

高等院校教材

# 微控制器原理与应用

徐安 陈耀 方春华 编著

白英彩 主审



科学出版社  
[www.sciencep.com](http://www.sciencep.com)

高等院校教材

# 微控制器原理与应用

徐安 陈耀 方春华 编著

白英彩 主审

科学出版社

北京

## 内 容 简 介

微控制器 (MCU) 即单片机, 是作为嵌入式系统的控制核心而设计、制造和应用的。本书从微控制器这一共性出发, 讲述了 MCU 的原理、结构、指令、汇编语言程序设计、基本模块和系统扩展、I<sup>2</sup>C、USB、CAN 等总线接口和 C 语言程序设计; 并根据 MCU 个性化、多功能化、网络化的普遍发展趋势, 介绍了多项 MCU 功能的增强。

本书选择了 LPC900(80C51 内核)、M68HC08 和 PIC 等系列中 3 种最有代表性的微控制器为典型, 较详细地介绍了它们的原理、性能和应用。

本书可作为普通高等院校电子信息、自动化、电气工程、机电一体化等专业本科生的教材, 也可供相关专业的工程技术人员参考。

### 图书在版编目 (CIP) 数据

微控制器原理与应用/徐安, 陈耀, 方春华编著. —北京: 科学出版社, 2006

(高等院校教材)

ISBN 7-03-017811-4

I. 微… II. ①徐…②陈…③方… III. 微控制器-高等学校-教材  
IV. TP332.3

中国版本图书馆 CIP 数据核字 (2006) 第 093332 号

责任编辑: 匡敏 余江 潘继敏/责任校对: 张琪

责任印制: 张克忠/封面设计: 陈敬

科学出版社出版

北京东黄城根北街 16 号

邮政编码: 100717

<http://www.sciencep.com>

源海印刷有限责任公司印刷

科学出版社发行 各地新华书店经销

\*

2006 年 9 月第 一 版 开本: B5(720×1000)

2006 年 9 月第一次印刷 印张: 31

印数: 1—3 000 字数: 586 000

定价: 39.00 元

(如有印装质量问题, 我社负责调换(路通))

## 前 言

微控制器(MCU)即单片机,是作为嵌入式系统的控制核心而设计、制造和应用的,这是微控制器万变不离其宗的共性。无论从存储空间、封装、引脚的安排,还是各种模块,直到每一项功能的实现,都深刻地体现着这一共性。

微控制器技术飞速发展,MCU世界五彩缤纷,新型号、新技术日新月异,只有从共性出发,理论联系实际,打下学以致用用的坚实基础,才能够明确方向,具备在计算机知识的海洋中,根据今后的各种不同需要继续深入学习与应用的能力,既不至于无所适从,又不拘泥于某一特定机型。

本书尝试从微控制器的共性出发,讲述了MCU的原理、结构、指令、汇编语言程序设计、基本模块和系统扩展、I<sup>2</sup>C、USB、CAN等总线接口和C语言程序设计;并根据MCU个性化、多功能化、网络化的普遍发展趋势,介绍了多项MCU功能的增强。

本书选择了LPC900(80C51内核)、M68HC08和PIC等系列中3种最有代表性的微控制器为典型,较详细地介绍了它们的原理、性能和应用,以利于各专业进行教学活动时量体裁衣。

课堂教学的时间极其有限,其首要任务应该是激发学生的学习热情,帮助学生树立基本的但必须是正确的概念。建议根据课堂精讲、课下勤学的教学原则,在启发、互动、形象的课堂氛围中,在主讲教师引导下,越过本课程、本章节的最初门槛;课下则举一反三,进一步探讨和实验印证。例如,指令的寻址方式是一个教学重点,也是个难点,建议首先对书中归纳的7种微控制器常用寻址方法进行深入的分析和理解,然后再用各型号微控制器的具体指令系统进行对比,通过实验,进一步领会微控制器指令寻址方式是如何解决快捷和广域的矛盾,为嵌入式系统服务的。

