
Quick BASIC

简明教程

任 军 主 编

郭晓军 郭元辉 刘险峰 编

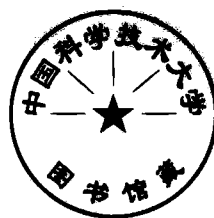


电子科技大学出版社

Quick BASIC 简明教程

任 军 主编

郭晓军 郭元辉 刘险峰 编



电子科技大学出版社

内 容 简 介

本书是 Quick BASIC 的入门教材。全书共分十一章,系统介绍了 Quick BASIC 语言的编程环境、语言成分、程序设计方法以及语言使用中有关的重点难点问题。考虑到读者多为初学者,所以本书以 DOS 下的 QBASIC 系统为入门环境,在读者逐渐掌握了教程内容的基础上,再深入到 Quick BASIC 编译系统。书中附有大量的例题和习题,可供总学时 60~80 的各类院校或培训班讲授 BASIC 语言使用,也适合自学者作为参考书。对广大读者来说,本书是一本详略恰当而实用的入门书。

Quick BASIC 简明教程

任 军 主编

郭晓军 郭元辉 刘险峰 编

*

电子科技大学出版社出版

(成都市建设北路二段四号) 邮编 610054

郫县唐昌印刷厂印刷

新华书店经销

*

开本 787×1092 1/16 印张 12.375 字数 315 千字

版次 1997 年 3 月第一版 印次 1997 年 3 月第一次印刷

印数 1—4000 册

ISBN 7—81043—691—0/TP·279

定价: 13.00 元

前 言

BASIC 语言从 1964 年问世以来,由于简单实用而受到用户欢迎,很快被广泛应用于各个领域。我国各高等院校也把高级 BASIC 作为计算机语言课的首选课程,并将其列为计算机等级考试的内容。但由于 BASICA 只能以解释方式运行,执行速度比较慢,而且所编的程序只能在 BASICA 这种具体的环境中运行。因此,美国 Microsoft 公司于 1987 年 11 月推出了 Quick BASIC,它克服了原有的 BASICA 结构化程度差和运算速度慢的缺点,并且在功能上大为增强。本书就是为非计算机专业的读者学习 Quick BASIC 提供的入门教程。本书具有如下几个特点:

1. **结构严谨** 本书在内容组织上,充分考虑了 Quick BASIC 系统的内在逻辑性,注重系统性、条理性和连贯性,并以结构化程序设计思想贯穿全书,使读者能在学完本书后,系统地完整地掌握 Quick BASIC 语言。

2. **重点突出** 根据我们多年从事教学工作的经验和体会,全书始终围绕提高能力、注重算法、适应等级考试这一主要目标,重点讲解读者普遍感到困难的部分,并辅以大量的例题和习题,使读者只花较少的时间和精力就能很快入门。

3. **由浅入深** 考虑到读者多为初学者,所以本书以 DOS 下的 QBASIC 系统为入门环境,在读者逐渐掌握了教程内容的基础上,再深入到 Quick BASIC 编译系统。这样,初学者不必把主要精力花在熟悉系统环境上,而能由浅入深地渐入佳境。

4. **立足实用** 本书在展开内容上,主要是强调实用。大部分例题均来源于实际应用或经典例题,部分题目取自等级考试的试题或模拟题,对学生参加计算机等级考试将大有帮助。

本书共分十一章,其中第一、二、七、十、十一章由任军编写,第五、六、八章由郭晓军编写,第三、四章由任军和郭元辉共同编写,第九章由刘险峰编写。黄坤、林万昌参加了部分章节的编写讨论。全书由任军统稿。

编 者

一九九七年二月

目 录

第一章 概 述	(1)
1.1 计算机的发展	(1)
1.2 电子计算机的应用范围	(2)
1.3 计算机的硬件与软件	(4)
1.3.1 硬件系统	(4)
1.3.2 软件系统	(6)
1.4 计算机解决问题的过程	(9)
1.4.1 利用计算机解决实际问题	(9)
1.4.2 程序设计过程	(9)
1.5 Quick BASIC 语言	(10)
1.5.1 Quick BASIC 的特点	(10)
1.5.2 Quick BASIC 的构成	(12)
1.5.3 Quick BASIC 的版本	(12)
1.5.4 Quick BASIC 的安装、启动与退出	(12)
1.6 DOS 下的 QBASIC 语言系统	(14)
1.6.1 QBASIC 的特点	(14)
1.6.2 QBASIC 与 BASICA、GW-BASIC 的比较	(14)
1.6.3 从 BASIC 到 QBASIC 的转换	(15)
思考与练习	(16)
第二章 DOS 的 QBASIC 编程环境	(17)
2.1 QBASIC 的启动与退出	(17)
2.2 QBASIC 的功能菜单	(18)
2.3 File 菜单命令	(19)
2.4 Edit 菜单命令	(20)
2.5 View 菜单命令	(21)
2.6 Search 菜单命令	(22)
2.7 Run 菜单命令	(22)
2.8 Debug 菜单命令	(22)
2.9 Options 菜单命令	(23)
2.10 Help 菜单命令	(23)
2.11 建立并执行一个简单的 QBASIC 文件的实例	(24)
思考与练习	(25)

