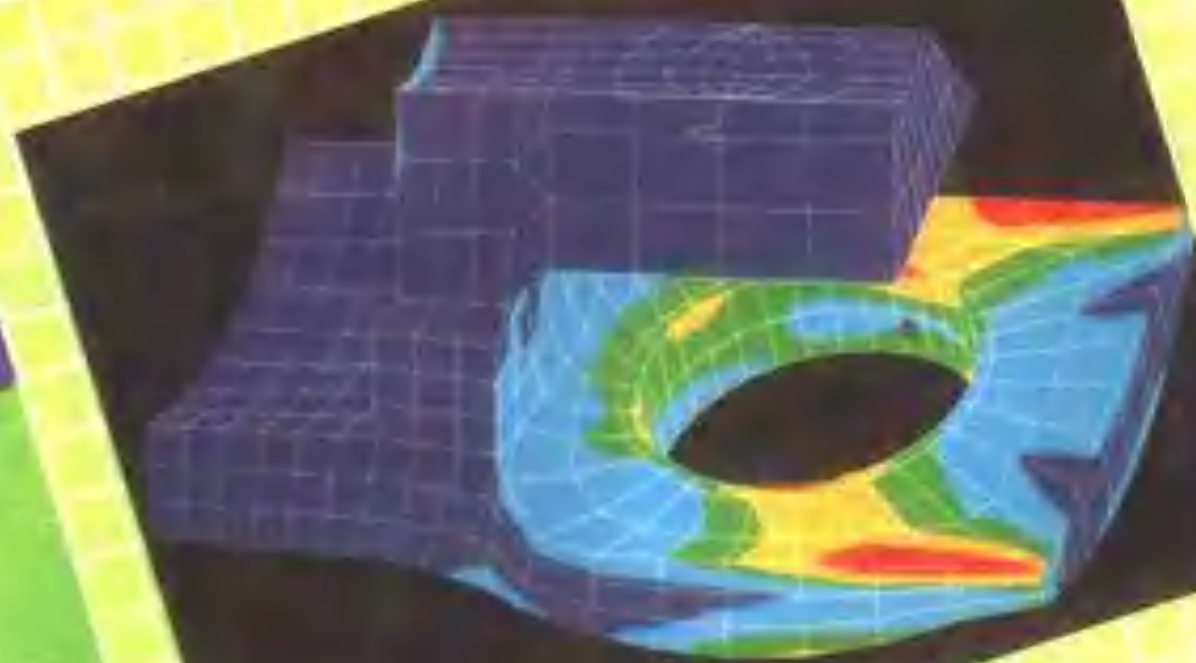


COMPUTER GRAPHICS DISPLAY TECHNIQUE

王知行等 编著

计算机图形显示技术



哈尔滨船舶工程学院出版社

970786

TP391.4
1082

:B

计算机图形显示技术

王知衍等 编著

哈尔滨船舶工程学院出版社

(黑)新登字第9号

内 容 简 介

本书系统地介绍了计算机图形显示技术的主要内容和最新成就。全书分八章:详细介绍CRT、LCD、等离子体显示和EL四种主要类型图形显示装置;EGA、VGA等微型机图形控制器,以及TIGA、8514/A、XGA等图形标准;Intel 82786、TMS 340×0图形处理器;“PC兼容”和RISC工作站;GKS、PHIGS、CGM、CGI和IGES等图形软件接口标准;微型机视频编程汇集以备应用时速查。

本书适合于高等院校与图象图形有关专业的本科生、研究生和教师以及广大科技人员使用,也可以用作教学参考书。

计算机图形显示技术

王知衍等 编著

*

哈尔滨船舶工程学院出版社出版

新华书店首都发行所发行

黑龙江省新华印刷厂印刷

*

开本 850×1168 1/32 印张 9.75 字数 240 千字

1993年3月 第1版 1993年3月 第1次印刷

印数:1—3000册

ISBN 7-81007-200-5/TP·9

定价:9.60元

序

当今科学技术一日千里,计算机领域尤其如此。面对信息爆炸的形势,面对堆积如山的文献,人们很自然地希望能缩短熟悉、掌握知识的过程。讲课、培训是一种好办法,不过它不是随时随地能实现的,因此适当的书籍依然是不可取代的。尽量有效地吸取已有成果的营养,这个原则无论是在科学研究上,还是看书学习上都是正确的。

