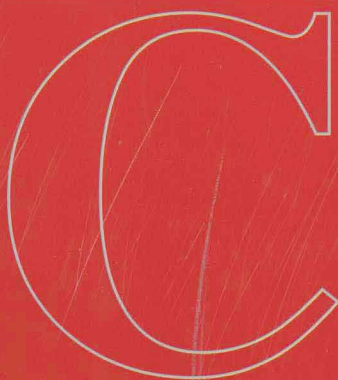


21世纪高等学校计算机基础实用规划教材

单片机原理、汇编 与C51及接口技术 (第2版)



朱定华 戴颖颖 朱悦 编著



清华大学出版社

21 世纪高等学校计算机**基础**实用规划教材

单片微机原理、汇编 与C51及接口技术

(第2版)

朱定华 戴颖颖 朱悦 编著

清华大学出版社

内 容 简 介

本书以 MCS-51 单片机为背景机,系统地介绍了微型计算机的原理及接口技术,主要包括微型计算机基础、汇编语言程序设计、MCS-51 单片机的内部接口、MCS-51 单片机的扩展方法、存储器及其扩展、常用可编程接口、AD 和 DA 转换芯片及 C51 程序设计等。

本书内容精练,实例丰富。其中大量的接口电路和程序是作者多年在科研和教学中反复提炼得来的,因而本书实用性很强。书中内容系统全面,论述深入浅出,循序渐进。可作为大专院校的“单片机”、“微机原理及其应用”或“微机原理及接口技术”等课程的教学用书,也可以作为参加各类电子制作、课程设计和毕业设计的教学参考书,还可供从事电子技术,计算机应用与开发的科研人员和工程技术人员使用参考。

使用本教材的读者如有什么困难,请到本教材作者编写的辅导书《单片微机原理、汇编与 C51 及接口技术——学习指导与实验》中去寻求帮助。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

单片微机原理、汇编与 C51 及接口技术/朱定华等编著.--2 版.--北京:清华大学出版社,2013

21 世纪高等学校计算机基础实用规划教材

ISBN 978-7-302-31856-9

I. ①单… II. ①朱… 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(2013)第 066320 号

责任编辑:魏江江 赵晓宁

封面设计:傅瑞学

责任校对:李建庄

责任印制:沈 露

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 刷 者:北京富博印刷有限公司

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:21 字 数:521 千字

版 次:2010 年 6 月第 1 版 2013 年 9 月第 2 版 印 次:2013 年 9 月第 1 次印刷

印 数:1~2000

定 价:39.00 元

产品编号:051996-01

出版说明

随着我国改革开放的进一步深化,高等教育也得到了快速发展,各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,加大了使用信息科学等现代科学技术提升、改造传统学科专业的投入力度,通过教育改革合理调整和配置了教育资源,优化了传统学科专业,积极为地方经济建设输送人才,为我国经济社会的快速、健康和可持续发展以及高等教育自身的改革发展做出了巨大贡献。但是,高等教育质量还需要进一步提高以适应经济社会发展的需要,不少高校的专业设置和结构不尽合理,教师队伍整体素质亟待提高,人才培养模式、教学内容和方法需要进一步转变,学生的实践能力和创新精神亟待加强。

教育部一直十分重视高等教育质量工作。2007年1月,教育部下发了《关于实施高等学校本科教学质量与教学改革工程的意见》,计划实施“高等学校本科教学质量与教学改革工程(简称‘质量工程’)”,通过专业结构调整、课程教材建设、实践教学改革、教学团队建设等多项内容,进一步深化高等学校教学改革,提高人才培养的能力和水平,更好地满足经济社会发展对高素质人才的需要。在贯彻和落实教育部“质量工程”的过程中,各地高校发挥师资力量强、办学经验丰富、教学资源充裕等优势,对其特色专业及特色课程(群)加以规划、整理和总结,更新教学内容、改革课程体系,建设了一大批内容新、体系新、方法新、手段新的特色课程。在此基础上,经教育部相关教学指导委员会专家的指导和建议,清华大学出版社在多个领域精选各高校的特色课程,分别规划出版系列教材,以配合“质量工程”的实施,满足各高校教学质量和教学改革的需要。

