



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

ARM 9

嵌入式系统开发与应用

◎ 董 胡 刘 刚 钱盛友 著



中国工信出版集团



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

ARM 9 嵌入式系统开发与应用

董 胡 刘 刚 钱盛友 著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

作为一种 16/32 位的高性能、低成本、低功耗嵌入式 RISC 微处理器, ARM 微处理器目前已经成为应用最为广泛的嵌入式微处理器。

本书全面系统地介绍了嵌入式系统开发的基本知识和方法。全书分为四部分。第一部分介绍嵌入式系统基本概念及广泛使用的 ARM 技术, 包括 ARM 处理器的体系结构、寻址方式、指令系统、汇编语言程序和 C 语言程序设计基础及 ARM 嵌入式硬件设计基础。第二部分介绍基于 ARM 920T 内核的三星 S3C2440 处理器, 以及基于 S3C2440 处理器的应用系统设计, 并以 S3C2440 为开发平台, 列举几个典型的基本功能部件的程序设计示例。第三部分介绍 Boot Loader 及实现、ARM Linux 系统移植与驱动开发。第四部分介绍 ARM ADS 集成开发环境及几个嵌入式系统应用开发实例。

本书可作为高等院校电子信息类、计算机类专业高年级本科学生和研究生教材, 或作为嵌入式系统应用设计人员的培训用书, 也可作为基于 ARM 的软件编程和硬件系统设计的参考手册。

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

ARM 9 嵌入式系统开发与应用/董胡, 刘刚, 钱盛友著. —北京: 电子工业出版社, 2015. 6

(卓越工程师培养计划)

ISBN 978 - 7 - 121 - 26032 - 2

I. ①A… II. ①董… ②刘… ③钱… III. ①微处理器 - 系统设计 IV. ①TP332

中国版本图书馆 CIP 数据核字 (2015) 第 097806 号

策划编辑: 袁 玺

责任编辑: 袁 玺 特约编辑: 刘宪兰

印 刷: 三河市兴达印务有限公司

装 订: 三河市兴达印务有限公司

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 × 1 092 1/16 印张: 18 字数: 460.8 千字

版 次: 2015 年 6 月第 1 版

印 次: 2015 年 6 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。

序

嵌入式系统技术是当今信息技术中最具生命力的新技术之一，嵌入式系统应用几乎无处不在：移动电话、家用电器、汽车……无不有它的踪影。嵌入式控制器因其体积小、可靠性高、功能强、灵活方便等许多优点，其应用已深入到工业、农业、教育、国防、科研以及日常生活等各个领域，对各行各业的技术改造、产品更新换代、加速自动化进程、提高生产率等方面起到了极其重要的推动作用。制造工业、过程控制、网络、通信、仪器、仪表、汽车、船舶、航空、航天、军事装备、消费类产品等方面均是嵌入式技术的应用领域。嵌入式系统技术是专用计算机技术，其目的是要把一切变得更简单、更方便、更普遍、更适用；通用计算机已发展为功能电脑，并已普遍进入社会，而嵌入式计算机发展的目标是专用电脑，实现“普遍化计算”。国内嵌入式产业现已成为 IT 产业中的重要新兴产业，这对渴望学习和掌握嵌入式技术应用的相关人员是一个非常好的契机。嵌入式系统包含硬件、操作系统、应用软件三部分，是一个综合性系统，要真正掌握和应用好嵌入式系统技术，一方面需要有对应的硬件学习平台，另一方面需要有针对性具体硬件平台的教科书作为指导。

本书首先系统地介绍嵌入式系统的基础知识，以嵌入式系统的软、硬件开发流程为主线，分析了 ARM 9 微处理器 S3C2440 的应用系统设计过程，列举了几个典型的基本功能部件程序设计示例；结合 S3C2440 开发板，对 Boot Loader 及实现、ARM Linux 系统移植与驱动开发进行了较详细的介绍，最后介绍了 Linux 下音视频文件编程与播放、Linux 下的网络编程及基于 Linux 的 MiniGUI 移植与裁剪的开发实例。全书清晰地展现出嵌入式系统开发涉及的相关技术与细节，采用流程框图的形式，直观形象地展现在读者面前，给人以耳目一新的感觉。通过阅读本书，读者不仅可以学习嵌入式系统实现的专门技术，同时对建立嵌入式系统开发与应用的全局观也大有裨益。

本书是作者多年从事嵌入式系统研发和为本科生讲授“嵌入式系统开发与应用”课程的结晶与升华。相信本书会对国内嵌入式系统教学和产品研发实践提供有益帮助，并产生积极的意义。

钱盛友

于湖南师范大学

前言

嵌入式系统是指以应用为核心,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、体积和功耗严格要求的专用计算机系统。作为嵌入式系统的核心,嵌入式微处理器常采用8位或16位微处理器。但由于这些微处理器系统的运行速度、寻址能力和功耗等问题,已很难满足很多相对复杂的嵌入式应用环境。目前开发的16/32位微处理器已逐步开始得到广泛应用,其中尤以32位的ARM 9最为突出。

在所有ARM 9微处理器系列中,ARM 920T微处理器系列应用最广,采用ARM 920T微处理器作为内核生产芯片的公司最多,同时其性价比也是最高的。因此,本书主要对ARM 920T微处理器S3C2440的结构原理进行介绍,并以此为基础详细介绍嵌入式系统的开发与应用。

本书的各章节内容安排如下。

第1章简要介绍嵌入式系统概念,内容涉及嵌入式系统的概念、特点、分类、结构、开发及应用等。通过对本章的学习,可使读者系统地建立起嵌入式系统开发的整体框架和知识体系。

