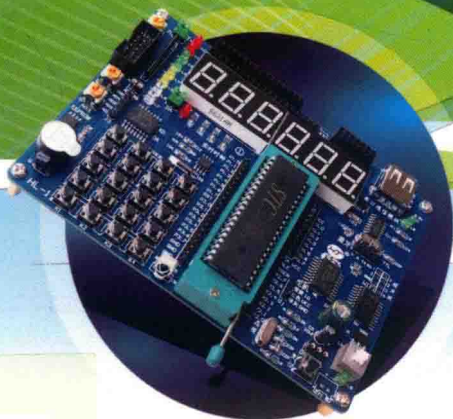




普通高等教育“十三五”规划教材  
电子信息科学与工程类专业规划教材

# 单片机原理与应用 (C51语言版)

◎ 欧伟明 刘剑 何静 凌云 等编著



 中国工信出版集团

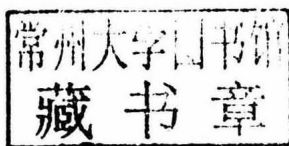
 电子工业出版社  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY  
<http://www.phei.com.cn>

普通高等教育“十三五”规划教材  
电子信息科学与工程类专业规划教材

# 单片机原理与应用

## (C51 语言版)

欧伟明 刘 剑 何 静 凌 云 等编著



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书以 89S51 为典型机, 论述单片机的基本结构与工作原理, 以及单片机应用系统的设计与开发方法。全书内容分为 13 章, 包括概述、单片机的结构和工作原理、指令系统、单片机 C51 语言程序设计基础、中断系统、定时器/计数器、单片机的串行口 UART、单片机常用并行接口技术、串行总线接口技术、单片机应用系统开发环境、基于嵌入式实时操作系统的单片机程序设计方法、基于 RTX51 的乐曲编辑器和发生器设计、数控电流源设计。书后附录给出了 18 个单片机课程设计题, 以及单片机 89S51 的指令系统。本书从工程应用出发, 突出单片机应用技术的新颖性和实用性; 此外, 本书为任课教师免费提供电子课件。

本书可作为高等学校单片机原理与应用、微机原理与接口技术课程的教材, 也可供从事单片机应用系统开发的工程技术人员参考, 还可作为各类电子设计竞赛的培训教材, 以及单片机课程设计的参考书和电类专业学生毕业设计的参考书。

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

单片机原理与应用: C51 语言版/欧伟明等编著. —北京: 电子工业出版社, 2019.4  
ISBN 978-7-121-36125-8

I. ①单… II. ①欧… III. ①单片微型计算机—C 语言—程序设计—高等学校—教材  
IV. ①TP368.1 ②TP312.8

中国版本图书馆 CIP 数据核字 (2019) 第 043864 号

策划编辑: 谭海平

责任编辑: 谭海平 特约编辑: 王 崧

印 刷: 北京虎彩文化传播有限公司

装 订: 北京虎彩文化传播有限公司

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: 21.75 字数: 556.8 千字

版 次: 2019 年 4 月第 1 版

印 次: 2019 年 8 月第 2 次印刷

定 价: 55.00 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, 88258888。

质量投诉请发邮件至 [zltz@phei.com.cn](mailto:zltz@phei.com.cn), 盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

本书咨询联系方式: (010) 88254552, [tan02@phei.com.cn](mailto:tan02@phei.com.cn)。

# 前 言

自 1971 年微型计算机问世以来,由于实际应用的需要,微型计算机向着两个方向发展:一是向着高速度、大容量、高性能的高档微机方向发展;二是向着稳定可靠、体积小、功耗低、价格低廉的单片机方向发展。单片机是微型计算机的一个重要分支,它的出现是计算机技术发展史上的一个重要里程碑,它使计算机从海量存储与高速复杂数值计算进入智能化控制领域。从此,计算机技术的两个重要领域——通用计算机领域和嵌入式计算机领域都取得了极其重大的进展。

单片机诞生于 20 世纪 70 年代。自美国 Intel 公司于 1976 年宣布并于 1977 年推出 MCS-48 单片机以来,单片机技术已经走过了 40 余年的历程。我国自 20 世纪 70 年代末 80 年代初就开始进行单片机的应用与开发工作。1987 年 10 月 27 日,我国在上海成立中国微计算机单片机学会,中国的单片机开发与应用经历了 30 余年。几十年来,单片机不是以其位数的高低来决定优劣的,而是以如何适合千变万化的应用产品的需求、高性价比的配置来决定优劣的。因此,高性价比、多功能、低功耗的 8 位单片机一直是单片机的主角。

