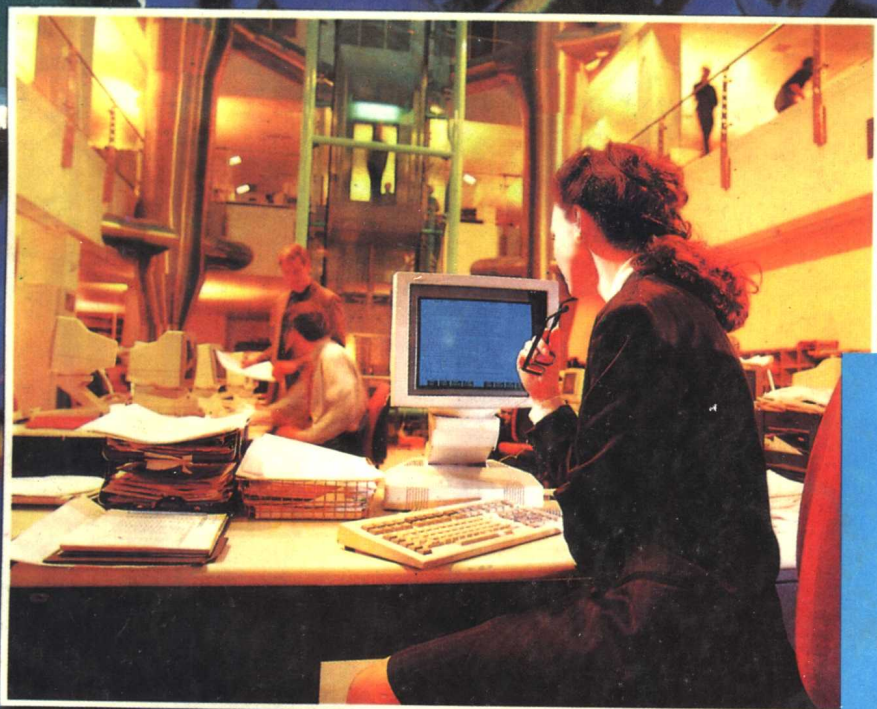


实用电脑丛书

实用电脑编程技巧

湖南出版社



李晓辉 著
刘 宏

实用电脑编程技巧

李晓辉 刘 宏 著

湖南出版社

[湘]新登字 001 号

责任编辑:阳 天

装帧设计:陈 新

实用电脑编程技巧

李晓辉 刘 宏 编著

*

湖南出版社出版、发行

(长沙市河西银盆南路 67 号)

湖南省新华书店经销 湖南省农科院印刷厂印刷

1995 年 6 月第 1 版第 1 次印刷

开本:787×1092 1/16 印张:11.25

字数:260000 印数:1—11000

ISBN 7—5438—0979—6

G·189 定价:12.20 元

前 言

一般初学电脑编程人员,都是通过学习某门语言(例如:BASIC、PASCAL、FORTRAN、C语言)来初步掌握编程方法,但一般的语言教科书都花大量的篇幅讲解该语言的文法和语法,并通过一些简单的程序设计例子让读者了解如何使用该门语言。在初步掌握这些简单的程序设计之后,读者总是想试着编写能够解决实际问题的程序系统。这时他们会发现自己对电脑系统的内部工作机制及原理了解不够,无法在编程中熟练地、充分地利用电脑系统资源。

本书的目的就是为那些已初步掌握电脑程序设计,特别是 Turbo C,但需要进一步掌握电脑编程技巧的编程人员,在电脑程序设计中充分地利用系统资源(如内存、显示器、键盘、文件操作、鼠标、打印机)而编写的。在电脑编程中,常常会遇到一些令人百思不解的“难点”,这往往会耗费你大量的时间和精力。常言道:“会者不难”,寥寥数语的指点,可能使你茅塞顿开。

C语言是被广泛使用的一种高级语言,它兼顾了多种高级语言的特点,并具有汇编语言的功能,从而可以直接实现对系统硬件及外围接口的控制,具有较强的系统处理能力。而 Turbo C更以其方便的集成开发环境、强大的图形功能而著称。所以本书给出的例子都是 Turbo C编写的,而且着重讨论了 Turbo C 的程序设计方法及技巧。但所用的一些概念同样适应其它语言。

本书有三大特点:

- 一、依照硬件资源分章,根据硬件特点又分为节,继而分为小节,每一小节讨论一个特点的编程任务。这样整书脉络清晰。
- 二、在每章或每节的开头都详细讲述了该硬件资源的工作原理,这对于任何语言编程都大有益处,而且是很有必要的。接着谈在 C 编程中如何对这些资源编程。
- 三、本书的每一节有大量的实例,便于读者掌握编程技巧。其中不少是经典技巧,读者可以根据需要把它们移植到所编的程序中去。

