

北京工业大学 张秀琼 吴定荣 编著

微型计算机原理 及应用基础

北京科学技术出版社

微型计算机原理及应用基础

张秀琼 吴定荣 编著

000545

北京科学技术出版社

内 容 提 要

全书共十章。内容包括微型计算机的基础知识; Z80 CPU的结构、引脚和时序; 以大量实例说明 Z80指令系统功能、汇编语言程序设计方法和技巧; 中断技术; 多种半导体存储器芯片及电路连接; 从软硬件结合的角度介绍输入/输出接口电路; 详细剖析了一个监控程序; 介绍了三种国产微型计算机TP801、TP805和BCM—Ⅲ,以及三种目前较为流行的16位微处理器,包括IBMPC的CPU Intel 8088。本书最后编选了大量的实验指导书和练习题,并附有参考答案。本书适于作为科技人员的培训教材和自学参考书,也可作为高等院校微型计算机课程的教材。

微型计算机原理及应用基础

张秀琼 吴定荣 编著

北京科学技术出版社出版

(北京西外大街140号)

房山县印刷厂印刷

北京市新华书店发行

各地新华书店经售

开本: 1/16 787×1092毫米 印张: 32 1/8字数: 700千字

1984年9月第一版 1984年9月第一次印刷

印数: 45,000 定价: 3.95元

统一书号: 15274·012

本社书号: 038

序

目前,电子计算机,特别是微处理机的应用,愈来愈广泛普遍,几乎已渗透到人类社会经济、生产、生活的各个领域,影响到社会全局。我们要密切注意当代世界上以电子计算机为主导的新技术迅猛发展的趋势,抓紧时机,努力抓好推广应用,发展我们的经济,实现到本世纪末工农业总产值翻两番的战略目标,进入现代化国家的行列。在首都,发展微型电子计算机的科研、生产和应用,更是加速首都四化建设的一项重要任务。国务院电子计算机和大规模集成电路领导小组制订的规划中,把北京作为计算机研究、生产的北方基地,同时把北京城市建设计算机工程列为国家重点项目。在中央书记处关于首都建设方针的四项指示和最近中共中央、国务院关于对“北京城市建设总体规划方案”的批复中,都强调指出北京要着重发展电子工业。因此,我们首都各行各业,都应该把学习电子计算机、掌握电子计算机,推广应用电子计算机,作为自己的光荣任务,要有高度的责任感和紧迫感。

学习电子计算机,掌握应用技术,并不神秘,既不是高不可攀,又不是轻而易举,象使用录音机和电视机那样方便。必须具有一定文化程度,经过训练、掌握基本知识,才能上机操作。至于编制软件等工作则需要有更多的科学技术知识和专业知识的人员才能承担。因此,培训专业人员的工作是当务之急。只有培养出大量硬件人材和软件人材,才能适应计算机工业生产和推广应用的需要。

为此,北京市人民政府计算机和大规模集成电路领导小组办公室和北京市科学技术委员会领导、组织了计算机技术培训工作以及培训教材的编写、出版工作。北京科学技术出版社从今年第三季度起,将陆续出版适合培训各类人员的教材。

这本《微型计算机原理及应用基础》,内容通俗易懂,编选了大量的实验和习题,附有习题参考答案。经北京市人民政府电子计算机和大规模集成电路领导小组办公室审定,推荐作为北京市计算机培训教材之一,供培训在职科技人员之用,也可供具有中等文化程度的干部、工人、学生阅读和参考。

电子计算机技术是一门综合性的、知识密集、技术密集的科学。通过对这本教材的学习,还只能说是初步掌握入门性的知识,今后一方面要循序渐进,逐步深入地掌握这门科学技术;更重要的是要把学到的知识应用于生产过程控制、事务管理、数据处理、科学计算、运筹优化、数量经济等方面,以提高工作效率、管理水平和经济效益。

北京市科学技术委员会 主任

陈 绳 武

一九八四年六月十三日

前 言

本书是依据编者几年来多次在本校及外校自动化专业和微型计算机培训班讲授《微型计算机》课程的讲义（该讲义自1981~1983年曾修订印刷三次），以及编者参与研制TP801A、TP801B和TP805等微型计算机的体会，加以综合整理而成。

