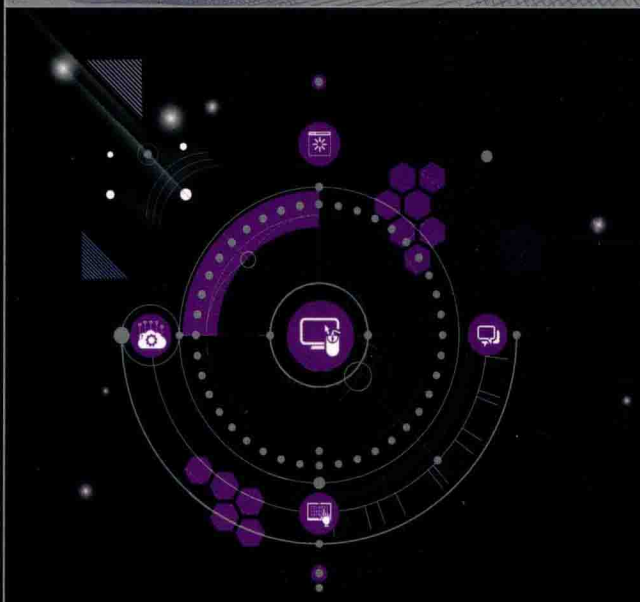
国家级实验教学示范中心联席会计算机学科规划教材
教育部高等学校计算机类专业教学指导委员会推荐教材
面向“工程教育认证”计算机系列课程规划教材

汇编语言教程与实验

◎ 刘军 编著



清華大學出版社

国家级实验教学示范中心联席会计算机学科规划教材
教育部高等学校计算机类专业教学指导委员会推荐教材
面向“工程教育认证”计算机系列课程规划教材

汇编语言教程与实验

◎ 刘军 编著

清華大學出版社
北京

内 容 简 介

汇编语言课程是计算机类专业的一门专业基础课，理论性和实践性非常强。本书将理论教学与实验有机结合，以 8086 CPU 为主，详细介绍汇编语言的基础知识和程序设计方法，主要内容包括：汇编语言基础知识、8086 微型机硬件组织、汇编指令与寻址方式、汇编语言程序格式与数据组织、数据传送程序、算术运算程序、位运算程序、串操作、分支程序设计、循环程序设计、子程序、中断与 DOS 功能调用、宏汇编技术、综合性程序设计案例等。在实验环境上，介绍 DEBUG 和 Masm for Windows 集成实验环境的使用方法。在内容上突出实践教学特色，将实验教学内容贯穿于整个教学过程，每章均附以一定的实验内容。通过多层次的上机实验，加强学生对汇编语言的理解，提高应用编程和程序调试能力。附录部分配有一定量的模拟试题及参考答案，供自我检测使用。

本书可以作为大学本科计算机及相关专业汇编语言课程（含实验环节）的教材或参考书。

图书在版编目(CIP)数据

汇编语言教程与实验/刘军编著. —北京：清华大学出版社，2018
（面向“工程教育认证”计算机系列课程规划教材）
ISBN 978-7-302-47237-7

Ⅰ. ①汇… Ⅱ. ①刘… Ⅲ. ①汇编语言—程序设计—高等学校—教材 Ⅳ. ①TP313

中国版本图书馆 CIP 数据核字(2017)第 122605 号

责任编辑：付弘宇　张爱华
封面设计：刘　键
责任校对：时翠兰
责任印制：刘海龙

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn，http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　邮　　编：100084
社 总 机：010-62770175　　邮　　购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质量反馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
课件下载：http://www.tup.com.cn，010-62795954

印 装 者：三河市金元印装有限公司
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：15.5　　**字　　数**：375 千字
版　　次：2018 年 5 月第 1 版　　**印　　次**：2018 年 5 月第 1 次印刷
印　　数：1～1200
定　　价：39.00 元

产品编号：075047-01

前言

汇编语言作为计算机类专业的一门专业基础课，是微机原理与接口技术等课程的重要基础。通过学习汇编语言，能够加深对计算机组成原理的理解，能够感知、体会和理解机器的逻辑功能，向上为理解各种软件系统的原理打下技术理论基础；向下为掌握硬件系统的原理打下实践应用基础。虽然高级语言和软件开发工具在计算机应用领域发展迅速，但汇编语言在底层编程中仍具有其特有的优势。因此，汇编语言仍然是计算机应用领域非常重要的专业技术基础。