本系列教材立足于计算机公共课程领域,以公共基础课为主、专业基础课为辅,横向满足高校多层次教学的需要。在规划过程中体现了以下一些基本原则和特点。

(1) 面向多层次、多学科专业,强调计算机在各专业中的应用。教材内容坚持基本理论适度,反映各层次对基本理论和原理的需求,同时加强实践和应用环节。

(2) 反映教学需要,促进教学发展。教材要适应多样化的教学需要,正确把握教学内容和课程体系的改革方向,在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养,为学生的知识、能力、素质协调发展创造条件。

(3) 实施精品战略,突出重点,保证质量。规划教材把重点放在公共基础课和专业基础课的教材建设上;特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版,逐步形成精品教材;提倡并鼓励编写体现教学质量和教学改革成果的教材。

(4) 主张一纲多本,合理配套。基础课和专业基础课教材配套,同一门课程可以有针对不同层次、面向不同专业的多本具有各自内容特点的教材。处理好教材统一性与多样化,基本教材与辅助教材、教学参考书,文字教材与软件教材的关系,实现教材系列资源配套。

(5) 依靠专家,择优选用。在制定教材规划时依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时,要引入竞争机制,通过申报、评审确定主题。书稿完成后要认真实行审稿程序,确保出书质量。

繁荣教材出版事业,提高教材质量的关键是教师。建立一支高水平教材编写梯队才能保证教材的编写质量和建设力度,希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21 世纪高等学校计算机基础实用规划教材

联系人:魏江江 weijj@tup.tsinghua.edu.cn

前 言

当今微型计算机技术发展形成两大分支,一大分支是以微处理器(Micro Processor Unit)为核心所构成的通用微机系统;另一大分支是微控制器(Micro Controller Unit),俗称单片机。面对这样“势均力敌”的两大分支,大专院校的“微机原理”课程究竟应选用何种机型为背景机来组织教学,也出现分歧。作为 21 世纪的工科大学生,既要掌握通用微机,又要掌握单片机,所以有些学校在学习以通用微机为背景机的“微机原理”课程后,又开设“单片机及接口技术”选修课。

MCS-51 单片机应用于控制目的时,功能已足够强大,已能满足控制领域中多数场合的要求。MCS-51 单片机正朝着高速、高性能和多功能方向发展。Philips 公司开发的 8xC552 单片机和 Cygnal 公司开发的 C8051Fxxx 单片机都以 80C51 为基础结构,并与 80C51 兼容。因此,选用 MCS-51 系列单片机作为“单片微机”课程的背景机,既可满足教学内容稳定,实验设备成熟便宜,又不失其先进性与实用性。本书在附录中较详细地介绍了 8xC552 单片机和 C8051Fxxx 单片机。

本书全面地介绍了 MCS-51 单片机的结构原理和应用技术,全书共 8 章,还有 5 个附录。第 1 章介绍微型计算机的基础知识,包括计算机中的数制和编码、微型计算机和单片机的结构和主要系列产品的特性。第 2 章介绍 MCS-51 单片机常用的汇编指令和伪指令以及指令的时序。第 3 章介绍汇编语言程序设计的基本技术。通过第 2 和第 3 章的学习,使读者能更透彻地了解汇编语言程序设计,为编程应用打下基础。第 4 章介绍 MCS-51 单片机的内部接口电路,包括中断系统、定时器、并行口和串行口,同时还介绍了计算机间的通信。第 5 章介绍 MCS-51 单片机的最小应用系统和扩展技术。第 6 章介绍存储器及其与微型计算机的接口技术。第 7 章介绍常用可编程接口芯片的功能与应用。第 8 章介绍 AD 和 DA 转换器与微型计算机的接口与应用。附录 A 介绍 MCS-51 单片机的 C 语言——C51,这部分内容是本教材第 1 版的第 9 章。附录中还可较详细地介绍了以 80C51 为基础结构并与其兼容的 8xC552 单片机和 C8051Fxxx 单片机。本教材第 1 版中的单片机实验汇入《单片微机原理、汇编与 C51 及接口技术——学习指导与实验》中。

本书每章后均附有思考题与习题。编写本书时,注意了理论和实践相结合,力求做到既有一定的理论基础,又能运用理论解决实际问题;既掌握一定的先进技术,又着眼于当前的应用服务。