本书第一、二、三、四、八章由李晓辉同志编写。第五、六、七章由刘宏同志编写。由于时间仓促和作者的水平有限,在编写中难免会有错误和不足之处,敬请广大读者批评指正。

李晓辉、刘 宏

一九九五年元月

目 录

第一章 微机的基本知识	(1)
§ 1.1 PC 机的内存	(1)
§ 1.1.1 PC 存储器的空间布局	(1)
§ 1.1.2 扩展和扩充内存	(1)
§ 1.1.3 存储器的工作方式	(3)
§ 1.2 进一步理解 DOS 系统	(5)
§ 1.2.1 DOS 的层次结构	(5)
§ 1.3 汉字操作系统	(7)
§ 1.3.1 汉字操作系统结构	(7)
§ 1.3.2 汉字的编码	(7)
§ 1.3.3 汉字操作系统的发展历史	(9)
§ 1.4 标准输入/输出	(10)
§ 1.4.1 标准输入/输出设备的重定向	(10)
§ 1.4.2 标准输入/输出的传输	(11)
§ 1.5 深入使用 DOS 命令	(11)
§ 1.5.1 PROMPT 命令(设置系统提示符)	(11)
§ 1.5.2 APPEND 命令的使用	(12)
§ 1.5.3 ATTRIB 命令	(13)
§ 1.5.4 PRINT 脱机批打印命令	(14)
§ 1.6 配置系统	(16)
§ 1.6.1 建立 CONFIG.SYS 文件	(16)
§ 1.6.2 配置命令	(16)
第二章 中断调用(BIOS,DOS)的应用	(20)
§ 2.1 中断概述	(20)
§ 2.1.1 PC 所提供的中断	(21)
§ 2.1.2 硬件中断及优先级	(22)
§ 2.2 调用 BIOS 服务以确定系统资源	(24)
§ 2.2.1 与 BIOS 和 DOS 的接口函数	(24)
§ 2.2.2 与 DOS 的接口	(28)
§ 2.3 确定系统资源	(28)
§ 2.3.1 系统当前配置情况	(28)

§ 2.3.2 内存服务	(29)
§ 2.3.3 确定 DOS 版本号	(30)
§ 2.3.4 读取和设置中断向量	(30)
§ 2.4 时钟/发声	(30)
§ 2.4.1 PC 机上的时钟系统	(30)
§ 2.4.2 PC/AT 机上的时钟系统	(31)
§ 2.4.3 DOS 的时间/日历服务	(31)
§ 2.4.4 延时函数	(32)
§ 2.4.5 发声	(33)
§ 2.5 Turbo C 和汇编语言混合编程	(33)
§ 2.5.1 段的组合	(33)
§ 2.5.2 变量和函数名的相互引用	(37)
§ 2.5.3 参数传递规则	(38)
§ 2.5.4 返回值传递规则	(39)
§ 2.5.5 寄存器规则	(40)
§ 2.5.6 混合编程示例	(41)
§ 2.5.7 行内汇编	(43)
第三章 键盘	(46)
§ 3.1 键盘工作原理	(46)
§ 3.1.1 键盘缓冲区	(47)
§ 3.2 Turbo C 的键盘输入	(49)
§ 3.2.1 控制台级(conio)键盘输入处理	(50)
§ 3.2.2 标准文件级键盘输入处理	(55)
§ 3.2.3 普通文件级键盘输入处理	(56)
§ 3.2.4 BIOS 级键盘输入处理	(56)
§ 3.3 <Ctrl-BREAK>和<Ctrl-C>	(60)
第四章 视频显示	(64)
§ 4.1 视频工作模式	(64)
§ 4.1.1 视频缓冲区组织形式	(65)
§ 4.2 文本屏幕输出	(68)
§ 4.3 Turbo C 的文本屏幕处理	(69)
§ 4.3.1 文本输出与操作	(69)
§ 4.3.2 窗口和显示方式控制	(72)
§ 4.3.3 属性控制	(74)
§ 4.3.4 状态查询	(75)
§ 4.4 窗口及菜单的制作	(77)
§ 4.4.1 窗口结构和窗口栈	(77)
§ 4.4.2 定义窗口的 16 个函数	(78)
§ 4.5 图形处理	(92)
§ 4.5.1 图形系统控制函数	(92)

