1.6 外围控制模块.....237	9.3.6 基于 OMAP5912 平台的 Linux 设备驱动程序.....255
9.2 基于 OMAP5912 的硬件平台设计.....238	本章小结.....256
9.2.1 电源管理模块.....239	思考与练习题.....256
9.2.2 存储模块.....240	缩略语257
9.2.3 音频处理模块.....240	参考文献260

第一章

嵌入式系统

第 1 部分

嵌入式系统

- 第 1 章 嵌入式系统概述
- 第 2 章 嵌入式系统设计基本概念

第 1 章

嵌入式系统概述

在本章，我们将讨论嵌入式系统的基本概念，具体讲解嵌入式系统是如何定义的，嵌入式系统的特点、体系结构和分类等问题，并举例说明嵌入式系统的应用领域。

本章学习要求：

- 定义嵌入式系统；
- 描述嵌入式系统的特点；
- 列出嵌入式系统的体系结构；
- 列出嵌入式系统的分类；
- 描述嵌入式系统的应用。

随着社会信息化的日益加强，计算机和网络已经全面渗透到日常生活的每一个角落。对于我们每个人来说，需要的已经不仅仅是那种放在桌上处理文档、进行工作管理和生产控制的计算机“机器”。任何一个普通人都可能拥有大小不一、形状各异、使用嵌入式技术的电子产品，小到手表、MP3、移动电话、PDA，大到电视、冰箱、电动脚踏车乃至汽车等数字化产品。

目前，各种各样的嵌入式系统大量应用到各个领域，从国防武器装备、网络通信设备到智能仪器、日常消费电子设备，再到生物微电子技术，处处都可以见到嵌入式系统的身影。

1.1 什么是嵌入式系统

什么是嵌入式系统呢？

根据美国电气和电子工程师协会（IEEE）的定义，嵌入式系统是用来控制、监视或辅助设备、机器或工厂操作的装置。

中国计算机学会微机专业委员会的定义是，嵌入式系统是以嵌入式应用为目的的计算机系统，可分为系统级、板级和片级。

- 系统级：各种类型的工控机、PC104 模块。
- 板级：各种类型的带 CPU 的主板及 OEM 产品。
- 片级：各种以单片机、DSP、微处理器为核心的产品。

嵌入式系统一般定义为以应用为中心、以计算机技术为基础，软硬件可裁剪，应用系统对功能、可靠性、成本、体积、功耗和应用环境有特殊要求的专用计算机系统。

从技术角度说，嵌入式系统是将应用程序、操作系统和计算机硬件集成在一起的系统。

从系统角度说,嵌入式系统是设计完成复杂功能的硬件和软件,并使其紧密耦合在一起的计算机系统。

从广义上讲,凡是带有微处理器的专用软硬件系统都可称为嵌入式系统,如各类单片机和 DSP 系统。这些系统在完成较为单一的专业功能时具有简洁高效的特点。但由于它们没有操作系统,管理系统硬件和软件的能力有限,在实现复杂多任务功能时,往往困难重重甚至无法实现。因此,我们更加推荐那些使用嵌入式微处理器构成独立系统,具有自己的操作系统,具有特定功能,用于特定场合的嵌入式系统。

简而言之,一个嵌入式系统就是一个硬件和软件的集合体,它包括硬件和软件两部分。其中硬件包括嵌入式处理器、控制器、数字信号处理器(DSP)、存储器及外设器件、输入输出(I/O)端口、图形控制器等;软件部分包括操作系统软件(嵌入式操作系统)和应用程序(应用软件),由于应用领域不同,应用程序千差万别。

1.2 嵌入式系统的特点

20 世纪 70 年代,Intel 公司推出有史以来第一个微处理器 4004。计算机的形态和应用因而出现了历史性的变化。基于高速数值解算能力的微型机表现出的智能化水平引起了控制专业人士的兴趣,他们要求将微型机嵌入到一个对象体系中,实现对象体系的智能化控制。例如,将微型计算机经电气加固、机械加固,并配置各种外围接口电路,安装到大型舰船中构成自动驾驶仪或轮机状态监测系统。这样一来,计算机便失去了原来的形态与通用的计算机功能。为了区别于原有的通用计算机系统,把嵌入到对象体系中,实现对象体系智能化控制的计算机,称作嵌入式计算机系统。嵌入式系统的嵌入性本质就是将一个计算机嵌入到一个对象体系中去。

典型的嵌入式系统几乎让人感觉不到它的存在。嵌入式系统特别强调“量身订做”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来,所以我们很难不经过“重大”修改而直接将一个嵌入式系统套用到其他的嵌入式产品上去。

