

高职高专计算机类规划教材

# 汇编语言程序设计

主编 甘 玲

参编 (以姓氏笔画为序)

张 运

李革新

崔玉玲

主审 李森林



机械工业出版社

本书是高职高专计算机专业汇编语言课程教材, 主要介绍 IBM-PC 及其兼容机汇编语言的基础知识和程序设计方法。全书共分 6 章, 分别介绍了微型计算机系统的组成、8086/8088 微处理器的结构和指令系统、汇编语言程序设计方法和技巧以及汇编语言的应用。书中提供了大量的例题, 每章后都附有习题, 附录中还给出了实验指导。

本书语言精炼、通俗易懂, 叙述由浅入深、循序渐进, 思路清晰、结构严谨。本书除作为高职高专计算机专业汇编语言课程教材外, 也可供普通高等院校计算机专业的学生使用, 同时也可作为工程技术人员自学的参考书。

## 图书在版编目 (CIP) 数据

汇编语言程序设计/甘玲主编. —北京: 机械工业出版社, 2001. 2  
高职高专计算机类规划教材  
ISBN 7-111-08659-7

I. 汇... II. 甘... III. 汇编语言-程序设计-高等学校: 技术学校-教材 N. TP313

中国版本图书馆 CIP 数据核字 (2000) 第 88110 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)  
责任编辑: 王小东 版式设计: 霍永明 责任校对: 张媛  
封面设计: 姚毅 责任印制: 路琳  
北京市密云县印刷厂印刷·新华书店北京发行所发行  
2001 年 2 月第 1 版·第 1 次印刷  
1000mm×1400mm B5·10.375 印张·402 千字  
0 001—5 000 册  
定价: 24.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换  
本社购书热线电话 (010) 68993821、68326677-2527

# 前 言

汇编语言是计算机专业的技术基础课,是操作系统等其它课程的必要先修课,对于培养学生掌握程序设计技术和调试程序都有重要作用。本书以 8086/8088 CPU 为对象介绍汇编语言的基础知识和程序设计方法,这些内容在学习其它类型 CPU 的汇编语言时也完全适用。

为了适应计算机的发展和应用,让更多的人了解计算机的工作原理,本书从应用的角度介绍微型计算机的结构、8086/8088 指令系统和汇编语言程序设计方法。借助汇编语言可以充分调动微型计算机的所有硬件特性并能直接控制硬件,从而让读者进一步弄清程序在计算机中的运行机制,能够在更深层次上对计算机及计算机语言有新的认识。

本书共分 6 章,第一章主要介绍微型计算机系统的组成、微处理器的发展、汇编语言的特点和计算机的运算基础;第二章主要介绍 8086/8088 微处理器系统结构;第三章主要介绍 8086/8088 寻址方式和指令系统;第四章主要介绍汇编语言的语句格式、程序结构、伪指令、宏指令和汇编语言程序的上机过程;第五章主要介绍程序设计的基本步骤,介绍顺序、分支、循环、子程序的程序设计方法和技巧;第六章主要介绍汇编语言在中断调用、DOS 功能调用和 BIOS 功能调用等方面的应用,同时还介绍汇编语言与 C 语言的接口方法。

本书除作为高职高专计算机专业汇编语言课程教材外,也可供普通高等院校计算机专业的学生使用,同时可作为工程技术人员自学的参考书。

本书由重庆工业职业技术学院甘玲主编,后勤工程学院李森林主审,上海机电工业学校张运、中原机械工业学校崔玉玲、陕西省工业职业技术学院李革新参加了编写。重庆工业职业技术学院唐健提出了许多宝贵的意见,在此表示衷心的感谢。