§ 4.5.2 画图和填充函数	(96)
§ 4.5.3 屏幕和视口管理函数	(98)
§ 4.5.4 图形方式下的文本输出函数	(100)
§ 4.5.5 颜色控制函数	(102)
§ 4.5.6 错误处理函数	(107)
§ 4.6 图形方式下输出文本的几种技巧	(107)
§ 4.6.1 格式化输出	(108)
§ 4.6.2 抹掉文本信息区域	(109)
§ 4.6.3 加亮	(111)
§ 4.6.4 滚动	(112)
§ 4.6.5 动画设计原理	(112)
第五章 目录管理与文件操作	(113)
§ 5.1 目录管理	(113)
§ 5.1.1 功能介绍	(114)
§ 5.2 文件操作	(121)
§ 5.2.1 文本文件与二进制文件	(121)
§ 5.2.2 缓冲方式	(121)
§ 5.2.3 文件指针	(121)
§ 5.2.4 文件的打开与关闭	(122)
§ 5.2.5 文件读写	(122)
§ 5.2.6 磁盘扇区的读写	(126)
第六章 打印机	(130)
§ 6.1 概述	(130)
§ 6.2 初始化打印机	(132)
§ 6.3 测试打印口的状态	(133)
§ 6.4 向打印口送一个字节	(134)
§ 6.5 多台打印机的切换	(134)
§ 6.6 设置打印参数	(135)
§ 6.7 注意如下几个问题	(139)
第七章 鼠标器	(140)
§ 7.1 概述	(140)
§ 7.2 功能介绍	(142)
第八章 驻留内存程序设计	(155)
§ 8.1 驻留的取得	(155)
§ 8.2 中断接管	(155)
§ 8.3 中断链结	(156)
§ 8.4 TSR 程序的大小	(156)
§ 8.5 环境切换	(158)
§ 8.5.1 堆栈	(158)

§ 8.5.2 程序段前缀(PSP)	(159)
§ 8.5.3 PSP 的切换	(161)
§ 8.5.4 磁盘传送区(DTA)	(162)
§ 8.6 键盘中断	(162)
§ 8.7 计时器中断	(163)
§ 8.8 DOS 的可重入性	(163)
§ 8.8.1 DOS 的三个堆栈	(163)
§ 8.8.2 DOS 忙标志	(164)
§ 8.8.3 DOS OK 中断	(164)
§ 8.8.4 驱动器 ROM—BIOS 中断	(165)
§ 8.8.5 DOS 严重错误中断	(165)
§ 8.8.6 DOS Ctrl—break 中断	(165)
§ 8.9 执行 TSR 实用程序	(166)
§ 8.10 删除 TSR 实用程序的驻留	(166)
§ 8.11 用 Turbo C 设计通用内存驻留程序	(167)
§ 8.12 TSR 范例: 时钟程序	(167)

第一章 微机的基本知识

本章介绍微机的一些基本知识,以便读者在了解微机内部的工作机制的基础上,深入理解微机的工作原理。这对于编程无疑是大有益处的。

§ 1.1 PC 机的内存

内存是程序设计中的一个基本要素,在你编写的程序中经常要分配一些存储单元和用到一些特殊的存储单元,而且你编写的程序必须装入内存之后才能运行,所以对存储器的了解是很有必要的。

§ 1.1.1 PC 存储器的空间布局

8088 CPU 的内存寻址空间为 1MB,在早期的 IBM PC 微机系统中配置的内存存储器通常不超过 640KB。其内存空间分配如图 1-1 所示。

图中 0—640KB 是常规内存,是 PC 机的基本内存。8088 CPU 可对它进行直接存取操作。对 80286 以上各类 CPU 在实地址方式下所能存取的实地址空间也指的是这个范围。640KB—1MB 这个范围统称为保留区。这个区域是内存地址空间中的只读区。其中 640KB—768KB 的区域用作显示缓冲区,共 128KB,这也是屏幕显示的映象区。这个区域的实际物理芯片在显示卡上。其容量大小取决显示方式,但通常不能超过 128KB。768—1024KB (1MB) 区用于各种适配卡的 BIOS 以及系统 BIOS 的 ROM Shadow。

§ 1.1.2 扩展和扩充内存

对于 80286 以上的各种 CPU,其寻址空间已大大超过了 1MB。此时的系统内存空间通常划分为:

- 常规内存 (Conventional Memory);
- 扩展内存 (Extended Memory);
- 扩充内存 (Expanded Memory)。

下面对各部分作简单的介绍。

1) 常规内存

与前面介绍的 8088 系统相同,常规内存也称为基本内存,占据 0—640KB 空间。80286 以上各类 CPU 在实方式下只能使用这部分内存运行自己的应用程序。

