

“东北林业大学优秀教材及学术专著
出版与奖励专项资金”资助出版



微型计算机原理与 接口技术

WEIXING JISUANJI YUANLI YU
JIEKOU JISHU

主编 赵 伟 苏建民



东北林业大学出版社



微型计算机原理与接口技术

主 编 赵 伟 苏建民



东北林业大学出版社
• 哈尔滨 •

内容简介

本书主要以当前国内外广泛流行的 16/32 位微处理器为背景,以 INTEL8086/8088 16 位机为基础,追踪 INTEL 主流系列高性能微型计算机的技术发展方向,较全面、系统、深入地讨论微型计算机系统的原理及应用、编程实例及接口技术,具体内容有:计算机中数的表示、数制转换、8086/8088 微型计算机的基本结构、系统总线、8086/8088 指令系统、汇编语言程序设计、DOS 功能调用、存储器的设计、CACHE 的原理、存储器的管理、输入输出的方式、中断技术、8259 中断控制器、8255 并行口、8250 串行口、模/数转换和数/模转换。

“微机计算机原理与接口技术”是非计算机专业的技术基础课,课程内容侧重于微型计算机的应用,教学对象是非计算机专业的学生,教学上应尽可能结合计算机硬件、软件设计的理论特点。

本书以学生掌握微型计算机硬件基础和汇编语言程序设计为目的,了解微处理器及系统、微型计算机存储器、输入输出与中断、可编程接口芯片、典型系统设计和计算机控制系统等实用技术,也是广大从事微型计算机应用与开发的科技人员的自学参考书。

版权专有 侵权必究

举报电话:0451-82113295

图书在版编目 (CIP) 数据

微型计算机原理与接口技术 / 赵伟, 苏建民主编. —
哈尔滨: 东北林业大学出版社, 2016. 3

ISBN 978-7-5674-0722-0

I. ①微… II. ①赵…②苏… III. ①微型计算机—
理论②微型计算机—接口技术 IV. ①TP36

中国版本图书馆 CIP 数据核字 (2016) 第 046561 号

责任编辑: 姚大彬

责任校对: 潘 琦

封面设计: 乔鑫鑫

出版发行: 东北林业大学出版社 (哈尔滨市香坊区哈平六道街 6 号 邮编: 150040)

印 装: 哈尔滨市石桥印务有限公司

开 本: 787mm×1092mm 1/16

印 张: 17

字 数: 420 千字

版 次: 2016 年 3 月第 1 版

印 次: 2016 年 3 月第 1 次印刷

定 价: 30.00 元

《微型计算机原理与接口技术》编委会

主 编	赵 伟	教 授	东北林业大学
	苏建民	教 授	东北林业大学
参 编	周修理	副教授	东北农业大学
	李东滨	副教授	哈尔滨理工大学
	朱振菊	讲 师	东北电力大学
	李 丹	讲 师	大庆师范学院
	朱自民	讲 师	东北林业大学
	刘淑玉	讲 师	东北林业大学

前 言

本书是非计算机专业学生学习“微型计算机原理与接口技术”课程的通用教材，主要以 8086/8088CPU 为例，分析了微处理器的基本结构、指令系统、存储系统及输入输出接口电路；同时，为适应形势的发展，还介绍了部分新型 CPU 技术、主板技术和总线构成等。

本书在每一章的前面介绍本章重点和知识点，内容较为新颖，除了对 8086/8088 重点介绍外，对以后 CPU 技术的发展有一定的介绍，还较为详细地介绍了总线的概念、分类、常用系统总线标准，以及多媒体技术。在存储系统部分，增加了高速缓存（Cache）、虚拟存储的基本概念和存储原理。

本书注重理论联系实际，从应用的角度出发，在讲清基本原理的基础上，强调汇编语言和接口电路分析、设计能力的训练。书中引入了大量的应用实例，读者可以学习如何分析和设计接口电路的基本方法和技巧，如何通过汇编语言对接口进行调控。

本书例题丰富，形式多样，对所举全部实例都有详细的分析和注释。例如在汇编语言程序设计部分，本书通过对每段程序添加详细解释，使读者能够较容易地理解和掌握汇编语言程序设计的思想。