本课程的参考学时数为 64 学时(不含实验)。

本书由朱定华、戴颖颖和朱悦编写。参加本书编写工作的人员还有蔡苗、蔡红娟、翟晟、周斌、黄松、吕建才、陈艳、林卫、程萍、张德芳、林威、李志文等。

由于作者水平有限,书中难免还存在一些不足和错误,殷切希望广大读者批评指正。

编者

2013 年 5 月于武昌

目 录

第 1 章 微型计算机基础	1
1.1 计算机中的数和编码	1
1.1.1 计算机中的数制	1
1.1.2 符号数的表示法	2
1.1.3 二进制数的加减运算	5
1.1.4 二进制数的逻辑运算与逻辑电路	7
1.1.5 二进制编码	9
1.1.6 BCD 数的加减运算	11
1.2 微型计算机和单片机	12
1.2.1 微型计算机常用的术语	12
1.2.2 微型计算机	12
1.2.3 单片机	15
1.3 MCS-51 单片机及其兼容单片机的基本组成和存储器配置	15
1.3.1 8051 单片机的基本组成	15
1.3.2 MCS-51 单片机的存储器	17
1.3.3 特殊功能寄存器	19
1.4 MCS-51 系列单片机及兼容机	22
1.4.1 51 子系列和 52 子系列	22
1.4.2 AT89 系列单片机	23
1.4.3 MCS-51 单片机的引线	25
习题与思考题	26
第 2 章 汇编语言与汇编程序	28
2.1 符号指令的寻址方式	28
2.2 常用指令	31
2.2.1 数据传送类指令	31
2.2.2 加减运算指令	34
2.2.3 逻辑运算及移位类指令	37
2.2.4 位操作指令	40
2.2.5 指令应用举例	41

2.3 伪指令	43
2.3.1 常量和标号	44
2.3.2 程序的定位和结束伪指令	45
2.4 指令的时序	46
习题与思考题	49
第 3 章 汇编语言程序设计的基本技术	52
3.1 顺序程序设计	52
3.1.1 乘除法指令	52
3.1.2 BCD 数加法调整指令 DAA	54
3.1.3 顺序程序设计举例	55
3.2 分支程序设计	57
3.2.1 条件转移指令	57
3.2.2 比较不等转移指令	58
3.2.3 无条件转移指令	59
3.2.4 应用举例	61
3.3 循环程序设计	62
3.3.1 减 1 非零转移指令 DJNZ	63
3.3.2 单重循环程序设计举例	64
3.3.3 多重循环程序	67
3.4 子程序设计	69
3.4.1 子程序的概念	69
3.4.2 子程序的调用指令与返回指令	71
3.4.3 子程序及其调用程序设计举例	72
习题与思考题	76
第 4 章 MCS-51 单片机内部接口电路	78
4.1 接口的基本概念	78
4.1.1 接口电路的功能	78
4.1.2 接口控制原理	79
4.1.3 串行接口 UART、SPI 和 I ² C/SMBus	80
4.1.4 并行接口	82
4.2 中断及 MCS-51 单片机的中断系统	82
4.2.1 中断和中断处理	82
4.2.2 MCS-51 单片机的中断系统	84
4.2.3 多个外部中断源的系统设计	87
4.3 定时器	88
4.3.1 定时器的结构	89
4.3.2 定时器的工作方式	90

4.3.3 定时器应用举例	93
4.4 并行输入输出接口	96
4.5 UART 串行输入输出接口	105
4.6 串行通信	111
4.6.1 双机通信	111
4.6.2 多机通信	112
4.6.3 MCS-51 单片机与 80x86 微型计算机的通信	118
4.7 52 子系列单片机	122
4.7.1 内部 RAM	122
4.7.2 定时器 T2	123
习题与思考题	128
第 5 章 单片机的最小应用系统与外部扩展	130
5.1 单片机的最小应用系统	130
5.1.1 单片机的时钟电路	130
5.1.2 复位电路及复位状态	131
5.1.3 MCS-51 单片机的片外总线结构	132
5.1.4 89S51 单片机的最小应用系统	134
5.1.5 8031 单片机的最小应用系统	134
5.2 单片机的外部扩展	134
5.2.1 外部扩展芯片与地址总线的连接	135
5.2.2 外部扩展芯片与数据总线的连接	137
5.2.3 外部扩展芯片与控制总线的连接	137
5.3 用 TTL 或 CMOS 芯片扩展简单的 I/O 接口	138
5.3.1 用寄存器扩展简单的输出接口	138
5.3.2 用三态缓冲器扩展输入接口	139
5.3.3 三态缓冲寄存器扩展输入输出接口	140
5.3.4 应用举例	141
5.3.5 开关量的驱动与隔离	146
习题与思考题	148
第 6 章 存储器及其扩展	149
6.1 存储器概述	149
6.2 常用的存储器芯片	150
6.2.1 存储器芯片的结构	150
6.2.2 随机读写存储器	151
6.2.3 只读存储器	152
6.3 存储器的扩展	154
6.4 串行 EEPROM 存储器及其应用	158

