



普通高等教育“十二五”电子信息类规划教材

嵌入式系统

主 编 张军朝
副主编 高保禄 杨晓峰



机械工业出版社
CHINA MACHINE PRESS

普通高等教育“十二五”电子信息类规划教材

嵌入式系统

主 编 张军朝

副主编 高保禄 杨晓峰

参 编 赵荣香 方 昀 李文婷 田 华

主 审 陈俊杰



机械工业出版社

本书介绍了嵌入式系统的基础知识、嵌入式语言基础、嵌入式系统应用设备、嵌入式硬件与固件的设计、开发步骤和案例,以及嵌入式系统的应用等。第1章和第2章介绍了嵌入式系统的基本知识。第3章介绍了嵌入式的语言基础。第4~8章介绍了有关嵌入式硬件与固件设计的进阶知识,对于嵌入式系统的实践工程师来说,这些知识是非常有帮助的。第9~12章说明了嵌入式系统设计的应用环境及案例,该章的内容对实践工程师和项目经理都是有益的。每章内容讲述得比较详细,并使用简明的文字说明以及表格、图例来阐明相应概念。每章的末尾均有总结,概括了该章的知识。为了方便学生在实验室也能进行相关的实验,本书还在第13章提供了实验指导。本书还介绍了微控制器与ARM处理器,列举了贴近实际的最新嵌入式设计案例。

本书从组件技术的视角出发,讲述了嵌入式系统的基本原理和技术。全书每一章涵盖一个专题,包括与嵌入式系统设计相关的若干主要内容,如指令系统、CPU、开发平台、程序设计与分析、进程和操作系统等。

本书特别适合作为计算机、软件工程、电子信息、通信工程、物联网工程、自动化、电气工程及其自动化、机电一体化、仪器仪表及相关专业高年级本科生和研究生的教材,也适合相关的工程技术人员参考。

图书在版编目(CIP)数据

嵌入式系统/张军朝主编. —北京:机械工业出版社, 2015. 7

普通高等教育“十二五”电子信息类规划教材

ISBN 978-7-111-50616-4

I. ①嵌… II. ①张… 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136906号

机械工业出版社(北京市百万庄大街22号 邮政编码100037)

策划编辑:王雅新 责任编辑:王雅新 路乙达

版式设计:霍永明 责任校对:陈越

封面设计:张静 责任印制:李洋

北京宝昌彩色印刷有限公司印刷

2015年9月第1版第1次印刷

184mm×260mm·20.75印张·563千字

标准书号:ISBN 978-7-111-50616-4

定价:45.00元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

服务咨询热线:010-88379833

机工官网:www.cmpbook.com

读者购书热线:010-88379649

机工官博:weibo.com/cmp1952

教育服务网:www.cmpedu.com

封面无防伪标均为盗版

金书网:www.golden-book.com

前 言

在信息技术飞速发展的今天,随着物联网的兴起,嵌入式系统迎来了前所未有的发展契机,其研究和应用范围涉及各行各业,遍及千家万户。嵌入式系统已成为人们重点关注和学习的热点。中国作为世界上最大的嵌入式技术市场,嵌入式技术的研究和应用也十分广泛,如智能建筑、智慧城市、数字矿井、工业控制、军事应用、物流管理、医疗电子等。嵌入式系统软件、硬件设计人才,系统应用开发人才,综合性人才的需求又进一步带动了嵌入式系统的开发与研究。因此研究嵌入式技术,掌握嵌入式系统原理、设计方法与应用具有极其重要的意义。

利用嵌入式系统已经成熟的平台和产品,与应用传感单元相结合,扩展物联和感知的支持能力,发掘某种领域物联网应用。从作为物联网重要技术组成的嵌入式系统的视角来看待物联网有助于深刻、全面地理解其本质。第一,物联网的核心仍然是互联网,是在互联网基础上的延伸和扩展的网络;第二,其用户端延伸和扩展到了任何物品与物品之间,进行信息交换和通信,必须具备嵌入式系统构建的智能终端。因此,物联网系统是通过射频识别(RFID)、红外感应器、全球定位系统、激光扫描器等信息传感设备,按约定的协议,把任何物品与互联网相连接,进行信息交换和通信的系统架构。

物联网不仅提供了传感器的连接,其本身也具有智能处理的能力,能够对物体实施智能控制,这就是嵌入式系统能做到的。物联网将传感器和智能处理相结合,利用云计算、模式识别等各种智能技术,扩充其应用领域。并从传感器获得的海量信息中分析、加工和处理出有意义的数据,以适应不同用户的不同需求,发现新的应用领域和应用模式。

截至2015年,我国已有370所高校设置了物联网工程专业。数以万计的读者急需一本嵌入式系统的参考书,伴随自己进入物联网应用的开发殿堂。为适应社会发展的需求,市面上也出现了一些新编的基于物联网发展的嵌入式系统教材,但都偏重于理论知识和一些验证性实验,忽视了软件基础,对于应用开发环境,更是很少涉足。本书是一本理论联系实际具有实用价值的教材,一方面满足高校本科和研究生教学需求,另一方面满足物联网嵌入式系统应用开发人员工作参考需求。

本书注重软硬件结合、理论与实践并重的原则,将嵌入式系统和物联网应用开发有机地结合起来,以ARM 9嵌入式处理器和Ubuntu 10.10桌面应用Linux操作系统为主线,详细地介绍了嵌入式操作系统和软硬件应用开发环境与工具,并结合IAR和Qt开发环境和工具,对嵌入式系统开发流程做了详尽的介绍,内容包括嵌入式系统概述、ARM 9处理器和架构、ARM 9处理器的内存管理、MMU地址映射、协处理器CP15、ARM 9异常处理、Cortex-A15介绍;嵌入式C/汇编语言基础、伪指令、ARM汇编语言程序设计、嵌入式C语言程序设计、ARM汇编语言与C/C++的混合编程;嵌入式Linux概述、ARM-Linux boot Loader内核映像载入、内核数据结构初始化、外设初始化、内核执行init进程、Shell启动、ARM-Linux内存管理单元(MMU)、ARM-Linux内存管理机制、进程虚拟空间管理、ARM-Linux进程管理和调度、ARM-Linux模块机制、ARM-

