

新世纪计算机课程系列精品教材

毛玉良 / 主 编
张赤斌 / 副主编

微机系统原理及应用

WEIJIXITONGYUANLIJIYINGYONG

(第2版)



东南大学出版社
Southeast University Press

新世纪计算机课程系列精品教材

微机系统原理及应用

(第2版)

主 编	毛玉良	
副主编	张赤斌	
参 编	田梦倩	王海燕
	许映秋	戴 敏

东南大学出版社
· 南 京 ·

内 容 提 要

本书基础部分讨论了一般性问题,其中包括计算机中的数、码和常用电路,微处理器系统的基本结构和运行原理,微处理器、存储器和 I/O 的重要概念以及它们之间的相互关系。应用部分以 MCS-51 微控制器为主,介绍了它的结构特点,存储器组织,指令系统及集成的定时、中断和串行通信等功能。

围绕 MCS-51 的总线及其时序,介绍了存储器和外围芯片的扩展技术,并强调与时序配合。在存储器扩展部分介绍了各种译码方法。在 I/O 扩展方面,重点介绍了典型接口的工作原理及其应用,本书对传统的外围接口芯片及其功能的介绍进行了仔细的取舍。

在以 MCS-51 为基础的汇编语言教学中,引入 ASM-51 宏汇编的规范,它支持多模块以及今后同 C51 高级语言的混合编程。为了帮助初学者理解和掌握,汇编语言程序示例都尽可能完整地列出(对应的 C51 的例程作为附录给出)。本书例题丰富、重点突出、循序渐进,力求通俗易懂,体系和内容安排来自于作者多年从事的微机课程教学的经验和体会。书中大量例题来源于作者多年从事科研和工程技术的积累,实用性强,术语规范。

本书可作为普通高等学校非计算机专业本科学生的教材,也可作为成人高等教育的培训教材,还可供广大科技人员自学参考。

图书在版编目(CIP)数据

微机系统原理及应用/毛玉良主编. —2 版. —南京:东南大学出版社,2012. 6
新世纪计算机课程系列精品教材
ISBN 978-7-5641-2809-8
I. ①微… II. ①毛… III. ①微型计算机—高等学校—教材 IV. ①TP36
中国版本图书馆 CIP 数据核字(2011)第 097823 号

微机系统原理及应用(第 2 版)

出版发行 东南大学出版社
出 版 人 江建中
社 址 南京市四牌楼 2 号
邮 编 210096

经 销 全国各地新华书店
印 刷 南京京新印刷厂
开 本 787 mm×1092 mm 1/16
印 张 21.75
字 数 568 千字
版 次 2006 年 8 月第 1 版 2012 年 6 月第 2 版
印 次 2012 年 6 月第 1 次印刷
书 号 ISBN 978-7-5641-2809-8
印 数 1—3000 册
定 价 43.00 元

(本社图书若有印装质量问题,请直接与营销部联系。电话:025-83791830)

第 2 版前言

本书是非计算机专业学生学习“微机原理及接口技术”的基础教材。主要是将 MCS-51 作为微控制器的一个典型系列,并围绕嵌入式系统的要求来阐述微机原理、技术及其应用。其出发点和编排与“MCS-51 单片机技术”类教材有明显的区别。

本书面向初学者的需要,内容安排上注重微机运行原理方面的基础知识,同时考虑了实用性、实践性教学环节的可操作性以及有关专业学时上的限制。本书的主要特点是:

(1) 内容编排更加符合实际教学过程。根据大多数学生的先修课程的实际,本书首先从硬件的角度概括介绍了计算机中的常用数制、码制和微机运行的一般技术原理;其次介绍 MCS-51 的体系、指令、汇编语言编程、集成数字部件及其应用;最后介绍基于总线的系统扩展、C51 编程和先进的 C8051F 系列微控制器的应用。所以本书非常适合初学微机原理与接口技术的学生,内容由浅入深,并结合丰富的实例,不断提高学生的应用能力。