6.4.1	24LC××	159
6.4.2	X25043/45	169
6.4.3	X84041	177
习题与思考题		181

第 7 章 常用可编程接口 182

7.1	可编程并行接口 8255	182
7.1.1	8255 的组成与接口信号	182
7.1.2	8255 的工作方式与控制字	183
7.1.3	3 种工作方式的功能	186
7.1.4	8255 在液晶显示器接口中的应用	190
7.2	可编程计数器/定时器 8253	197
7.2.1	8253 的组成与接口信号	198
7.2.2	计数器的工作方式及其与输入输出的关系	199
7.2.3	8253 的控制字和初始化编程	201
7.2.4	8253 的应用	202
7.3	可编程多功能接口 8155	204
7.3.1	8155 的组成与接口信号	204
7.3.2	8155 的命令状态字	206
7.3.3	8155 与 MCS-51 单片机的连接	207
7.3.4	8155 在键盘和七段显示器接口中的应用	209
7.4	键盘/显示控制器 8279	212
7.4.1	8279 的组成与接口信号	212
7.4.2	8279 的操作命令	214
7.4.3	8279 在键盘和显示器接口中的应用	215
7.5	HD7279 键盘、显示器接口	219
7.5.1	HD7279 的引线排列和接口信号	219
7.5.2	HD7279 的操作命令	219
7.5.3	HD7279 的操作时序	221
7.5.4	HD7279 的应用举例	222
习题与思考题		224

第 8 章 模拟通道接口 225

8.1	数模转换器及其与微型计算机的接口	225
8.1.1	8 位数模转换芯片 DAC0832	226
8.1.2	12 位数模转换芯片 DAC1210	233
8.1.3	10 位 D/A 转换器 AD7520	235
8.2	模数转换器及其与微型计算机的接口	235
8.2.1	8 位逐次逼近式 A/D 转换芯片 ADC0808	236

8.2.2	12 位逐次比较式数模转换芯片 AD574	237
8.3	串行模数转换器和数模转换器及其与单片微机的接口	240
8.3.1	串行模数转换器 TLC2543	240
8.3.2	串行输入 12 位电压输出数/模转换器 TLV5616	244
	习题与思考题	247
附录 A	MCS-51 单片机 C 语言——C51	248
附录 B	8xC552 单片机	267
附录 C	C8051Fxxx 单片机	281
附录 D	MCS-51 指令系统表	316
附录 E	MCS-51 指令速查表	320

1.1 计算机中的数和编码

1.1.1 计算机中的数制

计算机最早是作为一种计算工具出现的,所以其最基本的功能是对数进行加工和处理。数在机器中是以器件的物理状态来表示的。一个具有两种不同的稳定状态且能相互转换的器件就可以用来表示 1 位二进制数。二进制数有运算简单、便于物理实现、节省设备等优点,所以目前在计算机中数几乎全是采用二进制表示。但是二进制数书写起来太长,且不利于阅读和记忆;而 4 位二进制数有 16 个不同的状态 0000~1111,即是 1 位十六进制数;所以微型计算机中的二进制数都采用十六进制数来缩写。十六进制数用 0~9 和 A~F 等 16 个数码表示 4 位二进制数 0000~1111,这 16 个二进制数 0000~1111 的大小就是十进制数 0~15。1 个 8 位的二进制数用 2 位十六进制数表示,1 个 16 位的二进制数用 4 位十六进制数表示等。这样书写方便,又便于阅读和记忆,且转换方便,因此常用十六进制数来缩写二进制数。然而人们最熟悉、最常用的是十进制数。为此,要熟练地掌握十进制数、二进制数和十六进制数间的相互转换。它们之间的关系如表 1-1 所列。