第三章 QBASIC 语言的基本知识	(27)
3.1 字符集与保留字	(27)
3.2 程序结构和程序行	(28)
3.2.1 程序结构	(28)
3.2.2 程序行	(29)
3.3 数据类型	(30)
3.4 常量与变量	(31)
3.4.1 常量	(31)
3.4.2 变量	(32)
3.5 表达式与运算符	(34)
3.6 表达式运算	(37)
3.7 赋值语句 (LET)	(37)
3.8 注释语句 (REM)	(38)
3.9 暂停语句 (STOP)	(39)
3.10 结束语句 (END)	(40)
思考与练习	(40)
第四章 标准输入输出	(42)
4.1 数据输入	(42)
4.1.1 键盘输入语句 (INPUT)	(42)
4.1.2 行输入语句 (LINE INPUT)	(42)
4.1.3 读数据语句 (READ...DATA)	(43)
4.1.4 数据恢复语句 (RESTORE)	(44)
4.2 屏幕显示与打印	(44)
4.2.1 PRINT 语句	(44)
4.2.2 与 PRINT 相关的几个函数	(47)
4.2.3 WRITE 语句	(47)
4.2.4 PRINT USING 语句	(47)
4.2.5 LPRINT 与 LPRINT USING 语句	(49)
4.3 特殊输入操作	(49)
4.3.1 INPUT \$ 函数	(49)
4.3.2 INKEY \$ 函数	(50)
4.4 交换变量语句 (SWAP)	(50)
第五章 流程控制与程序设计初步	(52)
5.1 条件语句	(52)
5.1.1 单行结构条件语句	(52)
5.1.2 块结构条件语句	(52)
5.2 CASE 多项选择语句	(54)

5.3	GOTO 转向语句	(56)
5.3.1	GOTO 语句	(56)
5.3.2	ON...GOTO 语句	(56)
5.4	循环语句	(57)
5.4.1	FOR...NEXT 循环	(57)
5.4.2	WHILE...WEND 循环	(61)
5.4.3	DO 循环 (DO...LOOP)	(62)
5.4.4	多重循环与退出循环	(64)
5.5	程序举例	(66)
5.6	程序设计与结构化编程	(71)
5.6.1	QBASIC 程序设计	(71)
5.6.2	结构化编程技术	(73)
	思考与练习	(75)
第六章	数组与字符串操作	(77)
6.1	数组的定义	(77)
6.1.1	DIM 语句	(77)
6.1.2	OPTION BASE 语句	(77)
6.2	数组的释放和重新定义	(78)
6.2.1	数组的释放	(78)
6.2.2	数组的重新定义	(79)
6.3	数组应用举例	(79)
6.4	字符串的定义	(83)
6.4.1	字符串的概念	(83)
6.4.2	字符串常量和变量的定义	(84)
6.5	字符串的基本操作及常用字符串操作函数	(85)
6.5.1	字符串的合并	(85)
6.5.2	字符串的比较	(86)
6.5.3	字符串的查询	(87)
6.5.4	子字符串截取	(87)
6.5.5	字符的复制和字符串的修改	(89)
6.5.6	字符串和数值间的互转函数	(91)
	思考与练习	(92)
第七章	过程与函数	(94)
7.1	GOSUB 子程序与 DEF FN 函数	(94)
7.1.1	GOSUB 子程序	(94)
7.1.2	DEF FN 函数	(95)
7.2	SUB 子程序	(96)
7.2.1	SUB 子程序的定义	(96)

