



高等学校计算机科学与技术教材

微机原理

——结构、编程与接口 (第2版)

COMPUTER Science and Technology

□ 王克义 编著

- 原理与技术的完美结合
- 教学与科研的最新成果
- 语言精练，实例丰富
- 可操作性强，实用性突出



清华大学出版社



北京交通大学出版社

高等学校计算机科学与技术教材

微机原理

——结构、编程与接口

(第2版)

王克义 编著

清华大学出版社
北京交通大学出版社

·北京·

内 容 简 介

本书全面、系统地介绍了现代微型计算机的基本结构、工作原理和典型接口技术。主要内容包括：数在计算机中的运算与表示形式，计算机的基本结构与工作过程，微处理器的工作模式与编程结构，寻址方式与指令系统，汇编语言的基本语法，汇编语言程序设计及应用，微处理器的内部结构及外部功能特性，存储器及其接口，I/O接口技术，并行通信及其接口电路，串行通信及其接口电路，计数/定时技术，模拟接口，总线技术，高性能微处理器的先进技术及典型结构，嵌入式系统与嵌入式处理器等。

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。本书可作为普通高等院校理工科各专业计算机基础课程教材，也可作为自学考试和成人教育以及各类职业学校的教材。

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。

版权所有，侵权必究。侵权举报电话：010-62782989 13501256678 13801310933

图书在版编目(CIP)数据

微机原理:结构、编程与接口/王克义编著. —2版. —北京:北京交通大学出版社:清华大学出版社,2017.8

(高等学校计算机科学与技术教材)

ISBN 978-7-5121-3269-6

I. ①微… II. ①王… III. ①微型计算机-高等学校-教材 IV. ①TP36

中国版本图书馆CIP数据核字(2017)第139187号

微机原理——结构、编程与接口

WEIJI YUANLI——JIEGOU, BIANCHENG YU JIEKOU

责任编辑：谭文芳

出版发行：清华大学出版社 邮编：100084 电话：010-62776969 <http://www.tup.com.cn>
北京交通大学出版社 邮编：100044 电话：010-51686414 <http://www.bjtu.com.cn>

印刷者：北京时代华都印刷有限公司

经 销：全国新华书店

开 本：185 mm × 260 mm 印张：23.75 字数：603千字

版 次：2017年8月第2版 2017年8月第1次印刷

书 号：ISBN 978-7-5121-3269-6/TP·845

印 数：1~3 000册 定价：49.00元

本书如有质量问题，请向北京交通大学出版社质监局反映。对您的意见和批评，我们表示欢迎和感谢。

投诉电话：010-51686043, 51686008；传真：010-62225406；E-mail: press@bjtu.edu.cn。

第2版前言

“微机原理与接口技术”是高等学校理工科大学生的一门重要的计算机技术基础课程，也是理工科大学生学习和掌握计算机硬件技术基础、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，可使学生从理论和实践上掌握计算机/微型计算机的基本组成和工作原理，建立微机系统整机概念，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（如计算机系统结构、操作系统、计算机网络、嵌入式系统等）均具有重要意义。本书是该课程使用的基本教材。

本书坚持“基础是根本”的教学理念，注重知识整合，精心选择课程的核心知识和关键技术。全书以80x86/Pentium系列微型计算机为背景机，全面、系统地介绍了计算机及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4个知识单元：

- ① 计算机的基本结构及工作原理（第1、2、3、7、8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9、10、11、12、13、14章）；
- ④ 高性能微处理器及嵌入式系统入门（第15、16章）。

学习本书的预备知识为数字电路及程序设计的一般知识。

本书可供60~70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。每章后面列出的习题，主要供理解和复习本章基本内容而用，书后给出了部分习题的参考答案。

另外，鉴于“微机原理与接口技术”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，以培养学生的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信息科学技术学院（计算机系、智能科学系、电子学系及微电子学系）本科生及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同类教材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。

特别感谢北京大学主干基础课“微机原理”课程组王志军、依那、陆俊林、王道宪、杨延军、高繁民、李洁、让世美诸位老师对本教材编写工作的大力支持和帮助。

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清华大学出版社、北

京交通大学出版社领导的热情支持和指导，责任编辑谭文芳老师做了认真细致的工作，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。

