



普通高等教育“十一五”国家级规划教材

● 高等院校计算机专业及专业基础课系列教材

微型计算机 基本原理与应用 (第二版)

王克义 编著

北京大学出版社





普通高等教育“十一五”国家级规划教材

微型计算机基本原理与应用

(第二版)

王克义 编著



北京大学出版社
PEKING UNIVERSITY PRESS

内 容 简 介

本书全面、系统地介绍了微型计算机的基本组成结构、工作原理和典型接口技术。主要包括:数据在计算机中的表示与运算方法,计算机的基本结构与工作过程,微处理器结构,指令系统与汇编语言程序设计,存储器及其接口,输入/输出与DMA技术,中断及中断控制器,串并行通信及其接口电路,总线及总线标准,80x86/Pentium保护模式原理与结构,高性能微处理器的先进技术及典型结构等。

本书内容精练,层次清楚,实用性强;在注重讲解基本概念的同时,也十分注意反映微型计算机发展中的新知识、新技术。本书可作普通高校理工科各专业计算机基础课程教材,也可作为成人教育以及各类职业学校的教材。

图书在版编目(CIP)数据

微型计算机基本原理与应用/王克义编著. —2版. —北京:北京大学出版社,2010.7
(高等院校计算机专业及专业基础课系列教材)
ISBN 978-7-301-16554-6

I. 微… II. 王… III. 微型计算机—高等学校—教材 IV. TP36

中国版本图书馆CIP数据核字(2010)第002955号

书 名: 微型计算机基本原理与应用(第二版)

著作责任者: 王克义 编著

责任编辑: 沈承凤

标准书号: ISBN 978-7-301-16554-6/TP·1071

出版发行: 北京大学出版社

地 址: 北京市海淀区成府路205号 100871

网 址: <http://www.pup.cn> 电子信箱: zpup@pup.pku.edu.cn

电 话: 邮购部 62752015 发行部 62750672 编辑部 62752038 出版部 62754962

印刷者: 河北滦县鑫华书刊印刷厂

经 销 者: 新华书店

787毫米×1092毫米 16开本 34.25印张 855千字

1997年10月第1版

2010年7月第2版 2010年7月第1次印刷

定 价: 60.00元

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究

举报电话: (010)62752024 电子信箱: fd@pup.pku.edu.cn

前 言

“微型计算机原理”是高等学校理工科各专业的一门重要计算机基础课程,也是理工科大学学习和掌握计算机科学技术基础、汇编语言程序设计及常用接口技术的入门课程。通过本课程的学习,可以使学生在理论和实践上掌握微型计算机的基本组成和工作原理,建立计算机系统整体概念,具备利用微机技术进行软、硬件开发的初步能力。学习本课程对于掌握现代计算机的基本概念和技术,以及学习后续有关计算机课程(如计算机体系结构、操作系统、计算机网络、嵌入式系统等)均具有重要的意义。本书是该课程使用的基本教材。

众所周知,现代计算机技术发展极为迅速,新知识、新技术日新月异,层出不穷。本书命名为《微型计算机基本原理与应用》,旨在加强计算机基本概念、基本原理的阐述与介绍。同时,本书注重知识整合,在内容的选取上,尽力摒弃那些陈旧、重复的内容,精心选择课程内容体系中的核心知识和典型技术。在写法上,力求概念准确,表述清晰。

本书以 80x86/Pentium 系列微型计算机为背景机,全面、系统地介绍了微型计算机的基本结构、工作原理及典型接口技术。全书共分 16 章,从内容上可划分为四个知识单元:① 计算机的基本结构及工作原理(第 1、2、3、7、13、14 章);② 指令系统及汇编语言程序设计(第 4、5、6 章);③ I/O 接口技术(第 8、9、10、11、12 章);④ 高性能微处理器的先进技术及典型结构(第 15、16 章)。学习本书的预备知识为数字电路及逻辑设计基础知识。