本书循序渐进，易于理解。考虑到本书读者主要为非计算机专业学生，在进入这门课的学习之前并不具备计算机组成和结构方面的知识，是初次涉及计算机的内部结构，所以本书在内容次序的安排上注意由浅入深，突出重点；在文字叙述上力求通俗易懂；同时，在编写中加入了作者多年从事教学、科研的经验和体会。

通过本书的学习并结合上机实践，读者可以对微型计算机系统的组成和工作原理有一个初步了解，具备一定的汇编语言程序设计能力，并能够开发简单外部设备的应用控制系统。因此，本书不仅可以作为课堂教学用书，还能对学生以后的工作有一定指导作用。

本书由东北林业大学赵伟、苏建民主编，参加本书编写的还有东北农业大学周修理、哈尔滨理工大学李东滨、东北林业大学刘淑玉和朱自民、大庆师范学院李丹、东北电力大学朱振菊等。赵伟编写第一章、第二章，苏建民编写第三章、第八章，朱振菊编写第四章，李丹编写第五章，朱自民编写第六章，刘淑玉编写第七章，李东滨、周修理编写全书习题并负责全书的校稿。

由于计算机技术的发展日新月异，新技术层出不穷，加之时间仓促，编者水平有限，书中错误和不当之处在所难免，敬请各位读者和专家批评指正。

编 者

2015 年 10 月

目 录

第一章 计算机基础知识	(1)
第一节 计算机的发展	(1)
第二节 微型计算机中信息的表示	(5)
第三节 二进制数的算术运算和逻辑运算	(13)
第二章 微型计算机基础	(17)
第一节 微型计算机基本结构	(17)
第二节 8086 微处理器	(28)
第三节 8086 的引脚信号和工作模式	(35)
第四节 8086 CPU 的操作时序	(43)
第五节 系统总线	(49)
第六节 80386 微处理器	(60)
第七节 Pentium 微处理器	(64)
第三章 指令系统	(67)
第一节 概述	(67)
第二节 8086 的寻址方式	(68)
第三节 8086/8088 指令系统	(74)
第四章 汇编语言程序设计	(113)
第一节 汇编语言与汇编程序	(113)
第二节 伪指令	(119)
第三节 DOS 功能调用	(127)
第四节 汇编语言程序设计基础	(131)
第五节 常见程序设计举例	(143)
第五章 存储器系统	(152)
第一节 概述	(152)
第二节 随机存取存储器(RAM)	(154)
第三节 只读存储器(ROM)	(170)
第四节 高速缓冲存储器(Cache)	(182)
第六章 输入输出和中断技术	(188)
第一节 输入输出接口	(188)
第二节 简单接口电路	(192)
第三节 输入输出的控制方式	(198)
第四节 中断技术	(202)

第七章 常用数字接口电路	(218)
第一节 接口电路概述	(218)
第二节 可编程定时计数器 8253	(219)
第三节 可编程并行接口 8255	(232)
第四节 可编程串行通信接口 8250	(244)
第八章 A/D 与 D/A 转换器接口	(251)
第一节 D/A 转换器的接口方法	(251)
第二节 D/A 转换器接口电路设计	(253)
第三节 A/D 转换器接口基本原理与方法	(256)
第四节 查询方式的 A/D 转换器接口电路设计	(260)
参考文献	(265)

第一章 计算机基础知识

本章简单讲述了计算机的发展历史、发展趋势及其应用,介绍了计算机和微型计算机的基本概念、特点和性能指标。

在本章中,主要熟悉和掌握的要点如下:

- (1) 了解微处理器和微型计算机的基本概念、发展趋势及其应用;
- (2) 掌握微型计算机的特点和性能指标;
- (3) 理解计算机中的信息表示,包括数值和非数值信息的表示;
- (4) 掌握常用数制及其转换。

第一节 计算机的发展

一、计算机的产生和发展

(一) 计算机的产生

世界上第一台电子数字计算机诞生于 1946 年,取名为 ENIAC (埃尼阿克)。ENIAC 是英文 Electronic Numerical Integrator and Calculator (电子数字积分计算机) 的缩写。这台计算机主要是为解决弹道计算问题而研制的,是由美国宾夕法尼亚大学莫尔电气工程学院的 J. W. Mauchly (莫奇莱) 和 J. P. Eckert (埃克特) 主持研制的。ENIAC 计算机使用了 18 000 多个电子管、10 000 多个电容器、7000 个电阻、1500 多个继电器,耗电 150kW,质量达 30t,占地面积为 170m²。它的加法速度为每秒 5000 次。ENIAC 不能存储程序,只能存储 20 个字长为 10 位的十进制数。ENIAC 计算机的问世,宣告了电子计算机时代的到来。

