



普通高等教育“十一五”国家级规划教材

高等院校信息技术规划教材

微型计算机原理 与接口技术

孙力娟 李爱群 仇玉章 陈燕俐 周宁宁 编著

INFORMATION TECHNOLOGY
INFORMATION TECHNOLOGY
INFORMATION TECHNOLOGY
INFORMATION TECHNOLOGY



清华大学出版社



普通高等教育“十一五”国家级规划教材



高等院校信息技术规划教材

微型计算机原理 与接口技术

孙力娟 李爱群 仇玉章 陈燕俐 周宁宁 编著

清华大学出版社
北京

内 容 简 介

本书以 32 位微处理器为背景,讲述微型计算机原理、汇编语言程序设计和接口技术。内容主要包括: Pentium 微处理器内部结构、x86 基本指令和多媒体指令、汇编语言程序设计、总线概念及微型计算机系统典型总线、存储系统、输入/输出系统、中断系统、串行通信和并行接口、DMA 传送、数模和模数转换、保护模式下的程序设计和 Win32 汇编语言程序设计等。

本书可作为高等院校计算机专业及电类相关专业本科生微型计算机原理及应用、汇编语言程序设计、微型计算机接口技术及微型计算机原理与接口技术等课程的教材和参考书。通过删减适当章节,也适合非电类专业微型计算机原理及应用和微型计算机原理与接口技术等课程的教学,同时也可供自学者及从事计算机应用的工程技术人员参考。

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 13501256678 13801310933

图书在版编目(CIP)数据

微型计算机原理与接口技术 / 孙力娟等编著. —北京: 清华大学出版社, 2007. 2
(高等院校信息技术规划教材)

ISBN 978-7-302-14195-2

I. 微… II. 孙… III. ①微型计算机—理论 ②微型计算机—接口 IV. TP36

中国版本图书馆 CIP 数据核字(2006)第 142411 号

责任编辑: 马瑛珺 刘霞

责任校对: 梁毅

责任印制: 孟凡玉

出版发行: 清华大学出版社

<http://www.tup.com.cn>

c-service@tup.tsinghua.edu.cn

社总机: 010-62770175

投稿咨询: 010-62772015

地址: 北京清华大学学研大厦 A 座

邮编: 100084

邮购热线: 010-62786544

客户服务: 010-62776969

印刷者: 北京密云胶印厂

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: 185×260 印 张: 29.75 字 数: 706 千字

版 次: 2007 年 2 月第 1 版 印 次: 2007 年 2 月第 1 次印刷

印 数: 1~3000

定 价: 38.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话: 010-62770177 转 3103 产品编号: 021201-01

前言

foreword

微型计算机原理与接口技术是理工类学生学习和掌握微型计算机基本组成、工作原理、接口技术以及汇编语言程序设计的重要课程。通过本课程的学习,能够使学生具有微型计算机系统软硬件开发和应用的基本能力。

为了适应微型计算机发展水平,本书以 32 位微处理器 Pentium 作为背景,讲述微型计算机原理、汇编语言程序设计和接口技术。随着 Windows、Linux 等多任务操作系统逐渐成为当前主流操作系统,本书增加了保护模式及 Win32 汇编语言程序设计等方面内容,有一定深度,并具有较强实用性。

