

21 世纪高等教育系列规划教材

汇编语言编程技术

主编 李国安 李敏

Huibian Yuyan Biancheng jishu



郑州大学出版社

21 世纪高等教育系列规划教材

汇编语言编程技术

主编 李国安 李敏

Huibian Yuyan Biancheng jishu



郑州大学出版社

图书在版编目(CIP)数据

汇编语言编程技术/李国安,李敏主编. —郑州:郑州
大学出版社,2007.4

ISBN 978 - 7 - 81106 - 486 - 5

I. 汇… II. ①李…②李… 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 045622 号

郑州大学出版社出版发行

郑州市大学路 40 号

出版人:邓世平

全国新华书店经销

郑州文华印务有限公司印制

开本:787 mm × 1 092 mm

印张:19.75

字数:471 千字

版次:2007 年 4 月第 1 版

邮政编码:450052

发行部电话:0371 - 66966070

1/16

印数:1 ~ 2 500

印次:2007 年 4 月第 1 次印刷

书号:ISBN 978 - 7 - 81106 - 486 - 5 定价:29.80 元

本书如有印装质量问题,请向本社调换

作者名单

主 编 李国安 李 敏
副主编 杨 茂 阎喜亮 赵 轲
编 委 (以姓氏笔画排序)
朱 斌 李 敏 李国安
李秀菊 李留强 杨 茂
杨 锋 赵 轲 降雪辉
阎喜亮

内 容 提 要

本书以 80X86 和奔腾机为背景,以 MS-MASM 5.0/6.0 为蓝本,在简要介绍计算机基本知识的基础上,详细地介绍了汇编语言的语法知识和编程原理、编程技术,有一定的篇幅用于上机实践,收集编制了不少常用程序。

本书按照 2004 年教育部《关于进一步加强高等学校计算机基础教学的意见》中关于大专院校规划教材的要求编写,叙述准确,内容丰富,例题适当,习题多样。

本书可作为普通高等教育本科院校(全 15 章)及高等职业技术学院、高等专科学校、成人高校(前 12 章)的教学教材,亦可作为计算机、电子、电信、数控等相关专业工程技术人员的参考书以及计算机资格、水平、等级考试用书。

序 言

在我国高等教育逐步实现大众化的今天,高等院校的教育模式也已随之逐渐发生适应性变化。高等院校不仅担负着精英教育、通才教育的时代责任,而且尤须适应如今大众化教育的社会现实,担负起对各类高级技能应用型人才多样化培养的社会责任。这就需要不断地深化教学改革。

教学改革的重要一环是教材改革。在知识更新不断加快的现今时代,计算机硬、软件知识的更新尤其明显。因之,不断替换并及时充实新知识,从而推出新教材,便是不容忽视、必须重视的大事。

基于此,在明确了培养对象、培养目标、培养模式、课程体系、教材内容的前提下,成立了本教材编委会,着手研究并编写本教材。

本教材编委会编委毕业于国内一些知名院校。他们中既有长期从事计算机科研和生产一线的高级工程师和程序设计师,又有长期从事高校计算机教学的教授和讲师。主编李国安先生,自1969年北京林业大学本科毕业,调干入伍总参谋部计算技术研究所,接受中国科学院自动化研究所和清华大学计算机培训后,一直从事计算机工作、科研和大学计算机教学工作近40年,至今仍在大学讲台执教并著书,并兼任河南省高校计算机教育研究会理事、郑州市计算机学会常务理事、副秘书长等学术职务。其他编委毕业院校依次为:李敏,清华大学硕士;杨锋,郑州大学硕士;阎喜亮,北京师范大学;赵轲,河南工业大学;朱斌,清华大学在读研究生;李秀菊,中央党校;杨茂,河南工业大学博士;降雪辉,河南财经学院。上述人员组成编委会,知识面广,实践经验丰富,并有相当丰富的计算机科研和教学经验,这对于编写好本教材是很有利的条件。

由于本书主编对汇编语言有30多年的学习、编程、工作实践及教学实践,在整理多年来的读书笔记、编程经验和多本教案的基础上,又参阅了一些同类书籍而后编写本书,因此,本书最显明的特点是:使得原本枯燥艰涩相当难学的本课程,变得比较通俗易懂。

