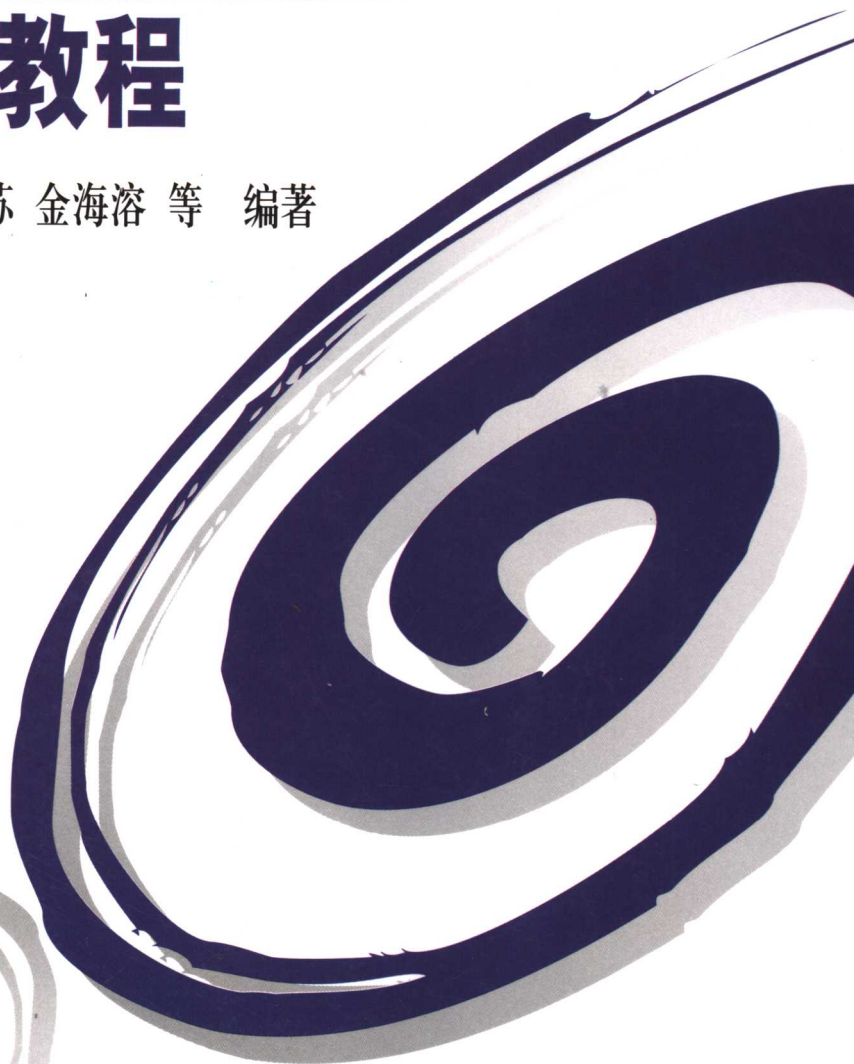




高等院校实验教材 信息技术类

汇编语言程序设计 实验教程

刘均周 苏金海溶 等 编著



科学出版社
www.sciencep.com

高等院校实验教材 信息技术类

汇编语言程序设计实验教程

刘 均 周 苏 金海溶 等 编著

科学出版社

北 京

内 容 简 介

本书结合目前广泛使用的 PC 机 Intel 8086 CPU 来开展教学,目标是通
过一系列使用 80x86 和 Windows 环境下 32 位汇编语言的实验练习,把汇
编语言程序设计的概念和理论知识融入到实践当中,从而加深对汇编语言程
序设计的认识和理解。全书共 21 个实验和 1 个实验总结。每个实验中包含背
景知识介绍和按步骤进行的实验指导等,实验内容的组织充分顾及了不同的
难易程度,富有挑战性。

本书是高等院校计算机、自动化、电子技术及相关专业“汇编语言程序
设计”课程的实验辅助教材,还可作为课程主教材单独使用。本书具有很好
的可读性和可操作性,读者在初步掌握计算机应用和程序设计语言的基础
上,也可选择本书作为进一步学习专业知识的自学教材或读物。

图书在版编目(CIP)数据

汇编语言程序设计实验教程/刘均等编著. —北京:科学出版社,2006
(高等院校实验教材 信息技术类)

ISBN 7-03-018078-X

I. 汇… II. 刘… 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 111884 号

责任编辑:陈晓萍/责任校对:赵 燕

责任印制:吕春眠/封面设计:三函设计

科 学 出 版 社 出版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

铭浩彩色印装有限公司印刷

科学出版社发行 各地新华书店经销

*

2006 年 9 月第 一 版 开本: B5 (720×1000)