虽然汇编语言对计算机硬件依赖性很强，不同的微处理器具有不同的指令系统，但其编程的基本原理相同。因此，本书采用典型的 8086 微处理器汇编语言编写，对于后继 80x86 机型及其他微处理器的汇编语言（例如：51 单片机系列汇编语言、ARM 系列处理器的汇编语言等）具有借鉴和指导意义。

本书是作者在总结 20 多年教学经验和课程改革的基础上编写而成的，在内容编排上，不同于技术手册，而是按照教学规律精心组织教学内容，层层递进，内容全面，重点突出，知识结构层次清晰，例题与实验操作有机结合，书中所有源程序均利用“Masm for Windows 集成实验环境”上机调试通过。本书并没有详尽论述所有的汇编指令和伪指令，而是有重点地选择典型的基本指令和伪指令，并将其分散到各章教学内容之中，使指令与程序设计结合在一起，使得教学内容更加实用，便于学生实际应用，有效地提高学生学习汇编语言的积极性。

根据“加强基础、培养能力、突出实践”的原则，在教学中采用多元化的授课方式。基础知识部分，以课堂教学为主，强调汇编语言的硬件基础知识和程序的基本格式。在汇编指令与寻址方式中，理论与实验相结合，利用 DEBUG 验证理论分析结果，使学生深刻理解指令寻址的内涵。在程序设计方法上，注重精讲设计程序的思路、采用的基本方法和技巧、使用指令的正确方法，将实践环节贯穿于整个教学过程，每章均安排了一定量的实验内容。同时，注意基本知识的融会贯通，力图建立一个完整的知识体系，避免学生割裂前后知识点间的因果关系。

全书内容共分 14 章。第 1 章介绍汇编语言必备的一些基础知识，重点是激发学生学习汇编语言的积极性，让学生在实验中感受汇编语言的特点。第 2 章主要介绍 8086 微处理器基本结构，重点是 CPU 寄存器和存储器结构，掌握逻辑地址、物理地址、偏移地址的概念，并进一步学习 DEBUG 的使用。第 3 章介绍 8086 CPU 指令系统的概况及常用的寻址方式，重点是与数据有关的寻址方式，详细的指令介绍放在了后续章节中。第 4 章介绍汇编语

言源程序格式和数据组织以及相关的伪指令，同时讲述汇编语言的上机操作过程，这是学习汇编语言编程的基础。第5～7章主要介绍数据传送指令、算术运算指令、逻辑运算和移位操作指令及顺序结构程序设计，将指令介绍与程序设计结合在一起，可以提高学生的学习兴趣，避免教学过于枯燥。第8章介绍串操作指令及其程序设计，要求掌握串操作指令的具体格式、功能和用法，能够编写串操作处理程序。第9章介绍分支程序设计，要求重点掌握条件转移指令的功能、分支程序结构，掌握简单分支程序设计和多分支程序设计的方法。第10章介绍循环控制指令和循环程序设计，要求掌握循环程序的基本结构、程序设计及上机调试方法。第11章介绍子程序的定义及其调用，掌握子程序的设计方法及参数传送方式，重点是不同参数传递的子程序设计方法。第12章介绍中断以及常见的DOS功能调用方法。在实际教学中，视需要可以将简单的1号和2号DOS功能调用提前到前面的章节中介绍，让学生尽早体验到汇编语言的输入输出操作。第13章重点介绍宏功能的使用过程。第14章提供了几个典型的综合性程序设计案例，引导学生开展综合性汇编语言程序设计。附录部分除了介绍实验环境、常用命令等，还提供了模拟试题及参考答案，供教师和学生选用。

