



北京市高等教育精品教材立项项目

普通高等教育计算机规划教材

微型机系统 与接口技术

孙一林 主编
彭波 等编著



提供电子教案

下载网址 <http://www.cmpedu.com>



机械工业出版社
CHINA MACHINE PRESS



北京市高等教育精品教材立项项目
普通高等教育计算机规划教材

微型机系统与接口技术

孙一林 主编

彭波 等编著

机械工业出版社

本书主要介绍了计算机基础知识、数据存储器（RAM、ROM）、计算机组成电路、计算机总线、机器指令系统以及输入/输出接口的组成和工作原理，并通过使用单片微计算机设计的一些应用系统加强对计算机原理的理解，同时掌握微计算机应用系统的硬、软件的设计。

本书可作为大学、大专等院校与计算机相关专业的本科或专科生使用的教材，也可作为学习微型计算机原理与应用的初、中级学者使用的参考书。

图书在版编目（CIP）数据

微型机系统与接口技术 / 孙一林主编. —北京：机械工业出版社，2010.1
（普通高等教育计算机规划教材）

ISBN 978-7-111-29566-2

I. 微… II. 孙… III. ①微型计算机-理论-高等学校-教材②微型计算机-接口设备-高等学校-教材 IV. TP36

中国版本图书馆 CIP 数据核字（2010）第 009220 号

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）

责任编辑：张宝珠

责任印制：洪汉军

北京市朝阳展望印刷厂印刷

2010 年 3 月第 1 版·第 1 次印刷

184mm×260mm·15.5 印张·381 千字

0001—3000 册

标准书号：ISBN 978-7-111-29566-2

定价：27.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务

网络服务

社服务中心：（010）88361066

门户网：<http://www.cmpbook.com>

销售一部：（010）68326294

教材网：<http://www.cmpedu.com>

销售二部：（010）88379649

读者服务部：（010）68993821

封面无防伪标均为盗版

出版说明

信息技术是当今世界发展最快、渗透性最强、应用最广的关键技术，是推动经济增长和知识传播的重要引擎。在我国，随着国家信息化发展战略的贯彻实施，信息化建设已进入了全方位、多层次推进应用的新阶段。现在，掌握计算机技术已成为 21 世纪人才应具备的基础素质之一。

为了进一步推动计算机技术的发展，满足计算机学科教育的需求，机械工业出版社聘请了全国多所高等院校的一线教师，进行了充分的调研和讨论，针对计算机相关课程的特点，总结教学中的实践经验，组织出版了这套“普通高等教育计算机规划教材”。

本套教材具有以下特点：

- (1) 反映计算机技术领域的新发展和新应用。
- (2) 注重立体化教材的建设，多数教材配有电子教案、习题与上机指导或多媒体光盘等。
- (3) 针对多数学生的学习特点，采用通俗易懂的方法讲解知识，逻辑性强、层次分明、叙述准确而精炼、图文并茂，使学生可以快速掌握，学以致用。
- (4) 符合高等院校各专业人才的培养目标及课程体系的设置，注重培养学生的应用能力，强调知识、能力与素质的综合训练。
- (5) 适合各类高等院校、高等职业学校及相关院校的教学，也可作为各类培训班和自学用书。

机械工业出版社

前 言

本书是一本适合大学、大专等理、工科院校学生学习计算机原理，掌握微型计算机应用的技术型实用教科书。本书分为原理篇和应用篇，原理篇阐述了计算机共同的组成成分和工作原理，应用篇是以 Intel 51 系列单片微型机为模型，分析其组成部分和工作原理，应用其设计实用系统，通过应用系统验证计算机工作原理。

为配合理解和掌握计算机的工作原理及其应用，每章配有相关的习题以及解答参考提示，应用篇中的实际应用系统硬、软件的设计都是在普通 PC 微型计算机上实现的，使用 ISP 技术完成应用系统硬、软件的结合，教材中涉及的每个实际应用系统都得到了实验的验证，实际应用系统硬、软件的设计文档可在出版社网站上下载到。

