面向21世纪

高等学校计算机类专业系列教材

汇编语言程序设计

Assembly Language Programming

李 强 温 春 编著

王 闵 主审

西安电子科技大学出版社

http://www.xduph.com

面向21世纪高等学校计算机类专业系列教材

汇编语言程序设计

Assembly Language Programming

李强　温春　编著

王闵　主审

西安电子科技大学出版社

2003

内 容 简 介

全书共分 10 章。第 1 章是基础知识，概括地介绍了数在计算机中的表示；第 2 章介绍了 80x86 和 Pentium 微处理器的结构及存储器的组成；第 3 章介绍的是寻址方式和指令系统；第 4 章讨论了伪指令和汇编语言的程序结构；第 5～9 章是本书的核心内容，主要讲述了汇编语言程序设计的基本方法和技巧；第 10 章介绍了中断和输入/输出程序的设计方法。

本书可以作为高等学校计算机科学、自动化和相关专业本科教材，也可以作为从事相关技术工作人员的参考书。

★ 本书配有电子教案，需要者可与出版社联系，免费索取。

图书在版编目（CIP）数据

汇编语言程序设计=Assembly Language Programming / 李强等编著.

—西安：西安电子科技大学出版社，2003.8

(面向 21 世纪高等学校计算机类专业系列教材)

ISBN 7-5606-1264-4

Ⅰ. 汇… Ⅱ. 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（2003）第 048111 号

策　　划　马乐惠

责任编辑　曹　华

出版发行　西安电子科技大学出版社（西安市太白南路 2 号）

电　　话　(029)8242885　8201467　　邮　编　710071

http://www.xduph.com　E-mail: xdupfxb@pub.xaonline.com

经　　销　新华书店

印刷单位　陕西画报社印刷厂

版　　次　2003 年 8 月第 1 版　2003 年 8 月第 1 次印刷

开　　本　787 毫米×1092 毫米　1/16　印张 21.375

字　　数　503 千字

印　　数　1～4 000 册

定　　价　23.00 元

ISBN 7 – 5606 – 1264 – 4 / TP • 0664（课）

XDUP 1535001-1

序

第三次全国教育工作会议以来，我国高等教育得到空前规模的发展。经过高校布局和结构的调整，各个学校的新专业均有所增加，招生规模也迅速扩大。为了适应社会对“大专业、宽口径”人才的需求，各学校对专业进行了调整和合并，拓宽专业面，相应的教学计划、大纲也都有了较大的变化。特别是进入21世纪以来，信息产业发展迅速，技术更新加快。面对这样的发展形势，原有的计算机、信息工程两个专业的传统教材已很难适应高等教育的需要，作为教学改革的重要组成部分，教材的更新和建设迫在眉睫。为此，西安电子科技大学出版社聘请南京邮电学院、西安邮电学院、重庆邮电学院、吉林大学、杭州电子工业学院、桂林电子工业学院、北京信息工程学院、深圳大学、解放军电子工程学院等10余所国内电子信息类专业知名院校长期在教学科研第一线工作的专家教授，组成了高等学校计算机、信息工程类专业系列教材编审专家委员会，并且面向全国进行系列教材编写招标。该委员会依据教育部有关文件及规定对这两大类专业的教学计划和课程大纲，对目前本科教育的发展变化和相应系列教材应具有的特色和定位以及如何适应各类院校的教学需求等进行了反复研究、充分讨论，并对投标教材进行了认真评审，筛选并确定了高等学校计算机、信息工程类专业系列教材的作者及审稿人。这套教材预计在2004年春季全部出齐。

审定并组织出版这套教材的基本指导思想是力求精品、力求创新、好中选优、以质取胜。教材内容要反映21世纪信息科学技术的发展，体现专业课内容更新快的要求；编写上要具有一定的弹性和可调性，以适合多数学校使用；体系上要有所创新，突出工程技术型人才培养的特点，面向国民经济对工程技术人才的需求，强调培养学生较系统地掌握本学科专业必需的基础知识和基本理论，有较强的本专业的基本技能、方法和相关知识，培养学生具有从事实际工程的研发能力。在作者的遴选上，强调作者应在教学、科研第一线长期工作，有较高的学术水平和丰富的教材编写经验；教材在体系和篇幅上符合各学校的教学计划要求。

