全国计算机软件专业技术资格考试

电视函授教学丛书

程序员级考试辅导
硬软件基础知识

何守才　主编

邵祖英　主审

机械电子工业部计算机技术培训中心

出 版 说 明

由国家人事部、国家科委、机械电子部、国务院电子信息系统推广应用办公室等部委组织的“中国计算机软件人员专业技术资格考试”从一九九〇年开始在全国统一举行。这是深化职务改革的一项重要措施，它将大大促进计算机应用人才的培养，促进人才和软件的国际交流，促进我国软件事业的发展，推动电子信息技术的广泛应用。

为配合这一考试，帮助广大软件技术人员做好应试准备和提高实际应用水平，经国家教委和机电部批准，在国务院电子信息系统推广应用办公室的领导和支持下，在中国计算机应用软件人员考试中心的指导和支持下，机电部计算机技术培训中心组织了北京、上海、南京、杭州和合肥等地区二十多个高等院校、研究所、企事业单位的三十多位专家教授，组成了编审委员会，按照考试大纲，在分析了历届考试试题的基础上，编写了这套全国电视函授教学丛书。这套丛书将于一九九一年初先出版以下六本：

程序员级考试辅导——硬软件基础知识

程序员级试题分析与解答（'87—'90）

高级程序员级考试辅导——计算机硬件系统

高级程序员级考试辅导——软件知识

高级程序员级考试辅导——软件设计方法和工具

高级程序员级试题分析与解答（'87—'90）

为了在中国教育电视台上通过卫星向全国播放，机电部计算机技术培训中心组织了京沪地区十七名专家教授，由培训中心电教室拍摄了与上述教材配套的教学录像带（约120学时）。我们期望，这套教材和配套录像带能对考生在复习应考时有所指导和帮助，也能受到广大在职计算机软件人员和大专院校师生的欢迎。

机电部计算机技术培训中心是部属的教育事业单位。该中心依托的中国计算机软件与技术服务总公司在全国组织的培训网，在三十多个省市建有分中心。该培训网的年培训量占全国同类培训量的50%以上，是我国唯一的全国性跨行业的计算机技术实体单位。我们衷心感谢中央和各地电子厅局、电振办、教委和仪表厅局的支持，衷心感谢上海市计算机应用软件培训中心以及各有关高等院校、研究所、企事业单位的专家教授们在编写教材、拍摄录像带和开展全国电视函授工作中给予的支持和帮助。

为慎重起见，本套教材将先作内部发行，待以后广泛征求意见并作必要的修改补充后再正式发行。欢迎各界同行提出宝贵意见，来函地址：北京学院南路55号电脑大厦机电部计算机技术培训中心，邮政编码：100081。

机械电子工业部计算机技术培训中心

一九九一春

前　　言

本书是全国计算机软件人员专业技术资格考试电视函授教学丛书之一，是程序员的基本教材，专门讲解计算机硬软件基础知识。内容包括：数制及机内代码、逻辑部件、CPU与指令系统、存贮器及I/O设备、基本数据结构、基本算法、文件系统、操作系统、数据库以及程序流程图等。每章末附有历届试题供读者练习，这些试题的解答可参阅本丛书中的另一本书："程序员级考试试题分析与解答（'87—'90）。我们希望读者将这两本书结合起来阅读。

本书由机电部计算机技术培训中心组织，由上海计算机应用软件培训中心理事长、上海第二工业大学何守才副教授主编，由机电部计算机技术培训中心主任邵祖英教授主审

本书各部分的作者如下：

第一章由上海电子信息办劳诚信高级工程师编写，第二章由上海交通大学白英彩教授编写，第三、四章由上海交通大学孙德文副教授编写，第五章由上海计算机厂杨瑞荪高级工程师编写，第六章由上海第二工业大学何守才副教授编写，第七章由华东师范大学徐国定副教授编写，第八章由何守才、高伟红编写，第九章由何守才编写，第十章由华东师范大学徐国定副教授和黄馥林副教授编写。参加各章节编写的还有同济大学杨振山教授、复旦大字胡运法副教授、合肥工业大学杨作慎教授和上海第二工业大学宣泰章副教授。