Linux 中断管理、ARM-Linux 文件系统；嵌入式设备概述、LED、键盘、ADC/DAC、RS232、RS485、触摸屏、LCD、SD、IC 卡、RFID、Zigbee、WiFi、蓝牙、GPRS；传感器的定义、分类、特性、热电阻式传感器、热电偶式传感器、辐射式温度计、应变式压力传感器、压电式压力传感器、电容式压力传感器、霍尔式压力传感器；嵌入式接口概述、GPIO 接口、SPI 总线、I²C 总线、I²S 总线；嵌入式设备驱动设计概述、字符设备驱动设计、块设备驱动设计、网络设备驱动、FLASH 设备驱动、Android 系统驱动；交叉开发环境概述、IAR EWARM8.0 应用、Keil4 应用、嵌入式 Linux 软件开发环境、模拟器、仿真器与调试；嵌入式图形界面编程概述、Qt 应用编程、MiniGUI 应用编程；Android 应用开发等方面。本书最后给出了一个可以贯穿教材的嵌入式系统和物联网应用有机结合的工程框架结构，并对配套的软硬件开发平台进行了较为详尽的介绍。

本书每章均配有内容小结和习题，以及既可以统一练习又可以化整为零分配到每一章练习的工程案例，同时配有免费电子课件，是教师首选的授课教材，也是学生自学的大纲指南，更是开发者首选的参考手册。

作者从事嵌入式系统应用开发 10 余年，主持开发的嵌入式应用系统有：基于 Zigbee 和 GPRS 的路灯照明调光节能控制系统、大型建筑工地太阳能 3G 无线视频监控系统、大型流域和城市防洪预警会商系统、城市火灾预警和消防装备全生命周期管理系统。其中基于 Zigbee 和 GPRS 的路灯照明调光节能控制系统已在太原市滨河东路景观照明系统工程、太原市汾河公园照明工程、长风商务区景观照明工程、汾东商务区路灯照明工程、江苏宜兴团氿公园景观照明工程、山东曹县路灯照明工程、河南中牟县路灯照明工程中推广应用。

本书共 13 章，1~12 章建议授课学时为 48 学时，每章 4 个学时。第 13 章实验练习时间为 8 个学时，总计 56 个学时。

张军朝担任本书主编并编写了第 1（1、2、4 节）、3、12 章；高保禄担任副主编并编写了第 2、11 章；杨晓峰担任副主编并编写了第 8、9 章；方昀编写了第 1（3 节）、4 章；李文婷编写了第 5（10~17 节）、6 章；赵荣香编写了第 5（1~9 节）、10 章；田华编写了第 7 章；第 13 章由北京中软国际教育科技有限公司提供。本书由太原理工大学陈俊杰教授主审。

本书在编写过程中得到了计算机专业教学指导委员会委员、太原理工大学陈俊杰教授，电气工程及其自动化专业教学指导委员会委员、太原理工大学宋建成教授，太原理工大学萧宝瑾教授、田慕琴教授、马春燕教授、许春雨副教授、曲兵妮副教授、吝伶俐副教授、陈燕副教授、任鸿秋副教授、耿蒲龙老师、高云广老师、郑丽君老师、张建花老师，北京中软国际教育科技有限公司田荣华经理，山西太原天地方圆电子科技有限公司赵荣香高工、植耀汉工程师自始至终的支持和帮助。太原理工大学姚宇硕士、段蛟龙硕士、施苗苗硕士、耿泽新硕士、郭俊硕士、于亚运硕士、饶元淇学士、范超群学士在编写和校对过程中也做了大量的工作。在此一并致以衷心的感谢！

编者力求将理论和实践相结合，科研和教学相结合，工程和教学相结合，硬件和软件相结合，先进和实用相结合，编写出高质量、高水平的教材，但由于水平有限，书中错误和不当之处在所难免，敬请读者谅解和指正。来函敬请至：zhangjunchao@tyut.edu.cn。