本书仍然保持前一版的写作风格,在内容上对原书进行了仔细的修订。前一版主要以汇编语言作为单片机编程语言,这一版主要采用 C51 语言进行单片机程序设计。在编著本书时,我们主要考虑了以下几点。

## (1) 关于单片机的选型

目前,国内外公认的单片机标准体系结构是美国 Intel 公司的 MCS-51 系列,其中的 8051 单片机由 Intel 公司以技术转卖的方式,被许多半导体生产厂家作为基核,发展了许多兼容系列,所有这些系列统称为 80C51 系列。因此,人们在设计单片机应用系统时,可以根据应用系统的要求,广泛选择最佳型号的单片机。然而,美国 Atmel 公司的单片机 AT89S51 是 80C51 系列的典型代表,所以本书以 89S51 芯片为主线介绍单片机的原理与应用。

## (2) 全书的整体架构

全书分为两大部分。第一部分,即本书的第 1 章~第 8 章,主要介绍单片机结构原理及基本应用,它既是继续学习单片机应用技术的基础,又是单片机原理与应用课程的经典内容。

第二部分,即本书的第 9 章~第 13 章,主要介绍当前的单片机应用新技术,以及单片机应用系统设计开发方法和工程设计实例。显然,第二部分是单片机原理与应用课程经典内容的扩展,主要目的是给读者提供继续学习和掌握单片机应用系统开发技术的精选内容,让读者了解当前单片机应用的新技术、开发小工具、开发环境、开发过程,从而达到初步掌握单片机应用系统设计与开发技术的目的。

## (3) 精心安排“经典内容”,认真撰写第 1 章~第 8 章

从工程应用的角度出发,通过精心安排本书第 1 章~第 8 章的内容,达到既讲透单片机的结构原理,又精简“经典内容”体系结构的目的。比如,单片机的工作方式有多种,但从工程应用的角度来看,主要用到的是复位工作方式、低功耗工作方式和编程工作方式。因此,本书在介绍复位电路的基础上,重点介绍复位工作方式、低功耗工作方式和 AT89S51

的 ISP 编程工作方式，而对其他工作方式只是略为提一下。又如，定时器/计数器有 4 种工作方式，其他教材一般按照顺序介绍工作方式 0~工作方式 3，但在实际的工程应用中，工作方式 0 很少采用而工作方式 1 应用最多，所以本书按照工作方式 1、工作方式 2、工作方式 3、工作方式 0 的顺序进行介绍，以便使教材的内容贴近实际需要。再如，单片机的存储器有程序存储器和数据存储器，而实际上特殊功能寄存器也属于单片机的存储器，因此本书将这三方面的内容放在一个小节内介绍，这种安排是有别于其他教材的。为保证本书中所用的实例程序的正确性，所选用的实例程序都通过了实际验证。

#### (4) 详细介绍串行总线接口技术

随着计算机技术和半导体技术的发展，MCU 芯片的内部资源越来越丰富，总线型单片机的非总线应用模式使用得越来越广泛；在 MCU 的外部，很少采用并行三总线（数据总线、地址总线、控制总线）的结构，而常常采用具有串行总线接口的外围芯片。因此，本书用一章的篇幅来详细介绍串行总线接口技术，包括 RS-232C、RS-485、SPI、I<sup>2</sup>C、1-Wire、CAN、USB 等。

#### (5) 介绍 C51 语言程序设计方法

目前，在单片机程序设计部分，讲授内容大多限于“汇编语言”，而在实际应用中，单片机程序设计在多年前就已进入“高级语言”阶段，各种单片机高级语言开发工具的相继出现，使得高级语言程序设计在可读性、可靠性和编程效率上都远超过汇编语言，德国 Keil Software 公司的 Keil C51 编译器就是典型代表。Keil C51 编译器是一种专为 MCS-51 系列单片机应用开发而设计的高效率 C 语言编译器，该编译器包括 C51 交叉编译器、A51 宏汇编器、BL51 连接定位器和基于 Windows 的集成化文件管理编译环境、多视窗软件仿真调试器等一系列开发工具，具有高效、可靠、使用方便等优点，其应用如今已十分普及。面向 MCS-51 系列单片机的 C 语言称为 C51 语言，它已经成为单片机的主流程序设计语言。