本教材由北京师范大学信息科学与技术学院孙一林副研究员主编，参加编写工作的有中国农业大学信息与电气工程学院彭波教授、北京师范大学信息科学与技术学院蔺东辉老师、北京联合大学师范学院电子信息系曾文琪老师、以及北京师范大学信息科学与技术学院饶俊阳、徐方磊、舒能等同学，另外，在该教材的编写过程中得到了李立文、曾跃进、袁钢、陈昕等教师和胡治国、崔永普、张伟娜、徐林、岳乐、孙苗等研究生的帮助，在此表示感谢。

由于作者水平有限，书中难免有错误，敬请读者谅解。如果读者有问题需要与作者讨论，请发送电子邮件到：sunyilin@elec.bnu.edu.cn。

编 者

目 录

出版说明

前言

第 1 部分 原 理 篇

第 1 章 微型计算机基础	1
1.1 计算机综述	1
1.1.1 计算机的发明和发展	1
1.1.2 计算机的种类和用途	2
1.2 逻辑代数基础	2
1.2.1 数据表示法	3
1.2.2 各种进制之间的数值转换	4
1.2.3 非数值型数据的二进制表示	6
1.2.4 二进制数的逻辑运算和算术运算	9
1.3 数字电路基础	13
1.3.1 基础逻辑门电路	13
1.3.2 复合逻辑运算电路	14
1.3.3 全加器电路	14
1.3.4 触发器电路	15
1.3.5 寄存器电路	15
1.3.6 计数器电路	15
1.3.7 编码器电路	16
1.3.8 译码器电路	16
1.4 习题	17
第 2 章 数据存储器	18
2.1 二进制数据存储体	18
2.1.1 存储器类型	18
2.1.2 二进制数据的存储形式	18
2.2 半导体存储器	19
2.2.1 RAM 存储单元	20
2.2.2 ROM 存储单元	22
2.2.3 半导体存储器的组成	25
2.3 磁存储器	27
2.3.1 磁存储器记录数据原理	28
2.3.2 磁存储器的组成	29
2.4 光存储器	30

2.4.1	光存储器记录数据原理	30
2.4.2	光存储器的组成	33
2.5	习题	34
第3章	计算机 CPU 组成电路	36
3.1	可编程顺序、分支、循环控制器	36
3.1.1	顺序循环控制器	36
3.1.2	可编程顺序循环控制器	37
3.2	冯·诺依曼体系	38
3.3	CPU 组成	39
3.3.1	寄存器	39
3.3.2	运算器	40
3.3.3	控制器	50
3.3.4	CPU 内部 Cache	53
3.3.5	CPU 内部结构	54
3.4	CPU 正常工作的基础外围电路	54
3.5	习题	55
第4章	计算机总线	56
4.1	总线概述	56
4.1.1	总线的组成	56
4.1.2	总线的连接	57
4.1.3	总线的分类	57
4.1.4	总线的特性	58
4.1.5	总线的应用	58
4.1.6	数据的传输	59
4.1.7	总线的指标	59
4.2	总线仲裁	60
4.2.1	集中式仲裁	60
4.2.2	分布式仲裁	62
4.3	总线通信方式	63
4.3.1	同步通信方式	63
4.3.2	异步通信方式	63
4.4	常用总线简介	64
4.4.1	CPU 总线	65
4.4.2	PCI 总线	65
4.4.3	USB 总线	67
4.4.4	I ² C 总线	69
4.5	习题	71
第5章	计算机指令系统	72
5.1	指令系统	72