相信这套精心策划、精心编审、精心出版的系列教材会成为精品教材，得到各院校的认可，对于新世纪高等学校教学改革和教材建设起到积极的推动作用。

系列教材编委会

2002年8月

高等学校计算机、信息工程类专业

系列教材编审专家委员会

主　任：杨　震（南京邮电学院副院长、教授）
副主任：张德民（重庆邮电学院通信与信息工程学院院长、教授）
　　韩俊刚（西安邮电学院计算机系主任、教授）
　　李荣才（西安电子科技大学出版社总编辑、教授）

计算机组

组　长：韩俊刚（兼）
成　员：（按姓氏笔画排列）
　　王小民（深圳大学信息工程学院计算机系主任、副教授）
　　王小华（杭州电子工业学院计算机分院副院长、副教授）
　　孙力娟（南京邮电学院计算机系副主任、副教授）
　　李秉智（重庆邮电学院计算机学院院长、教授）
　　孟庆昌（北京信息工程学院教授）
　　周　娅（桂林电子工业学院计算机系副主任、副教授）
　　张长海（吉林大学计算机科学与技术学院副院长、教授）

信息工程组

组　长：张德民（兼）
成　员：（按姓氏笔画排列）
　　方　强（西安邮电学院电信系主任、教授）
　　王　晖（深圳大学信息工程学院电子工程系主任、副教授）
　　胡建萍（杭州电子工业学院电子信息分院副院长、副教授）
　　徐　祎（解放军电子工程学院电子技术教研室主任、副教授）
　　唐　宁（桂林电子工业学院通信与信息工程系副主任、副教授）
　　章坚武（杭州电子工业学院通信工程分院副院长、教授）
　　康　健（吉林大学通信工程学院副院长、教授）
　　蒋国平（南京邮电学院电子工程系副主任、副教授）

总 策 划：梁家新
策　　划：马乐惠　云立实　马武装　马晓娟
电子教案：马武装

前　言

汇编语言是用户能够利用计算机硬件特性，直接控制硬件的程序设计语言。计算机的系统程序多数是使用汇编语言编写的，利用汇编语言可以编写出时空效率高的程序，在某些领域，汇编语言仍然是必不可少的编程语言之一。可以说，汇编语言是非常有效的一种语言，汇编语言程序设计也就成为计算机科学与技术专业的一门重要课程，是其他相关专业的一门必修或选修课。因此，学习和掌握汇编语言程序设计方法是非常必要的。我们在总结多年教学实践经验的基础上，编写了这本《汇编语言程序设计》教材。

以 Intel 80x86/Pentium 系列微处理器或者兼容的微处理器为 CPU 的微型计算机在国内得到了广泛使用。因此，本书以 80x86/Pentium 系列微处理器为基础，以讲述汇编语言程序设计的基本概念、基本理论和基本方法为原则，系统地介绍了 80x86/Pentium 微处理器的特点、汇编语言程序结构、数据组织、简单的汇编语言程序设计、复杂的汇编语言程序设计和高级汇编技术，采取由浅入深、循序渐进的方法，结合编程实例，描述中力求语言简练，便于读者理解和掌握。

全书共分 10 章。第 1 章是基础知识，概括地介绍了数在计算机中的表示；第 2 章介绍了 80x86/Pentium 系列微处理器的结构及存储器的组成，这既是汇编语言运行的硬件环境，又是本书后续内容的基础；第 3 章介绍的是寻址方式和指令系统；第 4 章则讨论了伪指令和汇编语言的程序结构；第 5～9 章是本书的核心内容，主要讲述了汇编语言程序设计的基本方法和技巧；第 10 章介绍了中断和输入/输出程序的设计方法。考虑到教学计划的课时安排，本书未涉及保护模式下汇编语言程序设计的方法和浮点数编程的内容。

本书第 1、2、4 章及第 6～10 章由李强编写并负责全书的修改、补充、定稿，第 3、5 章由温春编写并完成全书的绘图和全部程序的调试与验证。参加本书录入、校对、程序设计等工作的还有许少华、滕娇春、唐超、辛元、鱼群、刘新勇、宋庆雷、陈建熊等。

