

高校计算机教学系列教材

汇编语言程序设计

孙卫真 饶 敏 杨西珊 编著



北京航空航天大学出版社

高校计算机教学系列教材

汇编语言程序设计

孙卫真 饶 敏 杨西珊 编著

北京航空航天大学出版社

内容简介

本书以 Intel8086/8088 指令系统和宏汇编语言为主体。全面而系统地介绍了 16 位微处理器的汇编语言程序设计方法。前 3 章为基础,讲解 16 位基本整数指令及其汇编语言程序设计的知识;汇编语言程序设计的基础知识、IBM-PC 的寻址方式和指令系统、汇编语言程序格式。中间 4 章为程序设计主体,从不同的方面介绍了汇编语言程序设计的基本内容和实际应用知识,结合实际应用,深入浅出地分析了程序设计的组合单元:顺序程序设计、分支程序设计、循环程序设计和子程序设计。最后 1 章为提高部分,主要有汇编语言与 C/C++ 及其它高级语言的混合编程。各章配有丰富的习题和详细的上机指导,附录提供了 Debug 等工具的使用说明。本书特点:内容精炼、语言通顺易懂、结构清晰、实例丰富。适合高校本、专科计算机、电子、自动控制等专业作教材,也适合计算机应用开发人员作参考书。

图书在版编目(CIP)数据

汇编语言程序设计/孙卫真编著. —北京:北京航空航天大学出版社,2003.12

ISBN 7-81077-403-4

I. 汇… II. 孙… III. 汇编语言—程序设计
IV. TP313

中国版本图书馆 CIP 数据核字(2003)第 092398 号

汇编语言程序设计

孙卫真 饶 敏 杨西珊 编著

责任编辑 许传安

*

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(100083) 发行部电话:(010)82317024 传真:(010)82328026

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

北京市云西华都印刷厂印装 各地书店经销

*

开本:787×1092 1/16 印张:14.5 字数:371 千字

2003 年 12 月第 1 版 2003 年 12 月第 1 次印刷 印数:5 000 册

ISBN 7-81077-403-4 定价:20.00 元



总 前 言

科教兴国,教育先行,在全国上下已形成共识。在教育改革过程中,出现了多渠道、多形式、多层次办学的局面。同时,政府逐年加大教育的投入力度。教育发展了,才能有效地提高全民族的文化、科学素质,使我们中华民族屹立于世界民族之林。

计算机科学与技术的发展日新月异,其应用领域迅速扩展,几乎无处不在。社会发展的需求,促使计算机教育生气蓬勃。从普通高校的系统性教学,到远距离的电视、网上教学;从全面讲述,到不同应用领域的、星罗棋布的培训班;从公办的到民办的;从纸介教材到电子教材等等,可以说计算机教学异彩纷呈。要进行教学,就必须有教材。

面对我们这么大的国家和教学形势,在保证国家教学基本要求的前提下,应当提倡教材多样化,才能满足各教学单位的需求,使他们形成各自的办学风格和特色。为此,我们组织北京工业大学、北京航空航天大学、北京理工大学、南开大学、天津工业大学等高校的有丰富教学经验的教师编写了计算机教学系列教材,将陆续与师生见面。

系列教材包括以下各项。

(一) **基础理论**:离散数学。

(二) **技术基础**:电路基础与模拟电子技术;数字逻辑基础;计算机组成与体系结构;计算机语言(拼盘,选择使用),包括 C++ 程序设计基础、Visual Basic 程序设计基础、Matlab 程序设计基础、Java 程序设计基础、Delphi 语言基础、汇编语言基础等;数据结构;计算机操作系统基础;计算方法基础;微机与接口技术;数据库技术基础等。

(三) **应用基础**:计算机控制技术;网络技术;软件工程;多媒体技术等。

(四) **技术基础扩展**:编译原理与编译构造;知识工程——网络计算机环境下的知识处理。

(五) **应用基础扩展**:计算机辅助设计;单片机实用基础;图形、图像处理基础;传感器与测试技术;计算机外设与接口技术。

本系列教材主要是针对计算机教学编写的,供普通高校、社会民办大学、高等职业学校、业余大学等计算机本科或专科选用。其中一部分也适合非计算机专业本科教学使用。在这些教材的内容简介或前言中对使用范围均作了说明。