2006 年 9 月第一次印刷 印张: 19 1/2

印数: 1—3 000 字数: 377 000

定价: 26.00 元

(如有印装质量问题,我社负责调换(环伟))

销售部电话 010-62136131 编辑部电话 010-62138978-8003

前 言

高等教育的大众化对信息技术类专业课程的教学提出了更高的要求,相关课程的实验教学受到了充分的重视,为了培养适用的专业人才,迫切需要优秀的系列实验教材。

为此,我们邀请长期工作在教学第一线且年富力强的各课程资深专业教师,编写了这套“高等院校实验教材”。本系列教材在实验内容选择、实验步骤设计和实验文档组织等诸方面都做了精心的考虑和安排。首批实验教材所涉及的内容包括操作系统原理、软件工程、多媒体技术、网页设计与网站建设、信息资源管理、电子商务概论、汇编语言程序设计、动态网页技术、数据库技术、数据结构与算法和面向对象程序设计等专业课程,以及 Visual C++、Java、Delphi 等程序设计课程。

作为与课程主教材配套的实验教材,本套教材的编写原则是:依据课程教学大纲,充分理解课程的大多数主教材,遵循课程教学的规律和节奏,充分体现实验的可操作性,既可以与课程主教材辅助配套,也可以单独作为课程主教材使用,还可以是自学的实践教材。本书旨在很好地推动本课程的教学发展,帮助老师教,帮助学生学,帮助用户切实把握本课程的知识内涵和理论与实践的水平。

本书通过一系列使用 Intel 80x86 (MASM) 和 Windows 环境下 32 位汇编语言的实验练习,把汇编语言程序设计的概念和理论知识融入到实践当中,从而加深对 Windows 环境下 32 位汇编语言程序设计的认识和理解。每个实验均留有“实验总结”和“实验评价”部分;每个单元设计了“实验单元的学习评价”;全部实验完成之后的实验总结部分还设计了“课程学习能力测评”等内容,书后则提供了“实验成绩记录”。希望以此方便师生交流对学科知识、实验内容的理解与体会,方便老师对学生实验成绩的记录与管理,以及对学生学习情况进行必要的评估。

吕乐、王文、姜寒、陈晓敏等参加了本书的部分编写工作,本书的编写得到了浙江大学城市学院、浙江工业大学之江学院、温州大学城市学院、浙江商业职业技术学院、吉林农业大学等多所院校领导及师生的大力支持,在此一并表示感谢。

本书相关的实验素材和参考答案等可从科学出版社网站(www.sciencecp.com)的下载区下载。欢迎教师索取为教学配套的相关资料并与笔者进行交流(E-mail: zs@mail.hz.zj.cn; QQ: 81505050)。

周 苏

2006 年 8 月于西子湖畔

读 者 指 南

汇编语言是计算机系统提供给用户的最快、最有效的语言之一，也是能对硬件直接编程的计算机语言。因此，对空间和时间要求很高，或需要直接控制硬件的，都可以使用汇编语言进行程序设计。

本书结合目前广泛使用的 PC 机 Intel 8086 CPU 来开展教学，目的是通过一系列使用 80x86 和 Windows 环境下 32 位汇编语言的实验练习，把汇编语言程序设计的概念和理论知识融入到实践当中，从而加深对汇编语言程序设计的认识和理解。

全书各实验的操作平台采用主流的 Windows 2000/XP Professional 操作系统。大多数实验也同样适用于 Windows 的其他版本。

读 者 对 象

高等院校计算机、自动化、电子技术以及相关专业的“汇编语言程序设计”课程的学生可把此书作为学习的实验辅助教材和自学读物。对于已经具备初步的计算机应用和程序设计知识，并希望通过进一步学习得到提高的读者来说，本书也是一本继续教育的良好读物。

相信本书的实验内容有助于“汇编语言程序设计”课程的教与学，有助于读者对掌握和理解本课程内容建立起足够的信心和兴趣。

实 验 内 容

本书中的实验练习几乎覆盖了汇编语言程序设计教学的各个方面，内容涉及汇编语言编程基础、构建汇编语言编程环境、汇编语言程序设计、输入输出与中断、Win32 汇编语言程序设计，以及汇编语言与 C/C++ 的混合编程等，全书共 21 个实验练习、1 个实验总结和 1 组附录。每个实验都包含背景知识介绍和实验指导等，实验练习的难易程度不同，以帮助读者加深对教材中概念的理解。