第2章首先介绍ARM的概念、体系结构的演变与特征、ARM系列及ARM存储数据类型;接着,介绍ARM 9处理器工作状态、ARM 9处理器工作模式、ARM 9处理器寄存器组织,然后详细介绍ARM 9异常、ARM 9存储器和存储器映射I/O、协处理器接口、系统调试接口等。通过对本章的阅读,可使读者了解ARM 9编程模型的基本知识,为进一步的开发做准备。

第3章详细介绍ARM 9体系的指令系统和寻址方式,着重介绍32位的ARM 9指令集。16位的Thumb指令集为32位ARM 9指令集的一个子集,在了解了ARM 9指令集的基础上,就很容易理解Thumb指令。本章所介绍的内容适用于所有具有ARM 920T内核的ARM微处理器。

第4章介绍ARM汇编语言程序设计的基本知识。通过阅读本章,读者可以掌握ARM汇编语言的设计方法。

第5章介绍ARM嵌入式硬件设计基础知识。通过对本章的学习,读者可以掌握对元器件封装建立、原理图绘制、元器件布局及PCB布线的相关知识。

第6章详细介绍基于S3C2440系统的设计全过程,包括特殊功能寄存器及外围芯片的选型,各单元电路的设计步骤和实现细节等。通过对本章的阅读,具有一定嵌入式系统设计知识的读者应该可以掌握基于S3C2440的系统设计,同时由于ARM体系结构的一致性和系统外围电路的通用性,本章所描述的设计方法也同样适合于其他ARM芯片。

第7章详细介绍基于S3C2440系统的各功能模块的工作原理与应用编程,包括GPIO口、中断控制器、定时器工作原理与编程示例,Flash存储器的编程与擦除等。通过对本章的阅读,可使读者了解S3C2440各功能模块的编程方法,并在对应的嵌入式系统开发中加以充分利用。

第8章详细介绍Boot Loader及实现。首先介绍Boot Loader、Boot Loader的种类及操作模式。接着,介绍Boot Loader的启动方式与启动过程。最后对U-Boot编译、移植与调试做详细介绍。通过对本章的学习,可以使读者了解并掌握Boot Loader的工作原理及其实现方法。

第9章详细介绍Linux操作系统概述、Linux内核结构、目录与文件描述、进程调度与管理、开发流程、交叉编译环境、移植过程及硬件接口驱动设计方法等。通过对本章的学习,可以使读者掌握Linux系统移植与驱动开发的工作原理及方法。

第10章介绍ADS 1.2软件的基本组成部分,包括如何安装该软件,如何在CodeWarrior IDE集成开发环境下编写、编译链接工程,使读者能够掌握在ADS软件平台上开发用户应用程序的方法。本章还描述了如何使用AXD调试工程,使读者对于调试工程有个初步的理解,为进一步使用和掌

握调试工具起到抛砖引玉的作用。

第 11 章详细介绍三个嵌入式系统应用开发实例，分别是 Linux 下音/视频文件的编程与播放，Linux 下的网络编程，基于 Linux 的 MiniGUI 移植与裁剪。通过这三个开发实例，为读者进一步学习嵌入式技术并进行嵌入式系统开发应用起到举一反三的作用。

本书内容丰富，系统全面，重点突出，阐述相关知识循序渐进、由浅入深。各章节均安排了丰富的思考题，便于学生自学和自测。

本书由长沙师范学院董胡主著，刘刚、钱盛友参与了部分章节的编写和实例调试工作。全书由长沙师范学院电子与信息工程系马振中主任规划并初审，湖南师范大学博士生导师钱盛友教授主审。在编写过程中，得到了李列文博士、龙慧博士的帮助，在此表示感谢！

由于编者的水平有限，加之时间仓促，书中难免存在一些错误和不妥之处，恳请读者批评指正。

作 者

2015 年 4 月

目 录

第1章 概述	1	2.5.5 ARM 异常描述	24
1.1 嵌入式系统	1	2.5.6 ARM 异常向量表	25
1.1.1 嵌入式系统的概念	1	2.5.7 ARM 异常优先级	25
1.1.2 嵌入式系统的特点	1	2.6 ARM 存储器和存储器映射 I/O	25
1.1.3 嵌入式系统的分类	2	2.6.1 ARM 体系的存储空间	26
1.2 嵌入式系统的组成	2	2.6.2 ARM 存储器格式	26
1.2.1 嵌入式系统的组成结构	2	2.6.3 非对齐存储器访问操作	27
1.2.2 嵌入式系统的硬件特点	3	2.6.4 存储器映射 I/O	28
1.2.3 嵌入式系统的软件介绍	3	2.7 协处理器接口	29
1.3 嵌入式系统的应用领域	4	2.7.1 可用的协处理器	30
1.4 嵌入式处理器	5	2.7.2 关于未定义的指令	30
1.5 嵌入式操作系统	6	2.8 ARM 系统调试接口	30
1.5.1 嵌入式操作系统的种类	6	2.8.1 系统信号和调试工具	30
1.5.2 几种典型的嵌入式操作 系统介绍	7	2.8.2 JTAG 接口及应用	30
1.6 嵌入式系统的开发	9	2.8.3 ETM 接口	31
1.7 本章小结	10	2.9 ATPCS 介绍	31
思考与练习	10	2.10 本章小结	32
思考与练习	10	思考与练习	32
第2章 ARM 体系结构及工作方式	11	第3章 ARM 处理器的指令系统	33
2.1 ARM 体系结构简介	11	3.1 ARM 指令集概述	33
2.1.1 ARM 的概念	11	3.1.1 指令分类和指令格式	33
2.1.2 ARM 体系结构的演变	11	3.1.2 ARM 指令的条件码	36
2.1.3 ARM 体系结构的特征	12	3.2 ARM 处理器寻址方式	37
2.1.4 ARM 系列	13	3.2.1 寄存器寻址	37
2.1.5 ARM 存储数据类型	14	3.2.2 立即寻址	38
2.1.6 ARM 存储器层次	15	3.2.3 寄存器移位寻址	38
2.2 ARM 处理器工作状态	15	3.2.4 寄存器间接寻址	38
2.3 ARM 处理器工作模式	15	3.2.5 基址寻址	38
2.4 ARM 处理器寄存器组织	16	3.2.6 多寄存器寻址	38
2.4.1 ARM 状态下的寄存器组织	16	3.2.7 堆栈寻址	39
2.4.2 Thumb 状态下的寄存器组织	20	3.2.8 块复制寻址	39
2.5 ARM 异常	22	3.2.9 相对寻址	39
2.5.1 ARM 异常概述	22	3.3 ARM 指令分类介绍	40
2.5.2 ARM 异常响应	22	3.3.1 分支指令	40
2.5.3 ARM 异常返回	23	3.3.2 数据处理指令	41
2.5.4 ARM 异常进入/退出	23	3.3.3 存储器访问指令	47

