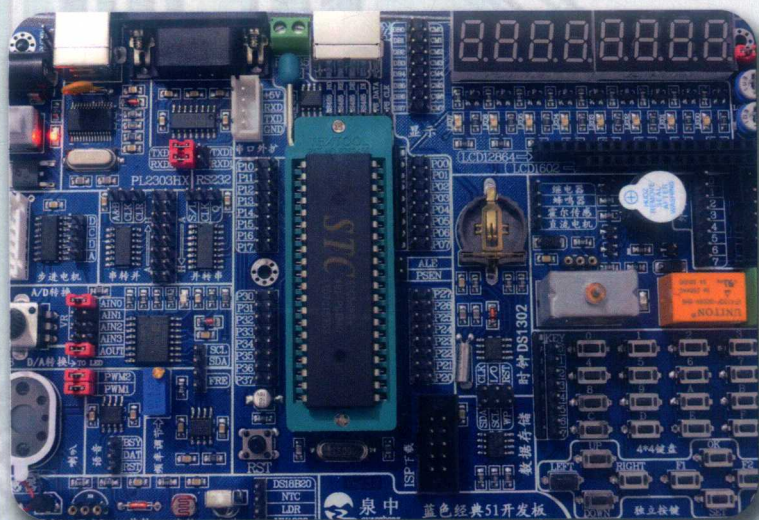


高等院校应用型本专科通用规划教材

MCS-51单片机 原理与应用

MCS-51 DANPIANJI YUANLI YU YINGYONG

主 编 吴静进 何尚平 万 彬



外
借

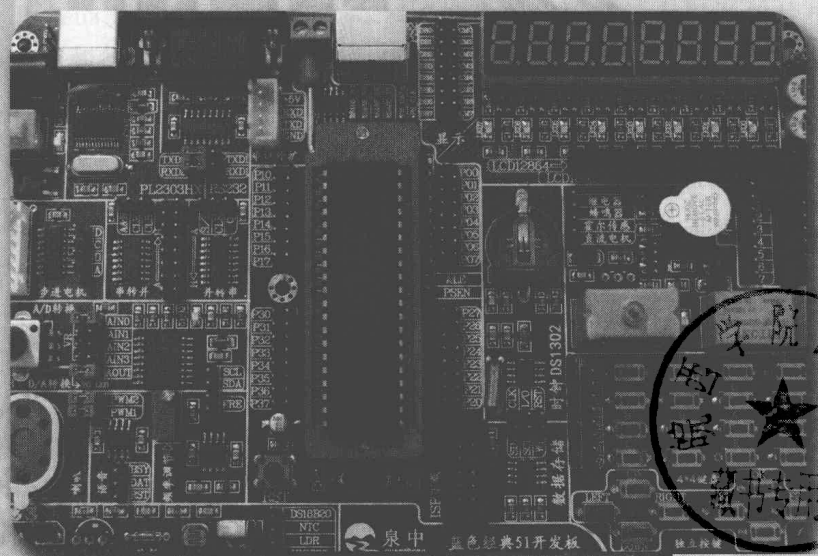


重庆大学出版社

高等院校应用型本专科通用规划教材

MCS-51单片机 原理与应用

主 编 吴静进 何尚平 万 彬



重庆大学出版社

内容提要

本书从单片机基础理论教学和工程实际应用角度出发,以 MCS-51 系列单片机为例,以易学实用为主,注重理论与实际相结合,循序渐进地讲述了 MCS-51 系列单片机基本概念、内部硬件结构、指令系统及汇编语言程序设计、中断系统、定时/计数器、串行通信技术、外围接口应用技术、C 语言程序设计、Keil 集成开发环境及 Proteus ISIS 仿真等内容。

本书既可作为应用型本科、专科院校的单片机原理及实践类课程教材,也可作为单片机应用初学者以及单片机开发人员的实用参考书。

图书在版编目(CIP)数据

MCS-51 单片机原理与应用 / 吴静进, 何尚平, 万彬

主编. -- 重庆: 重庆大学出版社, 2019. 1

高等院校应用型本专科通用规划教材

ISBN 978-7-5689-1402-4

I. ①M… II. ①吴… ②何… ③万… III. ①单片微型计算机—高等学校—教材 IV. ①TP368. 1

中国版本图书馆 CIP 数据核字(2018)第 277643 号

高等院校应用型本专科通用规划教材

MCS-51 单片机原理与应用

主 编 吴静进 何尚平 万彬

策划编辑:曾令维

责任编辑:曾令维 版式设计:曾令维

责任校对:张红梅 责任印制:张策

*

重庆大学出版社出版发行

出版人:易树平

社址:重庆市沙坪坝区大学城西路 21 号

邮编:401331

电话:(023) 88617190 88617185(中小学)