现存的多数以 MCS-51 为基础的教材,其内容的展开方式非常接近于微处理器(或微控制器)生产厂家的官方资料,其权威性自然不容质疑。但是,除非另有相关的先修课程,教师必须重组内容,才能保证教学顺利进行。本书面向初学者,内容安排更加通俗易懂。不过笔者也提醒读者在掌握一定的基础知识后,工程应用还是应该多看原始文档。

(2) 内容新颖完整。介绍了 Intel 的 ASM-51 宏汇编,因为它支持多模块编程。本书汇编语言程序实例均按宏汇编格式编写,并尽可能完整。汇编语言子程序大多按照 C51 高级语言的调用约定来写,这为汇编与 C51 混合编程打下了基础。关于 C51 编程的介绍靠后,这样不会冲淡汇编语言的基础性,又使 C51 的实用技术介绍水到渠成。

对于学时较少的专业,可以选学 C51 高级语言编程,而汇编语言程序设计只作一般了解。在本书的这一版中,对于以汇编语言形式给出的主要程序示例,也给出了它们的 C51 版本(附录 D)。需要指出的是,本书对 C51 的介绍方法比较简洁,是假定读者已经具备 C 或 C++ 基础的。如果读者未学过 C 或 C++,或虽然学过但大部分已经遗忘,那么会有一定困难的。

(3) 强调学用一致。中断是本课程的难点,本书突出中断技术的应用教学。从多年工程实践中我们体会到,几乎没有一个嵌入式程序开发可以不使用中断。

中断编程要考虑的问题比较多,我们的办法是多给出一些中断服务程序的例子,这样初学者有例子可作参照。在实践性环节中,我们也是通过实例,加强对学生的基本功训练。

(4) 精选常用外围芯片讲解扩展技术。通过 TTL 电路构成输入缓冲,输出锁存,带联络信号的输入/输出等,强化了接口原理的概念和技术的介绍;而具体到 8255、8253 和 8250,则以原理、工作方式和典型应用为主,特别是已淘汰的 8250 以 16C2552 代替。至于这些芯片本身,随着 SOC 发展方向的到来,都是可取代的,或者被吸收到 MCU 中去。像 8255 的实际用量已经在大幅度下降,8250 已经停产,这是大势所趋。

(5) C8051F 系列微控制器的应用。相对于经典的 MCS-51 芯片,该系列微控制器的数字部件种类和数量大幅增加、系统时钟频率也得到很大提高,并由于模拟接口的引入,性能优异,因此应用日益普及。此外,它还在某种程度上,代表着 MCS-51 系列在嵌入式系统应用领域的发展方向,其芯片设计和开发手段也更趋先进。为此,笔者以一个具体型号为例,设计了一块实验电路,在实践的基础上,本书第 2 版给出了该电路原理、开发过程和程序示例,期待与读者分享学习心得和经验。

本书第 1 版由毛玉良副教授策划并任主编,张赤斌教授任副主编。许映秋教授、田梦倩副教授、王海燕、戴敏等老师参加了本书的编写。其中田梦倩老师编写了第 1 章;张赤斌老师编写了第 2 章以及第 7,9,10,14 章的部分内容;许映秋老师编写了第 4 章以及第 12,13 章的部分内容;戴敏老师编写了第 15 章;王海燕老师编写了第 1 版的第 17 章;其余部分都是由主编完成并通稿。本书第 2 版修正了第 1 版的不少错误和瑕疵。

本书第 2 版第 17 章是新写的,内容是关于 C8051F 系列微控制器的应用和开发。在新编内容部分,东南大学机械学院硕士研究生陆向军同学参与了试验电路板的设计制作、示例程序的调试和验证工作,在该章成文过程中也付出了辛勤的劳动。原 8088/8086 内容的介绍在新版中删掉了,因为从反馈的情况看,该章内容在选用本教材的单位中实际使用得不多。

