普通高校“十一五”规划教材

朱　兵　彭宣戈　主编

汇编语言程序设计图文教程

北京航空航天大学出版社

普通高校“十一五”规划教材

汇编语言程序设计图文教程

朱　兵　彭宣戈　主编

北京航空航天大学出版社

内 容 简 介

汇编语言是计算机科学与技术专业学生的必修专业基础课程，汇编语言的应用在系统软件开发、实时控制和实时处理领域中有着重要的地位。本书以 Intel 8086/8088 CPU 为核心，全面介绍汇编语言的相关知识，阐述汇编语言程序设计的方法及技巧。本书在大量实例中均配以相应图片解释，使读者对每个实例的操作全过程一目了然，更容易理解和掌握汇编语言。

本书可作为高等院校、高职高专计算机与相关专业的教材，也可作为相关工程技术人员及自学者的参考书。

图书在版编目(CIP)数据

汇编语言程序设计图文教程/朱兵，彭宣戈主编. —北京：北京航空航天大学出版社，2009.3

ISBN 978-7-81124-562-2

Ⅰ.汇… Ⅱ.①朱…②彭… 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(2008)第 204733 号

汇编语言程序设计图文教程

朱 兵 彭宣戈 主编

责任编辑 孔祥燮 范仲祥

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(100191) 发行部电话：(010)82317024 传真：(010)82328026

http://www.buaapress.com.cn E-mail：bhpress@263.net

北京市松源印刷有限公司印装 各地书店经销

*

开本：787 mm×960 mm 1/16 印张：23.75 字数：532 千字

2009 年 3 月第 1 版 2009 年 3 月第 1 次印刷 印数：4 000 册

ISBN 978-7-81124-562-2 定价：35.00 元

《汇编语言程序设计图文教程》
编写委员会

主　编　朱　兵　彭宣戈

副主编　章　谋　冷　明

编　委　李　莉　曾小荟　谭　彬

王晓军　周旭艳　肖　斌

前　言

汇编语言是计算机能够提供给程序员使用的最快而又最有效的语言，也是能够直接控制计算机硬件的唯一语言。对于编写高性能的系统软件和应用软件，汇编语言是最有效的语言之一。对于高等院校计算机专业的学生，“汇编语言程序设计”是一门必修的专业基础课程。通过该课程的学习，能够使学生深入理解计算机内部完成各种复杂操作和运算的基本原理。该课程对于培养学生掌握程序设计基本技能和调试技术也十分重要。同时，汇编语言的主要应用领域是工业控制，而工业控制中使用的计算机和单片机有很多具有与 8086/8088 相似的结构。例如 Intel 公司的 MCS－51 单片机与 8086/8088 计算机结构非常相似，并且指令系统也几乎相同。因此，本书介绍的 8086/8088 汇编语言也是为掌握工控机的应用铺平道路。可以说，学习汇编语言与计算机硬件系统是相辅相成的。本书的目的就是使读者通过熟练掌握汇编语言而进入工控领域。

全书共分 9 章。第 1 章介绍汇编语言的基础知识与基本概念。第 2 章介绍 8086/8088 CPU 的内部结构，以及汇编语言程序上机调试的方法和手段。第 3 章介绍 8086/8088 CPU 的寻址方式及指令系统。第 4 章介绍伪指令。第 5 章介绍汇编语言程序设计方法。第 6 章介绍输入/输出方法。第 7 章介绍中断与基本输入/输出系统 BIOS。第 8 章介绍高级汇编语言技术，包括宏汇编、结构及条件汇编等。第 9 章介绍 80386 汇编语言程序设计基础。

本书由朱兵、彭宣戈任主编，章谋、冷明任副主编。其中第 1 章由江西省公安专科学校肖斌编写；第 2 章由深圳市警察训练学校李莉编写；第 3 章由井冈山大学王晓军编写；第 4 章由江西华忆电子工业中等专业学校章谋编写；第 5 章由井冈山大学朱兵编写；第 6 章由井冈山大学曾小荟编写；第 7 章由井冈山大学周旭艳编写；第 8 章由井冈山大学谭彬编写；第 9 章由井冈山大学冷明编写；附录由井冈山大学彭宣戈编写。全书由朱兵、彭宣戈提出框架，并负责统稿。

本书在编写过程中参考了国内有关书籍资料，在此谨向有关作者表示衷心地感谢。

由于编者的水平有限，加之时间仓促，本书缺点和不当之处在所难免，欢迎广大教师、同行专家以及各位读者批评指正。

编　者

2008 年 12 月

目 录

第1章 概 述

程序设计语言是开发计算机各种软件的工具，它经历了由低级语言到汇编语言，再到高级语言的发展过程。其中，汇编语言是一种能够充分利用计算机硬件特性的低级语言，它与计算机的结构有着非常紧密的联系。虽然高级语言能够实现绝大部分机器语言可以实现的功能，但汇编语言还是经常被用来改进计算机软件和硬件控制系统的工作效率，以及用于高级语言的程序调试，为计算机系统提供高速、高效的代码。因此学习和掌握汇编语言程序设计的内部细节对于计算机的学习非常有益。