2) 扩展内存

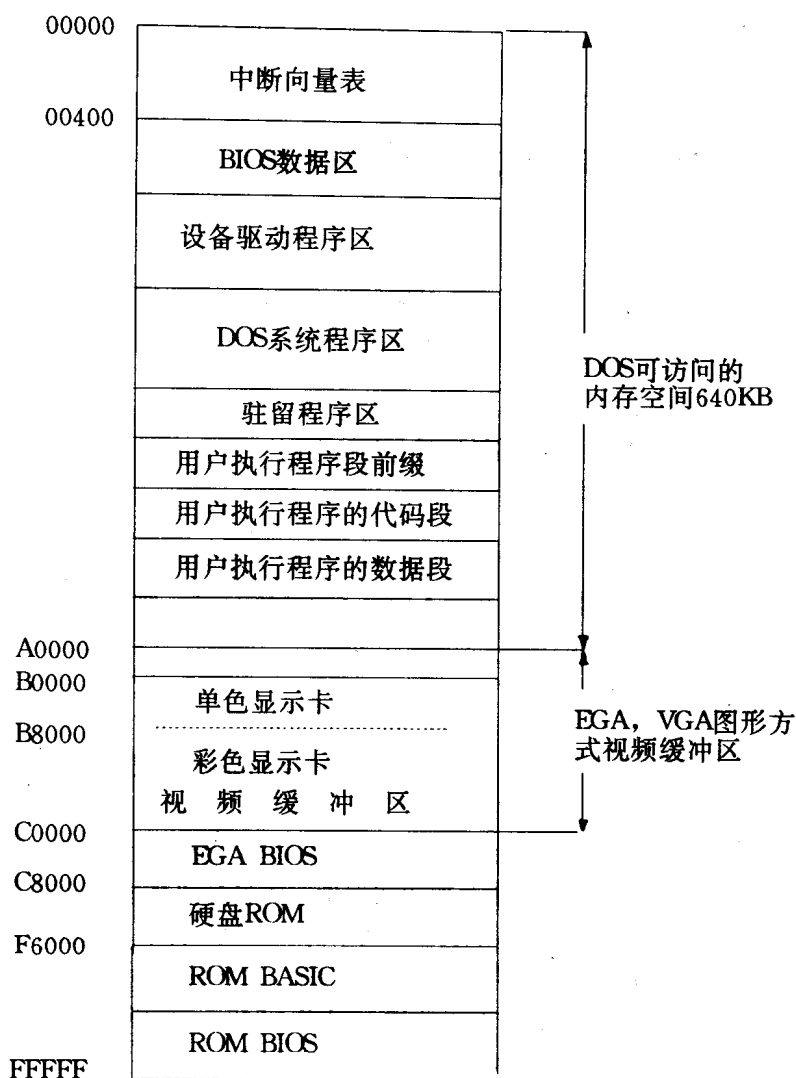


图 1—1

由于 80286 以上的微处理器的可寻址范围超过 1MB。因此在保护方式下可以使用更多的内存区。扩展内存是扩大微机内存容量，使之突破 1MB 的方法之一。扩展内存通常安装在主板上。扩展内存的地址空间分配如图 1—2 所示。

但是应该说明的是，目前只有诸如 Windows 3.0 以上的高级软件系统才真正使用了 1MB 以上的地址空间。DOS 系统通常只能采用 RAM 磁盘（虚拟盘）或 Shadow 存储器来使用 1MB 以上的扩展内存。在常规内存中运行的大多数程序（包括 DOS 本身）不能直接在扩展内存中运行，也不能使用扩展内存。为了有效、充分地利用扩展内存，4.0 版以上的 DOS 提供了扩展内存管理程序，给出了一组功能调用接口，使得扩展内存的使用变得简单可行。