5.1.1	指令的性能	72
5.1.2	指令的格式	73
5.1.3	指令的分类	74
5.1.4	指令助记符	75
5.2	指令执行流程	76
5.2.1	指令的存储	76
5.2.2	指令的执行	76
5.3	寻址方式	78
5.3.1	有效地址和形式地址	78
5.3.2	常见的寻址方式	79
5.4	CISC 和 RISC 的基本概念	82
5.4.1	CISC	82
5.4.2	RISC	83
5.4.3	CISC 和 RISC 的比较	83
5.5	习题	84
第 6 章	基础输入/输出系统	85
6.1	输入/输出接口	85
6.1.1	可编程输入/输出接口	85
6.1.2	输入/输出接口寻址	86
6.1.3	CPU 标准配置接口	87
6.1.4	输入/输出方式	87
6.2	并口接口	89
6.3	中断接口	91
6.3.1	中断概念	91
6.3.2	中断机制	91
6.3.3	可编程中断 I/O 接口	94
6.4	定时器/计数器接口	95
6.5	串口接口	96
6.5.1	串行数据传输——RS—232 标准	96
6.5.2	可编程串行通信 I/O 接口	99
6.6	习题	100

第 2 部分 应 用 篇

第 7 章	Intel 51 微型机硬件系统	101
7.1	Intel 51 系列微型计算机	101
7.2	Intel 51 微型机的 CPU	102
7.2.1	寄存器	102
7.2.2	运算器	103
7.2.3	控制器	103

7.3	Intel 51 微型机存储器结构	106
7.3.1	Intel 51 系统存储空间分配	106
7.3.2	随机存储器	107
7.3.3	只读存储器	109
7.3.4	Intel 51 系统存储空间名称	110
7.4	Intel 51 微型机外部信号线定义	110
7.5	Intel 51 微型机应用系统	113
7.5.1	最小工作系统	113
7.5.2	扩展应用系统	115
7.6	习题	118
第 8 章	Intel 51 微型机指令系统	119
8.1	Intel 51 系列微型机的 CPU 指令系统	119
8.1.1	指令的格式	119
8.1.2	指令操作码助记符以及操作数表示符号	119
8.1.3	寻址方式	120
8.2	数据传送类指令	121
8.2.1	数据传送指令	121
8.2.2	数据传送指令详解	122
8.3	算术运算类指令	125
8.3.1	算术运算指令	126
8.3.2	算术运算指令详解	126
8.4	逻辑运算类指令	130
8.4.1	逻辑运算指令	130
8.4.2	逻辑运算指令详解	130
8.5	控制转移类指令	133
8.5.1	控制转移指令	133
8.5.2	控制转移指令详解	134
8.6	位操作、位控制转移类指令	139
8.6.1	位操作、位控制转移指令	139
8.6.2	位操作、位控制转移指令详解	139
8.7	伪指令	142
8.7.1	伪指令	142
8.7.2	伪指令详解	142
8.8	指令程序	143
8.8.1	指令源代码程序格式	143
8.8.2	指令源代码程序设计	144
8.8.3	源代码程序的编译	145
8.8.4	源代码程序设计示例	145
8.9	习题	148

第 9 章 Intel 51 微型机并口应用	149
9.1 并口接口的工作原理	149
9.1.1 P_0 可编程输入/输出接口	149
9.1.2 P_1 可编程输入/输出接口	153
9.1.3 P_2 可编程输入/输出接口	153
9.1.4 P_3 可编程输入/输出接口	154
9.1.5 并口可编程寄存器的编址	155
9.2 并口接口应用设计	155
9.2.1 单一端口输出方波信号	155
9.2.2 交通灯控制应用设计	157
9.2.3 跑马灯控制应用设计	158
9.2.4 简单键盘输入应用设计	159
9.2.5 8 段 LED 数码管显示设计	162
9.3 习题	165
第 10 章 Intel 51 微型机中断应用	167
10.1 中断接口的工作原理	167
10.1.1 Intel 51 微型机中断管理流程	167
10.1.2 Intel 51 微型机的中断源	167
10.1.3 可编程中断接口的结构	168
10.1.4 中断接口可操作寄存器的定义	169
10.1.5 中断接口可编程序寄存器的编址	172
10.1.6 CPU 响应中断请求	172
10.1.7 中断服务程序框架	175
10.2 中断接口应用设计	176
10.2.1 外部中断系统硬件设计	176
10.2.2 外部中断服务程序设计	177
10.2.3 外部中断应用实例	180
10.3 习题	182
第 11 章 Intel 51 微型机定时器/计数器应用	184
11.1 定时器/计数器接口的工作原理	184
11.1.1 定时器/计数器的逻辑电路	184
11.1.2 定时器/计数器接口可操作寄存器的定义	185
11.1.3 定时器/计数器接口可编程寄存器的编址	188
11.1.4 计数器的 4 种工作模式	189
11.1.5 CPU 对定时器/计数器接口的管理	192
11.2 定时器/计数器接口应用设计	192
11.2.1 计数器应用设计	192
11.2.2 定时器应用设计	196
11.3 习题	205