张军朝

于太原理工大学 国交楼

目 录

前言

第1章 概述..... 1

1.1 嵌入式系统简介..... 1

1.1.1 嵌入式系统的历史发展..... 1

1.1.2 嵌入式系统的定义..... 2

1.1.3 嵌入式系统的分类..... 3

1.1.4 嵌入式系统的特点..... 3

1.1.5 嵌入式系统的组成..... 4

1.1.6 嵌入式系统的应用..... 6

1.1.7 嵌入式系统的发展趋势..... 7

1.1.8 嵌入式系统的学习..... 7

1.2 嵌入式微处理器..... 8

1.2.1 处理器的基本组成..... 8

1.2.2 嵌入式微处理器的特点..... 9

1.2.3 两种处理器架构..... 9

1.2.4 嵌入式微处理器的分类..... 10

1.2.5 嵌入式微处理器的技术指标..... 12

1.2.6 嵌入式微处理器的选型..... 13

1.2.7 嵌入式微处理器的发展方向..... 13

1.3 嵌入式操作系统..... 14

1.3.1 嵌入式操作系统的概念..... 14

1.3.2 嵌入式操作系统的分类..... 14

1.3.3 嵌入式操作系统的特性..... 15

1.3.4 经典的嵌入式操作系统..... 16

1.4 本书结构..... 18

1.5 小结..... 19

1.6 复习思考题..... 19

第2章 处理器和架构..... 20

2.1 ARM 处理器概述..... 20

2.2 ARM9 处理器架构..... 21

2.2.1 ARM9 处理器简介..... 21

2.2.2 ARM9 系列技术特点..... 21

2.2.3 ARM9 结构及特点..... 22

2.2.4 ARM9 与 ARM7 的比较..... 23

2.3 ARM9 处理器的内存管理..... 24

2.3.1 内存管理单元 (MMU) 概述..... 24

2.3.2 MMU 地址映射的实现..... 24

2.3.3 协处理器 CP15..... 28

2.3.4 设置 MMU..... 30

2.4 ARM9 异常处理..... 31

2.5 Cortex-A15 介绍..... 32

2.6 小结..... 33

2.7 复习思考题..... 33

第3章 嵌入式语言基础..... 34

3.1 伪指令..... 34

3.1.1 符号定义伪指令..... 34

3.1.2 数据定义伪指令..... 35

3.1.3 汇编控制伪指令..... 37

3.1.4 宏指令..... 38

3.1.5 其他常用的伪指令..... 39

3.2 ARM 汇编语言..... 43

3.2.1 ARM 汇编语言常用符号..... 43

3.2.2 ARM 汇编语言表达式和运算符..... 44

3.2.3 ARM 汇编语言程序结构..... 45

3.2.4 ARM 汇编语言的子程序调用..... 46

3.2.5 ARM 指令的寻址方式..... 46

3.2.6 ARM 指令简介..... 48

3.2.7 Thumb 指令简介..... 54

3.3 嵌入式 C 语言..... 57

3.3.1 C 语言简介..... 57

3.3.2 简单 C 语言程序的构成..... 58

3.4 ARM 汇编语言与 C/C++..... 59

的混合编程..... 59

3.4.1 C 程序调用汇编程序..... 60

3.4.2 汇编程序调用 C 程序..... 60

3.4.3 实例分析..... 61

3.5 小结..... 63

3.6 复习思考题..... 63

第4章 嵌入式 Linux 操作系统基础..... 65

4.1 嵌入式 Linux 概述..... 65

4.1.1 常见的嵌入式 Linux 系统..... 65

4.1.2 嵌入式 Linux 的版本控制..... 66

4.1.3 嵌入式 Linux 的优势..... 67

4.2 嵌入式 Linux 操作系统启动..... 67

4.2.1 使用 bootloader 将内核映像载入..... 68

4.2.2 内核数据结构初始化..... 68

4.2.3 外设初始化	69	5.10.3 接触式 IC 卡	113
4.2.4 内核执行 init 进程	69	5.10.4 IC 卡的优点	114
4.2.5 Shell 的启动	70	5.11 RFID	114
4.3 ARM-Linux 内存管理	70	5.11.1 RFID 的基本组成	114
4.3.1 内存管理单元 (MMU)	70	5.11.2 分类	115
4.3.2 ARM-Linux 的存储管理机制	71	5.11.3 RFID 基本原理	115
4.3.3 ARM-Linux 对进程虚拟 空间的管理	74	5.11.4 RFID 的特性	116
4.4 ARM-Linux 进程管理和调度	77	5.12 Zigbee	117
4.4.1 Linux 下进程的结构	77	5.12.1 Zigbee 技术基础	118
4.4.2 Linux 的进程调度	82	5.12.2 Zigbee 的节点	118
4.5 ARM-Linux 的模块机制	83	5.12.3 Zigbee 技术的特点	118
4.5.1 Linux 模块概述	83	5.13 WiFi	119
4.5.2 模块的加载	83	5.14 蓝牙	120
4.5.3 模块的卸载	84	5.14.1 射频特性	120
4.6 ARM-Linux 的中断管理	84	5.14.2 TDMA 结构	120
4.7 嵌入式 Linux 文件系统	88	5.14.3 使用跳频技术	120
4.7.1 Linux 文件系统简介	88	5.14.4 蓝牙设备的组网	121
4.7.2 虚拟文件系统	89	5.14.5 软件的层次结构	121
4.7.3 常见的嵌入式文件系统	89	5.15 GPRS	122
4.8 小结	94	5.16 小结	124
4.9 复习思考题	94	5.17 复习思考题	125
第 5 章 嵌入式设备	95	第 6 章 传感器	126
5.1 嵌入式设备概述	95	6.1 传感器概述	126
5.2 LED	95	6.1.1 传感器的定义	126
5.3 键盘	96	6.1.2 传感器的分类	126
5.3.1 传统键盘的介绍	96	6.1.3 传感器的特性	127
5.3.2 新型键盘的硬件和软件 实现原理	96	6.2 传感器原理及应用	129
5.4 ADC/DAC	98	6.2.1 传感器的组成	129
5.4.1 ADC	98	6.2.2 传感器的应用	129
5.4.2 DAC	101	6.3 验证语音、笔迹和指纹等 特征的传感技术	132
5.5 RS-232	103	6.3.1 AVVS 型验证语音和 笔迹的装置	132
5.6 RS-485	104	6.3.2 语音验证系统	133
5.7 触摸屏	105	6.3.3 用于身份验证的压电笔	133
5.8 LCD	106	6.3.4 签字验证书写台	134
5.8.1 LCD 原理综述	106	6.3.5 指纹自动识别仪	134
5.8.2 LCD 的驱动方式	107	6.4 小结	134
5.9 SD 卡	110	6.5 复习思考题	134
5.9.1 SD 模式	111	第 7 章 嵌入式接口	135
5.9.2 SPI 模式	111	7.1 嵌入式接口概述	135
5.10 IC 卡	112	7.1.1 接口功能	135
5.10.1 IC 卡的定义	112	7.1.2 接口结构	135
5.10.2 IC 卡的分类	112	7.1.3 接口数据传输方式	136