1944 年 7 月,美籍匈牙利科学家冯·诺依曼博士在莫尔电气工程学院参观了正在组装的 ENIAC 计算机。这台计算机的成功和不足促使他开始构思一个更完整的计算机体系方案。1946 年,他撰写了一份《关于电子计算机逻辑结构初探》的报告。该报告首先提出了“存储程序”的全新概念,奠定了存储程序式计算机的理论基础,确立了现代计算机的基本结构,这一结构被称为冯·诺依曼体系结构。这份报告是人类计算机发展史上的一个里程碑。根据冯·诺依曼提出的改进方案,科学家们研制出人类第一台具有存储程序功能的计算机——EDVAC。EDVAC 计算机由运算器、控制器、存储器、输入设备和输出设备这 5 个部分组成。它使用二进制进行运算操作。指令和数据存储到计算机中,计算机按事先存入的程序自动执行指令。

EDVAC 计算机的问世使冯·诺依曼提出的存储程序思想和结构设计方案成为现实。时至今日,现代电子计算机仍然被称为冯·诺依曼计算机。

(二) 计算机的发展阶段

从 1946 年美国成功研制世界上第一台电子数字计算机至今,按计算机所采用的电子器件来划分,计算机的发展经历了以下 4 个阶段。

第一阶段为 1946~1957 年,计算机采用的电子器件是电子管。电子管计算机的体积十分庞大,成本很高,可靠性低,运算速度慢。第一代计算机的运算速度一般为每秒几千次至几万次。软件方面仅仅初步确定了程序设计的概念,但尚无系统软件可言,软件主要使用机器语言,使用者必须用二进制编码的机器语言来编写程序,其应用领域仅限于军事和科学计算。

第二阶段为 1958~1964 年,计算机采用的电子器件是晶体管。晶体管计算机的体积缩小,质量减轻,成本降低,容量扩大,功能增强,可靠性大大提高。主存储器采用磁芯存储器,外存储器开始使用磁盘,并提供了较多的外部设备。它的运算速度提高到每秒几万次至几十万次。使用者能够使用接近于自然语言的高级程序设计语言方便地编写程序。应用领域也扩大到数据处理、事务管理和工业控制等方面。

第三阶段为 1965~1970 年,计算机采用了小规模集成电路和中规模集成电路。这时计算机的体积大大缩小,成本进一步降低,耗电量更省,可靠性更高,功能更加强大,其运算速度已达到每秒几十万次至几百万次,内存容量大幅度增加。在软件方面,出现了多种高级语言,并开始配有操作系统。操作系统使得计算机的管理和使用更加方便。此时,计算机已广泛用于科学计算、文字处理、自动控制与信息管理等。方面。

第四阶段从 1971 年起到现在,计算机全面采用大规模集成电路 (Large Scale Integrated, LSI) 和超大规模集成电路 (Very Large Scale Integrated, VLSI)。计算机的存储容量、运算速度和功能都有极大的提高,提供的硬件和软件更加丰富和完善。在这个阶段,计算机向巨型和微型两极发展,出现了微型计算机。微型计算机的出现使计算机的应用进入突飞猛进的发展时期。特别是微型计算机与多媒体技术的结合,将计算机的生产和应用推向了新的高潮。

现在,大多数计算机仍然是冯·诺依曼型计算机。人们正试图突破冯·诺依曼的设计思想,这一工作也取得了一些进展,如数据流计算机、智能计算机等,此类计算机统称为非冯·诺依曼型计算机。计算机主要向巨型化、微型化、网络化、智能化方向发展。

(三) 微型计算机的发展

作为第四代计算机的一个重要分支,微型计算机诞生于 20 世纪 70 年代初。微型计算机与其他大、中、小型计算机的区别,主要在于其中央处理器 (Central Processing Unit, CPU) 采用了大规模、超大规模集成电路技术,其他类型计算机的 CPU 则是由相当多的分离元件电路或集成电路所组成。为了将这两种 CPU 相区别,把微型计算机的 CPU 芯片称为微处理器 (Micro Processing Unit, MPU 或 Microprocessor)。

