

21 世纪高等学校计算机教育实用规划教材

汇编语言程序设计教程与实验

徐爱芸
向华
编著



清华大学出版社

TP313

97

2007

21 世纪高等学校计算机教育实用规划教材

汇编语言程序设计教程与实验

徐爱芸 向 华 编著

清华大学出版社
北 京

内 容 简 介

本书以目前广泛使用的 PC 为平台,详细介绍了宏汇编语言的基本概念、程序设计的基本方法与技巧。全书共 10 章,主要内容包括宏汇编语言、程序设计的基本技巧、输入/输出程序设计、32 位 CPU 扩展功能、汇编语言上机内容与要求。

本书内容详实,叙述细腻易懂,在章节安排上由浅入深,指令介绍与指令应用紧密结合,使得知识易于理解和掌握。

本书可作为高等院校计算机及相关专业“汇编语言程序设计”课程教材,也可作为独立学院、高职高专计算机及相关专业、有关工程技术人员的教学参考书。

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

汇编语言程序设计教程与实验/徐爱芸,向华编著.——北京:清华大学出版社,2007.2
(21 世纪高等学校计算机教育实用规划教材)

ISBN 978-7-302-14341-3

I. 汇… II. ①徐… ②向… 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(2006)第 154720 号

责任编辑:魏江江

责任校对:时翠兰

责任印制:王秀菊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn> 邮 编:100084

c-service@tup.tsinghua.edu.cn

社 总 机:010-62770175 邮购热线:010-62786544

投稿咨询:010-62772015 客户服务:010-62776969

印 刷 者:北京市清华园胶印厂

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185×260 印 张:16 字 数:370 千字

版 次:2007 年 2 月第 1 版 印 次:2007 年 2 月第 1 次印刷

印 数:1~4000

定 价:22.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:022570-01

读者意见反馈

亲爱的读者：

感谢您一直以来对清华版计算机教材的支持和爱护。为了今后为您提供更优秀的教材，请您抽出宝贵的时间来填写下面的意见反馈表，以便我们更好地对本教材做进一步改进。同时如果您在使用本教材的过程中遇到了什么问题，或者有什么好的建议，也请您来信告诉我们。

地址：北京市海淀区双清路学研大厦 A 座 602 室 计算机与信息分社营销室 收
邮编：100084 电子信箱：jsjjc@tup.tsinghua.edu.cn
电话：010-62770175-4608/4409 邮购电话：010-62786544

教材名称：汇编语言程序设计教程与实验

ISBN：978-7-302-14341-3

个人资料

姓名：_____ 年龄：_____ 所在院校/专业：_____

文化程度：_____ 通信地址：_____

联系电话：_____ 电子信箱：_____

您使用本书是作为：☐指定教材 ☐选用教材 ☐辅导教材 ☐自学教材

您对本书封面设计的满意度：

☐很满意 ☐满意 ☐一般 ☐不满意 改进建议_____

您对本书印刷质量的满意度：

☐很满意 ☐满意 ☐一般 ☐不满意 改进建议_____

您对本书的总体满意度：

从语言质量角度看 ☐很满意 ☐满意 ☐一般 ☐不满意

从科技含量角度看 ☐很满意 ☐满意 ☐一般 ☐不满意

本书最令您满意的是：

☐指导明确 ☐内容充实 ☐讲解详尽 ☐实例丰富

您认为本书在哪些地方应进行修改？（可附页）

您希望本书在哪些方面进行改进？（可附页）

电子教案支持

敬爱的教师：

为了配合本课程的教学需要，本教材配有配套的电子教案(素材)，有需求的教师可以与我们的联系，我们将向使用本教材进行教学的教师免费赠送电子教案(素材)，希望有助于教学活动的开展。相关信息请拨打电话 010-62776969 或发送电子邮件至 jsjjc@tup.tsinghua.edu.cn 咨询，也可以到清华大学出版社主页(<http://www.tup.com.cn> 或 <http://www.tup.tsinghua.edu.cn>)上查询。