7.2.2	SUB 子程序的调用	(97)
7.2.3	SUB 过程的编辑与应用举例	(98)
7.3	函数过程	(102)
7.3.1	函数的定义	(102)
7.3.2	函数的调用	(104)
7.3.3	函数的编辑及应用举例	(105)
7.4	值参数和变量参数	(106)
7.4.1	变量参数的调用方式	(107)
7.4.2	值参数的调用方式	(108)
7.4.3	变量参数和值参数使用总结	(109)
7.5	数组和记录作为参数传递	(110)
7.5.1	数组的传递	(110)
7.5.2	记录的传递	(111)
7.6	局部变量与全程变量	(112)
7.6.1	局部变量和它的作用域	(112)
7.6.2	全程变量和它的作用域	(113)
7.7	递归	(114)
	思考与练习	(117)
第八章	数据文件	(118)
8.1	QBASIC 文件的概念与基本操作	(118)
8.1.1	QBASIC 文件概述	(118)
8.1.2	QBASIC 数据文件的基本操作和基本文件组织形式	(118)
8.2	顺序文件	(119)
8.2.1	顺序文件的概念	(119)
8.2.2	顺序文件的读、写操作	(119)
8.2.3	顺序文件的组织和使用	(123)
8.3	随机文件	(126)
8.3.1	随机文件的概念	(126)
8.3.2	随机文件的组织和使用	(126)
	思考与练习	(131)
第九章	屏幕控制及图形处理	(133)
9.1	图形程序的基本概念	(133)
9.1.1	屏幕显示方式	(133)
9.1.2	像素和屏幕坐标	(133)
9.1.3	SCREEN 语句	(133)
9.1.4	显示器类型简介	(134)
9.2	屏幕控制语句与函数	(136)
9.2.1	LOCATE 语句和 WIDTH 语句	(136)

9.2.2	CSRLIN 和 POS 函数	(136)
9.2.3	CLS 语句	(137)
9.2.4	屏幕控制语句示例	(137)
9.3	颜色设置与测试	(138)
9.3.1	字符显示模式下的 COLOR 语句	(138)
9.3.2	图形显示模式下的 COLOR 语句	(141)
9.4	基本绘画语句	(141)
9.4.1	画点语句	(141)
9.4.2	画线语句	(142)
9.4.3	画圆、椭圆和圆弧	(144)
9.4.4	DRAW 语句	(146)
9.5	填 色	(147)
9.6	文本视窗与图形视窗	(148)
9.6.1	建立文本视窗语句	(148)
9.6.2	建立图形视窗语句	(148)
9.6.3	WINDOW 重新定义视窗坐标	(148)
9.6.4	图形及文本窗口示例	(149)
9.7	简单动画技术	(150)
9.7.1	GET 图形存储语句	(150)
9.7.2	PUT 图形恢复语句	(151)
9.7.3	屏幕分页动画	(151)
9.7.4	示例——移动小球	(152)
	思考与练习	(154)
第十章	陷阱技术	(156)
10.1	错误陷阱	(156)
10.1.1	设置程序性错误陷阱	(156)
10.1.2	错误陷阱程序的编写	(157)
10.1.3	从错误处理程序返回	(159)
10.2	功能键陷阱	(160)
10.3	时钟陷阱	(163)
10.4	音乐陷阱与音响功能	(164)
10.4.1	PLAY 语句	(164)
10.4.2	乐曲的后台演奏	(166)
	思考与练习	(168)
第十一章	Quick BASIC 编译系统	(169)
11.1	Quick BASIC 的菜单选择	(169)
11.1.1	File 菜单命令	(169)
11.1.2	Edit 菜单命令	(170)

11.1.3 View 菜单命令	(171)
11.1.4 Search 菜单命令	(171)
11.1.5 Run 菜单命令	(171)
11.1.6 Debug 菜单项	(172)
11.1.7 Calls 菜单命令	(173)
11.1.8 Help 菜单命令	(173)
11.2 建立可直接执行的 .EXE 文件	(173)
11.3 DOS 下直接编译和连接 .EXE 文件	(174)
11.3.1 编译命令 BC	(174)
11.3.2 连接命令 LINK	(177)
11.4 建立 Quick 库	(179)
思考与练习	(181)
附录 1 Quick BASIC 保留字	(182)
附录 2 编译、运行期间的出错代码表	(183)

随着科学技术的迅猛发展,计算机已成为各个学科领域必不可少的应用工具,这一时代的重要技术特征之一就是计算机技术在整个社会中的广泛应用,计算机进入了人们生活中的大多数领域。掌握计算机已经成为现代人的基本素质。人们广泛谈论的第三次浪潮、信息时代等等,都是以计算机技术与信息科学为标志的。计算机技术的飞速发展,给我们的产业、社会乃至人们的生活方式都带来了革命性的变革。