本书采用 C51 语言作为单片机程序设计语言。然而，我们认为目前在教学中不宜完全忽略汇编语言程序设计方法的介绍，因为在许多实时控制时序和时间要求十分苛刻的场合，尤其在控制接口硬件时，用汇编语言进行程序设计显得非常简洁。因此，作为选学内容，本书保留了 MCS-51 单片机指令系统并简要介绍了汇编语言程序设计方法。

第 11 章~第 13 章不仅采用了 C51 语言程序设计方法，有时还采用了 C51 语言程序调用汇编语言子程序的编程方法，这种安排对读者而言具有很好的指导作用。

#### (6) 介绍基于嵌入式实时操作系统的单片机程序设计方法

第 11 章介绍适用于 MCS-51 系列 8 位单片机的嵌入式实时操作系统 RTX51 及其应用方法。这个内容有别于嵌入式系统课程的内容：一方面，实时操作系统 RTX51 是适用于 MCS-51 系列 8 位单片机的，而嵌入式系统课程中介绍的实时操作系统是适用于 32 位嵌入式微处理器或 64 位嵌入式微处理器的；另一方面，实时操作系统 RTX51 易学好用，作者在实际的工程项目中使用它后觉得很好，因此将其放到本书中，这也是本书的亮点之一。

#### (7) 简单介绍单片机应用系统设计开发环境

第 10 章主要介绍单片机开发小工具、开发环境 Keil  $\mu$ Vision4 和 Proteus 以及开发步骤，目的是让读者了解单片机应用系统设计开发的方法和工程设计步骤，建立单片机应用系统设计的全局观念。一般来说，单片机开发环境的介绍需要较大的篇幅，且已有专门的书籍进行

了介绍,而要真正掌握单片机的开发环境,只有经过大量的训练后才能实现,因此对于这部分内容本书只做简单介绍,读者必须通过实际操作,不断积累经验,直至熟练运用单片机开发环境。

#### (8) 与工程应用相结合,选取完整的设计实例

第12章和第13章是两个完整的单片机应用系统的设计实例,取材于全国大学生电子设计竞赛的国家级获奖作品和实际的工程设计。这两章都给出了完整的系统设计过程,不仅给出了完整的系统硬件电路原理图,而且给出了完整的系统软件设计源程序代码,系统应用程序采用C51语言进行编写,或采用C51语言和MCS-51汇编语言混合编写。此外,这两章的编写体例是按照电气信息类专业本科毕业设计的论文格式要求进行撰写的,在书中还给出了所设计与制作实物的数码照片,以便增强实际效果。因此,本书的这两章不仅可供从事单片机应用系统开发的工程技术人员参考,还可作为各类电子设计竞赛的培训内容,以及单片机课程设计的参考内容和电气信息类专业学生毕业设计的参考内容。

本书以89S51为典型机,主要论述单片机的基本结构与工作原理,以及单片机应用系统的设计与开发方法。全书内容分为13章,内容包括概述、单片机的结构和工作原理、指令系统、单片机C51语言程序设计基础、中断系统、定时器/计数器、单片机的串行口UART、单片机常用并行接口技术、串行总线接口技术、单片机应用系统开发环境、基于嵌入式实时操作系统的单片机程序设计方法、基于RTX51的乐曲编辑器和发生器设计、数控电流源设计。书后附录给出了18个单片机课程设计题,以及单片机89S51的指令系统。

本书由欧伟明教授任主编,刘剑、何静、凌云任副主编。欧伟明教授撰写第1章、第4章、第11~12章,龙晓薇老师撰写第2章,李小宝博士撰写第3章,何静博士撰写第5章,李圣清教授撰写第6章,刘剑副教授撰写第7章、第10章,欧伟明教授、贺正芸老师撰写第8章,凌云教授撰写第9章,蒋中荣副教授撰写第13章,周玉副教授撰写附录A,李朝仑老师撰写附录B。全书由欧伟明教授统稿和定稿。

本书的撰写得到了湖南工业大学的张昌凡教授、贺素良教授(参加了本书前一版的撰写工作)、朱晓青教授、龙永红教授、张满生教授、李祥飞博士、李燕林老师的大力支持,他们给予了作者鼓励和关于本教材的编写意见;还得到了湖南工业大学电气与信息工程学院的毕业生刘张胜、庄永军、楚瑞玉、任杰、周韬、李军杰、蒙毓李、杨敬力、柳红新、付贵勇、欧阳文彦、郭仁的支持,他们对书中部分硬件电路和部分程序的初步调试做了有益的工作。在此一并表示衷心的感谢!