7.1.4 接口设计的一般方法	136	8.3.3 块设备驱动举例	174
7.2 嵌入式总线	137	8.3.4 与块设备相关的数据结构	178
7.2.1 总线及总线分类	137	8.3.5 改变块驱动程序设计	184
7.2.2 嵌入式系统总线的层次结构	137	8.4 其他设备驱动	185
7.2.3 衡量总线的参数	138	8.4.1 网络设备驱动	185
7.3 GPIO 接口	138	8.4.2 Flash 设备驱动	186
7.3.1 GPIO 概述	139	8.4.3 Android 系统驱动	187
7.3.2 GPIO 的引脚描述	139	8.5 小结	187
7.3.3 GPIO 的寄存器描述	139	8.6 复习思考题	188
7.3.4 GPIO 的构成	140	第9章 嵌入式开发环境	189
7.3.5 GPIO 的工作模式	141	9.1 交叉开发环境概述	189
7.3.6 GPIO 的优点	141	9.1.1 宿主机与目标板的连接方式	189
7.4 SPI 总线	142	9.1.2 文件传输	190
7.4.1 SPI 总线概述	142	9.1.3 宿主机环境	191
7.4.2 SPI 总线的工作原理	142	9.1.4 交叉编译工具链	191
7.4.3 SPI 总线的内部结构	143	9.1.5 嵌入式软件开发流程	192
7.4.4 SPI 总线的特点	144	9.2 IAR EWARM 集成开发环境	193
7.4.5 SPI 总线的应用	144	9.2.1 IAR EWARM 创建应用程序	194
7.5 I ² C 总线	145	9.2.2 测试应用程序	201
7.5.1 I ² C 总线概述	146	9.3 Keil 集成开发环境	202
7.5.2 I ² C 总线的工作过程	146	9.3.1 创建应用程序	202
7.5.3 I ² C 总线的内部结构	147	9.3.2 测试应用程序	209
7.6 I ² S 总线	148	9.4 嵌入式系统开发流程	212
7.6.1 I ² S 总线概述	149	9.5 小结	214
7.6.2 I ² S 总线的工作过程	149	9.6 复习思考题	214
7.6.3 I ² S 总线的内部结构	150	第10章 图形界面编程	215
7.7 小结	151	10.1 嵌入式图形编程概述	215
7.8 复习思考题	151	10.1.1 Linux 图形开发基础	215
第8章 嵌入式设备驱动设计	152	10.1.2 嵌入式 Linux 图形用户 界面简介	220
8.1 嵌入式设备驱动设计概述	152	10.2 Qt 编程基础	221
8.1.1 设备的分类及特点	153	10.2.1 Qt/Embedded 概述	222
8.1.2 设备文件及设备号	153	10.2.2 创建 Qt/Embedded 编程环境	224
8.1.3 与其他模块的关系	154	10.2.3 Qt Creator 介绍	232
8.1.4 如何使用模块机制	154	10.3 Qt 编程实例	234
8.1.5 sysfs 虚拟文件系统	156	10.3.1 实现功能	234
8.1.6 Linux 内核设备模型	158	10.3.2 实现原理	234
8.1.7 Linux 内核驱动模型	160	10.3.3 实现过程	234
8.2 字符设备驱动设计	164	10.4 MiniGUI 编程基础	243
8.2.1 设备驱动举例	164	10.4.1 事件驱动编程	243
8.2.2 相关的数据结构以及操作介绍	167	10.4.2 MiniGUI 的三种运行模式	243
8.2.3 增加字符设备驱动的功能	171	10.4.3 运行模式适用范围	244
8.3 块设备驱动设计	173	10.4.4 MiniGUI 进程	244
8.3.1 块设备的 I/O 特点	173	10.5 MiniGUI 编程实例	245
8.3.2 块设备的相关概念	174		

10.5.1 头文件	246	12.2 给排水监测控制子系统	282
10.5.2 程序入口点	247	12.2.1 给排水系统主要架构	282
10.5.3 加入层	247	12.2.2 给排水系统实现功能	283
10.5.4 创建和显示主窗口	247	12.2.3 给排水系统改造方案	284
10.5.5 进入消息循环	248	12.2.4 传感器选型	286
10.5.6 窗口过程函数	250	12.3 供配电监测控制子系统	289
10.5.7 屏幕输出	250	12.3.1 供配电系统改造整体方案	289
10.5.8 程序的退出	250	12.3.2 改造后供配电系统 的主要功能	289
10.6 小结	250	12.3.3 供配电系统具体改造方案	290
10.7 复习思考题	251	12.4 系统整体架构	294
第11章 Android 应用	252	12.5 软件应用系统	295
11.1 Android 概述	252	12.5.1 能耗监测数据采集功能模块	295
11.1.1 Android 背景介绍	252	12.5.2 能耗监测数据分析功能模块	295
11.1.2 Android 的系统构架	253	12.5.3 能耗监测智能决策功能模块	295
11.1.3 Android 应用程序框架	255	12.5.4 能耗监测远程控制功能模块	295
11.1.4 OMS 介绍	256	12.5.5 系统扩展功能	295
11.2 Android 开发环境搭建	257	12.6 工程施工内容	295
11.2.1 相关软件的下载与安装	257	12.6.1 通信网络构建及设备安装	295
11.2.2 虚拟设备的创建和使用	264	12.6.2 系统供电	296
11.3 Android 应用开发	267	12.6.3 接地	296
11.4 Android 应用开发扩展	272	12.7 系统特点	296
11.5 小结	277	12.8 效益分析	296
11.6 复习思考题	277	第13章 嵌入式系统实验指导	297
第12章 楼宇供能智能控制系统		13.1 实验大纲	297
应用案例	278	13.2 物联网嵌入式网关平台搭建	299
12.1 中央空调监测控制子系统	278	13.3 CC2530 基础实验——LED 闪烁	303
12.1.1 现场考察及用户意见总结	278	13.4 CC2530 基础实验——按键轮询	308
12.1.2 技术方案	279	13.5 CC2530 基础实验——LED、电机 控制实验	312
12.1.3 设备选型	280	参考文献	321
12.1.4 中央空调监测控制子系统 结构图	282		