掌握并训练使用计算机,已成了对各种类型工作的人们的一项基本要求,计算机知识和应用能力也已成为当代大学生知识和能力结构的一个重要组成部分,也是我国教育事业培养跨世纪人才最突出的需要加强的环节之一。因此,很自然地,有关计算机技术的课程,也就成了最重要也是最基本的课程之一。不仅对计算机专业如此,对非计算机专业也是如此;不仅对工科是如此,对理科、医科、农科乃至文科也是如此。当然,其要求的深度与广度有所不同。

Quick BASIC 语言这门课程是计算机类课程的基础课。通过本课程的学习,同学们要了解计算机的基本构成与简单的工作原理,懂得什么是程序,在遇到一个实际问题时,如何去分析问题、设计算法、用 Quick BASIC 语言编制程序,然后在计算机上输入与编辑源程序、编译、调试、修改和运行,直至得到满意的结果。对程序设计而言,要了解什么是程序设计方法,养成良好的编程习惯,从而建立起扎实、良好的软件设计基础。

计算机语言及程序设计是操作性很强的课程。这里所指的操作包括两方面的含义:一方面是在掌握最基本的程序设计方法、算法与程序设计语言的基础上编制程序的操作;另一方面则是在计算机上的上机操作。因此,重视并切实抓紧操作实践环节,对学好计算机语言和程序设计课程是至关重要的。

1.1 计算机的发展

远古时代的人们用结绳垒石等原始的方法记载和处理数据,随后人们发明了算盘和机械式加法器,尽管这些工具提高了数据的处理效率和准确性,但是都停留在手工作业的水平上,难以满足日益增大的数据量和信息处理的需要。19 世纪 80 年代,美国的 H·霍勒内斯发明了卡片制表机,这种机器能自动化地进行卡片的穿孔、效验、分类、整理、和制表等操作,使数据处理水平上了一个台阶。但是这种设备的性能和使用仍然受到机械本身的限制,直到电子计算机的出现,数据处理的方式才真正进入了自动化的时代。

我们现在所说的计算机,则是指数字式电子计算机 (Electronic Digital Computer)。

第一台数字式电子计算机,是本世纪 40 年代后期在美国研制成功的 ENIAC (Electronic Numerical Integrator And Calculator),即电子数值积分和计算机,1946 年正式投入运行,其速度为 5000 次/秒。它占地 1500 平方英尺 (140 平方米),重量超过 30 吨。它由约 1800 只真空

管, 700 只电阻, 10000 只电容及 6000 只开关组成, 耗电 150kW, 被用于编制弹道表及天气预报。第一代计算机从 1951 年持续到 1958 年。它们体积大, 价格昂贵, 启动费力且常常不可靠。其内部操作是通过真空管来管理的。这些真空管非常大, 产生的热量要通过专门的空调来处理。一般将这种电子管作成的计算机称为第一代计算机。

随着 50 年代晶体管的发明, 立即应用在计算机上取代电子管, 使其体积与耗电量大大减小, 速度加快, 可靠性提高。也就是在这个时候, 磁芯代替了磁鼓作为存储媒介, 数据被存储在计算机的磁带或磁盘上。辅助存储器的使用结束了计算机主存容量对程序和数据的规模的限制, 减少了穿孔纸带的使用, 采用磁带来进行输入和输出, 提高了计算机的速度。这是第二代计算机。

60 年代后期到 70 年代初, 集成电路 IC (Integrated Circuit) 的使用, 使计算机步入第三代。其主要标志是集成电路的发展, 集成电路代替了晶体管。有了集成电路以后, 成千上百的电子元件可以被集成到一块不到 1/8 平方英寸的硅片上。以后又出现了小型计算机, 它具有许多大型的计算机的功能, 但其体积更小, 存储容量更大而且更便宜, 它开始使用终端机, 即一个远离主机但通过电子线路与主机联系在一起的输入输出设备。这是第三代计算机的特点。

随着集成电路集成度的不断提高, 计算机又发展成为使用大规模集成电路 LSI (Large Scale Integrated Circuit) 与超大规模集成电路 VLSI (Very Large Scale Integrated Circuit) 的第四代。计算机的发展正在不断地、迅速地进行着。一些新发展的高新技术, 如激光技术、超导体技术、生物电技术等, 迅速地应用在计算机上, 使其性能与可靠性不断提高, 成本却越来越低, 不再是只有少数专家才能问津的神秘的东西。这些, 都为计算机的普及开辟了道路。而使用越普及, 市场需求量越大, 又反过来极大地刺激了技术与生产的发展。计算机的发展与普及正是进入了这样一种良性循环。