在编写过程中，我们对于每个问题，力求从概念、原理开始讲述，而最后落实到应用实例上。为了帮助读者加深对问题的理解，对难点和关键部分，编选了适量的例题、实验和练习题，并对绝大部分习题提供了参考答案。本书承蒙北京市人民政府电子计算机和大规模集成电路领导小组办公室推荐作为北京市培训教材之一，“大办”的柳维长同志对本书的编写提出了极其宝贵的意见和建议。北京工业大学副教务长、微型计算机研究及应用协会会长杨希武副教授对编写工作进行了指导并审核书稿。计算机科学系顾庆祚副教授仔细校阅了书稿。他们都提出了很中肯的意见和建议。计算机科学系领导同志对编写工作给予了热情支持和鼓励。承蒙杨志高等同志热情帮助，精心绘制了全书的插图，耐心、细致地核对、缮写全书稿件，致使本书得以迅速出版。编者在此表示衷心的感谢。

由于我们水平所限，以及时间仓促，错漏之处在所难免，恳请读者批评、指正。

编者 于北京工业大学

一九八四年四月

目 录

第一章 微型计算机的初级知识	(1)
§ 1.1 概述	(1)
1.1-1 微处理器和微型计算机发展历史	(1)
1.1-2 微型计算机中常用术语	(3)
§ 1.2 进位计数制及不同计数制之间的转换	(4)
1.2-1 进位计数制	(4)
1.2-2 不同进位制数之间的转换	(5)
§ 1.3 计算机中数和编码的概念	(7)
1.3-1 计算机中数的表示	(7)
1.3-2 编码的概念	(9)
§ 1.4 逻辑代数与逻辑电路	(10)
1.4-1 逻辑乘和“与”门	(11)
1.4-2 逻辑加和“或”门	(11)
1.4-3 逻辑非和“非”门	(11)
§ 1.5 逻辑部件	(12)
1.5-1 运算部件	(12)
1.5-2 触发器与寄存器	(15)
§ 1.6 电子计算机的组成原理	(20)
§ 1.7 微处理器、微型机和微型机系统	(21)
1.7-1 微处理器 (Microprocessor)	(21)
1.7-2 微型计算机 (Micro Computer)	(22)
1.7-3 微型计算机系统 (Micro Computer System)	(22)
第二章 Z80-CPU	(23)
§ 2.1 Z80-CPU 的结构特点	(23)
2.1-1 Z80-CPU 的主要特点	(23)
2.1-2 Z80-CPU 的结构	(24)
§ 2.2 Z80-CPU 电路引脚的说明	(33)
2.2-1 Z80-CPU 引脚的功能说明	(34)
2.2-2 Z80和8080读、写和中断信号的对比	(36)
§ 2.3 Z80-CPU 的时序	(36)
2.3-1 取指令操作码周期	(37)
2.3-2 存储器读或写周期	(38)
2.3-3 输入或输出周期	(39)
2.3-4 总线请求响应周期	(41)
2.3-5 中断请求响应周期	(41)
2.3-6 暂停指令周期	(42)

第三章 Z80-CPU 指令系统	(44)
§ 3.1 概述	(44)
§ 3.2 Z80-CPU 指令系统的分类、格式和寻址方式	(45)
3.2-1 指令分类	(45)
3.2-2 指令格式	(46)
3.2-3 寻址方式	(49)
§ 3.3 指令系统	(55)
3.3-1 数据传送和交换	(55)
3.3-2 数据块传送和检索	(63)
3.3-3 算术和逻辑运算指令	(69)
3.3-4 循环移位和移位指令	(78)
3.3-5 位操作指令	(85)
3.3-6 跳转、调用和返回指令	(86)
3.3-7 输入/输出 (I/O) 指令	(98)
3.3-8 基本CPU控制指令	(102)
第四章 汇编语言程序设计	(104)
§ 4.1 汇编语言	(104)
4.1-1 机器语言、汇编语言和高级语言	(104)
4.1-2 汇编语言的组成	(106)
4.1-3 汇编语言语句结构	(107)
§ 4.2 汇编语言程序设计	(113)
4.2-1 概述	(113)
4.2-2 汇编语言基本程序设计	(114)
§ 4.3 浮点数四则运算程序	(152)
4.3-1 在 Z80 机上使用程序实现浮点数运算	(152)
4.3-2 浮点数四则运算程序	(153)
第五章 Z80中断系统	(170)
§ 5.1 中断的概念	(170)
5.1-1 中断的产生	(170)
5.1-2 中断的重要性	(171)
5.1-3 中断源的种类	(171)
5.1-4 中断处理过程	(172)
5.1-5 多重中断的概念	(173)
§ 5.2 Z80 中断系统的特点	(173)
§ 5.3 中断的允许和中断的禁止	(176)
§ 5.4 Z80的优先中断级	(178)
5.4-1 中断排队	(178)
5.4-2 中断嵌套	(184)
§ 5.5 中断控制逻辑	(188)
§ 5.6 Z80-CPU 的中断响应	(190)

