

QUICK BASIC
CHENGXU SHEJI

(修订版)

刘秉刚 李兴成 李凤翔 编著
苗良 彭有道
重庆大学出版社

计算机应用基础系列教材

QUICK BASIC

结构化程序设计

计算机应用基础系列教材

73.87424
C110-X

阅览 8 室

QUICK BASIC 结构化程序设计

(修订版)

刘秉刚 李兴成 李凤翔 编著
苗 良 彭有道

重庆大学出版社

内 容 简 介

本书以面向应用,面向未来为出发点,深入浅出地介绍了最新的结构化语言 QUICK BASIC 的基本语法,它既保留了传统会话式 BASIC 易学易用的特点,又大大扩充了功能:运行速度快,内存不受限制,结构模块化,有丰富的字符处理、图形、声音等功能。本书在介绍语法的同时,密切联系常用算法的应用,这样有利于培养独立分析问题的能力和良好的程序设计习惯。

本书可供高等农业院校各专业作计算机应用基础课的教材,也适于其它高校非计算机专业的学生和有关科技工作者参考。

QUICK BASIC 结构化程序设计

刘秉刚 苗 良 彭有道 编著

责任编辑 韩 洁

*

重庆大学出版社出版发行

新华书店经销

重庆建筑大学印刷厂印刷

*

开本:787×1092 1/16 印张:16 字数:400 千

1997 年 7 月第 2 版 1998 年 3 月第 7 次印刷

印数:48001—53000

ISBN 7-5624-0501-8/TP·31 定价:17.00 元

修订版前言

QUICK BASIC 语言是 BASIC 家族中的第二代成员,它是当今世界上流行的、功能很强的一种编译型程序设计语言,比第一代 BASIC 更先进。第一、QUICK BASIC 采用了编译方式,大大提高了运行速度,但同时又保留了第一代 BASIC 易学易用的特点;第二、它可以充分利用微机的内存资源,不再受 64KB 的限制,因而处理问题的能力大为增强;第三、它是一种结构化语言,易阅读,便于设计和维护;第四、增强了非数值计算(文字处理、图形、声音)的功能;第五、采用屏幕菜单技术,使人机会话更直观方便,操作简单;第六、与第一代 BASIC 兼容,用户原来开发的 BASIC 程序大都可以不加任何修改就可以在 QUICK BASIC 环境中运行。美国微软公司的缔造者,现任总裁比尔·盖茨在一篇文章中曾写到:“我曾向程序员们挑战,为解决任一问题,他们可以选用任何工具编程,我打赌,我用 QUICK BASIC 会更快地编出同样的程序”。

进入 90 年代以来,BASIC 家族仍在迅速发展,目前它的第三代成员已活跃在计算机舞台上,其中在 WINDOWS 环境下运行的 Visual BASIC 就是其中的优秀代表。它既可以用来开发任何软件,又可以作为多媒体的开发工具。美国微软公司宣称:“不久的将来,微软公司将利用 BASIC(指第三代)完成大部分应用程序的可视程序设计,同时要将 BASIC 作为一种中央控制语言使用”。他们认为:“BASIC 由于具有最高的知名度,将仍然是熟知它的专业程序设计人员和实力用户的首选语言”。在 WINDOWS 环境下 Visual BASIC 与 QUICK BASIC 完全兼容,即用 QUICK BASIC 所写的程序可以在 Visual BASIC 环境下执行,要想掌握 Visual BASIC 的使用必须以 QUICK BASIC 为先导,不可操之过急。因此,作为程序设计的入门语种应首推 QUICK BASIC。

目前我国第一、第二、第三代 BASIC 都在流行,都有众多用户,我国国家教委组织的全国计算机等级考试的二级大纲中,对 BASIC 没有指定哪一代,因此,为了扩大本书的适用范围,本书充分注意了与第一代 BASIC 的兼容性。在讲述时,先讲两者兼容的部分,称为 BASIC,然后再讲 QUICK BASIC 的独有部分。