与其他同类教材不同的是,本书在第 1 章和第 2 章中有意识地融入了传统的“计算机组成原理”教材中的相关核心内容,如定点数与浮点数的表示与运算方法、典型的加减法硬件结构、硬布线控制器及微程序控制器的原理与实现技术等,并在第 3 章简要介绍了微处理器的编程结构之后,用第 4、5、6 三章的篇幅翔实地讲解了 80x86 的指令系统及汇编语言程序设计的基本概念与方法;后续章节则相继介绍了微处理器的内部结构及外部功能特性、存储器及其接口、输入/输出控制、可编程接口电路、总线及总线标准以及高性能微处理器的相关概念和技术等;其中,在第 15 章中,专门介绍了 80x86/Pentium 处理器的保护模式的原理和结构,如描述符及描述符表、保护模式的存储管理和地址转换以及多任务的实现及保护机制等;在第 16 章,重点介绍了现代高性能微处理器的多项先进技术和典型结构,如指令级并行及流水线中的“相关”及其处理技术、超标量流水线技术、超长指令字结构及 RISC 技术等。最后,还介绍了多核处理器的设计理念及现代 PC 机主板的结构和技术特点。

本书可供 48~60 学时的课堂教学使用,有些章节的内容可根据不同的教学要求进行适当取舍。每章后面列出的思考题与习题,主要供理解和复习本章基本内容而用,本书最后给出了部分习题的参考答案。

另外,鉴于“微型计算机原理”课程是技术性、实践性较强的课程,因此在教学中应安排相应的实验及编程上机环节。教师可根据具体实验设备及上机条件,安排适当的接口实验及汇编程序上机内容。对于尚不具备专门的微机接口实验设备的教学环境,教师可结合 PC 机上

已配备的键盘、鼠标及显示器等基本 I/O 设备,组织相应的接口实验内容,如键盘输入、显示器输出编程,鼠标器编程等,从而培养学生的 I/O 接口编程能力。关于这方面的内容,请参见第 6 章的介绍。

本书是在作者近年承担北京大学计算机系本科生、北京大学理科实验班教学实践的基础上编写而成的,并参考和吸收了国外较新的同类教科书及国内兄弟院校优秀教材的有关内容,在此,特向有关作者一并致谢。

在本书的编写和出版过程中,承蒙北京大学信息科学技术学院及北京大学出版社领导的热情支持和指导,北京大学出版社理科编辑室陈小红主任、责任编辑沈承凤高级工程师为本书的编辑、出版给予了大力支持和帮助。在此,谨向他/她们表示衷心的感谢。

由于编者的水平所限,书中一定存在不少差错和疏漏,诚请广大读者及专家批评指正。

编者

2010 年 2 月于北京大学

E-mail: wky@pku.edu.cn

目 录

第 1 章 数据在计算机中的表示与运算方法	(1)
1.1 进位计数制	(1)
1.1.1 进位计数制及其基数和权	(1)
1.1.2 计算机中几种常用的进位计数制	(1)
1.2 不同进位制数之间的转换	(4)
1.2.1 二进制数转换为十进制数	(4)
1.2.2 十进制数转换为二进制数	(4)
1.2.3 任意两种进位制数之间的转换	(7)
1.3 二进制数的算术运算和逻辑运算	(8)
1.3.1 二进制数的算术运算	(8)
1.3.2 二进制数的逻辑运算	(9)
1.3.3 移位运算	(10)
1.4 数据在计算机中的表示形式	(11)
1.4.1 机器数与真值	(11)
1.4.2 四种常见的机器数形式	(12)
1.4.3 数的定点表示与浮点表示	(16)
1.4.4 二-十进制编码	(20)
1.4.5 字符编码	(21)
1.4.6 二进制信息的计量单位	(22)
1.5 数据在计算机中的运算方法	(23)
1.5.1 定点数加减法运算	(23)
1.5.2 定点数乘法运算	(27)
1.5.3 定点数除法运算	(30)
1.5.4 浮点数的四则运算	(35)
习题一	(38)
第 2 章 计算机的基本结构与工作过程	(39)
2.1 计算机的基本结构	(39)
2.1.1 存储程序概念与冯·诺伊曼计算机基本结构	(39)
2.1.2 计算机的基本组成框图及功能部件简介	(40)
2.2 计算机的工作过程	(44)
2.2.1 指令与程序	(44)
2.2.2 程序的装入与运行	(45)
2.2.3 计算机的工作过程	(46)
2.3 控制器基本原理	(50)
2.3.1 硬布线控制器原理	(50)