3.3.4 协处理器指令	52	5.3.3 元器件摆放	100
3.3.5 杂项指令	54	5.4 PCB 布线	103
3.3.6 几个常用的伪指令	57	5.4.1 PCB 层叠结构	103
3.4 Thumb 指令分类介绍	59	5.4.2 布线规则设置	104
3.4.1 分支指令	59	5.4.3 布线	104
3.4.2 数据处理指令	60	5.5 本章小结	109
3.4.3 存储器访问指令	64	思考与练习	109
3.4.4 杂项指令	67	第6章 基于 S3C2440 的应用系统设计 ..	110
3.5 本章小结	67	6.1 S3C2440 特性概述	111
思考与练习	67	6.2 S3C2440 内部结构	114
第4章 ARM 汇编语言程序设计	70	6.3 S3C2440 引脚描述	116
4.1 ARM 汇编伪指令	70	6.4 特殊功能寄存器	120
4.1.1 符号定义伪指令	70	6.4.1 存储器控制器	120
4.1.2 数据定义伪指令	71	6.4.2 时钟与电源管理	123
4.1.3 汇编控制伪指令	74	6.5 系统的硬件选型与单元电路设计 ..	127
4.1.4 其他常用的伪指令	75	6.5.1 电源电路设计	127
4.2 ARM 汇编语言语句格式	79	6.5.2 晶振电路设计	129
4.2.1 ARM 汇编语言程序中常见的		6.5.3 复位电路设计	129
符号	80	6.5.4 串行接口电路设计	130
4.2.2 ARM 汇编语言程序中的表达		6.5.5 JTAG 调试接口电路设计	132
式与运算符	80	6.5.6 Flash 接口电路设计	138
4.3 ARM 汇编语言程序结构	83	6.5.7 SDRAM 接口电路设计	143
4.3.1 ARM 汇编语言程序结构	83	6.6 本章小结	146
4.3.2 ARM 汇编语言子程序调用	83	思考与练习	146
4.3.3 ARM 汇编语言和 C/C++ 的		第7章 部件工作原理与编程示例	147
混合编程	84	7.1 S3C2440 GPIO 口工作原理与	
4.4 ARM 汇编语言设计实例	85	编程示例	147
4.5 本章小结	87	7.1.1 概述	147
思考与练习	87	7.1.2 端口控制描述	147
第5章 ARM 嵌入式硬件设计基础	89	7.1.3 GPIO 编程实例	155
5.1 元器件封装建立	89	7.2 S3C2440 中断控制器的工作原理	
5.1.1 新建封装文件	89	与编程示例	157
5.1.2 设置库路径	89	7.2.1 概述	157
5.1.3 画元件封装	90	7.2.2 中断控制器操作	158
5.2 原理图绘制	94	7.2.3 中断源	158
5.2.1 添加元件库及放置元件	94	7.2.4 中断优先级	160
5.2.2 创建电气连接	96	7.2.5 中断控制器特殊寄存器	161
5.2.3 原理图绘制的其他操作	98	7.2.6 中断编程实例	169
5.3 元器件布局	99	7.3 S3C2440 定时器的工作原理与	
5.3.1 建立电路板	99	编程示例	170
5.3.2 导入网络表	99	7.3.1 概述	170