第 12 章 Intel 51 微型机串口应用	206
12.1 串行通信接口的工作原理	206
12.1.1 可编程串行通信接口逻辑电路	206
12.1.2 串行通信接口可操作寄存器的定义	207
12.1.3 串行通信接口可编程寄存器的编址	208
12.1.4 串行通信接口的 4 种工作模式	209
12.1.5 CPU 对串行通信接口的管理	213
12.2 串行通信接口应用设计	215
12.2.1 串行通信硬件设计	215
12.2.2 串行通信程序设计	217
12.3 习题	222
附录	224
附录 A ISP 技术简介	224
附录 B 各章习题参考提示	226
B.1 第 1 章习题参考提示	226
B.2 第 2 章习题参考提示	227
B.3 第 3 章习题参考提示	227
B.4 第 4 章习题参考提示	228
B.5 第 5 章习题参考提示	228
B.6 第 6 章习题参考提示	229
B.7 第 7 章习题参考提示	230
B.8 第 8 章习题参考提示	231
B.9 第 9 章习题参考提示	232
B.10 第 10 章习题参考提示	233
B.11 第 11 章习题参考提示	233
B.12 第 12 章习题参考提示	234

第 1 部分 原 理 篇

自从冯·诺依曼确定了计算机理论体系后，计算机的组成原理就一直延续至今。所有的计算机都是采用冯·诺依曼理论体系的方式进行工作的。到目前为止，计算机遵循着“摩尔定律”在变化，具体体现在计算机的运行速度，集成电路芯片的集成度，硬、软件功能等的提高。原理篇主要介绍计算机各部件的工作原理以及由各部件组成的计算机的工作原理。

第 1 章 微型计算机基础

本章简要介绍计算机的发明和发展以及计算机的种类和应用；二进制理论；微型计算机系统中主要使用的数字电路。

1.1 计算机综述

计算机的诞生使人类进入了信息时代，在计算机技术高速发展的今天，它正在以惊人的速度进入人类社会的每个角落，计算机的“无处不在”对人类的学习、生产、生活已经产生了巨大的影响，推动人类社会不断地向更高的层次发展。

1.1.1 计算机的发明和发展

计算机可以说是有史以来人类最伟大的发明之一，它的发明起源于 17 世纪利用齿轮原理实现加减运算的“计算器”；在 18 世纪和 19 世纪对其进行了改进，使其实现了乘除运算；在 19 世纪末 20 世纪初，人类发明了电子管，电子管的特性除了对信号可以进行放大外，还可以当开关器件使用，由于利用了电子管开关特性，使得计算机由机械时代进入了电子时代；在 20 世纪 30 年代末到 40 年代初期，美国爱荷华州立大学的 John Vincent Atanasoff（约翰·文森特·阿塔纳索夫）教授和他的研究生 Clifford Berry（克利福特·贝瑞）利用电子管开关特性，使用电容作为存储器，采用逻辑代数的二进位制理论研制出了世界上第一台实现逻辑运算简单功能的电子计算机，它标志着计算机进入了电子时代。

1946 年，世界上第一台正式用于科学计算的电子数字式计算机在美国宾夕法尼亚大学投入运行，被命名为 ENIAC（The Electronic Numerical Intergrator And Computer，电子数值积分计算机），它使用了 17 468 个真空电子管，耗电 174kW，占地 170m²，重为 30t，每秒钟进行 5 000 次加法运算。ENIAC 奠定了电子计算机的发展基础，开辟了一个计算机科学技术的新纪元，也是人类第三次产业革命开始的标志。