微型计算机具有体积小、功耗低、质量小、价格低、可靠性高以及使用方便等一系列优点,因此获得了广泛的应用和迅速的发展。微型计算机的发展从 1971 年 Intel 公司首先研制成功的 4 位 Intel 4004 微处理器算起,已经有 40 多年的历史,经历了如下几个阶段的演变。

1. 第一阶段

这一阶段（1971~1973 年）为 4 位或低档 8 位微处理器和微型计算机时代，典型产品是 Intel 4004 和 Intel 8008 微处理器以及由它们分别组成的 MCS-4 和 MCS-8 微型计算机。这种计算机的系统结构和指令系统均比较简单，主要用于家用电器和简单的控制场合，其主要技术特点如下：

- (1) 处理器为 4 位或低档 8 位。
- (2) 采用 PMOS 工艺，集成度低。
- (3) 运算功能较差，速度较慢。
- (4) 语言主要以机器语言或简单的汇编语言为主。

2. 第二阶段

这一阶段（1974~1978 年）为中高档 8 位微处理器和微型计算机时代，典型产品是 Intel 公司的 8080/8085，Motorola 公司的 MC 6800 和 Zilog 公司的 Z80 等微处理器以及各种 8 位的单片机，其主要技术特点如下：

- (1) 处理器为中高档 8 位。
- (2) 采用 NMOS 工艺，集成度比第一代提高 4 倍左右。
- (3) 运算速度提高 10~15 倍。
- (4) 采用机器语言、汇编语言或高级语言，后期配有操作系统。

3. 第三阶段

这一阶段（1979~1981 年）为 16 位微处理器和微型计算机时代，典型产品是 Intel 公司的 8086/8088，Motorola 公司的 MC 68000 和 Zilog 公司的 Z8000 等微处理器。其指令系统更加丰富、完善，采用多级中断系统、多种寻址方式、段式存储器结构、硬件乘除部件等，并配有强有力的软件系统，时钟频率为 5~10MHz，平均指令执行时间为 1 μ s，其主要技术特点如下：

- (1) 处理器为 16 位。
- (2) 采用 HMOS 工艺，集成度比第二代提高 10 倍左右。
- (3) 运算速度比第二代提高一个数量级。
- (4) 采用汇编语言、高级语言并配有软件系统。

4. 第四阶段

这一阶段（1981 年以后）为高性能的 16 位和 32 位微处理器和微型计算机时代，典型产品是 Intel 公司的 80386/80486，Motorola 公司的 MC 68030/68040 和 Zilog 公司的 Z80000 等微处理器。它们具有 32 位数据总线和 32 位地址总线，平均指令执行时间为 0.125 μ s，其主要技术特点如下：

- (1) 处理器为高性能的 16 位或 32 位处理器。
- (2) 采用 HMOS 或 CMOS 工艺，集成度在 100 万晶体管/片以上。
- (3) 运算速度再次提高。
- (4) 部分软件硬化。

各阶段微处理器的主要特点可概括如表 1-1 所示。

表 1-1 各阶段微处理器的特点

阶段 主要特点 比较项	第一阶段 1971~1973 年	第二阶段 1974~1978 年	第三阶段 1979~1981 年	第四阶段 1981 年以后
典型的微处理器芯片	Intel 4004 Intel 4040 Intel 8008	Intel 8080 MC 6800 Z80	Intel 8086/8088 MC 68000 Z8000	Intel 80186/80286/ 80386/80486/80586 MC 68030/68040 Z80000
字长/位	4/8	8	16	16/32
芯片集成度/（晶体管/片）	1 000~2 000	5 000~9 000	20 000~70 000	10 万个以上
时钟频率/MHz	0.5~0.8	1~4	5~10	10 以上
数据总线宽度/条	4/8	8	16	16/32
地址总线宽度/条	4~8	16	20~24	24~32
存储器容量	≤16KB 实存	≤64KB 实存	≤1MB 实存	≤4GB 实存 ≤64TB 虚存
基本指令执行时间/μs	10~15	1~2	<1	<0.125
软件水平	机器语言 汇编语言	汇编语言 高级语言 操作系统	汇编语言 高级语言 操作系统	汇编语言 高级语言 部分软件硬化