2.3.2	微程序控制器原理	(52)
2.3.3	一个简单的微程序实例	(57)
2.4	计算机系统的组成	(59)
2.4.1	硬件与软件	(59)
2.4.2	计算机系统的基本组成	(59)
2.5	微型计算机的产生和发展	(60)
2.5.1	世界上第一个微处理器和微型计算机的诞生	(60)
2.5.2	微型计算机的发展阶段	(61)
2.6	微型计算机的分类及主要技术指标	(62)
2.6.1	微型计算机的分类	(62)
2.6.2	微型计算机的主要技术指标	(63)
2.7	微型计算机的基本结构及系统组成	(64)
2.7.1	微型计算机基本结构	(65)
2.7.2	微型计算机的系统组成	(66)
	习题二	(66)
第3章	微处理器的编程结构	(68)
3.1	微处理器的工作模式	(68)
3.1.1	实模式	(68)
3.1.2	保护模式	(69)
3.1.3	虚拟 8086 模式	(69)
3.2	微处理器的编程结构	(69)
3.2.1	程序可见寄存器	(70)
3.2.2	80x86/Pentium 处理器的寄存器模型	(70)
3.3	实模式下的存储器寻址	(74)
3.3.1	实模式下的存储器地址空间	(74)
3.3.2	存储器分段技术	(75)
3.3.3	实模式下的存储器寻址	(76)
3.3.4	堆栈	(81)
3.4	实模式输入/输出地址空间	(81)
	习题三	(82)
第4章	寻址方式与指令系统	(83)
4.1	寻址方式	(83)
4.1.1	数据寻址方式	(83)
4.1.2	转移地址寻址方式	(90)
4.2	指令编码	(92)
4.2.1	指令编码格式	(92)
4.2.2	指令编码举例	(95)
4.3	8086 指令系统	(96)
4.3.1	数据传送指令	(96)
4.3.2	算术运算指令	(101)

4.3.3	逻辑运算与移位指令	(112)
4.3.4	串操作指令	(115)
4.3.5	转移指令	(119)
4.3.6	处理器控制指令	(126)
4.4	80286~Pentium 指令系统	(127)
4.4.1	80286 增强和扩充的指令	(127)
4.4.2	80386 增强和扩充的指令	(129)
4.4.3	80486 增强和扩充的指令	(132)
4.4.4	Pentium 增强和扩充的指令	(132)
4.4.5	Pentium Pro 增强和扩充的指令	(133)
4.4.6	Pentium II 增强和扩充的指令	(134)
	习题四	(134)
第 5 章	汇编语言的基本语法	(136)
5.1	汇编语言的特点	(136)
5.2	汇编语言程序结构和基本语法	(137)
5.2.1	示例程序	(137)
5.2.2	基本概念	(138)
5.2.3	指令语句	(144)
5.2.4	伪指令语句	(146)
5.2.5	结构与记录	(153)
5.2.6	宏指令	(155)
5.2.7	简化段定义	(158)
5.3	ROM BIOS 中断调用和 DOS 系统功能调用	(160)
5.3.1	ROM BIOS 中断调用	(160)
5.3.2	DOS 系统功能调用	(160)
5.4	汇编语言程序的上机过程	(161)
5.4.1	.exe 文件的上机过程	(161)
5.4.2	.com 文件及其上机过程	(169)
	习题五	(170)
第 6 章	汇编语言程序设计及应用	(174)
6.1	汇编语言程序设计的基本方法	(174)
6.1.1	程序设计的基本步骤	(174)
6.1.2	程序的基本结构形式	(175)
6.1.3	子程序设计	(177)
6.2	汇编语言的编程应用	(179)
6.2.1	I/O 与通信	(180)
6.2.2	声音与时钟	(180)
6.2.3	乐曲程序	(182)
6.2.4	键盘 I/O	(186)
6.2.5	鼠标器编程	(188)

