

■ 高等教育计算机学科“应用型”教材

# 32位

# 汇编语言程序设计

■ 穆玲玲 钱晓捷 主编

■ 贾宗璞 主审



电子工业出版社  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

32位

# 汇编语言程序设计

主 编 王 健 副 编 王 健  
参 编 王 健 王 健

清华大学出版社

www.tup.tsinghua.edu.cn

高等教育计算机学科“应用型”教材

# 32 位汇编语言程序设计

穆玲玲 钱晓捷 主 编

李玉玲 刘凤华 刘爱荣 副主编

贾宗璞 主 审

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书以 32 位 Intel 80x86 CPU 指令系统为主体, 以 MASM 6.x 和 MASM32 为开发工具, 以新的汇编语言程序设计课程教学为目标, 全面系统地介绍 32 位汇编语言程序设计方法。主要内容包括汇编语言程序设计的基础知识、32 位 Intel 80x86 CPU 的体系结构、MASM 伪指令和操作符、32 位程序格式和实模式程序格式以及程序结构设计方法。本书还从不同的方面介绍了汇编语言程序设计的深入内容和实际应用知识, 主要有: 实模式和保护模式下输入输出程序设计、汇编语言与 C/C++ 混合编程。本书各章配有丰富的实例、习题和详细的上机指导, 附录提供了 32 位调试程序 Windbg 的使用说明、MASM 和 LINK 命令参数以及本书所涉及指令的速查表。

本书可作为高等院校汇编语言程序设计课程的教材或参考书, 适用于计算机、电子、自动控制及相关学科的本、专科学生和成教学生, 计算机应用开发人员, 学习微机应用技术的读者等。

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。

### 图书在版编目 (CIP) 数据

32 位汇编语言程序设计 / 穆玲玲, 钱晓捷主编. —北京: 电子工业出版社, 2007.8

高等教育计算机学科“应用型”教材

ISBN 978-7-121-04054-2

I. 3… II. ①穆…②钱… III. 汇编语言—程序设计—高等学校—教材 IV. TP313

中国版本图书馆 CIP 数据核字 (2007) 第 082332 号

责任编辑: 张 旭

印 刷: 北京市天竺颖华印刷厂

装 订: 三河市金马印装有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1 092 1/16 印张: 19.25 字数: 518 千字

印 次: 2007 年 8 月第 1 次印刷

定 价: 26.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 [zits@phei.com.cn](mailto:zits@phei.com.cn), 盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

服务热线: (010) 88258888。

## 编 委 会

- 主 任:** 蒋宗礼 北京工业大学 教授
- 副主任:** 周清雷 郑州大学信息工程学院 副院长/教授  
甘 勇 郑州轻工业学院计算机与通信工程学院 院长/教授  
王传臣 电子工业出版社 副社长
- 委 员:** (按姓氏音序为序)
- 陈志国 河南大学计算机与信息工程学院 书记/教授  
贾宗璞 河南理工大学计算机科学与技术学院 院长/教授  
普杰信 河南科技大学电子信息工程学院 院长/教授  
钱晓捷 郑州大学信息工程学院 副教授  
王爱民 安阳师范学院计算机科学与技术系 系主任/教授  
王清贤 信息工程大学信息工程学院 系主任/教授  
翁 梅 河南农业大学信息与管理科学学院 系主任/副教授  
邬长安 信阳师范学院计算机系 系主任/教授  
徐久成 河南师范大学计算机与信息技术学院 院长/教授  
张红梅 河南工业大学信息学院 副院长/教授  
张亚东 河南财经学院信息学院 院长/教授  
郑秋生 中原工学院计算机学院 院长/副教授
- 秘书组:** 钱晓捷 郑州大学信息工程学院 副教授  
张 旭 电子工业出版社 副编审

## 出版说明

高等教育的教学改革及课程建设总是伴随着科技的进步与生产的发展而发展的。当前高等教育既要培养理论基础扎实、高素质的科研型人才,也要培养具有一定的理论基础更具有较高工程能力的应用型人才。为了满足普通高等院校面向应用的需求,进一步提高高等院校教学质量和教学水平,电子工业出版社与河南省计算机学会共同组织了高等教育计算机学科“应用型”教材的编写和出版工作。