5.6-1 CPU 对非屏蔽中断的响应.....	(190)
5.6-2 CPU对屏蔽中断的响应.....	(193)
§ 5.7 中断处理程序设计举例.....	(197)
5.7-1 利用 CTC 定时方式实现顺序控制	(197)
5.7-2 多级中断程序设计	(200)
第六章 微型计算机的存储器	(204)
§ 6.1 概述.....	(204)
§ 6.2 半导体存储器的分类.....	(204)
§ 6.3 读/写存储器 RAM	(207)
6.3-1 RAM 的组成.....	(207)
6.3-2 RAM芯片的实例.....	(211)
6.3-3 由 RAM 芯片组成微型机的读/写存储器.....	(219)
§ 6.4 只读存储器 ROM	(233)
6.4-1 概述	(233)
6.4-2 掩膜式MOS固定存储器的工作原理.....	(233)
6.4-3 可编程的只读存储器.....	(234)
6.4-4 EPROM与CPU的连接	(236)
第七章 输入/输出 (I/O) 接口电路.....	(237)
§ 7.1 概述.....	(237)
7.1-1 输入/输出 (I/O)接口的编址方式.....	(238)
7.1-2 输入/输出 (I/O)的控制方式.....	(239)
§ 7.2 Z80-PIO 并行输入/输出接口芯片	(241)
7.2-1 概述	(241)
7.2-2 PIO 的框图和引脚	(241)
7.2-3 PIO的初始化.....	(245)
7.2-4 PIO应用举例.....	(253)
§ 7.3 Z80-CTC 计数器/定时器芯片	(260)
7.3-1 概述.....	(260)
7.3-2 CTC的方框图及引脚.....	(261)
7.3-3 CTC的工作方式和初始化.....	(265)
7.3-4 CTC与CPU的连接.....	(276)
§ 7.4 CPU 与几种典型外设的接口电路	(276)
7.4-1 单个开关接口电路	(276)
7.4-2 显示器接口电路	(278)
7.4-3 键盘接口电路.....	(280)
7.4-4 行式打印机接口电路.....	(284)
7.4-5 盒式磁带机接口电路.....	(287)
第八章 简单监控程序的剖析	(290)
§ 8.1 概述.....	(290)

§ 8.2	TP-BUG 功能介绍	(291)
§ 8.3	TP-BUG 的总程序流程图、主要程序段及其说明	(293)
8.3-1	TP-BUG的总程序流程图	(293)
8.3-2	TP-BUG 的主要程序流程图及其说明	(293)
第九章	微型计算机系统介绍	(334)
§ 9.1	TP801单板微型计算机	(335)
9.1-1	主要性能指标	(335)
9.1-2	硬件结构框图	(337)
§ 9.2	TP805 灵巧微型计算机	(345)
§ 9.3	BCM-Ⅲ 多功能微型计算机	(347)
第十章	16位微处理器简介	(349)
§ 10.1	概述	(349)
§ 10.2	三种典型16位微处理器	(349)
10.2-1	Intel 8086 和Intel 8088	(349)
10.2-2	MC68000	(359)
10.2-3	Z8000	(361)
实验与习题		(363)
第一部分 编程实验和习题		(363)
实验一	单板机TP801的操作	(363)
实验二	数据传送和交换	(369)
实验三	数据块传送	(370)
实验四	基本算术和逻辑运算	(373)
实验五	跳转·检索	(376)
实验六	循环移位	(380)
习题一	8位传送指令练习	(388)
习题二	16位传送指令练习	(389)
习题三	数据传送和交换指令练习	(390)
习题四	数据块传送和检索指令练习	(390)
习题五	算术、逻辑运算指令练习 (一)	(391)
习题六	算术、逻辑运算指令练习 (二)	(391)
习题七	跳转、调用和返回指令练习	(393)
习题八	输入/输出指令练习	(394)
习题九	简单的循环程序编制	(395)
习题十	字符编码程序	(396)
习题十一	代码转换程序	(398)
习题十二	算术运算程序	(399)
第二部分 接口实验和习题		(400)
实验一	存储器扩充实验	(400)
实验二	数据显示实验	(401)