希望本教材在教学实践中接受检验,积累经验,不断完善,能成为高等院校相关专业的精品教材。

萧 曦

2006年12月9日
于清华大学

前 言

当前的世界,是以计算机技术为基础、为前提、为主导、为中心的科技飞速发展的世界,是知识更新日新月异的世纪。以计算机技术为主要因素使世界变成了一个地球村,使世界进入信息时代、读图时代、读数时代。计算机知识对于当今的人们变得愈来愈重要,缺乏计算机知识就意味着“现代文盲”这个理念逐渐为人们所共识,于是现今全国各类学校,纷纷开设计算机课程,甚至设置计算机专业、计算机系、信息工程学院等。

汇编语言是能够利用计算机硬件特性并能直接控制计算机硬件的语言,既面向硬件又面向问题的唯一编程语言,具有不可替代性。汇编语言编程技术是计算机、电子、电信、数控等专业必修的一门核心专业课程。学好本课程,对于普及和深入理解计算机组成原理、计算机工作原理及编程原理等,培养学生和编程者的程序设计能力,从事软件开发应用和硬件组装维护维修以及计算机研究能力等,都具有特别重要的作用。

本课程是计算机组成原理、计算机系统结构原理、计算机工作原理、接口技术、操作系统、计算机控制原理、编程原理、计算机组装与维护、单片机、单板机等诸多课程的先导课程。这些课程组成了以汇编语言为表述特征的计算机、电子、电信、数控诸专业课程中的系列主课程。这个特征决定着:汇编语言与其他课程相比有些难学。

本书编写着眼于理论联系实践,理论以必备、够用为原则,实践以例题、习题、上机为方式,以应用为目的。本课程注重要求:①系统了解基本知识;②能够通晓基本原理;③熟练掌握基本编程技能。汇编语言自身有一个系列,8086 汇编语言是最基础的,掌握了 8086 汇编语言,则可较容易地深入学习 80X86 和奔腾系列机汇编语言编程技术。

本课程讲授建议不少于 80~85 学时,其中理论课 50~55 学时,上机实践 30 学时。另可安排 15 个学时编程实习。大专课程重点是前 12 章,其他章节以学生自学为主。

本书由郑州华信学院李国安编写第 1~5 章;清华大学李敏编写第 6~7 章;郑州华信学院阎喜亮编写第 8、14 章,赵轲编写第 9 章,杨茂编写第 11 章;杨锋编写第 10、12 章;清华大学硕士研究生朱斌编写第 13 章;郑州水文水资源勘测局李秀菊编写第 15 章;降雪辉编写附录。

本书由李国安副教授统稿。杨作平、李丽参加初审稿,杨锋副教授统审稿。赵轲讲师负责电子教案制作。感谢所有参考文献的作者及出版社诸位编辑的支持和帮助。

尽管上述人员在本书编写、审阅过程中付出了艰苦劳动,但疏漏和问题仍难绝迹,诚恳期待各位专家、学者和工程技术人员、读者等提出宝贵意见,以不断完善本书。

感谢清华大学教授萧曦博士为本书作序。

李国安

2006 年 12 月 6 日于黄帝故里无梦斋

目 录

第1章 汇编语言基础知识	1
1.1 计算机语言	1
1.2 进位制数及相互转换	5
1.3 数据编码	9
习 题	12
第2章 8088/8086 中央处理器	13
2.1 8088/8086 中央处理器简介	13
2.2 控制器	13
2.3 运算器	15
2.4 存储器段节式管理	19
2.5 存储器堆栈技术	22
习 题	23
第3章 汇编语言寻址方式	24
3.1 汇编语言指令格式	24
3.2 汇编语言寻址方式	25
习 题	30
第4章 8088/8086 指令系统	31
4.1 程序开发条件	31
4.2 程序编写方法	32
4.3 程序编写遵循规则	33
4.4 8088/8086 指令系列	33
习 题	38
第5章 汇编语言指令语句	40
5.1 汇编语言字符集	40
5.2 数值表达式	41
5.3 汇编语言指令语句一般格式	49
5.4 伪指令语句	52
5.5 汇编语言程序设计举例及上机操作	61
习 题	65