欧伟明

2018年12月于湖南工业大学

# 目 录

第 1 章 概述	1
1.1 单片机概念与发展过程	1
1.1.1 单片机概念	1
1.1.2 单片机技术发展过程	1
1.1.3 单片机技术发展方向	3
1.1.4 常用数制与编码	4
1.2 单片机应用领域与嵌入式系统概念	5
1.2.1 单片机应用领域	6
1.2.2 嵌入式系统概念	6
1.3 单片机应用系统开发过程简述	8
1.3.1 单片机编程语言	8
1.3.2 单片机应用系统结构	9
1.3.3 单片机应用模式	10
1.3.4 单片机应用系统开发过程简介	11
1.4 本书特点与教材使用建议	12
1.4.1 本书编写指导思想	13
1.4.2 本书特点	15
1.4.3 教材使用建议	16
1.5 本章小结	18
1.6 思考题与习题	19
第 2 章 单片机的结构和工作原理	20
2.1 MCS-51 系列单片机概述	20
2.2 89S51 单片机引脚功能说明	21
2.2.1 89S51 的引脚图与封装	21
2.2.2 89S51 的引脚功能说明	22
2.2.3 89S51 的引脚应用特性	23
2.3 89S51 单片机内部结构	24
2.3.1 89S51 的基本组成	24
2.3.2 89S51 的 CPU	26
2.4 89S51 单片机的存储器	28
2.4.1 程序存储器	29
2.4.2 数据存储器	29
2.5 89S51 单片机的时钟电路与时序	32

2.5.1	时钟电路	32
2.5.2	基本时序单位	33
2.6	89S51 单片机的工作方式	35
2.6.1	复位工作方式和复位电路	35
2.6.2	低功耗工作方式	36
2.6.3	串行 ISP 编程方式	37
2.7	89S51 单片机的输入/输出端口	38
2.7.1	P0 端口	38
2.7.2	P1 端口	39
2.7.3	P2 端口	40
2.7.4	P3 端口	40
2.8	本章小结	41
2.9	思考题与习题	42
<b>第 3 章</b>	<b>指令系统*</b>	<b>43</b>
3.1	MCS-51 单片机指令概述	43
3.1.1	指令格式	43
3.1.2	符号说明	44
3.2	寻址方式	45
3.2.1	寄存器寻址方式	45
3.2.2	直接寻址方式	45
3.2.3	寄存器间接寻址方式	46
3.2.4	立即寻址方式	46
3.2.5	变址寻址方式	46
3.2.6	相对寻址方式	47
3.2.7	位寻址方式	47
3.3	89S51 单片机的指令系统	47
3.3.1	数据传送类指令	47
3.3.2	算术运算类指令	50
3.3.3	逻辑运算及移位类指令	53
3.3.4	控制转移类指令	54
3.3.5	位操作类指令	56
3.4	单片机汇编语言简介	58
3.4.1	汇编语言的语句格式	58
3.4.2	伪指令	59
3.4.3	单片机汇编语言程序设计	60
3.5	本章小结	63
3.6	思考题与习题	63

第 4 章 单片机 C51 语言程序设计基础	65
4.1 单片机 C51 语言概述	65
4.1.1 C51 语言在单片机应用系统开发中的优势	65
4.1.2 C51 语言与标准 C 语言的比较	65
4.1.3 编写 C51 语言程序的基本原则	66
4.2 C51 语言关键字与数据类型	67
4.2.1 标识符	67
4.2.2 关键字	68
4.2.3 数据类型	69
4.3 C51 语言数据	71
4.3.1 常量	71
4.3.2 变量	72
4.3.3 存储器类型和存储器模式	72
4.3.4 数组	74
4.3.5 指针	75
4.4 C51 语言对单片机硬件资源的控制	76
4.4.1 特殊功能寄存器 (SFR) 的定义	76
4.4.2 位变量的定义	77
4.4.3 存储器和外接 I/O 端口的绝对地址访问	78
4.5 C51 语言运算符和表达式	79
4.5.1 运算符	79
4.5.2 表达式	81
4.6 C51 语言流程控制语句	81
4.6.1 语句的概念和分类	81
4.6.2 判断分支 (if、switch 语句)	82
4.6.3 循环控制 (for、while 语句)	84
4.6.4 break、continue、return、goto 语句	85
4.7 C51 语言函数	86
4.7.1 函数的定义	87
4.7.2 函数的调用	88
4.7.3 C51 语言中断函数	89
4.8 C51 语言预处理命令	90
4.8.1 文件包含	90
4.8.2 宏定义	90
4.8.3 条件编译	91
4.9 C51 语言与汇编语言混合编程方法	91
4.9.1 C51 语言程序嵌入汇编语句	92
4.9.2 C51 语言程序调用汇编语言子程序	93