本书自 1992 年出版后,得到读者的欢迎,他们充分肯定了本书的特点:既讲了第一代 BASIC,又讲了第二代 BASIC;既讲了它们的相同点,又讲了它们的不同点;既适合于初学者,又适合于从第一代向第二代过渡的用户,同时还为第三代打下基础,因此,本书曾多次印刷还不够用。为了更好地满足广大用户的需要,今特修订再版。修订版主要修改了原书中的错误、增删了一些内容、加强了程序设计方法和技巧的例题,另外,为了便于用户上机,特增加了上机方法一章(第十三章)。

本书的第一版是集体劳动的成果,在拟定了编写大纲的基础上,分工负责,成文后又经过两次集体审稿,反复修改,最后,由主编对全书统一定稿。编写组由西南农业大学刘秉刚任主编(编写第一章)、由华中农业大学李兴成(编写第八、九章)、河北农业大学李凤翔(编写第四章)、山东农业大学苗良(编写第六章)、河南农业大学彭有道(编写第二章)任副主编,编委有贵州农学院朱世明(编写第三章)、沈阳农业大学佟立伟(编写第五章)、湖北农学院廖济众(编写第七

015137/02

章)、青海大学武春(编写第十章)、黑龙江八一农垦大学张树光(编写第十一章)、山西农业大学杨国强(编写第十二章)。

这次修订工作受主编委托由李凤翔(修改第一至第六章)、李兴成(修改第七至第十二章,并加写第十三章)完成。

由于编著者水平有限,书中的缺点错误在所难免,恳请读者批评指正。

编 者

1997.6

目 录

第一章 电子计算机的组成	1
§ 1.1 电子计算机的硬件系统	1
§ 1.2 电子计算机中数的表示与编码	3
§ 1.3 从机器语言到高级语言	8
§ 1.4 电子计算机的软件系统	10
本章小结	10
习题一	11
第二章 算法与程序设计	12
§ 2.1 算法的概念	12
§ 2.2 用流程图表示算法	14
§ 2.3 用结构化流程图(N-S图)表示算法	16
§ 2.4 结构化程序设计步骤	19
本章小结	20
习题二	20
第三章 BASIC 语言的基本知识	22
§ 3.1 BASIC 语言的发展	22
§ 3.2 BASIC 源程序的结构	23
§ 3.3 BASIC 字符集	24
§ 3.4 数据的类型	25
§ 3.5 常量	26
§ 3.6 变量	27
§ 3.7 标准函数	28
§ 3.8 表达式	29
§ 3.9 QUICK BASIC 的模块结构	31
本章小结	32
习题三	33
第四章 顺序结构	35
§ 4.1 引例	35
§ 4.2 赋值语句(LET)	36
§ 4.3 输出语句(PRINT)	38
§ 4.4 输入语句(INPUT、READ/DATA)	45

§ 4.5	END、STOP、REM 语句	50
§ 4.6	符号常数说明语句(CONST)	51
	本章小结	52
	习题四	53
第五章	选择结构	55
§ 5.1	引例	55
§ 5.2	关系表达式和逻辑表达式	55
§ 5.3	行 IF 语句	58
§ 5.4	块 IF	59
§ 5.5	选择结构的嵌套	61
§ 5.6	多路选择结构	65
§ 5.7	无条件转移语句(GOTO)	69
§ 5.8	多分支转移语句(ON GOTO)	70
	本章小结	72
	习题五	73
第六章	循环结构	75
§ 6.1	引例	75
§ 6.2	WHILE 循环语句(WHILE—WEND)	76
§ 6.3	FOR 循环语句(FOR—NEXT)	78
§ 6.4	DO 循环语句(DO—LOOP)	84
	本章小结	91
	习题六	92
第七章	数 组	94
§ 7.1	数组的基本概念	94
§ 7.2	数组说明语句(DIM)	97
§ 7.3	应用举例(一)	98
§ 7.4	QUICK BASIC 对数组的扩充	103
§ 7.5	应用举例(二)	106
	本章小结	110
	习题七	110
第八章	自定义函数与子程序	112
§ 8.1	自定义函数	112
§ 8.2	子程序	115
§ 8.3	QUICK BASIC 过程	122
§ 8.4	FUNCTION 子程序	122