与通用的计算机系统相比,嵌入式系统具有以下显著特点。

1. 系统内核小

由于嵌入式系统一般应用于小型电子装置,系统资源相对有限,所以内核较传统的操作系统要小得多。比如,ENE 公司的 OSE 分布式系统,内核只有 5KB;3Com 公司的 32 位嵌入式操作系统 Palm OS,内核为几十 KB;微软开发的基于掌上型电脑类操作的 32 位嵌入式操作系统 Windows CE,核心占 500KB 的 ROM 和 250KB 的 RAM,整个 Windows CE 操作系统包括硬件抽象层、Windows CE 内核、User、GUI、文件系统和数据库,大约共 1.5MB。

而传统的操作系统,如 Windows 的内核,则要大得多。微软正在着手精简的 Windows 7 内核代码,大小约为 25MB。

2. 专用性强

嵌入式系统通常是面向特定任务的,相对于一般通用 PC 计算平台,嵌入式系统的个性化很强,其中软件系统和硬件的结合非常紧密,一般要针对硬件进行软件系统的移植。即使在同一品牌、同一系列的产品中也需要根据硬件的变化和增减不断进行修改。针对不同的任务,往往需要对系统进行较大更改,有时甚至要废弃整个系统并重新进行设计。

3. 运行环境差异大

嵌入式系统使用范围极为广泛，其运行环境差异很大。如运行在冰天雪地的南北极、温度很高的汽车里、要求恒温 and 恒湿的科学实验室等，特别是在恶劣环境中，或者突然断电等情况下，要求系统仍能够正常工作。

4. 可靠性要求高

嵌入式系统不能像通用 PC 机一样，“死机”时通过手动重启计算机。嵌入式系统往往要长期在无人值守的环境下运行，甚至是常年运行，因此对可靠性的要求特别高。

如果说在普通应用（如手机等消费类电子产品）中，有少量错误还可以接受的话，那么，在一些特殊应用场合，如核电站、航天航空、工业控制、汽车制动等，系统的一个错误就可能造成很大的损失。为实现高可靠性要求这一目标，就需要建立科学的系统测试和可靠性评估体系，同时引入一些系统自动恢复的机制。如看门狗定时器，在软件失去控制后能使之重新开始正常运行。

5. 系统精简和高实时性操作系统

现在许多嵌入式系统要胜任的工作越来越复杂，嵌入式操作系统就成为嵌入式系统设计中必不可少的一个环节。嵌入式操作系统与传统操作系统的基本功能是一致的，但嵌入式操作系统还有其独特的需求和技术特点。

一是许多嵌入式系统应用具有实时性要求。嵌入式任务往往是时间关键性约束的，必须在某个时间范围内完成。比如，核电站中的堆芯温度控制系统，如果没有对堆芯过热作出及时处理，后果不堪设想。

二是由于嵌入式系统比通用 PC 系统资源少得多，要求嵌入式操作系统核心部分的体积必须尽可能小。

三是为了适应各种应用需求的变化，嵌入式操作系统还应该具有可裁减性、可伸缩性和易移植性的特点。

6. 具有固化在非易失性存储器中的代码

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 和 FLASH）芯片中。嵌入式系统开机后，必须有代码对系统进行初始化，以便其余的代码能够正常运行，这就是建立运行时的环境。比如，初始化 RAM 放置变量，测试内存的完整性，测试 ROM 完整性以及其他初始化任务。

为了系统的初始化，几乎所有系统都要在非易失性存储器中存放部分代码（启动代码）。为了提高执行速度和系统可靠性，大多数嵌入式系统常常把所有代码（或者其压缩代码）固化，存放在存储器芯片或处理器的内部存储器件中，而不使用外部存储介质。

7. 嵌入式系统开发工作和环境

嵌入式系统开发需要专门的开发工具和环境。由于嵌入式系统本身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、混合信号示波器等的。开发时有主机和目标机，主机用于程序的开发，目标机作为最后的执行机，开发时需要二者交替结合进行。

1.3 嵌入式系统的体系结构

通用计算机一般由下列部分组成：

- 微处理器;
- 大型存储器, 包括主存储器(半导体存储器——RAM、ROM 以及可快速访问的高速缓存)和辅助存储器(硬盘、磁盘和磁带中的磁性存储器, 以及 CD-ROM 中的光存储器);
- 输入单元, 例如键盘、鼠标、数字转换器、扫描仪等;
- 输出单元, 例如显示器、打印机等;
- 网络单元, 例如以太网卡、前端基于处理器的驱动程序等;
- I/O 单元, 例如调制解调器、附有调制解调器的传真机等。

嵌入式系统是将嵌入了软件的计算机硬件作为其最重要部分的系统, 它是一种专用于某个应用或产品的特殊的计算机系统。由于其软件通常嵌入在 ROM(只读存储器)中, 因此, 不像计算机那样需要辅助存储器。早期嵌入式系统自底向上包含 3 个部分: 硬件平台、嵌入式实时操作系统(RTOS)和嵌入式实时应用程序, 如图 1-1 所示。