6.2.6	图形显示	(191)
6.2.7	文件操作	(194)
6.3	Windows 汇编语言程序设计简介	(199)
6.3.1	Windows API 函数	(199)
6.3.2	动态链接库	(199)
6.3.3	指令集选择	(200)
6.3.4	工作模式选择	(200)
6.3.5	函数的原型定义	(200)
6.3.6	Windows 应用程序的基本结构框架	(201)
6.3.7	Win32 汇编语言应用程序实例	(201)
6.3.8	MASM32 汇编与连接命令	(203)
6.4	高级汇编语言技术	(203)
6.4.1	条件汇编	(203)
6.4.2	库的使用	(204)
6.4.3	汇编语言与高级语言的混合编程	(204)
	习题六	(209)
第 7 章	微处理器的内部结构及外部功能特性	(211)
7.1	微处理器的内部结构	(211)
7.1.1	总线接口单元 BIU	(211)
7.1.2	指令 Cache 与数据 Cache	(212)
7.1.3	超标量流水线结构	(212)
7.1.4	动态转移预测及转移目标缓冲器 BTB	(213)
7.1.5	指令预取器和预取缓冲器	(215)
7.1.6	指令译码器	(215)
7.1.7	执行单元 EU	(216)
7.1.8	浮点处理单元 FPU	(216)
7.1.9	控制单元 CU	(216)
7.2	微处理器的外部功能特性	(217)
7.2.1	微处理器的外部引脚信号	(217)
7.2.2	微处理器的总线时序	(222)
	习题七	(230)
第 8 章	I/O 接口与 DMA 技术	(232)
8.1	I/O 接口概述	(232)
8.1.1	I/O 接口的基本功能	(232)
8.1.2	I/O 接口的基本结构	(233)
8.1.3	I/O 端口的编址方式	(234)
8.1.4	I/O 接口的地址分配	(235)
8.1.5	I/O 接口的地址译码及片选信号的产生	(236)
8.1.6	I/O 指令	(237)
8.2	I/O 控制方式	(237)

8.2.1	程序控制方式	(237)
8.2.2	中断控制方式	(239)
8.2.3	DMA 方式	(240)
8.3	DMA 接口技术	(241)
8.3.1	DMA 控制器的基本功能	(241)
8.3.2	DMA 控制器的一般结构	(242)
8.3.3	DMA 控制器的工作方式	(244)
8.3.4	DMA 工作过程	(244)
8.4	可编程 DMA 控制器 8237	(245)
8.4.1	8237 的结构	(246)
8.4.2	8237 的引脚	(247)
8.4.3	8237 的工作方式	(249)
8.4.4	8237 的寄存器及其功能说明	(251)
8.4.5	特殊软件命令	(255)
8.4.6	8237 内部寄存器的寻址	(255)
8.4.7	8237 的工作周期与操作时序	(256)
8.4.8	8237 的页面寄存器	(258)
8.4.9	8237 的编程	(259)
8.4.10	8237 应用举例	(260)
习题八		(264)
第 9 章 中断及中断控制器		(266)
9.1	基本概念	(266)
9.1.1	中断	(266)
9.1.2	中断响应和处理过程	(267)
9.1.3	中断优先级和中断嵌套	(267)
9.2	80x86 实模式的中断系统	(270)
9.2.1	中断的分类	(270)
9.2.2	中断向量表	(270)
9.2.3	外部中断	(273)
9.2.4	内部中断	(275)
9.2.5	中断响应和中断处理过程	(277)
9.3	可编程中断控制器 8259A	(280)
9.3.1	8259A 的引脚功能	(281)
9.3.2	8259A 的内部结构	(281)
9.3.3	8259A 的工作过程	(283)
9.3.4	8259A 的工作方式	(283)
9.3.5	8259A 的级联使用	(285)
9.3.6	8259A 的控制字及编程使用	(287)
9.3.7	8259A 应用举例	(292)
9.4	中断服务程序设计	(293)