4.10	本章小结	94
4.11	思考题与习题	95
<b>第5章</b>	<b>中断系统</b>	<b>96</b>
5.1	中断	96
5.1.1	中断的概念	96
5.1.2	中断的条件和中断响应过程	97
5.2	89S51 中断系统结构与控制	98
5.2.1	89S51 的中断源和中断入口地址	98
5.2.2	89S51 的中断系统结构	99
5.2.3	中断控制	100
5.3	中断应用举例	105
5.3.1	单外部中断源系统的设计	105
5.3.2	多外部中断源系统的设计	106
5.4	本章小结	107
5.5	思考题与习题	108
<b>第6章</b>	<b>定时器/计数器</b>	<b>109</b>
6.1	定时器/计数器的结构与控制	109
6.1.1	89S51 定时器/计数器的结构	109
6.1.2	定时器/计数器的控制	110
6.2	定时器/计数器的4种工作方式	111
6.2.1	工作方式1	111
6.2.2	工作方式2	112
6.2.3	工作方式3	113
6.2.4	工作方式0	115
6.3	定时器/计数器的应用举例	115
6.3.1	脉冲信号的产生	115
6.3.2	脉冲宽度的测量	116
6.4	本章小结	117
6.5	思考题与习题	117
<b>第7章</b>	<b>单片机的串行口 UART</b>	<b>119</b>
7.1	串行通信概述	119
7.1.1	串行通信与并行通信	119
7.1.2	串行通信的分类	119
7.1.3	串行通信的数据传送方式	121
7.2	89S51 串行口 UART 的结构与控制	122
7.2.1	串行口 UART 的结构	122

7.2.2	串行口 UART 的工作方式	124
7.2.3	串行口 UART 的波特率计算	126
7.3	串行口 UART 的编程及应用实例	128
7.3.1	串行口 UART 的编程步骤	128
7.3.2	串行口 UART 应用实例	128
7.4	本章小结	131
7.5	思考题与习题	131
<b>第 8 章</b>	<b>单片机常用并行接口技术</b>	<b>133</b>
8.1	键盘接口	133
8.1.1	独立按键	134
8.1.2	矩阵键盘	136
8.2	LED 显示器接口	141
8.2.1	LED 数码管	141
8.2.2	LED 数码管静态显示接口	142
8.2.3	LED 数码管动态显示接口	144
8.3	DAC 接口	147
8.3.1	DAC0832 芯片介绍	147
8.3.2	DAC0832 与 89S51 的接口电路	148
8.3.3	利用 DAC0832 输出各种电压波形	149
8.4	ADC 接口	151
8.4.1	ADC0809 芯片介绍	151
8.4.2	ADC0809 与 89S51 的接口电路	153
8.4.3	ADC0809 应用举例	154
8.5	液晶显示模块 LCD1602 的接口	155
8.5.1	LCD1602 介绍	155
8.5.2	LCD1602 与 89S51 的接口电路	160
8.5.3	LCD1602 应用举例	161
8.6	外部并行三总线接口	164
8.7	大功率器件驱动接口	165
8.7.1	光耦接口	166
8.7.2	继电器接口	166
8.7.3	双向晶闸管输出接口	167
8.7.4	固态继电器输出接口	168
8.8	本章小结	169
8.9	思考题与习题	169
<b>第 9 章</b>	<b>串行总线接口技术</b>	<b>170</b>
9.1	EIA 系列总线标准及其接口	170