全书共有 15 章。

第 1 章为基础理论部分。包括数制、码制等基础知识,计算机组成以及微型计算机的发展历史。

第 2 章以 Pentium 为代表,介绍 32 位微处理器的结构、微处理器各部件的功能、引脚信号定义、工作模式,并对典型总线操作时序进行分析。

第 3 章介绍 80x86 的寻址方式和指令系统。

第 4 章介绍汇编程序的开发过程和伪指令。

第 5 章介绍实模式下汇编语言编程格式,并通过程序实例说明汇编语言程序设计的基本方法,包括分支、循环、子程序调用、代码转换、宏指令以及模块化程序设计等。

第 6 章介绍总线的概念、微型计算机系统中常用的总线标准和 32 位微型计算机总线结构。

第 7 章介绍微型计算机的存储器系统,包括存储器的概念、分类、存储器件以及和微处理器的连接等。

第 8 章介绍微型计算机系统输入/输出及 8254 定时/计数器及其应用。

第 9 章介绍微型计算机中断系统、中断控制器 8259A 和中断程序设计。

第 10 章介绍微型计算机系统串行通信及其应用。

第 11 章介绍并行接口芯片 8255A 及打印机接口。

第 12 章介绍 DMA 传送及 8237A DMA 控制器。

第 13 章介绍数模和模数转换原理以及接口芯片的功能。

第 14 章介绍 32 位微处理器保护模式下的运行机制及其编程方法。

第 15 章介绍 Win32 汇编语言的基础知识、Win32 汇编源程序的格式以及用 Win32 汇编语言编写 Windows 窗口程序的方法。

书中含有大量的程序实例,所有实例都经过上机验证,每章后均有习题。

本书第 1 章、第 2 章和第 6 章由李爱群老师编写;第 3 章、第 4 章、第 5 章、第 9 章和第 13 章、第 8.4 节、10.3 节、10.4 节和 11.2 节由仇玉章老师编写;第 7 章由周宁宁老师编写;第 8 章、第 10 章、第 11 章和第 12 章由孙力娟老师编写;第 14 章、第 15 章和 3.5 节由陈燕俐老师编写,并完成全书的整理。

在本书的编写过程中,洪龙老师、邓玉龙老师和薛明老师提出了许多宝贵建议,在此向他们表示衷心感谢!

由于编者水平有限,书中难免有错漏之处,恳请读者和同行批评指正。

编 者

2005 年 12 月

目录

Contents

第 1 章 计算机基础	1
1.1 计算机中的数制	1
1.1.1 常用计数制	1
1.1.2 数制转换	2
1.2 计算机中数据的编码	3
1.2.1 数值数据的编码与运算	3
1.2.2 字符的编码	6
1.3 浮点数基本概念	7
1.3.1 浮点数	8
1.3.2 浮点机器数	8
1.3.3 浮点数的数值范围	9
1.4 计算机系统的基本组成	10
1.4.1 计算机系统的硬件组成	10
1.4.2 计算机系统的软件组成	11
1.4.3 微型计算机的硬件结构	12
1.4.4 微型计算机的分类和发展	13
习题	15
第 2 章 80x86 微处理器	16
2.1 Intel 微处理器发展简况	16
2.2 32 位微处理器内部结构	17
2.2.1 Pentium 内部结构	18
2.2.2 Pentium 微处理器结构特点	19
2.2.3 32 位微处理器的编程结构	20
2.3 32 位微处理器的外部引脚	27
2.4 32 位微处理器的工作模式	31
2.4.1 80x86 的地址空间	32



2.4.2	实地址模式	32
2.4.3	保护虚拟地址模式介绍	34
2.4.4	虚拟 8086 模式介绍	35
2.5	32 位微处理器的典型时序	36
2.5.1	时钟周期、总线周期和指令周期	36
2.5.2	Pentium 总线周期的时序分析	36
	习题	38
第 3 章	指令系统	39
3.1	概述	39
3.1.1	指令的书写格式	39
3.1.2	符号指令的书写格式	39
3.2	80486 寻址方式	40
3.2.1	立即寻址	40
3.2.2	寄存器寻址	41
3.2.3	存储器操作数的寻址方式	41
3.2.4	80486 寻址方式的段约定和段超越	45
3.3	80486 标志寄存器	46
3.4	80486 基本集指令	49
3.4.1	传送类指令	49
3.4.2	算术运算指令	53
3.4.3	转移和调用指令	62
3.4.4	逻辑运算和移位指令	68
3.4.5	串操作指令	71
3.4.6	处理机控制指令	78
3.5	80x86 多媒体指令	79
3.5.1	MMX 指令	79
3.5.2	SSE 指令	84
	习题	85
第 4 章	宏汇编语言	87
4.1	汇编语言程序的开发过程	87
4.2	汇编源程序的语句类型	88
4.3	宏汇编基本语法	89
4.3.1	标号、变量和常量	89
4.3.2	运算符	90
4.4	数据定义伪指令	92