高等教育计算机学科“应用型”教材根据培养目标和对象不同,总结教学改革和教材建设经验,在基础理论方面做出了合理的取舍,同时融入了现代科技应用的成果。这是一种理论与实践、基础知识与现代技术有机结合的教材。

本套教材定位于国内普通高等院校本科、专科的学生,也适用于高职高专、成人教育的学生。教材内容充分考虑学生的知识水平、理解能力和教学要求,遵循由浅入深、循序渐进的原则,适合学生自学和教师教学。

本套教材符合相应教学大纲的基本要求,结合案例(实例)展开教学内容,侧重应用,突出实践,强调理论与实践结合。

本套教材努力从学习者(学生、自学者)的角度阐述理论知识,充分利用图表进行形象化表达,适当补充相关知识内容,引导读者阅读相关书籍。教材内容的选取注重帮助读者建立完整的知识结构,而不是仅仅掌握某个知识单元。教材内容关注计算机技术的迅猛发展,及时补充最新技术。

本套教材努力提供丰富的教学辅助资源,建立师生交流平台,以便于教师、学生使用。读者可以通过电子工业出版社的华信教育资源网站([www.huaxin.edu.cn](http://www.huaxin.edu.cn) 或 [www.hxedu.com.cn](http://www.hxedu.com.cn))了解本套教材的出版和服务的动态信息。

河南省计算机学会  
电子工业出版社

# 前 言

本书以 32 位 Intel 80x86 CPU 指令系统为例，以 MASM32 和 MASM6.x 组成的环境为开发工具，以 Windbg 为调试工具，以 32 位 Windows 控制台应用程序为例，以程序设计技术为主线，全面系统地介绍了汇编语言程序设计的方法，语言浅显，结构清晰，实例丰富。全书内容包括汇编语言程序设计的基础知识，32 位 Intel 80x86 CPU 的体系结构，数据操作编程，各种程序结构的编程，输入输出程序设计和汇编语言与 VC++ 混合编程等内容。

第 1 章：总结性地介绍了在计算机中数据的表示方法和二进制补码运算以及对标志位的影响。

第 2 章：从程序员的角度介绍了 32 位 Intel 80x86 CPU 的体系结构，主要包括 IA-32 CPU 寄存器组织和在 32 位保护模式与 16 位实模式下的内存管理方式。

第 3 章：以两个实例引出 32 位保护模式和 16 位实模式的汇编语言程序格式，介绍程序开发、语句格式、常用伪指令和操作符等内容。

第 4 章：以 MOV 指令和 JMP 指令为例，介绍汇编语言源程序中操作数和地址的寻址方式。

第 5 章：以常用数据操作类指令为例，说明汇编语言中顺序结构的设计方法，内容包括数据传送类指令、算术运算类指令和位操作指令。

第 6 章：介绍分支和循环结构设计的方法，引出转移指令、循环指令等内容。

第 7 章：介绍子程序的设计方法，引出子程序定义和调用指令，子程序声明和调用伪指令，Windows API 函数调用等内容。

第 8 章：介绍在 16 位实模式和 32 位保护模式下输入输出程序设计的方法，引出了输入输出指令、I/O 位图、DOS 调用和读写控制台的 API 函数调用等内容。

第 9 章：介绍串操作类指令，引出了各种数制转换和字符变换的算法。

第 10 章：讨论宏结构的设计方法。

第 11 章：介绍汇编语言和 VC++ 混合编程的基本方法。

书中各章配有一定数量的习题，附录介绍了 Windbg 和 MASM 汇编连接器使用方法，指令速查表，MASM 伪指令和操作符等内容。

本书是针对 32 位 Windows 环境下汇编语言程序设计而编写的，它具有以下特点：

- 全书的内容安排以汇编语言编程的主要技术为主线，将指令的介绍分散在不同程序设计技术中，使指令的学习和掌握汇编语言编程技术融为一体。
- 本书的实例大都以 32 位 Windows 应用程序实现，可以在 Windows 环境下调试执行。
- 将 MASM32 和 MASM6.x 结合，介绍由读者自行建立汇编语言开发环境的方法，以满足不同的开发和调试习惯；强调编程实践，在介绍基础知识后，很快引入完整的源程序开发。