实验三 Z80—PIO接口实验 (1)	(404)
实验四 Z80—PIO接口实验 (2)	(406)
实验五 Z80—CTC 接口实验	(410)
习题一 存储器扩充和测试	(412)
习题二 字符显示编程练习	(412)
习题三 PIO编程练习	(412)
习题四 CTC编程练习	(413)
习题五 TP801实时控制	(413)
习题参考答案	(417)
第一部分 编程习题答案	(417)
第二部分 接口习题答案	(434)
附录一	(445)
Z80 指令系统索引 (一) 标志操作摘要	(445)
Z80 指令系统索引 (二) (按指令功能分类)	(446)
附录二 TP-BUG监控程序清单	(459)
附录三 Intel 8086/8088指令系统	(501)

第一章 微型计算机的初级知识

§ 1.1 概 述

1.1-1 微处理器和微型计算机发展历史

微处理器是七十年代人类重要的创新之一。自从1971年第一块微处理器 Intel 4004 问世以来,仅有十三年的历史,而它的发展速度却是极其惊人的。它的性能和集成度,几乎每两年增加一倍,产量急剧增加。现在世界上各种微处理器、单片微型计算机等已有480多种,年产值达几十亿美元。

整个计算机工业发展速度本来就很快,然而微处理器及微型计算机出现以来,由于它的体积小,性能价格比最优,特别是价格低廉这一条,使它的应用深入到各个领域,大至航天工业,小到家庭、个人。它的发展之迅速,影响之深远,远远超过了它的前代。现在一小片微型机的功能,超过了50年代初期占地上百平方米、功耗成百千瓦的电子管计算机。性能价格比也提高很快,今天十多美元的单片微型机,性能达到十多年前成十万美元的晶体管计算机系统。

微处理器和微型计算机的发展历史,是和大规模集成电路的发展分不开的。60年代初期的硅平面管工艺和二极管晶体管逻辑电路的发展,促使小规模集成电路(SSI)在1963~1964年出现,金属氧化物半导体(MOS)晶体管工艺又把集成度显著提高。到了60年代后期,在一小片几个平方毫米的硅片上,已有可能容纳几千个晶体管,这就出现了大规模集成电路(LSI)。LSI器件体积小、功耗低、可靠性高。在1970年前期,已经可以生产1K(1024)位的存储器。这些技术和工艺已经为设计生产微处理器和微型计算机打下了基础。可以说,它们是以电子计算器为种子,半导体技术为土壤而结出的丰硕果实。

1971年第一片单片微处理器 Intel 4004问世。它采用了PMOS技术,在4.2毫米×3.2毫米的硅片上,集成了2250个晶体管,16条引脚双列直插式封装。它是4位微处理器,可作4位二进制的并行处理,功能是有限的,一般不适用于通用计算机。但是,配上只读存储器(ROM)、读写存储器(RAM)、移位寄存器以及输入/输出等4个芯片电路,便是MCS-4微型计算机。重要的是它可以装在电动打字机、照相机、电视机、台秤及许多家用器具上,赋予这些器具一定的“智能”,从而大大地提高了这些器具的工作质量。4004本来是作为高级袖珍计算器而设计的,后经改进,成为4040型微处理器,这就是第一代的微处理器。Intel 8008在1972年出现,这是8位微处理器。接着,于1973年又发表了另一种8位微处理器 Intel 8080。这时很多公司都对微处理器产生极大的兴趣,许多厂商都加入这一行业,生产出了一批8位微处理器,如 Motorola 6800, Signetics