实验和实践对学习和掌握微控制器的设计与应用极为重要,本书限于篇幅未能更多涉及,将在配套教材《微控制器原理与应用实验教程》中另行介绍。

多年来,白英彩、陈章龙、张友德、涂时亮、何立民、胡汉才、邵贝贝、马忠梅、周航慈、阮家栋、周立功、陈明计、谢瑞和、刘慧银、金炯泰、李学海、白中英等教授在微控制器教学的园地里辛勤耕耘,本书多处引用了他们的研究成果,谨在此表示衷心的感谢。

担任本书主审的上海交通大学白英彩教授认真审阅了全书,提出了指导性的建议和中肯的意见;研究生丁才云、张振富、吴丽蓉、郭凯杰、陆圣轩和吴霄等为图文制作输入和程序校核付出了辛勤劳动,在此谨致以深切的谢意。

本书由徐安执笔主编,陈耀参加第9章、第10章的编写,方春华对全书各章提出了修改意见。电子课件正在制作中,主要用于辅助任课教师教学。

本书为同济大学教材、学术著作出版基金委员会资助的同济大学“十五”规划教材。

限于编者水平,书中错误和不妥之处在所难免,热切期待着读者的批评指正。  
邮件地址:xuanb@online.sh.cn。

编者

2006年7月

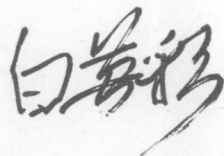
## 序 言

微控制器即单片微型计算机。由于在设备控制、过程控制和数字控制中表现出的优异特性和强大功能,微控制器已成为信息社会不可缺少的重要支柱,无时无刻不存在于当今的数字化时代。

我国各高校普遍开设了“微控制器(单片机)原理与应用”这一课程。《微控制器原理与应用》一书以新颖独特的视角面对浩瀚的微控制器世界,从“微控制器是作为嵌入式系统的控制核心而设计、制造和应用的”这一共性出发,选择3种最有代表性的微控制器为典型加以叙述,由浅入深,循序渐进;使读者能够触类旁通,举一反三,对微控制器的原理与应用有更本质的理解。

作者徐安教授自文革结束,师从王祖泽教授攻读研究生以来,一直在微控制器应用的科研和教学领域辛勤工作,成绩斐然;以其二十多年教学经验和诸多科研成果为基础,传承清华大学、上海交通大学、同济大学的严谨学风,编写了这部教材。该书可以为教师课堂精讲提供基础,为学生课下勤学拓展空间,并有利于各相关专业量体裁衣。

作为清华大学电机系的学长,我高兴地向电子与信息工程、自动控制、电气工程和机电一体化等有关专业的教师和学生推荐这本结构清晰、语言简明、共性凝炼、内容翔实的教材,更期待着徐安教授为我国微控制器教学做出更大的贡献。



2006年7月

# 目 录

序言	1
前言	2
第 1 章 概论	1
1.1 微控制器概述	1
1.2 MCU 和 MPU	1
1.3 微控制器和嵌入式系统	2
1.4 微控制器的应用开发	3
1.4.1 硬件和软件的协同设计	3
1.4.2 微控制器系统的设计方法	3
1.5 微控制器的历史与发展	5
习题与思考题	6
第 2 章 微控制器的结构、时序和工作方式	7
2.1 微控制器的结构	7
2.1.1 微控制器常用封装	7
2.1.2 微控制器的体系结构	8
2.1.3 微控制器的组成	9
2.2 微控制器时序	13
2.2.1 时钟周期、机器周期和指令周期	14
2.2.2 取指令和执行指令时序	15
2.3 微控制器工作方式	16
2.3.1 复位方式	16
2.3.2 程序执行方式	17
2.3.3 节电工作方式	17
2.3.4 程序的写入、校验和擦除	18
2.4 典型微控制器的特点	19
2.4.1 P89LPC936 的特点	19
2.4.2 MC68HC908GP32 的特点	21
2.4.3 PIC16F877 的特点	25
习题与思考题	29