4.5 宏汇编语言基本语句	95
习题	101
第 5 章 汇编语言程序设计	102
5.1 汇编源程序的编程格式	102
5.1.1 EXE 文件的编程格式	102
5.1.2 COM 文件的编程格式	103
5.1.3 EXE 文件和 COM 文件的内存映像	104
5.1.4 程序段前缀	105
5.1.5 返回 DOS 的其他方法	106
5.1.6 源程序堆栈段的设置	108
5.2 DOS 系统 I/O 功能调用	108
5.3 BIOS 键盘输入功能调用	113
5.4 文本方式 BIOS 屏幕功能调用	114
5.4.1 显示器	114
5.4.2 文本方式 BIOS 屏显功能调用	116
5.5 分支程序	118
5.6 循环程序	121
5.7 子程序及其调用	123
5.8 宏指令与条件汇编	128
5.8.1 宏指令与宏调用	128
5.8.2 条件汇编	131
5.9 代码转换	132
5.10 数值计算和数据处理	140
5.11 字符串的动态显示技术	154
5.12 模块化程序设计	157
5.12.1 支持模块化程序的伪指令	158
5.12.2 模块化程序的设计考虑	158
5.12.3 模块化程序设计举例	159
5.12.4 宏指令共享	164
习题	167
第 6 章 总线	168
6.1 总线基本概念	168
6.1.1 总线的类型与总线结构	168
6.1.2 总线的性能	169
6.1.3 总线信息的传送方式	170



6.2	典型总线标准	171
6.2.1	AT 总线	171
6.2.2	PCI 总线	174
6.3	通用外部总线标准	179
6.3.1	并行 I/O 标准接口 IDE 和 EIDE	180
6.3.2	并行 I/O 标准接口 SCSI	180
6.3.3	通用串行总线 USB	181
6.3.4	视频接口 AGP	187
6.4	32 位微型计算机总线结构	188
	习题	190
第 7 章	存储器系统	191
7.1	概述	191
7.1.1	存储系统概念	191
7.1.2	存储器的体系结构	192
7.1.3	存储器的分类	194
7.1.4	存储器的主要性能指标	195
7.2	随机存储器与只读存储器	197
7.2.1	RAM 的分类与常用 RAM 芯片的工作原理	197
7.2.2	ROM 的分类与常用 ROM 芯片的工作原理	203
7.3	微型计算机系统存储器的组织	206
7.3.1	存储器的扩展技术	206
7.3.2	CPU 与主存储器的连接	210
7.3.3	PC 机的存储器组织	213
	习题	217
第 8 章	输入/输出系统	219
8.1	概述	219
8.1.1	接口电路	219
8.1.2	输入/输出端口	220
8.1.3	输入/输出指令	221
8.2	微型计算机系统与输入/输出设备的信息交换	222
8.2.1	无条件传送方式	222
8.2.2	查询方式	223
8.2.3	中断控制方式	225
8.2.4	直接存储器存取方式	225
8.3	可编程定时器/计数器 8254	226

8.3.1	8254 的内部结构	226
8.3.2	8254 引脚功能	228
8.3.3	8254 的工作方式	229
8.3.4	8254 的控制字与编程方法	234
8.3.5	8254 在微型计算机系统中的应用	237
8.4	发声系统与音乐程序设计	239
8.4.1	PC 系列机发声系统	239
8.4.2	音乐程序设计举例	241
	习题	244
第 9 章	中断系统	245
9.1	中断的基本概念	245
9.2	80x86 中断指令	246
9.3	中断向量	247
9.4	微型计算机系统的中断分类	250
9.4.1	CPU 中断	250
9.4.2	软件中断	251
9.5	8259A 中断控制器	253
9.5.1	8259A 内部结构	253
9.5.2	8259A 中断管理方式	255
9.5.3	8259A 初始化	258
9.6	微型计算机系统可屏蔽中断	263
9.6.1	可屏蔽中断与非屏蔽中断	263
9.6.2	可屏蔽中断的硬件结构	264
9.6.3	硬件中断和软件中断的区别	266
9.7	日时钟中断	267
9.8	实地址模式定时中断程序设计	268
9.8.1	定时中断程序的设计方法	268
9.8.2	定时中断程序设计举例	269
9.9	实时时钟中断	277
9.9.1	实时时钟电路	277
9.9.2	周期中断	278
9.9.3	报警中断	281
9.10	键盘中断	285
9.10.1	键盘中断全过程	285
9.10.2	键代码生成	286
9.11	驻留程序	290
9.11.1	驻留程序的设计方法	290