7.3.2	定时器内部逻辑控制工作 流程	171	9.2.2	Linux 源码结构	204
7.3.3	脉宽调制 PWM 实现	171	9.2.3	Linux 内核配置及编译	204
7.3.4	定时器相关寄存器	171	9.3	Linux 目录与文件描述	205
7.3.5	定时器编程实例	173	9.3.1	Linux 目录结构	205
7.4	S3C2440 NAND Flash 存储器的 工作原理与编程示例	174	9.3.2	Linux 文件类型及文件属性 与权限	206
7.4.1	概述	174	9.3.3	Linux 文件系统类型	207
7.4.2	特性	174	9.4	Linux 进程调度与管理	208
7.4.3	软件模式	175	9.4.1	Linux 进程的定义	208
7.4.4	NAND Flash 控制器的寄存器 ..	175	9.4.2	Linux 进程的属性	208
7.4.5	NAND Flash 编程实例	176	9.4.3	Linux 进程调度	208
7.5	本章小结	182	9.5	Linux 系统下多线程	210
	思考与练习	182	9.5.1	Linux 线程	210
第 8 章	Boot Loader 及实现	183	9.5.2	基于 Linux 的多线程编程	210
8.1	Boot Loader 简介	183	9.5.3	Linux 线程属性的修改	212
8.2	Boot Loader 的种类	183	9.6	Linux 开发流程	213
8.3	Boot Loader 的操作模式	184	9.7	Linux 交叉编译环境	215
8.4	Boot Loader 的启动方式	184	9.7.1	Linux 交叉编译	215
8.4.1	网络启动方式	184	9.7.2	基于 S3C2440 的交叉编译 环境建立	216
8.4.2	磁盘启动方式	185	9.8	Linux 的移植过程	217
8.4.3	Flash 启动方式	185	9.8.1	Linux 内核要点	217
8.5	Boot Loader 的启动过程	185	9.8.2	Linux 移植项目	218
8.5.1	Boot Loader 的 stage1	185	9.8.3	制作根文件系统	219
8.5.2	Boot Loader 的 stage2	186	9.9	Linux 下硬件接口驱动设计方法 ..	221
8.6	U-Boot 编译、移植与调试	187	9.10	本章小结	223
8.6.1	U-Boot 介绍	187		思考与练习	223
8.6.2	U-Boot 编译	187	第 10 章	ARM ADS 集成开发环境	225
8.6.3	U-Boot 移植	189	10.1	ADS 1.2 集成开发环境组成	225
8.6.4	U-Boot 调试	190	10.1.1	CodeWarrior IDE 简介	225
8.7	Boot Loader 的实现	199	10.1.2	AXD 调试器简介	226
8.8	本章小结	200	10.1.3	命令行开发工具	226
	思考与练习	200	10.1.4	ARM 运行时库	234
第 9 章	ARM Linux 系统移植与驱动 开发	201	10.1.5	实用程序	236
9.1	Linux 操作系统概述	201	10.1.6	ADS 支持的软件	236
9.1.1	Linux 操作系统的发展	201	10.2	工程的编辑及调试	236
9.1.2	Linux 在嵌入式产品中的优点 ..	201	10.2.1	工程的编辑	237
9.1.3	Linux 版本	202	10.2.2	工程的调试	244
9.2	Linux 操作系统的内核结构	203	10.3	用 AXD 进行代码调试	245
9.2.1	Linux 内核结构	203	10.4	本章小结	247
				思考与练习	248

第 11 章 嵌入式系统应用开发实例	249	11.3.1 MiniGUI 简介	263
11.1 Linux 下音/视频文件编程与 播放	249	11.3.2 MiniGUI 的体系结构	263
11.1.1 Mplayer 简介	249	11.3.3 在宿主机上安装 MiniGUI ...	264
11.1.2 播放本地与远程音视频 文件	252	11.3.4 MiniGUI 的移植	265
11.2 Linux 下的网络编程	253	11.3.5 交叉编译应用程序到 开发板	269
11.2.1 TCP/IP 协议	253	11.3.6 IAL 引擎的移植	271
11.2.2 UDP 协议	254	11.3.7 MiniGUI 的裁剪	275
11.2.3 Socket 编程	254	11.4 本章小结	275
11.3 基于 Linux 的 MiniGUI 移植与 裁剪	263	思考与练习	275
		参考文献	276

第1章 概述

本章主要从嵌入式系统的概念、特点、分类、组成、应用等几个方面介绍嵌入式系统的基本知识，使读者对嵌入式系统建立一个完整的概念。

本章主要内容有：

- 嵌入式系统的概念、特点及分类
- 嵌入式系统的组成
- 嵌入式系统的应用领域
- 嵌入式处理器
- 嵌入式操作系统及开发

1.1 嵌入式系统

1.1.1 嵌入式系统的概念

嵌入式系统（Embedded System）实际上是一个专用的嵌入式计算机系统，国内一般定义为：以应用为中心，计算机技术为基础，软/硬件可裁剪，以适应应用系统对功能、可靠性、成本、体积、功耗有严格要求的专用计算机系统。嵌入式计算机系统是相对于通用计算机系统而言的，通用计算机系统要求满足各种不同的应用需求，因而要求有丰富的硬件资源、网络操作系统、高速的运算、海量的存储，其技术方向是面向总线速度的无限提升，存储容量的无限扩大。而嵌入式计算机系统则面向具体应用，要有针对具体应用的“量体裁衣”的软/硬件，操作系统一般采用实时操作系统，其技术方向是面向与对象系统密切相关的嵌入性能、控制能力与控制的可靠性。

1.1.2 嵌入式系统的特点

由于嵌入式系统是应用于特定环境下，针对特定用途来设计的系统，所以不同于通用计算机系统。同样是计算机系统，嵌入式系统是针对具体应用设计的“专用系统”。与通用的计算机系统相比，它具有以下显著特点。

1. 专用性强

由于嵌入式系统通常是面向某个特定应用的，所以嵌入式系统的硬件和软件，尤其是软件，都是为特定用户群来设计的，它通常都具有某种专用性的特点。

2. 实时性好

目前，嵌入式系统广泛应用于生产过程控制、数据采集、传输通信等场合，主要用来对宿主对象进行控制，所以都对嵌入式系统有或多或少的实时性要求。例如，对嵌入在武器装备中的嵌入式系统，在火箭中的嵌入式系统，一些工业控制装置中的控制系统等应用中的实时性要求就极高。也正因为这种要求，在硬件上嵌入式系统极少使用存取速度慢的磁盘等存储器，在软件上更是加以精心设计，从而可使嵌入式系统快速地响应外部事件。当然，随着嵌入式系统应用的扩展，有些系统对实时性要求也并不是很高，如近年来发展速度比较快的手持式计算机、掌上计算机等。但总体来说，实时性是对嵌入式系统的普遍要求，是设计者和用户重点考虑的一个重要指标。

3. 可裁剪性好

从嵌入式系统专用性的特点来看，作为嵌入式系统的供应者，理应提供各式各样的硬件和软件以备选用。但是，这样做势必会提高产品的成本。为了既不提高成本，又满足专用性的需要，嵌入式系统的供应者必须采取相应措施使产品在通用和专用之间进行某种平衡。目前的做法是把嵌入式