21 世纪高等学校计算机教育实用规划教材

已出版教材

计算机操作系统(颜彬 等编著)
汇编语言程序设计教程与实验(徐爱芸 等编著)
实用软件工程(陆惠恩 等编著)
数据库技术与应用——SQL Server(刘卫国 编著)
数据库技术与应用——SQL Server(金聪 等编著)
XML 实用技术教程(顾兵 等编著)
网络技术应用教程(梁维娜 等编著)
Java 程序设计实践教程(张思民 等编著)
AutoCAD 实用教程(张强华 等编著)
大学计算机网络公共基础教程(徐祥征 等编著)
计算机网络实用教程——技术基础与实践(刘四清 等编著)
多媒体技术教程——案例 训练与课程设计(胡伏湘 等编著)
32 位微型计算机原理·接口技术及其应用(第 2 版)(史新福 等编著)

即将出版教材

Java 语言程序设计实用教程(陆迟 等编著)
数字通信原理(颜彬 等编著)
VC++ 程序设计教程(马石安 等编著)
面向对象程序设计教程(马石安 编著)
多媒体素材处理(向华 编著)
计算机与通信硬件技术基础(郭荣宁 等编著)
计算机网络与多媒体技术(胡虚怀 等编著)

更详细的教材介绍请登录清华大学出版社网站 <http://www.tup.com.cn> 查询

联系人:魏江江 E-mail: weijj@tup.tsinghua.edu.cn 电话: 010-62770175-4604

出版说明

随着我国高等教育规模的扩大和产业结构调整的进一步完善,社会对高层次应用型人才的需求将更加迫切。各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,合理调整和配置教育资源,在改革和改造传统学科专业的基础上,加强工程型和应用型学科专业建设,积极设置主要面向地方支柱产业、高新技术产业、服务业的工程型和应用型学科专业,积极为地方经济建设输送各类应用型人才。各高校加大了使用信息科学等现代科学技术提升、改造传统学科专业的力度,从而实现传统学科专业向工程型和应用型学科专业的发展与转变。在发挥传统学科专业师资力量强、办学经验丰富、教学资源充裕等优势的同时,不断更新其教学内容、改革课程体系,使工程型和应用型学科专业教育与经济建设相适应。计算机课程教学在从传统学科向工程型和应用型学科转变中起着至关重要的作用,工程型和应用型学科专业中的计算机课程设置、内容体系和教学手段及方法等也具有不同于传统学科的鲜明特点。

为了配合高校工程型和应用型学科专业的建设和发展,急需出版一批内容新、体系新、方法新、手段新的高水平计算机课程教材。目前,工程型和应用型学科专业计算机课程教材的建设工作仍滞后于教学改革的实践,如现有的计算机教材中有不少内容陈旧(依然用传统专业计算机教材代替工程型和应用型学科专业教材),重理论、轻实践,不能满足新的教学计划、课程设置的需要;一些课程的教材可供选择的品种太少;一些基础课的教材虽然品种较多,但低水平重复严重;有些教材内容庞杂,书越编越厚;专业课教材、教学辅助教材及教学参考书短缺,等等,都不利于学生能力的提高和素质的培养。为此,在教育部相关教学指导委员会专家的指导和建议下,清华大学出版社组织出版本系列教材,以满足工程型和应用型学科专业计算机课程教学的需要。本系列教材在规划过程中体现了如下一些基本原则和特点:

(1) 面向工程型与应用型学科专业,强调计算机在各专业中的应用。教材内容坚持基本理论适度,反映基本理论和原理的综合应用,强调实践和应用环节。

(2) 反映教学需要,促进教学发展。教材规划以新的工程型和应用型专业目录为依据。教材要适应多样化的教学需要,正确把握教学内容和课程体系的改革方向,在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养,为学生知识、能力、素质协调发展创造条件。

(3) 实施精品战略,突出重点,保证质量。规划教材建设仍然把重点放在公共基础课和专业基础课的教材建设上;特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版,逐步形成精品教材;提倡并鼓励编写体现工程型和应用型专业教学内容和课程体系改革成果的教材。