由于嵌入式系统应用的硬件环境差异较大, 因此, 如何简洁有效地使嵌入式系统能够应用于各种不同的应用环境是嵌入式系统发展中必须解决的关键问题。

经过不断的发展, 原先嵌入式系统的 3 层结构逐步演化成为一种 4 层结构。这个新增加的中间层次位于操作系统和硬件之间, 包含了系统中与硬件相关的大部分功能。它通过特定的上层接口与操作系统进行交互, 向操作系统提供底层的硬件信息, 并根据操作系统的要求完成对硬件的直接操作。

由于引入了一个中间层次, 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, 而是面向由这个中间层次所代表的、逻辑上的硬件环境, 因此, 把这个中间层次叫做硬件抽象层(Hardware Abstraction Layer, HAL)。图 1-2 显示了引入 HAL 以后的嵌入式系统结构。HAL 的引入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。

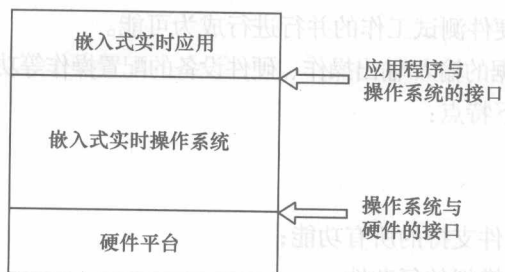


图 1-1 嵌入式系统的基本结构

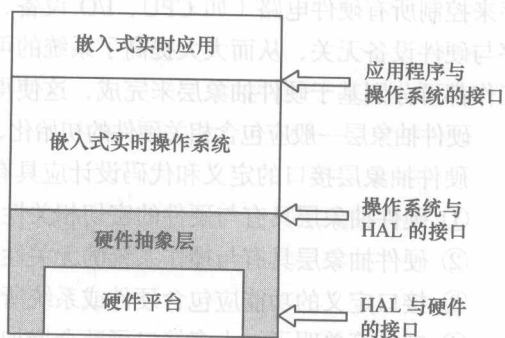


图 1-2 引入 HAL 后的嵌入式系统结构

1.3.1 硬件平台

嵌入式系统的硬件平台是以嵌入式处理器为核心, 由存储器、I/O 单元电路、通信模块、外部设备等必要的辅助接口组成的, 如图 1-3 所示。硬件平台是整个嵌入式实时操作系统和实时应用程序运行的硬件基础。

不同的应用通常有不同的硬件环境, 硬件平台的多样性是嵌入式系统的一个主要特点。在实际应用中, 除了微处理器和基本的外围电路以外, 其余的电路都可根据需要和成本进行裁剪、定制。

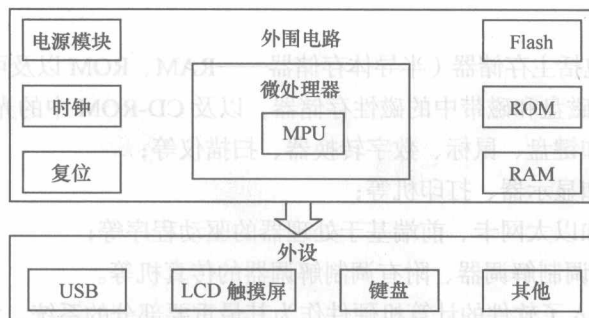


图 1-3 嵌入式系统硬件基本组成

1.3.2 硬件抽象层

硬件抽象层通过硬件抽象层接口向操作系统以及应用程序提供对硬件进行抽象后的服务。当操作系统或应用程序使用硬件抽象层 API 进行设计时，只要硬件抽象层 API 能够下层硬件平台上实现，那么操作系统和应用程序的代码就可以移植。

在整个嵌入式系统设计过程中，硬件抽象层同样发挥着不可替代的作用。传统的设计流程是采用瀑布式设计开发过程，首先是硬件平台的制作和调试，而后是在已经定型的硬件平台的基础上再进行软件设计。由于硬件和软件的设计过程是串行的，因此需要很长的设计周期；而硬件抽象层能够使软件设计在硬件设计结束前开始进行，使整个嵌入式系统的设计过程成为软硬件设计并行的 V 模式开发过程。这样两者的设计过程大致是同时进行的或是并发的，大大缩短了整个设计周期。

1. 硬件抽象层

硬件抽象层（Hardware Abstraction Layer, HAL）是位于操作系统内核与硬件电路之间的接口层，其目的是将硬件抽象化，为上层的驱动程序提供访问硬件设备寄存器函数包，即可以通过程序来控制所有硬件电路（如 CPU、I/O 设备、存储器等）的操作。这样就使得系统的设备驱动程序与硬件设备无关，从而大大提高了系统的可移植性。从软硬件测试的角度来看，软硬件的测试工作都可分别基于硬件抽象层来完成，这使得软硬件测试工作的并行进行成为可能。