本书是根据程序员级考试大纲，同时参考国内外的有关资料，并在分析历届试题的基础上编写的。在本书的编写过程中，本丛书的编审委员进行了指导和审校。特别是主编委上海复旦大学计算机系主任施伯乐教授对本书进行了审阅。参加本书审校的还有沈林兴、邓小敏、李大方、高伟红、田开亮、吕跃军、刘捷等。

由于本书编写时间仓促，参加编写的人员较多，编写风格不一，凡有不足或错误不处，敬请读者与有关专家指正。

编　者

1991年1月

内 容 简 介

本书是全国计算机软件人员专业技术资格考试电视函授教学丛书之一，是程序员级考试辅导的基本教材，专门讲解硬软件基础知识。内容包括：数制及机内代码、逻辑部件、CPU与指令系统、存贮器及I/O设备、基本数据结构、基本算法、文件系统、操作系统、数据库、程序流程图等。每章末附有历届试题作为习题。本书可作为计算机应用软件人员程序员级培训、辅导及自学教材，也可供大、中专院校师生和在职软件人员学习参考。

目 录

前言

第一篇 硬件基础知识

第一章 计算机数制及机内代码 …… (1)
1.1 计算机概述 …… (1)
1.1.1 电子计算机的分类 …… (1)
1.1.2 计算机的特点 …… (1)
1.2 数字化编码的概念 …… (2)
1.3 进位计数制 …… (2)
1.3.1 二进制数 …… (2)
1.3.2 八进制数 …… (2)
1.3.3 十六进制数 …… (3)
1.3.4 任意进制数 …… (3)
1.4 进位数制之间的转换 …… (4)
1.4.1 二进制转换成十进制 …… (4)
1.4.2 十进制整数转换成二进制整数——除2取余法 …… (4)
1.4.3 十进制小数转换成二进制小数——乘2取整法 …… (4)
1.5 机内数的表示法 …… (5)
1.5.1 原码表示法 …… (5)
1.5.2 补码表示法 …… (6)
1.5.3 反码表示法 …… (7)
1.5.4 移码表示法 …… (8)
1.5.5 定点数与浮点数 …… (8)
1.6 常用的十进制数的二进制编码 …… (9)
1.6.1 “8421”码 …… (10)
1.6.2 余3码 …… (10)
1.6.3 格雷码 …… (11)
1.7 字符编码 …… (11)
1.7.1 ASCII码 …… (11)
1.7.2 EBCDIC码及其它字节编码 …… (11)
1.8 常用的代码校验方法 …… (12)
1.8.1 校验编码的基本概念 …… (12)
1.8.2 奇偶校验码 …… (13)
1.8.3 纵横奇偶冗余校验 …… (14)
1.8.4 海明码 …… (14)

1.8.5 循环码……………………………………………………………………………………（ 15 ）
1.8.6 n中取m码 …………………………………………………………………………（ 16 ）
1.9 汉字编码…………………………………………………………………………………（ 17 ）
1.10 历届试题 ……………………………………………………………………………（ 17 ）
第二章 逻辑部件、CPU与指令系统 ……………………………………………………（ 21 ）
2.1 算数运算与逻辑运算…………………………………………………………………（ 21 ）
2.1.1 二进制数的计算机运算方法………………………………………………………（ 21 ）
2.1.2 十进制数的计算机运算方法………………………………………………………（ 28 ）
2.1.3 逻辑代数的基本运算……………………………………………………………（ 30 ）
2.1.4 逻辑式的化简和变换……………………………………………………………（ 31 ）
2.1.5 基本逻辑部件……………………………………………………………………（ 37 ）
2.2 计算机的主要部件……………………………………………………………………（ 42 ）
2.2.1 中央处理器CPU …………………………………………………………………（ 43 ）
2.2.2 存贮器（Memory）………………………………………………………………（ 45 ）
2.2.3 输入/输出设备 …………………………………………………………………（ 45 ）
2.2.4 总线（Bus）………………………………………………………………………（ 46 ）
2.3 指令系统………………………………………………………………………………（ 47 ）
2.3.1 指令的格式………………………………………………………………………（ 48 ）
2.3.2 常用的寻址方式…………………………………………………………………（ 49 ）
2.3.3 指令的分类与功能………………………………………………………………（ 52 ）
2.3.4 指令的执行过程…………………………………………………………………（ 56 ）
2.4 历届试验………………………………………………………………………………（ 59 ）
第三章 存贮器 ……………………………………………………………………………（ 64 ）
3.1 计算机的存贮系统……………………………………………………………………（ 64 ）
3.2 主存贮器………………………………………………………………………………（ 68 ）
第四章 外部设备 …………………………………………………………………………（ 72 ）
4.1 概述……………………………………………………………………………………（ 72 ）
4.2 常用输入/输出设备 …………………………………………………………………（ 73 ）
4.3 辅助存贮器……………………………………………………………………………（ 78 ）
4.4 历届试验（三、四章）………………………………………………………………（ 86 ）