ENIAC 诞生后，数学家 John Von Nouma（约翰·冯·诺依曼）提出了重大的改进理论，主要有两点：一是电子计算机应该以二进制为运算基础；二是电子计算机应采用“存储程序”

方式工作，并且进一步明确指出了整个计算机的结构应由 5 个部分组成：运算器、控制器、存储器、输入装置和输出装置。冯·诺依曼理论的提出，解决了计算机的运算自动化的问题和速度配合问题，对以后计算机的发展起到了决定性的作用。直至目前，所有的计算机都是采用冯·诺依曼理论体系的方式进行工作的。

在 ENIAC 诞生后的几十年里，计算机主要使用的电子器件有真空电子管，晶体管，中小规模集成电路和大规模、超大规模集成电路。每一次电子器件的改变都会引起计算机的更新换代。进入晶体管时代后，芯片制造厂商 Intel 公司的创始人之一 Gordon Moore（戈顿·摩尔）发现：在集成电路芯片上可容纳晶体管的数目每隔 18 个月会增加一倍，同时性能也将提升一倍，而价格却下降一倍。这一发现被称为“摩尔定律”。现在的 CPU（中央处理器）芯片已经做到了集成上亿个晶体管，工作频率在 2GHz 以上，每秒可执行的指令接近 1 千亿条。但是，由于人类的制造技术存在着极限，芯片上元器件的几何尺寸不可能无限制地缩小，因此，芯片的集成度不会无限地增加，摩尔定律将会在不久的将来失效。目前，计算机 CPU 技术正在向双核、多核并行处理方向发展，利用多处理器协同并行工作，以便达到提升计算机性能的目的。

1.1.2 计算机的种类和用途

自从电子计算机诞生以来，计算机的发展速度飞快，不仅有运算速度高达万亿次的并行计算机，也有应用于学习和实验室工作的微型计算机，还有广泛应用于工业生产和人们日常生活的嵌入式微处理器。

电子计算机就其规模、运算速度和系统功能而言，可分为巨型、大型、中型、小型和微型计算机。巨型计算机一般用在国防、天文、尖端科学等领域，实现大型的科学计算，例如，战略武器的设计、天气预报、石油勘探等；大中型计算机一般用在大中型企事业单位，满足企业产品设计等需求；小型计算机一般用在中小型企业事业单位和高等院校，满足高校的科学计算、程序设计等需求；微型计算机以其轻、小、价廉、易用等特点被广泛地应用于各行各业，例如，办公自动化、计算机辅助设计、图形图像的数据处理、学习、娱乐以及辅助各种工业或民用设备工作的嵌入式应用等，目前，微型计算机已经做到了无处不在，无所不用的地步。微型计算机与其他类型计算机之间的主要区别在于微型计算机的 CPU 是由一块大规模集成电路芯片组成，而巨型、大型、中型、小型计算机的 CPU 则是由多块大规模集成电路芯片组成。另外，在运算速度、存储容量以及功能等方面也有着较大的差别。但是，由于集成电路芯片技术发展迅速，各种类型的计算机也在飞速发展，因此，计算机种类之间是很难划分清楚的，也许昨天的中型机有可能就是今天的小型机了，或许就是明天的微型机了。

1.2 逻辑代数基础

逻辑代数是英国数学家乔治·布尔（George Bool）在 19 世纪中期创立的，又叫做布尔代数。它是反映和处理逻辑关系的数学工具。20 世纪 30 年代时期，美国人克劳德·香农（Claude E. Shannon）将逻辑代数应用到了开关电路（也被称为数字电路）中，从此开始了逻辑代数的应用时代，它奠定了数字电路设计的数学理论基础。

逻辑代数在数字电路的应用中，其逻辑变量的取值只有 0 和 1，0、1 并不代表数值的大小，它们只代表两种逻辑状态。例如，电平的高与低、开关的闭合与断开等。对数字电路中