本书的编写得到了解放军电子工程学院训练部教保处、教务处和系统工程教研室的大力支持，黄发文、江灏对本书的编写给予了支持并提出了许多宝贵的意见，在此对他们的帮助表示衷心的感谢。本书的初稿曾在解放军电子工程学院 97 级、98 级和 99 级指挥自动化、计算机技术与应用等专业试用，得到了许多同志的帮助，在此一并表示感谢。

西安电子科技大学计算机学院的王闵老师审阅了全书，并对全书内容的修改提出了宝贵意见，在此谨表衷心的感谢。

限于编者的水平，加之编写时间仓促，在本书的选材与内容安排上难免有不妥和错误之处，恳请读者和同行批评指正。

作　者

2003 年 4 月

目　录

第1章 基础知识

1.1 微型计算机系统的发展

1.1.1 微机技术的发展概况

随着半导体集成技术的发展，微型计算机的性能价格比日益提高，微型计算机已深入普及到人们的工作、学习、生活的各个领域。计算机已在处理速度、存储容量、人机交互、方便程度、直观程度、智能化等方面得到了飞速发展。

1. 处理速度

微型计算机的处理速度在不断提高，主要表现在以下几个方面：

(1) CPU 的主频不断提高，如从 4.77 MHz 的 8088CPU 提高到几个 GHz 的 PentiumⅣ，这主要是因为半导体制造工艺水平的提高。

(2) 增加了数据运算的宽度，从早期的 8 位发展到现在的 32 位、64 位、128 位甚至更高。

(3) 采用了 RISC 技术。从 CISC 体系结构转变为 RISC 体系结构后，在同样的工艺水平(即同样的主频、同样的工艺尺寸、同样的芯片面积)下，使 CPU 的速度和性能有了很大的提高。

(4) 降低电路工作电压，不断缩小工艺尺寸。工作电压的降低有利于工艺尺寸的缩小。

(5) 增加了芯片内并行工作的信息处理部件，如整数部件、浮点加法器、浮点乘法器、图形部件等。芯片内具有两个以上的并行指令流水线，每次并行地向各流水线分配多条指令以进行并行处理，这是利用硬件资源的重复来换取芯片性能的提高。

2. 存储容量

微机的计算、处理速度的提高必然要求存取信息的速度相应提高。当今存储器速度还远远跟不上 CPU 的速度，而存储容量的大小是与存取速度相矛盾的，因而只能采用分级存储方式。

(1) 芯片内的寄存器堆、指令 Cache 和数据 Cache (4～16 KB)都能在一个时钟周期完成存取，而且随着 VLSI 的发展其容量将继续增加。

(2) 芯片外的二级指令 Cache 和数据 Cache，其大小在 16～256 KB。

(3) 主存储器的大小在 1 MB 到几百个 MB。

(4) 外部存储器仍以软盘、硬盘、光盘为主要介质，其容量可在 1 MB 至上百个 GB。由于 RAM 芯片性能价格比提高，因而为克服外部存储器的慢速寻找，一般都采用 RAM

Cache(256 KB～2 MB)来减小等待时间。

除了解决大容量存储外，外部存储器还具有掉电时的信息保持特性。快擦存储器(Flash Memory)芯片的普及为外部存储提供了一种新的实现方法。

3. 人机接口

传统的键盘输入、显示、打印输出已经不能满足人们的需要，人们希望计算机有更生动、直观、灵活的用户界面，最好能像人一样既可以接收、识别并理解声、文、图信息，又能给出声音、视像等信息。微型计算机系统性能的提高，价格的下降，已经为这些技术的实用化提供了基础。计算机处理视像和声音的多媒体技术也逐步地进入到各个应用领域。

1) 汉字识别

汉字识别要比英文、数字识别难得多，因为汉字字数多、字体多。汉字识别是从扫描输入开始的，现在扫描仪的扫描精度已经完全够用，对静止图像输入来说其扫描精度与彩色分辨率也已能满足要求。通过对版面的分析和汉字的切分，计算机可对汉字进行识别。由于汉字字数多、字体多，因而其特征的选择与提取是识别的关键。印刷体汉字的识别已实用化，一般识别率为95%以上，识别速度为20～30字/秒。