计算机的运算速度已从最初的每秒几千次运算, 提高到目前的每秒几千亿次运算, 半个世纪提高了上亿倍, 其发展速度实在是惊人。我国自行设计与研制的银河系列巨型机, 标志着我国跻身于世界上少有的能制造亿次巨型机的行列。

现在, 电子计算机正在向着巨型化、微型化、网络化、智能化方向发展。

1.2 电子计算机的应用范围

计算机的应用几乎已渗入到社会生活的各个方面, 要系统地描述它的各种应用是相当困难的。因为它涉及的面实在是太广。可以说, 计算机应用是现代生活的重要支柱, 没有它是难以想象的。最近刚结束的人与计算机下国际象棋的世纪之战中, 计算机似乎又向人类的智慧提出了新的挑战。以下简要叙述计算机的几个重要应用领域。

1. 数值计算

著名数学家莱布尼兹曾说过: “让一些杰出的人才像奴隶般地把时间浪费在计算上是不值得的。” 因此, 最初发明计算机的目的, 是用于数值计算。数值计算是计算机应用最早也是最成熟的领域。因为, 现实生活中有大量的问题需要经过复杂的计算。例如, 天气预报中, 要考虑到各种因素的影响, 要求解含有很多未知数的高阶微分方程。如果采用人工计算, 则需要很长的时间才能得到结果。而天气预报的及时性要求使迟到的“预报”变得毫无意义。如果采用高速计算机来进行, 就能在预定的时间内作出预报。又如 1946 年研制出来的第一台 ENIAC 计算机就是用来计算火炮的弹道轨迹; 三峡电站的结构设计, 要根据地质条件、水文

条件、气象条件、承重情况、所采用的材料等来加以计算，其计算量是很大的。作人工计算时，由于工作量太大，只能忽略一些条件、假设一些条件，来使计算简化。这种比较粗略的计算的结果，既可能造成隐患，又会造成材料的浪费。采用高速计算机来计算，其设计周期短，结果也要准确得多。由于计算机具有处理速度快、存储容量大、计算精度高、有逻辑判断能力、运行自动化和可靠性好的特点，将其用在数值计算上会大大发挥其作用。

2. 数据处理与信息加工

信息社会的所有活动都受到信息的影响。信息是一种极为宝贵的资源，被称为除了动植物、矿物等自然资源以外的人类维持正常活动不可或缺的第三资源，其重要性越来越受到人们的重视。生产和科学研究实际上就是利用所掌握的信息，通过各种各样的方法来获取新的、有价值的信息的过程。而如何利用我们每天生活、生产和科研所产生的各种各样的极其丰富的数据，研究和分析这些数据，从中得到有用的具有指导意义的信息，并将其应用到社会生活中去，已经成为提高整个社会经济效益和社会效益的关键。

因为信息是用数据记载和传播的，所以对信息的处理具体就体现在对数据的处理上。数据处理是将原始的、分散的、杂乱的数据，利用一定的设备和方法进行处理，然后产生新的有意义的数据组合，即有用信息。数据处理一般包括对数据的收集、记载、分类、排序、存储、计算、加工、传输和制表等处理过程。如财务管理部门，需要将原始凭证收集起来，然后登记在帐册上或存储在计算机中，经过一定的计算和处理形成单位的会计帐、会计报表和财务统计图，这些处理结果或是保存起来备用或是分发上报有关部门。这一过程，就是简单化了的财务数据处理过程。

计算机的高速处理能力，大容量信息存储能力，加上现代化的通信手段构成网络，为信息的存储、传输、处理与管理提供了强有力的手段。全球飞机订票系统、银行管理系统、文献检索系统等都是很好的例子。

以我国的人口普查来说，一个人的记录包括 10 多项数据，全国汇总起来就是几百亿个数据，对这些数据进行分类统计，用人工进行几乎是不可能的，利用计算机进行处理，既快又准确。再如图书馆的图书情报检索系统，要在成千上万种书刊杂志上快速检索出用户需要的，必须借助与计算机。

3. 计算机辅助设计与制造 (CAD/CAM)