第 1 章 概 述



本章主要内容

- 嵌入式系统简介
- 嵌入式微处理器
- 嵌入式操作系统

本章主要从嵌入式系统简介、嵌入式微处理器、嵌入式操作系统三个方面对嵌入式系统进行了简单介绍。

1.1 嵌入式系统简介

物联网的兴起，为嵌入式系统提供了新的发展契机，在物联网兴起的大潮中，嵌入式系统飞速发展，迅速应用到各行各业，遍及千家万户，尤其是在智慧地球、感知中国、智能城市、平安小区、智能家居等与人们日常生活密切相关的活动领域中得到长足的发展。嵌入式系统软硬件应用开发人才需求的急剧上升，使得研究嵌入式技术，掌握嵌入式系统原理、设计方法与应用具有重要的现实意义。

本节主要从历史发展、定义、分类、特点、系统组成、应用领域、发展趋势等方面对嵌入式系统做简要介绍。

1.1.1 嵌入式系统的历史发展

从 20 世纪 70 年代单片机的出现到各式各样的嵌入式微处理器、微控制器的大规模应用，嵌入式系统已经有近 40 年的发展历史。

嵌入式系统的出现起源于单片机。70 年代单片机的出现，使得汽车、家电、工业机器、通信装置以及成千上万种产品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，但是这时的应用只是使用 8 位的芯片，执行一些单线程的程序，还谈不上“系统”的概念。

最早的单片机是 Intel 公司的 8048，它出现在 1976 年。与此同时 Motorola 公司推出了 68HC05，Zilog 公司推出了 Z80 系列，这些早期的单片机均含有 256B 的 RAM、4K 的 ROM、4 个 8 位并口、1 个全双工串行口、2 个 16 位定时器。之后在 20 世纪 80 年代初，Intel 又进一步完善了 8048，在它的基础上研制成功了 8051，这在单片机的历史上是值得纪念的一页，迄今为止，51 系列的单片机仍然是最为成功的单片机芯片，在各种产品中都有着非常广泛的应用。

从 80 年代早期开始，嵌入式系统的程序员开始用商业级的“操作系统”编写嵌入式应用软件，这使得其可以获取更短的开发周期，更低开发资金和更高的开发效率，“嵌入式系统”真正地出现了。确切地说，这个时候的操作系统是一个实时核，这个实时核包含了许多传统操作系统的特征，包括任务管理、任务间通信、同步与相互排斥、中断支持、内存管理等功能。

嵌入式操作系统中比较著名的有 Ready System 公司的 VRTX、Integrated System Incorporation

(ISI) 的 PSOS 和 Wind River Systems 公司的 VxWorks、QNX 公司的 QNX 等。这些操作系统都具有嵌入式的典型特点, 即它们均采用占先式的调度, 响应的时间很短, 任务执行的时间可以确定; 系统内核很小, 具有可裁剪, 可扩充和可移植性, 可以移植到各种处理器上; 较强的实时和可靠性, 适合嵌入式应用。这些嵌入式实时多任务操作系统的出现, 使得应用开发人员从小范围的开发解放出来, 同时也促使嵌入式有了更为广阔的应用空间。

20 世纪 90 年代以后, 随着对实时性要求的提高, 软件规模不断上升, 实时核逐渐发展为实时多任务操作系统 (RTOS), 并作为一种软件平台逐步成为目前国际嵌入式系统的主流。这时候更多的公司看到了嵌入式系统的广阔发展前景, 开始大力发展自己的嵌入式操作系统。除了上面的几家老牌公司所开发的嵌入式操作系统以外, 还出现了 Palm OS, WinCE, 嵌入式 Linux, Lynx, Nucleux, 以及国内的 Hopen, Delta Os 等嵌入式操作系统。随着嵌入式技术的发展前景日益广阔, 相信会有更多的嵌入式操作系统软件出现。

1.1.2 嵌入式系统的定义

嵌入式系统 (Embedded system), 是一种完全嵌入受控器件内部, 为特定应用而设计的专用计算机系统, 根据英国电器工程师协会 (U. K. Institution of Electrical Engineer) 的定义, 嵌入式系统为控制、监视或辅助设备、机器或用于工厂运作的设备。与个人计算机这样的通用计算机系统不同, 嵌入式系统通常执行的是带有特定要求的预先定义的任务。由于嵌入式系统只针对一项特殊的任务, 设计人员能够对它进行优化, 减小尺寸降低成本。

嵌入式系统的核心是由一个或几个预先编好程序以用来执行少数几项任务的微处理器或者单片机组成。与通用计算机能够运行用户选择的软件不同, 嵌入式系统上的软件通常是暂时不变的, 所以经常称为“固件”。

1. 从不同角度定义

1) 技术角度: 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、功耗等严格要求的专用计算机系统, 是将应用程序、操作系统和计算机硬件集成在一起的系统。

2) 系统角度: 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。嵌入式反映了这些系统通常是更大系统 (被称之为嵌入的系统) 的一个完整子系统。嵌入的系统可以包含多个嵌入式系统。

3) 广义定义: 任何一个非计算机的计算系统。

2. 从不同的组织结构定义

(1) IEEE 的定义

嵌入式系统是用于“控制、监视或者辅助操作的机器、设备或装置”。嵌入式系统通常是执行特定功能的设备, 其核心是嵌入式微处理器。嵌入式系统有严格的执行时序和稳定性要求, 是全自动的操作循环。