第6章 顺序结构编程技术	67
6.1 算术运算类指令	67
6.2 逻辑运算类指令	76
6.3 移位操作类指令	78
6.4 顺序结构编程技术	80
习 题	83
第7章 分支结构编程技术	86
7.1 分支程序结构	86
7.2 无条件转移类指令	87
7.3 条件转移类指令	88
7.4 分支程序设计方法	92
习 题	99
第8章 循环结构编程技术	105
8.1 循环程序基本结构	105
8.2 循环控制指令	106
8.3 字符串操作指令与重复前缀指令	109
8.4 循环程序控制方法	115
8.5 多重循环程序设计	120
习 题	122
第9章 子程序结构编程技术	131
9.1 子程序调用与返回指令	131
9.2 子程序一般结构	135
9.3 参数传递方法	138
9.4 子程序嵌套与递归	145
9.5 子程序编程技术及实用程序	150
习 题	162
第10章 输入/输出及系统功能调用	164
10.1 输入/输出概述	164
10.2 输入/输出指令	166
10.3 输入/输出传送方式	168
10.4 输入/输出系统功能调用及实用程序	170
习 题	177
第11章 中断技术	180
11.1 中断概念	180
11.2 中断编程技术	185
11.3 常用系统中断	187
11.4 磁盘文件管理	197

11.5 编程技术实例·····	207
习 题·····	215
第 12 章 汇编语言高级编程技术 ·····	217
12.1 宏汇编技术·····	217
12.2 重复汇编技术·····	225
12.3 条件汇编技术·····	228
习 题·····	230
第 13 章 混合编程技术 ·····	231
13.1 混合编程简介·····	231
13.2 混合编程调用协议·····	231
习 题·····	240
第 14 章 汇编语言模块编程技术 ·····	241
14.1 模块化技术简介·····	241
14.2 模块之间的通信接口·····	242
习 题·····	249
第 15 章 汇编语言实用编程技术 ·····	250
15.1 图形处理程序技术·····	250
15.2 声音处理程序技术·····	253
15.3 设备驱动程序技术·····	257
15.4 通信处理程序技术·····	270
习 题·····	280
附录 1 DEBUG 主要命令 ·····	281
附录 2 汇编程序错误提示信息 ·····	286
附录 3 8086 汇编语言指令表 ·····	292
参考文献·····	304

第1章 汇编语言基础知识

1.1 计算机语言

1.1.1 程序设计语言

打开电子计算机,让它工作,首先就需要给它下达命令。计算机按照操作者下达的命令进行操作,其操作结果应该让操作者能够看懂并且能够接受。这样就形成了人机对话。这就需要为能够实现人机对话而设计一套语法规则。这一套语法规则就构成了计算机语言。由于计算机语言是为了用来编写程序,因此又可以称之为程序设计语言。

1.1.2 机器语言

专门为计算机硬件设计、能让计算机硬件直接识别的计算机语言,称之为机器语言,这是最简单也是最基本的程序设计语言,人们称之为第1代计算机语言。

全世界的计算机机型五花八门,因此,每一种计算机机型就必然具备自己独特的机器语言。换言之,机器语言不具有通用性。

机器语言直接面向某种机型的机器硬件编程。换言之,针对某种机型的计算机,用配给该机的机器语言所编制的程序,才能直接被该机型机器所识别并执行,占用存储空间少、运行速度快、运行效率高。那么,机器语言是怎样表达给计算机的呢?举例说明。

例 1.1 将二进制数 00000010 送累加器,再加上 00000010,用 8086 机器指令实现。
可表示为 1011 0000 0000 0010

0000 0100 0000 0010

机器语言的缺点由此可见一斑。这些由一串串 0 和 1 编织起来的一条条指令,以及由这种指令组成的程序罗列开来,书写费事费时,识别记忆困难,编制极易出错,调试修改则更是麻烦。自己编制的程序,往往自己也难以辨认,而相互交流就等于传看天书。

因此,用机器语言编写的机器语言源程序,除厂家和核心系统软件专家、商家用于编写计算机的最底层的核心系统程序之外,作为一般程序员是很难直接使用的。因此,对于一般程序员来说,机器语言不是大众化的程序设计语言,是专家级程序设计语言。人们不