CAD (计算机辅助设计) 与计算机辅助制造 (CAM) 已广泛地使用于航空航天、汽车、轮船、机械、电子、化工、建筑、人工合成大分子等的设计中。它可以提高程序质量、加快设计速度。对一些工作来说，甚至是决定性的、不可替代的。例如，计算机本身所需要的超大规模集成电路，要求在一小小的硅片上，设计与制作几万，几十万，乃至几百万个晶体管及其他元件。元件之间的连线已细至微米以下。这样的芯片不用计算机来设计是不可能的。又比如波音公司的无图纸喷气机设计，如果将图纸全部绘出的话，其重量将有几千吨。CAD/CAM 的使用提高了设计质量和自动化程度，加快了新产品的设计和制造速度，大大方便了设计与制造。

除开计算机辅助设计与计算机辅助制造之外，用计算机来进行的辅助测量、辅助教学等，更是层出不穷。

4. 实时控制

将计算机用于各种自动装置、自动仪表、机床工具的工作过程，就称为实时控制，又称

为过程控制。实时控制是计算机的又一大应用领域。在高速运转、高层空间、高危险作业等场合，计算机起着无可替代的作用。它可以提高产量、节约劳动力、降低能耗、减轻劳动强度，从而带来巨大的经济效益。例如火箭发射、同步卫星转轨、卫星姿态控制、飞机盲着陆、核反应堆的控制等。现代化的钢铁、化工、水泥等大型生产线的管理、运转、连续生产与品位控制，也都离不开计算机。

5. 人工智能

人工智能理论的突破，向计算机技术发出了巨大的挑战。人工神经网络迫切需要大型、并行计算机。人们已经研究出了智能集成块，相信在不远的将来，我们可以看到并使用具有自识别、自学习能力的超高速计算机。

计算机的应用领域还很广阔，但我们可以从这简单的介绍中感受到计算机应用领域的广泛，同时也激励我们努力学习，掌握最新的科学技术知识。

1.3 计算机的硬件与软件

一个计算机应包含两部分：机器系统和程序系统。机器系统由存储器、控制器、运算器、输入和输出设备等组成。也就是我们常说的计算机主机及其外围设备，称为“硬设备”或“硬件”。程序系统是各种程序的总称，通称为“软设备”或“软件”，以便与硬设备相区别。软件可以分为两类：系统软件和应用软件。以下简单介绍计算机硬件系统和软件系统的基本概念。

1.3.1 硬件系统

所谓硬件（hardware）是指组成计算机系统的各种电子的、机械的元件、器件与部件，它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不同的。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仅由几台设备组成。但是，不管系统大小，一个计算机的硬件系统一般都是由如图 1.1 所示的几个基本部分组成。

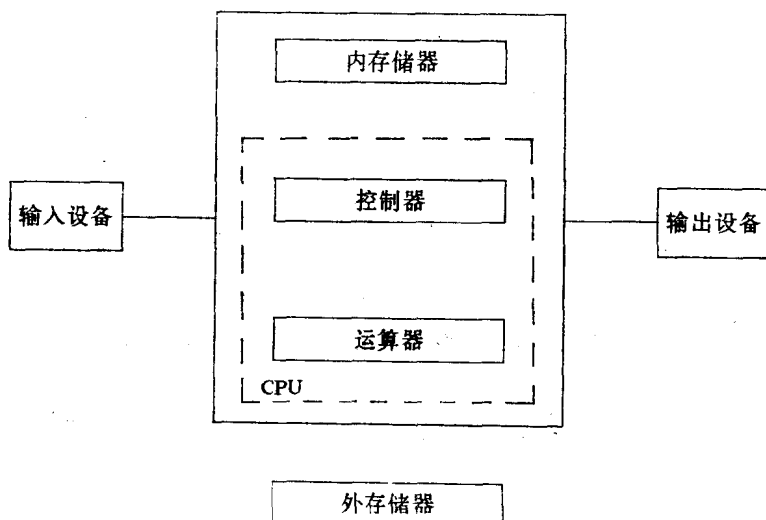


图 1.1 微机系统配置示意图

1. 中央处理单元 CPU (Central Processing Unit)

中央处理单元即 CPU, 是 Central Processing Unit 的缩写。CPU 是计算机的心脏, 对计算机所进行的信息输入、输出、传输、处理、存储等实施灵活、高效的管理。它由两部分组成:

(1) 控制器 CU (Control Unit)

它是计算机的神经枢纽, 可以根据事先编制好的程序, 指挥系统的各部分去完成每条指令所规定的动作。

(2) 算术逻辑单元 ALU (Arithmetic and Logical Unit)

实施算术与逻辑运算。例如, 程序规定要对两个数 A 与 B 作加法, 存结果 C。当控制器得到这条指令后, 就从存储器中分别读出 A 与 B 两个数, 送入算术逻辑单元, 实施运算。