2650, MOS TECHNOLOGY 6501、6502和 Rockwell PPS 8 等, 这就是第二代微处理器。自此, 便步入了微处理器、微型计算机的新时代。这时, 微处理器的设计和生产技术已经相当成熟, 并大多朝着如下几个方面努力:

- 提高硅片集成度;
- 提高功能和速度;
- 降低成本;
- 降低功耗;
- 减少组成微型机系统所需的硅片数目;
- 增加外围配套电路的种类并增强其功能;
- 把中央处理器 (CPU)、存储器和输入/输出电路做在一片硅片上; 等等。

1975~1976年出现了集成度更高、性能更强、速度更快的Z80。它是 Zilog 公司在 Intel 8080基础上加以提高而制造出的一种微处理器。Z80采用 NMOS 技术, 40个引脚, 它在好些方面都比 Intel 8080有所提高、有所改进。

自1978年以来, 微处理器又取得许多新成就, 典型的8位微处理器是 Intel 8080、M6800和 Z80, 这三种芯片被广泛应用于各个领域, 应用软件也非常丰富。因此, 一些生产其它类型8位微处理器的厂家也都向这三种微处理器靠近, 有的则改为生产这三种微处理器。但是, 他们并不是停留在原有水平上, 而是在保留原有指令系统的基础上, 翻新工艺, 提高速度, 降低功耗。例如 National Semiconductor 公司推出 NSC800, 它是采用 CMOS 工艺生产Z80。NSC800的工作频率为4 MHz, 而功耗却比 NMOS的 Z80降低很多。

今后, 8位微处理器的品种将减少, 各种产品逐渐靠近三、四种比较成功的8位微处理器的指令系统, 而这几种指令系统基本上保持不变或有所扩充和增强, 以便充分利用已经积累起来的软件, 但工艺、结构和性能将不断翻新和提高。

在1977年左右, 超大规模集成电路 (VLSI) 工艺宣告成功, 在一片硅片上可以集成一万个以上的晶体管。16 K位和64 K位存储器也生产出来了。在这个基础上, 从1978年开始研制16位微处理器, 如 Intel 8086, Zilog Z8000 和 Motorola 68000等。这些就是第一代超大规模集成电路微处理器, 也称为第三代微处理器。16位微处理器中, 还有一类是将原有小型机微型化的, 如 DGC 公司的 MICRO NOVA, DEC 公司的 LSI 11/23 等。

到了80年代初, 超大规模集成电路工艺, 已经可以在单片硅片上集成几十万个晶体管。32位的微处理器宣告制成了。如贝尔实验室的 Bellmac-32A的32位微处理器, 在一块硅片上集成了150,000个晶体管; HP的32位微处理器在1/4平方英寸的硅片上集成了450,000个晶体管, 采用了1微米 NMOS-III 工艺。相对于8位微处理器来说, 32位微处理器可以称为一种高级的微处理器, 例如 Intel 432, National Semiconductor 公司的 16032, Motorola 公司的68020等。32位微处理器的出现, 使微处理器开始进入一个崭新的时代, 微型计算机的概念也发生了变化。32位微型机无论从结构特点、功能、应用范围等方面看, 实质上是过去小型机的微型化。如 Intel 公司的微型计算机 iAPX432称为 Micro main-frame, 意思即是主机的微型化。其功能可以与 IBM 370/158 大型机

相比美。National Semiconductor 公司的16032执行 PASCAL 速度,甚至超过 Cyber 公司的大型机 Cyber173。

随着微处理器和它的支持芯片的出现,微型计算机和微型计算机系统也象雨后春笋般地出现了。微型计算机开始不是以整机出售的,而是用户根据自己的需要,用散件自行组装。自从1975年 MITS 公司生产了 Altair 8800微型机后,微型机才进入市场以整机出售。随之,不仅出售装好的整机,而且还出售完整的微型机系统。据统计,1979~1980年初,微型机的装机台数已达50万台。今后五年全世界微型机生产累计数量将突破1200万台,年增长率达59%。到1986年微型计算机的平均价格若每年下降20%,那时全世界微型机的销售额预计将超过110亿美元。