本系列教材在编写时,注重以下几点:(1) 面对计算机科学与技术动态发展的现实,在内容上应具有前瞻性;(2) 面对学以致用,既有系统的基础知识,又具有应用价值的实用性;(3) 具有科学性、严谨性。另外,力求使有限的版面具有最大的信息量,以使读者得到实惠。

能否实现这些愿望,只有师生在教学实践中评价。我们期望得到师生的批评和指正。

高校计算机教学系列教材编委会



高校计算机教学系列教材编委会成员

主 任:赵沁平

副 主 任(常务):陈炳和

顾 问:麦中凡

委 员(以姓氏笔划为序):

吕景瑜(北工大教授)

乔少立(社长,副教授)

麦中凡(北航教授,教育部工科计算机基础教学指导委员会副主任、中专计算机
教学指导委员会顾问)

苏开娜(北工大教授)

陈炳和(北工大教授)

张鸿宾(北工大博导)

郑玉明(北工大副教授)

金茂忠(北航博导)

赵沁平(北航博导,国务院学位办主任)



前 言

《汇编语言程序设计》是一门比较经典的计算机类专业基础课,是高等院校电子计算机软、硬件及计算机应用专业学生必修的核心课程之一。它不仅是计算机组成原理、操作系统等专业课程的必要先修课,而且还是微机原理、单片机应用等课程的学习基础。同时,该课程对于训练学生掌握程序设计技术,熟悉上机操作和程序调试技术都有重要作用。随着处理器的不断发展,课程的内容和重点也在不断更新。汇编语言程序设计由于其语言本身的特点,使得这一课程必须结合一种具体的处理器及其相关的指令系统来组织教学。因此,我们选择了比较经典的 Intel 8086/8088 系列处理器作为基础而编写了本书,因而本书中所有例题均可以在使用 Intel 8086/8088 系列的处理器的各种计算机上运行。本书适于初学者学习和使用。只要有一种高级语言程序设计基础,都可以通过学习本书掌握基本的汇编语言程序设计技术。因此,本书不仅可作为高等院校“汇编语言程序设计”课程的教材,也可以供需用汇编语言的工程技术人员及科研人员使用。

汇编语言是计算机能提供给用户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,因而在对于程序的空间和时间要求很高的场合,汇编语言是必不可少的。至于对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。另外,当我们需要为各类实际应用设计专有的系统时,出于种种需要,需要改写通用的操作系统中的部分操作,成为专用操作系统或嵌入式操作系统时,汇编语言与高级语言的连接和编译也是需要的。

出于以上的考虑,本书在编写上侧重汇编语言的基础结构和原理,并着重编写了与高级语言的接口技术和硬件控制技术,对于可以用高级语言实现的各类算法和计算予以简化,只保留基本的结构,例如分支和循环等。作为计算机汇编语言程序设计类教材,本书主要阐述了依托于处理器及其兼容处理器的汇编语言程序设计的方法和技术。全书共 8 章:第 1 章介绍基础知识——汇编语言所用的基础知识。已了解计算机基本原理的读者可以跳过这章。第 2、3 章说明处理器的指令系统及包括伪操作在内的汇编语言程序格式,并给出各种指令的使用举例;介绍了伪操作、汇编语言程序格式及汇编语言的上机过程;详细介绍 IBM/PC 机的指令系统和寻址方式。第 4、5、6、7 章叙述顺序、分支、循环、子程序等基本程序结构以及程序设计的基本方法和技术,并说明了 BIOS 和 DOS 系统功能调用的使用方法。第 8 章主要说明多