传真:(023) 88617186 88617166

网址:<http://www.cqup.com.cn>

邮箱:fxk@cqup.com.cn (营销中心)

全国新华书店经销

重庆升光电力印务有限公司印刷

*

开本:787mm×1092mm 1/16 印张:18.5 字数:464 千

2019年1月第1版 2019年1月第1次印刷

印数:1—3 000

ISBN 978-7-5689-1402-4 定价:45.00 元

本书如有印刷、装订等质量问题,本社负责调换

版权所有,请勿擅自翻印和用本书

制作各类出版物及配套用书,违者必究

前 言

单片机具有功能强、使用灵活、性价比高、体积小、面向控制等特点,广泛应用于工业控制、智能仪器仪表、现代传感器、数据采集与处理、机电一体化、消费电子、家用电器、通信、办公自动化、医疗器械、计算机控制等领域,自 20 世纪 70 年代问世以来,已经对人类社会做出了巨大贡献。

单片机作为典型、具有代表性的嵌入式系统,随着计算机技术、微电子技术的高速发展,目前各类院校很多专业已经普遍开设了单片机原理及其相关课程,并作为学习各种先进嵌入式微处理器、嵌入式微控制器、片上系统和数字信号处理器的基础。

近几年,随着单片机技术的迅速发展,也出现了很多各具特点的单片机产品,但是从使用数量、技术资料及开发工具等各方面综合考虑,MCS-51 系列单片机仍被广泛使用,具有较大优势。同时,单片机原理作为一门实践性、技术性很强的课程,在学好基础知识、基本技能的同时,最终在于掌握实际应用,因此本书在习近平新时代中国特色社会主义思想指导下,落实“新工科”建设新要求,从单片机基础理论教学和工程实际应用角度出发,以 MCS-51 系列单片机为例,以易学实用为主,注重理论与实际相结合,阐述问题重难点突出,遵循高等教育教学规律,循序渐进地系统讲述了 MCS-51 系列单片机基本概念、内部硬件结构、指令系统及汇编语言程序设计、中断系统、定时/计数器、串行通信技术、外围接口应用技术、C 语言程序设计、Keil 集成开发环境及 Proteus ISIS 仿真等内容。本书可作为单片机初学者以及单片机工程项目开发人员的实用参考书和各大中专院校自动化、电气工程及其自动化、电子信息工程、应用电子技术、通信工程、物联网、测控技术与仪器、机械制造及其自动化、机电一体化、车辆工程、材料成型与控制工程等相关专业的单片机教学及培训用书,并可根据教学实际需要而进行适当取舍和灵活安排。通过本书的学习,学生可掌握单片机应用开发的基础理论知识,提高学生实际操作能力和单片机应用系统的研发设计能力,符合社会对高素质、开拓型单片机人才的需求。

本书由南昌大学科学技术学院吴静进、何尚平和南昌职业学院万彬主编,参加本书编写工作的还有南昌大学科学技术学院许仙明和南昌大学科学技术学院南昌校区全天候开放实验室电机楼 302 的同学们。全书共 9 章,其中,许仙明编写了第 1 章,吴静进编写了第 2 章(除 2.1 节外)、第 4 章、第 5 章、第 6 章,何尚平编写了第 3 章、第 7 章、第 8 章、第 9 章,万彬编写了第 2 章的 2.1 节以及附录 A—D。全书由吴静进负责统稿。

南昌大学科学技术学院罗小青对本书进行了认真审阅,陈艳、朱淑云、黄灿英、吴敏、沈放、黄仁如、陈巍、谢芳娟、谢晖、吴浪武等提出了许多宝贵的意见和建议,特此感谢!

全书参考理论教学 40~60 学时,实验 20 学时。教学时可以根据实际情况,对各章讲授内容进行适当的取舍。

限于编者水平有限,书中不妥之处在所难免,恳请专家、同行老师和读者批评指正。

编 者

2018 年 9 月

目 录

第 1 章 微型计算机基础知识	1
1.1 单片机发展概况	1
1.2 计算机中数的表示方法	4
1.3 计算机系统组成	15
习题	19
第 2 章 MCS-51 单片机硬件的功能结构及内部组成	21
2.1 MCS-51 单片机内部结构及特点	21
2.2 MCS-51 单片机引脚及功能	24
2.3 存储器结构和配置	26
2.4 单片机的并行 I/O 接口	36
2.5 单片机时钟电路与 CPU 时序	41
2.6 空闲和掉电方式	48
习题	50
第 3 章 MCS-51 系列单片机指令系统及汇编语言程序设计	51
3.1 MCS-51 单片机汇编指令系统简介	51
3.2 MCS-51 单片机寻址方式	54
3.3 MCS-51 单片机指令系统	59
3.4 汇编语言程序设计基础	97
3.5 程序设计实例	101
习题	106
第 4 章 MCS-51 单片机的中断系统	108
4.1 中断概述	108
4.2 MCS-51 单片机的中断系统结构	110
4.3 中断处理过程	115
4.4 中断响应后中断请求的撤销	118
4.5 MCS-51 单片机的中断应用举例	118
4.6 外部中断源扩展	123
习题	124