(4) 主张一纲多本,合理配套。基础课和专业基础课教材要配套,同一门课程可以有多种具有不同内容特点的教材。处理好教材统一性与多样化,基本教材与辅助教材、教学参考书,文字教材与软件教材的关系,实现教材系列资源配套。

(5) 依靠专家,择优选用。在制订教材规划时要依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时,要引入竞争机制,通过申报、评审确定主编。书稿完成后要认真实行审稿程序,确保出书质量。

繁荣教材出版事业,提高教材质量的关键是教师。建立一支高水平的以老带新的教材编写队伍才能保证教材的编写质量和建设力度,希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21 世纪高等学校计算机教育实用规划教材编委会

联系人:丁岭 dingl@tup.tsinghua.edu.cn

前 言

“汇编语言程序设计”是高等院校计算机硬、软件及应用专业学生都必须学习的核心课程之一,它是计算机组成原理、操作系统及其他核心课程的基础课,也是微机原理、单片机应用等课程的学习基础。同时,该课程对于训练学生掌握程序设计技术,熟悉上机操作和程序调试技术都有重要作用。

“汇编语言程序设计”的内容十分丰富,知识量大,是一门教师普遍感到难教、学生普遍感到难学的课程。为了满足教师教学和学生学习的需要,作者根据本课程的特点及知识结构,在多年教授这门课程的教学经验的基础上编写了此书,以期能帮助学生快速熟练地掌握相关知识,同时为教师的教学提供指导。

本书共 10 章,分为两大部分和两个附录,各部分内容如下:

第一部分:第 1~9 章。详细介绍了 16 位 8086 CPU 的宏汇编程序设计。第 1 章介绍了学习汇编语言程序设计所需的数据编码与运算等若干预备知识。第 2 章介绍了 8086 CPU 内部结构和存储器管理,围绕汇编语言的七种寻址方式,阐述了它们的格式、功能、使用方法及区别。第 3 章主要介绍了宏汇编语言中的各种表达式、伪指令语句及常用的系统功能调用。第 4 章全面介绍了 8086 指令系统中数据传送指令、算术运算指令、逻辑运算指令、处理器控制指令。第 5 章系统介绍了程序的三种基本结构(顺序结构、分支结构及循环结构)的设计方法与技巧,同时对实现这些结构的转移指令、循环指令也进行了详细介绍。第 6 章介绍了串操作指令及程序设计。第 7 章介绍了宏的定义与调用的方法、子程序的设计方法与技巧、模块化程序设计技术。第 8 章介绍了输入/输出数据传送方式、中断技术、软中断程序设计及 DOS 与 BIOS 中断调用。第 9 章以典型的 32 位 80386 CPU 为例,介绍了 32 位 CPU 的工作模式、内部寄存器、内存管理、寻址方式、新增指令等扩展功能。