系统硬件和操作系统设计成可裁剪的,以便使嵌入式系统开发人员根据实际应用需要来量体裁衣,去除冗余,从而使系统在满足应用要求的前提下达到最精简的配置。

4. 可靠性高

由于有些嵌入式系统所承担的计算任务涉及产品质量、人身设备安全、国家机密等重大事项,加之有些嵌入式系统的宿主对象要工作在无人值守的场合,如危险性高的工业环境中,内嵌有嵌入式系统的仪器仪表中,在人际罕至的气象检测系统中及为侦察敌方行动的小型智能装置中等。所以与普通系统相比较,对嵌入式系统可靠性的要求极高。

5. 功耗低

很多嵌入式系统的宿主对象都是一些小型应用系统,如移动电话、PDA、MP3、飞机、舰船、数码相机等,这些设备不可能配有容量较大的电源,因此低功耗一直是嵌入式系统追求的目标。当然也是为了降低系统的功耗,嵌入式系统中的软件一般不存储于磁盘等载体中,而都固化在存储器芯片或单片系统的存储器之中。

6. 嵌入式系统开发需要开发工具和环境

由于其本身不具备自主开发能力,即使设计完成以后用户通常也是不能对其中的程序功能进行修改的,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/硬件设备及各种逻辑分析仪、混合信号示波器等。开发时往往有主机和目标机的概念,主机用于程序的开发,目标机作为最后的执行机,开发时需要交替结合进行。

1.1.3 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,这里根据嵌入式系统的复杂程度,可以将嵌入式系统分为以下四类。

1. 单个微处理器

单个微处理器系统可以在小型设备中(如温度传感器、烟雾和气体探测器及断路器)找到。这类设备是供应商根据设备的用途来设计的。这类设备受 Y2K 影响的可能性不大。

2. 不带计时功能的微处理器装置

不带计时功能的微处理器装置可在过程控制、信号放大器、位置传感器及阀门传动器等中找到。这类设备也不太可能受到 Y2K 的影响。但是,如果它依赖于一个内部操作时钟,那么这个时钟可能受 Y2K 问题的影响。

3. 带计时功能的组件

带计时功能的组件可见于开关装置、控制器、电话交换机、电梯、数据采集系统、医药监视系统、诊断及实时控制系统等。它们是一个大系统的局部组件,由它们的传感器收集数据并传递给该系统。这种组件可同 PC 一起操作,并可包括某种数据库(如事件数据库)。

4. 在制造或过程控制中使用的计算机系统

在制造或过程控制中使用的计算机系统,可通过计算机与仪器、机械及设备相连来控制这些装置的工作。这类系统包括自动仓储系统和自动发货系统。在这些系统中,计算机用于总体控制和监视,而不是对单个设备直接控制。过程控制系统可与业务系统连接(如根据销售额和库存量来决定订单或产品量)。

1.2 嵌入式系统的组成

1.2.1 嵌入式系统的组成结构

嵌入式系统由硬件和软件组成,两类不同的嵌入式系统结构模型如图 1.1 所示。硬件是整个嵌入式操作系统和应用程序运行的平台,不同的应用通常有不同的硬件环境。嵌入式系统的硬件部分包括处理器/微处理器、存储器、I/O 接口及输入/输出设备。嵌入式系统的软件由嵌入式操作系统

和应用程序组成。嵌入式操作系统完成嵌入式应用的任务调度和控制等核心功能，嵌入式应用程序运行于操作系统之上，对于一些简单的嵌入式应用系统，应用程序可不需要操作系统的支持，直接运行在底层，如图 1.1 (a) 所示，利用操作系统提供的机制完成特定功能的嵌入式应用。

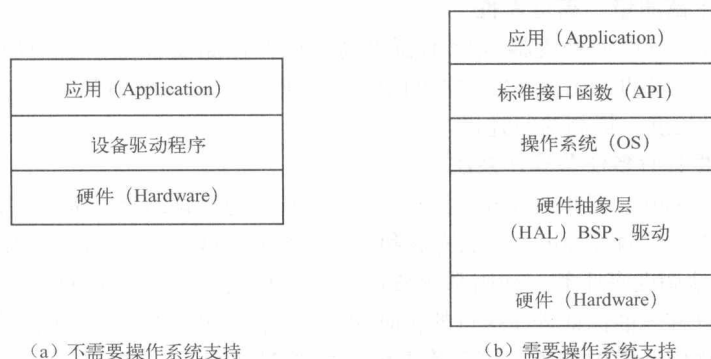


图 1.1 两类不同的嵌入式系统结构模型

硬件平台是整个系统的基础，针对于不同的应用有着不同的硬件平台。其中主芯片、外围器件都与应用紧密相关。硬件平台的多样性是嵌入式系统的一个主要特点。

嵌入式操作系统与硬件接口起到把操作系统与硬件平台相连接的作用。这是整个系统软件的底层部分，其中包括初始化整个系统软件的执行环境、操作系统的启动等。

嵌入式操作系统是整个系统的核心部分，起到承上启下的作用。它完成嵌入式应用任务的调度和控制等核心功能。其中还包括大量的设备驱动程序，通过这些设备驱动程序来和硬件平台打交道。嵌入式操作系统具有可精简、可配置、与上层应用紧密相连等特点。

应用程序与操作系统的接口主要是一些系统所提供的库函数，可供应用程序直接使用。例如，网络的套接字 (socket)、应用编程接口 (Application Programming Interface, API) 函数等。嵌入式应用程序是在操作系统的基础上进行开发的，因此相对于没有操作系统的系统而言，开发量小，系统运行更加稳定和可靠。