§ 8.5 SUB 子程序	125
§ 8.6 应用举例	128
§ 8.7 对 QUICK BASIC 过程的进一步讨论	131
本章小结	139
习题八	140
第九章 字符串处理	142
§ 9.1 字符串常数	142
§ 9.2 字符串变量及数组	143
§ 9.3 字符串变量的赋值	144
§ 9.4 字符串的连接	146
§ 9.5 字符串的比较	147
§ 9.6 取子字符串	150
§ 9.7 字符串的生成	151
§ 9.8 字符串与数的相互转换	154
§ 9.9 改变字符串中的字符	156
§ 9.10 设置和读取系统的日期及时间	157
§ 9.11 QUICK BASIC 对字符串处理的扩展	158
§ 9.12 应用举例	160
本章小结	166
习题九	168
第十章 图 形	170
§ 10.1 显式模式与屏幕坐标系统	170
§ 10.2 有关屏幕控制的语句	171
§ 10.3 图形的基本构成——点、线、圆	173
§ 10.4 图形的着色	178
§ 10.5 QUICK BASIC 中图形的窗口操作	180
本章小结	185
习题十	186
第十一章 文 件	187
§ 11.1 文件的基本概念	187
§ 11.2 顺序文件	189
§ 11.3 随机文件	195
§ 11.4 QUICK BASIC 对读/写随机文件的改进	198
本章小结	203
习题十一	203

第十二章 综合程序设计概要	204
§ 12.1 结构化程序设计方法.....	204
§ 12.2 程序设计综合举例.....	207
习题十二.....	224
第十三章 QUICK BASIC 上机方法	226
§ 13.1 QUICK BASIC 入门.....	226
§ 13.2 编辑 BASIC 源程序	231
§ 13.3 模块操作.....	236
§ 13.4 程序调试.....	239
§ 13.5 在 QUICK BASIC 内建立可执行文件	242
§ 13.6 QUICK 库的建立和使用	243
§ 13.7 命令小结.....	244

第一章 电子计算机的组成

§ 1.1 电子计算机的硬件系统

电子计算机是一个复杂的系统,它由硬件(Hardware)系统和软件(Software)系统组成。硬件系统是指由电子元器件和机电结构等构成的各种设备的物理实体;软件系统是指在计算机中运行的各种程序。本节先介绍它的硬件系统,然后在了解了有关的概念的基础上,再在 § 1.4 介绍它的软件系统。

计算机的硬件系统是由多台不同功能的设备相互连接而成。根据计算机的处理能力和用途的不同,系统的大小和配置是各不相同的。一个大中型的计算机系统,可以由几十台设备组成,而一个微型机系统可以由 4、5 台设备组成。但是,无论系统大小,一个计算机的硬件系统都由以下几个基本部分组成。

一、主机(Main Frame)

它包含中央处理机和主存贮器,是整个系统的核心。

1. 中央处理机

常常用 CPU 表示,它是英文 Central Processing Unit 的缩写。中央处理机中,主要由控制器和运算器构成。

(1)运算器

运算器是执行算术运算和逻辑运算的电子部件,又叫 ALU,即英文 Arithmetic Logic Unit 的缩写。它和算盘的功用相似,算盘是用人工进行操作的计算工具,而运算器能自动连续进行运算,算盘只能做加、减、乘、除等基本的算术运算,而运算器不仅能做算术运算,它还能进行逻辑判断、逻辑比较等基本逻辑运算。