由于计算机技术日新月异的发展,新技术不断涌现,尽管我们在概念的阐述上已经考虑到这些因素,但仍然可能有不当之处,加之时间仓促,难免有错误发生,敬请各位读者、专家和使用本教材的老师批评指正,以便再版时及时改正。

编 者

2011 年 7 月

于南京

目 录

0	绪论	(1)
1	计算机中的数和码	(4)
1.1	有限字长的二进制数	(4)
1.1.1	二进制数	(4)
1.1.2	数制及其表示方法	(5)
1.1.3	有限字长的二进制数	(5)
1.1.4	常用术语解释	(5)
1.2	十进制数到二进制数的转换	(6)
1.3	带符号二进制数的表示及其运算	(6)
1.3.1	带符号数的表示方法	(7)
1.3.2	补码到真值的转换	(8)
1.3.3	补码的运算	(8)
1.4	溢出及运算的有效性	(9)
1.4.1	无符号数与带符号数的溢出条件	(9)
1.4.2	进位标志 CY 与溢出标志 OV	(10)
1.5	BCD 码	(11)
1.6	ASCII 码	(11)
2	常用数字电路	(12)
2.1	逻辑电路	(12)
2.1.1	与运算	(12)
2.1.2	或运算	(12)
2.1.3	非运算	(12)
2.1.4	异或门	(13)
2.1.5	累加器	(13)
2.2	译码电路	(13)
2.2.1	地址译码	(13)
2.2.2	7 段数码管显示、译码	(14)
2.2.3	简单的实验元件	(14)
2.3	时序电路	(15)
2.3.1	触发器	(15)
2.3.2	锁存器	(15)

2.3.3 分频器与计数器	(16)
2.4 三态门和总线驱动	(16)
2.4.1 三态门	(16)
2.4.2 总线驱动	(17)
2.5 数字电路设计中采用的先进技术	(17)
2.5.1 PROM/PAL/GAL	(17)
2.5.2 CPLD/FPGA	(18)
2.5.3 数字系统中硬件方法与软件方法的特点与联系	(19)
3 微处理器	(20)
3.1 微处理器系统的组成	(20)
3.2 总线及总线时序	(21)
3.3 关于微处理器的基本概念	(23)
3.3.1 微控制器	(23)
3.3.2 嵌入式微处理器	(23)
3.4 CPU 的指令系统	(24)
3.4.1 指令	(24)
3.4.2 指令的编码	(24)
3.4.3 指令系统	(25)
3.4.4 指令的微操作	(25)
3.4.5 指令的重叠执行	(26)
3.4.6 执行速度方面的指标	(26)
3.4.7 精简指令计算机	(26)
3.5 CPU 中的寄存器	(26)
3.5.1 寄存器与程序运行	(27)
3.5.2 堆栈及堆栈指示器	(28)
4 存储器与存储管理	(30)
4.1 存储器概述	(30)
4.1.1 内存与外存	(30)
4.1.2 半导体存储器的结构	(30)
4.1.3 只读存储器	(31)
4.1.4 随机访问存储器	(32)
4.1.5 内存的主要性能指标	(32)
4.1.6 新型存储器	(32)
4.2 存储器的组织	(33)
4.2.1 存储器映象	(33)
4.2.2 存储器的地址空间	(34)