个模块相连接时有关的程序设计技术以及汇编语言程序与高级语言程序的连接技术。全书提供了大量程序实例,每章后均附有习题。

本书由孙卫真同志主编,其中第1~3章由饶敏同志编写;第4、5章由杨西珊同志编写。孙卫真同志编写了第6~8章,并进行全书统稿。

编著者

2003年8月



目 录

第 1 章 基础知识	1
1.1 前 言	1
1.2 计算机系统概述	1
1.2.1 硬 件	1
1.2.2 软 件	4
1.3 数据编码和数据运算	5
1.3.1 制式的转换	6
1.3.2 二进制算术运算和逻辑运算	8
1.3.3 计算机中数和字符的表示	11
1.4 汇编语言和汇编语言程序设计	19
1.4.1 什么是汇编语言	19
1.4.2 学习汇编语言的意义	19
1.5 中央处理单元——微处理器 8086/8088	21
1.5.1 8086/8088 微处理器功能结构	21
1.5.2 8086/8088 的存储器组织	25
小 结	28
习 题	28
第 2 章 IBM/PC 的指令系统和寻址方式	30
2.1 寻址方式	30
2.1.1 与数据有关的寻址方式	31
2.1.2 与转移指令有关的寻址方式	38
2.2 IBM/PC 机的指令系统	41
2.2.1 数据传送指令	41
2.2.2 算术运算指令	53
2.2.3 逻辑运算和移位指令	67
2.2.4 串处理指令	74
2.2.5 控制转移指令	79
2.2.6 处理机控制指令	91
小 结	93
习 题	93
第 3 章 8086/8088 汇编语言程序格式	96
3.1 汇编语言的语句	96



3.1.1 语句的结构	96
3.1.2 语句格式	97
3.2 伪操作	100
3.2.1 数据定义及存储器分配伪指令	100
3.2.2 表达式赋值位操作	105
3.2.3 段定义伪指令	106
3.2.4 程序开始和结束伪指令	108
3.2.5 过程定义伪指令	108
3.3 宏语言语句	109
3.3.1 宏定义、宏调用和宏展开	109
3.3.2 宏指令与子程序的区别	111
3.4 汇编语言程序的上机过程	112
3.4.1 概述	112
3.4.2 用 EDIT 建立 ASM 文件	113
3.4.3 用汇编程序 MASM 对源文件汇编产生目标文件 OBJ	114
3.4.4 用 LINK 产生 EXE 文件	115
3.4.5 程序的执行与调试	116
小 结	116
习 题	117
第 4 章 顺序程序设计	120
4.1 汇编语言程序设计的基本步骤	120
4.1.1 编写程序的步骤	120
4.1.2 程序流程图的画法	121
4.2 顺序程序设计	123
4.2.1 顺序程序的结构形式	123
4.2.2 顺序结构程序设计	123
习 题	125
第 5 章 分支程序设计	126
5.1 分支程序的结构	126
5.2 双分支程序设计方法	127
5.3 多分支程序设计方法	131
5.3.1 转移表法多分支程序设计	131
5.3.2 地址表法多分支程序设计	133
5.3.3 逻辑分解法多分支程序设计	135
习 题	136



第 6 章 循环控制	139
6.1 概 述	139
6.2 循环程序的基本结构形式与程序设计	139
6.2.1 基本结构形式	139
6.2.2 程序设计方法	140
6.2.3 多重循环程序设计	149
习 题	154
第 7 章 子程序设计	155
7.1 概 述	155
7.2 子程序设计指令	155
7.2.1 过程调用和返回指令	155
7.2.2 过程定义伪指令	159
7.3 子程序的参数传递	162
7.3.1 用寄存器传递参数	162
7.3.2 通过变量传递参数	163
7.3.3 通过地址表传送变量地址	164
7.3.4 通过堆栈传送数据	165
7.4 子程序的嵌套、递归与重入	167
7.4.1 子程序的嵌套	167
7.4.2 子程序的递归	168
7.4.3 子程序的重入	168
7.5 DOS 功能调用及应用	168
7.5.1 DOS 功能调用概述	169
7.5.2 几种常用的 DOS 功能调用	170
习 题	172
第 8 章 汇编语言与高级语言的混合编程	173
8.1 概 述	173
8.2 Turbo C 嵌入汇编方式	173
8.2.1 嵌入汇编语言的格式	173
8.2.2 汇编语句访问 C 语言的数据	174
8.2.3 嵌入汇编的编译过程	176
8.3 Turbo C 模块连接方式	177
8.3.1 混合编程的约定规则	177
8.3.2 混合模块的编译和连接	180
8.3.3 混合模块的参数传递	182
8.3.4 汇编语言程序对 C 语言程序的调用	189