第二部分:实验指导。第 10 章介绍了汇编语言上机实验,主要用于指导读者上机实验,提高编程能力,同时培养学生发现问题、解决问题的能力。

本书语言通俗易懂、讲叙细致,内容丰富,例题典型,结构安排合理,且注重理论知识与实际训练相结合,具有较强的适用性,对学生学好汇编语言的相关知识有较大的辅助作用。

本书可作为计算机及相关专业本、专科学生“汇编语言程序设计”课程的教材或参考资料,也可作为软件工程技术人员的学习参考书和研究生入学考试的应试辅导资料。

为方便教学,本书配有电子教案,任课教师如有需要,可与清华大学出版社联系,免费提供(E-mail: Weijj@tup.tsinghua.edu.cn)。

由于编写时间仓促,水平有限,虽然经过严格的审核、精细的编辑,本书在质量上有了-定的保障,但书中如有疏漏和不足之处,敬请各位读者批评指正。

编 者

目 录

第 1 章 汇编语言基础知识	1
1.1 计算机语言概述	1
1.1.1 机器语言	1
1.1.2 汇编语言	2
1.1.3 高级语言	2
1.1.4 汇编语言的特点	3
1.2 计算机中数据的表示方法	3
1.2.1 进位记数制	4
1.2.2 数制转换	4
1.2.3 计算机中数据的表示	6
习题	11
第 2 章 计算机内部结构与寻址方式	13
2.1 8086/8088 CPU 的内部结构	13
2.1.1 8086 CPU 的功能结构	13
2.1.2 8086 CPU 的寄存器结构	14
2.2 存储器组织	17
2.2.1 存储器的地址和内容	18
2.2.2 存储器的分段	19
2.3 堆栈	23
2.3.1 堆栈的特点	23
2.3.2 堆栈的作用	24
2.3.3 堆栈的操作	24
2.4 寻址方式	26
2.4.1 立即数寻址方式	26
2.4.2 寄存器寻址方式	27
2.4.3 存储器型寻址方式	27
2.4.4 外设型寻址方式	33
习题	34

第3章 汇编语言程序格式	36
3.1 汇编语言语句类型及格式	36
3.1.1 语句类型	36
3.1.2 指令语句格式	36
3.2 伪指令	37
3.2.1 段定义伪指令	38
3.2.2 数据定义	40
3.2.3 符号定义伪指令	44
3.2.4 与地址有关的伪指令	46
3.2.5 常用伪指令	47
3.3 表达式	51
3.3.1 常量	51
3.3.2 数值表达式	52
3.3.3 地址表达式	53
3.4 汇编语言源程序结构	53
3.5 常用 DOS 功能调用	54
习题	57
第4章 汇编语言指令系统	59
4.1 数据传送指令	59
4.2 算术运算指令	66
4.3 位操作指令	76
4.4 处理器控制指令	81
习题	83
第5章 汇编语言基本程序设计	86
5.1 汇编语言程序设计步骤	86
5.2 顺序结构程序设计	87
5.3 分支结构程序设计	90
5.3.1 转移地址的寻址方式	90
5.3.2 无条件转移指令	92
5.3.3 条件转移指令	93
5.3.4 分支结构程序设计	99
5.4 循环结构程序设计	106
5.4.1 循环控制结构	106
5.4.2 循环控制指令	107
5.4.3 循环程序设计方法	112

5.4.4 多重循环程序设计举例·····	116
习题·····	119
第6章 串操作程序设计 ·····	121
6.1 串操作概述·····	121
6.2 串操作指令及程序设计·····	123
习题·····	129
第7章 宏汇编与子程序设计 ·····	132
7.1 宏汇编·····	132
7.1.1 宏指令的使用·····	132
7.1.2 与宏有关的伪指令·····	135
7.1.3 宏库的建立与使用·····	137
7.2 子程序设计·····	139
7.2.1 子程序概述·····	139
7.2.2 子程序的调用与返回指令·····	140
7.2.3 子程序的定义·····	142
7.2.4 主程序与子程序的参数传递·····	147
7.2.5 子程序的嵌套·····	153
7.2.6 宏指令与子程序的比较·····	156
7.3 模块化程序设计·····	156
7.3.1 模块的装配连接·····	157
7.3.2 模块间的通信·····	161
7.3.3 模块化程序设计举例·····	163
习题·····	166
第8章 输入/输出程序设计 ·····	169
8.1 输入/输出概述·····	169
8.1.1 I/O端口的编址方式·····	170
8.1.2 数据传送方式·····	170
8.2 数据的输入/输出·····	172
8.2.1 I/O指令·····	172
8.2.2 程序查询I/O方式·····	172
8.2.3 中断I/O方式·····	173
8.3 键盘输入控制·····	182
8.4 屏幕输出控制·····	184
8.5 打印机输出控制·····	192
8.6 磁盘文件操作·····	197