表 1-1 十进制数、二进制数及十六进制数对照表

十进制	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
二进制	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
十六进制	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

为了区别十进制数、二进制数及十六进制数 3 种数制,可在数的右下角注明数制,或者在数的后面加一字母。如 B(binary)表示二进制数制;D(decimal 或不加字母)表示十进制数制;H(hexadecimal)表示十六进制数制。

1. 二进制数和十六进制数整数间的相互转换

根据表 1-1 所列的对应关系即可实现它们之间的转换。

二进制整数转换为十六进制数,其方法是从右(最低位)向左将二进制数分组:每 4 位为 1 组,最后一组若不足 4 位则在其左边添加 0,以凑成 4 位 1 组,每组用 1 位十六进制数表示。例如:

1111111000111B → 1 1111 1100 0111B → 0001 1111 1100 0111B=1FC7H

十六进制数转换为二进制数,只需用 4 位二进制数代替 1 位十六进制数即可。例如:

3AB9H=0011 1010 1011 1001B

2. 十六进制数和十进制数间的相互转换

十六进制数转换为十进制数十分简单,只需将十六进制数按权展开相加即可。例如:

$$1F3DH = 16^3 \times 1 + 16^2 \times 15 + 16^1 \times 3 + 16^0 \times 13$$
$$= 4096 \times 1 + 256 \times 15 + 16 \times 3 + 1 \times 13 = 4096 + 3840 + 48 + 13 = 7997$$

十进制整数转换为十六进制数可用除 16 取余法,即用 16 不断地去除待转换的十进制数,直至商等于 0 为止。将所得的各次余数,依倒序排列,即可得到所转换的十六进制数。如将 38947 转换为十六进制数,其方法及算式如下:

$$\begin{array}{r} 16 \overline{) 38947} \quad 3 \\ \underline{48000} \\ 16 \overline{) 2434} \quad 2 \\ \underline{3200} \\ 16 \overline{) 152} \quad 8 \\ \underline{128} \\ 16 \overline{) 24} \quad 9 \\ \underline{144} \\ 0 \end{array}$$

即 $38947 = 9823H$ 。

3. 二进制数和十进制数整数间的相互转换

把一个十进制数转换为二进制数,可以先把该数转换为十六进制数,然后再转换为二进制数,这样可以减少计算次数;反之,要把一个二进制数转换为十进制数,也可以采用同样的办法。若使用 2^n (2^n 的二进制数等于 1 后跟 n 个 0) 和十六进制数、十进制数的对应关系(见表 1-2)以及个别十进制整数和十六进制数的对应关系(如 $50 = 32H$ 、 $80 = 50H$ 、 $100 = 64H$ 等),则转换起来更为方便。例如:

$$38947 = 32768 + 4096 + 2048 + 32 + 3 = 8000H + 1000H + 800H + 20H + 3H = 9823H$$
$$1F3DH = 2000H - (80H + 40H + 3H) = 8192 - (128 + 64 + 3) = 7997$$

表 1-2 部分二进制数与十进制数的对应关系

	2^1	2^5	2^6	2^7	2^8	2^9	2^{10}	2^{11}	2^{12}	2^{13}	2^{14}	2^{15}	2^{16}	2^{20}	2^{30}	2^{40}	2^{50}	2^{60}
十六进制数	20	40	80	100	200	400	800	1000	2000	4000	8000	10 000						
十进制数	32	64	128	256	512	1024	2048	4096	8192	16 384	32 768	65 536						
常用的缩写							1K	2K	4K	8K	16K	32K	64K	1M	1G	1T	1P	1E

1.1.2 符号数的表示法

1. 机器数与真值

二进制数与十进制数一样,有正负之分。在计算机中,常用数的符号和数值部分一起编码的方法表示符号数。常用的有原码、反码和补码表示法。这几种表示法都将数的符号数码化。通常正号用 0 表示,负号用 1 表示。为了区分一般书写时表示的数和机器中编码表示的数,称前者为真值,后者为机器数,即数值连同符号数码 0 或 1 一起作为一个数,就称为机器数,而它的数值连同符号 + 或 - 称为机器数的真值。把机器数的符号位也当作数值的数,就是无符号数。