<b>第3章 微控制器的指令系统</b> .....	30
3.1 指令的表示与字节 .....	30
3.1.1 指令表示形式 .....	30
3.1.2 指令的字节 .....	30
3.2 指令的寻址方式 .....	31
3.2.1 隐含寻址 .....	31
3.2.2 立即寻址 .....	31
3.2.3 直接寻址 .....	32
3.2.4 寄存器间接寻址 .....	32
3.2.5 变址寻址 .....	34
3.2.6 相对寻址 .....	37
3.2.7 位寻址 .....	38
3.3 指令的功能 .....	38
3.3.1 数据传送指令 .....	38
3.3.2 算术运算指令 .....	40
3.3.3 逻辑运算指令 .....	40
3.3.4 移位指令 .....	40
3.3.5 位操作指令 .....	40
3.3.6 控制转移指令 .....	40
3.3.7 其他指令 .....	40
3.4 典型微控制器的指令系统 .....	40
3.4.1 MCS-51 指令系统 .....	40
3.4.2 M68HC08 指令系统 .....	57
3.4.3 PIC 指令系统 .....	74
习题与思考题 .....	79
<b>第4章 汇编语言程序设计</b> .....	80
4.1 汇编语言的格式与构成 .....	80
4.1.1 汇编语言的格式 .....	80
4.1.2 伪指令 .....	81
4.1.3 源程序的汇编 .....	85
4.2 汇编语言程序设计方法 .....	85
4.2.1 分支结构 .....	86
4.2.2 循环结构 .....	86
4.2.3 子程序调用 .....	88
习题与思考题 .....	90



<b>第 5 章 微控制器的存储器、时钟和定时/计数器模块</b> .....	91
<b>5.1 存储器模块</b> .....	91
5.1.1 半导体存储器的种类和特点 .....	91
5.1.2 微控制器内部的存储器 .....	93
5.1.3 ICP、ISP、IAP 和监控 ROM .....	95
5.1.4 P89LPC936 的存储器模块 .....	96
5.1.5 MC68HC908GP32 的存储器模块 .....	105
5.1.6 PIC16F877 的存储器模块 .....	111
<b>5.2 时钟模块</b> .....	121
5.2.1 时钟模块的构成 .....	121
5.2.2 P89LPC936 的时钟模块 .....	122
5.2.3 MC68HC908GP32 的时钟模块 .....	125
5.2.4 PIC16F877 的时钟模块 .....	132
<b>5.3 定时器/计数器及其扩展</b> .....	135
5.3.1 定时/计数器模块的构成 .....	135
5.3.2 P89LPC936 的定时/计数器模块 .....	136
5.3.3 MC68HC908GP32 的定时/计数器模块 .....	144
5.3.4 PIC16F877 的定时/计数器模块 .....	147
<b>5.4 输入捕捉、输出比较与脉宽调制</b> .....	150
5.4.1 CCP 原理 .....	150
5.4.2 P89LPC936 的 CCP 功能 .....	151
5.4.3 MC68HC908GP32 的 CCP 功能 .....	162
5.4.4 PIC16F877 的 CCP 功能 .....	166
<b>习题与思考题</b> .....	176
<b>第 6 章 I/O 端口与数据传送</b> .....	178
<b>6.1 I/O 数据传送方式</b> .....	178
6.1.1 几种用以类比的交通方式 .....	178
6.1.2 几种 I/O 数据传送方式 .....	179
<b>6.2 I/O 端口编址与类型</b> .....	181
6.2.1 I/O 端口编址 .....	181
6.2.2 I/O 端口类型 .....	182
<b>6.3 并行 I/O 端口</b> .....	182
6.3.1 微控制器 P89LPC936 的 I/O 端口 .....	184
6.3.2 MC68HC908GP32 的 I/O 端口 .....	188
6.3.3 微控制器 PIC16F877 的 I/O 端口 .....	190