(2) 微机学会的定义

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、片级。

1) 系统级: 各种类型的工控机, 如 STD、PC 工业控制机等, 其特点为配置操作系统, 系统软件及硬件资源丰富, 具有很好的人机界面。

2) 板级: 各种类型带 CPU 的主板及 OEM 产品, 早期单板机, 如 Z80 系列等。

3) 片级: 各种以单片机、DSP、微处理器为核心的产品。如以 8051 为 CPU 的系列产品、ARM 系列产品等。

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,管理系统软件和硬件的能力有限,在实现多任务复杂功能时,往往困难重重,甚至无法实现。

狭义上讲,强调使用嵌入式微处理器构成的独立系统,具有自己的操作系统,具有特定功能,用于特定场合的系统都可称为嵌入式系统。本书所指的嵌入式系统是指狭义上的嵌入式系统。

1.1.3 嵌入式系统的分类

1. 嵌入式系统的分类

不同的分类标准,有不同的分类结果。按照处理机的字长划分,嵌入式系统主要有以下几种:

(1) 典型的 8 位微处理系统 MCS-51 系列的单片机是 Intel 开发最成功的单片微处理器,是在低端嵌入式系统中使用最多的微处理器,在多方面得到应用。一般基于 8051 的系统无操作系统,软件的开发主要基于裸机开发,以汇编语言为主,少量用 C51。

(2) 典型的 16 位微处理系统 MCS-96 系列单片机、80186 嵌入式处理机、TI 的 16 位 DSP 芯片,支持操作系统。

(3) 典型的 32 位微处理系统 ARM 是应用较广泛的 32 位微处理器,实际上该系列的 MCU 芯片很多,但大都以 ARM 微内核为核心并集成不同的接口,软件系统基于嵌入式操作系统,软硬件资源丰富;32 位 DSP 的应用也很广泛。

2. 32 位嵌入式控制系统

采用 32 位 RISC 嵌入式微处理器和实时操作系统组成的嵌入式控制系统近几年应用越来越广,与传统基于单片机的控制系统和基于 PC 的控制方式相比,具有以下突出优点:

(1) 性能方面 采用 32 位 RISC 结构微处理器,主频从 30MHz 到 624MHz 或以上,处理能力大大超出单片机系统,接近 PC 机水平,但体积更小,能够真正地“嵌入”到设备中。

(2) 实时性方面 嵌入式控制器内嵌实时操作系统(RTOS),提供多任务支持,能完全保证控制系统的强实时性。

(3) 人机交互方面 高效的 GUI 嵌入式控制器可支持大屏幕的液晶显示器,提供功能强大的图形用户界面,输入方法多种多样。

(4) 系统升级方面 嵌入式控制器可为控制系统专门设计,其功能专一,成本较低,而开放的用户程序接口(API)保证了系统能够快速升级和更新;对硬件的适应性更好,具有良好的移植性,能支持尽量多的硬件平台;占有更少的硬件资源,例如,占用存储器十几 k 到几 kB,可靠性更高;提供强大的网络功能,支持 TCP/IP 协议及其他协议,协议栈可裁剪,例如,设计成可裁剪的微内核结构和模块化结构。

1.1.4 嵌入式系统的特点

嵌入式系统的硬件和软件必须根据具体的应用任务,以功耗、成本、体积、可靠性、处理能力等为指标来进行选择。嵌入式系统的核心是系统软件和应用软件,由于存储空间有限,因而要求软件代码紧凑、可靠,且对实时性有严格要求。

从构成上看,嵌入式系统是集软、硬件于一体,可独立进行工作的计算机系统;从外观上看,嵌入式系统对象是一个“可编程”的电子“器件”;从功能上看,它是对目标系统(宿主对

象) 进行控制, 使其智能化的控制器; 从用户和开发人员的不同角度来看, 与普通计算机系统相比较, 嵌入式系统具有以下特点:

1. 专用性强

嵌入式系统的个性化很强, 其中的软件系统和硬件的结合非常紧密, 一般要针对硬件进行系统的移植, 同时针对不同的任务, 往往需要对系统进行较大修改, 程序的编译下载要和系统相结合, 这种修改和通用软件的“升级”是完全不同的概念。

2. 体积小量化

嵌入式计算机把通用计算机系统中许多由板卡完成的任务集成在芯片内部, 从而有利于实现小型化, 方便将嵌入式系统嵌入目标系统中。

3. 实时性好

实时性的本质是任务处理所花费时间的可预测性, 即任务需要在规定的时限内完成。任务执行的时间可以根据系统的软、硬件的信息而进行确定性的预测。也就是说, 如果硬件可以做这件工作, 那么基于实时操作系统的软件将可以确定性的做这件工作。

4. 可裁剪性好

从嵌入式系统专用性的特点来看, 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, 力争在同样的硅片面积上实现更高的性能, 这样才能在具体应用中更具竞争力。

5. 可靠性高

由于有些嵌入式系统所承担的计算任务涉及被控产品的关键质量, 人身设备安全, 甚至国家机密等重大事务, 且有些嵌入式系统的宿主对象工作在无人值守的场合, 如在危险性高的工业环境和恶劣的野外环境中的监控装置里。所以, 与普通系统相比较, 嵌入式系统对可靠性的要求更高。

6. 功耗低

有许多嵌入式系统的宿主对象是一些小型应用系统, 如移动电话、MP3、数码相机等, 这些设备不可能配置交流电源或容量较大的电源, 因此低功耗也是嵌入式系统的一大特点。