8.4 汇编语言在 Visual C++ 中的应用	192
8.4.1 嵌入汇编语言指令	192
8.4.2 调用汇编语言过程	196
8.4.3 运用带参数的过程定义	199
习 题	201
附 录	204
附录一 指令表	204
附录二 伪指令表	208
附录三 MS-DOS 与 BIOS 调用表	209
附录四 DEBUG 命令表	219
参考文献	220



第 1 章

基础知识

1.1 前言

本章简要地介绍了计算机系统的基本概念和基本组成,这对于读者理解这些基本概念和掌握计算机系统软、硬件的基本组成的内容是必要的。

计算机的工作就是通过计算机的硬件连续不断地执行程序,也就是执行指令来解决人们交给计算机处理的实际问题的。因此必须要熟悉和理解什么是指令、程序、源程序、目标程序、机器语言、汇编语言和高级语言等一系列的基本概念。

在学习指令系统及汇编语言程序设计之前,首先要了解和掌握编程的硬件环境,也就是硬件编程模型。本课程是以 8086/8088 微处理器来讨论汇编语言程序设计的,所以必须熟练掌握 8086/8088 的内部可编程寄存器和存储器的编址。

1.2 计算机系统概述

计算机的基本工作原理是预先将程序存储起来,并根据程序对输入的数据进行处理。

1.2.1 硬 件

计算机的硬件由运算器、控制器、存储器、输入设备和输出设备五大部件组成。其中存储器又分内存储器和外存储器(简称为内存和外存)。外存和输入设备以及输出设备统称为外围设备或外部设备;运算器、控制器和内存合称为主机,而运算器和控制器两部分又称为中央处理器——CPU(Central Processing Unit)。随着大规模集成电路技术的迅猛发展,计算机五大组成部分中的运算器和控制器已经能够集成制造在一块集成电路芯片上了,这就是微处理器(Micro Processor—— μP),又称微处理机。

微型计算机(Micro Computer—— μC)是以微处理器为基础,配以内存以及输入/输出接口电路和相应的辅助电路而构成的。

微型计算机系统(Micro Computer System—— μCS)是指由微型计算机配以相应的外围设备(如打印机、显示器、磁盘机和磁带机等)及其它专用电路、电源、面板、机架以及足够的软件而构成的系统。

这三者之间的关系如图 1-1 所示。

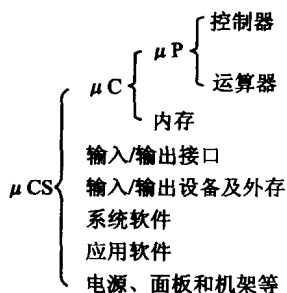


图 1-1 计算机系统结构图

1. 微处理器

微处理器主要由二部分组成。

(1) 运算器

运算器包括算术逻辑部件(ALU)和寄存器组。ALU 用来对数据进行算术、逻辑运算。运算结果和一些特征由寄存器组存储。

(2) 控制器

控制器包括指令寄存器、指令译码器以及定时与控制电路。根据指令译码的结果,以一定的时序发出相应的控制信号,用来控制指令的执行。

2. 存储器

存储器是计算机中专门用来存储程序 and 数据的设备。存储信息的基本单位是一个二进制位(bit),一位可存储一个二进制数 0 或 1。每 8 位组成一个字节(byte)。位编号如图 1-2 所示。