第 5 章 MCS-51 单片机的定时/计数器	126
5.1 定时/计数器概述	126
5.2 定时/计数器的控制	128
5.3 定时/计数器的工作模式及应用	130
5.4 定时/计数器综合应用	138
习题	140
第 6 章 MCS-51 单片机串行通信技术	142
6.1 串行通信基础	142
6.2 MCS-51 单片机串行接口	145
6.3 串行接口工作方式及应用举例	147
6.4 多机通信原理简介	157
习题	162
第 7 章 MCS-51 系列单片机外围接口应用技术	163
7.1 LED 状态指示和数码管显示技术	163
7.2 LCD 液晶显示器接口技术	166
7.3 键盘接口技术	173
7.4 A/D 接口技术	181
7.5 D/A 接口技术	186
7.6 步进电机	191
习题	196
第 8 章 单片机 C 语言程序设计及实例	197
8.1 Keil C51 简介	197
8.2 Keil C51 软件开发结构	197
8.3 Keil C51 与标准 C 语言	199
8.4 运算符与表达式	203
8.5 C51 程序的基本语句	208
8.6 指针	212
8.7 函数	213
8.8 C51 程序结构及应用要点	216
8.9 MCS-51 系列单片机 C 语言程序设计举例	218
习题	243
第 9 章 Keil 集成开发环境及 Proteus ISIS 仿真	244
9.1 Keil 集成开发环境	244
9.2 Proteus ISIS 简介	265

附录.....	279
附录 A MCS-51 单片机汇编指令表	279
附录 B 51 单片机汇编各类指令助记符	283
附录 C CGRAM 和 CGRAM 中字符码与代符图形对应关系	285
附录 D ASCII 表	286
参考文献.....	288

第 1 章 微型计算机基础知识

单片机是把微型计算机的各个功能部件:中央处理器(CPU)、随机存储器(RAM)、只读存储器(ROM)、I/O 接口、定时/计数器以及串行接口等集成在一块芯片上,构成一个完整的微型计算机,因而又称为单片微型计算机(Single Chip Microcomputer)。由于单片机面向控制性应用领域,装入各种智能化产品之中,因此又称为嵌入式微控制器(Embedded Microcontroller)。

1.1 单片机发展概况

单片机从 20 世纪 70 年代诞生以来,发展十分迅速,产品种类也非常多。下面对单片机做初步介绍。

1.1.1 单片机发展的各个阶段

1. 单片机发展的初阶阶段

1976 年,Intel 公司推出了 MCS-48 系列单片机,该系列单片机早期产品在芯片内集成有:8 位 CPU,1KB 程序存储器,64 B 数据存储器,27 根 I/O 口线和 1 个 8 位定时/计数器。这是一种真正的单片机。这个阶段的单片机因受工艺和集成度的限制,品种少,CPU 功能低,存储器容量小,I/O 部件种类和数量少,无串行接口,指令系统功能不强。

2. 性能发展完善阶段

1980 年,Intel 公司推出了 MCS-51 系列单片机。该系列单片机在芯片内集成有:8 位 CPU,4 KB 程序存储器,128 B 数据存储器,4 个 8 位并行接口,1 个全双工串行接口,2 个 16 位定时/计数器,寻址范围 64 KB,并集成有控制功能较强的布尔处理器。这个阶段的单片机结构体系比较完善,性能也大大提高,已开始应用到了各个领域。

3. 微控制器阶段

1982 年,Intel 公司推出 MCS-96 系列单片机,该系列单片机在芯片内集成有:16 位 CPU,8 KB 程序存储器,256 B 数据存储器,5 个 8 位并行接口,1 个全双工接口,2 个 16 位定时/计数器。寻址范围最大为 64 KB,片上还有 8 路 10 位 ADC,1 路 PWM(D/A)输出及高速 I/O 部件等。这个阶段的单片机性能不断完善,性价比显著提高,主要面向测控系统外围电路增强,使单片机可以方便灵活地用于复杂的自动测控系统及设备。因此,越来越倾向于微控制器了。



1.1.2 单片机的多样化产品

迄今为止,世界上的主要芯片厂家已投放市场的单片机产品多达 70 多个系列,500 多个品种。单片机的产品近况可以归纳为以下两个方面。

1.8051 系列单片机产品繁多,主流地位已经形成

通用微型计算机计算速度的提高主要体现在 CPU 位数的提高(16 位、32 位、64 位),而单片机更注重的是产品的可靠性、经济性和嵌入性。因此,单片机 CPU 位数的提高需求并不十分迫切。而多年来的应用实践已经证明,8051 的系统结构合理、技术成熟。因此,许多单片机生产商倾力于提高 8051 单片机的综合功能,从而形成了 8051 的主流产品地位。近年来推出与 8051 兼容的主要产品有:

- ①ATMEL 公司融入 Flash 存储器技术推出的 AT89 系列单片机。
- ②Philips 公司推出的 80C51、80C52 系列高性能单片机。
- ③华邦公司推出的 W78C51、W77C51 系列高速低价单片机。
- ④ADI 公司推出的 AduC8xx 系列高精度 ADC 单片机。
- ⑤LG 公司推出的 GMS90/97 系列低压高速单片机。
- ⑥MAXIM 公司推出的 DS89C420 高速(50MIPS)单片机。
- ⑦Gygnal 公司推出的 C8051F 系列高速 SOC 单片机等。

2. 非 8051 结构单片机不断推出,给用户提供了更为广泛的选择空间

在 8051 及其兼容产品流行的同时,一些单片机芯片生产厂商也推出了一些非 8051 结构的产品,影响较大的有:

- ①Intel 公司推出 MCS-96 系列 16 位单片机。
- ②Microchip 公司推出 PIC 系列 RISC 单片机。
- ③TI 公司推出 MSP430F 系列 16 位低电压、低功耗单片机。
- ④ATMEL 公司推出 AVR 系列 RISC 单片机。

1.1.3 单片机的发展趋势

①微型单片化。随着芯片集成度的提高,为单片机的微型化提供了可能。早期单片机大量使用双列直插式封装,随着贴片工艺的出现,单片机也大量采用了各种符合贴片工艺的封装方式,大大减小了芯片的体积,为嵌入式系统提供了可能。

②低功耗 CMOS 化。为了降低单片机的功耗,现在各单片机生产厂家基本采用了 CMOS 工艺。CMOS 芯片除了低功耗特性之外,还具有功耗的可控性,使单片机可以工作在功耗精细管理状态。但由于其物理特征决定了其工作速度不够高,而 CHMOS 则具备了高速和低功耗的特点,这些特征更适合于在低功耗系统中应用。目前生产的 CHMOS 电路已达到 LSTTL 的速度,传输延迟时间小于 2 ns,其综合优势已超过 TTL 电路。目前单片机的工作电流已降至毫安级甚至微安级,供电电压的下限已达 1~2 V,0.8 V 供电的单片机已经问世。低功耗化的效应不仅功耗低,而且带来了产品的高可靠性、高抗干扰能力及产品的便携化。

③高速化。早期 MCS-51 单片机的典型时钟频率为 12 MHz,目前西门子公司的 C500 系列

单片机的(与 MCS-51 兼容)时钟频率为 36 MHz;EMC 公司的 EM78 系列单片机的时钟频率高达 40 MHz;现在已有更快的 32 位,100 MHz 的单片机产品出现。

④内部资源增加。内部资源的增加除增大片内存储器的容量外,也增加了其他的配置,如 A/D 转换器、脉宽调制输出 PWM、正弦波发生器、CRT 控制器、LED 和 LCD 驱动器、串行通信接口、看门狗电路。

⑤通信及网络功能加强。在某些单片机内部由于封装了局部网络控制模块,甚至将网络协议固化在其内部,因此可以容易地构成网络系统。特别是在控制系统较为复杂时,构成一个控制网络十为有用。目前将单片机嵌入式系统和 Internet 连接起来已是一种趋势。

综上所述,单片机正朝着高性能和多品种的方向发展,但是由于 MCS-51 系列的 8 位单片机仍能满足绝大多数应用领域的需要,所以以 MCS-51 系列为主的 8 位单片机,在现在及以后的相当一段时期内仍然将占据单片机应用市场的主导地位。

1.1.4 单片机的特点及应用领域

1. 单片机的特点

从单片机应用角度来看,其主要特点如下:

①控制性能好,可靠性高。

②体积小,价格低,易于产品化。

从单片机的具体结构和处理过程上看,主要特点如下:

①在存储器结构上,多数单片机的存储器采用哈佛(Harvard)结构。

②在芯片引脚上,大部分采用分时复用技术。

③在内部资源访问上,采用特殊功能寄存器(SFR)的形式。

④在指令系统上,采用面向控制的指令系统。

⑤内部一般都集成一个全双工的串行接口。

⑥单片机有很强的外部扩展能力。

2. 单片机的应用领域

单片机由于具有体积小,功耗低,易于产品化,面向控制,抗干扰能力强,适用温度范围宽,可以方便地实现多机和分布式控制等优点,因而被广泛地应用于以下各种控制系统的分布式系统中:

①工业自动化控制。

②智能仪器仪表。

③计算机外围设备和智能接口。

④家用电器。

⑤多机应用。



1.2 计算机中数的表示方法

数制是人们对事物数量计数的一种统计规律。在日常生活中最常用的是十进制,但在计算机中,由于其电气元件最易实现的是两种稳定状态:器件的“开”与“关”;电平的“高”与“低”。因此,采用二进制数的“0”和“1”可以很方便地表示机器内的数据运算与存储。在编程时,为了方便阅读和书写,人们还经常用八进制数或十六进制来表示二进制数。虽然一个数可以用不同计数制形式表示它的大小,但该数的量值是相等的。

1.2.1 计算机中的数制

1. 进位计数制

当进位计数制采用位置表示法时,同一数字在不同的数位所代表的数值是不同的。每一种进位计数应包含两个基本的因素:

①基数 R (Radix):它代表计数制中所用到的数码个数。如:二进制计数中用到 0 和 1 两个数码;而八进制计数中用到 0~7 共 8 个数码。一般地说,基数为 R 的计数制(简称 R 进制)中,包含 0, 1, \dots , $R-1$ 个数码,进位规律为“逢 R 进 1”。

②位权 W (Weight):在进位计数制中,某个数位的值是由这一位的数码值乘以处在这一位的固定常数决定的,通常把这一固定常数称为位权值,简称位权。各位的位权是以 R 为底的幂。如十进制数基数 $R=10$,则个位、十位、百位上的位权分别为 $10^0, 10^1, 10^2$ 。

一个 R 进制数 N ,可以用以下两种形式表示:

①并列表示法,或称位置计数法:

$$(N)_R = (K_{n-1}K_{n-2}\cdots K_1K_0K_{-1}K_{-2}\cdots K_{-m})_R$$

②多项式表示法,或称以权展开式:

$$(N)_R = K_{n-1}R^{n-1} + K_{n-2}R^{n-2} + \cdots + K_1R^1 + K_0R^0 + K_{-1}R^{-1} + \cdots + K_{-m}R^{-m} = \sum_{i=n-1}^{-m} K_i R^i$$

其中: m, n 为正整数, n 代表整数部分的位数; m 代表小数部分的位数; K_i 代表 R 进制中的任一数码, $0 \leq K_i \leq R-1$ 。

2. 二进制数

二进制数中, $R=2, K_i$ 取 0 或 1,进位规律为“逢 2 进 1”。任一个二进制数 N 可表示为:

$$(N)_2 = K_{n-1}2^{n-1} + K_{n-2}2^{n-2} + \cdots + K_12^1 + K_02^0 + K_{-1}2^{-1} + \cdots + K_{-m}2^{-m}$$

例如: $(1001.101)_2 = 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$

3. 八进制数

八进制, $R=8, K_i$ 可取 0~7 共 8 个数码中的任意 1 个,进位规律为“逢 8 进 1”。任意一个八进制数 N 可以表示为:

$$(N)_8 = K_{n-1}8^{n-1} + K_{n-2}8^{n-2} + \cdots + K_18^1 + K_08^0 + K_{-1}8^{-1} + \cdots + K_{-m}8^{-m}$$

例如: $(246.12)_8 = 2 \times 8^2 + 4 \times 8^1 + 6 \times 8^0 + 1 \times 8^{-1} + 2 \times 8^{-2}$



4. 十六进制数

十六进制数, $R=16$, K_i 可取 $0 \sim 15$ 共 16 个数码中的任一个, 但 $10 \sim 15$ 分别用 A、B、C、D、E、F 表示, 进位规律为“逢 16 进 1”。任意一个十六进制数 N 可表示为:

$$(N)_{16} = K_{n-1}16^{n-1} + K_{n-2}16^{n-2} + \cdots + K_116^1 + K_016^0 + K_{-1}16^{-1} + \cdots + K_{-m}16^{-m}$$

例如: $(2D07.A)_{16} = 2 \times 16^3 + 13 \times 16^2 + 0 \times 16^1 + 7 \times 16^0 + 10 \times 16^{-1}$

表 1-1 给出了以上 3 种进制数与十进制数的对应关系。为避免混淆, 除用 $(N)_R$ 的方法区分不同进制数外, 还常用数字后加字母作为标注的方法。其中字母 B (Binary) 表示二进制数; 字母 Q (Octal 的缩写为字母 O, 为区别数字 0 故写成 Q) 表示八进制数; 字母 D (Decimal) 或不加字母表示十进制数; 字母 H (Hexadecimal) 表示十六进制数。

表 1-1 二、八、十、十六进制数码对应表

十进制	二进制	八进制	十六进制
0	0000B	0Q	0H
1	0001B	1Q	1H
2	0010B	2Q	2H
3	0011B	3Q	3H
4	0100B	4Q	4H
5	0101B	5Q	5H
6	0110B	6Q	6H
7	0111B	7Q	7H
8	1000B	10Q	8H
9	1001B	11Q	9H
10	1010B	12Q	0AH
11	1011B	13Q	0BH
12	1100B	14Q	0CH
13	1101B	15Q	0DH
14	1110B	16Q	0EH
15	1111B	17Q	0FH
16	10000B	20Q	10H