（四）计算机的发展趋势

未来的计算机将朝巨型化、微型化、网络化与智能化的方向发展。

（1）巨型化是指运算速度更快、存储容量更大、功能更强的超大型计算机。巨型机的运算速度可达每秒百亿次、千亿次甚至更高，其海量存储能力可以轻而易举地存储一个大型图书馆的全部信息。

（2）微型化是指计算机更加小巧、价廉，软件丰富，功能强大。随着超大规模集成电路的进一步发展，个人计算机（PC）将更加微型化。膝上型、书本型、笔记本型、掌上型、手表型等微型化个人电脑将不断涌现，进一步推动计算机的普及和应用。

（3）网络化是指将不同区域、不同种类的计算机连接起来，实现信息共享，使人们更加方便地进行信息交流。现代计算机的网络技术应用已引发了信息产业又一次革命。

（4）智能化是建立在现代科学基础上的、综合性很强的边缘学科。它是指通过让计算机模仿人的感觉、行为、思维过程的复杂机理，使计算机不仅具有计算、加工、处理能力，还能够像人一样“看”“说”“听”“想”和“做”，具有思维与推理、学习与证明的能力。未来的智能型计算机将会代替人类某些方面的脑力劳动。

二、计算机的特点和类型

（一）计算机的特点

计算机能进行高速运算，具有超强的记忆（存储）功能和灵敏准确的判断能力。计算机具有以下基本特点：

（1）运行高度自动化——由于计算机能够存储程序，一旦向计算机发出指令，它就

能自动快速地按指定的步骤完成任务。计算机能够高度自动化运行是它与其他计算工具的本质区别。

(2) 有记忆特性——计算机能把大量数据、程序存入存储器,进行处理和计算,并把结果保存起来。一般计算器只能存放少量数据,而计算机却能存储大量的数据和信息。随着计算机的广泛应用,计算机存储器的存储容量越来越大。

(3) 运算速度快——计算机的运算速度是计算机性能的重要指标之一。通常计算机以每秒完成基本加法指令的数目来表示计算机的运行速度。

(4) 计算精度高——计算机内部采取二进制数字进行运算,可以满足各种计算精度的要求。例如,利用计算机可以计算出精确到小数点后 200 万位的 π 值。

(5) 可靠性高——随着大规模和超大规模集成电路的发展,计算机的可靠性也大大提高,计算机连续无故障的运行时间可以达几个月,甚至几年。

(二) 计算机的类型

由于计算机技术的迅猛发展,计算机已成为一个庞大的家族。从计算机的规模及其用途等不同的角度可做以下分类。

1. 按照计算机的规模进行分类

(1) 巨型计算机——运算速度每秒超过 1 亿次的超大型计算机。巨型计算机可以被许多人同时访问。它对尖端科学、战略武器、气象预报、社会经济现象模拟等新科技领域的研究都具有极为重要的意义。世界上只有少数公司可以生产巨型计算机,如美国的克雷公司生产的 Cray-3,我国自行研制的银河Ⅱ号 10 亿次机和曙光 25 亿次机都是巨型计算机。

(2) 小巨型计算机——性能与巨型计算机接近,但采用了大规模集成电路和微处理器并行处理技术,体积大大减小,费用仅是巨型计算机的 1/10。

(3) 大型主机——运算速度可以达到每秒几千万次浮点运算。大型主机系统强大的功能足以支持远程终端几百个用户同时使用。

(4) 小型计算机——运算速度为每秒几百万次浮点运算。与大型主机一样,小型计算机也支持多用户。

(5) 微型计算机——以大规模集成电路芯片制作的微处理器为 CPU 的个人计算机。

2. 按照计算机的用途进行分类

(1) 通用计算机——具有广泛的用途和使用范围,可以应用于科学计算、数据处理和过程控制等。

(2) 专用计算机——适用于某一特殊的领域,如智能仪表、生产过程控制、军事装备的自动控制等。

第二节 微型计算机中信息的表示

一、计算机中的数制

(一) 进位计数制

日常生活中人们最熟悉的是十进制,但是在计算机中会接触到二进制、十进制和十六

进制。无论是哪种进制，其共同之处都是进位计数制。

所谓进位计数，就是在该进位数制中，可以使用的数字符号个数。 R 进制数的基数为 R ，能用到的数字符号个数为 R 个，即 $0, 1, 2, \dots, R-1$ 。 R 进制数中能使用的最小数字符号是 0 ，最大数字符号是 $R-1$ 。