第二篇 软件基础知识

第五章 基本数据结构和基本算法 ………………………………………………………（ 93 ）
5.1 基本数据结构…………………………………………………………………………（ 93 ）
5.1.1 基本概念…………………………………………………………………………（ 93 ）
5.1.2 线性表……………………………………………………………………………（ 95 ）
5.1.3 数组和串…………………………………………………………………………（ 97 ）
5.1.4 栈和队列…………………………………………………………………………（ 97 ）
5.1.5 树…………………………………………………………………………………（ 98 ）

5.2 基本算法………………………………………………………………（101）
5.2.1 查找（Search）……………………………………………………（101）
5.2.2 排序（Sort）………………………………………………………（101）
5.2.3 合并……………………………………………………………………（105）
5.2.4 字符串操作…………………………………………………………（109）
5.3 历届试验………………………………………………………………（110）
第六章 文件系统与操作系统………………………………………………（115）
6.1 文件系统………………………………………………………………（115）
6.1.1 文件系统的结构……………………………………………………（115）
6.1.2 文件类别与功能……………………………………………………（117）
6.1.3 文件的使用…………………………………………………………（120）
6.2 操作系统………………………………………………………………（122）
6.2.1 操作系统的分类及功能……………………………………………（122）
6.2.2 联机命令语言与作业控制语言的使用……………………………（125）
6.2.3 操作系统的四大管理………………………………………………（126）
6.2.4 操作系统的使用……………………………………………………（136）
6.2.5 常用的几种操作系统………………………………………………（136）
6.3 历届试题………………………………………………………………（138）
第七章 程序语言与语言处理程序的基础知识……………………………（145）
7.1 程序语言基础知识……………………………………………………（145）
7.1.1 语言所提供的数据类型和控制结构………………………………（145）
7.1.2 常用语言的知识……………………………………………………（158）
7.2 语言处理程序的基础知识……………………………………………（161）
7.2.1 汇编程序概述………………………………………………………（162）
7.2.2 编译程序概述………………………………………………………（163）
7.2.3 解释程序概述………………………………………………………（165）
第八章 数据库系统…………………………………………………………（167）
8.1 数据库系统的基本概念………………………………………………（167）
8.1.1 信息系统的三个世界………………………………………………（167）
8.1.2 数据库（DB）、数据库系统（DBS）数据库管理系统（DBMS）………（168）
8.1.3 逻辑数据库（LDB）物理数据库（PDB）……………………………（169）
8.1.4 数据库技术特征……………………………………………………（169）
8.2 数据模型………………………………………………………………（170）
8.2.1 层次模型（Hirearchical Model）…………………………………（170）
8.2.2 网络模型（Nefwork Model）………………………………………（172）
8.2.3 关系模型（Relational Model）……………………………………（172）
8.2.4 实体——关系模型（Entity-Kelationship Model）………………（173）
8.2.5 数据模型的相互转换………………………………………………（174）
8.3 数据库语言……………………………………………………………（175）
8.3.1 主语言与数据语言…………………………………………………（175）
8.3.2 数据描述和定义数据库中各种对象及其数据特征………………（175）

8.3.3 数据操作语言（DML）……（177）
8.4 SQL结构查询语言……（177）
8.5 常用的数据库管理系统……（182）
8.5.1 层次型数据库系统IMS……（182）
8.5.2 网状模型数据库管理系统DBTG……（183）
8.5.3 关系型数据库系统RDBS……（186）
8.6 历届试题……（191）

第三篇 程序编制

第九章 程序流程图……（193）
9.1 流程图的基本概念……（193）
9.1.1 程序流程图……（193）
9.1.2 流程符号与流程图的国家标准……（193）
9.2 程序流程图的基本概念……（197）
9.2.1 程序流程图的基本符号……（197）
9.2.2 程序流程图的基本形式……（198）
9.2.2 算法与程序流程图……（198）
9.3 三种基本形式的应用……（199）
9.3.1 顺序型的应用……（199）
9.3.2 选择型的应用……（200）
9.3.3 循环型的应用……（201）
9.4 表查找操作的程序流程图……（203）
9.4.1 顺序查找……（204）
9.4.2 二分查找法……（205）
9.4.3 区间查找……（206）
9.5 科学计算中的常用程序流程图……（207）
9.5.1 初等计算……（207）
9.5.2 方程式的数值求解法……（210）
9.5.3 数值积分……（212）
9.5.4 矩阵运算……（215）
9.6 其它几种设计工具的简单介绍……（217）
9.6.1 HIPO图……（217）
9.6.2 盒图（N-S图）……（217）
9.6.3 判定表……（219）
9.7 例题选解……（220）
第十章 汇编语言CASL……（224）
10.1 机器语言、汇编语言……（224）
10.2 COMET计算机的结构……（224）
10.3 汇编语言CASL中语句的有关规定……（225）

10.4 汇编格式指令 ……………………………………………… ([illegible])
10.5 伪指令 ……………………………………………………… ([illegible])
10.6 宏指令 ……………………………………………………… ([illegible])
10.7 程序设计实例 ……………………………………………… (234)
10.8 历届试题 …………………………………………………… (239)
附录 水平考试综合试题 ……………………………………… (245)

第一章　计算机数制及机内代码

1.1 计 算 机 概 述

1.1.1 电子计算机的分类

目前，计算机分为两大类型：电子模拟计算机和电子数字计算机。

电子模拟计算机所处理的电信号在时间上是连续的，称为模拟量。用电信号的幅值模拟数值或某一物理量的大小表示不同的数值。

电子数字计算机所处理的电信号在时间上是离散的，称为数字量。用输出信号电平的高低或脉冲的有无表示数值的0或1。这样就可以用一串彼此在时间上是离散的脉冲，或用一组触发器的输出电平的高低来表示一个数值，其不同的组合就能够表示大小不同的数值，增加组合的位数，就能增加数的表示范围和精度。采用数字化信息能够用各种存储器和寄存器加以保存。数字信息可以用来表示各种物理量和逻辑变量。所以数字计算机除了进行数字计算外，还能进行逻辑加工。到目前为止，数字计算机采用二进行制数字系统作为运算的基础，用数字逻辑电路实现各种功能，用逻辑代数为工具去设计逻辑电路。

数字计算机的种类也不少，一般按其用途来划分，可以分为通用计算机和专用计算机。前者主要用在科学计算、数据管理、信息管理等方面。后者则用在工业控制、军事、国防事业等专用设备上。

按其规模来分，则可分为巨型机、大型机、中型机、超级小型机、超级微型机及微型机等。

当然，在划分种类的时候划分的标准将随着时代的发展而改变。因为十年前的中型机，现在可能已不及一台小型机甚至只能相当于一台超级微型机的功能。因此，在计算机突飞猛进的时代里，实际上是没有一个绝对的一成不变的标准来衡量或划分计算机的种类、规模乃至其它性能的。

从1946年第一台电子计算机问世以来，其发展极为迅速，计算机性能和应用都有新的飞跃。从总的方面来说，计算机的发展方向是：巨型、微型、网络型和智能型。

1.1.2 计算机的特点

电子数字计算机（以下简称为计算机）有以下一些特点：

1.速度高。现代计算机速度，一般都能达到每秒完成几万次，几十万次的运算。更高的可达几千万次以至上亿次的运算。

2.精度高。它易于达到很高的计算精度，它的计算精度只决定于数字的位数而不决定于元件的精度。这是其它运算工具无法比拟的。

3.具有逻辑判断能力，从而能够完成很多逻辑性质的工作，模拟人类逻辑思维，大大扩充了计算机的应用范围。

4.自动化程度高。具有记忆装置，可以存贮问题求解步骤、计算的原始数据、中间结果和最后答案，因此计算机处理工作可自动完成。

5.能存贮和快速读取大量数据，为各种数据信息的高速处理创造了条件。

6.通用性强。计算机除了用于各种数值计算、解决各种工程设计中的数学问题外，还可以用于信息处理、语言翻译、实时控制等。在各行各业社会各个领域得到广泛应用。

1.2 数字化编码的概念

计算机的最基本功能是进行数的计算和处理加工。任何信息进入计算机和被计算机加工，必须采取数字化编码的形式，这不仅是指参加数值计算的操作数，而且包括各种非数值信息，如文字、符号、语言、图象等。

所谓编码，就是用少量、简单、基本的符号根据一定的规则相组合以表示大量复杂多样的信息。一切信息编码都有两个要素：第一是基本符号的种类；第二是基本符号相互组合的规则。在计算机中，要利用物理元件来实现信息编码，必须使基本符号尽可能少，编码规则尽可能简单唯一。在目前计算机中只使用两种基本符号，即“0”和“1”，它可以利用任何物理量的两个相对稳定状态来表示。既可有数值的含义，也有逻辑的含义（开和关、是和非等），又是对各种基本编码符号进行重编码的最基本要素。

1.3 进位计数制

1.3.1 二进制数

二进制数构成的基本符号及组合的规则是：

（1）它的数值部分，只可能取两个基本符号0或1。

（2）它是“逢二进一”的。不同的数码在不同的数位所代表的值也是不同的。

例 $(1011)_2=1\times2^3+0\times2^2+1\times2^1+1\times2^0=(8+2+1)_{10}=(11)_{10}$

$(1101.11)_2=1\times2^3+1\times2^2+1\times2^0+1\times2^{-1}+1\times2^{-2}$

$=(8+4+1+0.5+0.25)_{10}=(13.75)_{10}$

一般地，任意一个二进制数B，可以表示为：

$$B=b_n\cdot2^n+b_{n-1}\cdot2^{n-1}+\cdots+b_1\cdot2^1+b_0\cdot2^0$$

$$+b_{-1}\cdot2^{-1}+b_{-2}\cdot2^{-2}+\cdots+b_{-m}\cdot2^{-m}=\sum_{i=-m}^{n}b_i\cdot2^i$$

其中，b_i只能取值为1或0，由具体的数B确定。n、m为正整数，小数点左面有n+1位整数，小数点右面有m位小数。2是进位制的基数，所以称为二进制。

1.3.2 八进制数

类似地，八进制数构成的基本符号与组合的规则是：

（1）组成八进制数的基本符号是0、1、2、3、4、5、6、7和8。

（2）八进制数的组合规则是“逢八进一”。各种不同基本符号在不同位置所表示的值是不同的。

例 $(123)_8=1\times8^2+2\times8^1+3\times8^0=(64+16+3)_{10}=(83)_{10}$

$$(123.56)_8=1\times8^2+2\times8^1+3\times8^0+5\times8^{-1}+6\times8^{-2}$$
$$=(64+16+3+0.625+0.09375)_{10}$$
$$=(83.71875)_{10}$$

1.3.3 十六进制数

十六进制数构成的基本符号与组合的规则是：

（1）构成十六进制数的基本符号有十六个，即0、1、2、3、4、5、6、7、8、9、A、B、C、D、E和F。

（2）十六进制数的组合规则是"逢十六进一"。每个基本符号在不同数位，所表示的数值是不同的。

例 $(3AB\cdot12)_{16}=3\times16^2+A\times16^1+B\times16^0+1\times16^{-1}+2\times16^{-2}$

$$=(768+160+11+0.0625+0.0078125)_{10}$$
$$=(939.0703125)_{10}$$

1.3.4 任意进制数

综上几种计数制，不难看出，对于任意数制R的进位数制，应满足：

（1）具有R个不同基本符号，例如0、1、2、…R－1；

（2）数制的编码符合"逢R进位"规则，它的每一个数位对应一个基本符号值D_i，而R_i的单位赋予值R^i，则R^i称为R_i位的权。小数点左面各位的权依次是基数R的正次幂；而小数点右面各位的权依次是基数R的负次幂。

因此，一般地，R进制数N，可表示为：

$$N=D_n\cdot R^n+D_{n-1}R^{n-1}+\cdots+D_1R^1+D_0R^0+D^{-1}R^{-1}+\cdots+D_{-m}R^{-m}$$
$$=\sum_{i=-m}^{n}D_iR^i$$

其中，D_i可取如0、1、2…R－1中任一符号，它的值取决于数N；n、m为正整数；小数点左面有n＋1位正整数，小数点右面有m位小数；R为基数。所以称为R进制数。

2.2.6二—十—八—十六进制数码对照关系

二进制、十进制、八进制和十六进制数码的关系如表1.1所示。

表 1.1 二进制、十进制、八进制和十六进制数码表

二进制数	十进制数	八进制数	十六进制数	二进制数	十进制数	八进制数	十六进制数
0000	0	0	0	1000	8	10	8
0001	1	1	1	1001	9	11	9
0010	2	2	2	1010	10	12	A
0011	3	3	3	1011	11	13	B
0100	4	4	4	1100	12	14	C
0101	5	5	5	1101	13	15	D
0110	6	6	6	1110	14	16	E
0111	7	7	7	1111	15	17	F

1.4 进位数制之间的转换

二进制数适用于计算机，但人们习惯使用十进制数，所以，用计算机进行运算，则要做数制转换。下面讨论各种数制间的换算原理。

1.4.1 二进制转换成十进制

二进制数转换成十进制数，是根据二进制数的定义，将它按权展开相加。

例 $(10011.101)_2=1\times2^4+0\times2^3+0\times2^2+1\times2^1+1\times2^0+1\times2^{-1}+0\times2^{-2}+1\times2^{-3}$

$=(16+2+1+0.5+0.125)_{10}$

$=(19.625)_{10}$

1.4.2 十进制整数转换成二进制整数——除2取余法

例 把十进制数37转换成二进制数

$$0 \leftarrow 1 \leftarrow 2 \leftarrow 4 \leftarrow 9 \leftarrow 18 \xleftarrow[\div 2]{\text{商}} 37$$

$$\begin{array}{cccccc|l} | & | & | & | & | & | & \text{余数} \\ 1 & 0 & 0 & 1 & 0 & 1 & \\ | & | & | & | & | & | & \\ K_5 & K_4 & K_3 & K_2 & K_1 & K_0 & \end{array}$$

所以，$(37)_{10}=(100101)_2$

1.4.3 十进制小数转换成二进制小数——乘2取整法

例 把十进制小数0.8125转换成二进制小数。

$$0.8125 \xrightarrow[\times 2]{\text{小数部分}} 0.625 \xrightarrow[\times 2]{} 0.25 \xrightarrow[\times 2]{} 0.5 \xrightarrow[\times 2]{} 0$$

$$\begin{array}{lcccc} & | & | & | & | \\ \text{整数部分} & 1 & 1 & 0 & 1 \\ & | & | & | & | \\ & K_{-1} & K_{-2} & K_{-3} & K_{-4} \end{array}$$

所以，十进制小数0.8125的二进制小数为0.1101。

值得注意的是，在十进制小数转换成二进制小数时，不断用2乘不一定都能使尾数部分等于0，过程可能会无限制的进行下去。这时，可根据精度要求取m位，得到十进制小数的二进制小数的近似表达式。

综上所述，对于包括整数和小数的十进制数转换成二进制数时，由于转换法则不同，则要把它分成整数和小数两部分，然后，再把它们分别转换为二进制表达式，最后用小数点把这两部分连起来就可以了。

例如，把十进制数37.8125换算为二进制数。

整数 $(37)_{10}=(100101)_2$

小数 $(0.8125)_{10}=(0.1101)_2$

所以 $(37.8125)_{10}=(100101.1101)_2$

1.5机内数的表示法

二进制在计算机中的表示形式与人们的习惯不尽相同，这主要是为了“方便”机器的操作，减少硬件的开销，以及提高运算速度等诸方面原因而设计的。

一般地说，经常用“0”表示正号，用“1”表示负号，并且通常将符号位放在一个数码的最高位。常用的表示方法是：原码、补码、反码和移码等。

1.5.1 原码表示法

1.原码表示法的定义

假设x是小数，且小数点的左面一位是符号位，即

$$x=x_n\cdot x_{n-1}\cdots x_1$$

则x的原码定义为

$$[x]_{原}=\begin{cases}x & 0\leqslant x<1\\ 1-x & -1\leqslant x<0\end{cases}$$

例如，若$x=+0.0110011$，则$[x]_{原}=0.0110011$

若$x=-0.0110011$，则$[x]_{原}=1.0110011$

假设x是整数，即$x=x_nx_{n-1}\cdots x_1$。（即小数点在最后面）

则x的原码定义为

$$[x]_{原}=\begin{cases}x & 0\leqslant x<2^{n-1}\\ 2^{n-1}-x & -2^{n-1}<x\leqslant 0\end{cases}$$

例如，若$x=+01100110$，则$[x]_{原}=01100110$

若$x=-01100110$，则$[x]_{原}=11100110$

2.原码表示法的数值范围

不妨以8位二进位来介绍，设

$$x=x_7x_6\cdots\cdots x_0$$

为8位二进制小数，且x_7为符号位。则其最大表示数为$0.1111111=\left(+\frac{127}{128}\right)_{10}$，其最小表示数为$1.1111111=-\left(\frac{127}{128}\right)_{10}$

设$x=x_7x_6\cdots x_0$

为8位二进制整数，且x_n为符号位，则其最大表示数为$01111111=(+127)_{10}$，其最小表示数为$11111111(-127)_{10}$由定义可知原码中“零”的表示形式有二种，即。

$$[x]_{原}=1.00\cdots0$$

$$[x]_{原}=0.00\cdots0$$

因此它实际上只能表示出255个不同的数值。原码一般适用于机器的乘除法，而不适用于加减法。

1.5.2 补码表示法

1.补码表示法的定义

假设x是小数，且小数点的左面是符号位，即

$$x=x_{n-1}x_{n-2}\cdots x_0$$

则其补码的定义为

$$[x]_{补}=\begin{cases}x & 0\leqslant x<1\\ 2+x & -1\leqslant x<0\end{cases}$$

例如，若$x=+0.1100101$，则$[x]_{补}=0.1100101$

若$x=-0.1100101$，

则$[x]_{补}=2+x=2-0.1100101$

$=1.0011011$

假设x是整数，即

$x=x_nx_{n-1}\cdots x_1$。（小数点在最右面，x_n为符号位）

则x的补码定义为

$$[x]_{补}=\begin{cases}x & 0\leqslant x<2^{n-1}\\ 2^n+x & -2^{n-1}\leqslant x<0\end{cases}$$

例如，若$x=+0110101$则$[x]_{补}=00110101$

若$x=-0110101$则$[x]_{补}=11001011$

2 补码表示法的数值范围

设$x_7x_6\cdots x_0$是8位二进数码，x_7为符号位，则补码能表示的最大数为

$$0.1111111=+\left(\frac{127}{128}\right)_{10}$$

能表示的最小数是

$$1.0000000=(-1)_{10}$$

若$x_7x_6\cdots x_0$是8位整数，x_7为符号位。则其最大数为

$$01111111=(127)_{10}$$

最小数为

$$10000000=(-128)_{10}$$

在补码表示式中，“零”的表示只有一种形式。

因为$[x]_{补}=0.0000000$

当x被视为“负零”时，可以写成

$[x]_{补}=2+x=10+0.0000000$

$=10.0000000$

↑机器内自动丢失

但在机器中，只能存放8位二进制数，上述$[x]_{补}$的最高位被视为自动丢掉，因此，在机器中“负零”的表示也是

$$[x]_{补}=0.0000000$$

因此，补码能表示256个不同的数值。

3.补码的特点与应用

补码表示式的最大特点是便于计算机作加减法等运算，在计算机内用补码参加运算时可以用加减法代替减法运算。例如，设$x=0.11100$，$y=0.01001$，求$x-y=$？

若用补码表示式来参与运算，则便可用下面的等式：

$$[x]_{补}-[y]_{补}=[x]_{补}+[-y]_{补}$$

即

$$\begin{array}{lr} x & 0.11100 \\ +[-y]_{补} & 1.10111 \\ \hline x-y & =10.10011 \end{array}$$

机内被丢弃

1.5.3 反码表示法

1.反码表示法的定义

设x是小数，且小数点的左面是符号位即

$$x=x_n.x_{n-1}x_{n-2}\cdots x_1$$

则其反码的定义为

$$[x]_{反}=\begin{cases} x & 0\leqslant x<1 \\ (2-2^{-n})+x & -1<x\leqslant 0 \end{cases}$$

例如，若$x=+0.1100101$，则$[x]_{反}=0.1100101$。

若$x=-0.1100101$，则$[x]_{反}=1.0011010$。

假设x是整数，即

$x=x_{n-1}\cdot x_{n-2}\cdots x_0$（小数点在最右面，$x_n$是符号位）

则其反码的定义为

$$[x]_{反}=\begin{cases} x & 0\leqslant x<2^{n-1} \\ (2^n-1)+x & -2^{n-1}<x\leqslant 0 \end{cases}$$

例如，若$x=+0110101$，则$[x]_{反}=00110101$。

若$x=-0110101$，则$[x]_{反}=11001010$。

2.反码表示法数的数值范围

设$x_8\cdot x_7x_6\cdots x_1$为8位二进制小数，$x_8$为符号位，则反码能表示的最大数为$0.1111111=\left(\frac{127}{128}\right)_{10}$，能表示的最小数为1.0000000其值相当于$-\left(\frac{127}{128}\right)_{10}$。

若$x_8x_7\cdots x_1$是8位整数，x_8为符号位，则反码能表示的最大数为$01111111=(+127)_{10}$，能表示的最小数为10000000其值相当于$(-127)_{10}$。

在反码表示式中，零的表示式也不是唯一的。这一点只需从其定义出发即可明白。当x为正零时，

$$[x]_{反}=0.00\cdots0$$

当x为负零时，

$$[x]_反 = 2 - 2^{-n} + x$$
$$= 1.11\cdots1 - 0.00\cdots0$$
$$= 1.11\cdots1$$

从而，256个 8 位反码只能表示255个不同的数值。

1.5.4 移码表示法

1.移码表示法的定义

移码多用于表示浮点数的阶码。一般阶码是用n位整数来表示，且用移码表示法。

设$x_{n-1}x_{n-2}\cdots x_0$，其中x_{n-1}是符号位。则其移码的定义为

$$[x]_移 = 2^{n-1} + x \qquad -2^{n-1} \leqslant x \leqslant 2^{n-1} - 1$$

例如，$x = x_7x_6\cdots x_0$且x_7为符号位。

若$x = +1101010$则$[x]_移 = 1.1101010$

若$x = -1101010$则$[x]_移 = 0.0010110$

由于移码的定义是将原数增加了一个2^{n-1}的增量，因此，符号位为 1 时，移码为正值，而为 0 时，则为负值。这一点同上述三个码正好相反。

2.移码表示法数的数值范围

设$x = x_7x_6\cdots x_0$，则由移码的定义可以得到移码能表示的最大数为

$$1.1111111 = (+255)_{10}$$

能表示的最小数为

$$0.0000000 = (0)_{10}$$

由此可见，移码可以表示256个不同数值。

3.移码与外码的关系见下表。

真值表	补 码	移 码
+127	0,1111111	1,1111111
⋮	⋮	⋮
+1	0,0000001	1,0000001
0	0,0000000	1,0000000
⋮	⋮	⋮
−127	1,0000001	0,0000001
−128	1,0000000	0,0000000

可见，移码表示法中，最小数是用全 0 表示的。

1.5.5 定点数与浮点数

计算机中常用的数据表示格式有两种：一种是定点格式，即事先均约定机器中所有数据的小数点位置是固定不变的且通常将数据表示成纯整数或纯小数。另一种是浮点数。其表示格式如图1-1所示。

一、二进制数的定点表示法

在进行加、减运算前需要先按小数点进行对位。如把小数点的位置固定下来，规定它在数字的前面为小数，规定它在数字的最后面为整数，这样再进行运算，不需要对位操作了，这种小数地固定的数称定点数。

一个字长为n位的定点数，一位表示符号，n−1位用来表示数字。

对定点整数，n位二进制数（包括符号位）所表示的整数a的范围是：

$$0 \leqslant |a| \leqslant 2^{n-1} - 1$$

对于定点小数，n位二进制数（包括符号位）所表示的小数a的数值范围是：