6.4	异步串行通信接口 SCI .....	190
6.4.1	串行通信基础 .....	191
6.4.2	RS-232C 标准 .....	197
6.4.3	LPC900 的 SCI 模块 .....	199
6.4.4	M68HC08 的 SCI 模块 .....	207
6.4.5	PIC16F877 的 SCI 模块 .....	217
6.5	同步串行端口 SPI .....	228
6.5.1	SPI 的工作原理与模块构成 .....	228
6.5.2	LPC900 的 SPI 接口 .....	233
6.5.3	M68HC08 的 SPI 接口 .....	239
6.5.4	PIC16F87X 的 SPI 接口 .....	243
6.6	I <sup>2</sup> C 接口 .....	247
6.6.1	I <sup>2</sup> C 结构与数据传输 .....	247
6.6.2	LPC900 的 I <sup>2</sup> C 接口 .....	258
6.6.3	PIC16F87X 的 I <sup>2</sup> C 接口 .....	268
6.7	USB 接口 .....	270
6.7.1	接口特点 .....	270
6.7.2	结构和数据流 .....	271
6.7.3	USB 交换的包格式 .....	276
6.7.4	接口芯片和微控制器 .....	279
6.7.5	USB 系统的典型应用 .....	284
6.8	CAN 接口 .....	285
6.8.1	特点 .....	285
6.8.2	数据传送 .....	286
6.8.3	物理层设计 .....	288
6.8.4	CAN 总线的通信芯片与微控制器 .....	288
	习题与思考题 .....	298
<b>第 7 章</b>	<b>复位、中断、监控与节电 .....</b>	<b>300</b>
7.1	复位 .....	300
7.1.1	复位的作用、效果和复位源 .....	300
7.1.2	LPC900 的复位功能 .....	301
7.1.3	M68HC08 的复位功能 .....	304
7.1.4	PIC16F87X 的复位功能 .....	306
7.2	中断 .....	307
7.2.1	微控制器的中断源、中断功能和中断过程 .....	307

7.2.2	P89LPC936 的中断系统	310
7.2.3	MC68HC908GP32 的中断系统	313
7.2.4	PIC16F877 的中断系统	319
7.3	键盘模块与键盘中断	328
7.3.1	键盘与键盘中断(KBD)原理	328
7.3.2	LPC900 的键盘中断功能	330
7.3.3	M68HC08 的键盘中断功能	330
7.4	监视定时器 WDT	333
7.4.1	监视定时器原理	333
7.4.2	LPC900 的 WDT	334
7.4.3	M68HC08 的 WDT	337
7.4.4	PIC16F87X 的 WDT	339
7.5	电源监控与节电模式	341
7.5.1	LPC900 的电源监控和节电模式	342
7.5.2	M68HC08 的节电模式	343
7.5.3	PIC16F87X 的节电模式	350
	习题与思考题	352
<b>第 8 章</b>	<b>数模转换与模数转换</b>	<b>353</b>
8.1	D/A 原理与技术性能	353
8.1.1	数模转换器技术性能	353
8.1.2	脉宽调制型数模转换	354
8.1.3	T 型电阻网络型数模转换	355
8.2	A/D 原理与技术性能	359
8.2.1	模数转换器技术性能	359
8.2.2	逐次比较型模数转换	362
8.2.3	输入捕获型模数转换	371
8.3	ADC 模块	372
8.3.1	P89LPC936 的 ADC 模块	372
8.3.2	MC68HC908GP32 的 ADC 模块	379
8.3.3	PIC16F877 的 ADC 模块	382
	习题与思考题	387
<b>第 9 章</b>	<b>微控制器的 C 语言程序设计</b>	<b>388</b>
9.1	C 语言与微控制器	388
9.1.1	微控制器的编程语言	388
9.1.2	C 语言编译器与程序开发过程	390