为了表示方便,常把 8 位二进制数称为字节,把 16 位二进制数称为字,把 32 位二进制

数称为双字。对于机器数应将其用字节、字或双字表示,所以只有8位、16位或32位机器数的最高位才是符号位。

2. 原码

按上所述,数值用其绝对值,正数的符号位用0表示,负数的符号位用1表示,这样表示的数就称为原码。例如:

105 的真值是 +01101001B

105 的原码是 01101001B

-105 的真值是 -01101001B

-105 的原码是 1101001B

其中最高位为符号位,后面7位是数值。用原码表示时,+105和-105的数值部分相同而符号位相反。

原码表示简单易懂,而且与真值的转换方便。但若是两个异号数相加,或两个同号数相减,就要做减法。为了把减运算转换为加运算,从而简化计算机的结构,就引进了反码和补码。

3. 反码

正数的反码与原码一样,符号位为0,其余位为其数值;负数的反码为它的绝对值(即与其绝对值相等的正数)按位(连同符号位)取反。例如:

105 的真值是 +01101001B

105 的反码是 01101001B

-105 的真值是 -01101001B

-105 的反码是 10010110B

4. 补码

正数的补码,其符号位为0,其余位为其数值;负数的补码为它的绝对值(即与该负数的绝对值相等的正数)的补数。把一个数连同符号位按位取反再加1,可以得到该数的补数。例如:

105 的真值是 +01101001B

105 的补码是 01101001B

-105 的真值是 -01101001B

-105 的补码是 10010111B

求补数还可以直接求,方法是从最低位向最高位扫描,保留直至第一个“1”的所有位,以后各位按位取反。负数的补码可以由与其绝对值相等的正数求补得到。根据两数互为补数的原理,对补码表示的负数求补就可以得到该负数的绝对值。例如,对-105的补码10010111B(97H)求补,从右向左扫描,第一位就是1,故只保留该位,对其左面的7位均求反得其绝对值01101001B(69H),即补码数97H的真值是-69H($=-105$)。

一个用补码表示的机器数,若最高位为0,则其余几位即为此数的绝对值;若最高位为1,则其余几位不是此数的绝对值,把该数(连同符号位)求补,才得到它的绝对值。

当数采用补码表示时,就可以将其符号位作为一位数值与其他数值位一样参与数值计算,还把减法运算转换为加法运算,运算的结果也是一补码数。例如,64-10的补码运算如下:

$$64 - 10 = 64 + (-10)$$

$$64 = 40\text{H} = 0100\ 0000\text{B}$$

$$10 = 0\text{AH} = 0000\ 1010\text{B}$$

$$-10 = 1111\ 0110\text{B}$$

做减法运算过程如下:

$$\begin{array}{r} 0100\ 0000 \\ -0000\ 1010 \\ \hline 0011\ 0110 \end{array}$$

用补码相加过程如下：

$$\begin{array}{r} 0100\ 0000 \\ +1111\ 0110 \\ \hline 1\ 0011\ 0110 \\ \downarrow \\ \text{进位自然丢失} \end{array}$$

结果相同，其真值为：54(=36H=48+6)。

最高位的进位是自然丢失的，故做减法与用补码相加的结果是相同的。因此，在微型机中，凡是符号数一律是用补码表示的。一定要记住运算的结果也是用补码表示的。例如：

$$\begin{aligned} 34-68 &= 34+(-68) \\ 34 &= 22\text{H} = 0010\ 0010\text{B} \\ 68 &= 44\text{H} = 0100\ 0100\text{B} \\ -68 &= 1011\ 1100\text{B} \end{aligned}$$

做减法运算过程如下：

$$\begin{array}{r} 0010\ 0010 \\ -0100\ 0100 \\ \hline 1\ 1101\ 1110 \\ \downarrow \\ \text{借位自然丢失} \end{array}$$