字节的最低位称为第 0 位,最高位称为第 7 位。

1 个字节

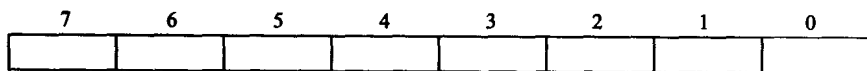


图 1-2 一个字节的位编号

8086/8088 的字长为 16 位,由 2 个字节组成。位编号如图 1-3 所示,字的最低位称为第 0 位,最高位称为第 15 位。

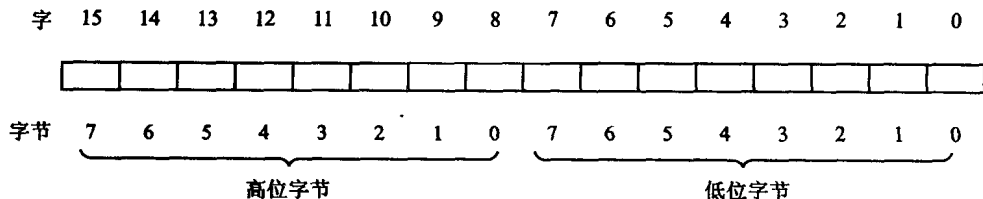


图 1-3 一个字的位编号

存储单元的编号叫地址。如果每个单元存储 8 位二进制数,称字节编址。如果每个存储单元存放一个字(对 8086/8088 即为 16 个二进制位)称为字编址。

(1) 存储容量

存储容量是指存储器可以容纳的二进制信息量,常以字节(符号为 B)为单位表示,定义如下:

$$1\text{KB(千字节)} = 2^{10} \text{ 字节} = 1024 \text{ 字节}$$

$$1\text{MB(兆字节)} = 2^{20} \text{ 字节} = 1024 \cdot 1024 \text{ 字节} = 1048576 \text{ 字节}$$

$$1\text{GB(千兆字节)} = 2^{30} \text{ 字节} = 1024 \cdot 1024 \cdot 1024 \text{ 字节} = 1073741824 \text{ 字节}$$



(2) 存储器中的数据组织

一个存储单元中存放的信息称为该存储单元的内容,图 1-4 表示了存储器里存放信息的情况。由图 1-4 可知:8 号存储单元中存放的信息为 34H,可表示为:(0008H)=34H。

其中 H 是十六进制数表示时的后缀,下同。

8086/8088 字长是 16 位,一个字存入存储器要占有相邻的两个字节存储单元。存放时,字的低位字节存入低地址的存储单元,字的高位字节存入高地址的存储单元。若是字操作,则 8 号存储单元中存放的信息为 1234H,可表示为:(0008H)=1234H。

如果用 X 表示某存储单元的地址,则 X 单元的内容可以表示为(X)。假如 X 单元中存放着 Y,而 Y 又是一个地址,则可用(Y)=(X)来表示 Y 单元的内容。上述表示对 X 地址的间址操作。如图 1-4 中

(0008H)=1234H

而 (1234H)=2F2EH

则对 8 号存储单元间址的字节操作(8bit)可表示为:((0008H))=2EH

字操作(16bit)可表示为:((0008H))=2F2EH

地址	单元内容
	
0008H	34H
0009H	12H
	
1234H	2EH
1235H	2FH
	

图 1-4 存储单元的地址和内容

3. 外部设备

中央处理器(CPU)和主存储器(Memory)构成计算机的主体,称为主机。主机以外的大部分硬件设备都称为外围设备或外部设备,简称外设。

(1) 外设接口

外设接口是 CPU 与外设之间传送信息的一个“界面”、一个连接部件。它一边通过 CPU 的系统总线同 CPU 连接,一边通过系统总线的三种信息通道——数据、控制和状态信息通道同外设连接,如图 1-5 所示。