9.2	51 系列微控制器 C 语言程序设计基础	390
9.2.1	KEIL 编译器	390
9.2.2	C51 程序设计基础	390
9.2.3	P89LPC936 内部资源的 C 编程	394
9.3	Motorola 微控制器 C 语言程序设计基础	395
9.3.1	ICC08 编译器和 08C 语言	395
9.3.2	Motorola 08C 语言编程	399
9.4	Microchip 微控制器 C 语言程序设计基础	401
9.4.1	PICC 编译器和 PICC C 语言	401
9.4.2	PICC C 语言编程	405
	习题与思考题	408
<b>第 10 章</b>	<b>嵌入式实时操作系统</b>	<b>409</b>
10.1	嵌入式实时操作系统基础	409
10.1.1	基本概念	409
10.1.2	嵌入式实时操作系统技术特点	410
10.2	几种典型的嵌入式实时操作系统	418
10.2.1	嵌入式实时操作系统 RTX51	418
10.2.2	嵌入式实时操作系统 PICOS18	421
10.2.3	嵌入式实时操作系统 Small RTOS51	425
10.2.4	嵌入式实时操作系统 $\mu\text{C}/\text{OS}-\text{II}$	426
10.3	嵌入式实时操作系统的移植与应用	429
10.3.1	RTX51 在单片机控制的 GPS 接收板上的应用	429
10.3.2	占先式实时内核 $\mu\text{C}/\text{OS}-\text{II}$ 在 MC68HC908GP32 上的移植	431
	习题与思考题	433
<b>第 11 章</b>	<b>微控制器应用系统的研制</b>	<b>434</b>
11.1	微控制器应用系统的开发	434
11.1.1	开发工具	434
11.1.2	微控制器开发系统功能	435
11.1.3	应用 ISP 功能进行开发调试	437
11.2	微控制器应用系统研制过程	437
11.2.1	总体设计	437
11.2.2	硬件设计	439
11.2.3	软件设计	439
11.2.4	可靠性设计	442
11.3	应用实例 1——T-40 型机车速度表检验仪	444

---

11.3.1 配置与工作过程 .....	444
11.3.2 系统构成 .....	445
11.3.3 语音合成技术的应用 .....	446
11.4 应用实例 2——基于 Small RTOS51 的肠营养液输液系统 .....	449
11.4.1 系统设计 .....	450
11.4.2 使用 Small RTOS51 的优点 .....	452
11.5 应用实例 3——基于 Ethernet 和 CAN 的楼宇设备控制系统 .....	453
11.5.1 系统总体设计 .....	453
11.5.2 区域控制器 .....	454
11.5.3 现场控制器 .....	455
11.5.4 应用系统特点 .....	456
习题与思考题 .....	456
<b>参考文献</b> .....	458
<b>附录 A MCS-51 指令表</b> .....	459
<b>附录 B M68HC08 指令表</b> .....	465
<b>附录 C PIC16F87X 指令表</b> .....	478

## 第1章 概 论

### 1.1 微控制器概述

微控制器 (microcontroller unit, MCU) 是专为控制和检测而设计制造的微型计算机, 因为它完全作嵌入式应用, 所以又被称为嵌入式微控制器 (embedded microcontroller); 又因为它在一片集成电路芯片上集成了计算机的三大部分: 中央处理器 (CPU)、存储器 (RAM 和 ROM) 和输入输出端口 (I/O Port), 在我国又被广泛称作单片机 (single chip microcomputer)。飞速发展的微控制器, 功能越来越强大, 成本越来越低, 促使经典电子系统迅速进化为全盘智能化的现代电子系统。微控制器作为最典型、最广泛、最普及的现代电子系统的基本核心, 已成为电气工程师和各行各业的技术人员实现检测和控制技术目的的有力工具。

微控制器应用的实例已是举不胜举。从家用电器到航天飞机, 一切现代化技术都离不开微控制器, 可以毫不夸张地说, 微控制器已经广泛渗透到了国民经济各个领域, 已经无时无刻不在影响每个现代人的生活。

### 1.2 MCU 和 MPU

微计算机技术的发展, 形成了两大分支: 微控制器和微处理器。

微处理器 (microprocessor unit, MPU) 是计算机的核心部件, 计算机系统的主要用途是科学计算、数据处理、图像分析、数据库管理、人工智能、数字模拟和仿真等。

微控制器主要用于控制目的, 所构成的检测控制系统要求有实时、快速的外部响应, 能迅速采集到大量数据, 再做出逻辑判断与推理, 实现对被控制对象的参数调整与控制。

随着计算机科学与微电子技术的高速发展, 这两大分支都得到了突飞猛进的发展。

由于人类对海量数值运算的无限需求, 8 位、16 位的 MPU 都已经被淘汰, 32 位和 64 位机也以每两三年换代一次的速度发展。计算机系统速度越来越快, 功能越来越强, 而用户的操作则越来越简单。