1.2.2 数制的转换

1. 各种进制数间的相互转换

(1) 各种进制数转换成十进制数

各种进制数转换成十进制数的方法是: 将各进制数先按位权展开成多项式, 再利用十进制运算法则求和, 即可得到该数对应的十进制数。

例 1.1 将数 $1001.101B$, $246.12Q$, $2D07.AH$ 转换为十进制数。

$$\begin{aligned} 1001.101B &= 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= 8 + 1 + 0.5 + 0.125 = 9.625 \end{aligned}$$

$$246.12Q = 2 \times 8^2 + 4 \times 8^1 + 6 \times 8^0 + 1 \times 8^{-1} + 2 \times 8^{-2}$$

$$= 128 + 32 + 6 + 0.125 + 0.03125 = 166.15625$$

$$2D07.AH = 2 \times 16^3 + 13 \times 16^2 + 0 \times 16^1 + 7 \times 16^0 + 10 \times 16^{-1}$$

$$= 8192 + 3328 + 7 + 0.625 = 11527.625$$

(2) 十进制数转换为二、八、十六进制数

任一十进制数 N 转换成 q 进制数, 先将整数部分与小数部分分为两部分, 并分别进行转换, 然后再用小数点将这两部分连接起来。

① 整数部分转换

整数部分转换步骤为:

第 1 步: 用 q 去除 N 的整数部分, 得到商和余数, 记余数为 q 进制整数的最低位数码 K_0 ;

第 2 步: 再用 q 去除得到的商, 求出新的商和余数, 余数又作为 q 进制整数的次低位数码 K_1 ;

第 3 步: 再用 q 去除得到的新商, 求出相应的商和余数, 余数作为 q 进制整数的下一位数码 K_i ;

第 4 步: 重复第 3 步, 直至商为零, 整数转换结束。此时, 余数作为转换后 q 进制整数的最高位数码 K_{n-1} 。

2	168	余数 0,	$K_0 = 0$
2	84	余数 0,	$K_1 = 0$
2	42	余数 0,	$K_2 = 0$
2	21	余数 1,	$K_3 = 1$
2	10	余数 0,	$K_4 = 0$
2	5	余数 1,	$K_5 = 1$
2	2	余数 0,	$K_6 = 0$
2	1	余数 1,	$K_7 = 1$

所以 $168 = 10101000B$

② 小数部分转换

小数部分转换步骤为:

第 1 步: 用 q 去乘 N 的纯小数部分, 记下乘积的整数部分, 作为 q 进制小数的第 1 个数码 K_{-1} ;

第 2 步: 再用 q 去乘上次积的纯小数部分, 得到新乘积的整数部分, 记为 q 进制小数的次位数码 K_{-i} ;

第 3 步: 重复第 2 步, 直至乘积的小数部分为零, 或者达到所需要的精度位数为止。此时, 乘积的整数位作为 q 进制小数位的数码 K_{-m} 。

例 1.2 将 0.686 转换成二、八、十六进制数(用小数点后 5 位表示)。

$$0.686 \times 2 = 1.372 \quad K_{-1} = 1 \quad 0.686 \times 8 = 5.488 \quad K_{-1} = 5 \quad 0.686 \times 16 = 10.976 \quad K_{-1} = A$$

$$0.372 \times 2 = 0.744 \quad K_{-2} = 0 \quad 0.488 \times 8 = 3.904 \quad K_{-2} = 3 \quad 0.976 \times 16 = 15.616 \quad K_{-2} = F$$

$$0.744 \times 2 = 1.488 \quad K_{-3} = 1 \quad 0.904 \times 8 = 7.232 \quad K_{-3} = 7 \quad 0.616 \times 16 = 9.856 \quad K_{-3} = 9$$

$$0.488 \times 2 = 0.976 \quad K_{-4} = 0 \quad 0.232 \times 8 = 1.856 \quad K_{-4} = 1 \quad 0.856 \times 16 = 13.696 \quad K_{-4} = D$$



$0.976 \times 2 = 1.952 \quad K_{-5} = 1 \quad 0.856 \times 8 = 6.848 \quad K_{-5} = 6 \quad 0.696 \times 16 = 11.136 \quad K_{-5} = B$
 $0.686 \approx 0.10101B \quad 0.686 \approx 0.53716Q \quad 0.686 \approx 0.AF9DBH$