4.2.3 哈佛结构	(34)
4.3 存储管理	(35)
4.3.1 逻辑地址与物理地址	(35)
4.3.2 存储器的段式管理	(35)
4.3.3 存储器的页式管理	(35)
4.3.4 保护	(36)
4.4 存储器扩展中的总线连接	(36)
4.4.1 8 位字长存储器的扩展	(37)
4.4.2 16 位字长存储器的扩展	(37)
4.4.3 32 位字长存储器的扩展	(37)
4.4.4 Cache 高速缓冲技术	(38)
5 计算机的输入/输出	(40)
5.1 输入/输出的概念	(40)
5.1.1 外部设备	(40)
5.1.2 接口(Interface)与接口技术	(40)
5.1.3 端口地址(Port Address)	(40)
5.1.4 端口及其编址方法	(41)
5.2 输入/输出的数据传输方式	(42)
5.2.1 程序传送方式	(42)
5.2.2 中断概述	(42)
5.2.3 中断传送方式	(43)
5.2.4 DMA 方式的数据传送	(44)
5.3 常用数字接口部件	(45)
5.3.1 定时器/计数器	(45)
5.3.2 并行口和串行口	(46)
5.3.3 I ² C 和 SPI	(46)
6 MCS-51 系列微控制器	(47)
6.1 MCS-51 系列微控制器概述	(47)
6.2 CPU 结构	(48)
6.2.1 运算器	(48)
6.2.2 布尔处理机	(49)
6.2.3 控制器	(49)
6.3 MCS-51 的存储器组织	(51)
6.3.1 程序存储器	(52)
6.3.2 内部数据存储器	(53)
6.3.3 特殊功能寄存器	(54)

6.3.4	外部数据存储器和 I/O	(56)
6.4	MCS-51 系列微控制器的引脚功能 P0~P3	(56)
6.4.1	P1 口的内部结构	(57)
6.4.2	P3 口的内部结构	(58)
6.4.3	P2 口的内部结构	(59)
6.4.4	P0 口的内部结构	(59)
6.5	单片方式以及总线扩展方式	(59)
6.5.1	单片方式	(59)
6.5.2	应用示例	(60)
6.5.3	单片机的总线扩展	(61)
7	MCS-51 指令系统	(62)
7.1	MCS-51 指令概述	(62)
7.1.1	指令分类	(62)
7.1.2	书写格式与缩略符	(62)
7.1.3	寻址方式	(63)
7.2	数据传送类指令	(65)
7.2.1	内部数据传送指令	(65)
7.2.2	栈操作指令	(66)
7.2.3	外部数据传送指令	(67)
7.3	运算类指令	(68)
7.4	逻辑运算类指令	(70)
7.5	控制类指令	(72)
7.5.1	程序转移指令	(72)
7.5.2	子程序调用和返回指令	(77)
7.6	位操作指令	(79)
7.6.1	位操作与位地址	(79)
7.6.2	位传送指令	(80)
7.6.3	位状态控制指令	(80)
7.6.4	位逻辑运算指令	(80)
7.6.5	基于位测试的控制转移指令	(81)
8	汇编语言程序设计	(82)
8.1	汇编语言语句结构	(82)
8.1.1	程序行的组成	(82)
8.1.2	宏汇编中的常数	(83)
8.1.3	宏汇编中的表达式	(83)
8.2	伪指令	(84)

8.2.1	段定义伪指令	(84)
8.2.2	符号定义伪指令	(85)
8.2.3	存储单元定义并初始化	(86)
8.2.4	保留地址伪指令	(87)
8.2.5	模块连接伪指令	(89)
8.2.6	其他伪指令	(89)
8.3	常用程序设计方法	(90)
8.3.1	最大值、最小值问题	(91)
8.3.2	多分支转移问题	(92)
8.3.3	二进制数据转为 BCD 码的问题	(94)
8.4	模块化程序设计	(97)
8.4.1	ASM-51 编程环境下的文件类型	(97)
8.4.2	各种文件的关系及生成流程	(98)
8.4.3	由两个模块构成的程序示例	(98)
9	MCS-51 的定时器/计数器与中断系统	(102)
9.1	MCS-51 的定时器/计数器	(102)
9.1.1	定时器/计数器的结构	(102)
9.1.2	定时器/计数器的工作方式	(103)
9.1.3	定时器/计数器 T0 和 T1 的应用实例	(107)
9.2	增强型 MCS-51 的 T2 定时器	(110)
9.2.1	控制寄存器	(110)
9.2.2	T2 的 16 位自动装载方式	(111)
9.2.3	T2 的 16 位捕获方式	(112)
9.2.4	T2 的波特率发生器方式	(112)
9.3	MCS-51 的中断系统	(113)
9.3.1	中断源	(113)
9.3.2	中断源的屏蔽	(114)
9.3.3	中断的优先级	(115)
9.3.4	中断向量	(116)
9.3.5	中断响应的条件和过程	(117)
9.3.6	中断服务程序的特点和编写要求	(118)
9.3.7	数码管动态显示技术	(120)
9.3.8	定时器应用的其他示例	(125)
10	MCS-51 的串行通信	(127)
10.1	串行通信基本概念	(127)
10.1.1	通信方式	(127)