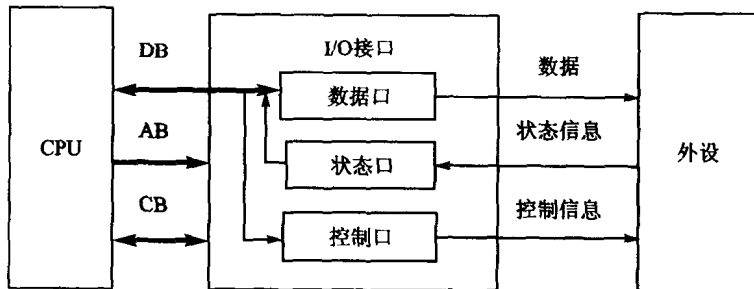


图 1-5 外设与 CPU 接口

(2) 端口

接口中可以由 CPU 进行读或写的寄存器称为端口。这些端口可分为“数据口”、“状态口”与“控制口”,分别存放数据信息、状态信息和控制信息。



(3) I/O(输入/输出)端口的编址方式

CPU 对外设的访问实质上是对外设接口电路中相应的端口进行访问。I/O 端口的编址方式有两种——统一编址和独立编址。

端口统一编址是存储器和 I/O 端口共用统一的地址空间,可以利用访问存储器(取数/存数)的指令完成 I/O 功能。

端口独立编址是使 I/O 端口的编址与内存编址加以区别,专设 I/O 指令。8086/8088 用 IN 指令完成数据输入,OUT 指令完成数据输出操作。

系统总线把 CPU、存储器和 I/O 设备连接起来,用来传输各部分之间的信息。系统总线由数据总线、地址总线和控制总线组成。

(1) 数据总线 DB(双向):从微处理器向内存、I/O 接口传送数据的通路或反之。

(2) 地址总线 AB(单向):是从微处理器向内存、I/O 接口传送地址的通路。

(3) 控制总线 CB(总体双向,某根单向):微处理器向内存、I/O 接口传送命令信号以及外界向微处理器传送状态信号等信息的通路。

1.2.2 软件

计算机软件是计算机系统的重要组成部分。它可分为系统软件 and 用户软件两大类。系统软件是由计算机厂商提供给用户的一组程序。这些程序是用户使用机器时为产生、准备和执行用户程序所必需的。用户软件则是用户自行编制的各种程序。图 1-6 表示了计算机软件的层次。下面我们简要地介绍系统软件的组成。

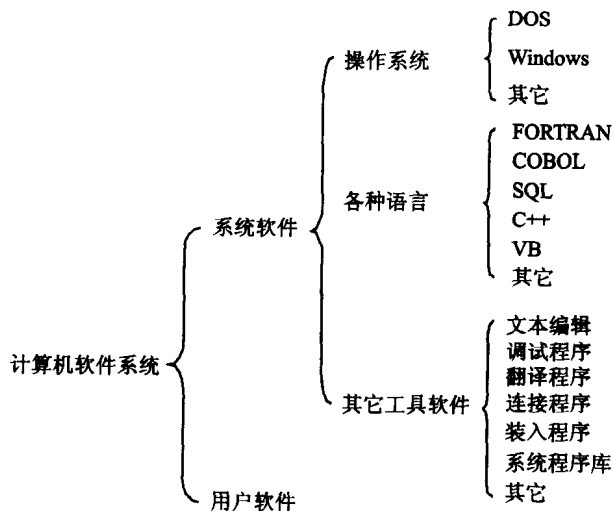


图 1-6 计算机软件层次图

系统软件的核心称为操作系统(operating system)。操作系统是管理程序的集合。它的主要作用是对系统的软、硬件资源进行合理的管理,为用户创造方便、有效和可靠的计算机工作环境。



操作系统的主要部分是常驻监督程序(monitor),只要一开机它就开始运行。它可以接受用户命令,并使操作系统执行相应的动作。

I/O 驱动程序(I/O driver):用来对 I/O 设备进行控制和管理。当系统程序或用户程序需要使用 I/O 设备,只要发出命令,执行 I/O 驱动程序,就完成 CPU 和 I/O 设备之间的信息传送。

文件管理系统(file management):用来处理存放在外存中的大量信息。它可以和外存的设备驱动程序相连接,对存放在其中的信息以文件(File)形式进行存取、复制及其它管理操作。

文本编辑程序(text editor):文本是指由字母、数字和符号等组成的一组信息,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。文本编辑程序用来建立、输入或修改文本,并使它存入内存或外存中(磁盘)。例如,PC 机提供的编辑程序 EDIT 用来建立源文件,修改文本,有插入、删除、编辑和显示等功能。中西文字处理程序 WPS、写字板等可提供全屏编辑功能,并能提供各种功能及命令菜单,使文本的建立和修改更加方便。