3) 扩充内存

扩充内存通常是指插在扩充槽上的扩充内存板这一区域。为了使用扩充内存必须安装扩

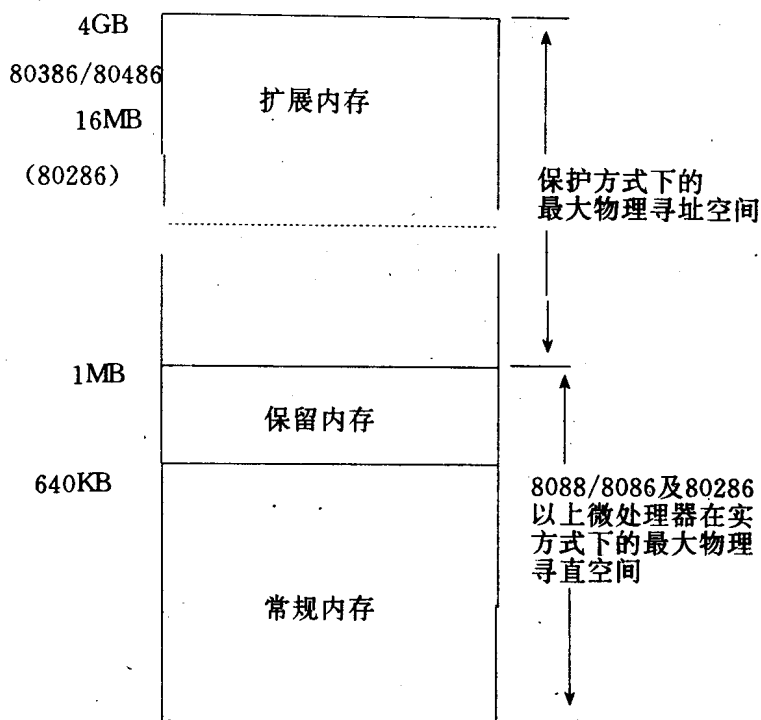


图 1—2

充内存管理程序。最有名的是 EMS3.2 扩充内存规范。它的基本思想是把所有的扩充内存分为 16KB 大小页面集。由 EMS 内存管理程序把这些页面换入或换出保留内存中的 64KB 区域 (EMS 页缓冲区), 每次可转换 4 个页面, 从而使扩充内存得到使用。扩充内存与保留内存的页缓冲区之间的关系如图 1—3 所示。

在系统设置中 BIOS 设置程序通常自动测试并给出常规内存 (或者基本内存 Base memory)、扩展内存和扩充内存的大小, 有的版本也要求用户自行设定扩展内存 (Extended memory; 或简称为 Ext. memory) 和扩充内存 (Expanded memory) 的大小。严格地说, 扩充内存和扩展内存的区别并不只是由于其物理位置不同 (如扩展内存存在主板上, 扩充内存存在扩充槽上), 更主要的是它们采用的管理程序的管理方式是完全不同的。

§ 1.1.3 存储器的工作方式

在 80286 以上档次的微机中, 有二种存储器工作方式, 即真实地址方式和虚拟地址保护方式。通常简称为实地址方式, 亦称实方式 (Real mode) 和保护方式 (Protected mode)。

在实方式下, 80286 (或更高档) 微处理器与 8086/8088 微处理器兼容, 其寻址机构、中断结构以及段的大小相同。在实方式下, 最大寻址空间为 1MB, 其物理地址由一个 20 位的段地址和一个 16 位的偏移量相加而成, 并由处理器直接产生。因此, 在实方式下, 80286 及以上的各种处理器与 8088 的软件目标代码兼容, 可以直接执行 8088 的二进制代码。在系统加电或复位时, 80286 以上的各档次微机都处于实方式下。

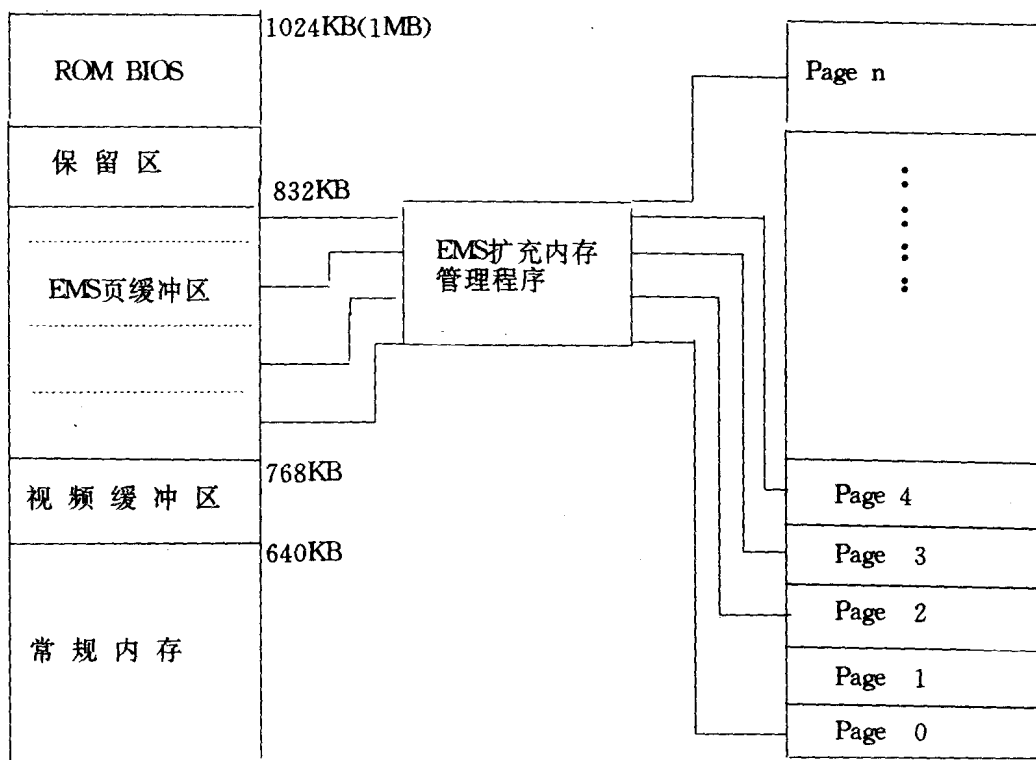


图 1-3

实方式下最大寻址空间为 1MB，主要是受到了 8088 芯片本身的限制。8088 的内存地址为 20 位 ($2^{20}=1\text{MB}$)，为了保持与 8088 的兼容性（即应用程序在二进制上兼容），80286 以上的微处理器尽管其寻址范围已远远超过 1MB，但在实方式下，它们仍被限制在 1MB 的寻址范围内。