9.4.1	中断服务程序的一般结构	(293)
9.4.2	在中断向量表中置入中断向量	(294)
9.4.3	中断服务程序设计	(296)
习题九		(297)
第 10 章 并行通信及其接口电路		(299)
10.1	可编程并行接口的组成及工作过程	(299)
10.1.1	可编程并行接口的组成及其与 CPU 和外设的连接	(299)
10.1.2	可编程并行接口的数据输入输出过程	(300)
10.2	可编程并行通信接口 8255A	(300)
10.2.1	8255A 的性能概要	(300)
10.2.2	8255A 芯片引脚分配及引脚信号说明	(301)
10.2.3	8255A 内部结构方块图	(302)
10.2.4	8255A 的控制字	(303)
10.2.5	8255A 的工作方式	(305)
10.2.6	8255A 的状态字	(311)
10.2.7	8255A 应用举例	(312)
习题十		(315)
第 11 章 串行通信及其接口电路		(317)
11.1	串行通信	(317)
11.1.1	串行通信的特点	(317)
11.1.2	串行通信涉及的常用术语和基本概念	(318)
11.2	串行通信接口标准	(323)
11.2.1	RS-232C 简介	(323)
11.2.2	RS-449、RS-423A 及 RS-422A	(324)
11.2.3	RS-485 标准	(324)
11.2.4	20 mA 电流环串行接口	(325)
11.3	串行接口的基本结构与功能	(326)
11.3.1	串行异步接口的基本结构与功能	(326)
11.3.2	串行同步接口的基本结构与功能	(328)
11.4	可编程串行通信接口 8251A	(329)
11.4.1	USART	(329)
11.4.2	8251A 的基本的功能和工作原理	(330)
11.4.3	8251A 对外接口信号	(332)
11.4.4	8251A 的编程	(335)
11.4.5	8251A 应用举例	(339)
习题十一		(342)
第 12 章 可编程定时器/计数器及其应用		(343)
12.1	概述	(343)
12.2	可编程计数器/定时器 8253	(343)
12.2.1	8253 的主要功能	(343)

12.2.2	8253 的结构框图	(344)
12.2.3	8253 的引脚	(346)
12.2.4	8253 的工作方式	(346)
12.2.5	8253 的初始化编程	(352)
12.2.6	8253 的读出操作	(355)
12.3	8253 的应用	(356)
习题十二		(358)
第 13 章	存储器及其接口	(360)
13.1	概述	(360)
13.1.1	存储器的分类	(360)
13.1.2	存储器的性能指标	(361)
13.1.3	存储系统的层次结构	(361)
13.1.4	内存储器的基本结构及其数据组织	(363)
13.2	半导体存储器及其典型芯片	(365)
13.2.1	可读写存储器 RAM	(366)
13.2.2	只读存储器 ROM	(377)
13.3	存储器接口技术	(384)
13.3.1	存储器与 CPU 连接时应考虑的问题	(384)
13.3.2	存储器接口中的片选控制	(385)
13.3.3	存储器扩展	(388)
13.3.4	存储器接口分析与设计举例	(390)
13.3.5	16 位、32 位、64 位存储器接口	(392)
13.4	改进存储器性能的相关技术	(394)
13.4.1	双端口存储器	(394)
13.4.2	并行主存系统	(395)
13.5	高速缓存(Cache)	(397)
13.5.1	Cache 基本原理	(397)
13.5.2	Cache 的组织方式	(399)
13.5.3	Cache 控制器 82385	(401)
13.5.4	Cache 的更新方式及替换算法	(404)
13.6	虚拟存储器	(406)
13.6.1	虚拟存储器的工作原理	(406)
13.6.2	80x86 的虚拟存储技术	(408)
习题十三		(409)
第 14 章	总线及总线标准	(411)
14.1	概述	(411)
14.1.1	总线	(411)
14.1.2	总线的分类	(411)
14.1.3	总线标准	(413)
14.1.4	总线仲裁	(413)