- 详细介绍在实模式和保护模式下实现输入输出的方法,有利于读者进一步学习接口技术课程。
- 介绍了汇编语言与 C/C++混合编程,以及多模块化程序设计的方法,能够应用于实际软件系统的开发。

本书内容自成体系,不需要其他先修知识。当然读者如果具备 C/C++程序的编程经验,可以更好地理解第 7 章子程序语言规范和第 11 章混合编程的内容。汇编语言程序设计是软件设计,并且与计算机的硬件相关,因此读者在计算机或微机原理等方面的知识对深刻理解体会指令功能有很大的好处。

穆玲玲同志承担本书的主编工作,提供写作大纲,并编写序论和第 3, 7, 8 章;李玉玲同志编写第 6, 9 章,刘风华老师编写第 5 章并整理附录,钱晓捷同志对本书的编写提出了宝贵意见,并提供了大量的图表、实例和课件素材,编写输入输出子程序库;刘爱荣同志编写本书的上机指导;史苇杭同志编写第 1 章和习题解答;张行进同志编写第 2 章;周华同志编写第 10 章;周宇同志编写第 4, 11 章。全书由主编穆玲玲进行了认真的统稿,理顺了各章节结构,验证修改了全部程序,编写了输入输出宏库,制作了课件。贾宗璞老师认真审阅了全书,提出了宝贵的修改意见,在此表示诚挚的谢意。

读者可以登录华信教育资源网或 [http://www10.zzu.edu.cn/ie\\_mulingling/](http://www10.zzu.edu.cn/ie_mulingling/), 下载本书配套的电子课件、上机指导、开发环境、源程序实例、输入输出宏库和子程序库等内容。

限于编者的学识水平,书中难免有疏漏和不当之处,敬请广大同行及读者指正。欢迎读者,尤其是使用本书的教师和学生,与本书作者共同探讨相关教学内容、教学方法等问题。主编的电子邮箱: [mulingling@zzu.edu.cn](mailto:mulingling@zzu.edu.cn)。

穆玲玲

2007 年 4 月



# 目 录

绪论	1
第 1 章 计算机中数据的表示	3
1.1 数制	4
1.1.1 十进制	4
1.1.2 二进制	4
1.1.3 十六进制	5
1.1.4 八进制	5
1.1.5 数制间的转换	5
1.2 有符号数的表示	8
1.2.1 原码	8
1.2.2 反码	9
1.2.3 补码	10
1.3 二进制运算	11
1.3.1 算术运算	11
1.3.2 逻辑运算	14
1.4 编码	15
1.4.1 BCD 码	15
1.4.2 ASCII 码	15
1.4.3 UNICODE 标准	17
习题	17
第 2 章 微型计算机体系结构	19
2.1 微机系统概述	20
2.1.1 计算机的硬件	20
2.1.2 计算机的软件	21
2.2 IA-32 微处理器	22
2.2.1 Intel 微处理器发展	22
2.2.2 IA-32 CPU 的操作模式	25
2.2.3 IA-32 CPU 功能结构	26
2.2.4 IA-32 CPU 寄存器组	28
2.3 IA-32 CPU 存储器管理	34
2.3.1 内存管理模型	35
2.3.2 存储器模型与操作模式的关系	40
2.4 外设	40
习题	41