本书的教学资源有 PPT 课件、汇编上机工具、题目及参考答案等，欢迎读者选用，有需要者可直接与编者联系索取（E-mail: wky@pku.edu.cn）。

编者

2017 年 6 月于北京大学信息科学技术学院

第 1 版前言

“微机原理”是高等学校理工科大学生一门重要的计算机技术基础课程，也是理工科大学生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，可以使学生从理论和实践上掌握微型计算机的基本组成和工作原理，建立微机系统整机概念，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。

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 17 章，从内容上可划分为四个知识单元：① 微型计算机的基本结构及工作原理（第 1、2、3、7、13、15 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、14 章）；④ 高性能微处理器及相关技术（第 16、17 章）。学习本书的预备知识为数字电路及逻辑设计基础知识。

本书可供 60~70 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交通大学出版社、清华大学出版社领导的热情支持和指导，责任编辑谭文芳老师为此付出了艰辛和智慧，在此谨向他们表示衷心的感谢。

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

2009 年 2 月于北京大学

目 录

第1章 数在计算机中的运算及表示形式	1
1.1 进位记数制	1
1.1.1 进位记数制及其基数和权	1
1.1.2 几种常用的进位记数制	1
1.2 不同进位制数之间的转换	3
1.2.1 二进制数转换为十进制数	3
1.2.2 十进制数转换为二进制数	3
1.2.3 任意两种进位制数之间的转换	5
1.3 二进制数的算术运算和逻辑运算	6
1.3.1 二进制数的算术运算	6
1.3.2 二进制数的逻辑运算	7
1.3.3 移位运算	8
1.4 数在计算机中的表示形式	9
1.4.1 机器数与真值	9
1.4.2 常见的机器数表示形式	10
1.4.3 数的定点表示与浮点表示	14
1.4.4 二-十进制编码	18
1.5 二进制信息的计量单位	19
习题1	19
第2章 计算机的基本结构与工作过程	21
2.1 计算机的基本结构	21
2.1.1 冯·诺依曼计算机基本结构	21
2.1.2 计算机的基本组成框图及功能部件简介	22
2.2 计算机的工作流程	25
2.2.1 指令与程序	25
2.2.2 计算机的基本工作流程	26
2.3 计算机系统的组成	28
2.3.1 硬件与软件	28
2.3.2 计算机系统的基本组成	29
2.4 微型计算机的分类及主要技术指标	29
2.4.1 微型计算机的分类	29
2.4.2 微型计算机的主要技术指标	30
2.5 微型计算机的基本结构及系统组成	31
2.5.1 微型计算机基本结构	31

2.5.2 微型计算机的系统组成	33
习题2	33
第3章 微处理器的工作模式与编程结构	34
3.1 微处理器的工作模式	34
3.1.1 实模式	34
3.1.2 保护模式	34
3.1.3 虚拟 8086 模式	35
3.2 微处理器的编程结构	35
3.2.1 程序可见寄存器	35
3.2.2 80x86/Pentium 处理器的寄存器模型	35
3.3 微处理器的寻址机制	40
3.3.1 存储器分段技术	40
3.3.2 实模式下的存储器寻址	41
3.3.3 堆栈	45
习题3	46
第4章 寻址方式与指令系统	47
4.1 寻址方式	47
4.1.1 数据寻址方式	47
4.1.2 转移地址寻址方式	52
4.2 指令编码	54
4.2.1 指令编码格式	54
4.2.2 指令编码举例	57
4.3 8086 指令系统	58
4.3.1 数据传送指令	58
4.3.2 算术运算指令	64
4.3.3 逻辑运算与移位指令	73
4.3.4 串操作指令	77
4.3.5 转移指令	80
4.3.6 处理器控制指令	87
4.4 80286 ~ Pentium 指令系统	88
习题4	88
第5章 汇编语言的基本语法	90
5.1 汇编语言的特点	90
5.2 汇编语言程序结构和基本语法	91
5.2.1 示例程序	91
5.2.2 基本概念	92
5.2.3 指令语句	98
5.2.4 伪指令语句	101
5.2.5 宏指令	107