运算器是在控制器的指挥下,从主存贮器中取出数据进行运算,结果传送回主存贮器,形成运算器和主存贮器之间的数据流,在两者之间进行信息交换。

(2)控制器

它好比人的神经中枢,是整个系统的最高司令部,它指挥系统各个部分协调地工作,使系统自动地执行程序。

控制器是由复杂的逻辑电子线路组成。它按事先设计好的指令和程序发出各种控制用的命令,以便系统有条不紊地运行。比如,它从主存贮器中顺序地取出一条一条的指令,每取一条,就要分析这条指令(称为译码),根据这条指令的功能向各功能部件发控制信息,执行这条指令规定的任务。一条指令执行完毕,控制器将获得执行情况的反馈信息,以后又自动地去取下一条指令,又进行译码、执行,重复这一过程,直至程序结束。

2. 主存储器

简称 MS,即英文 Main Storage 的缩写。又叫内部存储器,或称内存。它和人的大脑相似,用来存放计算机运行时的程序和数据。

具有记忆能力的器件种类很多,有半导体存储器、磁介质存储器、光存储器等。现在的计算机中都是用半导体存储器作为主存储器。半导体存储器是由若干集成电路芯片组成,在一块芯片上有成千上万个的存储单元电路,就象构成生物组织的细胞一样。这种单元电路具有两种不同的状态,常称之为双稳态电路。因此可以用 0 和 1 来表示这两种状态。这个单元电路是存储信息的最小单位,称为一个位(即 bit,常译成“比特”)。这个单位很小,常常把 8 个位合起来称为 1 个字节(Byte)。即 $1\text{Byte}=8\text{bits}$ 。

实际上,计算机的存储容量很大,常常还用 KB、MB、GB 作单位:

$$1\text{KB}=2^{10}\text{Byte}=1024\text{Byte}$$

$$1\text{MB}=2^{20}\text{Byte}=1024\text{KB}$$

$$1\text{GB}=2^{30}\text{Byte}=1024\text{MB}$$

根据功能的不同,半导体存储器可分为以下两类:

(1)随机存储器 又叫 RAM(Random Access Memory)。即读写存储器,既可以向它存入信息,又可以从它取出信息。我们称“存入”的动作叫做写(Write)操作,“取出”的动作叫“读”(Read)操作。RAM 是读、写均可的。它有“取之不尽,去旧纳新”的特性。取之不尽的意思是:从 RAM 某单元中读出了信息之后,该单元的信息仍保持不变;去旧纳新的意思是:向某单元写入信息之后,该单元原有的信息将被写入信息代替,即原来的信息不再保留。RAM 只有当电源接通的状态下能存储信息,一旦断电,存储的信息就要全部丢失。所以,它只供计算机运行时存放当前的程序和数据,不利于长期保存信息。

(2)只读存储器 又叫 ROM(Read Only Memory)。和 RAM 不同,只能从它读出信息,一般不能向它写入信息(只能用专门的写入器可以向它写入程序)。一旦写入了程序,无论电源接通或断开,写入的信息都不丢失。因此,常常把一些系统程序写在 ROM 里,供系统启动或运行时使用,这种 ROM,就成了固化软件,又称固件(Firm ware)。

二、外部存储设备

对于大批量的信息存储,只有半导体存储器是不够用的。常常在主存之外配备外部存储器,简称外存,以辅助主存之不足,故又称为辅助存储器,简称辅存。磁盘机和磁带机就是现在广泛使用的外存设备。

磁盘机、磁带机都是根据磁感应原理制成的。它们的结构、特点及用途简述于下:

1. 磁盘机

有磁性介质的圆盘绕转轴旋转,活动的读写头在径向作直线运动,从而在盘面上形成若干同心圆,构成磁道,信息记录在磁道上,磁盘机又分两类:

硬磁盘机(Fixed Disk):将盘片密封在一个盒子内,以保持盘面的洁净,盘片数随容量大小而不同。如一般微机中的硬盘机容量有 20MB、40MB,甚至几百 MB 的。大中型机中的硬盘机容量有 200MB、300MB、600MB、2GB 等。