硬件抽象层一般应包含相关硬件的初始化、数据的输入/输出操作、硬件设备的配置操作等功能。

硬件抽象层接口的定义和代码设计应具有以下特点：

- ① 硬件抽象层具有与硬件的密切相关性；
- ② 硬件抽象层具有与操作系统的无关性；
- ③ 接口定义的功能应包含硬件或系统所需硬件支持的所有功能；
- ④ 定义简单明了，太多接口函数会增加软件模拟的复杂性；
- ⑤ 可测性的接口设计有利于系统的软硬件测试和集成。

2. 板级支持包

板级支持包（Board Support Package, BSP）是现有的大多数商用嵌入式操作系统实现可移植性所采用的一种方案，是硬件抽象层的一种实现。BSP 隔离了所支持的嵌入式操作系统与底层硬件平台之间的相关性，使嵌入式操作系统能够通用于 BSP 所支持的硬件平台，从而实现嵌入式操作系统的可移植性和跨平台性，以及嵌入式操作系统的通用性、复用性。

BSP 是相对于操作系统而言的，不同的操作系统对应于不同定义形式的 BSP。例如，对于同一个 CPU 来说，要实现同样的功能，VxWorks 的 BSP 和 Linux 的 BSP 的写法和接口定义却完全不同。因此，BSP 一定要按照具体操作系统 BSP 的定义形式来写（BSP 的编程过程大多数是在某一个成型

的 BSP 模板上进行修改), 这样才能与上层操作系统保持正确的接口, 良好地支持上层操作系统。

BSP 实现的功能大体有以下两个方面。

- **系统启动时, 完成对硬件的初始化。**例如, 对系统内存、寄存器以及设备的中断进行设置。这是比较系统化的工作, 它要根据嵌入式开发所选的 CPU 类型、硬件以及嵌入式操作系统的初始化等多方面决定 BSP 应实现什么功能。
- **为驱动程序提供访问硬件的手段。**驱动程序经常要访问设备的寄存器, 对设备的寄存器进行操作。如果整个系统为统一编址, 则开发人员可直接在驱动程序中用 C 语言的函数访问设备寄存器。但是, 如果系统为单独编址, 则 C 语言就不能直接访问设备中的寄存器, 只有汇编语言编写的函数才能进行对外围设备寄存器的访问。BSP 就是为上层的驱动程序提供访问硬件设备寄存器的函数包。

1.3.3 实时操作系统

实时多任务操作系统 (Real Time multi-tasking Operation System, RTOS) 简称实时操作系统, 主要用来完成嵌入式实时应用的任务调度和控制等核心功能。这些功能是通过内核服务函数形式交给用户调用的, 也就是 RTOS 的系统调用, 或者叫做 RTOS 的 API。

实时操作系统可根据实际应用环境的要求对内核进行裁剪和重新配置, 根据不同的应用, 其组成有所不同, 但实时内核、网络组件、文件系统和图形接口等几个重要组成部分是不太变化的。实时操作系统的引入大大提高了嵌入式系统开发的效率, 减少了系统开发的总工作量, 而且提高了嵌入式应用软件的可移植性。

RTOS 的体系结构如图 1-4 所示。

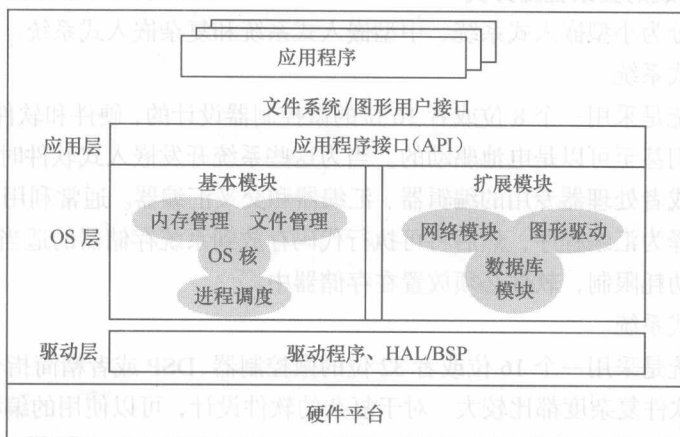


图 1-4 RTOS 的体系结构图

1.3.4 实时应用程序

实时应用程序运行于操作系统之上, 利用操作系统提供的实时机制完成特定功能的嵌入式应用。不同的系统需要设计不同的嵌入式实时应用程序。应用程序是面向被控对象和用户的, 当需要用户操作时, 往往需要提供一个友好的人机界面。

1.4 嵌入式系统的分类

根据不同的分类标准, 嵌入式系统有不同的分类方法。