5.2.6	简化段定义	110
5.3	ROM BIOS 中断调用和 DOS 系统功能调用	111
5.3.1	ROM BIOS 中断调用	111
5.3.2	DOS 系统功能调用	112
5.4	汇编语言程序的上机调试	112
5.4.1	建立源文件	113
5.4.2	汇编	113
5.4.3	连接	116
5.4.4	运行	117
5.4.5	调试	117
	习题 5	120
第 6 章 汇编语言程序设计及应用		123
6.1	汇编语言程序设计的基本方法	123
6.1.1	程序设计的基本步骤	123
6.1.2	程序的基本结构形式	123
6.1.3	子程序设计	126
6.2	汇编语言的编程应用	128
6.2.1	I/O 与通信	128
6.2.2	声音与时钟	129
6.2.3	乐曲程序	130
6.2.4	键盘 I/O	133
6.2.5	鼠标器编程	135
6.2.6	图形显示	138
6.3	Windows 环境下汇编语言程序设计	142
6.3.1	Windows API 函数	142
6.3.2	动态链接库	142
6.3.3	指令集选择	143
6.3.4	工作模式选择	143
6.3.5	函数的原型定义	143
6.3.6	Windows 应用程序的基本结构框架	144
6.3.7	Win32 汇编语言应用程序实例	144
6.3.8	MASM 32 汇编与连接命令	145
6.4	汇编语言与高级语言的混合编程	146
6.4.1	内嵌汇编	146
6.4.2	在 C 程序中直接调用汇编子程序	148
6.4.3	汇编语言程序调用 C 函数	151
	习题 6	151
第 7 章 微处理器的内部结构及外部功能特性		153
7.1	微处理器的内部结构及相关技术	153

7.1.1	总线接口单元 BIU	153
7.1.2	指令 Cache 与数据 Cache	154
7.1.3	超标量流水线结构	154
7.1.4	动态转移预测及转移目标缓冲器 BTB	155
7.1.5	指令预取器和预取缓冲器	157
7.1.6	指令译码器	158
7.1.7	执行单元 EU	158
7.1.8	浮点处理单元 FPU	158
7.1.9	控制单元 CU	159
7.2	微处理器的外部功能特性	159
7.2.1	微处理器的外部引脚信号	159
7.2.2	微处理器的总线时序	165
	习题 7	168
第 8 章 存储器及其接口		170
8.1	概述	170
8.1.1	存储器的分类	170
8.1.2	存储器的性能指标	171
8.1.3	存储系统的层次结构	171
8.1.4	内存储器的基本结构及其数据存储格式	173
8.2	半导体存储器的结构及工作原理	175
8.2.1	可读写存储器 RAM	175
8.2.2	只读存储器 ROM	184
8.3	存储器接口	190
8.3.1	存储器与 CPU 连接时应考虑的问题	190
8.3.2	存储器接口中的片选控制	191
8.3.3	存储器扩展	192
8.3.4	存储器接口分析与设计举例	194
8.3.5	双端口存储器	196
8.4	高速缓存	196
8.4.1	Cache 基本原理	196
8.4.2	Cache 的组织方式	198
8.4.3	Cache 的更新方式及替换算法	200
8.5	虚拟存储器	202
8.5.1	虚拟存储器的工作原理	202
8.5.2	80x86 的虚拟存储技术	203
	习题 8	204
第 9 章 I/O 接口技术		206
9.1	I/O 接口概述	206
9.1.1	I/O 接口的基本功能	206