2. 存储器 MS (Main Storage)

又叫主存储器, 用以存放计算机运行时的程序和数据。首先简单介绍几个有关计算机内信息表示与存储的最基本的概念与术语。

我们知道, 计算机内在的电子器件只能处于“通”或“断”两种状态之一, 往往以“0”, “1”来表示其状态。以此为基础的数字系统称为二进制系统 (Binary), 它与我们日常使用的 0~9 十个数字为基础的十进制系统 (Decimal) 有很大的不同。二进制中的每一位, 即一个“0”或“1”, 称为一位 (Bit)。8 位组合起来, 称为一个字节 (Byte), 它是计算机信息存储与传输中最基本的单位。在存储容量很大时, 使用字节作为单位显得太小, 所以常用 KB ($1\text{KB} = 2^{10} \text{ Bytes} = 1024 \text{ Bytes}$), MB ($1\text{MB} = 2^{20} \text{ Bytes} = 1048576 \text{ Bytes}$) 与 GB ($1\text{GB} = 2^{30} \text{ Bytes} = 1073741824 \text{ Bytes}$) 来表示。

由于计算机的性能不同, 它可以同时处理的字节数也不相同。我们将计算机能同时处理字节的数目称为一个机器字 (Word) 或字长。例如, 同时只能处理一个字节的机器称为 8 位机, 同时能处理 2 个字节的称为 16 位机, 同时能处理 4 个字节的机器称为 32 位机, 等等。性能越高的计算机, 其机器字就越长, 标志着处理速度越快, 结果的精度越高。

存储器是在计算机内用以存储信息的重要部件, 目前采用超大规模集成电路形式的半导体存储器。不同类型的计算机的内存容量从几十千到几十兆不等。

存储器又分为随机存储器 (RAM) 和只读存储器 (ROM)。

RAM (Random Access Memory) 每一个存储器可以读出也可以写入信息。写入的信息保留到下一次新的信息写入或计算机断电为止。

ROM (Read Only Memory) 存储单元内的信息是事先写好并长期保留的, 即使计算机断电也不会丢失。用户可以读出存储的信息, 但不能写入新的信息去取代它。

半导体做成的内存, 除少量单元为 ROM 外, 绝大部分为 RAM, 它们的存取速度很快。但是, 限于成本较高, 以及要保持很高的存取速度, 使得内存的容量一般不是很大。RAM 内存储的信息。一旦断电, 就会全部丢失。

上述的 CPU 及内存构成计算机的主机。

3. 输入输出设备及其接口

计算机与外界的信息交换, 由系统中的输入输出设备通过相应的接口来完成。

(1) 输入设备 (input devices)

键盘是最常用的输入设备, 操作者用它向计算机输入指令、程序、数据等。

其他的输入设备还有光电扫描仪、光笔、鼠标器、模拟/数字信号变换器 (A/D) 等。通

信设备也可作为输入设备向计算机输入信息。

(2) 输出设备 (output device)

最常用的输出设备有显示器、打印机、绘图仪等。计算机也可以通过通信设备向外界或其他计算机输入信息。

(3) 外存储器 (auxiliary storage)

计算机内存的信息是不能长期保存的, 随着关机断电而消失。况且, 内存也因为容量较小, 不便使用。因此, 计算机系统都带有外存储器。目前使用得最广泛的外存储器是磁带、磁盘与光盘。它们的成本比半导体成本低, 容量比内存大, 且记录的信息可以长期保存。

在向计算机输入信息时, 外存储器可看作输入设备; 而将内存的信息转存入外存时, 又可将外存储器看作输出设备。

(4) 输入/输出设备接口 (input/output interface)

接口是专门的器件与电路, 把输入/输出等外围设备与计算机主机连接在一起, 并按照一定的规定, 使它们协调地工作, 完成信息的交换。

上面所讨论的计算机主机 (包括 CPU 与内存存储器)、外围设备 (输入/输出设备、外存储器等), 连同接口, 构成了一个计算机的硬件系统。

在讨论计算机的硬件系统时, 很有必要单独提及微型计算机。

从 70 年代开始, 随着集成电路技术的发展, 一些厂家开始把由很多器件组成的 CPU 做在几块, 甚至一块半导体芯片上, 这就是微处理器 (Microprocessor)。由这种微处理器做成的计算机, 称为微型计算机 (Microcomputer), 简称微机。微机一出现, 就以其结构简单、成本低廉、使用方便而风靡全世界。可以毫不夸张地说, 正是由于微机的产生与发展, 在很大程度上促进了计算机的普及与使用, 从而又推动了计算机自身的发展。