逻辑变量的操作，则完全遵循逻辑代数中的规则和公式。

1.2.1 数据表示法

表示数据的方法有多种形式，在人们的日常生活中，采用具有 0~9 十个符号的十进制数来表示一个数据。在数字电路和计算机中处理信息时，则采用的是只有 0 和 1 两个符号的二进制数来表示一个数据。另外，为阅读和书写方便，常采用八进制、十六进制数来表示一个数据，它们是简化和缩短人与计算机对数据描述的一种过渡性的中间数制。

十进制、二进制等都被称为进位计数制，它们是按照进位方式实现计数的一种规则，是用统一的符号和规则表示数据的一种方法。在计算机对数据的处理中，将涉及十进制、二进制、八进制、十六进制等数据。

1. 十进制

一个十进制数可采用下述形式来表示：

$$(N)_{10} = (a_{n-1}a_{n-2} \cdots a_1a_0.a_{-1}a_{-2} \cdots a_{-m})_{10} = \sum_{i=-m}^{n-1} a_i 10^i$$

式中， N 表示一个十进制数； a_i 是 N 中的一位数，表示 0~9 符号中的一个； n 和 m 为正整数， n 表示整数部分的位数， m 表示小数部分的位数；数字“10”表示十进制的基数，“ 10^i ”表示处于 i 位置的 a_i 具有的权重（或称为位权、位值、权值，权重是指同一个数字符号放在不同位置时，它所代表的数值的大小是不同的）。例如，一个十进制数 123.456 按照权重方式展开，其表现形式为

$$(123.456)_{10} = 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 + 4 \times 10^{-1} + 5 \times 10^{-2} + 6 \times 10^{-3}$$

在用十进制计数时，其进位规律是“逢 10 进 1”。

为便于计算机识别，统一数据的表示形式，以及区别其他进制的数据，一个十进制数的表示是在该十进制数据后加一个字母 D (Decimal)。例如，123.456D，字母 D 可省略。

2. 二进制

一个二进制数可采用下述形式来表示：

$$(N)_2 = (a_{n-1}a_{n-2} \cdots a_1a_0.a_{-1}a_{-2} \cdots a_{-m})_2 = \sum_{i=-m}^{n-1} a_i 2^i$$

式中， N 表示一个二进制数； a_i 是 N 中的一位数，被称为 Bit（位），取值为 0 和 1； n 和 m 为正整数， n 表示整数部分的位数， m 表示小数部分的位数；数字“2”表示二进制的基数，“ 2^i ”表示处于 i 位置的 a_i 具有的权重。例如，一个二进制数 11010011 按照权重方式展开，其表现形式为

$$(11010011)_2 = 1 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$

在用二进制计数时，其进位规律是“逢 2 进 1”。

为便于计算机的识别，统一数据的表示形式，一个二进制数的表示是在该二进制数据后加一个字母 B (Binary)，例如，11010011B。

3. 八进制

一个八进制数可采用下述形式来表示：

$$(N)_8 = (a_{n-1}a_{n-2} \cdots a_1a_0.a_{-1}a_{-2} \cdots a_{-m})_8 = \sum_{i=-m}^{n-1} a_i 8^i$$

式中, N 表示一个八进制数, a_i 是 N 中的一位数, 表示 0~7 符号中的一个; n 和 m 为正整数, n 表示整数部分的位数, m 表示小数部分的位数; 数字“8”表示八进制的基数, “ 8^i ”表示处于 i 位置的 a_i 具有的权重。例如, 一个八进制数 123.456 按照权重方式展开, 其表现形式为

$$(123.456)_8 = 1 \times 8^2 + 2 \times 8^1 + 3 \times 8^0 + 4 \times 8^{-1} + 5 \times 8^{-2} + 6 \times 8^{-3}$$