9.1.2	I/O 接口的基本结构	207
9.1.3	I/O 端口的编址方式	207
9.1.4	I/O 接口的地址译码及片选信号的产生	209
9.1.5	I/O 指令	210
9.2	I/O 控制方式	210
9.2.1	程序控制方式	210
9.2.2	中断控制方式	211
9.2.3	DMA 方式	212
9.3	DMA 技术	213
9.3.1	DMA 控制器的基本功能	213
9.3.2	DMA 控制器的一般结构	214
9.3.3	DMA 控制器的工作方式	215
9.3.4	DMA 工作过程	216
9.3.5	可编程 DMA 控制器 8237	217
9.4	中断系统	217
9.4.1	基本概念	218
9.4.2	80x86 实模式的中断系统	220
9.4.3	可编程中断控制器 8259A	226
9.4.4	中断服务程序设计	226
习题 9	231
第 10 章	并行通信及其接口电路	232
10.1	可编程并行接口的组成及工作过程	232
10.1.1	可编程并行接口的组成及其与 CPU 和外设的连接	232
10.1.2	可编程并行接口的数据输入输出过程	233
10.2	可编程并行接口 8255A	233
10.2.1	8255A 的性能概要	233
10.2.2	8255A 芯片引脚分配及引脚信号说明	234
10.2.3	8255A 内部结构框图	235
10.2.4	8255A 的控制字	236
10.2.5	8255A 的工作方式	237
10.2.6	8255A 的状态字	242
10.2.7	8255A 应用举例	243
习题 10	244
第 11 章	串行通信及其接口电路	246
11.1	串行通信	246
11.1.1	串行通信的特点	246
11.1.2	串行通信涉及的常用术语和基本概念	246
11.2	串行通信接口标准	251
11.2.1	RS-232C	251

11.2.2	RS-485	252
11.3	可编程串行通信接口 8251A	252
11.3.1	USART	252
11.3.2	8251A 的基本功能和工作原理	253
11.3.3	8251A 对外接口信号	255
11.3.4	8251A 的编程	258
11.3.5	8251A 应用举例	261
	习题 11	263
第 12 章	计数/定时技术	265
12.1	概述	265
12.2	可编程计数器/定时器 8253	265
12.2.1	8253 的主要功能	265
12.2.2	8253 的结构框图	266
12.2.3	8253 的引脚	267
12.2.4	8253 的工作方式	268
12.2.5	8253 的初始化编程	273
12.2.6	8253 的读出操作	276
12.3	8253 的应用	277
	习题 12	279
第 13 章	模拟接口	281
13.1	模拟接口概述	281
13.2	D/A 转换器	282
13.2.1	D/A 转换器的工作原理	282
13.2.2	D/A 转换器的主要技术指标	285
13.2.3	D/A 转换器芯片	286
13.2.4	D/A 转换器芯片与微处理器的接口	288
13.2.5	D/A 转换器的应用	289
13.3	A/D 转换器	290
13.3.1	基本概念	290
13.3.2	A/D 转换器的工作原理	292
13.3.3	A/D 转换器的主要技术指标	293
13.3.4	A/D 转换器芯片	293
13.3.5	A/D 转换器芯片与微处理器的接口	295
	习题 13	297
第 14 章	总线技术	299
14.1	概述	299
14.1.1	总线	299
14.1.2	总线的分类	299
14.1.3	总线标准	301

14.1.4	总线仲裁	301
14.2	PCI 总线	303
14.2.1	概述	303
14.2.2	PCI 总线的结构及特点	303
14.3	USB 总线	305
14.3.1	USB 概述	305
14.3.2	USB 的拓扑结构	306
14.3.3	USB 线缆及连接器	306
14.4	高速总线接口 IEEE 1394	307
14.5	CAN 总线	308
14.5.1	CAN 总线概述	308
14.5.2	CAN 总线特性	308
14.5.3	CAN 总线的相关概念和技术	309
14.5.4	CAN 通信协议的帧结构	310
14.5.5	CAN 总线位的数值表示	313
14.5.6	CAN 总线传输距离与传输速率的关系	313
14.5.7	CAN 总线的应用环境	314
	习题 14	315
第 15 章	高性能微处理器的先进技术及典型结构	316
15.1	高性能微处理器所采用的先进技术	316
15.1.1	指令级并行	316
15.1.2	超标量技术	317
15.1.3	超长指令字结构	318
15.1.4	超级流水线技术	318
15.1.5	RISC 技术	319
15.2	高性能微处理器举例	320
15.2.1	64 位处理器 Alpha 21064	320
15.2.2	Itanium 处理器—IA-64 架构的开放硬件平台	322
15.3	多核处理器简介	325
15.3.1	复杂单处理器结构所遇到的挑战	325
15.3.2	多核处理器的出现	325
15.3.3	多核处理器结构的主要特点	326
15.4	现代 PC 主板典型结构	327
15.4.1	芯片组、桥芯片及接口插座	327
15.4.2	Pentium PC 主板结构	328
15.4.3	Pentium 4 PC 主板的 I/O 组织结构	329
	习题 15	330
第 16 章	嵌入式系统与嵌入式处理器	331
16.1	嵌入式系统概述	331