9.11.2	驻留程序设计举例	293
9.11.3	驻留程序的解驻	296
习题	300
第 10 章	微型计算机系统串行通信	301
10.1	串行通信基础	301
10.1.1	串行通信类型	301
10.1.2	串行数据传输方式	302
10.1.3	串行异步通信协议	303
10.2	可编程串行异步通信接口芯片 8250	305
10.2.1	8250 的内部结构	306
10.2.2	8250 的引脚功能	307
10.2.3	8250 内部寄存器	310
10.2.4	8250 的初始化编程	314
10.3	串行通信程序设计	315
10.3.1	BIOS 通信软件	315
10.3.2	串行通信的外部环境	318
10.3.3	串行通信程序设计	319
10.4	可编程串行通信接口芯片 8251A	323
习题	333
第 11 章	并行 I/O 接口	335
11.1	可编程并行 I/O 接口芯片 8255A	335
11.1.1	8255A 的内部结构及外部引脚	335
11.1.2	8255A 的控制字与初始化编程	338
11.1.3	8255A 的工作方式	339
11.2	8255A 应用	347
11.3	打印机并行接口	356
11.3.1	打印机并行接口标准	356
11.3.2	打印机适配器	357
11.3.3	打印机接口编程	359
习题	364
第 12 章	DMA 控制器	365
12.1	概述	365
12.2	8237A DMA 控制器	366
12.2.1	8237A 的内部结构和引脚功能	366

12.2.2	8237A 内部寄存器	370
12.2.3	8237A 的时序	374
12.3	8237A 的应用	376
12.3.1	8237A 的初始化编程	376
12.3.2	8237A 在 IBM PC/AT 系统中的应用	377
习题	378
第 13 章	数模和模数转换	379
13.1	数模转换	379
13.1.1	数模转换原理	379
13.1.2	DAC 0832 简介	380
13.2	模数转换	382
13.2.1	模数转换原理	382
13.2.2	ADC 0809 简介	383
习题	385
第 14 章	保护模式及其编程	386
14.1	保护模式下的存储管理	386
14.1.1	分段管理	387
14.1.2	分页管理	391
14.1.3	虚拟存储器	393
14.1.4	保护机制	394
14.2	保护模式下的程序调用和转移	396
14.2.1	系统段描述符、门描述符和任务状态段	396
14.2.2	任务内的段间转移	400
14.2.3	任务间的转移	402
14.3	保护模式下的中断和异常	403
14.3.1	中断和异常分类	403
14.3.2	中断和异常类型	404
14.3.3	中断和异常的处理过程	405
14.3.4	中断和异常处理后的返回	406
14.4	保护模式下的输入/输出保护	407
14.5	操作系统类指令	408
14.5.1	实地址模式和任何特权级下可执行的指令	409
14.5.2	实地址模式和在特权级 0 下可执行的指令	409
14.5.3	只能在保护模式下执行的指令	410
14.6	保护模式下的程序设计	411



14.6.1	实地址模式与保护模式切换	411
14.6.2	保护模式下中断和异常程序设计	421
14.6.3	输入/输出保护及任务切换	429
习题	435
第 15 章	Windows 汇编语言编程初步	436
15.1	Windows 基础	436
15.2	Win32 汇编源程序的格式	438
15.2.1	源程序结构	438
15.2.2	Windows API 函数的应用	439
15.3	Win32 汇编可执行文件的生成	442
15.3.1	汇编和链接	443
15.3.2	调试 Win32 汇编程序	445
15.4	Win32 汇编基本语法	445
15.4.1	标号和变量	445
15.4.2	结构	447
15.4.3	子程序	448
15.4.4	高级语法	450
15.5	创建 Windows 下的窗口程序	454
15.5.1	窗口程序的运行过程	454
15.5.2	窗口程序示例	455
习题	461
参考文献	462

计算机基础

1.1 计算机中的数制

数制是数的表示方法。可以用各种进制来表示数,如二进制、十进制、八进制和十六进制等。由于使用电子器件表示两种状态比较容易实现,也便于存储和运算,所以,电子计算机中一般采用二进制数。但人们又习惯于使用十进制,因此在学习和掌握计算机的原理之前,需要了解各种进制的表示法及其相互关系和转换方法。

1.1.1 常用计数制

1. 十进制数

在程序设计中,广泛使用十进制数。十进制数的特点是:每一位有 0~9 这 10 种数码,故基数为 10,高位权是低位权的 10 倍,加减运算的法则为“逢十进一,借一当十”。

2. 二进制数

在电子计算机内部,所有信息都以二进制数形式出现。二进制数的特点是:只有两个不同的数字符号,即 0 和 1,因此基数为 2,高位权是低位权的 2 倍,加减运算的法则为“逢二进一,借一当二”。