实验 1：汇编语言编程基础。包括汇编语言的计算环境、使用 DEBUG 调试工具、用 DEBUG 熟悉指令功能和寻址方式、用 DEBUG 熟悉 Intel 80x86 指令等实验。通过实验理解汇编语言程序设计的基本概念，了解 Intel CPU 的组成，熟悉 8086/8088 寄存器组、8086/8088 存储器组织、8086/8088 标志等与汇编语言程序设计相关的基础概念；了解汇编语言的应用领域和学科，了解汇编语言的发展

和版本；通过因特网搜索与浏览，了解 80x86 汇编语言的计算环境，了解因特网网络环境中主流的汇编语言程序设计技术网站；掌握通过专业网站不断丰富汇编语言程序设计最新知识的学习方法，尝试通过专业网站的辅助与支持来开展汇编语言程序设计的应用实践；初步掌握调试程序 DEBUG 的使用，熟悉 DEBUG 常用命令，特别是 D、R、E、A、U、T、G 等命令的运用；利用调试程序 DEBUG 学习 80x86 CPU 的数据和转移地址寻址方式，学习和掌握 80x86 微处理器的基本指令系统。

实验 2：构建汇编语言编程环境。包括构建 MASM 编程环境、汇编语言程序的格式等实验。通过实验学习建立汇编语言 MASM 实验环境，学习汇编语言程序的编辑、汇编、连接与调试操作；初步了解汇编语言程序设计，掌握汇编语言程序格式，熟悉汇编语言源程序的编辑和汇编、目标模块的连接以及可执行程序的调试。

实验 3：汇编语言程序设计。包括汇编语言简单程序设计、循环程序设计、分支程序设计、子程序设计和宏汇编程序设计等实验。通过实验熟悉汇编语言源程序的编辑和汇编、目标模块的连接以及可执行程序的调试，熟悉 80x86 CPU 的基本指令系统；掌握编写循环处理程序的方法及技巧，了解和掌握程序设计过程中算法的选择；熟悉汇编语言各种跳转指令的功能和用法，学会用程序流程图来分析和设计分支与多分支程序，了解和掌握程序设计过程中算法的选择；了解汇编语言子程序的概念，掌握子程序的定义、调用和返回，掌握设计子程序及过程的基本方法；通过顺序、循环和分支等结构来掌握汇编语言程序设计的主要方法，了解宏汇编、重复汇编、条件汇编和模块化程序设计等汇编语言程序的概念，掌握宏汇编程序设计的基本方法。

实验 4：输入输出与中断。包括输入输出程序设计、中断处理程序编写、常用系统中断——时钟显示、键盘和显示器、使用鼠标的程序设计等实验。通过实验学习和了解汇编语言程序设计中输入输出的相关知识，掌握通过 Windows 的“设备管理器”了解计算机输入输出设备属性的方法，学习直接输入输出端口程序设计方法，在了解扬声器硬件端口及控制原理的基础上，编写简易的声音输出程序；学习和了解汇编语言程序设计中有关中断处理的相关知识，了解软中断的基本编写步骤；通过对汇编语言时钟显示程序的分析，熟悉汇编语言输入输出程序设计的基本概念，熟悉和时间相关的系统中断及调用；掌握在 DOS 和 BIOS 中对屏幕和键盘中断的调用方法，了解对鼠标进行汇编语言程序设计的要求。

实验 5：Win32 汇编语言程序设计。包括 32 位微处理器与 32 位指令集、保护模式下的 32 位汇编语言程序设计等实验。通过实模式/虚拟 86 模式下的 32 位汇编语言程序设计实验，了解 32 位 CPU 结构及存储管理的变化；通过编写简单的 Win32 实验程序，了解和熟悉 Win32 编程环境、Win32 汇编程序结构和开发

过程。

实验 6: 汇编语言与 C/C++ 的混合编程。包括 C/C++ 程序与汇编语言程序的连接、嵌入汇编与模块调用、汇编语言在 Visual C++ 中的应用等实验。通过实验熟悉汇编语言与 C/C++ 语言混合程序设计方法, 了解嵌入式汇编和模块调用汇编方法; 学习汇编语言与 Visual C++ 语言混合编程的知识, 了解 Visual C++ 嵌入式汇编方法和模块调用汇编方法。