9.1.1	RS-232C 总线	170
9.1.2	RS-485 总线	172
9.1.3	单片机与 PC 之间的通信	174
9.2	SPI 总线	176
9.2.1	SPI 总线简介	176
9.2.2	SPI 总线通信协议	177
9.2.3	E <sup>2</sup> PROM 存储器 AT93C46 及其应用	177
9.3	I <sup>2</sup> C 总线	180
9.3.1	I <sup>2</sup> C 总线简介	180
9.3.2	I <sup>2</sup> C 总线通信协议	181
9.3.3	I <sup>2</sup> C 接口存储器 AT24C02 及其应用	183
9.4	1-Wire 单总线	191
9.4.1	1-Wire 单总线简介	191
9.4.2	温度传感器 DS18B20 及其应用	193
9.5	USB 总线	198
9.5.1	USB 总线原理	198
9.5.2	USB 总线通信接口设计实例	200
9.6	CAN 总线	202
9.6.1	CAN 总线简介	203
9.6.2	CAN 总线控制器	204
9.6.3	CAN 总线通信接口设计实例	204
9.7	本章小结	205
9.8	思考题与习题	206
<b>第 10 章</b>	<b>单片机应用系统开发环境</b>	<b>207</b>
10.1	单片机应用系统的调试方法	207
10.1.1	硬件调试方法	207
10.1.2	软件仿真调试方法	209
10.2	Keil $\mu$ Vision4 集成开发环境	210
10.2.1	Keil $\mu$ Vision4 的主要特性	210
10.2.2	Keil $\mu$ Vision4 集成开发环境设置方法	211
10.2.3	Keil $\mu$ Vision4 工程应用	216
10.2.4	Keil C51 主要头文件介绍	226
10.3	Proteus 8 仿真软件	228
10.3.1	Proteus 8 主界面介绍	228
10.3.2	Proteus 8 绘制电路原理图	230
10.3.3	Proteus 8 仿真调试	232
10.4	单片机应用系统开发小工具	233
10.4.1	波特率初值计算工具	233

10.4.2	数码管编码器 .....	233
10.4.3	定时器计算工具 .....	234
10.4.4	串口调试助手 .....	234
10.5	本章小结 .....	235
10.6	思考题与习题 .....	236
<b>第 11 章</b>	<b>基于嵌入式实时操作系统的单片机程序设计方法 .....</b>	<b>237</b>
11.1	嵌入式实时操作系统的概念 .....	237
11.1.1	嵌入式系统的特征 .....	237
11.1.2	嵌入式实时操作系统的概念 .....	238
11.2	在电子系统设计中引入 RTOS 的意义 .....	238
11.2.1	两种软件开发模式的比较 .....	239
11.2.2	嵌入式应用中使用嵌入式 RTOS 的必要性 .....	239
11.2.3	嵌入式操作系统环境下的应用软件设计 .....	240
11.2.4	嵌入式操作系统环境下的应用软件调试 .....	241
11.3	嵌入式实时操作系统 RTX51 的介绍 .....	241
11.3.1	RTX51 的技术参数 .....	241
11.3.2	几个概念 .....	242
11.3.3	RTX Tiny 内核分析 .....	245
11.3.4	RTX Tiny 内核源代码 .....	249
11.4	基于 RTX51 的单片机程序设计方法 .....	251
11.4.1	目标系统需求 .....	251
11.4.2	软件设计指导方针 .....	251
11.4.3	任务划分的原则 .....	252
11.4.4	应用程序架构 .....	254
11.5	本章小结 .....	256
11.6	思考题与习题 .....	256
<b>第 12 章</b>	<b>基于 RTX51 的乐曲编辑器和发生器设计 .....</b>	<b>257</b>
12.1	设计任务 .....	257
12.2	方案设计与论证 .....	257
12.2.1	以 FPGA 为核心的实现方案 .....	257
12.2.2	以 MCU 为核心的实现方案 .....	257
12.3	系统硬件设计 .....	258
12.3.1	系统硬件电路原理图 .....	258
12.3.2	人机交互界面 .....	259
12.4	基于 RTX51 的系统软件设计 .....	260
12.4.1	乐曲的表示方法 .....	260
12.4.2	编辑乐曲的软件实现方法 .....	261