在短短的 20 年来, 微机已更新了几代, 从最初的 8 位机, 发展成为 16 位机、32 位机、64 位机等。运算速度提高的同时, 存储器容量也成百倍的提高, 而成本与价格却不断大幅度下降。在硬件发展的同时, 大量的软件工作人员开发了难以数计的软件, 使得微机的使用越来越方便, 功能越来越强。

1.3.2 软件系统

1. 计算机语言

在讨论软件系统之前, 我们先来介绍计算机语言。因为只有硬件系统, 计算机还不能按人的意图办事。因为它只能识别 0 和 1 两种状态。迄今为止, 计算机所作的每一件事, 无论复杂还是简单, 都是使用计算机的人命令它去作的。第一步做什么、第二步做什么、什么时候做什么、如何做等等。人们用以指挥计算机的指令 (instruction) 序列称为程序 (program)。而软件 (software), 则是计算机程序、算法、规则、相关的文档、以及在计算机运行时必须的数据的总称。

由于语言是人们交流思想与信息的工具。所使用的语言, 只有在交流的双方都能理解时, 交流才能成功。当人们使用计算机时, 命令机器作预定的信息处理与交换工作时, 必须首先使计算机“理解”人的意图并执行指令。由此而产生了计算机语言问题。

(1) 机器语言 (Machine Language)

前面已讨论过, 计算机的各种元件只有“通”或“断”两种状态, 用“0”、“1”来表示。因此, 计算机只能接受并执行二进制方式写成的指令, 称为机器指令。每种计算机机器指令

的序列称为机器语言，或低级语言 (Low Level Language)。一个 16 位计算机的一个字长为 16 位，也就是说，由 16 个二进制 (0 或 1) 组成一条指令或者其他信息。16 个 0 和 1 可组成各种各样的排列组合，通过线路变成电信号，让计算机执行不同的动作。例如：

1011011000000000

在某一类计算机中，这条指令的作用是让计算机进行一次加法。

又如：

1011010100000000

这条指令的作用是让计算机进行一次减法。

这两条命令所规定的操作非常简单，但这种二进制代码的形式是很难为人们所理解与记忆的。

使用机器语言时，计算机可以接受并执行，但人们却难以记忆，因为它距离我们日常生活中习惯使用的语言实在太远。而且，每种机器的机器语言都不相同。也就是说，为这种机器写的程序，在另外一个机器上是无法运行的，这给计算机的推广使用造成了极大的困难。

(2) 符号语言 (Symbolic Language)

鉴于机器语言的难于理解、记忆，人们便使用一些符号来代表相应的二进制代码，这就是符号语言，也称汇编语言。比如，将一个数取出，加 1 之后，将结果存回去的程序，用符号语言可以表示为

LD A, (60H)

INC A

LD (60H), A

其中，LD 是指装入，A 是运算器，60H 是地址 01100000 的 16 进制表示，INC A 是将 A 中的数加 1。因此，执行这段程序时，将地址为 60H 中的数调入运算，加 1 之后，将结果存回 60H。可见，这段程序与前面的机器语言程序完全是一回事，只是采用了另外一种符号而已。

使用符号语言，人们容易理解一些了，但计算机可不能接受与执行，因为它只能接受二进制代码。因此，需要一个程序，把上述符号语言写成的程序逐条“翻译”成二进制代码。这种翻译程序就是汇编程序 (ASSEMBLER)。汇编过程如图 1.2 所示。

应该指出的是，符号语言只是在一定程度上改善了机器语言的可读性，并没有改变程序对机器语言的依赖性，因而同样是低级语言。

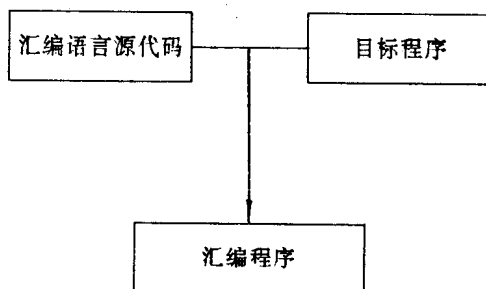


图 1.2 汇编过程示意图

(3) 高级语言 (High-level Language)

从计算机诞生起，人们就在寻求一种使机器语言比较接近人的自然语言，并可以在各类计算机中使用的计算机语言。