10.1.2 串行通信的波特率	(129)
10.1.3 异步串行通信的常用物理标准	(129)
10.2 MCS-51 的串行通信控制器	(130)
10.2.1 与串行通信相关的特殊功能寄存器的用法介绍	(131)
10.2.2 串行口的工作方式	(132)
10.3 串行通信控制器的应用	(136)
10.3.1 移位寄存器方式应用举例	(136)
10.3.2 串行通信的应用举例	(139)
10.3.3 接收中断的循环缓冲区	(147)
10.3.4 多机通信的中断接收	(148)
11 MCS-51 的总线	(151)
11.1 MCS-51 系统的时钟和复位电路	(151)
11.1.1 时钟电路	(151)
11.1.2 复位电路	(152)
11.2 MCS-51 的总线	(154)
11.2.1 总线的构成	(154)
11.2.2 总线的时序	(155)
11.3 应用举例	(160)
11.4 存储器访问中的等待	(161)
12 MCS-51 的存储器扩展电路	(162)
12.1 存储器扩展	(162)
12.1.1 存储器结构	(162)
12.1.2 存储器的地址容量扩展	(163)
12.1.3 译码电路	(164)
12.1.4 存储器在实际应用中的技巧	(166)
12.1.5 存储器的时序要求	(167)
12.2 程序存储器扩展	(169)
12.2.1 扩展示例	(169)
12.2.2 微控制器的 ISP、IAP 概要介绍	(172)
12.3 扩充数据存储器	(172)
12.4 DS12C887B 日历时钟的扩展	(178)
12.5 集成到 MCU 中的扩展数据存储器	(184)
13 I/O 接口电路及其扩展	(185)
13.1 简单并行口扩展	(185)
13.1.1 简单的并行口输入	(185)

13.1.2	带锁存的并行口输入	(186)
13.1.3	简单的并行口输出	(187)
13.2	8255 通用可编程并行接口芯片	(188)
13.2.1	8255 简介	(188)
13.2.2	8255 的工作说明	(190)
13.2.3	8255 与 MCS-51 总线的接口和编程	(192)
13.3	8253 可编程外围定时器/计数器	(196)
13.3.1	8253 的内部结构和引脚	(196)
13.3.2	8253 的控制字和工作方式	(197)
13.3.3	8253 与 MCS-51 总线的接口和编程	(200)
13.4	SC16C2552 通用 UART 通信控制器	(203)
13.4.1	SC16C2552 的功能描述	(203)
13.4.2	芯片的引脚功能和内部结构	(203)
13.4.3	SC16C2552 的寄存器	(205)
13.4.4	SC16C2552 的扩展电路	(209)
13.4.5	SC16C2552 的编程举例	(210)
13.5	液晶点阵屏的接口设计	(214)
14	模拟量的输入和输出	(217)
14.1	D/A 转换器	(217)
14.1.1	D/A 转换器的基本原理	(217)
14.1.2	D/A 转换器的主要技术指标	(219)
14.2	D/A 的应用电路	(219)
14.2.1	D/A 电路的单极性和双极性设计	(219)
14.2.2	DAC0832	(221)
14.3	A/D 转换器	(223)
14.3.1	A/D 转换器原理	(223)
14.3.2	A/D 转换器的主要技术指标	(225)
14.4	典型 A/D 转换器的接口设计	(226)
14.4.1	8 位 8 通道 ADC0809 及其应用	(226)
14.4.2	12 位 AD574B 及其应用	(230)
15	C51 高级语言编程	(234)
15.1	C51 的数据类型	(234)
15.2	Keil C51 扩展关键字	(235)
15.3	存储区域与存储模式	(235)
15.3.1	存储区域	(235)
15.3.2	存储模式	(237)