1.1 汇编语言概述

自从第1台计算机 ENIAC 于1946年诞生以来，计算机的发展经历了电子管、晶体管、集成电路和超大规模集成电路4代，目前正朝着智能化的第5代计算机发展。计算机的应用已渗透到社会和生活的各个领域，人们与计算机进行交流的“语言”也经历了机器语言、汇编语言和高级语言3个重要阶段的发展，正朝着“自然语言”的方向发展。

1.1.1 机器语言

机器语言是一种计算机能直接识别和执行的用二进制代码表示的机器指令的集合。程序设计者利用它可以直接控制计算机的硬件。机器语言具有直接执行和速度快等特点。不同型号的计算机其机器语言是不相通的，即按照某种型号计算机的机器指令系统编制的程序，不能在另一种型号的计算机上执行。

用机器语言编写程序，编程人员要首先熟记所用计算机的全部指令代码和代码的含义。手工编写程序时，程序员需自己处理每条指令和每一数据的存储分配及输入/输出，还要记住编程过程中每步所使用的工作单元处于何种状态。因此，使用机器语言编写程序是一件十分烦琐的工作，花费的时间往往是程序实际运行时间的几十倍或几百倍；而且编出的程序全是0和1的指令代码，直观性差，还容易出错。现在，除了计算机生产厂家的专业人员外，绝大多数程序员已经不再学习机器语言了。

1.1.2 汇编语言

汇编语言是用助记符表示指令功能的计算机语言。用汇编语言编写的程序称为汇编语言源程序。汇编语言源程序无法被计算机直接执行，必须用专门的系统程序将它翻译成机器代

码，计算机才能识别并执行，而这一翻译过程称为汇编。

与机器语言相比，汇编语言具有以下几个特点：第一，它使用符号来表示操作码和地址码，这种符号便于记忆，称之为记忆码；第二，汇编过程中自动处理存储单元的分配，无须程序员做存储分配工作；第三，程序员可以直接书写十进制数。

汇编语言实现了对机器语言的抽象：首先表现在将机器语言的每一条指令符号化，指令码代之以记忆符号，地址码代之以符号地址，使得其含义显现在符号上而不再隐藏在编码中，可让人望“文”生义；其次表现在汇编语言摆脱了具体计算机的限制，可在不同指令集的计算机上运行，只要该计算机配备汇编程序即可。

1.1.3 高级语言

虽然汇编语言用符号替代机器指令，从一定程度上简化了程序的编写、阅读和调试，但是没有经过汇编语言训练的程序设计人员，编写汇编语言程序还是十分困难的，不利于计算机的推广与应用。于是，人们尝试设计一种脱离具体的计算机，面向过程、更符合人类思维和更容易被人们理解和学习的语言，从而演化出高级语言。20 世纪 50 年代末，第一个用于科学计算的高级语言——FORTRAN 语言诞生了，随后各种高级语言如雨后春笋般涌现出来。目前，世界上使用的高级语言多达数百种，其中具有代表性的高级语言有 C、C++、Visual Basic 和 Delphi 等常用高级语言。

由于计算机只能识别由 0 和 1 代码组成的机器语言程序，所以各种高级语言的源程序也必须经过相应的编译程序或解释程序翻译成目标程序，才能被计算机识别并执行。

1.1.4 3 种语言的特点比较

机器语言是由 0 和 1 代码组成的面向机器的语言。机器语言程序的编写、阅读和调试都十分困难；但它是计算机可直接识别并执行语言程序，内存占用少，执行速度快。

汇编语言是一种用符号替代机器指令的低级语言。与机器语言相比，汇编语言易于理解和记忆，汇编语言程序也易于编写、阅读与调试。由于其语句与机器指令语句一一对应，所以具有内存占用少，执行速度快等特点，并且能直接控制计算机的硬件设备，充分发挥计算机的硬件功能。从本质上讲，汇编语言仍然是面向机器的语言，这就要求程序设计者熟悉计算机的内部结构，掌握计算机的指令系统以及对应的伪指令，这就给学习汇编语言带来了困难。另外，汇编语言的指令语句与程序设计者所要完成的任务有很大差别。汇编语言的指令语句只能实现简单的操作，如寄存器数据的移位、寄存器与内存之间数据的传送、数据的算术逻辑运算等，而这些远非程序设计者所要完成的任务。例如，要做 1+2 的操作并将结果在屏幕上显示出来，必须使用一系列汇编指令并进行 DOS 功能调用才能实现；而高级语言只需要一条指令即可完成该操作。

高级语言是一种类似于自然语言和数学描述语言的程序设计语言。其主要特点是：面向

过程，脱离具体的机器。高级语言易于编写、阅读和调试，且可移植性好。但是高级语言源程序必须经过编译后才能翻译成计算机可识别、执行的目标程序，所以不能产生有效的机器语言代码，致使其占用内存空间大，程序运行速度较慢。