目前 8 位微型机因为软件齐全、价廉而处于主导地位。8 位微型机今后将继续积累软件。16位微型机出现才只有短短几年的历史,但已经历了两代更新而推进到第三代。而由微型机阵列构成的巨型机的方案也提出不少,有人预言:将来的计算机系统,将是由微处理器组成的微型机网络和阵列结构的机器。

1.1-2 微型计算机中常用术语

1. 位 (Bit)

位是计算机所能表示的最基本最小的数据单位, Bit 是 Binary Digit 的缩写。Bit 只能有两种状态“0”或“1”。

2. 字 (Word)

字是 CPU 与输入/输出 (I/O) 设备和存储器之间传送数据的基本单位。它是数据总线的宽度(根数)。微型计算机字长有: 1 位、4 位、8 位、16 位、32 位等。

3. 字节 (Byte)

一字节 = 8 位 (8 bit)。

4. K 和 KB

$K = 1024 = 2^{10}$; $KB = 1024 \text{ Byte}$, 是用来计算存储器存储容量的单位。

5. 波特率 (Baud)

波特率是数据传送速率的单位, 即 位/秒, 例如波特率为 300 则传送每位数据用 3.33ms。

6. ASCII 码

ASCII 是 American Standard Code for Information Interchange 的英文缩写, 它是美国标准通讯编码, 一般用 7Bit 组合来表示数字、英文字母、符号等可以打印的符号。

7. BCD 码

BCD 是 Binary Code Decimal 的缩写, 它是用 4Bit 组合来表示十进制 0~9 十个数码。

8. 指令 (Instruction)

规定计算机进行某种操作的命令。它是计算机自动控制的依据。目前计算机只能直接识别由 0 和 1 编码组合的指令。

9. 程序 (Program)

程序是指令的有序集合, 是为完成特定任务 (Job) (由计算机能识别的指令) 编排而成的。

10. 指令系统 (Instruction Set)

指令系统指一台计算机所能识别的全部指令。

11. 地址 (Address)

地址指存储器的单元号, 类似于我们住户的住址。例如, 某微型机存储器为 64K ($64 \times 1024 = 65536$) Byte, 则存储器单元 (Location) 的编号从 0 到 65535。每一个 8Bit 就称为一单元, 每一个单元都有唯一的地址号码。

§ 1.2 进位计数制及不同计数制之间的转换

1.2-1 进位计数制

按进位的原则进行计数, 称为进位计数制。每一种进位计数制都有自身的数码个数, 例如十进制有十个数码, 二进制只有两个数码, 而十六进制就有十六个数码。

一、十进制数

十进制数有 0、1、2、3、4、5、6、7、8、9 共十个数码。计数时, 是按“逢十进一”的原则计数的。因此, 同一个数码在不同的位置, 代表不同的数值大小。例如, 6666 这个数中, 从右往左数, 第 1 个 6 代表 6 本身, 第二个 6 代表 6×10^1 (六十), 第三个 6 代表 6×10^2 (六百), 第四个 6 代表 6×10^3 (六千)。所以这个数可以写成 $6666 = 6 \times 10^3 + 6 \times 10^2 + 6 \times 10^1 + 6 \times 10^0$ 。

一般地说, 任意一个十进制数 D , 都可以表示为:

$$D = D_{n-1} \times 10^{n-1} + D_{n-2} \times 10^{n-2} + \dots + D_1 \times 10^1 + D_0 \times 10^0$$

n 为数的位数。上式称为按权展开式。

二、二进制数

二进制数只有两个数码 0 和 1。计数时, 是按“逢二进一”的原则计算的。因此, 不同的数码在不同的位置, 代表不同的数值。例如:

$$(1010)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = (8 + 2)_{10} = (10)_{10}$$

一般地说, 任意一个二进制数 B , 都可以表示为:

$$B = B_{n-1} \times 2^{n-1} + B_{n-2} \times 2^{n-2} + \dots + B_1 \times 2^1 + B_0 \times 2^0$$