15.3.3	指针变量	(237)
15.3.4	绝对地址访问	(238)
15.4	C51 的函数	(238)
15.4.1	函数的声明	(238)
15.4.2	函数的参数与返回值	(239)
15.4.3	函数的寄存器组	(239)
15.4.4	中断函数	(239)
15.5	C51 模块化程序设计	(240)
15.6	C51 与 ASM-51 宏汇编语言的混合编程	(243)
15.6.1	函数名的转换	(244)
15.6.2	参数传递	(244)
15.6.3	函数的返回值	(246)
15.6.4	C 语言和汇编语言的接口示例	(246)
16	Keil 51 应用基础	(249)
16.1	μVision2 的界面	(249)
16.2	编辑源程序文件	(250)
16.2.1	编辑一个汇编语言程序	(250)
16.2.2	编辑一个 C 语言程序	(251)
16.3	创建工程文件	(252)
16.3.1	新建工程	(252)
16.3.2	打开已存在的工程	(257)
16.4	汇编/编译和链接	(257)
16.5	程序调试功能	(258)
16.5.1	进入调试模式	(259)
16.5.2	单步执行	(259)
16.5.3	在线汇编	(260)
16.5.4	断点设置	(260)
16.5.5	观察窗口	(261)
16.5.6	存储器窗口	(261)
16.5.7	串行窗口	(261)
16.5.8	性能分析窗口	(262)
16.6	实例分析	(262)
16.6.1	实例一	(262)
16.6.2	实例二	(263)
16.7	在线仿真调试功能	(268)

17	C8051F 系列微控制器及其应用	(269)
17.1	C8051F340 功能及实验电路简介.....	(269)
17.1.1	C8051F340 功能简介	(269)
17.1.2	C8051F340 实验电路简介	(270)
17.2	开发工具介绍及其使用.....	(270)
17.2.1	仿真器.....	(270)
17.2.2	集成开发环境的建立	(271)
17.2.3	建立工程	(272)
17.3	功能配置及初始化编程.....	(273)
17.3.1	系统时钟及复位源	(273)
17.3.2	端口和交叉开关的应用	(278)
17.3.3	中断源和定时器时钟选择	(282)
17.3.4	A/D 功能设定	(287)
17.3.5	EMIF 的设置	(293)
17.4	应用实例.....	(296)
附 录		(301)
附录 A	ASCII(美国信息交换标准码)字符表	(301)
附录 B	MCS-51 系列单片机指令表	(302)
附录 C	指令的周期数物指令编码的字节数	(307)
附录 D	C51 版例子	(310)
附录 E	C8051F340 实验板原理图	(314)
习 题		(316)
样 卷		(323)
参 考 文 献		(333)

计算机技术的发展以及应用方面的普及速度,甚至使年轻的读者都难免感叹。在计算机知识的教学方面,观念也一直在发生着变化。

以往对计算机的分类,主要是取决于其功能和规模:比如大型、中型和微型计算机,或者根据用途分为通用与专用。但无论是用 CPU 的集成度、运算速度还是计算机的存储容量,都不能刻画不断变化着的计算机世界。将通用计算机分为主机、服务器和 PC 机,这在 Internet 时代,是比较贴切的。这种划分代表了需求和相应的处理能力。