附录: 包括 Intel 80x86 指令系统、汇编语言 MASM 伪指令和操作符、中断向量地址一览、DOS 系统功能调用 (INT 21H)、BIOS 功能调用和汇编出错提示信息等内容。

各个实验练习之间的难度不断增加, 从这个意义上讲, 应该在完成前面部分的练习之后再进行后面实验的练习。如果在做练习时遇到了困难, 你可能需要搜索更早的练习来帮助解决问题。

尽管各个实验练习中包含了对关键概念和知识的简要介绍, 但本书毕竟不同于主教材。如果在做这些练习的同时, 学习相关的“汇编语言程序设计”课程, 则可以从实验练习中获取更多的知识。

实验要求

根据不同的教学安排和要求, “汇编语言程序设计”的实验学时数也有所不同。

致教师

汇编语言程序设计的应用面广, 涉及技术领域宽泛, 也被人们赋予了很高的期望值。另一方面, 虽然全部计算机应用技术都有实践性的要求, 但汇编语言程序设计对于应用基础理论来指导开发实践却有着特别的需求。因此, 要让学生真正理解汇编语言程序设计的基础理论知识, 具备将汇编语言技术应用于社会实践的能力, 积极加强汇编语言程序设计课程的实验环节是至关重要的。

作为一本汇编语言程序设计的实验教材, 本书通过提供一组与单元知识密切相关的实验练习作为对汇编语言程序设计主教材的补充, 有助于学生对理论知识的理解, 有助于提高学生的应用和开发能力。

为方便教师对课程实验的组织, 我们在实验内容的选择、实验步骤的设计和实验文档的组织等诸方面都做了精心的考虑和安排。任课教师不需要投入很多精力来设计实验练习。相反, 教师和学生都可以通过本书提供的实验练习来理解概念和实现应用。

根据经验, 虽然部分实验确实能够在一次上机实习课的时间内完成, 但学生普遍存在着两方面的问题:

1) 常常会忽视对每个实验的“背景知识”的阅读和理解, 而一味只求完成实验步骤。

2) 在实验步骤完成之后, 没有时间对实验内容进行消化, 从而不能很好地进行相关的实验总结。

因此, 为保证实验质量, 建议老师重视这两个教学环节的组织, 例如:

1) 要求学生对实验内容进行预习, 并把预习重点放在实验的“背景知识”部分。实验指导老师在实验开始初期对学生的预习情况进行检查, 计入实验成绩。

2) 明确要求学生重视对实验内容的理解和体会, 要求认真完成“实验总结”。为此, 一般不要求当堂完成实验。

例如:

本次实验成绩 (5 分) = 认真预习 (1 分) + 完成实验步骤 (2 分) + 撰写实验总结 (1 分) + 实验总结表达优秀 (1 分)

本书提供了足够数目的实验练习, 教师可以从中选取最适合的部分。我们也希望有机会不断更新本书的版本, 使教师有不断选择新的、更好的实验的机会。

对于那些基础较好的学生, 可以在现有实验的基础上, 在应用实践方面做出一些要求和指导, 以进一步发挥学生的潜能和激发学习的主动性和积极性。

每个实验均留有“实验总结”和“教师评价”部分; 每个单元设计了“实验单元的学习评价”; 全部实验完成之后的实验总结部分还设计了“课程学习能力测评”等内容, 书后则提供了“实验成绩记录”。希望以此方便师生交流对学科知识、实验内容的理解与体会, 方便老师对学生实验成绩的记录与管理, 以及对学生学习情况进行必要的评估。

致学生

对于计算机及其相关专业的学生以及喜欢计算机的一般学生和读者来说, 汇编语言程序设计肯定是需要掌握的重要专业基础知识之一。但是, 单凭课堂教学和一般作业, 要真正领会汇编语言程序设计课程所介绍的概念、原理、方法和技巧等, 是很困难的。

另一方面, 经验表明, 学习尤其是真正体会和掌握汇编语言程序设计的最好方式是对它进行充分的实践。无疑, 通过了解、熟悉和掌握汇编语言开发工具, 是应用汇编语言程序设计技术的重要途径。

本书为读者提供了一个研究汇编语言程序设计的学习方法, 你可以由此来体验汇编语言程序设计的知识及其应用技巧。当然, 通过实验, 也一定能够很好地提高读者的汇编语言程序设计能力。