在保护方式下，程序将使用虚拟地址。由 CPU 自动地把每个任务（进程）的虚拟地址转换成实地址。这种方式的特点是能够提供存储保护，确保每个任务的程序和数据的安全性。

80286 在保护方式下，有 1G (10^9) 字节的虚拟地址空间和 16M 字节的实地址空间。80386 在保护方式下，有 64T (10^{12}) 字节的虚拟地址空间和 4G (10^9) 字节的实地址空间。

在 80286 中，每段为 64K 字节。1G 的虚拟地址空间可分为 16000 段。因此段地址是指虚拟地址空间的 16000 个 64K 字节段中的某一个；而有效地址是指段内的偏移量。虚地址到实地址之间的转换是由 80286 片内的存储管理部件自动完成的。

在 80386 中，每段为 4G 字节。64T 字节的虚拟地址空间也可分为 16000 段，也由片内的存储管理部件实现管理。

§ 1.2 进一步理解 DOS 系统

MS-DOS 是美国 Micro-Soft 公司为 IBM PC 微机开发的磁盘操作系统, 和其他单用户、单作业的微机操作系统一样, MS-DOS 的功能主要是进行文件管理和设备管理, 其中文件系统负责建立、删除、读写和检查各类文件; 而 I/O 系统则负责驱动外围设备, 例如, 显示器、键盘、磁盘、打印机等。

§ 1.2.1 DOS 的层次结构

MS-DOS 操作系统内部分为多个层次, 最低层与系统硬件相联系, 而最高层又与用户的各种应用软件联系起来, 这样可以使用户不必关心低层的硬件, 从而更有效地使用整个系统。

图 1-4 表示 MS-DOS 操作系统的层次结构:

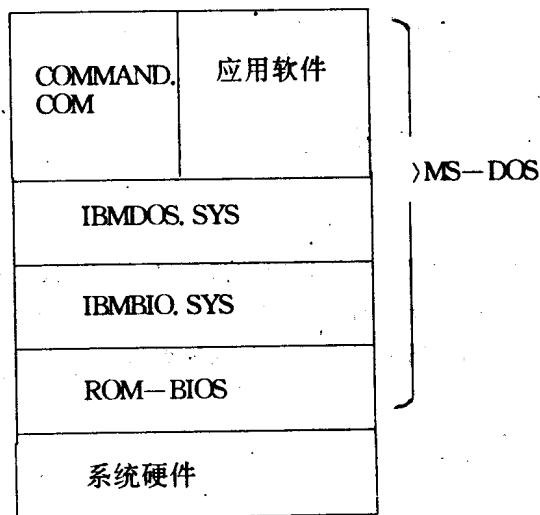


图1-4

(1) ROM-BIOS: 这一部分存放在只读存储器中, 程序被固化在 ROM 芯片中, 无法更改。ROM-BIOS 称为基本 I/O 系统, 它提供一些硬件设备的最基本、最初步的服务管理功能。

ROM-BIOS 的主要功能如下:

- 系统自检: 当系统加电后, 要由 ROM-BIOS 的自测试子程序对存储器 and 系统上的其它设备进行测试, 一旦发现问题, 则说明系统硬件有故障。

- 引导系统: 启动 DOS 操作系统, 用引导程序把磁盘上的 DOS 文件, 读入到存储器中。

- 基本 I/O 的管理: 这是 ROM-BIOS 的主要部分, 它包括一些服务子程序。用来对 IBM PC 上所有标准的外围设备, 如键盘、显示器、软盘、异步通讯口、打印机, 进行服务管理。

以提供中断服务程序形式来完成这些最基本、最初步的服务管理,很多参考书列出了 BOIS 中断调用一览表。

(2) IBMIO. SYS: 这部分是以文件 IBMIO. SYS 存放在磁盘上的,当系统启动后将它调入内存。它的任务是扩充 ROM-BOIS 的功能。

(3) IBMDOS. SYS: 这部分是 DOS 的系统的核心,存贮在 IBMDOS. SYS 文件中,由 DOS 服务性的例行程序组成,包括 DOS 软中断和 DOS 功能调用。前者是由专用中断调用的例行程序,后者是一些共享一个中断系统调用的 (21h) 的例行程序。这些例行程序,为用户提供了一套独立的硬件系统功能,它包括:

- 文件的记录和管理
- 内存管理
- 字符设备的输入/输出
- 假脱机
- 提取实时时钟

从层次的角度看,IBMDOS 中的这些例行程序要调用低层的 BIOS 设备驱动程序,如 DOS 功能 INT 21H 从键盘读入的调用 (AH=1),其内部要调用低层的 BIOS 键盘输入程序 (INT 16H),然而对用户来说,使用 DOS 的例行程序和 BOIS 中的服务程序其方式是完全一致的,都可以通过指令:

INT <中断向量号>

来实现。

调用这些例行程序的一般过程是:把功能号放入 AH 中,设定相应的入口参数,然后执行指令:

INT <软中断号>

在执行指令后,分析相应的出口参数,如果发现调用出错,则据出错类型进行相应处理。

(4) COMMAND. COM: 这是 DOS 的命令处理程序,是用户与操作系统的接口。它的任务是分析并执行在 DOS 状态下用户键入的命令,包括从磁盘上将程序加载到内存中并执行。

COMMAND. COM 可分为如下三个部分:

- 驻留部分
- 初始化部分
- 暂驻模块

驻留部分放在内存中 IBMIO. SYS 和 IBMDOS. SYS 程序的后面,并且与那些程序一样,亦成了 DOS 常驻部分。该部分主要处理 <CTRL-C>和 <CTRL-BREAK>,也处理出错和其它暂驻程序的终止和退出。

初始化部分在系统启动时,首先被加载于驻留部分之上,它主要用于处理 AUTOEXEC. BAT 批文件,完成这些之后便退出内存。

暂住部分装在内存高地址区,而且所占据的内存空间可被应用程序改作它用。这部分是 COMMAND. COM 中的命令解释程序;而每当 DOS 要用命令解释程序时,其驻留部分就先检验,看看解释程序是否仍在那儿,如果已被其它程序覆盖,则应从磁盘上重新加载到内存。

COMMAND 所接收的用户命令可分为三类:

- 内部命令

- 外部命令
- 批处理文件

内部命令是 COMMAND.COM 本身的代码实现的, 包括 COPY、DIR、DEL 等。包含在 COMMAND 的暂驻部分中。

外部命令指的是存贮在磁盘上的可执行文件。如 CHKDSK、APPEND、DISKCOPY 等。要执行这些 DOS 命令时, 首先必须将它们加载到内存中, 而一旦执行完便退出内存。

批处理文件是含有一些内外命令或其他批处理文件名的文件, 由 COMMAND.COM 每次读入该批处理文件的一行, 顺序地执行。

为了解释用户命令, COMMAND.COM 首先检查是否为内部命令, 若非内部命令则搜索具有相同文件名的外部命令和批处理文件。首先在当前目录中搜索, 然后按 PATH 指定的目录查找。在此过程中, 首先找带 COM 扩展名的文件, 其次才是带 EXE 和 BAT 的文件。

如果找到一个匹配的可执行文件, 则利用 DOS 的 EXEC 功能, 在用户可用空间建立程序段前缀 (PSP), 随后将用户程序加到 PSP 区域上面, 设置好各个寄存器, 最后将控制权交给用户程序。当程序终止时, 由 DOS 的终止功能释放所占的内存, 并将控制权交还给 COMMAND.COM。

§ 1.3 汉字操作系统

§ 1.3.1 汉字操作系统结构

汉字操作系统 CCDOS 是在 MS-DOS 的基础上, 对其中的文件管理系统 (IBMDOS) 和基本输入输出系统 (ROM-BIOS) 扩充汉字功能而成的, 使之能进行汉字文件管理和汉字设备管理, 最主要的是解决汉字的输入、显示、打印、传输这几个功能。

图 1-5 是 CCDOS 的层次结构图。

上节提到的 ROM-BIOS, 它控制着系统所必需的主要外部设备的工作。用户软件在使用键盘输入、CRT 显示及磁盘操作时, 频繁地调用 ROM-BIOS 中的各个驱动模块。因此为了响应汉字对外部设备的调用要求, 就必须在 ROM-BIOS 之外再添加用以进行汉字键盘输入、汉字显示、汉字打印等操作的驱动模块。这部分软件在编写中尽量利用原来的 ROM-BIOS 程序, 使之更为简炼。扩充部分称为 CCBios, 是 CCDOS 的核心部分。而对 IBMDOS.SYS 部分的修改, 只限于一些字符处理和引导程序的适配, 但这也是汉字内码能在原西文操作系统上运行的关键处。