得不进一步研究新的程序设计语言。

1.1.3 汇编语言

由于机器语言的诸多缺点,人们只好另辟蹊径,采用人们容易识别和理解的一些符号来代替机器语言的二进制代码,而这一些符号是采用该条指令的英语单词缩写符,用来表示该英语词意的操作功能码,能帮助人们易于识别和记忆,于是,就称之为助记符。以助记符为主要特征而形成的新式计算机语言,称之为助记符语言。

很明显,用助记符语言编写的源程序不能直接为计算机所识别,必须事先编制好一个能识别并转换这种源程序的专用机器语言程序,它能够将助记符语言源程序,转换成机器能够识别的机器语言程序,这个专用机器语言程序称之为汇编程序。能够被汇编程序进行汇编的助记符语言源程序,称之为汇编语言源程序。这就是第2代计算机语言。

汇编语言不再使用二进制字码编程,而是用英语单词的缩写词代替指令功能码和机器部件名,用十六进制数代替二进制数。举例说明如下。

例 1.2 将数值 2 送入累加器,并将累加器内容加 2,用汇编语言实现。

可表示为 `MOV AX,2` ;将十六进制数 2 送入累加器 AX

`ADD AX,2` ;AX 内容加上十六进制数 2,运算结果存入 AX

`MOV` 是 `MOVE`(传送)的简写,`ADD` 是 `ADDITION`(加)的简写,AX 代表累加器,2 是二进制数 `00000010B` 的十六进制数表达形式。

很明显,用汇编语言编写出来的程序既保留了机器语言的优点,又易于书写、易于记忆、易于调试、易于修改、易于交流,因而也就易于大众化、社会化、商品化。

一条用汇编语言编写的指令,对应于一条乃至多条用机器语言编写的指令,因而也就大大简化了程序总篇幅。由于汇编语言能够直接利用计算机的硬件系统特性,能够直接利用 CPU 的机器语言指令系统和各种寻址方式,能够将计算机的功能全面地提供给程序设计者。

1.1.4 汇编程序

程序员用汇编语言编写的一条条指令组成的程序,称之为汇编语言源程序。汇编语言源程序包含伪指令、宏指令、执行指令和注释等。

汇编程序对汇编语言源程序进行编译时,首先对伪指令、宏指令和注释部分进行处理,从而使得汇编语言源程序支离破碎,这就需要对编译过的源程序进行再次的汇集编排,重新分配内存地址,生成相应的目标代码程序,这也正是汇编程序名称的起因。

汇编程序对汇编语言源程序进行译、汇、编,生成相应目标代码程序的过程,称为汇编。

经过汇编而产生的目标代码程序,还须经过连接程序将其连接装配成可执行的代码程序才能运行(图 1.1)。

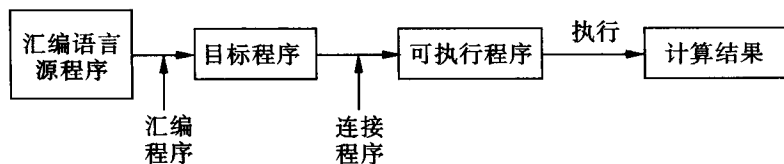


图 1.1 汇编语言源程序汇编示意图

汇编语言常常应用于系统软件开发、扩充,修改系统设备,对现场进行检测、实时监控、通信,或者用于对运行速度、存储空间、设备的体积有着特别要求的实用程序设计。

由于用汇编语言编写源程序,要求程序员必须面对计算机器件一级编程,因此编程者须对所使用的计算机的组成原理、工作原理、系统结构、器件性能等应有深刻了解。换言之,用汇编语言编程,要涉及计算机底层硬件,应该学习和掌握的内容多而且枯燥,须记忆的知识琐碎而且繁杂,而自学者的困难就会更大些。很明显,汇编语言是专业级语言。

在现实中,既懂硬件又能用汇编语言编程的程序员是不多的。如果设计一种语言,编程时,涉及面对硬件编程时,才通过混编而调用汇编语言编写的各种功能的子程序,岂不更好。于是,人们仍有必要继续探索新的计算机语言。

1.1.5 算法语言

电子计算机的主要功能,是代替人类进行繁杂的计算、庞大的数据处理和各种信息的传输、存储等,这样,许多种类的数学公式式语言——第3代计算机语言——也就应运而生。人们设计出用单词的缩写和数学公式及数据等,按照一定规则,编写成一条条语句。这种以数学公式为主要特征而编写的算法程序的语言,称之为**算法语言**。举例说明如下。