与通用计算机不同的另一类计算机被称为嵌入式计算机。它不是以通用计算机的形态出现,而是将应用了计算机技术的芯片,嵌入到应用对象中去。如工业控制、武器装备、仪器仪表、个人通信终端、家用电器控制、娱乐设施或器件等,都大量使用嵌入式计算机技术。

1972 年,当第一个 4 位的微处理器在 Intel 公司诞生的时候,人们期望用它来制造手持计算器。当 8 位的微处理器诞生的时候,人们用它来制造台式的个人计算机。同时,其工业应用也开始了。也就是说,在早期,即 1980 年左右,微处理器并没有分成通用和嵌入式应用。

当 8086/8088 经 80286/80386 向更高的目标——Pentium 发展的过程中,人们发现微处理器在应用上的定位开始日益明显。Intel 公司 1983 年左右推出的 80186/80188 就是为嵌入式应用服务的。它将 8086/8088 CPU 与外围功能芯片集成在一起,定位于工业应用。一二十年前出产的数控机床里面就可能找到这样的芯片。日本公司运用 80188/80186 的技术生产芯片,推出了 V20 和 V40 等,并大量应用到数控机床和其他领域。但这类芯片从没用于通用计算机的制造,所以并不广为人知。

从 80286 起,微处理器的设计者一开始就考虑它将来如何与操作系统结合,以生产出功能更强、更好用的通用计算机。从工作在通信行业的朋友那里能了解到,80386 甚至 Pentium 也可以直接应用在大容量的数字交换机上,但这种应用已经是非常专业的了,跟通用计算机主板的设计一样,必须依赖 Intel 公司的技术支持,否则就难以在短期内完成设计任务。

专业化、知识和技术在向高科技企业集中,这是大势所趋。80X86 体系的发展,集成了更多计算机领域的发展成就、更多的原理和技术。

除了 Intel 公司以外, Motorola、AMD 等公司在推动微处理器从 8 位发展到 16 位再到 32 位的过程中,都有精彩的故事。而现在微处理器在高端的发展方向是多核、64 位。不断提升的性能,一定能使我们的计算机变得更加优异,使我们的生活和学习更加方便。

让我们再来看一看微处理器在嵌入式领域的发展和应用。Intel 公司早在 1977—1978 年间,就利用其微处理器技术,开发了 8 位的 8048 微型控制器。经过改进,于 1981 年推出了 8051 系列微控制器。如今在通用计算机的键盘中,就有一个基于 8048 的控制器。到 1983 年,又推出了 8096 系列的 16 位微控制器。

二十多年过去了,人们本以为 8051 系列在流行了一阵以后也会如 8088 一样被淘汰。从教学角度人们也不看好它。这个以实用为主的、集成了 CPU、存储器和典型功能器件的芯片,

使用十分灵活的器件,一边被我们广泛地用着,一边却不被我们看好。

随着时间的推移,嵌入式微处理器出现了 32 位,时钟频率也从几兆上升到几百兆。可是,以 MCS-51 为代表的 8 位微处理器的销量不减反升,8 位 MCU 市场一直以 16% 的年复合增长率在增长。到 2004 年,MCU 与 DSP 的销售额占整个半导体市场的 10% 以上,可谓半导体家族中的重要分支。MCU 在 2004 年的销售额达到 120 亿美元,出货量达 70 亿片。在 MCU 产品中,8 位 MCU 和 32 位 MCU 被认为是两大支柱产品,其中,8 位 MCU 虽然面临各种新生代产品的挑战,地位仍岿然不动,而 32 位 MCU 伴随新兴通信网络的应用,也开始进入快速增长期,成为众多企业竞争的焦点。