<b>第3章 汇编语言基础</b> .....	42
3.1 汇编语言程序开发过程.....	43
3.1.1 搭建汇编语言开发环境.....	43
3.1.2 汇编程序开发步骤.....	44
3.2 汇编语言程序结构.....	48
3.2.1 汇编语言的语句格式.....	49
3.2.2 汇编语言程序格式.....	49
3.2.3 汇编语言程序格式定义伪指令.....	52
3.3 参数、变量和标号.....	55
3.3.1 数值型参数.....	56
3.3.2 变量定义伪指令.....	61
3.3.3 变量和标号的属性.....	68
习题.....	70
<b>第4章 IA-32 指令格式和寻址方式</b> .....	72
4.1 IA-32 指令格式.....	73
4.2 机器代码格式.....	73
4.3 操作数寻址方式.....	75
4.3.1 立即数寻址.....	75
4.3.2 寄存器寻址.....	77
4.3.3 存储器寻址.....	77
4.4 转移地址寻址方式.....	84
4.4.1 段内直接寻址.....	85
4.4.2 段内间接寻址.....	85
4.4.3 段间直接寻址.....	86
4.4.4 段间间接寻址.....	86
习题.....	87
<b>第5章 数据操作</b> .....	89
5.1 数据传送.....	90
5.1.1 通用数据传送指令.....	90
5.1.2 地址传送指令.....	93
5.2 算术运算.....	96
5.2.1 加法指令.....	96
5.2.2 减法指令.....	98
5.2.3 乘法指令.....	101
5.2.4 除法指令.....	103
5.2.5 符号扩展指令.....	105
5.3 位操作.....	109
5.3.1 逻辑运算指令.....	109
5.3.2 位测试指令.....	112
5.3.3 位扫描指令.....	113

5.3.4	移位指令	114
5.3.5	循环移位指令	116
5.3.6	双精度移位指令	118
	习题	119
<b>第 6 章</b>	<b>分支和循环程序设计</b>	<b>122</b>
6.1	转移指令	123
6.1.1	无条件转移指令 JMP	123
6.1.2	条件转移指令 Jcc	125
6.2	分支程序设计	129
6.2.1	单分支结构程序设计	130
6.2.2	双分支结构程序设计	131
6.2.3	多分支结构程序设计	134
6.3	循环结构设计	139
6.3.1	用转移指令实现循环结构	140
6.3.2	循环指令	145
	习题	149
<b>第 7 章</b>	<b>子程序设计</b>	<b>152</b>
7.1	堆栈	153
7.2	子程序	158
7.2.1	子程序定义伪指令	158
7.2.2	子程序调用和返回指令	159
7.2.3	子程序的参数传递	164
7.2.4	子程序的局部变量	168
7.2.5	子程序的嵌套、递归与重入	170
7.2.6	子程序的应用	173
7.3	Windows API 函数	176
7.4	子程序伪指令的应用	181
7.5	模块化程序设计	183
7.5.1	创建多模块程序	183
7.5.2	子程序库	188
	习题	189
<b>第 8 章</b>	<b>输入输出程序设计</b>	<b>191</b>
8.1	底层输入输出程序设计	192
8.1.1	输入输出指令	193
8.1.2	I/O 程序设计	195
8.1.3	输入输出保护	199
8.2	高层输入输出程序设计	203
8.2.1	中断指令	203
8.2.2	实模式下的系统功能调用	205
8.2.3	利用 Windows API 实现 Win32 控制台输入输出程序	210

8.3	输入输出子程序库	215
8.3.1	32 位输入输出子程序库	215
8.3.2	实模式输入输出子程序库	218
	习题	220
<b>第 9 章</b>	<b>串操作编程</b>	<b>222</b>
9.1	相关指令	223
9.1.1	标志位操作指令	223
9.1.2	串操作指令	224
9.2	串操作应用实例	231
	习题	234
<b>第 10 章</b>	<b>宏结构程序设计</b>	<b>235</b>
10.1	宏汇编	236
10.2	重复汇编	244
10.3	条件汇编	246
10.4	宏的实例	251
	习题	254
<b>第 11 章</b>	<b>汇编语言与 Visual C++ 的混合编程</b>	<b>256</b>
11.1	Visual C++ 嵌入汇编方式	257
11.1.1	嵌入式汇编语句的格式	257
11.1.2	嵌入式汇编语句的规定	258
11.1.3	嵌入式汇编语句的使用	261
11.2	Visual C++ 与汇编语言的模块连接	264
11.2.1	混合编程的约定规则	264
11.2.2	C++ 程序调用汇编子程序	267
11.2.3	汇编语言调用 C++ 子程序	270
	习题	272
<b>附录</b>		<b>275</b>
附录 A	汇编程序 MASM 的伪指令和操作符	276
附录 B	宏汇编命令 ML 和连接命令 LINK 常用参数速查	278
附录 C	调试程序 Windbg	280
附录 D	输入输出宏命令	283
附录 E	I/O 子程序	284
附录 F	指令速查表	285
附录 G	调试程序 DEBUG 主要命令	291
<b>参考文献</b>		<b>295</b>