1.2.2 嵌入式系统的硬件特点

嵌入式系统的核心硬件是各种类型的嵌入式处理器。嵌入式处理器一般具备以下 4 个特点。

(1) 对实时多任务有很强的支持能力，能完成多任务并且有较短的中断响应时间，从而使内部的代码和实时内核的执行时间减少到最低限度。

(2) 具有功能很强的存储区保护功能。这是由于嵌入式系统的软件结构已模块化，而为了避免在软件模块之间出现错误的交叉作用，需要设计强大的存储区保护功能，同时也有利于软件诊断。

(3) 可扩展的处理器结构，以能最迅速地开发出满足应用的最高性能的嵌入式微处理器。

(4) 嵌入式微处理器必须功耗很低，尤其是用于便携式的无线及移动通信设备中靠电池供电的嵌入式系统更是如此，需要功耗只有 mW 甚至 μW 级。

1.2.3 嵌入式系统的软件介绍

在嵌入式系统中，嵌入式操作系统是嵌入式系统应用的核心。嵌入式操作系统是嵌入式系统软件/硬件资源的控制中心，它以尽量合理有效的方法组织多个用户共享嵌入式系统的各种资源。其中用户指的是系统程序之上的所有软件。所谓合理有效的方法，指的就是操作系统如何协调并充分利用硬件资源来实现多任务。嵌入式操作系统还有一个特点就是针对不同的平台，系统不是直接可用的，一般需要经过针对专门平台的移植操作系统才能正常工作。

嵌入式系统的软件平台包括系统软件与应用软件，它是实现嵌入式系统功能的关键，与通用计算机不同，嵌入式软件主要有以下特点。

1. 软件要求固态化存储

为了提高执行速度和系统可靠性,嵌入式系统中的软件一般都固化在存储器芯片或嵌入式微控制器本身之中,而不是存储于磁盘等载体中。

2. 软件代码要求高质量、高可靠性

尽管半导体技术的发展使处理器速度不断提高、片上存储器容量不断增加,但在大多数应用中,存储空间仍然是宝贵的,还存在实时性的要求。为此要求程序编写和编译工具的质量要高,以减小程序二进制代码长度,提高执行速度。

3. 系统软件需要实时多任务操作系统

在多任务嵌入式系统中,对重要性各不相同的任务进行统筹兼顾的合理调度是保证每个任务及时执行的关键,单纯通过提高处理器速度是无法完成和没有效率的,这种任务调度只能由优化编写的系统软件来完成。为了合理地调度多任务,利用系统资源,用户必须自行选配 RTOS (Real - Time Operating System) 开发平台,这样才能保证程序执行的实时性、可靠性,并减少开发时间,保障软件质量。

4. 嵌入式系统软件开发过程中,可采用汇编语言或 C 语言

对于嵌入式程序开发设计,可以采用 ARM 汇编语言或 C 语言去完成这一工作。

1.3 嵌入式系统的应用领域

由于嵌入式系统具有体积小、性能好、功耗低、可靠性高以及面向行业应用的突出特征,目前已广泛地应用于军事国防、消费电子、信息家电、网络通信、工业控制等领域。其应用领域包括如下几个方面。

1. 工业控制

基于嵌入式芯片的工业自动化设备将获得长足的发展,目前已经有大量的 8/16/32 位嵌入式微控制器在应用中,网络化是提高生产效率和产品质量、减少人力资源主要途径,如工业过程控制、数字机床、电力系统、电网安全、电网设备监测、石油化工系统。就传统的工业控制产品而言,低端型采用的往往是 8 位单片机。但是随着技术的发展,32/64 位的处理器逐渐成为工业控制设备的核心,在未来几年内必将获得长足的发展。

2. 交通管理

在车辆导航、流量控制、信息监测与汽车服务方面,嵌入式系统技术已经获得了广泛的应用,内嵌 GPS 模块, GSM 模块的移动定位终端已经在各种运输行业获得了成功的使用。目前 GPS 设备已经从尖端产品进入了普通百姓的家庭,只需要几千元,就可以随时随地确定位置。

3. 信息家电

信息家电将被作为嵌入式系统最大的应用领域,其中冰箱、空调等的网络化、智能化将引领人们的生活步入一个崭新的空间。即使你不在家里,也可以通过电话线、网络进行远程控制。在这些设备中,嵌入式系统将大有用武之地。

4. 家庭智能管理系统

水、电、煤气表的远程自动抄表,安全防火、防盗系统,其中嵌有的专用控制芯片将代替传统的人工检查,并实现更高、更准确和更安全的性能。目前在服务领域,如远程点菜器等已经体现了嵌入式系统的优势。

5. POS 网络及电子商务

公共交通无接触智能卡 (Contactless Smartcard, CSC) 发行系统,公共电话卡发行系统,自动售货机,各种智能 ATM 终端将全面走入人们的生活,到时手持一卡就可以行遍天下。

6. 环境工程与自然

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。在很多环境恶劣,地况复杂的地区,嵌入式系统将实现无人监测。

7. 机器人

嵌入式芯片的发展将使机器人在微型化、高智能方面优势更加明显,同时会大幅度降低机器人的价格,使其在工业领域和服务领域获得更广泛的应用。

这些应用中,可以着重投入控制方面的应用。就远程家电控制而言,除了开发出支持 TCP/IP 的嵌入式系统之外,家电产品控制协议也需要制订和统一,这需要家电生产厂家来做。同理,所有基于网络的远程控制器件都需要与嵌入式系统之间实现接口,然后再由嵌入式系统来控制并通过网络实现控制。所以,开发和探讨嵌入式系统有着十分重要的意义。

1.4 嵌入式处理器

嵌入式处理器是嵌入式系统的核心,是控制、辅助系统运行的硬件单元。范围极其广阔,从最初的4位处理器,目前仍在大规模应用的8位单片机,到最新的受到广泛青睐的32/64位嵌入式CPU。