软磁盘机(Floppy Disk):它的磁盘和普通唱片相似,盘片是可卸的,盘片用纸袋封住,一片一片单独使用,根据直径大小,有 8"、5"、3"等几种。每张盘的容量不等,低密度盘能记录

360KB、高密度盘能记录 1.2MB、1.4MB。

2. 磁带机

计算机专用的磁带机和普通家庭用录音机原理相同,但是在转速、记录密度、磁道数、容量等特性方面不一样,专用磁带机的要求要高些。在低档微机中也可用录音机作外存。在大中型计算机中使用专用磁带机作外存。

磁带上的信息,只能顺序写入和顺序读出,存取速度比磁盘机低,但它容量大、可以拆卸,价格比磁盘机低,所以暂时不用的信息,可以存在磁带上作为后备。

磁盘上的信息,可以随机地写入和读出,存取速度远比磁带机高,虽不及半导体存贮器的速度快,但是断电后不会丢失信息。因此,常常用它存贮重要的程序和数据,和主存配合,使计算机更有效的运行。

磁盘机和磁带机既可以从主机输出信息,写在盘上,又可以从它把信息读出送入主机,因此,相对主机而言,它既是输出设备,又是输入设备。

近年来发展起来的光盘,也是大容量的外存设备,已经投入使用。

三、输入设备

人们使用计算机的时候,就要把信息告诉计算机。把程序和数据送到计算机中去的装置就是输入设备。键盘(Keyboard)是最常用的输入设备,还有软磁盘输入机、卡片读入机、纸带读入机、数字化仪等。此外,光学符号阅读机、语音识别输入设备也已取得很大进展。

四、输出设备

计算机处理的结果要返回给人们,这就要用输出设备。常用的有阴极射线管显示器(CRT)、打印机、绘图仪等等。

阴极射线管显示器,就是常说的显像管,有单色和彩色两类。分辨率有高、中、低 3 档。在荧屏上,既可以显示计算机处理的结果,又可以看到内存中的程序和数据。既可以显示字符(英文或汉字)又可以显示复杂的图形。

打印机的种类很多,常用的有点阵式打印机、宽行打印机、激光打印机等。不仅可以用来作源程序清单和计算结果的输出,而且还可以用它输出表格、图形、编排书刊。

综上所述,控制器、运算器、存贮器、输入设备、输出设备等 5 个部分以一定方式相互连接成为一个和谐的整体。

§ 1.2 电子计算机中数的表示与编码

一、二进制数

在数学和日常生活中,我们最熟悉的是逢 10 进位的十进制数。它有两个基本特征:

1. 数位从右至左,每位代表的数值是按 10 的指数增加的,如个、十、百、千...等,这个数值,我们常称为位权值。例如 2184 这个数可以用多项式表示为

$$\begin{array}{cccccccc}
 2 & 1 & 8 & 4 & = & 2 \times 10^3 & + & 1 \times 10^2 & + & 8 \times 10^1 & + & 4 \times 10^0 \\
 \text{千} & \text{百} & \text{十} & \text{个} & & \text{千} & & \text{百} & & \text{十} & & \text{个} \\
 \text{位} & \text{位} & \text{位} & \text{位} & & \text{位} & & \text{位} & & \text{位} & & \text{位}
 \end{array}$$

可见,十进制数的位权值是以 10 为底的幂,因此任何一个十进制数都可以用一个多项式表示:

$$(N)_{10} = a_n \cdot 10^n + a_{n-1} \cdot 10^{n-1} + \cdots + a_1 \cdot 10^1 + a_0 \cdot 10^0 + a'_{-1} \cdot 10^{-1} + \cdots + a_{-m} \cdot 10^{-m}$$

————— 整数部分 ————— ————— 小数部分 —————

2. 每一位数上要有 10 个不同的符号:0、1、2、3、4、5、6、7、8、9。在多项式中的 a_n 、 a_{n-1} 、 $\cdots$,可以是这 10 个中的任意一个。