在开始每一个实验之前, 请务必预习各个实验的“背景知识”部分。“背景知识”是教科书的补充和延伸, 也和实验内容有着密切的联系。完成实验后, 请

认真组织“实验总结”，把感受、认识、意见、建议等表达出来，这能起到“画龙点睛”的作用，也可以和老师进行积极的交流。

实 验 设 备

个人计算机在学生，尤其是专业学生中的普及，使得我们有机会把实验任务分别利用课内和课外时间来完成，以获得更多的锻炼。这样，对实验室和个人计算机的配置就有不同的要求。

实验室设备与环境

用来进行汇编语言程序设计实验的实验室环境，大都需要其计算机设备安装有“命令提示符”环境，部分实验内容需要上网条件。

由于部分实验有可能无法一次完成，有些实验在内容上有一定的互通性和连贯性，所以，实验室设备应能帮助并注意提醒学生妥善保存其实验内容。

个人实验设备与环境

用于汇编语言程序设计实验的个人计算机环境，一般建议安装 Windows 2000 Professional 或 Windows XP Professional 操作系统。个人计算机环境需要为实验准备足够的硬盘存储空间，以方便实验软件的安装和实验数据的保存。

在利用个人计算机完成实验时，要重视理解在操作中系统所显示的提示甚至警告信息，注意保护自己数据和计算环境的安全，做好必要的数据备份工作，以免产生不必要的损失。

由于有些实验在内容上有一定的互通性和连贯性，所以，要注意妥善保存自己的实验内容。

没有设备时如何使用本书

如果本书的读者由于某些客观原因无法获得必要的实验设备时，也不必失望，我们相信您仍将从本书中受益。全书以循序渐进的方式介绍了每个实验的背景知识和实验任务，其中也包含了相当一部分知识内容。读者通过认真阅读“背景知识”，仔细分析实验中给出的实例和程序代码，也能在一定程度上有所收获。

目 录

实验 1 汇编语言编程基础	1
1.1 汇编语言的计算环境	1
1.2 使用 DEBUG 调试工具	9
1.3 用 DEBUG 熟悉指令功能和寻址方式	34
1.4 用 DEBUG 熟悉 Intel 80x86 指令	47
实验 2 构建汇编语言编程环境	75
2.1 构建 MASM 编程环境	75
2.2 汇编语言程序的格式	85
实验 3 汇编语言程序设计	101
3.1 汇编语言简单程序设计	101
3.2 汇编语言循环程序设计	110
3.3 汇编语言分支程序设计	122
3.4 汇编语言子程序设计	137
3.5 宏汇编程序设计	146
实验 4 输入输出与中断	154
4.1 输入输出程序设计	154
4.2 中断处理程序编写	165
4.3 常用系统中断——时钟显示程序	171
4.4 常用系统中断——键盘和显示器中断	178
4.5 使用鼠标的程序设计	191
实验 5 32 位汇编语言程序设计	202
5.1 32 位微处理器与 32 位指令集	202
5.2 保护模式下的 32 位汇编语言程序设计	216
实验 6 汇编语言与 C/C++ 混合编程	228
6.1 C/C++ 程序与汇编语言程序的连接	228
6.2 嵌入汇编与模块调用	238
6.3 汇编语言在 Visual C++ 中的应用	251
实验 7 汇编语言程序设计实验总结	262
7.1 实验的基本内容	262
7.2 实验的基本评价	265
7.3 课程学习能力测评	265
7.4 汇编语言程序设计实验总结	267

7.5 实验总结评价 (教师)	267
实验成绩记录	268
附录	270
附录 1 Intel 80x86 指令系统	270
附录 2 汇编语言 MASM 伪指令和操作符	277
附录 3 中断向量地址一览	279
附录 4 DOS 系统功能调用 (INT 21H)	280
附录 5 BIOS 功能调用	287
附录 6 汇编出错提示信息	291
主要参考文献	298



实验 1

汇编语言编程基础

1.1 汇编语言的计算环境

(实验估计时间: 90 分钟)

1.1.1 背景知识

汇编语言是计算机能够提供给用户使用的最快而又最有效的语言,也是能够利用计算机所有硬件特性并能直接控制硬件的唯一语言。因此,在对程序的空间和时间要求很高的场合,汇编语言的应用是必不可少的。至于很多需要直接控制硬件的应用场合,则更是非用汇编语言不可了。也正因为此,汇编语言程序设计是计算机专业和电子、自动控制等相关专业的重要课程。

所有用计算机语言编写出来的程序,在运行时都在内存中以机器码的形式存储。机器码可以被比较准确地翻译成汇编语言,这是因为汇编语言与机器语言的兼容性最好。所以,几乎所有计算机程序的跟踪、调试工具都是以汇编语言形式出现的。