用补码相加过程如下：

$$\begin{array}{r} 0010\ 0010 \\ +1011\ 1100 \\ \hline 1101\ 1110 \end{array}$$

结果相同。因为符号位为 1，所以结果为负数。对其求补，得其真值：-00100010B，即为-34(-22H)。

由上面两个例子还可以看出，当数采用补码表示后，两个正数相减，若无借位，化为补码相加就会有进位；若有借位，化作补码相加就不会有进位。

5. 8 位二进制数的范围

8 位二进制数，将其看作无符号数和符号数，它所表示的数的大小是不同的。为加深读者的印象，将其列于表 1-3 中。

表 1-3 8 位二进制数(2 位十六进制数)的大小

8 位二进制数	2 位十六进制数	无符号数	原码数	反码数	补码数
0000 0000	00	0	0	0	0
0000 0001	01	1	1	1	1
0000 0010	02	2	2	2	2
⋮	⋮	⋮	⋮	⋮	⋮
0111 1101	7D	125	125	125	125
0111 1110	7E	126	126	126	126
0111 1111	7F	127	127	127	127
1000 0000	80	128	-0	-127	-128
1000 0001	81	129	-1	-126	-127

续表

8 位二进制数	2 位十六进制数	无符号数	原码数	反码数	补码数
1000 0010	82	130	-2	-125	-126
⋮	⋮	⋮	⋮	⋮	⋮
1111 1101	FD	253	-125	-2	-3
1111 1110	FE	254	-126	-1	-2
1111 1111	FF	255	-127	-0	-1

由表 1-3 可知,8 位二进制无符号数的数值范围为 00H~FFH(0~255)。8 位二进制反码数的数值范围为 80H~7FH(-127~127)。8 位二进制原码数的数值范围为 FFH~7FH(-127~127)。8 位二进制补码数的数值范围为 80H~7FH(-128~127)。原码数 80H 和 00H 的数值部分相同、符号位相反,它们分别为-0 和+0;补码数 80H 的最高位既代表了符号为负又代表了数值为 1,80H 的真值是-128(-80H)。

对于一个 16 位的二进制数,若把它看成无符号数,则其数值范围为 0000H~FFFFH(0~65 535);若把它看成反码数,则其数值范围为 8000H~7FFFH(-32 767~32 767);若把它看成原码数,则其数值范围为 FFFFH~7FFFH(-32 767~32 767);若把它看成补码数,则其数值范围为 8000H~7FFFH(-32 768~32 767)。

上述分析表明,若 8 位二进制补码数运算结果超出-128~127,16 位二进制补码数运算结果超出-32 768~32 767,则产生溢出。小于-128 或小于-32 768 的运算结果称为下溢出,大于 127 或大于 32 767 的运算结果称为上溢出。产生溢出的原因是数据的位数少了,使得结果的数值部分挤占了符号位的位置,为了避免产生溢出,应该将数位扩展。

6. 二进制数的扩展

二进制数的扩展是指一个数据从位数较少扩展到位数较多,如从 8 位(字节)扩展到 16 位(字),或从 16 位扩展到 32 位(双字)。一个二进制数扩展后,其数的符号和大小应保持不变。

无符号数的扩展是将其左边添加 0。例如,8 位无符号二进制数 F8H 扩展为 16 位无符号二进制数,则为 00F8H。

对于用原码表示的二进制数,它的正数和负数仅 1 位符号位相反,数值位都相同。所以,原码二进制数的扩展是将其符号位向左移至最高位,符号位即最高位与原来的数值位间的所有位都填入 0。例如,68 用 8 位二进制数表示的原码为 44H,用 16 位二进制数表示的原码为 0044H;-68 用 8 位二进制数表示的原码为 C4H,用 16 位二进制数表示的原码为 8044H。

补码表示的二进制数的符号位向左扩展若干位后,所得到的补码数的真值不变。所以,对于用补码表示的二进制数,正数的扩展应该在其前面补 0,而负数的扩展,则应该在前面补 1。例如,68 用 8 位二进制数表示的补码为 44H,用 16 位二进制数表示的补码为 0044H;-68 用 8 位二进制数表示的补码为 BCH,用 16 位二进制的补码表示为 FFBCH。

反码表示的二进制数的扩展与补码相同。

1.1.3 二进制数的加减运算

计算机把机器数均当作无符号数进行运算,即符号位也参与运算。运算的结果要根据