用 10 个不同形状的字符表示 10 个数,但是在计算机的硬件中,还不可能找到 10 种不同状态的物理量来表示 10 个数。因此,数制必须修改。我们知道,电子计算机是基于电子器件的特性来工作的,这些电子线路处理的是数字信号,它们表现为电位的高与低,电路的接通与断开等两种状态,我们用“0”、“1”来描述。这样就可以用 0、1 两个状态表示数,以 2 为底的幂作位权值,建立逢 2 进位的二进制数。例如

$$(1101)_2 = 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$$

表 1-1 列出从 0~15 的二进制数的表示。

表 1-1 各种数制对应表

数	十进制	二进制	八进制	十六进制	数	十进制	二进制	八进制	十六进制
零	0	0	0	0	八	8	1000	10	8
一	1	1	1	1	九	9	1001	11	9
二	2	10	2	2	十	10	1010	12	A
三	3	11	3	3	十一	11	1011	13	B
四	4	100	4	4	十二	12	1100	14	C
五	5	101	5	5	十三	13	1101	15	D
六	6	110	6	6	十四	14	1110	16	E
七	7	111	7	7	十五	15	1111	17	F

任何一个二进制数都可以用以下多项式表示:

$$(N)_2 = a_n \cdot 2^n + a_{n-1} \cdot 2^{n-1} + \cdots + a_1 \cdot 2^1 + a_0 \cdot 2^0 + a_{-1} \cdot 2^{-1} + \cdots + a_{-m} \cdot 2^{-m}$$

————— 整数部分 ————— ————— 小数部分 —————

式中 a_n 、 a_{n-1} 、 $\cdots$ 是 0 或 1

二、二—十进制间的转换

1. 二进制数转换为十进制数

按位权值展开,求多项式之和,就得到十进制数。

为了区别不同的数制,我们把数值用圆括弧括起来,在右下角用小号字体的 2、10 等来注明,如 $(110101)_2$ 表示是二进制数, $(216)_{10}$ 表示是十进制数。

(1)整数的转换

$$\begin{aligned}\text{例如 } (110101)_2 &= 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\ &= 32 + 16 + 0 + 4 + 0 + 1 \\ &= (53)_{10}\end{aligned}$$

(2)小数的转换

$$\begin{aligned}\text{例: } (0.101)_2 &= 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= 0.5 + 0 + 0.125 \\ &= (0.625)_{10}\end{aligned}$$

(3)既有整数部分又有小数部分的数,转换时,分别按整数和小数转换,再用小数点连起来。

$$\begin{aligned}\text{例如 } (110101.101)_2 &= (110101)_2 + (0.101)_2 \\ &= (53)_{10} + (0.625)_{10} \\ &= (53.625)_{10}\end{aligned}$$

2. 十进制数转换为二进制数

(1)整数的转换 把十进制整数连续除以 2,记录其余数,就得到二进制数,这个方法简称为除 2 取余法。

例如

$$\begin{array}{r} 2 \overline{) 59} \\ \underline{2 \overline{) 29}} \\ \underline{2 \overline{) 14}} \\ \underline{2 \overline{) 7}} \\ \underline{2 \overline{) 3}} \\ \underline{2 \overline{) 1}} \\ 0 \end{array}$$

余数

1.....最低位 a_0
1 a_1
0 a_2
1 a_3
1 a_4
1.....最高位 a_5

按由高位到低位写下来就得到:

$$(59)_{10} = (111011)_2$$

(2)小数的转换 把十进制小数连续乘以 2,取出其积的整数,就得到二进制数,这个方法简称为乘 2 取整法。

例如

乘积的整数部分

0.5625

↓

a^{-1} 最高位 $\rightarrow 1 \leftarrow$

a^{-2} $0 \leftarrow$

a^{-3} $0 \leftarrow$

a^{-4} 最低位 $\rightarrow 1 \leftarrow$

$$\begin{array}{r} \times 2 \\ 1.1250 \end{array}$$

$$\begin{array}{r} \times 2 \\ 0.2500 \end{array}$$

$$\begin{array}{r} \times 2 \\ 0.5000 \end{array}$$

$$\begin{array}{r} \times 2 \\ 1.0000 \end{array}$$

于是得到 $(0.5625)_{10} = (0.1001)_2$

小数换算中,不是都能得到小数部分乘积为零的结果,这时,只能按精度要求取若干位,作为它的近似值。

例如	0.3
	$\times 2$
0	0.6
	$\times 2$
1	1.2
	$\times 2$
0	0.4
	$\times 2$
0	0.8
	$\times 2$
1	1.6