汇编语言程序直接与硬件打交道。如果你想搞懂程序在计算机中执行时的具体过程,也就是搞清楚计算机的每个组成部分究竟在干什么和怎么干,对于一个真正的硬件发烧友或者开发者来说,汇编语言就是一个重要的工具。

但是,对初学者而言,汇编的许多命令太复杂也很抽象,往往学习很长时间也不一定能写出漂亮的程序,以致影响了我们学习汇编语言的兴趣。确实,写程序不是汇编语言的强项,但这并不妨碍我们可以在学习和熟悉汇编语言的基础上,来尝试应用 DEBUG 工具等,分析和读懂一些与硬件相关的较小的程序,这有时会比完成一个程序更有成就感。某些高深的指令事实上只对有经验的汇编程序员有用。在学习汇编语言的初期,我们可以将注意力集中在少数几个最重要的指令上,例如 CMP、LOOP、MOV、JNZ 等。

1. 计算机组成

计算机系统分成硬件和软件两大部分。

冯氏结构的计算机硬件主要由运算器、控制器、存储器和输入/输出设备构成。20 世纪 70 年代初期, 运算器和控制器开始集成在一个芯片上, 构成了计算机的核心部件, 即中央处理单元 CPU。计算机的存储器又分为主存储器和辅存储器, 输入和输出设备又统称为外部设备(外设)。计算机软件分成系统软件和应用软件。基于微处理器的计算机系统构成如图 1-1 所示。

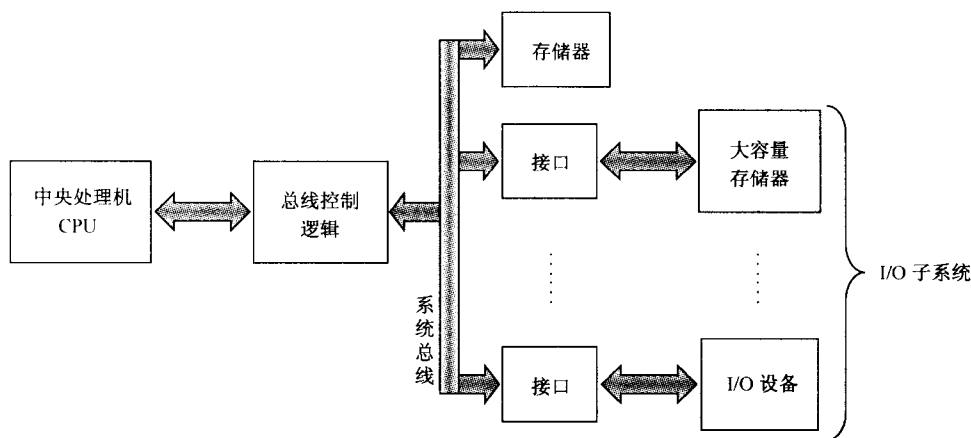


图 1-1 计算机系统构成

汇编语言以英语助记符的形式来表达处理器指令, 例如加法指令为 **ADD**, 转移跳转指令为 **JMP** 等, 是一种面向处理器的低级程序设计语言。利用汇编语言可以编写与计算机硬件密切相关的程序, 以有效地控制硬件; 汇编语言产生的可执行程序容量小、效率高、运行速度快。相对于广泛应用的高级程序设计语言来说, 用汇编语言编写程序比较繁琐, 调试比较困难, 且功能也不够强大。

2. 数据表示

十进制(0~9)是我们习惯使用的数据进制, 二进制(0、1)是计算机内部采用的数据进制, 十六进制(0~9、A~F)则是二进制的缩略形式, 它用 1 位来表达二进制的 4 位。在进行汇编语言程序设计时, 这 3 种进制数据间经常需要相互转换。

对于有符号数据, 计算机内部采用原码、反码和补码表示方法。补码是计算机缺省使用的表达方式。

BCD 码是用二进制编码的十进制数, 经常采用的是 8421 码, 它只使用 0000~1001 这 10 个编码; ASCII 码是美国标准信息交换码, 在其基础的 128 个编码中, 前 32 个和最后一个作为控制字符, 其他 95 个编码表示可显示和打印的字符, 包括数字和大小写英文字母。读者应该熟悉 ASCII 的编码规律。

3. Intel CPU 及其发展

CPU（中央处理器）是可以执行电脑所有算术/逻辑运算与基本 I/O 控制功能的一块芯片。某种汇编语言只能用于特定的 CPU。也就是说，不同的 CPU 的汇编语言指令甚至语法亦不相同。