而目前世界上具有嵌入式功能特点的处理器已经超过1000种,流行体系结构包括MCU、MPU等30多个系列。鉴于嵌入式系统广阔的发展前景,很多半导体制造商都大规模生产嵌入式处理器,并且公司自主设计处理器也已经成为未来嵌入式领域的一大趋势,其中从单片机、DSP到FPGA有着各式各样的品种,速度越来越快,性能越来越强,价格也越低。目前嵌入式处理器的寻址空间可以从64KB到16MB,处理速度最快可以达到2000 MIPS,封装从8个引脚到144个引脚不等。

根据其现状,嵌入式处理器可以分成下面几类。

1. 嵌入式微处理器

嵌入式微处理器(Micro Processor Unit, MPU)是由通用计算机中的CPU演变而来的。它的特征是具有32位以上的处理器,具有较高的性能,当然其价格也相应较高。但与计算机处理器不同的是,在实际嵌入式应用中,只保留和嵌入式应用紧密相关的功能硬件,去除其他的冗余功能部分,这样就以最低的功耗和资源实现嵌入式应用的特殊要求。和工业控制计算机相比,嵌入式微处理器具有体积小、质量轻、成本低、可靠性高的优点。目前主要的嵌入式处理器类型有Am186/88、386EX、SC-400、Power PC、68000、MIPS、ARM/StrongARM系列等。

2. 嵌入式微控制器

嵌入式微控制器(Microcontroller Unit, MCU)的典型代表是单片机,从20世纪70年代末单片机出现到今天,虽然已经经过了30多年的历史,但这种8位的电子器件目前在嵌入式设备中仍然有着极其广泛的应用。单片机芯片内部集成ROM/EPROM、RAM、总线、总线逻辑、定时/计数器、看门狗、I/O、串行口、脉宽调制输出、A/D、D/A、Flash RAM、EEPROM等各种必要功能和外部设备(简称外设)。与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。微控制器的片上外设资源一般比较丰富,适合于控制,因此称微控制器。

由于MCU低廉的价格,优良的功能,所以拥有的品种和数量最多,比较有代表性的包括8051、MCS-251、MCS-96/196/296、P51XA、C166/167、68K系列以及MCU 8XC930/931、C540、C541,并且支持I2C、CAN-Bus、LCD及众多专用MCU和兼容系列。目前MCU约占嵌入式系统70%的市场份额。近来Atmel出产的Avr单片机由于其集成了FPGA等器件,所以具有很高的性价比,势必将推动单片机获得更大的发展。

3. 嵌入式DSP

嵌入式DSP(Embedded Digital Signal Processor, EDSP)是专门用于信号处理方面的处理器,其在系统结构和指令算法方面进行了特殊设计,具有很高的编译效率和指令的执行速度。在数字滤波、FFT、谱分析等各种仪器上DSP获得了大规模的应用。

DSP的理论算法在20世纪70年代就已经出现,但是由于专门的DSP处理器还未出现,所以这

种理论算法也只能通过 MPU 等分立元件实现。MPU 较低的处理速度无法满足 DSP 的算法要求,其应用领域仅仅局限于一些尖端的高科技领域。随着大规模集成电路技术的发展,1982 年世界上诞生了首枚 DSP 芯片。其运算速度比 MPU 快了几十倍,在语音合成和编码解码器中得到了广泛应用。至 20 世纪 80 年代中期,随着 CMOS 技术的进步与发展,第二代基于 CMOS 工艺的 DSP 芯片应运而生,其存储容量和运算速度都得到成倍提高,成为语音处理、图像硬件处理技术的基础。到 20 世纪 80 年代后期,DSP 的运算速度进一步提高,应用领域也从上述范围扩大到了通信和计算机方面。20 世纪 90 年代后,DSP 发展到了第五代产品,集成度更高,使用范围也更加广阔。

目前最为广泛应用的是 TI 的 TMS320C2000/C5000 系列,另外如 Intel 的 MCS-296 和 Siemens 的 TriCore 也有各自的应用范围。

4. 片上系统

片上系统(System on Chip, SoC)是追求产品系统最大包容的集成器件,是目前嵌入式应用领域的热门话题之一。SoC 最大的特点是成功实现了软/硬件无缝结合,直接在处理器片内嵌入操作系统的代码模块。而且 SoC 具有极高的综合性,在一个硅片内部运用 VHDL 等硬件描述语言,实现一个复杂的系统。用户不需要再像传统的系统设计一样,绘制庞大复杂的电路板,一点点地连接焊制,只需要使用精确的语言,综合时序设计直接在器件库中调用各种通用处理器的标准,然后通过仿真之后就可以直接交付芯片厂商进行生产。由于绝大部分系统构件都是在系统内部,整个系统就特别简洁,不仅减小了系统的体积和功耗,而且提高了系统的可靠性,提高了设计生产效率。

由于 SoC 往往是专用的,所以大部分都不为用户所知,比较典型的 SoC 产品是 Philips 的 Smart XA。少数通用系列如 Siemens 的 TriCore, Motorola 的 M-Core,某些 ARM 系列器件, Echelon 和 Motorola 联合研制的 Neuron 芯片等。

预计不久的将来,一些大的芯片公司将通过推出成熟的、能占领多数市场的 SoC 芯片,一举击退竞争者。SoC 芯片也将在语音、图像、影视、网络及系统逻辑等应用领域中发挥重要作用。

1.5 嵌入式操作系统

嵌入式操作系统是一种支持嵌入式系统应用的操作系统软件,它是嵌入式系统(包括硬/软件系统)极为重要的组成部分,通常包括与硬件相关的底层驱动程序、系统内核、设备驱动接口、通信协议、图形界面、标准化浏览器等。嵌入式操作系统具有通用操作系统的基本特点,如能够有效管理越来越复杂的系统资源;能够把硬件虚拟化,使得开发人员从繁忙的驱动程序移植和维护中解脱出来;能够提供库函数、驱动程序、工具集,以及应用程序。与通用操作系统相比较,嵌入式操作系统在系统实时高效性、硬件的相关依赖性、软件固化化及应用的专用性等方面具有较为突出的特点。