$$(0.3)_{10} = (0.01001\cdots)_2 \approx (0.01001)_2$$

(3)既有整数部分又有小数部分的数,转换时要把整数部分和小数部分分别计算,再把两部分加起来,就是一个二进制数。

例如 $(59.3)_{10} = (59)_{10} + (0.3)_{10}$

上例已求出 $(59)_{10} = (111011)_2$

$$(0.3)_{10} \approx (0.01001)_2$$

所以 $(59.3)_{10} \approx (111011.01001)_2$

三、八进制数和十六进制数

计算机只能识别二进制数,当数值愈大时,二进制数表示的位数就愈多,例如

$$(11011011011.001)_2 = (1755.125)_{10}$$

这个二进制数有 11 位,小数有 3 位。(可以表示到上千的十进制数),在读数和书写时,都极不方便。因此,在编写程序时又常常使用八进制数和十六进制数。

和十进制、二进制的规则相仿,八进制数采用 8 个符号(即 0、1、2、3、4、5、6、7),逢 8 进位。十六进制数采用 16 个符号(即 0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F),逢 16 进位。

例如 $(16)_8 = 1 \times 8^1 + 6 \times 8^0 = (14)_{10}$

$$(25)_{16} = 2 \times 16^1 + 5 \times 16^0 = (37)_{10}$$

以下介绍各种数制之间的转换方法:

1. 八进制、十六进制转换为十进制 方法与上面二—十进制的转换方法相类似。如上例所示。

2. 八进制数转换成二进制数 把每位八进制数用所对应的 3 位二进制数表示,就转换为它的二进制数。

例:

$$\begin{array}{ccc} (3) & 2 & (6)_8 \\ \downarrow & \downarrow & \downarrow \\ (011) & (010) & (110) \end{array}$$

即 $(326)_8 = (11010110)_2$

3. 十六进制数转换成二进制数 把每位十六进制数用所对应的 4 位二进制数表示,就转换为它的二进制数。

例如

(7	A	3) ₁₆
↓	↓	↓
(0111)	(1010)	(0011)

$$(7A3)_{16} = (11110100011)_2$$

4. 二进制整数转换为八进制数 从最低位开始,向左每 3 位分为一组,不够 3 位的用 0 补足 3 位,将每组的二进制数按对应的八进制数写出,就转换成了八进制数;二进制小数,则从小数点开始向右每 3 位划为一组,不够三位,用 0 补足 3 位,写出其对应的八进制数,例如:

(11101001110.1101)₂

分组为

←	→					
(011)	(101)	(001)	(110)	.	(110)	(100)
↓	↓	↓	↓	.	↓	↓
3	5	1	6	.	6	4

对应的八进制数为:

即 $(11101001110.1101)_2 = (3516.64)_8$

5. 二进制整数转换为十六进制数 从最低位开始,向左每 4 位分为一组,不够 4 位的,用 0 补足 4 位,将每组的二进制数按对应的十六进制数写出,就转换成了十六进制数;二进制小数,则从小数点开始向右每 4 位划为一组,不够 4 位用 0 补足 4 位,写出其对应的十六进制数。

例如

(10110 11.01101)₂

分组为

←	→			
(0101)	(1011)	.	(0110)	(1000)
↓	↓	.	↓	↓
5	B	.	6	8

对应的十六进制数

即 $(1011011.01101)_2 = (5B.68)_{16}$

四、编 码

我们用 0、1 来表示二进制数,是数的一种编码表示。编码的方法在我们生活中也是常见的,如通信中的邮政代码、电话号码、电报中的莫尔斯码,身份证代码、学号代码、汉字的国际码、区位码等等。