个人计算机自 1981 年推出至今，其 CPU 发展过程为：Intel 8086→80286→80386→80486→Pentium（奔腾）……，当然还有 AMD、CYRIX 等兼容旁支。一般情况下，后面的产品兼容前面 CPU 的功能，只不过比前面的产品增加了一些新的指令（如奔腾 MMX 的多媒体指令集、迅驰的移动计算指令集等），并且寄存器扩大（如 386 的 32 位 EAX 寄存器）和增加（如 486 的 FS 寄存器）。如果需要确保汇编程序可以适用于各种机型，一般推荐使用 Intel 8086 汇编语言。

CPU 的任务是执行存放在存储器里的指令序列，包括完成算术逻辑操作、担负 CPU 和存储器以及 I/O 之间的数据传送任务等。CPU 中除算术逻辑部件和控制逻辑部件外，还包括高速缓冲存储器、工作寄存器等，如图 1-2 所示。

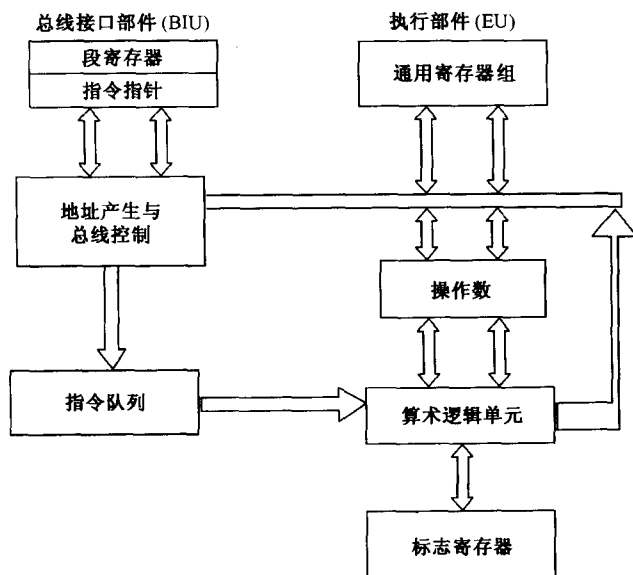


图 1-2 8086 CPU 内部结构示意图

4. 8086/8088 寄存器组

寄存器（Register）是 CPU 内部的元件，所以在寄存器之间进行的数据传送速度非常快。8086 CPU 寄存器的分组如图 1-3 所示。

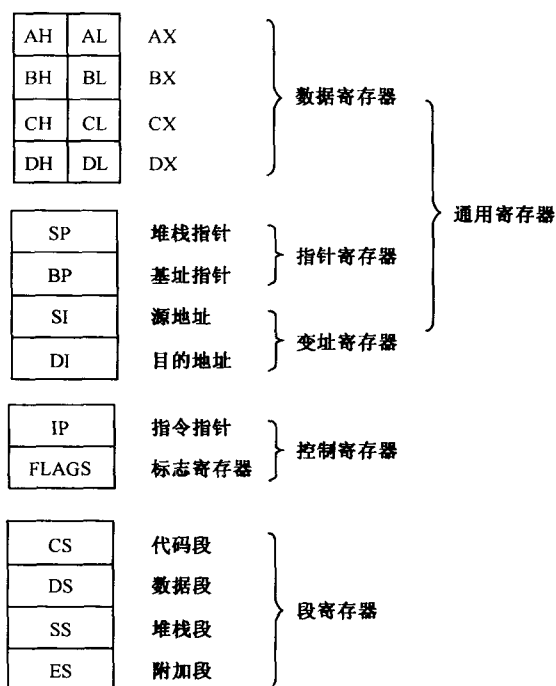


图 1-3 8086 CPU 寄存器分组

寄存器的基本用途是：

- 1) 对寄存器内的数据执行算术及逻辑运算。
- 2) 保存在寄存器内的地址可用来指向内存的某个位置，即寻址。
- 3) 可以用在与计算机的外部设备之间进行数据读写。

8086/8088 的每个寄存器都有其各自的专用目的。8086/8088 的 4 个 16 位数据寄存器是：AX（累加寄存器，常用于运算）、BX（基址寄存器，常用于地址索引）、CX（计数寄存器，常用于计数）、DX（数据寄存器，常用于数据传递）。4 个数据寄存器又可以分成 8 个 8 位通用寄存器，即 AH/AL、BH/BL、CH/CL 和 DH/DL（高 8 位/低 8 位）。