2) 语音识别

汉语中一个汉字对应一个单音节，能独立发音的单音节(包括4种声调)只有1283个，从这点看对汉语的语音识别较为有利。但汉语中同音字很多，因而单音节识别率的提高是有难度的，利用字构成词的相关信息，词的识别率可以提高。中词表的特定人与小词表的非特定人的汉语语音识别已初步达到实用水平。

3) 视频图像输入

从摄像机、录像机和彩电中来的视频图像信息，通过视频处理部件转换成二进制代码，以一定的图像文件格式存入计算机的外部存储器中。以VGA全屏显示方式计算(分辨率为640×480，共256种色彩)则显示一帧所需存放的信息为300 KB。如果以50帧计算，则存放连续半小时的视像信息需要27 GB，对当前微型机来说这个信息量太大了。

由于图像对视觉的感受是整体的，部分的少量的错误不会引起直观的错误，因而图像信息的存储可采用不完全复真的压缩和还原方法。如用JPEG算法可使图像信息压缩为原来的1/8～1/25，静止图像即相当于一帧视像，需存储300 KB/幅，经过压缩后存储，即使有1万幅图像也只需120 MB的存储量。因而，当今微型机系统完全可以用于图像数据库。

4. 语音合成

声音的输出可分为两种情况，一种是声音采样存储后再回放，一种是文一语的转换。

对语音和音乐的采样要求是不同的。语音采样频率在16～32 kHz，精度在8位左右即可，音乐的采样频率在40 kHz，精度要14位左右。如果不采取压缩方法的话，连续半小时的乐曲存储量为144 MB，这个存储量还是很大的。因而，一般都采用压缩算法，语音可压缩为原来的1/8，而音乐仅能压缩到原来的1/4。用上述采样方法存储后再回放，可以和原信号毫无差异。数字录音技术即采用这种方法。

文一语的转换，首先要切分词、字，再找词、字的语言库，然后组成句子。句子中的语调实现是较为困难的。有限词汇的语音合成已经达到实用水平。

5. 多媒体技术产品

上述的技术为多媒体技术产品提供了基础。

1.1.2 Intel 微处理器体系及其演变

在介绍 Intel 微处理器体系之前，先介绍微处理器(Microprocessor)、微型计算机(Microcomputer)和微型计算机系统(Microcomputer System)这三个概念。

微处理器，简称 μP 或 MP，是指由一片或几片大规模集成电路组成的具有运算器和控制器功能的中央处理机部件，即 CPU(Control Processing Unit)。微处理器本身并不等于微型计算机，它仅仅是微型计算机的中央处理器。

微型计算机，简称 pC 或 MC，是指以微处理器为核心，配上由大规模集成电路制作的存储器、输入/输出接口电路及系统总线所组成的计算机(简称微型机)。如果把 CPU、存储器和输入/输出接口电路都集成在单片芯片上，这样构成的微型计算机称为单片微型计算机(简称单片机)。

微型计算机系统是指以微型计算机为中心，配以相应的外围设备、电源、辅助电路(统称硬件)以及控制微型计算机工作的系统软件所构成的计算机系统，简称 μCS 或 MCS。

在 20 世纪 70 年代，Intel 的 8080、8085，Motorola 的 6800、6808 与 Zilog 的 Z80 都是 8 位微处理器，但发展到 16 位及 16 位以上微处理器时，只有 Intel 与 Motorola 两种产品，Zilog 公司落后了。Motorola 公司的 MC68000 功能强于 Intel 8086，在工作站领域得到广泛使用。但由于 IBM 公司推出的 IBM PC、PC/XT 与 PC/AT 及大量兼容机采用了 Intel 公司的微处理器芯片，使得 Intel 公司实力大增。尤其是 Intel 32 位微处理器芯片 80386、80486 与 1993 年推出的 Pentium，使 Intel 微处理器成为微型机市场首选的 CPU 芯片类型。目前，Intel 的 Pentium 系列微处理器更使人有一枝独秀的感觉。