§ 1.3.2 汉字的编码

1) 汉字的输入码

这是用户在键盘上敲入的各种汉字编码, 又称外码。外码是与汉字的输入方法相对应的, 即每一个汉字对不同的输入方法有不同外码。

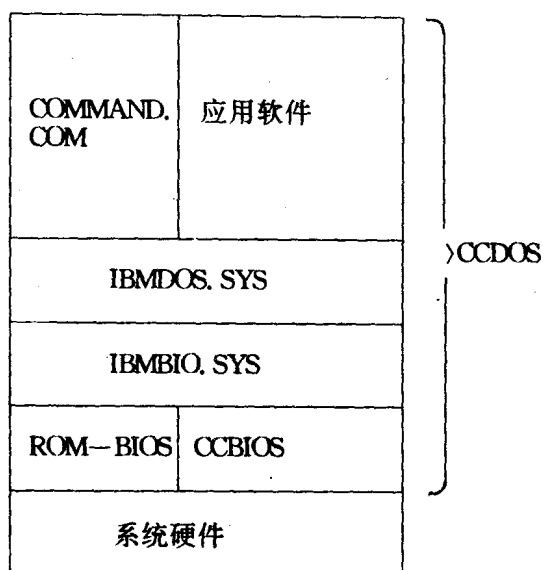


图1-5

例如：在“五笔字型”汉字输入方法状态下

“李”字的外码是 sb 或 sbf,

在“拼音”汉字输入方法下,

李字的外码是 li。

2) 汉字的内码

内码是系统中用来表示西文或中文信息的代码,在 IBM-PC 原西文系统中,每个西文字符的机内码是采用一个字节来表示的 ASCII 码,一般只使用前七位表示 128 种字符,汉字系统中的机内码的选取必须考虑到:

- 不能产生二义性,即 ASCII 码和汉字内码应严格区分
- 汉字机内码的长度应顾及显示体制和打印体制,并应尽可能短
- 应与国标 GB2312-80 汉字字符集尽量保持简单对应关系,以便对汉字字库的处理、查找。

据此,直接使用 GB2312-80 规定的国标码是不行的,因为国标码规定组成每个汉字的两个字节代码为

7	6	5	4	3	2	1	0
0	×	×	×	×	×	×	×

7	6	5	4	3	2	1	0
0	×	×	×	×	×	×	×

它们每个字节的高位均为“0”。这样,当 ASCII 和国标码同时存在时,国标码往往被误认为两个独立的 ASCII 码,从而产生二义性。

CCDOS 使用的汉字内码是一种高位为“1”的两字节内码。此方案是将 GB2312-80 规定的国标码的每个字节中的最高位置“1”作为汉字的机内码。

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
1	×	×	×	×	×	×	×	1	×	×	×	×	×	×	×

3) 代码转换

对每一种汉字输入方法，均有一个代码转换程序，把输入的每一个汉字的外码转换成统一的机内码。

(1) 区位码：指在国标 GB2312-80 汉字字符集中的区号、位号。

(2) 国标码：指在国标 GB2312-80 汉字字符集中的汉字编码。

先将区位码转成国标码，再按国标码处理。

区位码转换成国标码的方法是：

分别将十进制的区码和位码转换成十六进制，然后这两个字节分别加上 20H 就得到国标码。

以“荣”为例，它的区位码为 40，17，化成十六进制为 28H，11H 即 2811H。

“荣”的国标码 = 2811H + 2020H = 4831H。

“荣”的机内码 = 4831H + 8080H = C8B1H。

因此：

汉字的机内码 = 汉字的区位码 + 2020H = 汉字的国标码 + 8080H；

汉字的国标码 = 汉字的区位码 + 2020H。

§ 1.3.3 汉字操作系统的发展历史

电子工业部第六所研制的 CCDOS2.1 是最早的、较完整的汉字操作系统。它奠定了以后的汉字操作系统的基础，带动了我国汉字系统应用的起步，并提出了汉字系统的一些约定和规范。如屏幕底行显示汉字提示、输入法等等，并在当时得到广泛的应用，应该说这是汉字操作系统的第一次飞跃。在其后几年推出的一些汉字操作其系统技术上都没有什么大的突破。

Super-CCDOS (北大与香港金山公司合作的) 汉字操作系统的出现，带来了中文操作系统的第二次飞跃，在许多技术上有所突破；

(1) 以前汉字操作系统兼容性差。例如：对不同的显示卡，打印机就要求配不同的汉字显示驱动程序和打印机驱动程序；而 Super-CCDOS 自身能够适应不同的显示卡、打印机，克服了原来汉字系统程序繁多，安装困难之不足。

(2) Super-CCDOS 比以前的汉字操作系统占用更少内存。用户可以根据自已的需要把一种输入方法挂到汉字系统上，而不是系统把所有的输入方法都挂上去。Super-CCDOS 能够完全从内存中退出，释放汉字系统占用的内存空间、返回西文操作系统下。

(3) 与以前的汉字操作系统，Super-CCDOS 用户介面简单、明了、操作方便。

由于 Super-CCDOS 有这些优点，故目前在我国被广泛应用，可以说是当前的主流操作