例 1.3 将数值2赋值给变量AX,并将变量AX内容加2,用BASIC语言实现。

10 AX = 2 ;将2赋值给AX (AX在此只能作变量)

20 AX = AX + 2 ;将AX + 2运算后,再存放到AX

面对只进行数据计算和数据处理这样的问题,用算法语言编写程序,比用汇编语言编写程序更易于书写、易于记忆、易于调试、易于修改、易于交流,所编写的程序的大众化、社会化、商品化的效果也就更好。

很明显,作为非计算机专业的一般编程者,他无须对计算机原理、系统结构、器件性能等等作深刻地了解,从而使得算法语言更接近于大众,因此人们认为这些算法语言**高级**,于是俗称其为**高级语言**,就像俗称计算机为**电脑**一样。算法语言成了大众级语言。

但是,算法语言或者叫做高级语言既不可能替代机器语言,更不可能替代汇编语言。因为,算法语言编写出来的是**语句**,主要进行数据处理;而机器语言和汇编语言编写出来的是**指令**,只有指令才能利用计算机的所有硬件特性并且能够直接控制底层硬件运行。

由本节内容可知:机器语言是面向计算机底层硬件的语言,算法语言是面向问题的程序设计语言,而汇编语言是既须面向计算机底层硬件、又能面向问题的程序设计语言。汇编语言是一门十分重要的程序设计语言,汇编程序是计算机系统软件中的核心软件。

算法语言种类繁多,从较早的 FORTRAN 语言说起,仅经常使用的就有数百种,而且还在不断地设计出新的算法语言以及同系列新的版本等,最常见的就有数十种,计有:ALGOL、COBOL、BASIC、C、PASCAL、LISP、LOGO、Java,等等。

人们在不断地设计出新的算法语言之同时,原有的程序设计语言本身又在自己的系列方面推演和变化着。比如,从 FORTRAN 语言中简化出来为大众易于接受从而普及得很快很广的 BASIC 语言之后,又自成系列,陆续产生出 BASICA、GBASIC、QB(Quick BASIC)、VB(Visual BASIC)等语言;继 PASCAL 语言之后,又出现了 Delphi 语言;继 C 语言之后,又陆续产生出 C++、Visual C++ 等语言(图 1.2)。

Java 语言是 Sun MicroSystem 公司于 1995 年 5 月推出的一种面向对象的程序设计语言,它脱胎于 C++ 语言,对 C++ 语言作了大量简化、修改、补充,使其具有面向对象、简单、安全、与平台无关、多线程等特性,是继承现有的种种编程语言的优秀成果而发展起来的新时代程序设计语言。随着 Internet 技术迅速普及、推广和发展,World Wide Web(万维网)迅速流行,Java 在编制 Applet(小应用程序)方面将越来越受到 Internet 用户欢迎。

Visual BASIC、Visual C++ 以及 Delphi 等语言充分体现面向对象的技术优势,这些极为简便的程序设计语言,代表着程序设算法语言的发展趋势(图 1.3)。

人们将面向对象的程序设计语言称为第 4 代计算机语言。

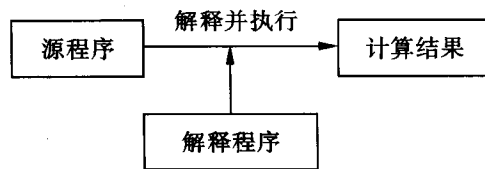


图 1.2 解释类语言程序示意图

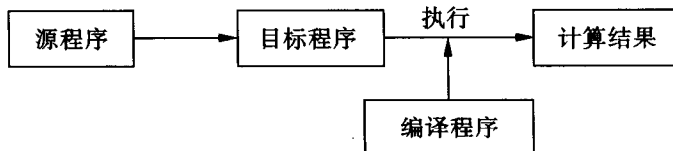


图 1.3 编译类语言程序示意图

1.1.6 智能语言

从 20 世纪 80 年代起,法、英、美、日等国的计算机专家开始研制第 5 代计算机,并一致取名为智能机。为智能机配置的核心语言称为 PROLOG,即程序逻辑语言,亦称智能机语言、第 5 代计算机语言。它是以一阶谓词演算为背景的交互式语言,相当于传统计算机中的汇编语言,是汇编语言的高能版本。