微处理器是微型机的心脏，它是利用超大规模集成电路技术将计算机 CPU 集成在一块硅片上。微型机性能的优劣基本取决于所选用的微处理器芯片功能的强弱。

1.1.3 Intel 微处理器的主要特点

1971 年推出的 4004 及其改进型 4040，是一种 4 位的微处理器芯片，它具备 MPU 的基本特点，有专用指令读入键盘数据，进行十进制运算，每次处理 4 位数据，能与键盘、存储器、显示器一起构成一个简单系统。

1972 年推出了 8008 及其改进型 8080，8080 是一种 8 位的微处理器芯片，寻址空间为 64 KB。它集成了 16 位的算术逻辑单元(ALU)和六个通用寄存器，以及程序计数器、堆栈指示器、指令寄存器与译码器等，指令集包括 78 条指令。

1978 年推出的 8086 是一种 16 位的微处理器，它的内部结构是 16 位的，数据总线也是 16 位的。它有包括乘法与除法指令的 16 位运算指令，既能处理 16 位数据，也能处理 8 位数据，在汇编语言上与 8080、8085 兼容，并增加了多条 16 位操作指令，有 20 根地址线，直接寻址能力可达 1 MB。8088 是 8086 的简化产品，它不是真正的 16 位芯片，它的内部总线是 16 位的，外部总线是 8 位的，是一种准 16 位芯片。

80186 与 80188 是另一种高集成度的 16/8 位微处理芯片。它将 8086/8088 与其他常用的 20 多个器件集成在一块芯片上，同时增加了 I/O(输入/输出)指令，寻址空间仍为 1 MB，可以使用数值协处理器 80187，但没有存储管理与保护部分。

1982 年推出的 80286 是真正的 16 位、具有存储管理与保护机制的微处理器芯片，它有实地址模式和虚拟地址保护模式两种运行方式。虚拟地址保护模式也称作保护模式。在实地址模式中，80286 兼容了 8086 的全部功能，8086 的汇编语言源程序可以不做任何修改在 80286 中运行。在保护模式中，它将实地址模式的能力与对存储器的管理，对虚拟存储器的支持，以及对地址空间的保护集为一体，因而能可靠地支持多用户系统。

1985 年推出了高性能 32 位微处理器 80386，它与 8086、80286 相兼容，这标志着 32 位微处理时代的到来。继 80386 之后，Intel 公司于 1989 年又推出了 80486 微处理器；1993 年又推出了更高性能的微处理器——Pentium。目前，Pentium 系列的微处理器已形成 Pentium Ⅰ、Pentium Ⅱ、PentiumⅢ、PentiumⅣ四个系列多种型号的产品。

1.1.4 Intel 微处理器的性能

1．80386、80486 和 Pentium 的主要技术特点

80386 是 Intel 公司为支持多任务操作，适应多种操作系统而设计的 32 位微处理器。由于 80386 有 32 位的寄存器和数据通路，因此，可以支持 32 位的地址与数据类型。80386 能够支持 4 GB 的内存物理空间，64 TB 的虚拟存储空间；支持多任务系统；支持段式管理与页式管理；支持四级特权集，对任务与任务之间、任务与操作系统之间进行严格的保护隔离。由于采用指令流水线工作方式、较高的总线带宽、片内地址转移等措施，系统速度可以达到 4 MIPS。

80486 芯片相当于将 80386 与一片数值协处理器 80387、一片 8 KB 的高速缓冲存储器 Cache 集成在一起，其中 8 KB Cache 用来存放指令与数据。由于访问内部 Cache 的速度远高于访问存储器的速度，因而可以大大提高系统性能，减少了处理器使用外部总线的时间，对内部 Cache 的操作完全由系统自动进行，对用户是透明的。80486 的常用指令运行时间为 1 个时钟周期，系统速度可达 20 MIPS。

1992 年，Intel 公司正式将继 80486 之后新一代微处理器称为 Pentium。Intel 不采用 80586 这一提法的原因主要有两点：一是商标的版权问题；二是说明 Intel 的微处理器设计由 CISC 技术向 RISC 技术过渡。出于商业原因，Intel 在产品宣传中不突出 RISC 技术，而只用“超标量”，即 Pentium 采用双流水线超标量体系结构。