8.6.1 文件概述.....	197
8.6.2 文件操作的系统功能调用.....	198
8.6.3 文件管理程序设计.....	199
习题.....	201
第9章 32位机新增指令及功能	203
9.1 80386的工作机制	203
9.1.1 80386 CPU的内部结构	203
9.1.2 80386 寄存器	203
9.1.3 80386 存储器管理	206
9.1.4 存储模型与段的简化定义.....	210
9.1.5 80386的寻址方式	212
9.1.6 80386 新增指令	212
9.2 80486 系统	216
9.2.1 80486 CPU 结构	217
9.2.2 80486 的内存管理和高速缓存	217
9.2.3 80486 扩充指令	218
9.3 Pentium 系统	219
9.3.1 Pentium CPU 结构	219
9.3.2 Pentium 扩充指令	220
9.4 程序举例	221
习题.....	223
第10章 实验指导	224
10.1 汇编语言上机过程及调试工具	224
10.2 分支程序设计	228
10.3 循环程序设计	229
10.4 子程序设计	229
10.5 中断程序设计	229
10.6 磁盘文件管理程序	231
10.7 模块化程序设计	233
10.8 综合程序设计	234
附录A ASCII 码表	236
附录B 调试工具 DEBUG	238
参考文献	244

计算机语言的发展经历了由机器语言、低级语言、高级语言到第四代语言的过程。汇编语言是一种面向机器的低级语言,它以助记符的形式表示每一条计算机指令,每条指令对应着计算机硬件的一个具体操作,因此,汇编语言与计算机有着密不可分的关系,是唯一能够充分利用计算机硬件特性并直接控制硬件设备的语言。利用汇编语言进行程序设计体现了计算机硬件和软件的结合。

1.1 计算机语言概述

1.1.1 机器语言

机器语言是计算机唯一能直接识别和执行的计算机语言。由于计算机硬件本身只能识别二进制代码,在计算机发展的初期,人们使用二进制代码来编写程序,使用二进制代码构成机器指令,这种二进制编码的计算机语言就是机器语言。机器语言描述的程序称为目标程序,只有目标程序才能被 CPU 直接执行。指令是用于指出计算机进行的操作和操作对象的一组代码,一条指令通常由操作码和操作数两部分组成。其中,操作码指出计算机进行的具体操作,如加法、减法等;操作数说明操作的对象。操作码比较简单,只需对每一种操作指定确定的二进制代码就可以了;操作数比较复杂,首先它可以有一个、两个或三个,分别称为单操作数、双操作数或三操作数,其次,操作数可能存放在不同的地方,既可以存放在寄存器中,也可以存放在存储器中,甚至直接存放在指令中。下面的二进制代码序列就是一条 Intel 8086 的机器指令。

39	24	23	0
10000000	00000110	01100100	00000000 00010100

指令中的前 16 位 10000000 00000110 为操作码,指出该指令为加法操作;后 24 位为操作数,第一个加数是二进制码 00000000 01100100,表示操作数存放在内存地址为 100 的字节单元中,第二个加数是二进制码 00010100,表示操作数为 20,它直接写在指令中。处理器逐条取指令、分析指令、执行指令,来完成人们所赋予的任务。

一台计算机全部指令的集合构成该计算机的指令系统。指令系统是计算机基本功能的体现,不同的机器指令对应的二进制代码序列各不相同。机器语言是面向机器的,不同机器之间的语言是不通用的,这也是机器语言是“低级”语言的含义所在。用二进制编写

程序相当麻烦,写出的程序也难以阅读和调试。

1.1.2 汇编语言

汇编语言是一种采用助记符表示相应机器指令的操作码和操作数,按照一定格式书写的面向机器的程序设计语言。汇编语言指令与机器指令基本上是一一对应的,其指令只有经过翻译程序(称为汇编程序)编译后才能变成机器指令。汇编语言是面向机器的语言,处理器不同,相应的汇编语言就不同,因此,汇编语言是低级语言。汇编语言用助记符来表示指令的操作码和操作数,用符号表示地址、常量和变量,助记符一般都是英语单词的缩写,因而便于人们书写、阅读和检查。例如,对于前面的例子,用宏汇编语言来书写则应为:

```
ADD WORD PTR DS:[100],20
```