# 绪 论

计算机在人类社会的工作和生活的各个领域扮演了越来越多的角色。许多人每天坐在计算机前，用 Word 编写文件，用 IE 网上冲浪，用 Outlook 发邮件，偶尔也会在电脑上玩玩小游戏。所有这些能够满足人们某种具体要求的工具叫做软件。大多数人都是软件的使用者，而程序员的任务就是用某种程序设计语言编制这些软件。进行程序设计的语言很多，可以分为低级语言和高级语言。低级语言包括机器语言和汇编语言，高级语言有 C, C++, Java, Delphi, Powerbuilder 等。

程序设计语言中最容易掌握和使用的是高级程序设计语言。高级程序设计语言的特点是接近人类自然语言的语法习惯和数学表达式，与计算机的内部硬件结构无关，因此在用高级语言编写程序时，程序员可以将精力集中于解决问题的算法设计上，而不需要考虑计算机的硬件结构。但是高级语言编写的程序不能被计算机执行，需要借助于编译程序翻译成机器指令后才能够执行，而在这个编译的过程中，可能会造成一定的冗余，浪费一部分地址空间，降低程序的执行速度，这是程序员不能控制的。但是用可以直接控制机器硬件的低级语言就可以解决这个问题。

其实，程序员最早进行软件开发采用的就是机器语言。机器语言是指计算机能够理解和执行的机器指令代码，这些代码是由一串二进制位 0 和 1 组成的。将二进制代码序列输入计算机的内存中，计算机就可以直接解释执行。但是二进制数字在编程的时候非常难记，写出的程序也难以阅读和调试。比如完成两个数据 100 和 256 相加的功能，在 Intel 32 位 CPU 上对应的代码是：

```
10111000 01100100 00000000 00000000 00000000  
00000101 00000000 00000001 00000000 00000000
```

这段代码看起来毫无意义，几乎没有人能够直接读懂它的功能。用机器语言编写程序的最大缺点就是难以理解，容易出错，又很难发现错误。所以机器语言只是在计算机发展的早期或不得已的情况下才被使用。现在，除了有时在程序某处需要直接用机器指令填充外，几乎没有人采用机器语言编写程序了。

为了克服机器语言的缺点，人们就用便于记忆，并能描述指令功能的助记符号来表示机器指令代码；这些助记符号称做助记符，一般是表明指令功能的英语单词或其缩写。用助记符表示的指令叫做汇编格式指令。汇编格式指令与机器指令一一对应，但是比机器指令要容易记忆和理解。汇编格式指令以及使用它们编写程序的规则就形成汇编语言（Assembly Language）。用汇编语言书写的程序就是汇编语言程序，也叫做汇编语言源程序。比如完成两个数据 100 和 256 相加的功能的 MASM 汇编语言程序是：

```
mov eax, 100  
add eax, 256
```

熟悉了有关助记符及对应指令的功能以后，很容易读懂这一段程序。

汇编语言是一种符号语言，它用助记符表示操作码，比机器语言容易理解和掌握，也容易

调试和维护。当然汇编语言写出的程序也需要翻译成机器语言后,才能够被计算机执行,这个翻译的过程叫做“汇编”。完成汇编工作的程序是汇编程序。汇编语言和机器语言都是面向机器的,因此都叫做机器语言,也叫做低级语言。

与高级语言相比,汇编语言复杂、难懂,与处理器密切相关,因此汇编语言程序的通用性和移植性比较差。编写汇编语言程序,要求程序员比较熟悉计算机硬件系统、考虑许多细节问题,因此汇编语言编程具有编程烦琐、调试维护困难的问题。但是汇编语言可以直接有效地控制计算机硬件部件,执行速度和效率与机器语言接近,不存在地址空间和执行时间的浪费,编写的程序较为简练,运行速度较高,在“时间”和“空间”效率上都可以达到最优化。将高级语言和汇编语言结合起来编写程序可以取长补短,能达到事半功倍的效果。

汇编语言的主要应用场合有:

① 程序要具有较快的执行速度,或者只能占用较小的存储容量。例如,操作系统的核心代码,实时控制应用程序,智能仪器仪表的控制程序等;

② 程序与计算机硬件密切相关,程序要直接、有效地控制硬件。例如,外设的驱动程序,I/O 接口电路的初始化程序段;

③ 大型软件需要提高性能、优化处理的部分。例如,计算机系统频繁调用的子程序、动态链接库等;

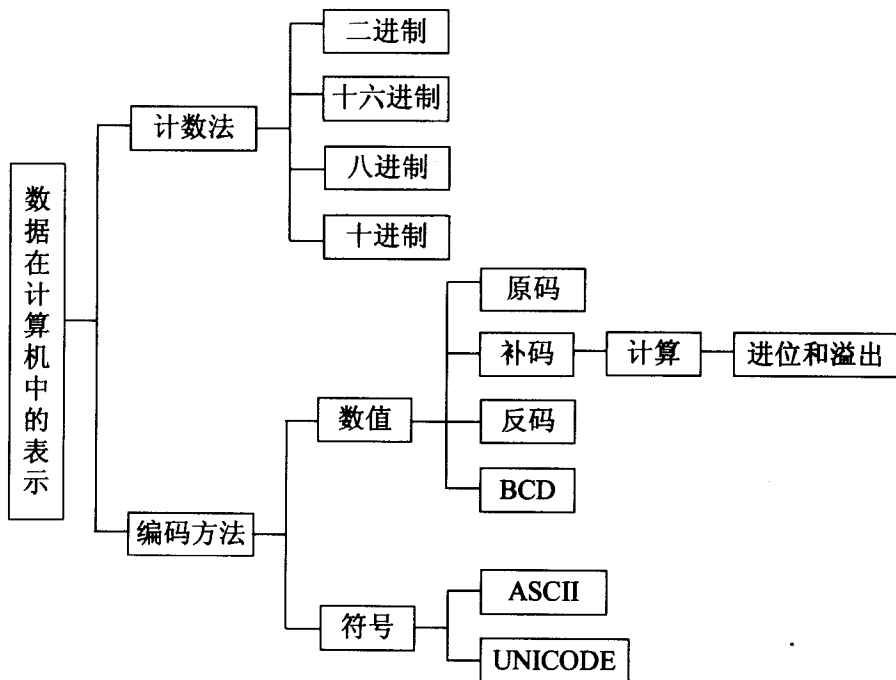
④ 没有合适的高级语言或只能采用汇编语言的时候。例如,开发最新的处理器程序时,暂时没有支持新指令的编译程序;

汇编语言还有许多实际应用。例如分析具体系统尤其是该系统的低层软件,加密解密软件,分析和防治计算机病毒,等等。

本书以 Windows 操作系统和 IA-32 处理器为背景,按照微软的宏汇编 MASM 的格式全面介绍汇编语言程序设计的各种技术。

# 第 1 章

## 计算机中数据的表示



计算机最基本的功能是进行数据的计算和加工处理。数据在计算机中是以器件的物理状态来表示的。在计算机中采用二进制数表示数据，即计算机中要处理的所有数据，都采用二进制表示，所有的字母、符号也都要用二进制编码来表示。在程序中，为了表示方便，经常使用十六进制数书写。

## 1.1 数 制

数制是以表示数值所用的数字符号的个数来命名的，如十进制、二进制、十六进制、八进制等。各种数制中数字符号的个数称为该数制的基数。一个数值可以用不同数制表示它的大小，不同数制的表示形式虽然不同，但数值的量是相等的。在日常生活中，最常用的是十进制数。但是计算机只能识别 0/1 二进制代码，所以进入计算机的信息都要转换成 0/1 代码后才能进行处理。本节说明如何利用各种不同的数制表示现实中的数值以及不同数制之间如何相互转换。