Pentium 的内部总线是 32 位，但与外部存储器的接口总线是 64 位，因而处理器与存储器之间数据传输速率达 528 MB/s，它的软件与 80386、80486 兼容。其主要特点是：指令 Cache 与数据 Cache 分开，容量均为 8 KB；硬件上，两个整数执行流水线分开，地址产生部件与算术逻辑单元 ALU 分开；可以在一个运行周期中发送两条整数指令；具有片上浮点寄存器、加法与乘/除法器；可以在一个运行周期发送一条或两条浮点指令；采用转移预测方法，使用两个预取缓冲器，一个以顺序方式预取指令，一个按转移预测设置的转移目标缓存 BTB 预取指令，所需指令永远是在执行前预先取出的。Pentium 的片上存储管理部件与 80386、80486 完全兼容。

Pentium 支持多用户操作系统，可以在 Windows NT、OS/2、UNIX、Solaris 操作系统中运行，可用于局域网 client/server、虚拟实景、文字与语音识别、三维模型运算等更为广泛的应用领域。

2．80386、80486 和 Pentium 微处理器的性能比较

80386 有 32 位的寄存器和数据通道，支持 32 位的地址和数据类型，是为支持多任务操作而进行优化的操作系统所设计的 32 位微处理器。与 80386 一样，80486 也使用了指令流水线、RISC 的设计思想。80486 用静态 RAM 作为指令、数据共用的 Cache，它采用成组传送方式，能在一个时钟周期内传送 32 位数据。80486 的性能高于 80386。

从 8086 到 80386，再到 80486 直至 Pentium，可以看出 Intel 微处理器的发展趋势和特点：

① 遵循开放式标准，保持兼容性；

② 结构设计从 CISC 向 RISC 过渡；

③ 功能升级，扩大应用范围；

④ 提高基于 Intel 微处理器的微型机性能，保持整机的兼容性。

现在，微型机的应用已从简单的数值计算、文字处理向多媒体、局域网、工程 CAD、模式识别中的文字与语音识别、专家系统以及以图形界面为特征的立体图形与动态视频图像处理等方向发展。将多个中央处理器、高速数字运算单元及“智能”人机界面综合起来，以适应动态全景视频图像处理和语音与文字识别技术的要求。

1.2 汇编语言程序设计

1.2.1 程序设计语言

程序是为解决某一问题而编写在一起的指令序列，计算机程序可以用高级语言编写，也可以用汇编语言编写。

指令是规定计算机执行特定操作的命令。CPU 就是根据指令来指挥和控制微型机各部分协调地动作，以完成规定的操作。任何一条指令都包括两部分：操作码和地址码。操作码指明要完成操作的性质，如加、减、乘、除、数据传送、移位等；地址码指明参加上述规定操作的数据存放地址或操作数。计算机全部指令的集合称为计算机指令系统，指令系统准确定义了计算机的处理能力。

高级语言程序就是用高级语言编写的程序。如 C 语言源程序、PASCAL 语言源程序等均是高级语言程序。计算机是不能直接运行高级语言源程序的，要执行高级语言源程序，就必须把用高级语言编写的源程序翻译成用机器指令表示的目标程序，这样就需要有各种解释程序或编译程序。

在计算机中，以二进制代码形式存在的指令，叫机器码指令。机器码指令构成的指令系统叫机器语言，用机器语言编写的程序叫机器语言程序。用助记符构成的指令系统叫汇编语言(Assemble Language)，用汇编语言编写的程序叫汇编语言程序。汇编语言程序翻译

成机器语言程序的过程称为汇编(Assemble)，高级语言程序翻译成机器语言程序的过程叫解释(Interpretation)或编译(Compilation)。

通常将汇编、解释、编译前的程序叫源程序，而将翻译后的机器语言程序叫目标程序。完成汇编、解释、编译的程序则分别称为汇编程序(Assembler)、解释程序(Interpreter)、编译程序(Compiler)。

1.2.2 汇编语言

我们知道，计算机系统由计算机硬件和计算机软件两个部分组成，有时又将计算机的硬件与软件统称为计算机资源。

计算机的硬件是指构成计算机系统的物理实体或物理装置。由运算器、控制器、存储器、输入/输出接口等部件构成主机，称为计算机。主机再配以输入/输出设备，便构成了计算机的硬件系统。