本课程的教学总学时建议为48～64课时，可以根据实际的教学需要进行适当增减。具体学时分配如下表所示：

建议学时分配表

章　次	章节名称	总　课　时	授课学时	实验学时
1	基础知识	2	1	1
2	8086微型机硬件组织	4	2	2
3	汇编指令与寻址方式	6	4	2
4	程序格式与数据组织	6	4	2
5	数据传送程序	6	4	2
6	算术运算程序	6	4	2
7	位运算程序	4	2	2
8	串操作	4	2	2
9	分支程序设计	6	4	2
10	循环程序设计	6	4	2
11	子程序	4	2	2
12	中断与DOS功能调用	4	2	2
13	宏汇编技术	2	1	1
14	综合性程序设计案例	4	2	2
合计		64	38	26

为了让老师能较为方便地讲授，本书免费提供所有章节的PPT课件，也提供了书中所有实例的源程序供读者执行和修改。这些配套资料请从清华大学出版社网站 www.tup.com.cn 下载，下载与使用中的相关问题请联系 fuhy@tup.tsinghua.edu.cn。

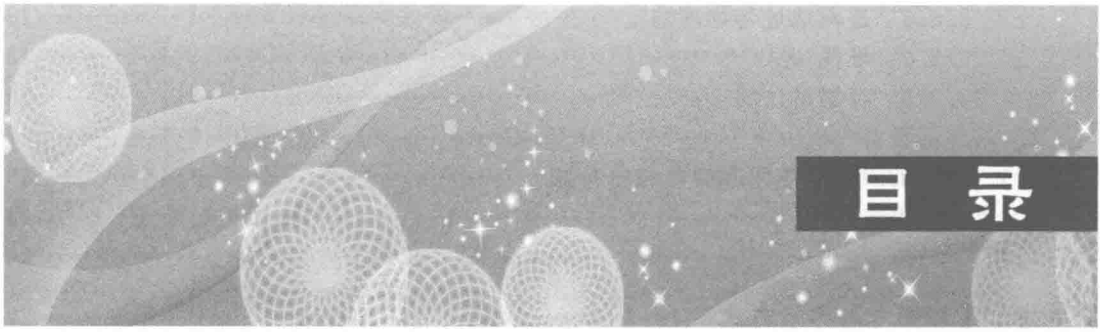
本书是由天津财经大学刘军教授独立编写完成，体现了作者在多年讲授汇编语言课程的过程中积累的宝贵教学经验，是重点课程建设的重要成果。在编写过程中，汲取了国内外

经典的优秀教材之精华，融入了作者的教学体会并精心组织和编排。本书在编写过程中得到了任春明老师、钟家民老师以及各位同行专家的支持，本书的出版工作也得到了清华大学出版社的大力支持，在此表示衷心的感谢。

尽管作者精心编写，但书中难免有疏漏之处，敬请同行专家和读者指正。作者的电子邮箱：liujun@tjufe.edu.cn。

刘 军

2018年1月



目 录

第1章 基础知识

汇编语言是面向硬件的程序设计语言，只有在具备一定计算机基础知识的基础上去学习，才能有效地应用汇编语言编程。因此，本章主要介绍计算机语言的发展以及汇编语言的特点，介绍数制与信息编码等必备的基础知识，并通过 DEBUG 调试工具初步认识汇编语言。

1.1 计算机语言的发展

1.1.1 计算机语言概述

计算机语言从低级到高级发展，经历了机器语言、汇编语言、高级语言几个阶段，每个阶段的计算机语言都有各自的特点。

1. 机器语言

在计算机发展的初期，由于计算机硬件本身的限制，只能识别二进制代码，于是就使用二进制代码构成机器指令来编写程序，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。一条机器指令通常由操作码和操作数两部分组成。其中，操作码指出计算机所进行的具体操作，如加法、减法等；操作数说明操作的对象，如数据或数据存放的地址。机器语言的特点是计算机可直接识别并执行，但依赖于硬件，不同类型机器之间的语言不通用，可移植性差，程序难以理解和调试，指令不便于记忆。

2. 汇编语言

由于机器语言的指令是用二进制代码表示，难以阅读和记忆，给程序设计带来很多困难，于是提出了有助于记忆的符号(或称助记符)来表示二进制代码指令编写程序，这就出现了汇编语言。汇编语言采用助记符表示指令的操作码和操作数，便于阅读、记忆和调试。用汇编语言编写的程序，计算机不能直接识别，必须将其翻译成由机器指令组成的目标程序后，CPU 才能执行，这个翻译过程称为汇编。汇编语言指令与机器语言指令基本上是一一对应的，汇编语言与计算机硬件密切相关，处理器不同，汇编语言就不同。尽管汇编语言还是一种面向机器的语言，不同的 CPU 具有不同的汇编指令，但共性内容是相同的。因此，掌握一种类型的汇编语言，有助于学习其他汇编语言。

3. 高级语言

高级语言类似于自然语言，它采用一组通用的英文单词、数学表达式及规定的符号，按照一定的语法规则和逻辑关系来编写程序。高级语言具有较强的通用性和硬件无关性，是

一种独立于计算机硬件的通用语言。用高级语言编程不必了解和熟悉计算机的指令系统，易学易用。高级语言也要翻译成机器语言才能在计算机上执行，通常有解释型和编译型两种执行方式。

高级语言程序是在未考虑计算机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。

1.1.2 学习汇编语言的必要性

虽然高级语言和软件开发工具在计算机应用领域发展迅速，但汇编语言在底层编程中仍具有其特有的优势。学习汇编语言，有助于深刻理解计算机内部工作机理，更好地对微型计算机系统进行开发和应用。

1. 汇编语言的特点

(1) 汇编语言与计算机硬件密切相关。汇编语言中的指令采用助记符表示，它与机器语言指令基本上是一一对应的，因此它与计算机有着密切的关系，不同类型的 CPU 其汇编语言不同。虽然学习汇编语言比高级语言困难，但它是掌握计算机内部结构的最佳途径。通过汇编语言程序设计，能够更好地理解计算机运行程序的机理，深入了解计算机硬件结构，充分认识计算机。

(2) 汇编语言程序执行效率高。汇编语言是直接在硬件上工作的编程语言，每一条汇编指令都对应着一条机器指令，且汇编语言程序能充分利用计算机硬件特性，如它允许利用寄存器、存储器、标志位等编程。用汇编语言编写的程序在汇编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。在采用相同算法的前提下，任何高级语言程序的效率都远不如汇编语言程序。

2. 汇编语言具有的优势

(1) 能够直接访问与硬件相关的存储器或 I/O 端口。

(2) 能够不受编译器的限制，对生成的二进制代码进行完全的控制。

(3) 能够对关键代码进行更准确的控制，避免因线程共同访问或者硬件设备共享引起的死锁。

(4) 能够根据特定的应用对代码做最佳的优化，提高运行速度。

(5) 能够最大限度地发挥硬件的功能。

(6) 汇编语言用来编制系统软件和过程控制软件，其目标程序占用内存空间少，运行速度快，有着高级语言不可替代的用途。

汇编语言的应用主要是系统程序、效率代码、I/O 驱动程序。某些快速处理、位处理、访问硬件设备等一般都是用汇编语言编写，这正是汇编语言特有的优势。即使在当今计算机语言飞速发展的时代，高级语言也不可能完全替代汇编语言。例如 Linux 内核，虽然绝大部分代码是用 C 语言编写的，但仍然不可避免地在某些关键地方使用了汇编代码。由于这部分代码与硬件的关系非常密切，即使是 C 语言也会显得力不从心，而汇编语言则能够很好地扬长避短，最大限度地发挥硬件的性能。因此，汇编语言仍然是计算机应用领域非常重要的专业技术基础。

1.2　数制与信息编码

1.2.1　数制

按进位方式计数的数制叫作进位记数制，简称进位制。在我们日常生活中，人们习惯于用十进制，即“逢十进一”的数来表示数据。但在计算机内部，数据是以二进制形式表示的，二进制数只有“0”和“1”两个数字，便于用在具有两种稳定状态的物理元件或数字电路中，如双稳态电路来保存二进制信息等。二进制数运算简便，相应的电路设计也十分简单。为了编程者书写数据或输入数据的方便，常用八进制和十六进制。

一个任意的 R 进制数 N，都可写成

$$\begin{aligned} N &= K_n K_{n-1} \cdots K_1 K_0 . K_{-1} K_{-2} \cdots K_{-m} \\ &= K_n \cdot R^n + K_{n-1} \cdot R^{n-1} + \cdots + K_1 \cdot R^1 + K_0 \cdot R^0 \\ &\quad + K_{-1} \cdot R^{-1} + K_{-2} \cdot R^{-2} + \cdots + K_{-m} \cdot R^{-m} \end{aligned} \tag{1-1}$$

式中 m、n 为正整数；R^i 是对应位的位权；R 为 R 进制的基数。所谓基数，就是指在该记数制中每个数位 K_i 可能用到的数字符号的个数，其系数可为 0～(R－1)。每个数位计满 R 后就向高位进位，即“逢 R 进一”，在 R 进制数中相邻两个数位的权相差 R 倍，亦即当小数点向左移一位时，数值缩小 R 倍；而当小数点向右移一位时，数值扩大 R 倍。

基数 R=2 时，为二进制数，权为 2^i，K_i 为 0，1 两个数字中的一个，逢二进位。类似地，当基数 R=8 时，为八进制数，权为 8^i；当基数 R=10 时，为十进制数，权为 10^i；当基数 R=16 时，为十六进制数，权为 16^i。

在使用不同进位记数制的数值时，通常在一个数的末尾用一个标识字母来标识。二进制数用字母 B(Binary)，八进制数用字母 O(Octal)或 Q，十进制数用字母 D(Decimal)，十六进制数用字母 H(Hexadecimal)。如果数的尾部没有任何字母，则计算机接收到的数就默认为是十进制数。例如：101101B，127Q，178D 或 178，3CH 等。表 1-1 列出了常用的几种记数制。

表 1-1　常用的记数制

数　制	基　数	数　码
二进制(Binary)	2	0，1
八进制(Octal)	8	0，1，2，3，4，5，6，7
十进制(Decimal)	10	0，1，2，3，4，5，6，7，8，9
十六进制(Hexadecimal)	16	0，1，2，3，4，5，6，7，8，9，A，B，C，D，E，F

1.2.2　数制之间的转换

由于计算机采用二进制运算，但用计算机解决实际问题时对数值的输入输出通常使用十进制，这就需要有一个十进制向二进制转换或由二进制向十进制转换的过程。在表达数据或存储单元地址时，经常还会用到八进制和十六进制。这些数制之间的转换关系，是学习汇编语言必备的基础知识。

1. *R* 进制数转换为十进制数

对于任意 *R* 进制数，按照式(1-1)写成位权展开式，相加求和，就可以得到对应的十进制数。

【例 1-1】 二进制、八进制、十六进制数转换为十进制数。

$$101.11(B)=1\times2^2+0\times2^1+1\times2^0+1\times2^{-1}+1\times2^{-2}=5.75$$

$$101(O)=1\times8^2+0\times8^1+1\times8^0=65$$

$$101A(H)=1\times16^3+0\times16^2+1\times16^1+10\times16^0=4122$$

2. 十进制数转换为 *R* 进制数

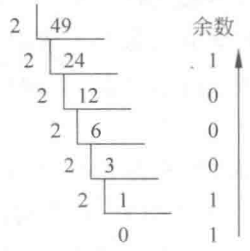
将十进制数转换为 *R* 进制数时，将整数部分和小数部分分别进行转换，然后合并起来即可。

整数部分：采取除以 *R* 取余法，即将十进制整数不断地除以 *R* 取余数，直到商为 0，余数从右到左排列，首次得到的余数在最右面，最后得到的余数在最左面。

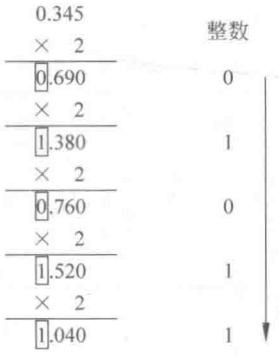
小数部分：采取乘以 *R* 取整法，即将十进制小数不断地乘以 *R* 取整数，直到小数部分为 0 或达到一定精度时为止，整数从左到右排列，首次得到的整数在最左面，最后得到的整数在最右面。

【例 1-2】 十进制数 49.345 转换为二进制数。

(1) 整数部分：除以 2 取余，得到二进制数 110001。



(2) 小数部分：乘以 2 取整，得到二进制小数 0.01011。



(3) 合并结果，即 49.345(D)= 110001.01011(B)。

3. 二进制、八进制、十六进制数之间的转换

由于八进制数、十六进制数与二进制数之间有固定的对应关系：$2^3=8$，$2^4=16$，即一位八进制数可以用 3 位二进制数表示；一位十六制数可以用 4 位二进制数表示，所以可以采用数位对应法进行转换。表 1-2 列出了几种数制之间的对应关系。

表 1-2　不同数制对应关系

十进制	二进制	八进制	十进制	二进制	十六进制	十进制	二进制	十六进制
0	000	0	0	0000	0	8	1000	8
1	001	1	1	0001	1	9	1001	9
2	010	2	2	0010	2	10	1010	A
3	011	3	3	0011	3	11	1011	B
4	100	4	4	0100	4	12	1100	C
5	101	5	5	0101	5	13	1101	D
6	110	6	6	0110	6	14	1110	E
7	111	7	7	0111	7	15	1111	F

1）二进制与八进制数的相互转换

按照数制之间的对应关系，二进制转换为八进制时，以小数点为界向左右两边分组，每 3 位为一组，不足 3 位时补 0；八进制转换为二进制时，每一位八进制数用对应 3 位二进制数代替即可。

【例 1-3】 二进制数 1101101110.10011 转换为八进制数。

001 101 101 110.100 110＝1556.46(O)

1　5　5　6　4　6

【例 1-4】 八进制数 5124 转换为二进制数。

5124(Q)＝101 001 010 100 (B)

2）二进制与十六进制数的相互转换

按照数制之间的对应关系，二进制转换为十六进制时，以小数点为界向左右两边分组，每 4 位为一组，不足 4 位时补 0；十六进制转换为二进制时，每一位十六进制数用对应 4 位二进制数代替即可。

【例 1-5】 二进制数 1101101110.110101 转换为十六进制数。

0011 0110 1110.1101 0100＝36E.D4(H)

3　6　E　D　4

【例 1-6】 十六进制数 2A1D 转换为二进制数。

2A1D(H)＝0010 1010 0001 1101 (B)

1.2.3　二进制数的运算

1. 算术运算

二进制数的算术运算包括加法、减法、乘法和除法四则运算，其运算规则简单易学，分别列举如下。

1）加法规则

0＋0＝0　　0＋1＝1

1＋0＝1　　1＋1＝0(向高位进位 1)

2）减法规则

0－0＝0　　0－1＝1(向高位借位 1)

1－0＝1　　1－1＝0

3）乘法规则

$0\times0=0$　　$0\times1=0$

$1\times0=0$　　$1\times1=1$

4）除法规则

$0\div0$（无意义）　　$0\div1=0$

$1\div0$（无意义）　　$1\div1=1$

2. 逻辑运算

二进制数的逻辑运算是按位进行且相互独立的，基本的逻辑运算有与、或、非三种，还有一种经常使用的是异或运算。下面分别加以简单介绍。

1）逻辑与运算

逻辑与也称逻辑乘，表示当 A、B 事件同时都为真时，结果才为真。其真值表如表 1-3 所示。

表 1-3　与运算真值表

A	B	F=A×B
0	0	0
0	1	0
1	0	0
1	1	1

2）逻辑或运算

逻辑或运算，表示当 A、B 事件只要有一个为真时，结果就为真。其真值表如表 1-4 所示。

表 1-4　或运算真值表

A	B	F=A+B
0	0	0
0	1	1
1	0	1
1	1	1

3）逻辑非运算

逻辑非运算，表示同原事件 A 含义相反。其真值表如表 1-5 所示。

表 1-5　非运算真值表

A	$F=\overline{A}$
0	1
1	0

4）逻辑异或运算

逻辑异或运算，表示当 A、B 事件取值相同时，结果就为假；只有当 A、B 事件取值不同时，结果才为真。其真值表如表 1-6 所示。