1.2 计算机中数据和字符的表示

计算机只能识别0和1代码，进入计算机的任何信息都必须转换成0和1代码。本节说明计算机如何利用0和1代码表示现实中的各种数值和字符等内容。

1.2.1 数 制

在汇编语言设计中，经常用到的数制有二进制、十进制和十六进制。

1. 二进制

在计算机中，为便于存储及计算的物理实现，采用二进制。二进制数的基数为0和1，各个数值的权为2^{i-1}(i为数值所在位置，如小数点左边第一位为个位，i=1；小数点右边为0位，i=0)，并且二进制遵循“逢二进一，退一当二”的运算规则。

二进制数$a_n a_{n-1}\cdots a_1 . b_0 b_1 \cdots b_m$可表示为：

$$a_n\times 2^{(n-1)}+a_{n-1}\times 2^{(n-2)}+\cdots+a_1\times 2^{(1-1)}+b_0\times 2^{(0-1)}+b_1\times 2^{(-1-1)}+\cdots+b_m\times 2^{(-m-1)}$$

其中，a_i和b_j非0即1。

2. 十进制

在人们的生活中，已经习惯使用十进制。十进制数的基数有10个，即0～9十个数，各个数值的权为10^{i-1}(i为数值所在位置)，并且十进制遵循“逢十进一，退一当十”的运算规则。

十进制数$a_n a_{n-1}\cdots a_1 . b_0 b_1 \cdots b_m$可表示为：

$$a_n\times 10^{(n-1)}+a_{n-1}\times 10^{(n-2)}+\cdots+a_1\times 10^{(1-1)}+b_0\times 10^{(0-1)}+b_1\times 10^{(-1-1)}+\cdots+b_m\times 10^{(-m-1)}$$

其中，a_i和b_j为0～9中任意的一个数。

3. 十六进制

由于二进制数书写较长，容易出错，因此常用易于与二进制数转换的十六进制数来描述二进制数。十六进制数的基数有16个，即0～9、A～F，各个数值的权为16^{i-1}(i为数值所在位置)，并且遵循“逢十六进一，退一当十六”的运算规则。

十六进制数$a_n a_{n-1}\cdots a_1 . b_0 b_1 \cdots b_m$可表示为：

$$a_n\times 16^{(n-1)}+a_{n-1}\times 16^{(n-2)}+\cdots+a_1\times 16^{(1-1)}+b_0\times 16^{(0-1)}+$$
$$b_1\times 16^{(-1-1)}+\cdots+b_m\times 16^{(-m-1)}$$

其中，a_i和b_j为0～9或A～F中的任意一个数。

注意： 汇编语言中通常用B或b结尾表示该数据为二进制数；用字母H或h结尾表示该

数据为十六进制数；用字母 D 或 d 结尾表示该数据为十进制数。如果数据结尾没有跟任何字符，则默认该数据为十进制数。例如 10101100B、1223D、1100 和 0ABCDH，分别为二进制、十进制、十进制和十六进制数。其中，十六进制数如果数值的开头为 A～F，则必须在前面加 0；否则汇编程序认为该数值为一个标号，从而会导致在编译过程中出错。

4. 不同进制数之间的转换

不同进制数之间的转换主要有：二进制或十六进制数转换为十进制数，十进制数转换为二进制数，二进制数转换为十六进制数以及十六进制数转换为二进制数。

二进制或十六进制数转换为十进制数口诀：将各个数值乘以其相应的权，最后相加所得到的和即为等值的十进制数。

十进制数转换为二进制数口诀：十进制整数部分不断除以 2，取余，最后从下往上排列；小数部分不断乘以 2，取整，最后从上往下排列；小数点位置不变。

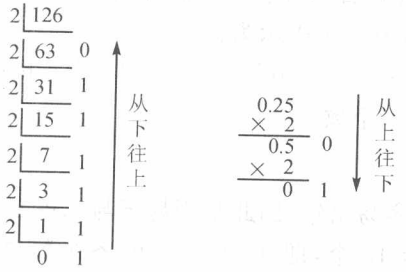
二进制数转换为十六进制数口诀：从小数点向两边分，每 4 位一组，左边不够 4 位在前面补 0，右边不够 4 位在后面补 0，最后将每组二进制数转换为相应的十六进制数值即可，小数点位置不变。

十六进制数转换为二进制数口诀：每位十六进制数值转换为 4 位相应的二进制数，小数点位置不变。

说明：十进制数转换为十六进制数可以先将十进制数转换为二进制数，再将二进制数转换为十六进制数。

例 1－1 将 126.25D 转换为二进制数。

解：根据十进制数转换为二进制数口诀，将整数部分 126 不断除以 2，取余数；小数部分不断乘以 2，取整。其运算过程如下：



整数部分转换结果为 1111110B，小数部分转换结果为 0.01B，所以 126.25D＝1111110.01B。

例 1－2 将 0ABCD.3678H 转换为二进制。

解：十六进制数转换成二进制数只需将十六进制的每个数值转换为对应的 4 位二进制数值即可。其转换过程如下：