16.1.1	嵌入式系统简介	331
16.1.2	嵌入式系统的组成	331
16.1.3	嵌入式系统的分类	334
16.2	嵌入式处理器	335
16.2.1	ARM 系列处理器	335
16.2.2	Intel 8051 系列微控制器	338
16.3	ARM 体系结构	339
16.3.1	ARM 处理器的工作状态	339
16.3.2	ARM 处理器的运行模式	339
16.3.3	ARM 处理器的异常处理机制	340
16.4	ARM 指令系统	342
16.4.1	ARM 指令系统的主要特点	342
16.4.2	ARM 指令的基本格式	342
习题 16		342
附录 A	DOS 功能调用 (INT 21H) (节选)	343
附录 B	BIOS 中断调用 (节选)	344
附录 C	调试程序 DEBUG 的使用	345
	部分习题参考答案	350
	参考文献	363

第 1 章 数在计算机中的运算及表示形式

本章重点介绍数在计算机中的运算与表示方面的基础知识。

1.1 进位记数制

1.1.1 进位记数制及其基数和权

进位记数制（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，常用的进位记数制是二进制、八进制及十六进制。

研究和讨论进位记数制的问题涉及两个基本概念，即基数和权。在进位记数制中，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，这个常数叫作位权，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 几种常用的进位记数制

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} 。

实际的数字系统以及人们日常使用的进位记数制并不仅仅是十进制，其他进位制的计数规律可以看成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”。

2. 二进制

二进制数的基数 $R=2$ ，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：

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

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，把一个八进制数每位变换为相等的 3 位二进制数，组合在一起就变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， $(53)_8 = (101011)_2$ 。

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 1-2】 十六进制数 D3F 转换为二进制数。

十六进制	D	3	F
	↓	↓	↓
二进制	1101	0011	1111

所以， $(D3F)_{16} = (110100111111)_2$ 。

由上面的介绍可以看出，使用八进制或十六进制表示具有如下优点。

☞ 容易书写、阅读，也便于人们记忆。

☞ 容易转换成可用电子开关元件存储、记忆的二进制数。所以，它们是数字系统和计算机中所普遍采用的数据表示形式。

1.2 不同进位制数之间的转换

一个数从一种进位制表示变成另外一种进位制表示，称为数的进位制转换。实现这种转换的方法是多项式替代法和基数乘法。下面结合具体例子来讨论这两种方法的应用。

1.2.1 二进制数转换为十进制数

【例 1-3】 将二进制数 101011.101 转换为十进制数。

这里，只要将二进制数用多项式表示法写出，并在十进制中运算，即按十进制的运算规则算出相应的十进制数值即可。

$$\begin{aligned} (101011.101)_2 &= (2^5 + 2^3 + 2^1 + 2^0 + 2^{-1} + 2^{-3})_{10} \\ &= (32 + 8 + 2 + 1 + 0.5 + 0.125)_{10} \\ &= (43.625)_{10} \end{aligned}$$

这个例子说明，为了求得某二进制数的十进制表示形式，只要把该二进制数的按权展开式写出，并在十进制系统中计算，所得结果就是该二进制数的十进制形式，即实现了由二进制数到十进制数的转换。

顺便指出，用类似的方法可将八进制数转换为十进制数。

【例 1-4】 将八进制数 155 转换为十进制数。

$$\begin{aligned} (155)_8 &= (1 \times 8^2 + 5 \times 8^1 + 5 \times 8^0)_{10} \\ &= (109)_{10} \end{aligned}$$

上述这种用以实现不同进位制数之间转换的方法，称为“多项式替代”法。

1.2.2 十进制数转换为二进制数

1. 十进制整数转换为二进制整数

十进制整数转换为二进制整数的基本方法称为基数除法或“除基取余”法。可概括为：“除基取余，直至商为0，注意确定高、低位”。如例 1-5 所示。

【例 1-5】 将十进制数 935 转换为二进制数。

整个演算过程表示如下：