从以上例子可以看出,二进制表示的数越精确,所需的数位就越多,这样,不利于书写和记忆,而且容易出错。另外,若用同样数位表示数,则八、十六进制数所表示数的精度较高。所以在汇编语言编程中常用八进制或十六进制数作为二进制数的编码,来书写和记忆二进制数,便于人机信息交换。在 MCS-51 系列单片机编程中,通常采用十六进制数。

(3) 二进制数与八进制数之间的相互转换

由于 $2^3 = 8$,故可采用“合3为1”的原则,即从小数点开始分别向左、右两边各以3位为1组进行二一八换算;若不足3位的以0补足,便可将二进制数转换为八进制数。

例 1.3 将 1111011.0101B 转换为八进制数。

解:根据“合3为1”和不足3位以0补足的原则,将此二进制数书写为:

```
001 111 011 . 010 100
  1   7   3   .   2   4
```

因此,其结果为 $1111011.0101B = 173.24Q$ 。

例 1.4 将 1357.246Q 转换成二进制数。

解:根据“1分为3”的原则,可将该十进制数书写为:

```
  1   3   5   7   .   2   4   6
001 011 101 111 . 010 100 110
```

其结果为 $1357.246Q = 1011101111.01010011B$ 。

(4) 二进制数与十六进制数之间的相互转换

由于 $2^4 = 16$,可采用“合4为1”的原则,从小数点开始分别向左、右两边各以4位为1组进行二一十六换算;若不足4位以0补足,便可将二进制数转换为十六进制数。

例 1.5 将 1101000101011.001111B 转换成十六进制数。

解:根据“合4为1”的原则,可将该二进制数书写为:

```
0001 1010 0010 1011 .0011 1100
  1     A   2   B . 3     C
```

其结果为 $1101000101011.001111B = 1A2B.3CH$ 。

反之,采用“1分为4”的原则,每位十六进制数用4位二进制数表示,便可将十六进制数转换为二进制数。

例 1.6 将 4D5E.6FH 转换成二进制数。

解:根据“1分为4”的原则,可将该十六进制数书写为:

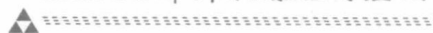
```
  4     D   5     E   . 6     F
0100 1101 0101 1110 . 0110 1111
```

其结果为 $4D5E.6FH = 100110101011110.01101111B$ 。

1.2.3 二进制数的运算

1. 二进制数的算术运算

二进制数不仅物理上容易实现,而且算术运算也比较简单,其加、减法遵循“逢2进1”“借



1 当 2”的原则。

以下通过 4 个例子说明二进制数的加、减、乘、除运算过程。

(1) 二进制加法

1 位二进制数的加法规则为：

$$0+0=0; \quad 0+1=1; \quad 1+0=1; \quad 1+1=10 \quad (\text{有进位})$$

例 1.7 求 $11001010\text{B} + 11101\text{B}$ 。

解：	被加数	11001010
	加数	11101
	进位	(00110000)
	和	11100111

则 $11001010\text{B} + 11101\text{B} = 11100111\text{B}$ 。

由此可见,两个二进制数相加时,每 1 位有 3 个数参与运算(本位被加数、加数、低位进位),从而得到本位和以及向高位的进位。

(2) 二进制减法

1 位二进制数减法规则为：

$$1-0=1; \quad 1-1=0; \quad 0-0=0; \quad 0-1=1 \quad (\text{有借位})$$

例 1.8 求 $10101010\text{B} - 10101\text{B}$ 。

解：	被减数	10101010
	减数	10101
	借位	(00101010)
	差	10010101

则 $10101010\text{B} - 10101\text{B} = 10010101\text{B}$ 。

(3) 二进制乘法

1 位二进制乘法规则为：

$$0 \times 0 = 0 \quad 0 \times 1 = 0 \quad 1 \times 0 = 0 \quad 1 \times 1 = 1$$

例 1.9 求 $110011\text{B} \times 1011\text{B}$ 。

解：	被乘数	110011
	乘数	1011
		110011
		110011
		000000
		110011
	积	1000110001

则 $110011\text{B} \times 1011\text{B} = 1000110001\text{B}$ 。

由运算过程可以看出,二进制数乘法与十进制数乘法类似,显然,这种算法计算机实现时很不方便,一般计算机中不采用这种算法。对于没有乘法指令的微型计算机来说,常采用比较、相加、与部分积右移相结合的方法进行编程来实现乘法运算。

(4) 二进制除法

二进制除法的运算过程类似于十进制除法的运算过程。



例 1.10 求 $100100B \div 101B$ 。

解:

$$\begin{array}{r}
 000111 \\
 \sqrt{100100} \\
 \underline{101} \\
 1000 \\
 \underline{101} \\
 0110 \\
 \underline{101} \\
 1
 \end{array}$$