其中,ADD 为助记符,DS:[100] 表示在当前数据段偏移地址为 100 单元中的内容是目的的操作数,而源操作数是 20,相加的结果存入目的操作数所在的单元中。比较此两种语言,可以看出,汇编语言比机器语言要直观得多,方便人们的书写与记忆。

用汇编语言编写的程序称为汇编源程序,计算机不能直接识别,必须将其翻译成由机器指令组成的程序后,CPU 才能执行,这一过程称为“汇编”。用于将汇编源程序翻译成机器语言的程序称为汇编程序,这种由源程序经过机器翻译转换成的机器语言程序也称为目标程序。目标程序还不能直接交给 CPU 执行,它还需要通过连接程序装配成可执行程序才能被执行。它们之间的关系如图 1-1 所示。

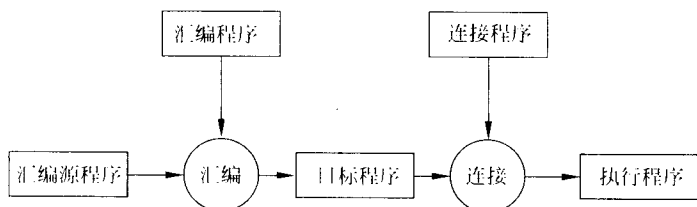


图 1-1 汇编程序与目标程序、可执行程序之间的关系

汇编程序 MASM.exe 是一种专门用于将 Intel 8086 的汇编语言源程序翻译成机器语言程序的翻译器,它在执行汇编程序时首先进行语法检查,指出错误的位置及类型,编程人员要一一进行修改,直到源程序无错,汇编程序才执行翻译工作,因此,汇编程序决定了汇编语言的语法。本书以 Microsoft 公司的 MASM.exe V5.0 作为汇编程序。

连接程序 LINK.exe 具有将多个目标程序装配在一起的功能,它也可以将目标程序与预先编写好的一些放在子程序库中的子程序连接在一起,构成较大的执行文件。汇编语言的源程序一般以 .asm 作为文件扩展名,目标程序以 .obj 作为文件扩展名,执行程序以 .exe 作为文件扩展名。

1.1.3 高级语言

高级语言是一种与具体的计算机硬件无关、独立于机器类型的通用语言,比较接近人类自然语言的语法,用高级语言编程不必了解和熟悉机器的指令系统,更容易被人们掌握

和使用。高级语言采用接近自然语言的词汇,其程序的通用性强,易学易用,这些语言面向计算机求解问题的过程,不依赖具体机器。高级语言也要翻译成机器语言才能在机器上执行。其翻译有两种方式,一种是把高级语言程序翻译成机器语言程序,然后在机器上执行,这种翻译称为编译方式,大多数高级语言如 PASCAL 语言、C 语言等都是采用这种方式;另一种是直接把高级语言程序在机器上运行,一边解释一边执行,这种翻译方式称为解释方式,如 BASIC 语言就采用这种方式。

高级语言源程序在不对硬件编程时,是在未考虑机器结构特点情况下编写的,经过翻译后的目标程序往往不够精练,过于冗长,加大了目标程序的长度,占用较大存储空间,执行时间较长。

1.1.4 汇编语言的特点

汇编语言使用助记符和符号地址,所以它要比机器语言易于掌握,与高级语言相比较,汇编语言有以下特点:

1. 汇编语言与机器关系密切

汇编语言中的指令是机器指令的符号表示,与机器指令是一一对应的,因此它与机器有着密切的关系,不同类型的 CPU,有不同的汇编语言,也就有各种不同的汇编程序。汇编语言源程序与高级语言源程序相比,其通用性和可移植性要差得多。

2. 汇编语言程序效率高

由于构成汇编语言主体的指令是用机器指令的符号表示的,每一条指令都对应一条机器指令,且汇编语言程序能直接利用机器硬件系统的许多特性,如它允许程序员利用寄存器、标志位等编程。用汇编语言编写的源程序在编译后得到的目标程序效率高,主要体现在空间效率和时间效率上,即目标程序短、运行速度快这两个方面,在采用相同算法的前提下,任何高级语言程序在这两方面的效率与汇编语言相比都望尘莫及。