14.2	PCI 总线	(415)
14.2.1	概述	(415)
14.2.2	PCI 总线的系统结构及特点	(415)
14.2.3	PCI 总线的引脚信号	(417)
14.2.4	PCI 总线周期	(419)
14.2.5	PCI 总线操作时序	(421)
14.2.6	I/O 地址空间	(422)
14.2.7	PCI 总线的配置空间	(423)
14.2.8	PCI 总线 BIOS	(428)
14.3	USB 总线	(432)
14.3.1	USB 概述	(432)
14.3.2	USB 的结构	(433)
14.3.3	USB 协议简介	(434)
14.4	高速总线接口 IEEE 1394	(438)
14.5	高速图形端口 AGP	(439)
	习题十四	(441)
第 15 章	80x86/Pentium 保护模式原理与结构	(442)
15.1	保护模式概述	(442)
15.2	保护模式的主要数据结构	(443)
15.2.1	段描述符	(443)
15.2.2	描述符表	(443)
15.2.3	段选择符	(444)
15.3	保护模式的寄存器模型	(445)
15.3.1	全局描述符表寄存器	(446)
15.3.2	中断描述符表寄存器	(448)
15.3.3	局部描述符表寄存器	(449)
15.3.4	控制寄存器	(450)
15.3.5	任务寄存器	(450)
15.3.6	改变功能的寄存器	(451)
15.4	保护模式的存储器管理和地址转换	(452)
15.4.1	物理地址、线性地址与逻辑地址	(452)
15.4.2	虚拟地址和虚拟地址空间	(453)
15.4.3	虚拟地址空间的分段	(454)
15.4.4	物理地址空间和虚实地址转换	(455)
15.4.5	段式地址转换	(456)
15.4.6	页式地址转换	(458)
15.4.7	转换旁视缓冲器 TLB	(462)
15.5	描述符的格式及功能定义	(463)
15.5.1	非系统段描述符	(463)
15.5.2	系统段描述符	(465)

15.6	保护模式的系统控制指令集	(467)
15.7	多任务和保护	(469)
15.7.1	保护和特权级	(470)
15.7.2	任务切换和任务状态段	(472)
15.7.3	保护模式的中断和异常操作	(475)
15.7.4	保护模式的 I/O 保护	(477)
15.8	虚拟 8086 模式	(479)
	习题十五	(479)
第 16 章	高性能微处理器的先进技术及典型结构	(481)
16.1	高性能微处理器所采用的先进技术	(481)
16.1.1	流水线中的“相关”及其处理技术 ^[6]	(481)
16.1.2	指令级并行	(484)
16.1.3	超标量技术	(485)
16.1.4	超长指令字结构	(485)
16.1.5	超级流水线技术	(486)
16.1.6	RISC 技术	(486)
16.2	高性能微处理器举例	(488)
16.2.1	64 位处理器 Alpha 21064	(488)
16.2.2	Itanium(安腾)处理器——IA-64 架构的开放硬件平台	(490)
16.3	多核处理器简介	(492)
16.3.1	复杂单处理器结构所遇到的挑战	(492)
16.3.2	多核处理器的出现	(493)
16.3.3	多核处理器结构的主要特点	(494)
16.4	现代 PC 机主板典型结构	(494)
16.4.1	芯片组、桥芯片及接口插座	(495)
16.4.2	Pentium PC 主板结构	(496)
16.4.3	Pentium III-1GHz 处理器典型主板布局	(497)
	习题十六	(498)
附录一	8086/8088 指令编码格式	(499)
附录二	DOS 功能调用(INT 21H)	(504)
附录三	BIOS 中断调用	(509)
附录四	调试程序 DEBUG 的使用	(513)
	部分习题参考答案	(517)
	参考文献	(534)