12.4.3	播放乐曲的软件实现方法	262
12.4.4	系统软件流程框图	264
12.5	系统源程序清单	265
12.5.1	C51 语言主程序	265
12.5.2	读 AT24C02 汇编语言子程序	276
12.5.3	写 AT24C02 汇编语言子程序	278
12.5.4	键盘扫描汇编语言子程序	280
12.5.5	实时操作系统 RTX51 Tiny 内核程序	282
12.6	系统设计总结	282
<b>第 13 章</b>	<b>数控电流源设计</b>	<b>283</b>
13.1	设计任务	283
13.2	方案设计与论证	283
13.2.1	D/A 转换模块设计方案的论证与比较	284
13.2.2	恒流源模块设计方案的论证与比较	284
13.2.3	数据采集模块设计方案的论证与比较	285
13.2.4	辅助电源、主电源设计方案的论证与比较	286
13.2.5	键盘、显示器设计方案的论证与比较	287
13.3	理论计算与 EWB 仿真	287
13.3.1	采样电阻值的确定	287
13.3.2	D/A 转换器分辨率的确定	288
13.3.3	TLC5618 参考电压的确定	288
13.3.4	主电源参数的确定	288
13.3.5	用 EWB 进行电路仿真	288
13.4	系统硬件设计	290
13.4.1	MCU 微控制器、键盘、显示器电路图	290
13.4.2	D/A 转换模块、恒流源模块的电路图	290
13.4.3	数据采集模块的电路图	293
13.4.4	辅助电源、主电源的电路图	293
13.5	系统软件设计	297
13.5.1	主程序流程框图	297
13.5.2	设置输出电流给定值功能函数程序流程框图	298
13.5.3	设置电流步进值功能函数程序流程框图	298
13.5.4	键盘扫描程序流程框图	299
13.6	系统测试方法与结果分析	299
13.6.1	测试使用的仪器	299
13.6.2	恒流特性的测试	300
13.6.3	电流步进值为 1mA 的测试	300
13.6.4	纹波电流的测试	301

13.6.5	输出电流范围的测试 .....	301
13.6.6	输出电压的测试 .....	301
13.6.7	1~99mA 内任意电流步进值设置功能的测试 .....	302
13.6.8	测试结果分析 .....	302
13.7	系统使用说明书 .....	302
13.7.1	键盘界面 .....	303
13.7.2	菜单操作 .....	304
13.8	系统源程序清单 .....	304
13.8.1	C51 语言主程序 .....	304
13.8.2	键盘扫描汇编语言子程序 .....	315
13.8.3	写 TLC5618 的汇编语言子程序 .....	317
13.8.4	读 MC14433 的汇编语言子程序 .....	318
13.8.5	显示缓冲器的汇编语言子程序 .....	319
13.9	系统设计总结 .....	320
附录 A	单片机课程设计 .....	321
附录 B	89S51 指令表 .....	330
参考文献	.....	334

# 第 1 章 概 述

本章主要对单片机概念与发展过程、单片机应用领域与嵌入式系统概念、单片机应用系统开发过程、本书编写指导思想与教学安排建议等内容,进行简要介绍,以便读者对单片机及其应用技术有一个初步了解。

## 1.1 单片机概念与发展过程

自 1971 年微型计算机问世以来,由于实际应用的需要,微型计算机向着两个方向发展:一是向着高速度、大容量、高性能的高档微机方向发展;二是向着稳定可靠、体积小、功耗低、价格低廉的单片机方向发展。单片机是微型计算机的一个重要分支,它的出现是计算机技术发展史上的一个重要里程碑,它使计算机从海量存储与高速复杂数值计算进入智能化控制领域。从此,计算机技术的两个重要领域——通用计算机领域和嵌入式计算机领域都取得了极其重大的进展。

### 1.1.1 单片机概念

单片机在一块半导体硅片上集成了计算机的所有基本功能部件,包括中央处理器(CPU)、存储器(RAM 和 ROM)、输入/输出接口电路、中断系统、定时器/计数器和串行通信接口电路等。因此,单片机只需要与适当的软件及适当的外部设备相结合,就可以构成一个完整的计算机应用系统。

单芯片形式的微型计算机简称单片机,“单片机”一词真实地反映了单片机的形态和本质。由于单片机主要应用于各种测控系统中,因此目前国外普遍称单片机为微控制器(MicroController Unit, MCU)。鉴于单片机在应用时通常是测控系统的核心并融入被控对象,即以嵌入被控对象的方式进行使用,为强调其嵌入式应用特点,我们又称其为嵌入式微控制器(Embedded MicroController Unit, EMCU)。