由于编者水平有限,编写时间仓促,书中错漏之处,恳请读者批评指正。

编者

2000 年 10 月

# 目 录

## 前言

|                                     |    |
|-------------------------------------|----|
| <b>第一章 微型计算机基础知识</b> .....          | 1  |
| <b>第一节 微型计算机系统概述</b> .....          | 1  |
| 一、微型计算机系统的组成 .....                  | 1  |
| 二、微处理器的发展进程 .....                   | 5  |
| <b>第二节 汇编语言简介</b> .....             | 6  |
| 一、计算机语言的分类 .....                    | 6  |
| 二、汇编语言的特点 .....                     | 7  |
| <b>第三节 计算机的运算基础</b> .....           | 8  |
| 一、进位计数制及其相互转换 .....                 | 8  |
| 二、机器数 .....                         | 11 |
| 三、常用的名词术语及二进制编码 .....               | 14 |
| 四、数的运算方法 .....                      | 16 |
| 习题一 .....                           | 20 |
| <b>第二章 8086/8088 微处理器系统结构</b> ..... | 21 |
| <b>第一节 8086/8088 微处理器结构</b> .....   | 21 |
| 一、微处理器结构 .....                      | 21 |
| 二、程序执行过程 .....                      | 23 |
| 三、寄存器阵列 .....                       | 23 |
| <b>第二节 8086/8088 存储器组织</b> .....    | 27 |
| 一、存储单元 .....                        | 27 |
| 二、存储单元的地址 .....                     | 27 |
| 三、存储单元的内容 .....                     | 28 |
| 四、存储器地址的分段 .....                    | 28 |
| 五、堆栈结构 .....                        | 30 |
| 六、专用和保留的存储单元 .....                  | 32 |
| 习题二 .....                           | 33 |
| <b>第三章 8086/8088 指令系统</b> .....     | 35 |
| <b>第一节 指令格式和操作数类型</b> .....         | 35 |
| 一、指令格式 .....                        | 35 |
| 二、操作数类型 .....                       | 35 |
| 三、段缺省和段超越 .....                     | 36 |

|                             |            |
|-----------------------------|------------|
| 第二节 寻址方式 .....              | 36         |
| 一、立即寻址方式 .....              | 36         |
| 二、直接寻址方式 .....              | 37         |
| 三、寄存器寻址方式 .....             | 38         |
| 四、寄存器间接寻址方式 .....           | 39         |
| 五、变址寻址方式 .....              | 40         |
| 六、基址变址寻址方式 .....            | 41         |
| 七、串寻址方式 .....               | 43         |
| 八、I/O 端口寻址方式 .....          | 43         |
| 九、隐含寻址方式 .....              | 44         |
| 第三节 8086/8088 指令系统 .....    | 44         |
| 一、数据传送类指令 .....             | 44         |
| 二、算术运算类指令 .....             | 54         |
| 三、逻辑运算与移位类指令 .....          | 68         |
| 四、串操作类指令 .....              | 73         |
| 五、控制转移类指令 .....             | 79         |
| 六、输入/输出类指令 .....            | 92         |
| 七、处理器控制类指令 .....            | 93         |
| 习题三 .....                   | 94         |
| <b>第四章 汇编语言程序设计基础</b> ..... | <b>100</b> |
| 第一节 汇编语言的语句格式 .....         | 100        |
| 一、字符集 .....                 | 100        |
| 二、语句分类 .....                | 100        |
| 三、语句格式 .....                | 101        |
| 第二节 汇编语言程序的基本结构 .....       | 111        |
| 一、汇编语言源程序的一般结构 .....        | 111        |
| 二、段地址的装填 .....              | 112        |
| 三、常用系统功能调用 .....            | 113        |
| 四、源程序代码段模板 .....            | 115        |
| 第三节 伪指令 .....               | 117        |
| 一、数据定义伪指令 .....             | 117        |
| 二、符号定义伪指令 .....             | 120        |
| 三、段定义伪指令 .....              | 122        |
| 四、过程定义伪指令 .....             | 124        |
| 五、模块定义伪指令 .....             | 124        |
| 六、模块通信伪指令 .....             | 125        |
| 七、列表控制伪指令 .....             | 125        |
| 第四节 宏操作指令 .....             | 126        |