微控制器的发展直接利用了 MPU 的成果, 也发展了 16 位、32 位及 64 位机

型,目标是高性能、高可靠性、低电压、低功耗、低噪声和低成本。在以 ARM 为代表的 32 位微控制器日益发展的同时,现阶段世界年产量以数十亿计的 8 位微控制器仍占据着重要地位。

为了满足多种多样检测控制对象的要求,构成各种专用控制器与多机控制系统,微控制器的发展还表现在接口及其性能上,例如,各种传感器接口、工业对象的电气接口、伺服驱动的功率接口、人机对话接口、通信网络接口等;接口的高速互用、中断处理;模/数、数/模转换的速度和精度,位操作和功率驱动,程序运行监控和信号实时处理等性能。

### 1.3 微控制器和嵌入式系统

近代电子计算机进入了微型机时代后,微型计算机在信息处理、逻辑分析、决策判断和输入输出管理等自动控制领域表现出无与伦比的优势,并以其可接受的小型体积和现场可靠性嵌入到各种对象体系中,构成了形形色色的自动化系统:机车、舰船、生产线、大型机床、微型机器人等。

面向检测控制对象、嵌入到应用系统中的计算机系统称为嵌入式系统。实时性是其主要特征,在可靠性、物理尺寸、重启动和故障恢复方面也有特殊要求,因而比通常的计算机应用设计更为复杂,涉及面也更为广泛。由于被嵌入对象的体系结构、应用环境要求不同,所以嵌入式系统也有各种类型。

微控制器是典型的嵌入式系统。它应嵌入式计算机系统应用的要求而生,并以嵌入式应用为唯一目的。微控制器从体系结构到指令系统都是按照嵌入式应用特点专门设计的,能最好地满足面向控制对象、应用系统的嵌入、现场的可靠运行及控制品质的各种要求。在其基本体系结构上,衍生出了能满足各种应用要求的兼容系统,成为发展最快、品种最多、数量最大的嵌入式系统。

从形式上可将嵌入式系统分为系统级、板级和器件级:

系统级为各种类型的工控机,包括通用机改装的工控机、各种总线方式的工控机或模块组成的工控机,如 PC104 等。

板级有各种类型的带 CPU 的主板及 OEM 产品。

器件级则以微控制器最为经典。其 IC 器件的体积小,现场运行环境的可靠性高,满足了许多小型对象的嵌入式应用要求。不少半导体厂商还以嵌入式应用为目标,将通用 CPU 改造成嵌入式微处理器 (embedded microprocessor unit, EMPU),形成了具有良好的开发环境与操作系统支持的器件级嵌入式系统,例如,由 80386 改造成的 386EX。

## 1.4 微控制器的应用开发

### 1.4.1 硬件和软件的协同设计

硬件和软件的协同设计在微控制器的应用开发中占有重要地位。

微控制器系统将硬件和软件结合起来,在与外界发生交互并受到时间约束的环境中,完成特定的功能或任务;作为大系统或大的电子设备中的一部分,在没有人工干预的情况下进行实时自动控制。其中,软件用以实现有关功能并使系统具有适应性和灵活性;硬件(处理器、ASIC、存储器等)用以满足性能乃至安全的需要。

微控制器系统的设计与硬件提供的支持(包括开发手段)及软件技术的发展紧密相关。应用先进的硬件设计和开发技术,不但可以获得所需的性能(如速度、电源、低成本和可靠性等),而且还能取代部分软件,完成特定的功能。软件的设计也离不开硬件的支持(如处理器、高级缓冲和并行能力等),高性能、多功能的硬件可以提高软件开发和设计效率,保证了软件质量。而软件设计技术和开发手段,也可以充分发挥硬件的作用,提高系统的性能。在保证系统性能的前提下,微控制器系统的设计要综合考虑硬件和软件的任务分工(包括考虑用硬件代替软件,或用软件置换硬件);硬件设计和软件设计的并行、交替和协同;硬件和软件的合成。