在用八进制计数时, 其进位规律是“逢 8 进 1”。

八进制数是二进制数的一种缩写形式, 便于阅读和书写。一个八进制数的表示是在该八进制数据后加一个字母 O (Octal), 例如, 123.456O。

4. 十六进制

一个十六进制数可采用下述形式来表示:

$$(N)_{16} = (a_{n-1}a_{n-2} \cdots a_1a_0.a_{-1}a_{-2} \cdots a_{-m})_{16} = \sum_{i=-m}^{n-1} a_i 16^i$$

式中, N 表示一个十六进制数; a_i 是 N 中的一位数, 表示 0~9 和 A、B、C、D、E、F 十六个符号中的一个; n 和 m 为正整数, n 表示整数部分的位数, m 表示小数部分的位数; 数字“16”表示十六进制的基数, “ 16^i ”表示处于 i 位置的 a_i 具有的权重。例如, 一个十六进制数 ABC.DEF 按照权重方式展开, 其表现形式为

$$(ABC.DEF)_{16} = A \times 16^2 + B \times 16^1 + C \times 16^0 + D \times 16^{-1} + E \times 16^{-2} + F \times 16^{-3}$$

在用十六进制计数时, 其进位规律是“逢 16 进 1”。

十六进制数是二进制数的另一种缩写形式, 字母 A~F 使用大、小写均可, 一个十六进制数的表示是在该十六进制数据后加一个字母 H (Hexadecimal), 例如, ABC.DEFH 或 abc.defH。

1.2.2 各种进制之间的数值转换

一个数据可以使用各种进制的数值来表示, 因此, 描述数据的各种进制数值之间是可进行转换的。

1. 二进制数转换为十进制数

二进制数转换为十进制数的规则为: 将二进制数按位权展开, 用十进制运算法则求和, 即可得到相应的十进制数。例如, 将二进制数 11010011B 转换为十进制数过程如下:

$$\begin{aligned} (11010011)_2 &= 1 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\ &= 128 + 64 + 0 + 16 + 0 + 0 + 2 + 1 = (211)_{10} \end{aligned}$$

二进制数 11010011B 转换为十进制数为 211D。

2. 十进制数转换为二进制数

十进制数转换为二进制数的规则为: 将十进制数的整数部分和小数部分分别进行转换, 然后合成结果。十进制数整数部分的转换采用“除 2 取余、将余数倒级连”的方法, 将十进制整数反复除 2 直到商为 0, 并取出各次相除的余数, 并将余数倒向级连 (即最后出现的置于

最高位，最先出现的置于最低位)，即得到相应的二进制数。小数部分的转换采用“乘2取整法”，将十进制小数乘2，所得的积中取出整数部分（0或1），作为二进制数的最高位，去掉整数部分后留下的小数部分继续乘2取整，依次得到二进制数的第二位、第三位、……，直到积为0或满足精度要求。

例如：将十进制数 123.456D 转换二进制数。其过程为先将整数和小数部分分开，按“除2取余法”转换 123，再按“乘2取整法”转换 0.456，然后将转换的两个结果合并。