则 $100100B \div 101B$ 商为 $111B$, 余数为 $1B$ 。

二进制数除法是二进制数乘法的逆运算, 在没有除法指令的微型计算机中, 常采用比较、相减、余数左移相结合的方法进行编程来实现除法运算。由于 MCS-51 系列单片机指令系统中包含有加、减、乘、除指令, 因此给用户编程带来了许多方便, 同时也提高了机器的运算效率。

2. 二进制数的逻辑运算

(1) “与”运算(AND)

“与”运算又称逻辑乘, 运算符为 \cdot 或 \wedge 。“与”运算的规则如下:

$$0 \wedge 0 = 0 \quad 0 \wedge 1 = 1 \wedge 0 = 0 \quad 1 \wedge 1 = 1$$

例 1.11 若二进制数 $X = 10101111B$, $Y = 01011110B$, 求 $X \wedge Y$ 。

$$\begin{array}{r}
 10101111 \\
 \wedge \quad 01011110 \\
 \hline
 00001110
 \end{array}$$

则 $X \wedge Y = 00001110B$ 。

(2) “或”运算(OR)

“或”运算又称逻辑加, 运算符为 $+$ 或 \vee 。“或”运算的规则如下:

$$0 \vee 0 = 0 \quad 0 \vee 1 = 1 \vee 0 = 1 \quad 1 \vee 1 = 1$$

例 1.12 若二进制数 $X = 10101111B$, $Y = 01011110B$, 求 $X \vee Y$ 。

$$\begin{array}{r}
 10101111 \\
 \vee \quad 01011110 \\
 \hline
 11111111
 \end{array}$$

则 $X \vee Y = 11111111B$ 。

(3) “非”运算(NOT)

“非”运算又称逻辑非, 如变量 A 的“非”运算记作 \bar{A} 。“非”运算的规则如下:

$$\bar{1} = 0 \quad \bar{0} = 1$$

例 1.13 若二进制数 $A = 10101111B$, 求 \bar{A} 。

$$\bar{A} = \overline{10101111B} = 01010000B$$

由此可见, 逻辑“非”可使 A 中各位结果均发生反变化, 即 0 变 1 , 1 变 0 。

(4) “异或”运算(XOR)

“异或”运算的运算符为 \oplus , 其运算规则如下:

$$0 \oplus 0 = 0 \quad 0 \oplus 1 = 1 \quad 1 \oplus 1 = 0 \quad 1 \oplus 0 = 1$$

例 1.14 若二进制数 $X = 10101111\text{B}$, $Y = 01011110\text{B}$, 求 $X \oplus Y$ 。

$$\begin{array}{r} 10101111 \\ \oplus 01011110 \\ \hline 11110001 \end{array}$$

则 $X \oplus Y = 11110001\text{B}$ 。

1.2.4 数及字符在计算机内的编码

1. 机器数与真值

实际的数值是带有符号的,既可能是正数,也可能是负数,前者符号用“+”号表示,后者符号用“-”号表示,运算的结果也可能是正数,也可能是负数。于是在计算机中就存在着如何表示正、负数的问题。

由于计算机只能识别 0 和 1,因此,在计算机中通常把一个二进制数的最高位作为符号位,以表示数值的正与负(若用 8 位表示一个数,则 D7 位为符号位;若用 16 位表示一个数,则 D15 位为符号位),并用 0 表示“+”,用 1 表示“-”。

例如: $N1 = +1011$, 在计算机中用 8 位二进制数可表示为:

数值 符号	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	1	0	1	1

2. 原码、补码与反码

1) 原码

正数的符号位用 0 表示,负数的符号位用 1 表示,数值部分用真值的绝对值来表示的二进制机器数称为原码,用 $[X]_{\text{原}}$ 表示。

(1) 正数的原码

若真值为正数 $X = +K_{n-2}K_{n-3}\cdots K_1K_0$ (即 $n-1$ 位二进制正数),

$$\text{则 } [X]_{\text{原}} = 0 K_{n-2}K_{n-3}\cdots K_1K_0$$

(2) 负数的原码

若真值为负数 $X = -K_{n-2}K_{n-3}\cdots K_1K_0$ (即 $n-1$ 位二进制负数),

$$\begin{aligned} \text{则 } [X]_{\text{原}} &= 1 K_{n-2}K_{n-3}\cdots K_1K_0 \\ &= 2^{n-1} + K_{n-2}K_{n-3}\cdots K_1K_0 \\ &= 2^{n-1} - (-K_{n-2}K_{n-3}\cdots K_1K_0) \\ &= 2^{n-1} - X \end{aligned}$$

例如: $+115$ 和 -115 在计算机中(设机器字长为 8 位)的原码可分别表示为:

$$[+115]_{\text{原}} = 01110011\text{B}; [-115]_{\text{原}} = 11110011\text{B}$$

2) 补码与反码

(1) 补码的概念

在日常生活中有许多“补”数的事例。如钟表,假设标准时间为 6 点整,而某钟表却指在 9