计算机软件系统是指计算机系统所使用的各种程序的集合。它的功能是利用计算机提供的逻辑功能来合理地组织计算机的工作，以便简化人们使用计算机的环节，并为用户提供一个便于掌握、操作简便的工作环境。计算机软件通常包括：操作系统、数据库系统、计算机网络软件、汇编程序、各种高级语言的编译或解释程序、各种标准程序库、各种应用程序等。作为计算机重要组成部分的软件系统，汇编语言在其中占有重要的地位。汇编语言程序设计是计算机专业人员的基本功。那么，什么是汇编语言及汇编语言程序设计？与高级语言相比，汇编语言又有什么特点？如何学习汇编语言及如何使用汇编语言进行程序设计呢？

汇编语言(Assembler Language)是一种面向机器的程序设计语言，这是一种用符号表示的低级程序语言，通常是为特定的计算机专门设计的，与机器语言很接近。用这种语言写成的程序，需经过汇编程序翻译成机器语言程序才能被计算机执行。汇编语言指令和翻译成的机器语言之间的关系，基本上是一一对应的关系。但有的汇编语言中可以有宏指令，它与一串特定的机器指令相对应(对应方式由用户按一定规则自行定义)，这样的汇编语言有时也叫宏汇编语言。由此可看出，汇编语言是一种与计算机的代码机器指令一一对应，并紧密依赖于某一具体计算机的、面向机器的语言。

机器语言是利用0、1的各种组合来构成指令的，采用的是二进制编码。没有配备任何软件的计算机称之为裸机，它只能识别“0”和“1”两种代码。为了使用这样的计算机，程序设计人员只能用机器指令或机器语言编写程序，这就要求程序设计人员熟记计算机以“0”、“1”表示的全部指令，这是十分困难的事情。用机器语言编写程序时，工作量大，易于出错且不易修改。为了摆脱用机器指令编程的困难，出现了用指令符号来编写程序的方法，用符号语言编写的程序称为符号程序。在编制程序时，只要记住指令的助记符就可以了，而助记符可使用指令的英文名称的缩写，较指令编码容易记忆。例如，取数指令用LDA表示、加法指令用ADD表示等。对这种符号语言再作一些扩充就是所谓的汇编语言。机器语言与汇编语言的比较见表1-1。

这里有一个问题：汇编语言程序为人们编写程序、阅读和交流程序都带来了方便，但机器只认识用机器语言编写的目标程序，即0、1代码，换句话说，机器根本就识别不了汇

编语言程序。为了让机器能识别人们用汇编语言编写的程序，在计算机执行每条汇编指令之前，必须将其先翻译成机器语言的目标程序。完成这一功能的程序就是前面提及的汇编程序。

汇编程序的主要功能是把用汇编语言编写的源程序加工成机器语言写的目标程序。汇编过程有以下三步：

① 用汇编语言编写出源程序；

② 将源程序输入到计算机内，由汇编程序把它加工成计算机能执行的目标程序；

③ 执行目标程序，得到计算结果。

表 1-1 机器语言与汇编语言的比较

机器语言程序		汇编语言程序		
地址码	指令代码	地址符号	指令符号	注释
			·LOC 101	定位伪指令，程序从101号地址开始存放
101	020106	START:	LDA 0，CNO1；	X→L0
102	024107		LDA 1，CNO2；	Y→L1
103	107000		ADD 0，1；	X+Y→L1
104	044110		STA 1，CNO3；	Z=X+Y→(CNO3)
105	063077		HALT；	停机
106	X	CNO1：	X；	原始数据
107	Y	CNO2：	Y；	原始数据
110	Z	CNO3：	Z；	计算结果
			·END	程序汇编结束

其中，源程序是由一串符号化的指令组成，当汇编程序加工源程序时，总是从头到尾，一个符号接着一个符号的阅读，称为扫描源程序。从头到尾扫描一次源程序为扫描一遍，一般汇编程序是扫描两遍源程序。第一遍扫描把源程序中所有出现的名字进行造表，确定每个名字将占用的内存位置；第二遍扫描时，按所造出的表，把每条符号化指令变换成数码形式的机器指令。