“在嵌入式系统发展的 20 年历程中,流行过多次 8 位 MCU 的‘被替代论’,例如 16 位机推出时,CPLD/FPGA 出现后,32 位 ARM 应用热潮中,都有人扬言它们会替代 8 位 MCU,然而他们都错了。”嵌入式领域专家、北京航空航天大学何立民教授介绍说。他分析了 8 位 MCU 长盛不衰的原因:虽然某些 32 位 MCU 的价位下降,使原先使用 8 位 MCU 勉强胜任的电子系统转而使用 32 位 MCU,但 8 位 MCU 也在新形势下不断寻找新出路,如向 SoC 内核或智能器件方向发展,从而拓展出新的发展空间。他还说道,“而且,在很多 32 位的系统中,也有许多 8 位 MCU 的智能控制单元。就拿通用 PC 机来说,它们的键盘控制就采用了 8 位 MCU。”所以说,8 位 MCU 的应用空间是无限的,它将永不衰败。

成本因素是 8 位 MCU 存在的重要理由之一。虽然目前市场上一些低端 32 位 MCU 的价格已经在 50 元以下,但 30 元以下的 MCU 产品中,仍然以 8 位 MCU 为主。10 元以下产品就只有选择 8 位 MCU。

除了成本,8 位 MCU 在市场上永葆活力的另一个重要原因源自于它的不断变革,这使得 8 位 MCU 具有很高的市场适应性。8 位 MCU 曾进行的变革被总结称为“三次革命”:第一次是 Philips 公司将 8 位 MCU CMOS 化,不仅降低了 8 位 MCU 的成本,还使它能采用各种封装形式,集成各种外设,从而不断丰富产品系列;第二次是 Atmel 公司将 Flash 技术引入 8 位 MCU,迅速降低了带存储器 MCU 产品的成本,大大简化了用户对 MCU 擦写的流程,使用户可以快速对产品程序进行修改,从而满足市场对更短开发周期的需求;第三次革命是 Philips、Atmel、Silicon Labs、ADI、TI 等公司使 8 位 MCU 成为一个片上系统,集成了数字、模拟、存储器、传感器、看门狗等多种技术,提高了产品的集成度和实用性。

Silicon Labs 公司改进了目前的 8051 核心,把它的运行速度提高到 100 MIPS,这一速度足以替代 16 位 MCU 甚至低端 DSP 产品。此外,他们的 MCU 可以集成 24 位 ADC 和 12 位 DAC、晶体振荡器和传感器等,方便了用户在某些领域的设计和制造,并可提高终端产品的可靠性。

这些变革都给 8 位 MCU 增添了新活力,预示它们还有更加广阔的发展空间。MCS-51 系列 8 位微处理器历经二十多年不被淘汰的事实,使我们相信,实用性强不见得就不典范;虽然功能器件过于灵活了,但用的人多了,也就有了规范,学习时还是有规律可循的。

MCS-51 系列的成功与 Intel 公司采取的技术开放性策略也是分不开的。目前,全世界有最著名的 70 余家半导体厂商获得了该系列内核的授权,生产了数以百计的、各具特色的该系列微控制器。包括其他优秀的微控制器在内,世界市场上目前有 4 千余种微控制器在销售。正是由于微控制器的普及应用,使得计算机技术深入到世界的每一个角落。

嵌入式处理器和微控制器与通用计算机中的 CPU,虽然功能定位有很大的差距,但它们

毕竟是同源的。对于初学者来说,从本书的 MCS-51 系列微控制器入手,可以了解计算机运行的基本原理、基本的应用电路、基本的嵌入式编程技术,并可应用于工程实践,也可为以后深入学习高端的计算机硬件知识打下坚实的基础。

当前,Freescale(飞思卡尔)、AVR,PIC 等系列微控制器的发展也方兴未艾,而且各品牌都给出各种档次和适合不同应用的产品。在开发手段上也都非常具有竞争性。一个普遍的趋势就是良好的集成开发环境加 C 高级语言支持。本书以 MCS-51 为基础,介绍硬件开发的基本知识和汇编语言编程,再过渡到 C51 高级语言程序设计。相信读者在有了一定应用基础之后,将来换用其他系列微控制器,也不会遇到很大的困难。