目前计算机书籍不少,相对来说,详细叙述性的手册类的更多。在面对新型设备时,只有系统手册、操作手册。初次接触厚厚的手册时,心理上就有压力。面对具体、丰富的内容很难建立起一个总体概念,无论是理解和掌握都比较困难。因此一般都有指南性质的书来简要地加以说明,它们帮助人们着重理解、建立清晰的框架,克服只见树木不见森林的弊病。当然,某个具体问题的解决,还是要依靠手册类的书籍。不过这时,人们有清醒的认识,再钻到厚厚的条文解释中去时就有目标而不觉得枯燥。

计算机图象图形是目前的热门,是计算机走向新时代的标志。作者通过多年的 CAD 以及图象图形项目的开发研究,感到应该有一本书做一些综述性的工作。本书就是试图尽可能系统地、广泛地反映最新知识和成就,回答人们普遍关心而现成资料缺乏或零碎的问题。

本书对微型计算机和图形工作站的图形显示技术进行阐述,具体内容包括: CRT、LCD、等离子体显示和 EL 四种主要类型图形显示装置; EGA、VGA 等微型计算机图形控制器以及 TIGA、8514/A、XGA 等图形标准; Intel 82786、TMS 340×0 图形处理器; “PC 兼容”和 RISC 工作站; GKS、PHIGS、CGM、CGI 和 IGES 等图形软件

接口;最后是微型计算机视频编程汇集以备应用时速查。

本书对显示装置作了比较详细的介绍,其原因之一是,它是图形显示的基础,近年来发展势头迅猛;再则,见到的资料不多,人们普遍在互相转告着一些名词术语,但不清楚是什么意思。

本书也花了大量篇幅介绍图形处理器、工作站和图形软件接口标准。内容新,又比较集中,特别适合关心新技术又想了解比较全面情况的人士。

相对来说,本书没有花太多的篇幅介绍微型计算机的图形及其编程。其原因是,目前这方面的书籍资料很多,需要具体编程的人尽可以找那些工具书来看,一般来说,它们篇幅浩大,足以超过本书的全部。因此本书不想重复这些工作。作为本书的特点,作者是这样来处理的;把微型计算机图形系统及其发展,以及编程要点作一简明介绍,这主要由第三、四章完成;然后,第八章以视频编程汇集的形式提供给读者速查之用。就是说,一方面努力使读者有一个清晰的认识,另一方面又兼顾编程实用的要求,实际上,第八章是供那些有编程能力的人用的。对那些没有编程经验的人,可以阅读第三、四章,然后找一本详细的编程手册来实际编程,这时他可以查阅本书第八章以节省时间。

本书一共八章,其中第五章由曾明执笔,第八章由王学林执笔,其余由王知衍执笔,全书由王知衍修改定稿。

本书适合于高等院校与图象图形有关专业的本科生、研究生和教师以及广大科技人员使用,也可以用作教学参考书。

由于作者水平有限,所涉及的范围虽然远不全面、但已相当广泛,加上计算机发展神速,书中错误和不当之处不可避免,恳请广大读者不吝赐教。

作者

于哈尔滨科技大学

1992. 9.

目 录

第一章 概 述	1
1.1 计算机图形图象	1
1.2 微型计算机图形显示回顾	3
1.3 图形显示面临的问题	4
1.4 图形处理器	6
1.5 图形软件标准	8
第二章 图形显示装置	11
2.1 阴极射线管装置	11
2.2 液晶显示装置	26
2.3 气体等离子体显示装置	36
2.4 场致发光显示装置	53
2.5 显示装置展望	65
第三章 图形控制器	70
3.1 单色显示和 CGA	71
3.2 EGA 和 CGE400	87
3.3 MCGA 和 VGA	94
第四章 图形标准的进展	124
4.1 Super VGA	125
4.2 TIGA	137
4.3 8514/A	141
4.4 XGA	147
第五章 图形处理器	156
5.1 Intel 82786 图形协处理器	156
5.2 TMS 34010 图形系统处理器	165

5.3 TMS 34020 图形系统处理器	178
5.4 视频 RAM	192
第六章 图形工作站	195
6.1 图形系统的发展	196
6.2 “PC 兼容”工作站	203
6.3 RISC 工作站	213
第七章 图形软件接口标准	223
7.1 图形程序设计标准	224
7.2 图形元文件	235
7.3 工作站和设备的接口	237
7.4 应用程序接口	239
第八章 微型计算机视频编程汇集	251
8.1 视频 BIOS 功能提要	252
8.2 标准 IBM 视频模式及兼容显示	254
8.3 视频 BIOS 功能说明	256
术语索引	283
参考书目及文献	297

第一章 概 述

计算机图形(Graphics)与图象(Image)技术目前正以飞快的速度向前发展,它使计算机以一个崭新的面貌出现,广泛应用到国民经济的各个领域。无论是卫星发射、工业控制、设计制造,企业管理、科学教育、地理地质、气象、医学、军事,还是广告、影视,就是计算机本身操作环境,比如用户图形界面 UGI(User Graphics Interface)到处可见计算机图形图象技术的魅力。