在计算机中通常都采用二进制数, 而不采用人们日常都很熟悉的十进制数。这是因为二进制只有“0”和“1”两个数码, 人们可以用两种相反的状态来表示数码。例如: 用高电平代表“1”, 用低电平代表“0”; 用开关的打开代表“0”, 而闭合代表“1”等等两种相反的物理状态来表示“0”和“1”。

三、十六进制数

十六进制数要用 0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F 等共十六个数码表示。计数时，它是“逢十六进位”。同样，数码在不同数位代表不同的值。例如：

$$(6666)_{16} = 6 \times 16^3 + 6 \times 16^2 + 6 \times 16^1 + 6 \times 16^0$$

通常，任意一个十六进制数 H，可以表示为：

$$H = H_{n-1} \times 16^{n-1} + H_{n-2} \times 16^{n-2} + \dots + H_1 \times 16^1 + H_0 \times 16^0$$

表1-1为二进制数、十进制数和十六进制数之间的转换对照表。

表1-1 十进制数、二进制数、
十六进制数对照表

十进制	二进制	十六进制	十进制	二进制	十六进制
0	0 0 0 0	0	9	1 0 0 1	9
1	0 0 0 1	1	10	1 0 1 0	A
2	0 0 1 0	2	11	1 0 1 1	B
3	0 0 1 1	3	12	1 1 0 0	C
4	0 1 0 0	4	13	1 1 0 1	D
5	0 1 0 1	5	14	1 1 1 0	E
6	0 1 1 0	6	15	1 1 1 1	F
7	0 1 1 1	7	16	1 0 0 0 0	10
8	1 0 0 0	8	17	1 0 0 0 1	11

用二进制数表示数值时，书写很长。

所以，在微型机中通常用四位十六进制数来表示 8 位二进制数，用四位十六进制数来表示 16 位二进制数。因为二进制数与十六进制数之间的转换极为方便，而且书写简单，一目了然。

四、八进制数

八进制数用 0、1、2、3、4、5、6、7 共八个数码表示。它是按“逢八进

一”的原则计数的。数码在不同的数位就表示不同的数值。例如：

$$(6666)_8 = 6 \times 8^3 + 6 \times 8^2 + 6 \times 8^1 + 6 \times 8^0$$

通常，任意一个八进制数 Q 可以表示为：

$$Q = Q_{n-1} \times 8^{n-1} + Q_{n-2} \times 8^{n-2} + \dots + Q_1 \times 8^1 + Q_0 \times 8^0$$

在小型机里大多采用八进制数表示。而微型机却用得不多。

1.2-2 不同进位制数之间的转换

一、二进制数与十进制数之间的转换

1. 二进制转换成十进制

这种转换很简单，只要将二进制数的按权展开式相加即可。例如：

$$(10111010)_2 = 1 \times 2^7 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^1 = (186)_{10}$$

2. 十进制数转换成二进制数

十进制整数转换为二进制整数的方法是：连续用 2 去除要转换的十进制整数，直至商为 0 止。每次的余数即为相应数位的二进制数码，第一次除得到的余数是二进制的最低有效位 (LSD = Least Significant Digit) B_0 ，最后一次除 2 得到的余数是二进制的最高有效位 (MSD = Most Significant Digit) B_{n-1} 。