(二) 二进制、十进制、十六进制

计算机中常用到二进制、十进制和十六进制，它们的基本符号集如表 1-2 所示。

表 1-2 几种进位制数

进 制	计数原则	基本符号
二进制	逢二进一	0, 1
十进制	逢十进一	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
十六进制	逢十六进一	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

注：十六进制的数字符号 A~F 分别对应十进制的 10~15。

二、各计数制的相互转换

(一) 十进制数转换成二进制数

把十进制整数转换成二进制整数的规则是“除 2 取余”，即将十进制数除以 2，得到一个商数和余数；再将其商数除以 2，又得到一个商数和余数；以此类推，直到商数等于 0 为止。每次所得的余数（0 或 1）就是对应二进制数的各位数字。在最后得到二进制数时，将第一次得到的余数作为二进制数的最低位，最后一次得到的余数作为二进制数的最高位。

【例 1-1】将十进制整数 56 转换成二进制数。

2	56		余数为 0	二进制数的最低位
2	28		余数为 0	
2	14		余数为 0	
2	7		余数为 1	
2	3		余数为 1	
2	1		余数为 1	
	0		商数为 0，转换结束	二进制数的最高位

↑
倒
序
取
余

因此，十进制数 56 的二进制数是 111000。

(二) 十进制数转换成十六进制数

将十进制整数转换成十六进制整数的规则是“除 16 取余”。十六进制数计数的原则是“逢十六进一”。在十六进制数中，用 A 表示 10，B 表示 11，C 表示 12，D 表示 13，E 表示 14，F 表示 15。

【例 1-2】将十进制数 89 转换成十六进制数。

将十进制数 89 转换成十六进制数的过程如下：

16	89		余数为 9
16	5		余数为 5
	0		商数为 0，转换结束

十进制数 89 转换成十六进制数是 59。

(三) 将二进制数转换成十进制数与十六进制数

(1) 将二进制数转换成十进制数。

【例 1-3】将二进制数 10111 转换成十进制数。

将二进制数 10111 转换成十进制数的方法如下：

$$(10111)_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 16 + 0 + 4 + 2 + 1 = (23)_{10}$$

(2) 将二进制数转换成十六进制数。

将一个二进制数转换为十六进制数的方法是：将该二进制数从右向左每 4 位分成一组，组间用逗号分隔。每一组代表一个 0~9, A, B, C, D, E, F 之间的数。表 1-3 中列出了二进制数与十六进制数的对应关系。

表 1-3 二进制数与十六进制数的对应关系

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

【例 1-4】将二进制数 111010011 转换成十六进制数。

将二进制数 111010011 转换成十六进制数的方法如下：

0001, 1101, 0011

↓ ↓ ↓

1 D 3

二进制数 111010011 转换成十六进制数是 1D3H。

(四) 十六进制数转换成十进制数

【例 1-5】将十六进制数 1A8FH 转换成十进制数。

将十六进制数 1A8FH 转换成十进制数的方法如下：

$$(1A8F)_{16} = 1 \times 16^3 + 10 \times 16^2 + 8 \times 16^1 + 15 \times 16^0 = 4096 + 2560 + 128 + 15 = (6799)_{10}$$

十六进制数 1A8FH 的十进制数为 6799。

三、计算机中数据的存储单位

计算机中数据和信息常用单位有位、字节和字。

(一) 位 (bit)

位是计算机中最小的数据单位。它是二进制的个数位，简称位。一个二进制位可表示两种状态 (0 或 1)。两个二进制位可表示 4 种状态 (00, 01, 10, 11)。n 个二进制位可表示 2^n 种状态。

(二) 字节 (Byte)

字节是表示存储空间大小最基本的容量单位，也被认为是计算机中最小的信息单位。

8 个二进制位为一个字节。除用字节为单位表示存储容量外，通常还用到 KB（千字节）、MB（兆字节）、GB（千兆字节或吉字节）、TB（千千兆字节）等单位来表示存储器（内存、硬盘等）的存储容量或文件的大小。所谓存储容量指的是存储器中能够包含的字节数。

它们之间存在下列换算关系：

1B = 8bit

1KB = 1024B

1MB = 1024KB

1GB = 1024MB

1TB = 1024GB

（三）字长

字长是计算机存储、传送、处理数据的信息单位。用计算机一次操作（数据存储、传送和运算）的二进制位最大长度来描述，如 8 位、16 位等。字长是计算机性能的重要指标，字长越长，在相同时间内就能传送越多的信息，从而使计算机运算速度越快；字长越长，计算机就有越大的寻址空间，从而使计算机的内存储器容量更大；字长越长，计算机系统支持的指令数量越多，功能就越强。计算机的档次不同，其字长也不同。按字长可以将计算机划分为 8 位机、16 位机（如 286 机，386 机）、32 位机（如 586 机）、64 位机等。计算机的字长是在设计机器时规定的。

四、计算机中数的表示方法

（一）机器数和真值

各种数据在计算机中表示的形式称为机器数，其特点是采用二进制记数制，数的符号用 0、1 表示，小数点则隐含表示而不占位置。机器数对应的实际数值称为数的真值。

机器数有无符号数和带符号数两种，但在机器数中没有符号位。无符号数表示正数，对于无符号数，若约定小数点的位置在机器数的最低位之后，则是纯整数；若约定小数点的位置在机器数的最高位之前（符号位之后），则是纯小数。对于带符号数，机器数的最高位是表示正、负的符号位，其余位则表示数值。

为了便于运算，带符号的机器数可采用原码、反码和补码等不同的编码方法，机器数的这些编码方法称为码制。

在计算机中，无论是数值还是数的符号，都只能用 0、1 来表示。通常专门用数的最高位作为符号位：0 表示正数，1 表示负数。例如（以 8 位为例），+18 在机器中表示为 00010010，-18 在机器中表示为 10010010。这种在计算机中使用的、连同符号位一起数字化的数，称为机器数。

机器数所表示的真实值则叫真值。例如机器数 10110101 所表示的真值为 -53（十进制数）或 -0110101（二进制数）；机器数 00101010 的真值为 +42（十进制数）或 +0101010（二进制数）。

可见，在机器数中，0、1 取代了真值的正、负号。

（二）有符号数的机器数的表示方法

实际上，机器数可以有不同的表示方法。有符号数，机器数常用的表示方法有原码、

反码、补码三种。

(1) 原码。上述机器数表示方法,即最高位表示符号、数值位用二进制绝对值表示的方法,便为原码表示方法。

设机器数位长为 n , 则二进制数 X 的原码可定义为

$$[X]_{\text{原}} = \begin{cases} X = 0X_1X_2\cdots X_{n-1} & X \geq 0 \\ 2^{n-1} + |X| = 1X_1X_2\cdots X_{n-1} & X \leq 0 \end{cases}$$

n 位原码表示数值的范围为

$$-(2^{n-1} - 1) \sim +(2^{n-1} - 1)$$

它对应于原码的 $111\cdots 1 \sim 011\cdots 1$ 。

数 0 的原码有两种不同的表示形式:

$$[+0]_{\text{原}} = 000\cdots 0$$

$$[-0]_{\text{原}} = 100\cdots 0$$

原码表示简单、直观,与真值间转换方便,但用它做加减法运算不方便,而且 0 有 +0 和 -0 两种表示方法。

(2) 反码。正数的反码表示与原码相同;负数的反码是将其对应的正数各位(连同符号位)取反得到,或将其原码除符号位各位取反得到。可见,反码的定义可表示为

$$[X]_{\text{反}} = \begin{cases} 0X_1X_2\cdots X_{n-1} & X \geq 0 \\ 0\bar{X}_1\bar{X}_2\cdots\bar{X}_{n-1} & X \leq 0 \end{cases}$$

或者

$$[X]_{\text{反}} = \begin{cases} X & X \geq 0 \\ 2^{n-1} - |X| & X \leq 0 \end{cases}$$

例如, $[+3]_{\text{反}} = 00000011$ (设为 8 位), $[-3]_{\text{反}} = 11111100$ (设为 8 位)。

n 位反码表示数值的范围为

$$-(2^{n-1} - 1) \sim +(2^{n-1} - 1)$$

它对应于反码的 $100\cdots 0 \sim 011\cdots 1$ 。

数 0 的反码也是两种形式:

$$[+0]_{\text{反}} = 000\cdots 0 (\text{全 } 0)$$

$$[-0]_{\text{反}} = 111\cdots 1 (\text{全 } 1)$$

将反码还原为真值的方法是:反码 $\rightarrow$ 原码 $\rightarrow$ 真值,即 $[X]_{\text{原}} = [[X]_{\text{反}}]_{\text{反}}$ 。或者说,当反码的最高位为 0 时,后面的二进制序列值即为真值,且为正;最高位为 1 时,则为负数,后面的数值位要按位求反才为真值。