7. 不可垄断性

嵌入式系统的工业基础是以应用为中心的“芯片”设计和面向应用的软件产品开发。硬件平台多, 软件资源丰富, 无法垄断。

1.1.5 嵌入式系统的组成

1. 嵌入式系统的结构及应用

嵌入式系统的软、硬件框架结构和嵌入式系统硬件描述, 如图 1-1 及图 1-2 所示。



图 1-1 嵌入式系统的软、硬件框架结构

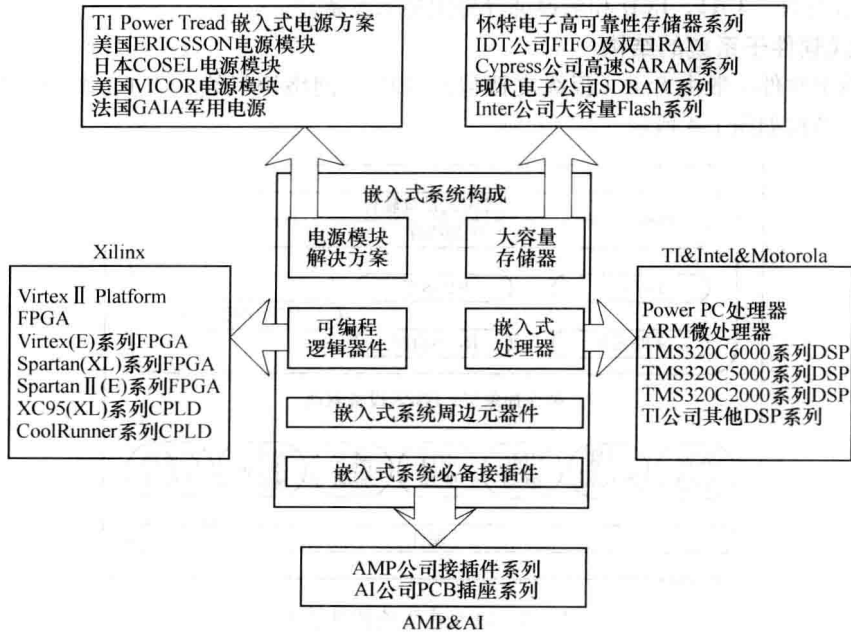


图 1-2 嵌入式系统硬件描述

2. 嵌入式外围设备

嵌入式外围设备是指在嵌入式硬件系统中，除了嵌入式处理器之外的完成存储、通信、I/O、调试等功能的其他部件，如图 1-3 所示，其分类如下：

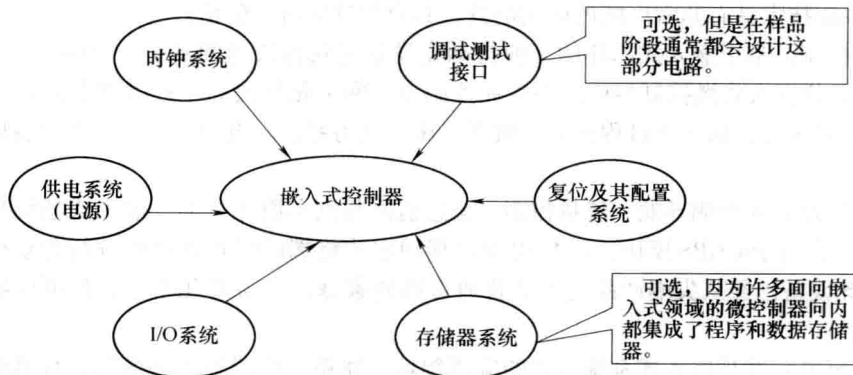


图 1-3 嵌入式系统硬件组织

(1) 存储器类型 静态易失型存储器 (RAM/ SRAM)，动态存储器 (DRAM)，非易失型存储器 (ROM、EPROM、EEPROM、FLASH)。其中，FLASH (闪存) 由于可擦写次数多，存储速度快，容量大及价格便宜等优点在嵌入式领域得到广泛应用。

(2) 接口类型 目前存在的所有接口在嵌入式系统中都有其广泛的应用，但是以下几种接口的应用最为广泛，包括 RS-232 接口 (串口)，IrDA (红外)，SPI (串行设备接口)，I²C，USB，Ethernet，CAN，蓝牙，A/D，D/A，GPRS，GPS，I²S 和普通并口。

(3) 显示类型 CRT, LCD 和触摸屏等外围显示设备。

3. 嵌入式软件子系统的结构

嵌入式系统软件一般由 BSP (板级支持包)、EOS、网络协议线、GUI (图形用户界面) 应用软件组成, 结构如图 1-4 所示。

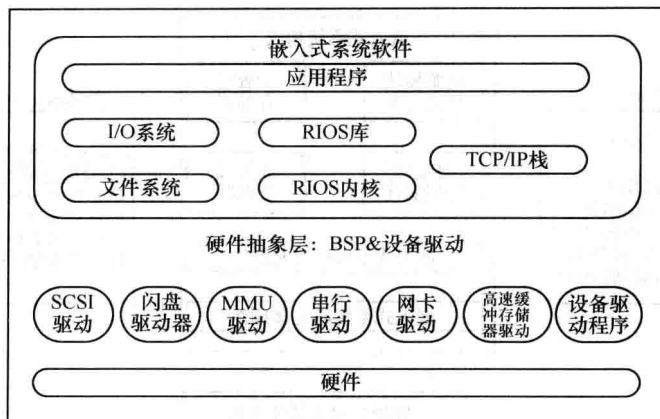


图 1-4 嵌入式系统软件结构

1.1.6 嵌入式系统的应用

嵌入式系统的应用十分广泛, 涉及工业生产、日常生活、工业控制、航空航天等多个领域, 而且随着电子技术和计算机软件技术的发展, 不仅在这些领域中的应用越来越深入, 而且在其他传统的非信息类设备中也逐渐显现出其用武之地。

嵌入式系统技术具有非常广阔的应用前景, 其应用领域可以包括:

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 位、16 位、32 位嵌入式微控制器应用在工业控制中。网络化是提高生产效率和产品质量、减少人力资源的主要途径, 如工业过程控制、数控机床、电力系统、电网安全、电网设备监测、石油化工系统等。

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, 就可以随时随地找到你的位置。

3) 信息家电: 这是嵌入式系统最大的应用领域, 冰箱、空调等的网络化、智能化将引领人们步入一个崭新的生活空间。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、燃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。