为了有效地运用所有的内存空间，8086/8088 有 4 个 16 位段寄存器，专门用来保存段地址，即 CS、SS、DS 和 ES，依次指明代码段、堆栈段、数据段和附加数据段的首地址。

一个程序在执行时，需要确定其程序代码、数据和堆栈各自要用到内存的哪些位置，通过设定段寄存器 CS、DS、ES、SS 来指向这些起始位置。由于 8086/8088 只支持实模式寻址（即最大段内偏移为 64K），因此，单个数据段和程序段的大小被限制在 64K 以内。一个程序可以定义多个代码段，只是每段不能超过

64K, 所以, 程序代码长度允许超过 64K。但 COM 系统文件不能大于 64K, 因为这种可执行文件不支持段跨越, 也就是它只支持 small 模式。

此外, 还有一些特殊功能的寄存器, 如: IP 指令指针寄存器, 与 CS 配合使用, 指示代码段内指令的偏移地址, 可跟踪程序的执行过程; SP 堆栈指针寄存器, 与 SS 配合使用, 指示堆栈段的当前栈顶, 可指向目前的堆栈位置; BP 基址指针寄存器, 可用作 SS 的一个相对基址位置; SI 源变址寄存器, 可用来存放相对于 DS 段之源变址指针; DI 目的变址寄存器, 可用来存放相对于 ES 段之目的变址指针。还有一个 FLAGS 标志寄存器, 其中有 9 个有意义的标志位。

5. 8086/8088 的存储器组织

内存是计算机在工作中储存信息的地方。内存组织中有许多可存放数值的存储位置(存储单元), 每个存储单元具有一个唯一的编号, 叫“物理地址”。8086 地址总线有 20 位(20 条信号线), 所以 CPU 可以寻址的主存容量是 2^{20} , 即 1MB, 这也是 DOS 的有效控制范围。而 8086 CPU 能做的运算仅限于处理 16 位数据, 即只有 0 到 64K, 所以, 必须用分段寻址才能访问整个内存空间。

在微处理器内部或者在进行程序设计时, 都采用分段的逻辑地址, 表达为“段基址:偏移地址”, 将其转换为物理地址的方法是: “段基址左移 4 位加偏移地址”。

6. 8086/8088 的标志

8086/8088 共有 9 个标志位, 保存在 16 位的 FLAGS 标志寄存器(程序状态字 PSW)中, 分成 6 个状态标志和 3 个控制标志。

状态标志用于提供指令执行结果的辅助信息, 有进位 CF、零位 ZF、符号 SF、奇偶 PF、溢出 OF 和辅助进位 AF 标志等。

控制标志会影响处理器执行指令的方式, 有方向 DF、中断 IF 和单步 TF 标志, 依次控制串操作指令、能否响应可屏蔽中断、是否进入单步执行方式等。

1.1.2 实验目的

1) 理解汇编语言程序设计的基本概念, 熟悉汇编语言、汇编语言程序、汇编等概念的定义和内容。

2) 了解 Intel CPU 的组成, 熟悉 8086/8088 寄存器组、8086/8088 存储器组织、8086/8088 标志等与汇编语言程序设计相关的基础概念。

3) 了解汇编语言的应用领域和学科, 了解汇编语言的发展和版本。

4) 通过因特网搜索与浏览, 了解 80x86 汇编语言的计算环境, 了解因特网

网络环境中主流的汇编语言程序设计技术网站。

5) 掌握通过专业网站不断丰富汇编语言程序设计最新知识的学习方法, 尝试通过专业网站的辅助与支持来开展汇编语言程序设计的应用实践。

1.1.3 工具/准备工作

在开始本实验之前, 请回顾教科书的相关内容。

请联系指导老师或者熟悉汇编语言开发与应用的人士, 了解汇编语言程序设计技术在本专业知识结构链中所处的地位, 在信息技术产品开发中所起的作用, 找找看你周围谁是汇编语言程序设计的高手? 谁在使用汇编语言?

需要准备一台带有浏览器, 能够访问因特网的计算机。

1.1.4 实验内容与步骤

1) 请查阅有关资料, 简单叙述下列概念:

① 汇编语言: _____

② 汇编语言源程序: _____

③ 汇编程序: _____

④ 汇编: _____

2) 请简述:

① 与机器语言相比, 汇编语言有何优缺点?

② 与高级语言相比, 汇编语言有何优缺点?

3) 请简述: 据你了解, 哪些场合需要使用汇编语言?