### 1.4.2 微控制器系统的设计方法

通常,设计微控制器系统的方法有以下三种:

#### 1. 以印制板计算机辅助设计软件和在线仿真器为主要工具

根据微控制器应用系统的功能要求,先划分功能模块,再对模块进行硬件和软件功能实现分配。

硬件设计主要是根据性能参数的要求,对各功能模块所需要的元器件进行选择 and 组合,选择的基本原则是选购性价比最高的通用元器件。从模块到系统找到相对优化的方案,画出电路原理图。硬件设计的关键是利用印制板(PCB)计算机辅助设计(CAD)软件对系统的元器件进行布局和布线,然后加工印制板、装配和硬件调试。

软件设计贯穿整个系统的设计过程,主要包括任务分析、资源分配、模块划分、流程设计和细化、编码调试等。软件设计的工作量主要集中在程序调试,在线仿真器(in-circuit emulator, ICE)对软件调试的作用非常重要。



## 2. 以嵌入式操作系统和电子设计自动化综合开发平台为主要工具

嵌入式操作系统 (embedded operating system, EOS) 和高性能的电子设计自动化 (EDA) 综合开发平台的推出, 为开发复杂的微控制器系统应用软件提供了底层支持和高效率开发平台。EOS 是一种功能强大、应用广泛的实时多任务系统软件, 与通用系统机中的 OS 相比, 内核短小精悍、开销小、实时性强和可靠性高, 还提供各种设备的驱动程序和 TCP/IP 协议支持。用户可以通过应用程序接口 (API) 调用的函数形式来实现各种资源管理, 用户程序可以在 EOS 的基础上开发并运行。

随着微电子工艺技术的发展, 硬件设计师可以利用各种通用的可编程半定制逻辑器件逐步把原先要通过印制板线路互连的若干标准逻辑器件设计制作成专用集成电路 (ASIC), 把印制板布局和布线的复杂性转换成半定制器件内配置的复杂性, 增加了可编程应用的灵活性; 大大减少了印制板的面积和接插件的数量, 使印制板上互连器件的线路、装配和调试费用越来越少, 降低了系统综合成本; 更重要的是降低了系统功耗, 提高了系统工作速度, 大大提高了系统的可靠性和安全性。

设计人员从过去选择和使用标准通用集成电路器件, 逐步转向利用各种 EDA 工具和标准的半定制逻辑器件, 如复杂可编程逻辑器件 (CPLD) 和现场可编程门阵列 (FPGA) 等, 自己设计和制作部分专用的集成电路器件, 然后把用户定义的集成电路 (customer specified integrated circuit, CSIC)、可编程外围器件、所选择的 ASIC 与微控制器在印制板上布局、布线, 最终构成系统。

## 3. 以 IP 内核库为设计基础, 软硬件协同设计单片系统

目前, 集成设计已进入单片系统 (system on a chip, SOC) 设计阶段, 并开始实用。单片系统设计从整个系统性能要求出发, 把微处理器、模型算法、芯片结构、外围器件各层次电路直至器件的设计紧密结合起来, 并通过建立在全新理念上的系统软件和硬件的协同设计, 在单个芯片上完成整个系统的功能。达到高密度、高速度、高性能、小体积和低电压等指标, 特别是低功耗要求。

Philips、Atmel 和 TI 等厂商就是利用 Intel MCS-51 的 IP (intellectual property) 内核模块, 开发出了各具特长的与 Intel MCS-51 兼容的微控制器。

常用的 IP 内核模块有各种不同的 CPU (32/64 位 CPU 或 8/16 位微控制器)、32/64 位 DSP、DRAM、RAM、E<sup>2</sup>PROM、Flash Memory、A/D、D/A、MPEG/JPEG、LJSB、PCI、标准接口、网络单元、编译器、编码/解码器和模拟器件模块等。具有丰富的 IP 内核模块库才能快速地设计 ASIC 和 SOC, 从而尽快占领市场。