在计算机中只给数的表示进行了编码还不够,要处理数的符号是正还是负怎样办?处理英文字符、加减乘除,以至汉字又怎么办?所以都得用 0、1 来给它们编码。现在最通用的一种给字符的编码是 ASCII 代码(即 American Standard Code for Information Interchange 的缩写),例如英文字母 A,用 8 位二进制数 01000001 表示,换成十六进制表示为 $(41)_{16}$,各种字符的 ASCII 代码,列在附录一中。

计算机要做各种运算和处理,这些动作也需要编码,例如要它做 $5+6$ 的加法,就要有取数、进行加、送数等操作,对这些操作的说明也用 0、1 的符号串编码,这就形成了计算机的指令码。

总之,计算机只能识别 0、1 两个符号。和计算机沟通信息,就得要以这两个最简单的符号为基础,对各种信息形式进行变换,把计算机不能直接识别的信息变换为计算机能识别的信息。

§ 1.3 从机器语言到高级语言

一、机器语言

直接用“1”、“0”组成的机器指令编写的程序,这就是机器语言源程序。对计算机来说,这是它唯一能直接“听”得懂的语言。所以,常常称之为面向机器的语言。但是,对使用计算机的人来说,这是十分难懂的语言,它难读、难记、难写,容易出错,不同机型又不通用。显然人和机器之间的通信存在巨大的鸿沟,只有填补上这个鸿沟,使用的人愈是方便容易,机器又能懂得,计算机才能发挥更大的作用。为此,人们研究了一种汇编语言。

二、汇编语言

把用二进制数表示的指令,用一些符号来表示,如用表示操作的英文缩写来代替指令代码,用 16 进制数表示数字,上一节中的几条指令就可写为如下形式:

```
LD      A,          (2BH)
LD      HL,         2CH
ADD     A,          (HL)
LD      (2DH),      A
```

第一条 LD 即 Load 的缩写,表示“取数”的操作,A 表示累加器,(2BH)括号内的十六进制数是内存地址。它的含义是把存放在内存第 43 号地址的数(已存放有数“5”)取出来,放到累加器 A 中。

第二条 LD 仍为取数,HL 表示一个暂时存放数据的寄存器名,它的含义是把内存地址号 44(表示为十六进制数 2CH),放在寄存器 HL 中。

第三条 ADD,是“加”的意思,操作数中指出把 A 和 HL 所指示的 44 号地址中存放的数相加(即将 5 和 6 相加,结果为 11),并把结果放在 A 中。

第四条 LD,送数,把计算结果从 A 中送到内存地址为 45(表示为十六进制数 2DH)的存储单元中存放。

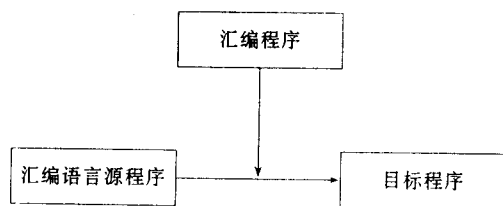


图 1-1 汇编过程

这种用符号代替后的指令,就叫汇编语言,又称符号语言,像 LD、ADD 这类符号称为指令符号或助记符。用汇编语言编写的程序,称为汇编语言源程序,常简称为汇编语言程序。

这种语言,相对机器语言就容易读、容易记、容易写了。但是,机器却一点也不懂了。因此,计算机是无法直接执行的。一个不懂汉语的外国人到中国来要和中国人直接交谈,那是无法进行对话的,所以,只好求助翻译。在计算机中,也同样采取这种方法,人们编写程序用汇编语言,然后请一位翻译,把汇编语言程序翻译成机器能懂得的机器语言程序,这个翻译过程,叫做“汇编”。汇编后产生的机器代码称为目标程序。翻译可以由人手工完成,但做起来既繁琐单