翻译程序(translator):把用户用汇编语言或高级语言编写的程序(称为源程序)翻译成机器语言程序(称为目标程序)。汇编语言的翻译程序称为汇编程序(assembler)。高级语言的翻译程序有两种形式。一种是将高级语言编写的源程序一边进行解释,一边执行。这种翻译程序称为解释程序(interpreter),如 BASIC 采用这种形式。另一种是先把高级语言编写的源程序翻译成机器语言程序,先存储起来,然后再执行。这种翻译程序称为编译程序(compiler)。多数高级语言如 C、PASCAL 等都采用这种形式。

连接程序(linker):用来把要执行的程序与库文件(子程序库中的子程序)或其它已翻译过的子程序(能完成一种独立功能的程序模块)连接在一起,形成机器能执行的程序。

装入程序(loader):用来把程序从外存储器传送到内存,以便机器执行。例如,计算机开机后需要立即启动装入程序,把常驻监督程序装入内存,使机器运转起来。

调试程序(debug):系统提供给用户的能监督和控制用户程序的一种工具程序。它可以装入、修改、显示、单步或连续执行一个程序。在 PC 机上,简单的汇编语言程序可以通过调试程序来建立、修改和执行。

系统程序库(system library)和用户程序库(user library):各种标准程序、子程序及一些文件的集合称为程序库。它可以被系统程序或用户程序调用。操作系统还允许用户建立程序库,以提高不同类型用户的工作效率。



1.3 数据编码和数据运算

现代计算机有数字电子计算机和模拟电子计算机两大类。目前大量使用的计算机属于数字电子计算机,只能接受 0、1 形式的二进制数据。但现实中,计算机处理的信息形式各种各样,既有文字、数字、图形、图像等静态信息,也有声音、动画、活动影像等动态信息。无论哪种



形式的信息,都需要输入设备把这些信息转换成 0、1 组合的二进制数形式送入计算机,计算机才能进行存储与处理。

能够进行算术运算得到明确数值概念的数称为数值数据。数值数据有小数、整数和浮点数三种,并且可能是正数或者负数;而以二进制数形式进入计算机的声音、图像、文字等信息称为非数值数据。本小节介绍计算机中数值数据与非数值数据信息的表示方法。

1.3.1 制式的转换

在人们的社会生产活动和日常生活中,大量使用着各种不同的进位计数制,不仅有应用十分普遍的十进制,还有六十进制(如分、秒的计时)、十二进制(如十二个月为一年)等。在现代计算机中,数的表示则采用了二进位计数制。

数据无论使用哪种进位制,都包含两个基本要素:“基数”(radix)与各位的“位权”(weight)。

1. 基 数

一种计算机允许选用基本数字符号的个数叫基数。在基数为 R 的计数制中,包含 R 个不同的数字符号,每个数位计满 R 就向高位进位,即“逢 R 进位”。

几种常用的进位数制:

二进制	$R=2$	基本符号	0、1
八进制	$R=8$	基本符号	0、1、2、3、4、5、6、7
十进制	$R=10$	基本符号	0、1、2、3、4、5、6、7、8、9
十六进制	$R=16$	基本符号	0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F

计算机采用二进制。它的一个很大缺点就是表示一个数所需位数多,人们阅读、书写、记忆等不太方便。例如十进制数 57_{10} ,它的二进制表示需要 6 位,即 111001_2 。为了便于人们阅读与书写,经常用八进制数或十六进制数来代替二进制数。为表示进位制,在数字后面加一个缩写的字母作标识(见表 1-1):

表 1-1 几种进位制的标识

	标识字母	原 文	注 释
二进制数	B	Binary	避免字母 O 误认作数字 0,标识改为 Q
八进制数	Q	Octal	
十进制数	D	Decimal	
十六进制数	H	Hexadecimal	

例如 35D、110B、75Q 和 36H,从其最后一个标识字母就可以知道它们分别是十进制、二进制、八进制和十六进制数,通常不加标识时默认是十进制数。

十、二、八和十六进制数之间的对应关系见表 1-2。