|                           |            |
|---------------------------|------------|
| 一、宏定义指令 .....             | 126        |
| 二、取消宏指令 .....             | 128        |
| 三、重复块指令 .....             | 128        |
| 四、带参重复块指令 .....           | 129        |
| 五、串重复块指令 .....            | 129        |
| 六、宏指令与子程序的区别 .....        | 129        |
| 第五节 汇编语言程序的上机过程 .....     | 131        |
| 一、用编辑程序建立源文件 .....        | 133        |
| 二、用汇编程序产生目标代码文件 .....     | 133        |
| 三、用连接程序产生可执行文件 .....      | 136        |
| 四、运行可执行文件 .....           | 137        |
| 习题四 .....                 | 137        |
| <b>第五章 汇编语言程序设计 .....</b> | <b>141</b> |
| 第一节 设计初步 .....            | 141        |
| 一、程序设计的基本步骤 .....         | 141        |
| 二、流程图 .....               | 143        |
| 第二节 顺序程序设计 .....          | 144        |
| 一、查表程序 .....              | 144        |
| 二、两字相乘程序 .....            | 145        |
| 三、拆字程序 .....              | 146        |
| 四、单字节压缩 BCD 数加法运算程序 ..... | 147        |
| 第三节 分支程序设计 .....          | 148        |
| 一、单分支程序设计 .....           | 148        |
| 二、多分支程序设计 .....           | 151        |
| 三、应用程序举例 .....            | 154        |
| 第四节 循环程序设计 .....          | 163        |
| 一、循环程序的引入 .....           | 164        |
| 二、循环程序的结构 .....           | 165        |
| 三、单重循环程序设计 .....          | 166        |
| 四、多重循环程序设计 .....          | 169        |
| 五、应用程序举例 .....            | 171        |
| 六、“位”控制循环程序 .....         | 178        |
| 第五节 子程序设计 .....           | 180        |
| 一、子程序 .....               | 180        |
| 二、子程序结构 .....             | 183        |
| 三、子程序段内调用和返回 .....        | 185        |
| 四、子程序段间调用和返回 .....        | 190        |
| 五、保存与恢复寄存器 .....          | 192        |

|                              |            |
|------------------------------|------------|
| 六、主程序与子程序之间的参数传递 .....       | 193        |
| 习题五 .....                    | 202        |
| <b>第六章 汇编语言的应用</b> .....     | <b>205</b> |
| 第一节 输入输出程序设计 .....           | 205        |
| 一、输入输出过程 .....               | 205        |
| 二、查询输入输出方式 .....             | 206        |
| 三、中断输入输出方式 .....             | 209        |
| 四、直接数据通道传送 (DMA) 方式 .....    | 210        |
| 五、应用举例——使 PC 机发声 .....       | 211        |
| 第二节 中断系统 .....               | 213        |
| 一、概念 .....                   | 213        |
| 二、8086/8088 的中断系统 .....      | 215        |
| 第三节 DOS 功能调用 .....           | 225        |
| 一、概述 .....                   | 225        |
| 二、常用 DOS 功能调用 .....          | 226        |
| 三、文件管理模块 .....               | 231        |
| 第四节 BIOS 功能调用 .....          | 237        |
| 一、概述 .....                   | 237        |
| 二、常用 BIOS 功能调用 .....         | 237        |
| 三、图形显示程序设计 .....             | 240        |
| 四、键盘 I/O .....               | 246        |
| 第五节 汇编语言与 C 语言的接口 .....      | 251        |
| 一、Turbo C 与汇编语言的接口方法 .....   | 251        |
| 二、自动产生汇编语言的框架程序 .....        | 254        |
| 三、编译、连接、运行接口程序 .....         | 258        |
| 习题六 .....                    | 258        |
| <b>附录</b> .....              | <b>260</b> |
| 附录 A 8086/8088 指令系统一览表 ..... | 260        |
| 附录 B 指令对标志的影响 .....          | 266        |
| 附录 C DEBUG 命令一览表 .....       | 267        |
| 附录 D DOS 系统调用 .....          | 269        |
| 附录 E BIOS 中断功能调用 .....       | 273        |
| 附录 F 汇编程序出错信息 .....          | 276        |
| 附录 G 实验指导 .....              | 286        |
| 实验一 DEBUG 的使用 .....          | 286        |
| 实验二 汇编语言程序的基本操作 .....        | 282        |
| 实验三 顺序程序设计 .....             | 287        |
| 实验四 分支程序设计 .....             | 288        |

|             |                     |            |
|-------------|---------------------|------------|
| 实验五         | 循环程序设计 .....        | 297        |
| 实验六         | 子程序设计 .....         | 303        |
| 实验七         | I/O 程序设计 .....      | 310        |
| 实验八         | 汇编语言与 C 语言的接口 ..... | 318        |
| <b>参考文献</b> | .....               | <b>322</b> |

# 第一章 微型计算机基础知识

计算机的产生、发展和应用,带着人类在科学道路上迅猛前进。自1946年第一台电子计算机问世后,其发展速度相当惊人,随着主要电子元件从电子管、晶体管、中小规模集成电路发展到大规模集成电路,70年代初研制出了第一台微型计算机。众所周知,最初的电子计算机只是作为一种现代化的计算工具,而微型计算机的出现为计算机的广泛应用开拓了极其广阔的前景,展示了它在信息社会中日益显要的地位,特别是微型计算机的科技水平、生产规模和应用程度已成为衡量一个国家现代化水平的重要标志。微型计算机的发展和应用已引起全社会的关注,并已转化成巨大的推动社会前进的生产力。

微处理器是微型计算机的“心”,指挥这颗“心”跳动的是它的指令系统,是人们赋予它的程序,汇编语言是与微处理器最密切的程序设计语言。本章将介绍微型计算机系统的组成、汇编语言的特点和计算机的运算基础。

## 第一节 微型计算机系统概述

计算机系统由硬件系统(Hardware)和软件系统(Software)两部分组成。微型计算机系统也是由硬件系统和软件系统组成,只不过它的硬件大部分集中放在主机箱中。

### 一、微型计算机系统的组成

#### (一) 硬件系统

计算机硬件系统由运算器、控制器、存储器、输入/输出设备五大部分组成,其中运算器和控制器构成中央处理器CPU(Central Processor Unit),中央处理器和存储器(Memory)构成主机。

微型计算机硬件系统由主机和外围设备构成,主机由微处理器(Microprocessor)、存储器和输入/输出(I/O)接口三个部分组成,各部分由系统总线把它们连接在一起,系统总线分为地址总线、控制总线和数据总线。主机的结构如图1-1所示。

#### 1. 主机

微型计算机硬件系统的核心是主机,主机的核心部件是微处理器。

(1) 微处理器 微处理器微缩在一片或几片大规模集成电路芯片上,它的任务是处理存放在存储器中的程序指令:从存储器中取出指令,进行译码,执行算术运算和

逻辑运算，再存放数据，并控制整个微型计算机自动、协调地完成程序功能。

(2) 存储器 存储器是微型计算机的记忆部件，类似于人的大脑。主机中的存储器称为内存储器，简称内存。内存由一个个存储单元组成，每个存储单元都有一个地址，人们编写的程序及程序中所用到的数据、编码信息及中间结果在程序运行过程中一般放在内存中。内存分为只读存储器

ROM (Read-Only Memory) 和随机读写存储器 RAM (Random Access Memory) 两类。ROM 用于存放起动、自检、设备驱动程序等永久性信息；RAM 中存放的信息读出时不变，写入时新内容将取代原来的内容，断电后其中所有的内容都将丢失。

(3) 输入/输出接口 输入/输出接口是主机与外围设备之间通信的桥梁，由寄存器和译码器组成。微处理器有自己的内部特性，用户不能更改；输入/输出设备也有自己的特性，其工作速度比微处理器慢得多，因此，要使两者结合起来协调工作，就需要适当的接口。输入/输出接口的任务是处理主机与外围设备之间的数据传送，把外围设备的状态传入主机，接收主机发出的各种控制信号，控制外围设备执行操作。

## 2. 外围设备

外围设备一般包括输入/输出设备和大容量存储器两类设备。输入/输出设备是指负责计算机进行通信的外围设备，如键盘、显示器、打印机等。大容量存储器则是指可存储大量信息的外存储器，如磁盘、磁带、光盘等，简称外存。由于内存的容量有限，所以计算机使用外存作为内存的后援设备，外存的容量一般比内存大得多，但从外存中存取信息的速度比从内存中存取信息的速度慢得多。所以除了必要的系统程序外，一般的程序是存放在外存中，要运行该程序时，才把它从外存传送到内存的某个区域，然后由微处理器控制执行。

## 3. 系统总线

系统总线把微处理器、存储器、输入/输出接口和外围设备连接起来，用来传送它们之间的信息。系统总线包括数据总线、地址总线和控制总线。数据总线负责传送数据，数据包括指令代码、原始数据、中间数据和结果数据；地址总线用来指出数据的来源地和目的地；控制总线负责控制总线的动作，传送微处理器对

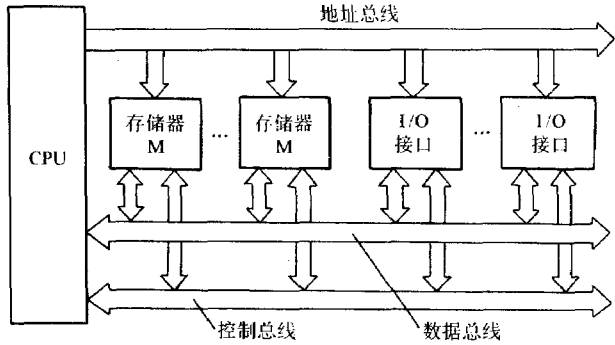


图 1-1 主机的基本组成

存储器或输入/输出设备的控制命令和输入/输出设备对微处理器 CPU 的请求信号。系统总线的工作由总线控制逻辑负责指挥。

由于构成微型计算机硬件系统的各个部件，如：机箱、主机板、电源、软盘驱动器、硬盘驱动器以及显示卡、软盘/硬盘卡和串行、并行输入/输出卡等都被设计成了模块结构，它们的机械安装尺寸约定为统一规格，并采用不易接错的插、接口方式连接。因此，完全可以用世界各地不同厂家生产的微型计算机部件组装成一台合格的兼容微型计算机。微型计算机发展过程中的这一开放性举措是相当重要的，它打破了个别生产厂家的垄断，推动了大规模生产技术的发展，使硬件系统的价格逐步降低，为微型计算机步入家庭成为真正意义上的“个人计算机”奠定了坚实的基础。

## （二）软件系统

微型计算机的软件系统是由系统软件和应用软件两大部分组成。系统软件是厂家提供给用户、帮助用户编写和调试应用程序的程序集合，它包括如下内容：操作系统、监控程序、输入/输出驱动程序、文本编辑程序、编译和解释程序、调试程序、连接定位程序、数据库管理系统等；应用软件则是为解决某一个实际问题而编写的程序集合，如：科学计算程序、数据处理程序、企业管理程序、电算化软件等。

### 1. 操作系统

操作系统 (Operating System) 是系统软件的指挥中枢，它的主要作用是统一管理微型计算机的所有资源，包括微处理器、存储器、输入/输出设备以及其它的系统软件和应用软件。用户在使用微型计算机时，无需过问系统中各个资源的分配和使用情况，也不必为各种输入/输出设备编制设备驱动程序，用户只需要正确使用操作系统提供的各种命令和系统调用功能，就可以使应用程序在操作系统的控制下自动而协调地运行。目前微型计算机常用的操作系统有：MS-DOS、Windows 3.X、Windows 95/98、Windows 2000、Windows me、Windows NT、Netware、Macintosh、OS/2Wrap、Unix、Linux 等。

操作系统的主要部分是常驻内存的监控程序。只要一开机，监控程序就驻留内存，通过键盘接受用户的命令，从而控制操作系统执行相应的操作。

### 2. 输入/输出驱动程序

输入/输出驱动程序是用来对输入/输出设备进行管理和控制。当系统程序或应用程序需要使用输入/输出设备时，通过调用输入/输出驱动程序对相应的设备发出命令，从而完成微处理器和输入/输出设备之间的信息传送。

### 3. 文本编辑程序

文本编辑程序是用来输入和编辑文本并将其存入存储器中。文本是指由数字、字母、符号等信息所组成的文件，它可以是一个用汇编语言或高级语言编写的程

序，也可以是一组数据或一份报告。例如 MS-DOS 中的 EDIT 就是一个能方便地输入和编辑文本的全屏幕文本编辑程序。

#### 4. 编译和解释程序

编译和解释程序是把人们编写的程序翻译成微型计算机能识别的二进制代码的一种系统程序。编译程序 (Compiler) 方式是先把高级语言程序翻译成机器语言程序，然后再执行；而解释程序 (Interpreter) 方式则是一边翻译一边执行。

#### 5. 调试程序

调试程序是系统提供给用户的能监控用户程序的一种工具，例如 MS-DOS 中的 DEBUG，它可以建立、修改、显示和逐条执行一个程序。通常，简单的汇编语言程序可以通过 DEBUG 来建立、修改和执行。

#### 6. 连接定位程序

连接定位程序 (LINK) 是用来把要执行的程序与库文件或其它已经翻译的子程序连接起来，形成微型计算机能够运行的执行文件。

概括起来，微型计算机的硬件系统就像是人的躯壳，而软件就像是人的灵魂，两者相互依存、缺一不可。微型计算机系统的组成如图 1-2 所示。

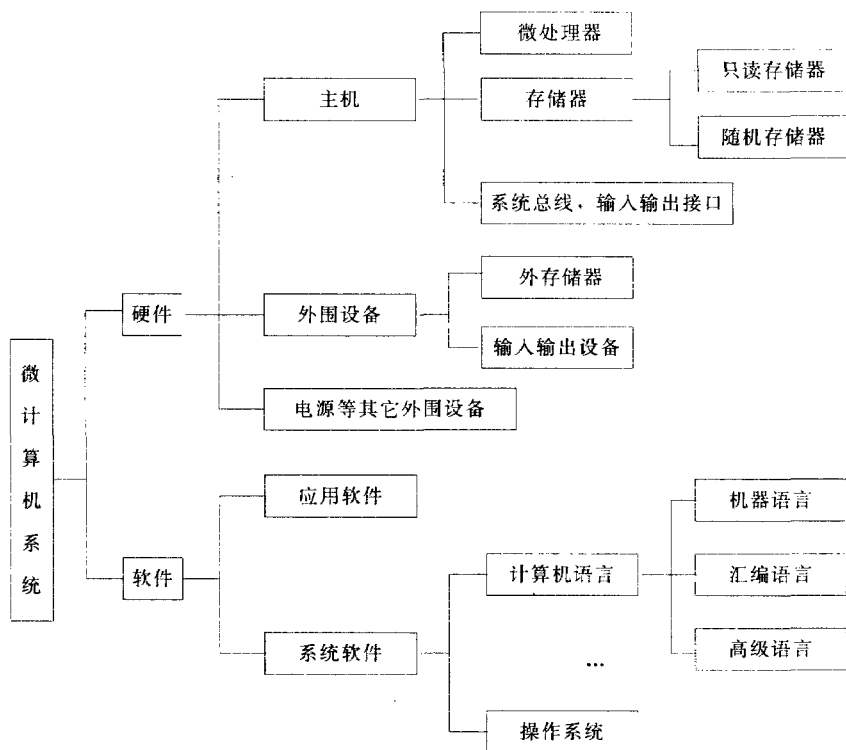


图 1-2 微型计算机系统的组成

## 二、微处理器的发展进程

微处理器的历史可追溯到 1971 年，在这短短的近 30 年里，微处理器的发展日新月异，产品像潮水般涌向市场，推动着全球经济的大发展。下面以 Intel 公司的 80x86 系统为例介绍微处理器的发展历史。

1971 年，美国旧金山南部的 Intel 集成电子产品公司推出了世界上第一代微处理器 4004，它是 4 位微处理器，每个芯片上含有 2300 个晶体管，支持 45 条指令。从此以后，Intel 便与微处理器结下了不解之缘。

1973 年，Intel 公司推出了第二代微处理器 8080，它是 8 位微处理器，每个芯片上含有 4900 个晶体管，运算速度提高了一个数量级，指令系统比较完善，寻址能力有所增强。

以后微处理器的发展进程如表 1-1 所示。

表 1-1 微处理器发展历史简表

| CPU                   | 推出年份        | 主频/MHz  | CPU 插槽类运            | 内部<br>cache/KB | 外部<br>cache                  | 集成<br>晶体管数 | 总线<br>宽度 |
|-----------------------|-------------|---------|---------------------|----------------|------------------------------|------------|----------|
| 8086                  | 1978 年 6 月  | 4.77    | 焊接在主板上              | 无              | 无                            | 2.9 万      | 16 位     |
| 8088                  | 1979 年 6 月  | 4.77    | 焊接在主板上              | 无              | 无                            | 2.9 万      | 8 位      |
| 80286                 | 1982 年 2 月  | 6~20    | 焊接在主板上              | 无              | 无                            | 13.4 万     | 16 位     |
| 80386                 | 1985 年 10 月 | 12.5~33 | 焊接在主板上              | 8              | 有                            | 27.5 万     | 32 位     |
| 80486                 | 1989 年 4 月  | 25~100  | Socket3<br>(238 插孔) | 8              | 有                            | 120 万      | 32 位     |
| Pentium<br>(P54C)     | 1993 年 3 月  | 60~66   | Socket4<br>(320 插孔) | 16             | 有                            | 310 万      | 32 位     |
|                       | 1993 年 6 月  | 75~200  | Socket5<br>(320 插孔) | 16             | 有                            | 330 万      | 32 位     |
| Pentium Pro           | 1995 年 11 月 | 133~200 | Socket8<br>(387 插孔) | 16             | CPU 内集成<br>256~512KB         | 550 万      | 32 位     |
| Pentium MMX<br>(P55C) | 1997 年 1 月  | 166~233 | Socket7<br>(321 插孔) | 32             | 有                            | 450 万      | 32 位     |
| Pentium II            | 1997 年 5 月  | 233~450 | Slot1 插槽            | 32             | 与 CPU、Tag<br>RAM 一起集<br>成在板上 | 750 万      | 64 位     |

## 第二节 汇编语言简介

### 一、计算机语言的分类

计算机语言种类很多，按语言层次可分为：机器语言、汇编语言和高级语言。下面举例说明分别用三种语言编制的程序。

**例 1-1** 已知  $a=2$ 、 $b=6$ 、 $c=3$ ，求  $Y=a+b-c$  的值并显示结果。

#### (一) 机器语言

机器语言是计算机能直接识别的程序设计语言，由二进制代码组成，因此这种语言的可读性很差，使用这种语言编写程序很不方便。

实现例 1-1 的机器语言程序如下：

```
B8 02 00
05 06 00
2D 03 00
8B D0
80 CA 30
B4 02
CD 21
B4 4C
CD 21
```

#### (二) 汇编语言

汇编语言是一种面向机器的程序设计语言，是机器语言符号化的描述。通常，汇编语言的执行语句与机器语言的指令是一一对应的。相对机器语言来说，汇编语言一般用与操作含义相同的英语单词或单词缩写来作为指令操作符，因而可读性较好，但是编写比较复杂的程序还是很困难。

实现例 1-1 的汇编语言程序如下：

```
SSEG    SEGMENT STACK 'STACK'
        DB 64 DUP (?)
SSEG    ENDS
CSEG    SEGMENT
        ASSUME CS: CSEG, SS: SSEG
START:  MOV    AX, 2
        ADD   AX, 6
        SUB   AX, 3
        MOV   DX, AX
```

```

OR      DL, 30H
MOV     AH, 2
INT     21H
MOV     AH, 4CH
INT     21H
CSEG   ENDS
END     START

```

### (三) 高级语言

高级语言是最接近于人类自然语言的程序设计语言。由于高级语言脱离了机器指令，因而可读性好，但同时也造成高级语言对硬件的控制能力很弱，此外生成执行代码效率较低。当前广泛使用的高级语言有：C、FoxPro、Pascal、Fortran、Cobol、Basic 等，下面以 C 语言为例说明。

实现例 1-1 的 C 语言程序如下：

```

#include <stdio. h>
int main ( )
{
    int a, b, c, y;
    a=2;
    b=6;
    c=3;
    y=a+b-c;
    printf ("%d\n", y);
    return 0;
}

```

## 二、汇编语言的特点

由于微型计算机的迅速发展，特别是它在控制、网络与通信等方面的广泛应用，使汇编语言程序设计技术不仅为系统程序员使用，而且为广大微型计算机用户所普遍使用。

汇编语言是计算机提供给用户的最快而且又最有效的语言，它能透彻地反映、巧妙充分地运用计算机的硬件功能及特点，便于编程人员根据自己的需要灵活地编制各种程序，控制计算机的运行。在对运行速度要求很高的场合，汇编语言是必不可少的，如操作系统、编译程序等，多数是用汇编语言编写的。

汇编语言是机器语言的符号化表示，因此，汇编语言能产生最快和最有效的代码。通常情况下，用汇编语言编写的程序生成的执行代码长度只有相应用 C 语言编写的程序生成的执行代码长度的 1/10。

通常把用汇编语言编写的程序称为汇编语言源程序,在本书中简称源程序。汇编语言源程序与高级语言程序一样,必须由翻译程序翻译成代码程序,才能被计算机认识,此代码程序称为源程序的目标程序。将源程序翻译成目标程序的翻译程序通常称为汇编程序,将源程序翻译成目标程序的过程通常称为汇编。IBM-PC机有两个汇编程序:小汇编程序ASM和宏汇编程序MASM, MASM的功能比ASM强,它支持宏汇编,所以建议使用MASM。

### 第三节 计算机的运算基础

#### 一、进位计数制及其相互转换

##### (一) 各种进位计数制的特点

按进位的原则进行计数的方法称为进位计数制,简称进位制。日常生活中多用十进制,而在计算机中则采用二进制。由于二进制不易书写和阅读,所以又引入了八进制和十六进制。

##### 1. 十进制

十进制的特点是:用0~9十个不同的数字符号,且按逢10进位的原则进行计数。例如:827.31,小数点左边的第1位代表个位,它的值是 $7 \times 10^0 = 7$ ;左边第2位代表十位,它的值是 $2 \times 10^1 = 20$ ;左边第3位代表百位,它的值是 $8 \times 10^2 = 800$ ;小数点右边第1位是十分位,它的值是 $3 \times 10^{-1} = 0.3$ ;右边第2位是百分位,它的值是 $1 \times 10^{-2} = 0.01$ 。十进制数827.31的展开式为:

$$827.31 = 8 \times 10^2 + 2 \times 10^1 + 7 \times 10^0 + 3 \times 10^{-1} + 1 \times 10^{-2}$$

其中“10”称为十进制的“基数”,“ $10^2$ 、 $10^1$ 、 $10^0$ 、 $10^{-1}$ 、 $10^{-2}$ ”称为十进制数各相应位的“权”。每一位的值等于该位数字与该位权的乘积,各位值的累加和表示整个数的大小。

##### 2. 二进制

二进制的特点是:用0和1两个不同的数字符号,且按逢2进位的原则进行计数。例如1101.01,与十进制类似有展开式:

$$\begin{aligned} (1101.01)_2 &= 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} \\ &= 13.25 \end{aligned}$$

其中“2”称为二进制的“基数”,“ $2^3$ 、 $2^2$ 、 $2^1$ 、 $2^0$ 、 $2^{-1}$ 、 $2^{-2}$ ”称为二进制数各相应位的“权”。二进制各位的值也是各位数字与该位权的乘积,各位值的累加和表示整个数的十进制大小。

##### 3. 八进制

八进制的特点是:用0、1、…、7八个不同的数字符号,且按逢8进位的原则进行计数。“8”是八进制的“基数”,8的各次幂是八进制各位的“权”。八进制

的求值方法类似于二、十进制，例如：

$$(314)_8 = 3 \times 8^2 + 1 \times 8^1 + 4 \times 8^0 = 204$$

#### 4. 十六进制

十六进制的特点是：用 0、1、…、9、A、B、C、D、E、F 十六个不同的数字符号，且按逢 16 进位的原则进行计数。“16”是十六进制的“基数”，16 的各次幂是十六进制各位的“权”。十六进制的求值方法类似于二、八、十进制。例如：

$$(6AE)_{16} = 6 \times 16^2 + A \times 16^1 + E \times 16^0 = 1536 + 160 + 14 = 1710$$

二、八、十、十六进制数可分别在数字后面加后缀 B、Q（或 O）、D、H 表示。例如：1011B、27Q、16D、2AH 分别表示二进制数 1011、八进制数 27、十进制数 16、十六进制数 2A。十进制数经常省略后缀 D。

### (二) 不同进位制之间的转换

#### 1. 二进制与十进制之间的互换

(1) 二进制转换为十进制 由前述可知，只需将二进制数每一位的数字与该位的权相乘，再将各位的数值加在一起就得到了相应的十进制数。

$$\begin{aligned} \text{例 1-2} \quad 1101011.01\text{B} &= 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^3 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-2} \\ &= 64 + 32 + 8 + 2 + 1 + 0.25 \\ &= 107.25 \end{aligned}$$

#### (2) 十进制转换为二进制

1) 十进制整数转换为二进制整数的方法是：用 2 不断地去除要转换的十进制数，直至商为 0。每次的余数即为二进制数位，最初得到的余数是二进制整数的最低位。这就是所谓的“除 2 取余”法。

**例 1-3** 将十进制数 357 化成二进制数。

$$\begin{array}{r} 2 \overline{) 357} \\ 2 \overline{) 178} \dots\dots\dots 1 \\ 2 \overline{) 89} \dots\dots\dots 0 \\ 2 \overline{) 44} \dots\dots\dots 1 \\ 2 \overline{) 22} \dots\dots\dots 0 \\ 2 \overline{) 11} \dots\dots\dots 0 \\ 2 \overline{) 5} \dots\dots\dots 1 \\ 2 \overline{) 2} \dots\dots\dots 1 \\ 2 \overline{) 1} \dots\dots\dots 0 \\ 0 \dots\dots\dots 1 \end{array}$$

因此：357 = 101100101B

2) 十进制小数转换为二进制小数的方法是：用 2 不断地去乘要转换的十进制小数，直至乘积的小数部分为 0。每次所得的整数部分即为二进制数位，最初得到