第 1 章 数据在计算机中的表示与运算方法

本章首先介绍计算机中几种常用进位计数制及不同进位制数之间的转换方法,然后介绍数据在计算机中的表示与运算方法,最后简要介绍几种常见的字符编码形式,为后续章节的学习打下必备的基础。

1.1 进位计数制

1.1.1 进位计数制及其基数和权

进位计数制(简称进位制)是指用一组固定的数字符号和特定的规则表示数的方法。在人们日常生活和工作中,最熟悉最常用的是十进制,此外还有十二进制、六十进制等。在数字系统和计算机领域,常用的进位计数制是二进制、八进制及十六进制。

研究和讨论进位计数制的问题涉及两个基本概念,即基数和权。在进位计数制中,一种进位制所允许选用的基本数字符号(也称数码)的个数称为这种进位制的基数。不同进位制的基数不同。例如在十进制中,是选用 0~9 这 10 个数字符号来表示的,它的基数是 10;在二进制中,是选用 0 和 1 这两个数字符号来表示的,它的基数是 2,等等。

同一个数字符号处在不同的数位时,它所代表的数值是不同的,每个数字符号所代表的数值等于它本身乘以一个与它所在数位对应的常数,这个常数叫做位权,简称权(weight)。例如十进制数个位的位权是 1,十位的位权是 10,百位的位权是 100,以此类推。一个数的数值大小就等于该数的各位数码乘以相应位权的总和。例如:

$$\text{十进制数 } 2918 = 2 \times 1000 + 9 \times 100 + 1 \times 10 + 8 \times 1$$

1.1.2 计算机中几种常用的进位计数制

1. 十进制

十进制数有 10 个不同的数字符号(0、1、2、3、4、5、6、7、8、9),即它的基数为 10;每个数位计满 10 就向高位进位,即它的进位规则是“逢十进一”。任何一个十进制数,都可以用一个多项式来表示,例如:

$$312.25 = 3 \times 10^2 + 1 \times 10^1 + 2 \times 10^0 + 2 \times 10^{-1} + 5 \times 10^{-2}$$

式中等号右边的表示形式,称为十进制数的多项式表示法,也叫按权展开式;等号左边的形式,称为十进制的位置记数法。位置记数法是一种与位置有关的表示方法,同一个数字符号处于不同的数位时,所代表的数值不同,即其权值不同。容易看出,上式各位的权值分别为 10^2 、 10^1 、 10^0 、 10^{-1} 、 10^{-2} 。一般地说,任意一个十进制数 N 都可以用多项式表示法写成如下形式:

$$(N)_{10} = \pm (K_{n-1} \cdot 10^{n-1} + K_{n-2} \cdot 10^{n-2} + \cdots + K_1 \cdot 10^1 + K_0 \cdot 10^0 + K_{-1} \cdot 10^{-1} \\ + \cdots + K_{-m} \cdot 10^{-m}) = \pm \sum_{i=-m}^{n-1} K_i \cdot 10^i$$

其中, $K_i (n-1 \leq i \leq -m)$ 表示第 i 位的数字符号, 可以是 10 个数字 (0~9) 符号中的任何一个, 由具体的十进制数 N 来确定。 m, n 为正整数, m 为小数位数, n 为整数位数。

实际的数字系统以及人们日常使用的进位计数制并不仅仅是十进制, 其他进位制的计数规律可以看成是十进制计数规律的推广。对于任意的 R 进制来说, 它有 R 个不同的数字符号, 即基数为 R , 计数进位规则为“逢 R 进一”。数 N 可以用类似上面十进制数的多项式表示法书写如下:

$$(N)_R = \pm (K_{n-1} \cdot R^{n-1} + K_{n-2} \cdot R^{n-2} + \cdots + K_1 \cdot R^1 + K_0 \cdot R^0 + K_{-1} \cdot R^{-1} + \cdots + K_{-m} \cdot R^{-m}) = \pm \sum_{i=-m}^{n-1} K_i \cdot R^i$$