例如：将 $(19)_{10}$ 转换成二进制数

$$19 \div 2 = 9 \quad \text{余数 } 1 \text{ (LSD)}$$

$$9 \div 2 = 4 \quad \text{余数 } 1$$

$$4 \div 2 = 2 \quad \text{余数 } 0$$

$$2 \div 2 = 1 \quad \text{余数 } 0$$

$$1 \div 2 = 0 \quad \text{余数 } 1 \text{ (MSD)}$$

转换结果为: 10011 B (B 表示二进制数)。

二、任意进位制数与十进制数之间的转换

任意进位制数与十进制数之间的转换, 与二进制、十进制之间转换相类似。例如: 十六进制数转换成十进制数, 只要将它按权展开, 然后相加就可以了。

例如: $(FA)_{16} = F \times 16^1 + A \times 16^0$

$$= 15 \times 16 + 10 \times 1$$

$$= (250)_{10} \quad \text{也可写成 } 250 D \text{ (D 表示十进制数) 或 } 250.$$

十进制数转换成十六进制数, 要连续用 16 除欲要转换的十进制数, 直至商为 0。然后, 将每次除得的余数顺序排列起来。

例如: 将 $(253)_{10}$ 转换成十六进制数

$$253 \div 16 = 15 \quad \text{余数 } 13 = (D)_{16}$$

$$15 \div 16 = 0 \quad \text{余数 } 15 = (F)_{16}$$

结果为: $(FD)_{16}$, 也可写成 FDH (H 表示十六进制数)。

三、十六进制数与二进制数之间的转换

因为 $2^4 = 16$, 所以十六进制数的每一个数码都可以用四位二进制数来表示, 见 (表 1-1)。只要熟悉它们之间的对应关系, 十六进制数与二进制数之间的转换是很方便的。

1. 十六进制数转换为二进制数

只要把每一位十六进制数码用相应的四位二进制数码来编码即可以。

例如: 将十六进制数 AFH 转换成二进制数

$$\begin{array}{ccc} \text{A} & & \text{F} \\ \hline 1 & 0 & 1 & 0 & & 1 & 1 & 1 & 1 \end{array}$$

$$\therefore (AF)_{16} = (1010 \ 1111)_2$$

或写成 AFH = 1010 1111B

2. 二进制数转换为十六进制数

转换的方法是从二进制数的最低有效位起由右向左, 每四位分为一组, 最后不足四位的前面补 0, 然后把每四位二进制数, 用相应的十六进制数码代替即可。

例如: 将 10 1100 1001 B 转换成 16 进制数。

$$\begin{array}{ccc} 0 & 0 & 1 & 0 & & 1 & 1 & 0 & 0 & & 1 & 0 & 0 & 1 & B \\ \hline & & 2 & & & C & & & & & 9 & & H \end{array}$$

$\therefore (10\ 1100\ 1001)_2 = (2C9)_{16}$
或写成 $10\ 1100\ 1001\text{B} = 2\text{C}9\text{H}$

§ 1.3 计算机中的数和编码的概念

如前面所述, 在计算机中, 大多数场合都采用二进制数。不仅数据用二进制数表示, 字母、符号也用二进制编码来表示, 甚至控制计算机进行操作的命令也用二进制编码来表示。

1.3-1 计算机中数的表示

一、定点数与浮点数

计算机要处理的数有无符号数和有符号数。有符号数有正数、负数、整数和小数等。

那么, 计算机中数是怎样表示的呢?

计算机的数可以用定点法或浮点法表示, 分别称为定点数和浮点数。采用定点数的计算机称为定点机; 采用浮点数的计算机称为浮点机。

1. 定点数

我们知道, 在普通数字中, 区分正负数是在数的绝对值前面加上符号来表示, 即: “+”号表示正数, “-”号表示负数。计算机中符号也数码化了。即用一位二进制数来表示符号。一般是: 用“0”表示正数, 用“1”表示负数。

定点数是小数固定的数, 用它的最高位表示数的符号。小数点有两种表示方法: 小数点固定在符号位之后, 符号位右边的第一位就是小数点后面的第一位数。

例如: 定点机中读出的两个数是

01010100B及11010100B

则它们分别表示:

+0.1010100B及-0.1010100B

如果计算机用八位来表示数, 则用一位表示符号, 用其余的七位表示数值。它所能表示的绝对值最大的数为0.1111111B; 所能表示绝对值最小的数为0.0000001B。

显然, 在定点机中, 参加运算的数, 它们的绝对值必须小于1。所以, 在定点机中, 在算题之前要选择合适的比例尺, 将所有的数事先通通都变为小数, 如果运算结果大于1, 最高位的数就会丢失掉, 因而结果就不正确, 发生这种情况, 就称为“溢出”(Overflow)。

小数点也可以固定在有效数位的最后, 这种表示法, 只能表示整数, 即参加运算的数均是整数。它还是用最高有效位来表示符号, 其余表示数值。

2. 浮点法

采用浮点法表示数, 是指小数点位置不是固定的, 而是浮动的。

例如: 十进制数39.6可以变为下列各种形式: