

刘路放 著



C语言的窗口式 图形界面设计

——自带汉字环境的应用软件编程



西安交通大学出版社

C 语言的窗口式图形界面设计

——自带汉字环境的应用软件编程

刘 路 放 著

西安交通大学出版社

内 容 提 要

本书介绍了如何为使用C语言编写的应用程序设计一个“自带汉字环境”的窗口式图形用户界面。书中系统地介绍了VGA显示卡、鼠标、键盘和扩充存储器等设备的编程方法,并在此基础上建立了一个内含式汉字编程环境HANENV,包括一个头文件和200多个库函数。将HANENV加入Turbo C之后,就可以在编写应用程序时直接引用其库函数。这些库函数的功能包括汉字的输入、存储和显示;时钟、定时器和闪烁光标;全屏幕数据录入和画图;窗口、滚动条、按键式菜单以及代码表等窗口式界面部件等。同时在HANENV系统中还提供了计算器、调色板、文件目录窗口以及屏幕平滑移动等功能,使用HANENV编写的软件经编译后可以直接在西文DOS环境下运行而具有强大的汉字处理能力。因此HANENV系统是使用C语言编写华丽而实用的用户界面应用程序的最佳编程环境。

本书配有两张同名软盘,软盘中除了HANENV系统的所有头文件、库函数及其源程序以外,还包括若干用于字库编辑、造字的工具软件和一个使用HANENV编写的窗口式文本编辑程序。

以上两张软盘(1#,2#)随本书一起发行,每套定价60.00元。除此之外,本书还配有一套扩展字库盘,内容包括32×32点阵、36×40点阵的宋、仿宋、黑、楷体汉字库(系HANENV系统专用的横排显示汉字库,与普通中文操作系统中用于打印的竖排字库不同),每套4张软盘(3#,4#,5#,6#)共80.00元,有兴趣的读者可直接向西安交通大学出版社邮购。(邮费另加15%)

汇款请寄:西安交通大学出版社发行科

邮 编:710049

户 名:西安交通大学出版社

开户银行:西安市工商银行互助路分理处

帐 号:235-114260-90801

(陕)新登字 007 号

C 语言的窗口式图形界面设计 ——自带汉字环境的应用软件编程

刘 路 放 著

责任编辑 林全

*

西安交通大学出版社出版发行

(西安市咸宁西路28号 邮政编码710049)

西安电子科技大学印刷厂印装

各地新华书店经销

*

开本:787×1092 1/16 印张:25.25 字数:775千字

1996年6月第1版 1996年6月第1次印刷

印数:1—5000

ISBN7-5605-0831-6/TP·121 配套程序软盘,书总定价:95.00元

前 言

用户界面是现代应用程序中最重要的组成部分。封面是否新颖醒目,版面是否美观大方,操作是否方便易学,这些都是决定应用软件成败的关键之处。编写过应用程序的程序员们都知道,用户界面部分的编程工作量最大,有时可达整个程序工作量的 1/2 以上。因此,如何编写出性能优良、省工省时的界面程序,就成为应用软件设计者们共同关心的课题之一。

近来窗口式界面随着 WINDOWS 的推广而流行起来。窗口式界面摆脱了字符显示方式的限制,充分利用了图形模式的高分辨率和丰富的色彩,使得应用程序的面貌为之一新。同时窗口式界面大多采用鼠标器作为辅助输入设备,大大方便了用户的操作。但遗憾的是,目前窗口式界面主要由 WINDOWS 平台提供支持,DOS 下的应用程序如果要采用窗口式用户界面,就得自己编写所有的支撑环境,工作量相当庞大。

另外,我国的应用程序设计人员在进行应用程序界面设计时还要解决汉字的输入输出问题。通常的选择是利用某种现有的中文操作系统。这样做的优点是汉字处理独立于应用软件,所有的汉字处理工作均由中文操作系统解决,应用程序人员的负担较轻。但是要为自己的应用程序选择一个合适的中文操作系统也是一个令人头痛的问题:汉字系统的处理速度、显示和打印的效果、可否显示用于封面和标题的大尺寸汉字、汉字输入方法的选择、所用的程序设计语言和汉字操作系统之间的兼容性,以及各种汉字系统的用户群和正在开发的应用软件的用户群是否吻合等因素都应该考虑。因此,近来在应用软件开发厂商中兴起了一种被称为“自带汉字环境”的应用软件开发技术,即把中文操作系统中的汉字处理技术和应用软件有机地结合起来,使得编写出的应用软件本身就具有汉字处理的能力,在西文操作系统下就能直接处理汉字信息的输入输出工作。这种应用软件具有使用方便、体积小和运行速度快的优点,很受用户欢迎。然而要编写出性能优良的“自带汉字环境”的应用软件需要有丰富的设计中文操作系统的经验,因此这种技术目前主要掌握在一些规模较大的软件公司手中,并且多数使用汇编语言作为其编程工具。在目前,对于一般使用高级语言的应用程序设计人员来说,从零开始开发用于汉字处理的各种子程序不但是一件非常麻烦的工作,而且很难找到有关的参考资料。

本书旨在为使用 C 语言的应用程序设计人员和具有一定水平的程序爱好者们介绍有关开发“自带汉字环境”的应用软件的各种技术,包括汉字的输入输出、汉字库的组织以及开发性能良好的汉字窗口式应用软件界面。利用这些技术,我们设计了一个 C 语言的汉字处理软件包——HANENV 系统。HANENV 系统由一个头文件 HANENV.H 和近 200 个函数、若干汉字库文件组成,另外还带有若干工具软件。利用 HANENV 系统开发自带汉字环境的应用软件就像利用 C 语言的图形函数库开发图形软件一样方便。

HANENV 系统为应用软件编程人员提供的支持包括显示各种点阵的汉字或字符、区位、拼音和通用码表式汉字输入法模块(目前支持五笔字型输入法)、方便灵活的正文工作状态和闪烁光标以及多种窗口式用户界面部件,如多功能按键式菜单和图标式菜单、滚动条、代码表、全屏幕数据编辑函数和调色板等,特别是提供了设置大于实际屏幕尺寸的逻辑屏幕的功能,此时实际屏幕变为活动视窗,可以上下左右平滑移动,特别适合于设计大幅面的应用软件界面。HANENV 系统还为应用程序提供了全面的鼠标支持和访问扩充存储器的支持。另外,HANENV 系统还提供了一定的绘图能力。把这些功能结合起来,就可以设计出风格各异、性能优良的用户界面。

HANENV 系统是使用 Turbo C 2.0 开发的,也可以和 Borland C++ 联合使用。在编写 HANENV 系统的各个库函数时我们没有采用 Turbo C 的图形函数库。但是 HANENV 系统和 Turbo C 的 GRAPHICS.LIB 并不矛盾,在开发应用程序时可以同时使用以上两者编写程序。

应该说明的是,由于我们采用了直接写屏技术并充分利用了扩充存储器,所以利用 HANENV 系统编写的应用程序的汉字处理速度超过了许多中文系统的速度。

应用软件的汉字编程环境 HANENV 系统的全套库函数及其源程序、头文件、汉字库以及若干工具软件均作为本书的配套软件发行。

HANENV 系统是一个开放的软件系统,所有库函数的源程序均对读者开放。因此,读者可以根据自己的具体情况对其进行修改和扩充。我们衷心欢迎读者利用 HANENV 系统设计、编写自己的应用程序,并在自己的编程实践中对 HANENV 系统进行检验、修改和补充。我们希望读者能够就自己在使用 HANENV 系统时发现的问题或者对其所做的补充修改,例如新开发的库函数和作者进行交流,以使其更加完善。

在本书的撰写以及 HANENV 系统的设计过程中,得到了诸多老师和同志们的鼓励和帮助,在此表示衷心的感谢。在构思和写作本书的一年多里,我的父亲刘清阳教授对本书的写作提出了许多中肯的意见,李燕梅同志为本书的出版做了许多辅助性的工作,没有他们的共同努力本书就不可能问世得这样快。本书的责任编辑林全同志为本书的出版做了大量的工作,在此一并表示谢意。

刘 路 放

1996 年 6 月于西安交通大学计算机教育中心

绪 论

计算机是使用拼音文字的西方人发明的。现代计算机的典型形象就是一台显示器和一个由西文打字机演变而来的键盘。就计算机的使用而言,拼音文字的好处是显而易见的:只需使用几十个,至多百十个不同的键便足以直接输入任何文章或指令,显示时也只要考虑为数不多的符号即可。因此,无论是计算机的设计还是使用都比较简单。然而,在将计算机这一现代科技革命的结晶引入中国以后,立刻就产生了怎样将我们古老的建立在表形和表意基础上的方块汉字和计算机相结合的问题,具体说来,就是怎样解决汉字信息在计算机中的存储、交换以及汉字的输出与输入等问题。

首先,要将所有的汉字一一制作成键盘上的按键就成了一个非常困难的问题,而且即使我们造成了这种键盘(体积肯定相当庞大)之后,要想在上面很快地找出自己所需要的汉字也是相当困难的。目前解决这一问题的主要方法是限制计算机所能处理的汉字数目。80年代初制定的国家标准《信息交换用汉字编码字符集·基本集》(GB2312-80)中一共收录了6,763个常用和次常用汉字以及一些其他图形符号。考虑到《康熙字典》收录了42 174个汉字,《辞海》收录了16 534个汉字,而现在一般估计古今汉字累计已达六万多个,因此可以说上述国家标准规定的汉字集规模是太小了。因此,近年来汉字编码集的研究工作还在继续,其中比较重要的成果有国际标准化组织制定的多八位编码体系(ISO-10646),在该体系中汉字和世界各国文字统一编码。上述国家(国际)标准和编码体系提供了汉字的编码,同时也解决了汉字在计算机内部如何存放的问题。就国内目前的大多数汉字系统而言,基本上都是采用了在GB2312-80所规定的两字节汉字编码(称为国标码)上加以某种转换(为了兼容处理西文符号)而产生的变形码作为汉字机内码。

其次,由于西文的字符种类少、图象简单,为了提高显示或打印的速度,通常在设计计算机时,都把所有字符的图象转换成二值矩阵(通常称为点阵)后固化在计算机或打印机的只读存储器中。但是由于汉字的数量多、结构复杂,其点阵占用的存储量很大,将其固化在只读存储器中的方法代价较高,目前主要用于汉字打印机中。在计算机的主机中,必须采用多种灵活的手段处理汉字库的存储和点阵的读取,才能达到较高的输出速度和质量。

第三,即使采用了经过挑选的常用汉字集(如国标汉字集),对于制作汉字专用键盘来说其符号的数目还是太多了。因此,实际上在输入汉字时要采用间接的手段,使用某种输入码(码长 >1)在标准的西文键盘上进行汉字的输入工作。目前已有数百以至上千种汉字输入法问世,其中最常用的也有区位、拼音、五笔字型等十几种。

最后,人们希望计算机在处理汉字信息的同时仍保留其对西文的处理能力,即所谓“中西文兼容”,因为目前有大量的计算机软件是用西文写成的。

目前国内的计算机界基本上都是采用汉字操作系统来解决上述问题。汉字操作系统相当于一个“二传手”,由键盘输入的汉字输入码经过汉字操作系统转换成汉字机内码,再将其送入各个应用程序去处理。处理的结果再由汉字操作系统变换成汉字的点阵显示或打印出来,如图 0-1 所示。

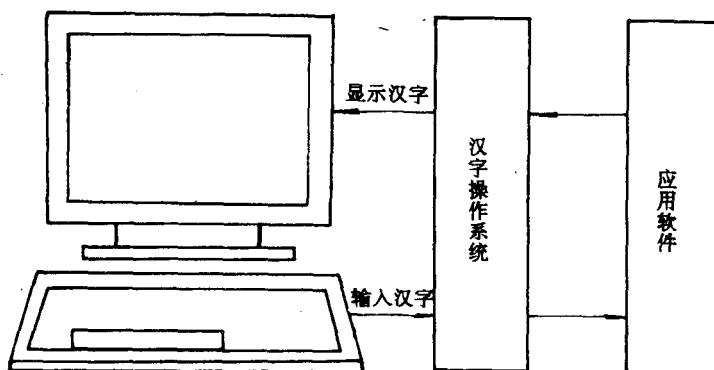


图 0-1 汉字操作系统

这样做的好处是汉字操作系统独立于应用软件。理论上说,一台计算机上只要装上一套汉字操作系统就应该可以供所有需要显示汉字的应用软件使用。但是由于汉字处理工作的复杂性,特别是目前微型计算机处理西文软件的特殊性,能够做到既兼容西文原版软件,汉字显示速度又高的汉字操作系统还很少见。现在市面上的汉字操作系统很多,各有各的优缺点和适用范围,各有各的用户群。所以,在计算机中装有几套不同的汉字操作系统供不同的应用软件使用的情况很普遍,这就给用户带来了许多麻烦。

另一方面,从应用软件开发人员的角度来看,如何为自己的应用软件选择一个合适的汉字操作系统也不容易。因此,近来兴起了一种被称为“自带汉字环境”的应用软件开发技术,即把汉字操作系统中的汉字处理技术和应用软件有机地结合起来,使得编写出的应用软件本身就具有汉字处理的能力,在西文操作系统下就能处理汉字信息。但是要编写出性能优良的“自带汉字环境”的应用软件需要丰富的设计汉字操作系统的经验,因此这种技术目前主要掌握在一些规模较大的软件公司手中,并且多数使用汇编语言编程。本书的目地就是为使用 C 语言的应用程序设计人员和具有一定水平的程序爱好者介绍有关开发“自带汉字环境”的应用软件的各种技术,包括汉字的显示、汉字库的存储、汉字的输入以及怎样开发性能良好的汉字应用软件界面。

二

要使自己编写的应用软件能够有效地处理汉字信息,必须解决好三个问题:汉字编码、汉字输出和汉字输入。一般来说,解决汉字编码问题不是应用程序设计人员要考虑的问题,而是国家标准设计委员会的任务。我们只需像大多数汉字系统的设计者一样,选用一种异形国标码即可。在本书介绍的 HANENV 系统中我们选用将国标码的两个字节的最高位都

置为 1 的汉字机内码,这也是绝大多数汉字操作系统所采用的格式,如图 0-2 所示。

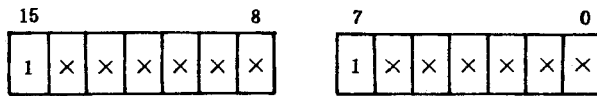


图 0-2 汉字机内码

汉字的输出又可以根据输出设备的不同分为显示、打印以及由绘图机输出等。由于绘图机等的汉字输出工作大多有专用的软件包支持,而现在的大多数打印机又都装有硬汉字库,支持汉字的直接打印输出,所以在 HANENV 系统中我们主要考虑汉字的显示问题(同时为了方便应用程序设计人员拷贝屏幕图形的需要,我们还为 HANENV 系统设计了一个屏幕图形拷贝函数,见 11.5)。为了提高汉字显示速度,我们采用直接写屏的方法显示汉字,因此利用 HANENV 编写的应用软件的汉字显示速度超过了许多汉字操作系统。在设计 HANENV 系统时我们特别考虑了现代应用程序的图形用户界面对汉字显示的要求,不但可以显示标准的 16×16 点阵汉字,还可以显示 24×24 、 32×32 、 40×36 或者任意其他尺寸的汉字和图形符号,而且在显示时还可以分别指定所要显示的汉字或字符的横向和纵向放大倍数、颜色以及背景等参数。

不同的应用软件对汉字处理的要求也不相同,大体上可以分为以下几种:一种是只需要显示少量在编写程序时已经确定的汉字,例如各种控制软件的菜单、用户界面等;第二种是在编写程序时无法预知需要显示哪些汉字,但在程序运行时并不需要由键盘输入汉字的应用场合,例如文本阅读器;第三种是汉字的输入输出功能都需要,如各种文字编辑软件、数据库和管理信息系统等。对于以上三种情况,使用 HANENV 系统编写程序时可以采用不同的策略:只需要显示少量汉字时可以使用小汉字库,这样程序结构简单、显示速度最快;如果不需要输入汉字就可以不挂接汉字输入模块,以节省内存和缩小程序体积;如果需要制作比较考究的软件封面,还可以使用 HANENV 系统提供的工具软件制作特大点阵(最大可达 96×96)的汉字字模,加工成小字库或者 C 语言的全局数组的源程序供应用程序调用。在 HANENV 系统中汉字库可以根据机器的配置情况选择装载在扩展存储器中或硬盘上,小字库还可以直接编译成目标模块而和程序连为一体,装入内存。

HANENV 系统目前配有区位、全拼、双拼、图形符号等基本汉字输入法和一个通用汉字输入法接口,只要配上不同的输入码表就可以实现各种汉字输入法(目前配有五笔字型词组输入码表)。为了方便编写 WINDOWS 风格的用户界面, HANENV 系统提供了全面的鼠标支持,包括鼠标控制的屏幕平滑移动、虚拟小键盘、按键式菜单、滚动条(scroll box)和一批基本鼠标应用函数。

除此而外, HANENV 系统还包括诸如按键式多功能菜单、时钟、定时器和代码表等较复杂的功能模块的库函数。

在开发管理信息系统等与数据库处理有关的应用软件时, HANENV 系统还可以与图文数据库系统 C-BASE(参看刘路放编著,《C 语言的汉字处理与图文数据库技术》一书,西安交通大学出版社出版)联合使用编程,以取得最佳效果。

三

HANENV 系统使用 Turbo C 2.0 开发,也可以和 Borland C++ 联合使用。为了提高汉字输出的速度,我们没有使用 Turbo C 的图形函数库。但是 HANENV 系统和 Turbo C 的 GRAPHICS.LIB 并不矛盾,在开发应用程序时可以同时使用以上两者编写程序(参看 10.1)。

同样,为了提高汉字的显示速度,我们也没有采用调用 BIOS 的显示中断的方法,而是通过直接对 VGA 显示卡的各寄存器和显示存储器(VRAM)操作来显示汉字,即所谓直接写屏。直接写屏的速度很快,但操作起来比较复杂。本书第 1 章对 VGA 显示卡的结构和编程有比较详细的说明。

一般说来,在 C 语言中调用 BIOS 中断或 DOS 功能调用可以有如下几种方法:

1. Turbo C 中有相应的处理函数,例如使用函数 kbhit 调用键盘中断,putch 调用显示器中断等;
2. 使用 Turbo C 的通用中断调用函数 int86、int86x、intdos 和 intdosx;
3. 使用伪变量和中断调用函数 getinterrupt;
4. 使用行间汇编。

第一种方法最省事,但不是所有的功能调用都有相应的处理函数;后三种方法的效果差不多,处理速度一种比一种快,但使用方便程度却是一种比一种差。作为在速度和方便之间作均衡的结果,我们选用了第 3 种方法,即使用伪变量和中断调用函数 getinterrupt 的方法处理 BIOS 中断和 DOS 的功能调用,例如鼠标器编程,调用 XMS 规范等。所谓伪变量就是 80x86 处理器芯片中的寄存器,在 Turbo C 中规定了一套符号与之对应,例如通用寄存器 ax、bx、cx 和 dx 分别使用了伪变量—AX、—BX、—CX 和—DX 表示。因此使用伪变量可以直接对寄存器操作,大大提高了程序的效率。

对于一些对处理速度要求特别高,或者要求直接对某些硬件操作的函数,例如画点函数和处理屏幕平滑移动的有关函数,我们则采用了行间汇编编写程序。在整个 HANENV 系统中,这类函数为数甚少,因为我们发现,只要恰当地设计程序的结构,使用 C 语言一样能够设计出高效率的程序来,而且程序结构清晰,易于修改和交流,可移植性好。

四

本书共 15 章,分为四篇。第 1 篇“环境与设备”包括第 1 章至第 5 章,主要介绍 HANENV 的编程基础,内容有 VGA 显示卡的工作原理和编程方法、中断 33H 及鼠标驱动程序设计、键盘缓冲区操作、扩展存储器规范(XMS)的使用以及扩展存储器的直接调用,包括相应的函数设计,以及 HANENV 系统的初始化与装配;第 2 篇“汉字处理”包括第 6 章至第 9 章,介绍各种显示汉字的函数设计、汉字库的组织 and 调用方法、汉字输入模块的设计以及光标、时钟与定时器等函数模块的设计;第 3 篇“用户界面程序设计”包括第 10 章和第 13 章,介绍如何设计 WINDOW 风格的用户界面及一批通用编程部件;第 4 篇“HANENV 系统应用”包括第 14 章至第 15 章,系统地介绍 HANENV 系统的头文件和 HANENV 系统中所有函数的使用方法。

应用软件的汉字编程环境 HANENV 系统的全套库函数及其源程序、头文件、汉字库以

及若干工具软件均作为本书的配套软件发行。关于 HANENV 系统的装配可以参看 5.2 节。由于 HANENV 系统的大部分函数的源程序已经作为本书各章的例子出现, 所以如果读者没有购买到本书的配套软盘, 也可以根据本书各章所介绍的原理自行编写其余函数。

HANENV 系统是一个开放的软件系统, 所有库函数的源程序均对读者开放。因此读者可以根据自己的具体情况对其进行修改和扩充。

目 录

绪论

第1篇 环境与设备

第1章 EGA/VGA 显示卡的基本工作原理及编程	(2)
1.1 EGA/VGA 卡的显示模式	(2)
1.2 VGA 卡的结构	(4)
1.3 BIOS 的显示器中断	(6)
1.4 VGA 卡的寄存器	(12)
1.5 EGA/VGA 卡使用小结	(26)
第2章 鼠标应用程序设计	(27)
2.1 鼠标的初始化	(28)
2.2 自制鼠标光标驱动	(30)
2.3 测试和设置鼠标状态	(40)
2.4 HANENV 系统中关于鼠标应用的其他设置	(47)
第3章 键盘操作	(48)
3.1 Turbo C 的键盘操作库函数	(48)
3.2 DOS 系统功能调用	(49)
3.3 使用 BIOS 的键盘中断编程	(50)
3.4 键盘缓冲区	(58)
第4章 扩充存储器编程	(62)
4.1 扩充存储器与扩充存储器调用规范 XMS	(62)
4.2 直接访问扩充存储器	(64)
4.3 利用扩充存储器管理规范 XMS 访问扩充存储器	(70)
第5章 HANENV 系统的初始化与装配	(80)
5.1 HANENV 系统的初始化	(80)
5.2 HANENV 系统的装配	(89)
5.3 HANENV 系统的工具软件	(92)

第2篇 汉字处理

第6章 汉字显示	(103)
6.1 显示一个像素点	(105)
6.2 显示一个字模点阵	(109)
6.3 以更快的速度显示汉字	(115)
6.4 汉字点阵的放大	(116)

6.5 HANENV 系统中的汉字(字符)显示函数族	(121)
第7章 汉字库的组织	(123)
7.1 HANENV 的汉字库结构	(123)
7.2 从汉字库中取字模点阵	(128)
7.3 HANENV 系统的字库及其生成	(130)
第8章 汉字输入模块的设计	(135)
8.1 汉字输入函数 gethan	(135)
8.2 输入法模块及其装入	(141)
8.3 拼音输入法模块的设计	(152)
8.4 通用输入法模块的设计	(159)
第9章 光标、时钟和定时器	(174)
9.1 BIOS 的时钟中断 1CH	(176)
9.2 光标与汉字的文本工作方式设计	(181)
9.3 正文工作方式	(181)
9.4 时钟与定时器	(188)
9.5 鼠标、光标和时钟函数应用小结	(194)

第3篇 用户界面程序设计

第10章 屏幕作图	(197)
10.1 HANENV 系统和 Turbo C 的图形库联合编程	(197)
10.2 横竖线、框和矩形块	(198)
10.3 构造对话框和按键	(206)
10.4 直线和曲线	(210)
第11章 图形用户界面设计	(218)
11.1 提示和对话框	(218)
11.2 滚动条	(225)
11.3 代码表	(235)
11.4 调色板	(251)
第12章 全屏幕数据编辑	(260)
12.1 基本数据编辑函数	(260)
12.2 各种类型数据字段的编辑函数	(266)
12.3 设计一个全屏幕数据录入、编辑界面程序	(269)
第13章 菜单程序设计	(278)
13.1 多功能按键式菜单及其应用	(278)
13.2 图标式菜单	(301)

第4篇 HANENV 系统应用

第14章 HANENV 系统的头文件	(311)
第15章 HANENV 系统的库函数	(336)
参考文献	

第 1 篇

环境与设备

EGA/VGA 显示卡的基本工作原理及编程

微型计算机的显示系统由监视器和显示卡组成,在台式机中监视器是一个独立的外部设备,而在笔记本机、笔式机中则与主机装配在一起。显示卡是插在主机中主板上的一块插卡(在某些计算机中显示卡的有关线路被直接设计在主板上,因此没有独立的显示卡),计算机对显示屏幕的所有操作都是通过显示卡进行的。显示卡和监视器是配套使用的,一种显示卡只能配用一种监视器。

显示卡的种类很多,例如 MDA 卡、CGA 卡、EGA 卡、MCGA 卡以及 VGA 卡等等。但目前流行得最广的是 VGA 卡或 VGA 的扩充兼容卡(例如 SVGA 卡、TVGA 卡等),因为 VGA 卡不仅显示功能强大,而且还可以兼容 MDA 卡、CGA 卡、EGA 卡的所有显示模式。HANENV 系统的汉字显示就是针对 VGA 显示卡设计的。应该说明的是,由于 VGA 卡高度兼容 EGA 卡,所以在大多数资料中是将这两种显示卡放在一起介绍的,通称为 EGA/VGA 显示卡。本书介绍的 HANENV 系统的绝大部分内容只需略加修改(主要是在屏幕尺寸方面)也可以应用于 EGA 显示卡。

1.1 EGA/VGA 卡的显示模式

EGA/VGA 显示卡的显示模式可以分为字符显示模式和图形显示模式两大类,共 15 种,见表 1-1。

字符显示模式是为了快速显示西文字符而设计的,不宜用来直接显示汉字。通常显示汉字需要在图形模式下将汉字的字模作为图形块逐点画出。从表 1-1 中可以看出,在 VGA 的各种图形显示模式中模式 12H 最适合于显示汉字。模式 12H 的分辨率最高,可达 640×480 点,如果使用 16×16 点阵的汉字库,行间距取两点,显示 25 行汉字一共需要 $(16+2) \times 25 = 450$ 线,还剩下 30 线可以用作汉字输入的提示行。每行可以显示 $640 \div 16 = 40$ 个汉字,或者 80 个西文(半角)字符,和常用的字符显示模式(如模式 03H)正好相同。模式 12H 允许在 $262\ 114(256K)$ 种颜色中选择 16 种颜色同屏显示,色彩比较丰富。因此和大多数汉字操作系统一样,在设计 HANENV 系统时我们也选用 VGA 的模式 12H 作为工作模式。但应注意 12H 为 VGA 显示卡所独有的显示模式。因此如果需要在使使用 EGA 显示卡的计算机上

实现汉字显示,可以选用图形模式 10H。该模式和模式 12H 最接近,只是分辨率略低。在模式 10H 下如果仍然选用 16×16 点阵的汉字字模和两个像素的行间距,那就只能显示 19 行汉字了,除去一行提示行,只能显示 18 行正文。

表 1-1 VGA 卡的显示模式

模式	类型	分辨率	颜色数	可用显示卡				显示地址
00H	字符	40×25	16	CGA	EGA	MCGA	VGA	B800
01H	字符	40×25	16	CGA	EGA	MCGA	VGA	B800
02H	字符	80×25	16	CGA	EGA	MCGA	VGA	B800
03H	字符	80×25	16	CGA	EGA	MCGA	VGA	B800
04H	图形	320×200	4	CGA	EGA	MCGA	VGA	B800
05H	图形	320×200	4	CGA	EGA	MCGA	VGA	B800
06H	图形	640×200	2	CGA	EGA	MCGA	VGA	B800
07H	字符	80×25	2	EGA	VGA			B800
0DH	图形	320×200	16	EGA	VGA			A000
0EH	图形	640×200	16	EGA	VGA			A000
0FH	图形	640×350	2	EGA	VGA			A000
10H	图形	640×350	16	EGA	VGA			A000
11H	图形	640×480	2	MCGA	VGA			A000
12H	图形	640×480	16	VGA				A000
13H	图形	320×200	256	MCGA	VGA			A000

显示模式的设置可以通过调用 BIOS 的显示中断的 00H 号功能进行,其调用方法如下:

入口寄存器设置:

AH = 00H

AL = 模式号

返回寄存器:无

程序 1-1 设置 VGA 显示模式 12H 和 03H。

在 HANENV 系统的初始化函数 init_hanenv() 中有如下程序段:

```

/* -----
   设置 VGA 模式 12H(640 行×480 列×16 色)
   ----- */
_AH = 0x00;
_AL = 0x12;
geninterrupt(0x10);

```

用于设置 VGA 的图形显示模式 12H。

在函数 close_hanenv() 中有如下程序段:

```

/* -----

```

设置 VGA 模式 03H(25 行×80 列的字符显示模式)

```
----- */
_AH = 0x00;
_AL = 0x03;
geninterrupt(0x10);
```

用于恢复 VGA 的字符显示模式 03H。

1.2 VGA 卡的结构

下面以模式 12H 为例介绍 VGA 卡的基本结构。

显示存储器(VRAM)用来存放要在屏幕上显示的数据,由 256K 字节的动态随机存储器构成,分为 4 个颜色平面。这 4 个颜色平面各有 64K 大小,在内存中占用相同的地址空间,起始地址为 A000:0000。在显示时屏幕上的一个象素由 4 位表示,每个颜色平面 1 位,因此每个象素可以是 16 种颜色中的一种。显示存储器中的每个字节存放相邻的 8 个水平象素。颜色平面的映射原理如图 1-1 所示。

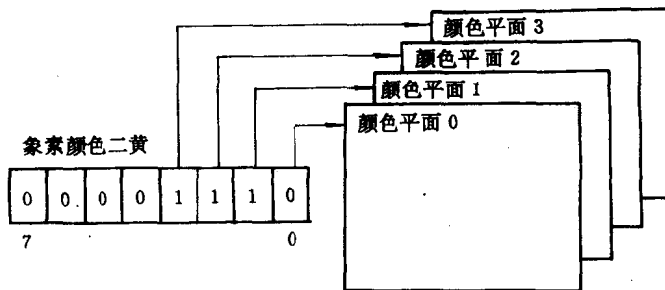


图 1-1 颜色平面的映射原理

具体说来,显示屏幕上的象素位置(X,Y) 到显示存储器的位地址之间的转换可以按下列公式进行计算:

$$\text{字节地址} = Y \times 80 + X/8$$

$$\text{位地址} = 7 - (X \bmod 8)$$

存放在位平面中的象素颜色数据还要经过数模转换寄存器(DAC) 的处理才能变为屏幕上的实际色彩,其原理如图 1-2 所示。

从图 1-2 中可以看出,VGA 显示卡通过显示存储器 4 个颜色平面的 4 位数据来选择调色板寄存器,调色板寄存器的低 4 位与由颜色选择寄存器的 C4、C5、C6 和 C7 位组成的高 4 位一起形成 DAC 颜色寄存器的八位地址,再由 DAC 颜色寄存器的值决定最终在屏幕上显示的象素的颜色。DAC 颜色寄存器的字长为 18 位,分别用其中的 6 位表示红、绿、蓝三种原色的分量。在设置模式 12H 时,16 个调色板寄存器和 256 个颜色寄存器分别被置为系统省缺值,见表 1-2。

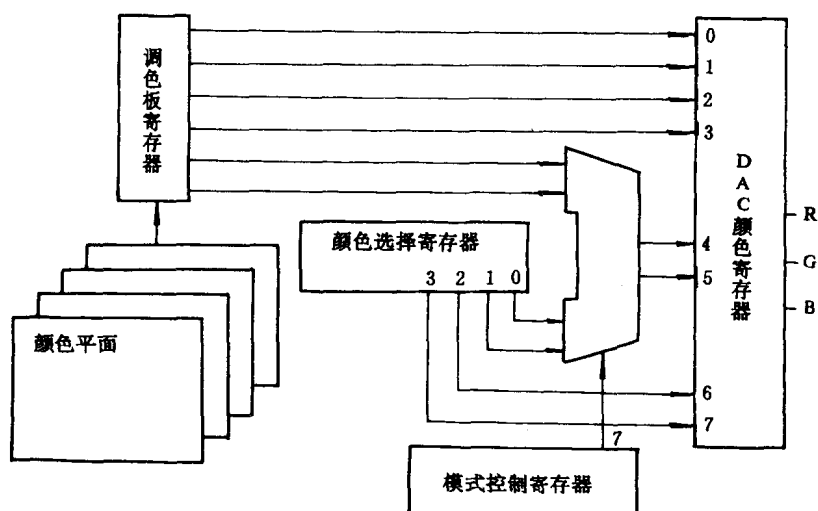


图 1-2 VGA 卡的颜色控制原理

表 1-2 模式 12H 的颜色省缺值

颜色号	颜色名称	调色板寄存器的值	红	绿	蓝
0	BLACK(黑)	0	0	0	0
1	BLUE(蓝)	1	0	0	42
2	GREEN(绿)	2	0	42	0
3	CYAN(青)	3	0	42	42
4	RED(红)	4	42	0	0
5	MAGENTA(洋红)	5	42	0	42
6	BROWN(棕)	6	42	42	0
7	LIGHTGRAY(浅灰)	20	42	42	42
8	DARKGRAY(深灰)	56	0	0	21
9	LIGHTBLUE(亮蓝)	57	0	0	63
10	LIGHTGREEN(亮绿)	58	0	42	21
11	LIGHTCYAN(亮青)	59	0	42	63
12	LIGHTRED(亮红)	60	42	0	21
13	LIGHTMAGENTA(亮洋红)	61	42	0	63
14	YELLOW(黄)	62	42	42	21
15	WHITE(白)	63	42	42	63

除了显示存储器以外,VGA 卡中还有 CRT 控制器、串并转换器、属性控制器、时序发生器等部件,其结构见图 1-3。