其中 $K_i (n-1 \leq i \leq -m)$ 表示第 i 位的数字符号, 可以是 R 个数字符号中的任何一个, 由具体的 R 进制数 N 来确定。 m, n 为正整数, m 为小数位数, n 为整数位数。若 $R=2$, 即为二进制计数制。它是数字系统特别是电子计算机中普遍采用的进位计数制。

2. 二进制

二进制的基数为 2, 即它所用的数字符号个数只有两个 (“0” 和 “1”)。它的计数进位规则为“逢二进一”。

在二进制中, 由于每个数位只能有两种不同的取值 (要么为 0, 要么为 1), 这就特别适合于使用仅有两种状态 (如导通、截止; 高电平、低电平等) 的开关元件来表示, 一般是采用电子开关元件, 目前绝大多数是采用半导体集成电路的开关器件来实现。

对于一个二进制数, 也可以用类似十进制数的按权展开式予以展开, 例如二进制数 11011.101 可以写成:

$$(11011.101)_2 = 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$$

一般地说, 任意一个二进制数 N , 都可以表示为:

$$(N)_2 = \pm (K_{n-1} \cdot 2^{n-1} + K_{n-2} \cdot 2^{n-2} + \cdots + K_1 \cdot 2^1 + K_0 \cdot 2^0 + K_{-1} \cdot 2^{-1} + \cdots + K_{-m} \cdot 2^{-m}) = \pm \sum_{i=-m}^{n-1} K_i \cdot 2^i$$

其中, K_i 为 0 或 1, 由具体的数 N 来确定。 m, n 为正整数, m 为小数位数, n 为整数位数。

二进制数的优点不仅仅是由于它只有两种数字符号, 因而便于数字系统与电子计算机内部的表示与存储。它的另一个优点就是运算规则的简便性, 而运算规则的简单, 必然导致运算电路的简单以及相关控制的简化。后面我们将具体讨论二进制算术运算及逻辑运算的规则。

3. 八进制

八进制数的基数 $R=8$, 每位可能取 8 个不同的数字符号 0、1、2、3、4、5、6、7 中的任何一个, 进位规则是“逢八进一”。

由于 3 位二进制数刚好有 8 种不同的数位组合 (如下所示), 所以一位八进制数容易改写成相应的 3 位二进制数来表示。

八进制	0	1	2	3	4	5	6	7
二进制	000	001	010	011	100	101	110	111

这样, 把 1 个八进制数每位变换为相等的 3 位二进制数, 组合在一起就成了相等的二进制数。

例 1.1 将八进制数 53 转换成二进制数。

八进制	5	3
	↓	↓
二进制	101	011

所以, $(53)_8 = (101011)_2$ 。

例 1.2 将八进制数 67.721 转换成二进制数。

八进制	6	7.	7	2	1
	↓	↓	↓	↓	↓
二进制	110	111.	111	010	001

所以, $(67.721)_8 = (110111.111010001)_2$ 。

例 1.3 将二进制数转换成八进制数。

二进制	101	111	011.	011	111
	↓	↓	↓	↓	↓
八进制	5	7	3.	3	7

所以, $(101111011.011111)_2 = (573.37)_8$ 。

显然, 用八进制比二进制书写要简短、易读, 而且与二进制间的转换也较方便。

4. 十六进制

十六进制数的基数 $R=16$, 每位用 16 个数字符号 0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F 中的一个表示, 进位规则是“逢十六进一”。

由于 4 位二进制数刚好有 16 种不同的数位组合(如下所示), 所以一位十六进制数可以改写成相应的 4 位二进制数来表示:

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

这样, 把一个十六进制数的每位变换为相等的 4 位二进制数, 组合在一起就变成了相等的二进制数。

例 1.4 十六进制数转换为二进制数。

十六进制	D	3	F
	↓	↓	↓
二进制	1101	0011	1111

所以, $(D3F)_{16} = (110100111111)_2$ 。