除此之外，汇编程序还具有查错、修改、打印等功能。当用户编写的汇编语言程序不符合汇编语言所要求的书写格式和语法要求时，汇编程序就会指出源程序的某个位置出了什么性质的错误等，这就是汇编程序的查错功能。汇编程序提供修改源程序的方法简便，用户把修改的要求提供给汇编程序，由汇编程序实现对源程序的自动修改。在汇编过程中，当发现错误时，将错误信息打印出来，必要时可打印出名字表及目标程序，还可打印出修改后源程序的文本等。

实用的汇编语言中，还包括一些用户编程时十分需要的指令，这些指令并无对应的机器指令，这些指令称为伪指令。

众所周知，用高级语言编写的程序，机器是不能直接执行的，需要由编译程序或解释程序将它翻译成对应的机器语言，计算机才能接受并执行。通过编译或解释程序生成的机器语言程序往往比较冗长，占用存储空间较大，执行速度慢。用高级语言编写程序时，程序员无法直接利用机器硬件系统的许多特性，如寄存器、标志位以及一些特殊指令等，影响许多程序设计技巧的发挥。而汇编语言程序是直接利用机器提供的指令系统编写程序的，

与机器语言程序一一对应，所以占用存储空间少，执行速度快，特别是在一些有实时控制要求的场合，与使用汇编语言相比，使用高级语言就不那么方便了，二者的区别主要表现在以下几个方面：

(1) 与汇编语言程序相比，高级语言程序要多占用0.5～1倍的存储单元，执行程序花费的时间要长0.5～2倍。

(2) 汇编语言可直接使用输入/输出指令通过低级接口与外设打交道，而高级语言就不那么方便了。

使用汇编语言编写程序，能充分发挥机器硬件的作用，高效地使用机器。汇编语言是编写计算机通用软件的基础。一般来说，某些对执行时间和存储容量有较高要求的程序，通常使用汇编语言来编制，汇编语言的应用场合是：

- 计算机控制系统(包括实时控制系统)；
- 计算机检测系统；
- 外设驱动程序；
- 智能化仪器仪表；
- 高性能软件等。

1.2.3 学习汇编语言的方法

由于汇编语言是面向机器的语言，是与机器密切相关的一种程序设计语言。所以要掌握汇编语言，使用汇编语言进行程序设计时，就必须对机器的一些特性有所了解，这样才能编出正确的汇编语言程序。概括起来，应了解的机器特性有：

(1) 微处理器的结构(如8086/8088、80386、80486或Pentium等)，存储器的组织方式、存储器的基本单元即编址的基本单位、容量即基本单元的数量是多少。该机存储器的基本单元是什么？是字节还是字、双字等。以相邻字节为单位进行操作的单元有哪些种类？例如，字、半字、双字等。其字长是多少？存储容量多大？地址表示的方法及地址变换机构又是怎样的?是否支持虚拟存储器等。

(2) 控制器的功能及相互关系。如，通用寄存器(包括累加器)的数量、功能、标志位的内容及使用等。

(3) 数据类型及格式。该机器能处理哪些类型的数据？例如，定点数、浮点数、十进制数、字符数据的格式是什么样的？每种数据在机器内的表示方法(即机内格式)又是怎样的？

(4) 指令格式、长度、功能及寻址方式。对机器指令系统的深入了解是汇编语言程序设计的重要任务之一，它包括：指令格式和长度、寻址方式、指令的种类、每一条指令的功能，如何使用这些指令进行程序设计。

(5) 控制器的其他特性。汇编语言程序设计要了解算术逻辑部件和控制器的状态、中断、时钟等。应了解哪些中断与程序设计有关？中断时的现场保护在什么地方？如何屏蔽与开放中断？等等。

(6) 外部设备的特性。外设是计算机与外部世界进行交换信息的手段。进行汇编语言程序设计时，必须了解有哪些外部设备可供使用；每种外设的具体使用方法如何(包括信息交换的机构和交换方式)；如何使外设与算术逻辑部件和控制器协调工作(包括速度匹配)；有