1.5.1 嵌入式操作系统的种类

一般情况下,嵌入式操作系统可以分为两类:一类是面向控制、通信等领域的实时操作系统,如 Windriver 公司的 VxWorks、ISI 的 pSOS、QNX 系统软件公司的 QNX、ATI 的 Nucleus 等;另一类是面向消费电子产品的非实时操作系统,这类产品包括个人数字助理(PDA)、移动电话、机顶盒、电子书、WebPhone 等。

1. 非实时操作系统

早期的嵌入式系统中没有操作系统的概念,程序员编写嵌入式程序通常直接面对裸机及裸设备。在这种情况下,通常把嵌入式程序分成两部分,即前台程序和后台程序。前台程序通过中段来处理事件,其结构一般为无限循环;后台程序则掌管整个嵌入式系统软/硬件资源的分配、管理以及任务的调度,是一个系统管理调度程序。这就是通常所说的前/后台系统。一般情况下,后台程序也叫任务级程序,前台程序也叫事件处理级程序。在程序运行时,后台程序检查每个任务是否具备运行条件,通过一定的调度算法来完成相应的操作。对于实时性要求特别严格的操作通常由中断

来完成,仅在中断服务程序中标记事件的发生,不再做任何工作就退出中断,经过后台程序的调度,转由前台程序完成事件的处理,这样就不会造成在中断服务程序中处理费时的事件而影响后续和其他中断。

实际上,前/后台系统的实时性比预计的要差。这是因为前/后台系统认为所有的任务具有相同的优先级别,即是平等的,而且任务的执行又是通过 FIFO 队列排队,因而对那些实时性要求高的任务不可能立刻得到处理。另外,由于前台程序是一个无限循环的结构,一旦在这个循环体中正在处理的任務崩溃,使得整个任务队列中的其他任务得不到机会被处理,从而造成整个系统的崩溃。由于这类系统结构简单,几乎不需要 RAM/ROM 的额外开销,因而在简单的嵌入式应用中被广泛使用。

2. 实时操作系统

实时操作系统是指能在确定的时间内执行其功能并对外部的异步事件做出响应的计算机系统。其操作的正确性不仅依赖于逻辑设计的正确程度,而且与这些操作进行的时间有关。“在确定的时间内”是该定义的核心。也就是说,实时系统是对响应时间有严格要求的。

实时系统对逻辑和时序的要求非常严格,如果逻辑和时序出现偏差将会引起严重后果。实时系统有两种类型:软实时系统和硬实时系统。软实时系统仅要求事件响应是实时的,并不要求限定某一任务必须在多长时间内完成;而在硬实时系统中,不仅要求任务响应要实时,而且要求在规定的时间内完成事件的处理。通常,大多数实时系统是两者的结合。实时应用软件的设计一般比非实时应用软件的设计困难。实时系统的技术关键是如何保证系统的实时性。

实时多任务操作系统是指具有实时性,能支持实时控制系统工作的操作系统。其首要任务是调度一切可利用的资源完成实时控制任务,其次才着眼于提高计算机系统的使用效率,重要特点是要满足对时间的限制和要求。实时操作系统具有如下功能:任务管理(多任务和基于优先级的任务调度)、任务间同步和通信(信号量和邮箱等)、存储器优化管理(含 ROM 的管理)、实时时钟服务、中断管理服务。实时操作系统具有如下特点:规模小,中断被屏蔽的时间很短,中断处理时间短,任务切换很快。

实时操作系统可分为可抢占型和不可抢占型两类。对于基于优先级的系统而言,可抢占型实时操作系统是指内核可以抢占正在运行任务的 CPU 使用权并将使用权交给进入就绪态的优先级更高的任务,是内核抢了 CPU 控制权让别的任务运行。不可抢占型实时操作系统使用某种算法并决定让某个任务运行后,就把 CPU 的控制权完全交给了该任务,直到它主动将 CPU 控制权还回来。中断由中断服务程序来处理,可以激活一个休眠态的任务,使之进入就绪态;而这个进入就绪态的任务还不能运行,一直要等到当前运行的任务主动交出 CPU 的控制权。使用这种实时操作系统的实时性比不使用实时操作系统的系统性能好,其实时性取决于最长任务的执行时间。不可抢占型实时操作系统的缺点也恰恰是这一点,如果最长任务的执行时间不能确定,系统的实时性就不能确定。

可抢占型实时操作系统的实时性好,优先级高的任务只要具备了运行的条件,或者说进入了就绪态,就可以立即运行。也就是说,除了优先级最高的任务,其他任务在运行过程中都可能随时被比它优先级高的任务中断,让后者运行。通过这种方式的任務调度保证了系统的实时性,但是,如果任务之间抢占 CPU 控制权处理不好,会产生系统崩溃、死机等严重后果。

1.5.2 几种典型的嵌入式操作系统介绍

嵌入式操作系统种类繁多,但大体上可分为商用型和免费性两种。目前商用型的操作系统主要有 VxWorks、Windows CE、Palm OS、QNS 和 LYNX 等。它们的优点是功能稳定、可靠,有完善的技术支持和售后服务,而且提供了如图形用户界面和网络支持等高端嵌入式系统要求的许多高级的功能;缺点是价格昂贵且源代码封闭,这就大大影响了开发者的积极性。目前免费性的操作系统主要有 Linux 和 $\mu\text{C}/\text{OS-II}$,它们在价格方面具有很大的优势。比如嵌入式 Linux 操作系统以价格低廉、