### 1.1.1 十进制

十进制采用 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 这十个数字和一个小数点符号来表示任意的十进制数。其特点是：逢十进一，基数为 10。每个数位具有不同的权值，可以表示为  $10^i$  ( $i$  表示第  $i$  个数位)，如个位、十位及百位的权分别为 1、10 及  $10^2$ ，十分位、百分位的权分别为  $10^{-1}$  和  $10^{-2}$ 。每个数位上的数字所表示的量是这个数字和该数位权值的乘积。因此，任意十进制数可按权展开为 10 的  $i$  次幂多项式。例如，374.53 的多项式表示形式为：

$$374.53 = 3 \times 10^2 + 7 \times 10^1 + 4 \times 10^0 + 5 \times 10^{-1} + 3 \times 10^{-2}$$

对于  $n$  位整数  $m$  位小数的任意十进制数  $N_{10} = K_{n-1}K_{n-2} \cdots K_0.K_{-1}K_{-2} \cdots K_{-m}$ ，可用多项式表示

$$\text{为：} \quad N_{10} = \sum_{i=-m}^{n-1} k_i 10^i \quad (1-1)$$

其中  $i$  表示数的某一位； $k_i$  表示第  $i$  位的数字，它可以是 0~9 中的任一数字； $m$  和  $n$  为正整数；式中 10 为十进制的基数。

### 1.1.2 二进制

在电子计算机中，数是以器件的物理状态来表示的，计算机中采用双稳态电子器件作为保存信息的基本元件，因此在计算机中采用二进制数来表示数据。在二进制中，只有 0 和 1 两个数字，它的基数为 2，每个数位上的权是  $2^i$ 。

对于  $n$  位整数  $m$  位小数的任意二进制数  $N_2 = K_{n-1}K_{n-2} \cdots K_0.K_{-1}K_{-2} \cdots K_{-m}$ ，可以用多项式表示

$$\text{为：} \quad N_2 = \sum_{i=-m}^{n-1} k_i 2^i \quad (1-2)$$

其中  $k_i$  为 0 或 1。



### 1.1.3 十六进制

计算机中采用二进制表示数据的优点是物理实现容易且运算规则特别简单,缺点是书写冗长。因此常用十六进制(有时用八进制)代替二进制来书写数据。在十六进制中,有十六个数字符号,包括十进制数字0~9和英文字母A, B, C, D, E和F(也可使用相应的小写字母a~f),其中A, B, C, D, E, F分别与十进制中的10, 11, 12, 13, 14, 15这6个数相对应。十六进制数的基数为16,每一数位上的权是16的某次幂。

对于  $n$  位整数  $m$  位小数的任意八进制数  $N_{16}=K_{n-1}K_{n-2}\cdots K_0.K_{-1}K_{-2}\cdots K_{-m}$ , 可以用多项式表示为:

$$N_{16} = \sum_{i=-m}^{n-1} k_i 16^i \quad (1-3)$$

其中  $k_i$  为 0~9, A~F 中的一个数码。

### 1.1.4 八进制

计算机中有时也用八进制代替二进制来显示数据。在八进制中,有0~7八个数字符号,其基数为8,每一数位上的权是8的某次幂。

对于  $n$  位整数  $m$  位小数的任意八进制数  $N_8=K_{n-1}K_{n-2}\cdots K_0.K_{-1}K_{-2}\cdots K_{-m}$ , 可以用多项式表示为:

$$N_8 = \sum_{i=-m}^{n-1} k_i 8^i \quad (1-4)$$

其中  $k_i$  为 0~7 中的一个数码。

汇编语言中数据用字母 B 或 b 结尾表示该数据采用二进制(Binary)书写,用字母 H 或 h 结尾表示采用十六进制(Hexadecimal)书写,用字母 O 或 o 结尾表示采用八进制(Octal)书写。采用十进制(Decimal)书写数据可以用字母 D 或 d 结尾,也可以不加结尾字母。写一个十六进制数如果最高位数字为字母(A~F),则字母前面必须加一个0,以便与变量名相区别。

### 1.1.5 数制间的转换

#### 1. 非十进制数转换成十进制数

非十进制数转换成十进制数比较简单,可先将其按定义展开为多项式,再将系数及权值用十进制表示,进行乘法与加法运算,所得结果即为该数对应的十进制数。

**例 1-1:** 将二进制数 1101.101 转换为十进制数。

$$\begin{aligned} 1101.101B &= 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= 8 + 4 + 1 + 0.5 + 0.125 \\ &= 13.625 \end{aligned}$$