按“除2取余法”将十进制整数 123 转换为二进制数如图 1-1 所示。

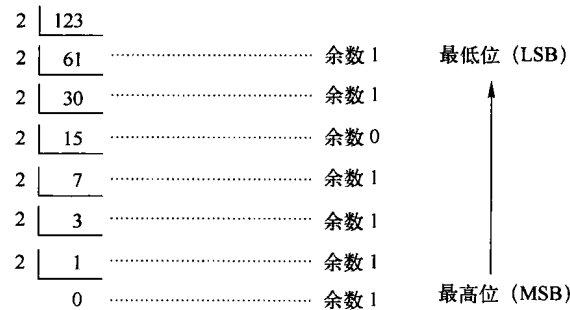


图 1-1 “除2取余法”将十进制数转换为二进制

十进制整数 123D 转换为二进制数为 1111011B。

按“乘2取整法”将十进制小数 456 转换为二进制数如图 1-2 所示。

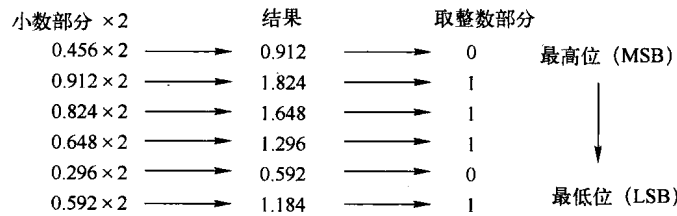


图 1-2 “乘2取整法”将十进制小数转换为二进制

十进制小数 0.456D 转换二进制数为 0.011101B。

合并转换结果，十进制数 123.456D 转换二进制数为 1111011.011101B。

3. 二进制数与八进制数之间的转换

八进制数的每一位对应于 3 位二进制数的 8 种不同取值的组合，八进制数和二进制数之间的转换（对应关系）如表 1-1 所示。

表 1-1 八进制数与二进制数之间的对应关系

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

将二进制数转化为八进制数的规则为：以小数点为基准，将二进制数的整数和小数部分每 3 位分为一组，不足 3 位的分别在整数的最高位前和小数的最低位后加“0”补足，然后每组用等值的八进制数替代，即得八进制数。八进制数转换为二进制数的过程正好与上述过程

相反，将每位八进制数用 3 位二进制数代替即可。例如：

$$(123.456)_8 = (001\ 010\ 011\ .\ 100\ 101\ 110)_2 = (1\ 010\ 011\ .\ 100\ 101\ 11)_2$$

4. 二进制数与十六进制数之间的转换

十六进制数的每一位对应于 4 位二进制数的 16 种不同取值的组合，十六进制数和二进制数之间的转换（对应关系）如表 1-2 所示。

表 1-2 十六进制数与二进制数之间的对应关系

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

将二进制数转化为十六进制数的规则为：以小数点为基准，将二进制数的整数和小数部分每 4 位分为一组，不足 4 位的分别在整数的最高位前和小数的最低位后加 0 补足，然后每组用等值的十六进制数替代，即得十六进制数。十六进制数转换为二进制数的过程正好与上述过程相反，将每位十六进制数用 4 位二进制数代替即可。

例如，一个十六进制数 ABC.DEFH 根据表 1-2 的对应关系转换为二进制数如下：

$$(ABC.DEF)_{16} = (1010\ 1011\ 1100\ .\ 1101\ 1110\ 1111)_2$$

1.2.3 非数值型数据的二进制表示

由于计算机是数字电路组成的，数字电路又是以数字逻辑为基础的，因此，计算机只认识二进制形式的数字。当使用计算机处理非数值型数据，例如，文字、符号、控制命令、声音、图像等时，则需要对它们进行“编码”。所谓编码是用一串二进制数表示某种信息的过程。为使所有的计算机都认识经过编码的数据，所以，“编码”需要“国际化”，即需要有国际标准，例如，BCD、ASCII、Unicode 等码就是具有国际标准的、用二进制数表示的非数值型信息。“编码”为计算机处理非数值型信息提供了可能。

1. BCD 码

BCD（Binary Coded Decimal）码是二进制编码的十进制数，即十进制的 0~9 十个数分别用若干位二进制数来表示。由于 0~9 是十个数，需要用 4 位二进制数来表示，4 位二进制数有 16 种组合，只需要用 10 种组合就可以表示 0~9 十个数，当按自然顺序选择 0000~1001 来表示 0~9 十个数时，其编码方案被称为 8421BCD 码；当选择 0011~1100 来表示 0~9 十个数时，其编码方案被称为余 3BCD 码。表 1-3 给出了十进制的 10 个数和其 BCD 码的对应关系。

表 1-3 十进制的 10 个数和其 BCD 码的对应关系

十进制数	二进制数	8421 码	余 3 码	十进制数	二进制数	8421 码	余 3 码
0	0000	0000	0011	5	0101	0101	1000
1	0001	0001	0100	6	0110	0110	1001
2	0010	0010	0101	7	0111	0111	1010
3	0011	0011	0110	8	1000	1000	1011
4	0100	0100	0111	9	1001	1001	1100