智能机语言不仅能面向机器、面向问题、面向过程、面向对象,而且能够面向用户。

所谓面向用户,既是说,任何用户,只须向计算机输入要求,计算机自动编程器会运用智能机语言自动编写出用户需求的应用程序,从而运行程序,取得结果(图 1.4)。

很明显,智能机语言将使智能计算机变成傻瓜计算机。

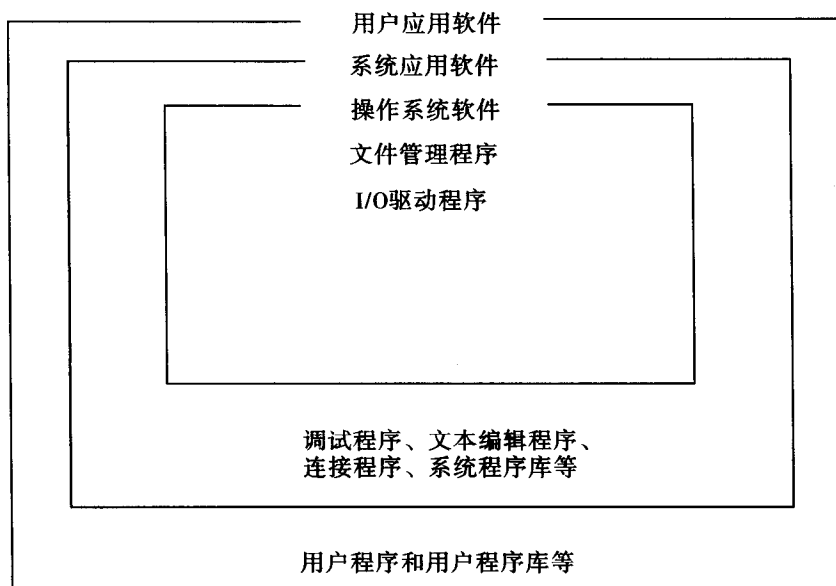


图 1.4 计算机软件层次示意图

1.2 进位制数及相互转换

电子计算机是现代数据信息处理工具,数据表示问题是计算机工作的焦点之一。

我们在日常活动中,习惯使用十进制 (Decimal) 数据,而计算机硬件的设计和构造决定了只能直接识别二进制 (Binary) 数据,而在汇编语言程序设计中,使用十六进制 (Hexadecimal) 数据则尤其显得方便。

1.2.1 十进制数、二进制数

十进制使用的数码有十个,即 0、1、2、3、4、5、6、7、8、9,其基数为 10,采用逢十进一计数法。

例 1.4 按数位的权位展示一个十进制数如 1024。

$$1024 = 1 * 10^3 + 0 * 10^2 + 2 * 10^1 + 4 * 10^0 = 1000 + 0 + 20 + 4$$

计算证明,对于任意一个十进制正数,均可表示为

$$N = K_n * 10^n + K_{n-1} * 10^{n-1} + \dots + K_0 * 10^0 + K_{-1} * 10^{-1} + \dots + K_{-m} * 10^{-m}$$

(表达式 1.1)

其中, N 为任一个十进制正数, $K_n, K_{n-1}, \dots, K_{-m}$ 为系数, 10 为基数, $10^n, 10^{n-1}, \dots, 10^{-m}$ 为权值。

二进制使用的数码有两个, 即 0, 1, 其基数为 2, 采用逢二进一计数法(表 1.1)。

表 1.1 二进制数位权值与十进制数值对照表

二进制位	8	7	6	5	4	3	2	1	0
权	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
十进制数值	256	128	64	32	16	8	4	2	1

例 1.5 按数位的权值, 展开一个二进制数如 1101, 并换算成十进制数。

$$\begin{aligned}
 1101\text{ B} &= 1 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0 \\
 &= 8 + 4 + 0 + 1 \\
 &= 13\text{ D}
 \end{aligned}$$

计算证明, 对于任意一个二进制正数, 均可表示为

$$N = K_n * 2^n + K_{n-1} * 2^{n-1} + \dots + K_0 * 2^0 + K_{-1} * 2^{-1} + \dots + K_{-m} * 2^{-m} \quad (\text{表达式 1.2})$$