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、高智能方面的优势更加明显, 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, 可以着重于在控制方面的应用。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。同样的道理, 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。

1.1.7 嵌入式系统的发展趋势

信息时代、数字时代的到来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, 未来嵌入式系统的发展趋势有:

1. 系统工程化

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。目前很多厂商已经充分考虑到这一点, 在主推系统的同时, 将开发环境也作为重点推广。比如三星在推广 ARM7、ARM9 芯片的同时还提供开发板和板级支持包 (BSP), 而 WindowCE 在主推系统时也提供 Embedded VC++ 作为开发工具, 还有 VxWorks 的 Tonado 开发环境, DeltaOS 的 Limda 编译环境等。

2. 功能多样化

随着因特网技术的成熟和带宽日益提高, 以往设备的功能不再单一, 结构更加复杂, 如电话、手机等。这就要求芯片设计厂商在芯片上集成更多的功能, 满足应用功能的升级。硬件方面采用更强大的嵌入式处理器, 如 32 位、64 位 RISC 芯片或信号处理器 DSP 增强处理能力, 同时增加功能接口 (如 USB) 和扩展总线类型 (如 CAN BUS) 以加强对多媒体、图形等的处理, 逐步实施片上系统 (SOC) 的概念。软件方面采用实时多任务编程技术和交叉开发工具技术来控制功能复杂性, 简化应用程序设计, 保障软件质量并缩短开发周期。

3. 网络化

未来嵌入式设备为了适应网络发展的要求, 必然要在硬件上提供各种网络通信接口。新一代的嵌入式处理器已经开始内嵌网络接口, 支持 TCP/IP 协议。IEEE1394、USB、CAN、Bluetooth 或 IrDA 通信接口中的一种或几种, 同时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, 实现随时随地上网。

4. 精简系统内核、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, 利用最少的资源实现最适当的功能, 选用最佳的编程模型并不断改进算法, 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. 交互友好化

嵌入式设备能与用户亲密接触, 最重要的因素就是它能提供友好的用户界面、图像界面以及灵活的控制方式。另外, 嵌入式设备对专业知识要求低, 甚至不需要嵌入式的知识就能让人们很快很容易地掌握嵌入式产品的使用方法, 因此未来嵌入式系统需要更高的可交互性和高度抽象化。

1.1.8 嵌入式系统的学习

嵌入式系统的应用非常广泛, 因此对于计算机专业、电子信息专业、自动化专业以及机电一体化

化等专业的学生以及需要掌握嵌入式的人员来说,学习嵌入式系统及其开发应用是非常重要的。

要学好嵌入式系统及其开发应用,掌握嵌入式技术基础是关键。技术基础决定了一个人学习知识、掌握技能的能力。嵌入式技术融合了具体应用系统技术、嵌入式处理器技术、系统芯片 SOC 设计制造技术、应用电子技术和嵌入式操作系统及应用软件技术,具有极高的系统集成性,可以满足不断增长的信息处理技术对嵌入式系统设计的要求。

学习与嵌入式系统相关的基础知识,主要包括基本硬件知识,如嵌入式处理器及接口电路等,并要求至少掌握一种嵌入式处理器的体系结构,至少了解一种操作系统。对于应用编程,要求编程人员掌握 C、C++ 语言及汇编语言程序(至少要会 C 语言)和交叉编译,对处理器的体系结构、组织结构、指令系统、编程模式、一般应用编程要有一定的了解,在此基础上必须在实际工程实践中掌握一定的实际项目开发的技能。

其次,对于嵌入式系统的学习,必须要有一个较好的嵌入式系统开发平台和开发环境。功能全面的开发平台一方面为学习提供了良好的开发环境,另一方面开发平台本身也是一般的典型实际应用系统。在教学平台上开发一些基础例程和典型实际应用例程,对于初学者和实际工程应用者来说也是非常必要的。

学习嵌入式系统必须对基本内容有深入地了解。在处理器指令系统、应用编程学习的基础上,重要的是加强外围功能接口应用的学习,主要是人机接口及通信接口。

嵌入式操作系统也是嵌入式系统学习的重要部分,在此基础上才能进行各种设备驱动应用程序的开发。

本书注重使学生掌握嵌入式系统及其开发应用,从嵌入式系统所涉及的基础知识讲起,从硬件到软件,从内部结构到外围接口,从指令系统到程序设计,直到嵌入式系统的设计。

1.2 嵌入式微处理器

嵌入式微处理器是由通用计算机中的 CPU 演变而来的。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、重量轻、成本低、可靠性高的优点。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。

1.2.1 处理器的基本组成

无论是嵌入式系统还是通用计算机系统所使用的处理器(CPU)其基本工作原理和结构都是一样的,都包括运算器、控制器、寄存器(组)等三个主要部分。运算器主要完成对二进制信息的算术运算、逻辑运算和各种移位操作。算术运算主要包括定点加、减、乘、除运算;逻辑运算主要有逻辑与、逻辑或、逻辑异或和逻辑非等操作;移位操作主要完成逻辑左移和右移、算术左移和右移,以及其他一些移位操作。控制器是处理器的指挥和控制中心,它把运算器、存储器、I/O 设备等联系成一个有机的系统。简单来说,控制器的工作就是周而复始地完成取指令、分析指令、执行指令的工作。寄存器(组)是为了避免频繁地访问存储器而在 CPU 内部提供的暂时存放参加运算的数据和中间结果的单元。寄存器有通用寄存器和专用寄存器之分,它们的区别在于存储数据的性质不同,起到的作用也不同。在运算过程中及运算结束后,运算器中还要设置相应的寄存器来记录运算的一些特征情况,如是否溢出、结果的符号位、结果是否为零等。不同体系 CPU 寄存器的组织、名称、功能和存取方法都不一样,这也是汇编语言缺少通用性的原