3. 十六进制数

十六进制数是把 4 位二进制数作为一组,每一组用等值的十六进制数来表示。十六进制数的特点是:每一位有 0~9 和 A~F 这 16 种数码,因此基数为 16,高位权是低位权的 16 倍,加减运算的法则为“逢十六进一,借一当十六”。

4. 二十进制数

二进制对计算机而言是最方便的,但是人们习惯于用十进制来表示数。为解决这一矛盾可采用二十进制数。二十进制数是计算机中十进制数的表示方法,就是用 4 位二进制数编码表示 1 位十进制数,简称为 BCD 码(Binary Coded Decimal)。

用四位二进制数编码表示一位十进制数,有多种表示方法,计算机中常用的是 8421

BCD 码,它的表示规则,以及与十进制之间的等价关系如表 1-1 所示。

表 1-1 BCD 码与十进制数的转换

二进制数	十进制数	BCD 码	二进制数	十进制数	BCD 码
0000	0	0000	1000	8	1000
0001	1	0001	1001	9	1001
0010	2	0010	1010	10	非法 BCD 码
0011	3	0011	1011	11	非法 BCD 码
0100	4	0100	1100	12	非法 BCD 码
0101	5	0101	1101	13	非法 BCD 码
0110	6	0110	1110	14	非法 BCD 码
0111	7	0111	1111	15	非法 BCD 码

例如: $(3456)_{10} = (0011\ 0100\ 0101\ 0110)_{\text{BCD}}$

BCD 码是十六进制数的一个子集,1010~1111 是非法 BCD 码。

1.1.2 数制转换

1. 二进制、十六进制数→十进制数

二进制、十六进制以至任意进制的数转换为十进制数的方法较简单,根据按权展开式把每个数位上的代码和该数位的权值相乘,再求累加和即可得到等值的十进制数。如:

$$(1101.11)_2 = 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} = (13.75)_{10}$$

$$(E5A)_{16} = 14 \times 16^2 + 5 \times 16^1 + 10 \times 16^0 = (3674)_{10}$$

2. 十进制数→二进制数

十进制数转换为二进制数时,根据该十进制数的类型来决定转换方法。

(1) 十进制整数→二进制数

方法为:“除 2 取余”,即十进制整数被 2 除,取其其余数,商再被 2 除,取其其余数……直到商为 0 时结束运算。然后把每次得到的余数按倒序规律排列,即可得到等值的二进制数。如:

$$N = (14)_{10} = (1110)_2$$

$$\text{运算过程为: } 14 \div 2 = 7 \quad \text{余数} = 0 \cdots \cdots D_0$$

$$7 \div 2 = 3 \quad \text{余数} = 1 \cdots \cdots D_1$$

$$3 \div 2 = 1 \quad \text{余数} = 1 \cdots \cdots D_2$$

$$1 \div 2 = 0 \quad \text{余数} = 1 \cdots \cdots D_3$$

所以 $N = D_3 D_2 D_1 D_0 = (1110)_2$

(2) 十进制纯小数→二进制数

方法为:“乘 2 取整”,即把十进制纯小数乘以 2,取其整数(不参加后继运算),乘积的小数部分再乘以 2,取整……,直到乘积的小数部分为 0。然后把每次乘积的整数部分

按正序规律排列,即可得到等值的二进制数。如:

$$N=(0.8125)_{10}=(0.1101)_2$$

运算过程为:	$0.8125 \times 2 = 1.625$	乘积的整数部分 = $1 \cdots \cdots D_{-1}$
	$0.625 \times 2 = 1.25$	乘积的整数部分 = $1 \cdots \cdots D_{-2}$
	$0.25 \times 2 = 0.5$	乘积的整数部分 = $0 \cdots \cdots D_{-3}$
	$0.5 \times 2 = 1.0$	乘积的整数部分 = $1 \cdots \cdots D_{-4}$

所以 $N=(0.1101)_2$

有些纯小数,不断地“乘 2 取整”也不能使其乘积的小数部分为 0,此时只能进行有限次运算,根据需要取其近似值。

(3) 十进制带小数→二进制数

方法为:整数部分“除 2 取余”,小数部分“乘 2 取整”,然后再进行组合。例如:

$$(14.8125)_{10}=(1110.1101)_2$$

3. 二进制数→十六进制数

以小数点为界,四位二进制数为—组,不足四位用 0 补全,然后每组用等值的十六进制数表示。如:

$$(1101110.11)_2=(0110\ 1110.1100)_2=(6E.C)_{16}$$

在汇编语言中十六进制数用后缀“H”表示。所以:

$(1A2B)_{16}$ 应写成 1A2BH

4. 十六进制数→二进制数

把十六进制数的每一位用等值的二进制数来替换,如:

$$(17E.58)_{16}=(0001\ 0111\ 1110.0101\ 1000)_2=(101111110.01011)_2$$

1.2 计算机中数据的编码

计算机的中心任务就是处理信息。对于计算机系统,不同的信息应有不同的编码方法。计算机处理的信息,主要有数值数据和非数值数据两大类。数值数据是指日常生活中接触到的数字类数据,主要用来表示数量的多少,可以比较大小;非数值数据中最常用的数据是字符型数据,它可以用来表示文字信息,供人们直接阅读和理解;其他的非数值类型数据主要用来表示图画、声音和活动图像等。

1.2.1 数值数据的编码与运算

计算机中使用的数值数据,有无符号数和有符号数两种。在计算机中如何表示一个有符号数呢?最常用的方法是:把二进制数的最高一位定义为符号位,符号位为 0 表示正数,符号位为 1 表示负数,这样就把符号“数值化”了。有符号数的运算,其符号位上的 0 或 1 也被看作数值的一部分参加运算。

通常,把用“+”、“-”表示的数称为真值数,把用符号位上的 0、1 表示正、负的数称为机器数。机器数可以用不同的方法来表示,常用的有原码、反码和补码表示法。

1. 机器数的原码、反码和补码

数 X 的原码记作 $[X]_{\text{原}}$,反码记作 $[X]_{\text{反}}$,补码记作 $[X]_{\text{补}}$ 。

例如,当机器字长 $n=8$ 时:

	符号 ↓		符号位 ↓
设真值数	$X = +5 = +0000101$	原码机器数写成	$[X]_{\text{原}} = 00000101$
	$X = -5 = -0000101$		$[X]_{\text{原}} = 10000101$
	$X = +0 = +0000000$		$[X]_{\text{原}} = 00000000$
	$X = -0 = -0000000$		$[X]_{\text{原}} = 10000000$
设真值数	$X = +5 = +0000101$	反码机器数写成	$[X]_{\text{反}} = 00000101$
	$X = -5 = -0000101$		$[X]_{\text{反}} = 11111010$
	$X = +0 = +0000000$		$[X]_{\text{反}} = 00000000$
	$X = -0 = -0000000$		$[X]_{\text{反}} = 11111111$
设真值数	$X = +5 = +0000101$	补码机器数写成	$[X]_{\text{补}} = 00000101$
	$X = -5 = -0000101$		$[X]_{\text{补}} = 11111011$
	$X = +0 = +0000000$		$[X]_{\text{补}} = 00000000$

由上述例子可以得出以下结论:

- ① 机器数比真值数多一个符号位。
- ② 正数的原、反、补码的数值部分和真值数相同。
- ③ 负数原码的数值部分与真值相同;
负数反码的数值部分为真值数按位取反;
负数补码的数值部分为真值数按位取反末位加 1。
- ④ 没有负零的补码,或者说负零的补码和正零的补码相同。
- ⑤ 由于补码表示的机器数更适合运算,为此计算机系统中负数一律用补码表示。
- ⑥ 机器字长为 n 位的原码数,其真值范围是 $-(2^{n-1}-1) \sim +(2^{n-1}-1)$;
机器字长为 n 位的反码数,其真值范围是 $-(2^{n-1}-1) \sim +(2^{n-1}-1)$;
机器字长为 n 位的补码数,其真值范围是 $-(2^{n-1}) \sim +(2^{n-1}-1)$ 。

2. 整数补码的运算

为理解补码数是怎样进行加减运算的,首先引入几个概念。

(1) 模

模是计量器的最大容量。一个 4 位寄存器能够存放 0000~1111 共计 16 个数,因此它的模为 2^4 。一个 8 位寄存器能够存放 $0 \cdots 0 \sim 1 \cdots 1$, 共计 256 个数,因此它的模为 2^8 , 以此类推, 32 位寄存器的模是 2^{32} 。