(3) 补码。正数的补码表示与原码相同;负数的补码是将对应的正数各位(连同符号位)取反加 1 (最低位加 1) 而得到的,或将其原码除符号位外各位取反加 1 而得到。可见,补码的定义可用表达式表示为

$$[X]_{\text{补}} = \begin{cases} 0X_1X_2\cdots X_{n-1} & X \geq 0 \\ 10\bar{X}_1\bar{X}_2\cdots\bar{X}_{n-1} + 1 & X \leq 0 \end{cases}$$

或者

$$[X]_{\text{补}} = \begin{cases} X & X \geq 0 \\ 2^n + X = 2^n - |X| & X \leq 0 \end{cases}$$

例如, $[+3]_{\text{补}} = 00000011$ (设为 8 位), $[-3]_{\text{补}} = 11111101$ (设为 8 位)。

n 位补码表示数值的范围为

$$-2^{n-1} \sim +(2^{n-1} - 1)$$

它对应于补码的 $100\cdots 0 \sim 011\cdots 1$ 。

数 0 的补码只有一个, 即

$$[+0]_{\text{补}} = [-0]_{\text{补}} = 000\cdots 0 \text{ (全 0)}$$

将补码还原为真值的方法是: 补码 $\rightarrow$ 原码 $\rightarrow$ 真值, 即 $X_{\text{原}} = [[X]_{\text{补}}]_{\text{补}}$ 。或者说, 若补码的符号位为 0, 则其后的数值的值即为真值, 且为正; 若符号位为 1, 则应将其后的数位按位取反加 1, 所得结果才是真值, 且为负。

综上所述, 可以得出以下几点结论:

(1) 原码、反码、补码的最高位都是表示符号位。符号位为 0 时, 表示真值为正数, 其余位为真值。符号位为 1 时, 表示真值为负, 其余位除原码外不再是真值; 对于反码, 需按位取反才是真值; 对于补码, 则需按位取反加 1 才是真值。

(2) 对于正数, 三种编码都是一样的, 即 $[X]_{\text{原}} = [X]_{\text{反}} = [X]_{\text{补}}$; 对于负数, 三种编码互不相同。所以, 原码、反码、补码本质上是用来表示负数在机器中的三种不同的编码方法的。

(三) 数的定点和浮点运算

当所要处理的数含有小数部分时, 就有一个如何表示小数点的问题。在计算机中并不用某个二进制位来表示小数点, 而是隐含规定小数点的位置。

根据小数点位置是否固定, 数的表示方法可分为定点表示和浮点表示, 相应的机器数就叫定点数和浮点数。

所谓定点数就是小数点的位置固定不变的数。小数点的位置通常有两种约定方式: 定点整数 (纯整数, 小数点在最低有效数值位之后) 和定点小数 (纯小数, 小数点在最高有效数值位之前)。

通常, 对于任意一个二进制数 X , 都可表示成

$$X = 2^J \times S$$

其中, S 为数 X 的尾数; J 是数 X 的阶码; 2 是阶码的底。尾数 S 表示数 X 的全部有效数字, 阶码 J 则指出小数点的位置。 S 值和 J 值都可正可负。当 J 值可变时, 表示是浮点数。

(1) 定点数。在计算机中, 根据小数点固定的位置不同, 定点数有定点 (纯) 整数和定点 (纯) 小数两种。当阶码 $J=0$, 若尾数 S 为纯整数时, 说明小数点固定在数的最低位之后, 即称为定点整数; 当阶码 $J=0$, 若尾数 S 为纯小数时, 说明小数点固定在数的最高位之前, 即称为定点小数。

定点整数和定点小数在计算机中的表示形式没什么区别, 其小数点完全靠事先约定而隐含在不同位置, 如图 1-1 所示。

(2) 浮点数。当要处理的数是既有整数又有小数的混合小数时, 采用定点数格式很不方便。为此, 人们一般都采用浮点数进行运算。

浮点数一般由 4 个字段组成, 其一般格式如图 1-2 所示。其中, 阶码一般用补码定