3. 特殊的使用场合

汇编语言可以实现高级语言难以胜任甚至不能完成的任务。汇编语言具有直接和简捷的特点,用它编制程序能精确地描述算法,充分发挥计算机硬件的功能。在过程控制、多媒体接口、设备通信、内存管理、硬件控制等方面的程序设计中,用汇编语言直接方便,执行速度快,效率高。

汇编语言提供了一些模块间相互连接的方法,一个大的任务可以分解成若干模块,将其中执行频率高的模块用汇编语言编写,可以大大提高大型软件的性能。

1.2 计算机中数据的表示方法

计算机中的数据信息包括数值数据和非数值数据,数值数据有确定的值,用来表示数的大小,对它们能进行各种算术运算和处理;其余数据为非数值数据,如字符数据,它可以方便表示字符与文字信息。

1.2.1 进位记数制

数可以用不同的进制来表示,日常生活中通常用十进制计数法表示数据。在计算机中,为了便于信息的存储和计算,采用二进制计数法表示数据,计算机能识别、计算和存储的是由0和1两种符号构成的二进制数,而二进制数在书写和记忆时都显得过于冗长、麻烦。十进制数虽然是人们最熟悉的一种进位计数法,但它与二进制之间相互转换比较复杂。为了便于人们对二进制的描述,应该选择一种易于与二进制数相互转换的数制。显然,使用 2^n 作为基数的八进制数和十六进制数是能适合人们的这种要求的,可以说八进制数和十六进制数是二进制的两种缩写形式,三位二进制数压缩为一位八进制数,四位二进制数压缩为一位十六进制数。同一个数值可以用不同的数制表示,不同的数制可以相互转换。汇编语言中只有整数可以直接处理,所以,在此仅讨论整数在不同数制间的表示方法。

任何一个数制都涉及到以下三个问题:

1. 计数符号

计数符号用于书写和区别数值,任何一个 K 进制的数符必然包含 K 个有序的符号。通常使用的四种进制的计数符号分别如下:

- 二进制的计数符号有0、1;
- 八进制的计数符号有0、1、2、3、4、5、6、7;
- 十进制的计数符号有0、1、2、3、4、5、6、7、8、9;
- 十六进制的计数符号有0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F。

2. 基数和权

任何一个 K 进制数都可以表示成如下形式:

$$a_n r^n + a_{n-1} r^{n-1} + \cdots + a_0 r^0 + a_{-1} r^{-1} + \cdots + a_{-m} r^{-m} \quad (1.1)$$

其中: r 称为一个数制的基数, r^i 称为 K 进制数第 i 位的权, $a_n, a_{n-1}, \cdots, a_0, \cdots, a_{-m}$ 为计数符号。

下面举例说明上述四种进制的计数形式的表示:

二进制数 $10101101 = 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$

八进制数 $46373 = 4 \times 8^4 + 6 \times 8^3 + 3 \times 8^2 + 7 \times 8^1 + 3 \times 8^0$

十进制数 $6952 = 6 \times 10^3 + 9 \times 10^2 + 5 \times 10^1 + 2 \times 10^0$

十六进制数 $8D1E = 8 \times 16^3 + 13 \times 16^2 + 1 \times 16^1 + 14 \times 16^0$

3. 计数规则

当 K 进制数进行加法或减法运算时,其计数规则就是“逢 K 进1,借1当 K ”。

1.2.2 数制转换

1. 十进制数与非十进制数之间的转换

(1) 十进制数转换成非十进制数

将十进制数除以基数取余,直到商为0为止,最后从下往上取余数,即得到相应的非十进制数。

【例 1-1】 将十进制数107转换为二进制数、八进制数和十六进制数。

解: $(107)_{10} = (K_6 K_5 K_4 K_3 K_2 K_1 K_0)_2 = (110\ 1011)_2$