其中, N 为任一个二进制正数, $K_n, K_{n-1}, \dots, K_{-m}$ 为系数, 2 为基数, $2^n, 2^{n-1}, \dots, 2^{-m}$ 为权值。

对比十进制和二进制正数两种表示, 除了基数 10 和 2 外, 其余相同。于是, 设 R 为任一个十进制数或二进制数或十六进制数的基数, 则可推理出

$$N = K_n * R^n + K_{n-1} * R^{n-1} + \dots + K_0 * R^0 + K_{-1} * R^{-1} + \dots + K_{-m} * R^{-m} \quad (\text{表达式 1.3})$$

当遇到有多种计数制的数据在一起的时候, 应该在十进制数的尾部添加 D 或括号 (.....) 加下标 10, 在二进制数的尾部加 B 或括号 (.....) 加下标 2, 在十六进制数的尾部加 H 或括号 (.....) 加下标 16。例如, 例 1.5 中的二进制数又可写为 $(1101)_2$ 。由于人们对十进制数最常使用, 因而对十进制数的标识符 D 可以省略。

1.2.2 十六进制数

二进制数可以被计算机直接识别, 方便了存储, 其抗干扰性也相当强。但是, 程序员及操作员阅读、书写、分析、输入等却很不方便, 于是在编程中采用十六进制数。

十六进制数使用的数码有 16 个, 即 0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F, 其中 A ~ F 对应于十进制数的 10 ~ 15, 基数为 16, 采用逢十六进一计数法。若十六进制数的首位为字母字符时, 则应在该首位字母字符的前面添加一个数码 0, 以便明确该数据为数值数据, 而非字符数据。但该 0 不占用存储器单元位置。

在用汇编语言编写的程序中以及在调试系统中的数, 常使用十六进制数, 汇编列表文件中代码指令, 以及内存地址单元号一般也用十六进制数来表达。

将十进制数转换为十六进制数的规则,和十进制数转换为二进制数相类似,其区别仅仅在于:其整数部位应该依次除以16,其小数部位应该依次乘以16(表1.2)。

表1.2 十六进制数与十进制数值对照表

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

1.2.3 进位制数之间的相互转换

由于表达式 -1 、 -2 、 -3 成立,那么二~十六进制数就可以按照此表达式规则转换成十进制数了。

例1.6 将二进制数1011.011转换成十进制数。

$$1011.011 \text{ B} = 1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 1 * 2^0 + 0 * 2^{-1} + 1 * 2^{-2} + 1 * 2^{-3} \\ = 11.375 \text{ D}$$

例1.7 将十六进制数212.A转换成十进制数。

$$212.A \text{ H} = 2 * 16^2 + 1 * 16^1 + 2 * 16^0 + 10 * 16^{-1} \\ = 530.625 \text{ D}$$

将十进制数转换为二进制数的规则是:十进制正数的整数部分依次除以2倒取余,小数部分依次乘以2正取整。

例1.8 将十进制数46.61125转换成二进制数,小数精确到6位。

商	余数	小数运算结果	小数的取整
$46/2 = 23$	0	$0.61125 * 2 = 1.2225$	1
$23/2 = 11$	1	$0.2225 * 2 = 0.445$	0
$11/2 = 5$	1	$0.445 * 2 = 0.89$	0
$5/2 = 2$	1	$0.89 * 2 = 1.78$	1
$2/2 = 1$	0	$0.78 * 2 = 1.56$	1
$1/2 = 0$	1	$0.56 * 2 = 1.02$	1

倒取整数,正取余数,于是 $46.61125 \text{ D} = 101110.100111 \text{ B}$

例1.9 将十进制数6091.625转换成十六进制数,小数精确到5位。

商	余数	小数运算结果	小数的取整
$6091/16 = 380$	11(B)	$0.625 * 16 = 10$	A
$380/16 = 23$	12(C)		
$23/16 = 1$	7		
$1/16 = 0$	1	于是 $6091.625 \text{ D} = 17\text{CB}. \text{AH}$	

例1.10 将十进制数67.48分别转换成二进制数和十六进制数,小数精确到5位。