在汉语中,单片机的称呼简洁通俗,以致单片机一词约定俗成而沿用至今,因此本书仍然使用“单片机”一词。

### 1.1.2 单片机技术发展过程

单片机诞生于 20 世纪 70 年代。自美国 Intel 公司于 1976 年宣布并于 1977 年推出 MCS-48 单片机以来,单片机技术已走过了 40 余年的历程。我国自 20 世纪 70 年代末 80 年代初就开始进行单片机的应用与开发工作。1987 年 10 月 27 日,我国在上海成立中国微计算机单片机学会,中国的单片机开发与应用经历了 30 余年。几十年来,单片机不以其位数的高低来决定其优劣,而以如何适合千变万化的应用产品的需求、高性价比的配置来决定其优劣。因此,高性价比、多功能、低功耗的 8 位单片机一直是单片机的主角。

单片机作为微型计算机的一个重要分支,应用面很广,发展很快。如果将 8 位单片机的推出作为起点,那么单片机的发展历史大致可分为以下几个阶段。

### 1. 第一阶段 (1976—1978 年): 单片机探索阶段

工控领域对计算机提出了嵌入式应用的要求,首先是实现单芯片形态的计算机,以满足构成大量中小型智能化测控系统的要求。因此,这一阶段的任务是探索计算机的单芯片集成。“单片机”一词即由此而来。

在单片机探索阶段,单片机体系结构有两种模式,即通用 CPU 模式和专用 CPU 模式。

(1) 通用 CPU 模式。采用通用 CPU 和通用外围单元电路的集成方式,这种模式以美国 Motorola 公司的 MC6801 单片机为代表,它将通用 CPU、增强型 6800+和 6875 (时钟)、6810 (128B RAM)、2×6830 (1KB ROM)、1/2 6821 (并行 I/O 端口)、1/3 6840 (定时器/计数器)、6850 (串行 I/O 端口) 集成在一个芯片上构成,使用 6800 CPU 指令系统。

(2) 专用 CPU 模式。采用专门为嵌入式系统要求设计的 CPU 与外围电路集成的方式。这种模式以美国 Intel 公司的 MCS-48 单片机为代表,其 CPU、存储器、定时器/计数器、中断系统、I/O 端口、时钟以及指令系统都是按嵌入式系统要求而专门设计的。

事实证明,这两种模式都是可行的。专用 CPU 模式能充分满足嵌入式应用的要求,成为今后单片机发展的主要体系结构模式;通用 CPU 模式则与通用计算机兼容,应用系统开发方便,成为后来嵌入式微处理器的发展模式。

### 2. 第二阶段 (1978—1982 年): 单片机完善阶段

Intel 公司在 MCS-48 单片机基础上推出了完善的、典型的 MCS-51 系列单片机。它在以下几个方面奠定了典型的通用总线型单片机体系结构:

(1) 规范的总线结构。有 8 位数据总线、16 位地址总线及多功能的异步串行通信接口 UART。

(2) CPU 外围功能单元电路的集中管理模式。

(3) 提供体现工控特性的位地址空间及位操作方式。

(4) 指令系统趋于丰富和完善,并且增加了许多突出控制功能的指令。

### 3. 第三阶段 (1982—1990 年): 微控制器形成阶段

8 位单片机的巩固和发展以及 16 位单片机的推出阶段,正是单片机向微控制器发展的阶段。Intel 公司推出的 MCS-96 系列单片机,将一些用于测控系统的模数转换器、程序运行监视器、脉宽调制器等纳入芯片内部,体现了单片机的微控制器特征。随着 MCS-51 系列的广泛应用,许多半导体制造厂商竞相使用 80C51 为内核,将许多测控系统中使用的电路技术、接口技术、多通道 A/D 转换部件、可靠性技术等应用到单片机,增强了外围电路功能,强化了智能控制的特征。

为了满足测控系统的嵌入式应用要求,这一阶段单片机的主要技术发展方向是满足测控对象要求的外围电路的增强,从而形成了不同于 Single Chip MicroComputer 的特点的微控制器,微控制器 (MicroController Unit, MCU) 一词源于这一阶段,至今微控制器 (MCU) 是国际上对单片机的标准称呼。

在微控制器形成阶段,单片机技术的主要发展如下:

(1) 外围功能电路集成到了芯片内部。满足模拟量输入的 ADC; 满足伺服驱动的 PWM; 满足高速 I/O 控制的高速 I/O 端口; 保证程序可靠运行的程序监视定时器 WDT。

(2) 出现了为满足串行外围扩展要求的串行扩展总线及接口,如 SPI、I<sup>2</sup>C、MICROWIRE、USB、1-Wire 等。