计算机摆脱了人们习惯的看法,认为它不过是一种和一大堆枯燥数字有关的专门化设备。由于计算机多媒体(Multi-media)技术的崛起,声、像及其他和人类感观功能相应功能的引入,使得计算机成为人们亲近的宠物。其中图形图象和人类的视觉相应,而视觉又是人们从外界获取信息的主要渠道,使得计算机图形图象首先成为人们不遗余力去发展的目标,至今已获得巨大的成功,而其发展速度丝毫不减。

随着国际上在高清晰度电视 HDTV (High Definition Television)技术方面的激烈竞争,它和计算机图形图象技术两者互相促进,推波助澜,在人们面前展现了一幅波澜壮阔的发展形势。可以断言,图形图象技术会有长足的进步和明显的突破,成为人们不可缺少的朋友。

1.1 计算机图形图象

计算机图形和图象是互为联系的两个概念。我们往往说,图形

是矢量结构,而图象是点阵结构。举例来说,描述一个圆,从图形角度来说主要是两组数据,圆心坐标及圆半径;而从图象来说,则是阵列象素(比如圆周上的各象素为明,背景象素为暗)。当然,这里没有涉及颜色等属性,也没有讨论三维复杂形体,但基本思想是相类同的。可以说,图形图象是一个事物的两个侧面,或两种描述方法。这也就是人们时常混用的原因。

与计算机图形、图象相关的学科有:计算机图形学(Computer Graphics)、图象处理(Image Processing)、模式识别(Pattern Recognition)和计算几何。计算机图形学研究用数字计算机生成、处理和显示图形。图形及其处理主要分两类,一类是二维工程图、地图、曲线图表等线条;另一类是三维曲面、实体等类似真实物体照片的影调图。要生成图形,就要有原始数据或数学模型,比如输入一个球心坐标及其半径、光照条件、色彩等,经计算机计算处理后形成一个具有真实感的立体圆球。计算机图形大量应用在计算机辅助设计(CAD)等方面。

图象处理将客观世界中原有物体的映象处理成数字化图象,研究如何消除噪声,压缩图象数据以便传输和存储,用对比增强技术突出图象中某些特征,用复原技术使模糊图象清晰以及重构图象等等。图象处理广泛应用于卫星遥感图片、工业射线探伤、人体CT扫描等等。

模式识别研究分析和识别输入的图象,找出其中蕴含的内在联系或抽象模型。计算机图形学着重讨论将数据和几何模型变成图象,而模式识别讨论从图象提取数据和模型,两者互为逆过程。举例来说,计算机图形学技术可把输入的圆心坐标和半径处理成在屏幕或图纸上表现出的圆;而模式识别技术则可从对图纸上圆的输入(扫描输入或摄象输入)中提取出圆的概念及其圆心坐标和半径。

计算几何专门研究几何模型和数据处理,着重讨论几何形体的计算机表示和分析、综合,研究建立几何形体数学模型的各种方

法,提高算法效率,在计算机内存储和管理模型数据等。

科学技术的发展使得各学科相互交叉和渗透,以上几门学科的界线也越来越模糊。因此在本书中虽然是讨论图形显示技术,却难免和上述学科有各种各样的联系,但侧重面放在计算机图形方面。

1.2 微型计算机图形显示回顾

1950年,第一台计算机控制的显示器安装在美国麻省理工学院的 Whirlwind I 型计算机上。该显示器使用了阴极射线管(CRT),只能产生简单的图形。短短 40 年间,特别是随着微型计算机和计算机工作站的迅速发展,图形显示装置、图形控制器、图形处理器、图形系统及相关编程技术和接口体系都获得了令人瞩目的成果,同时也面临着进一步的挑战。

IBM 公司最初推出 PC 机(个人计算机)的时候,它的单色显示适配器 MDA(Monochrome Display Adapter)既没有彩色也没有图形。如要需要图形,也只提供彩色图形适配器 CGA(Color Graphics Adapter),它在 320×200 象素分辨率时有四种颜色,在 640×200 象素分辨率时就只有两种颜色可用。正文显示粗糙但明了。

Hercules Computer Technology 公司通过迅速提供适应正文/图形显示的技术打开了市场。此时正文字体分辨率高,为 720×348 象素,图形显示仍为单色。它成为 MDA 本应该充任的缺省标准。过了几年,IBM 报之以增强型图形适配器 EGA(Enhanced Graphics Adapter),分辨率为 640×350 象素,16 种色彩。但是当时 EGA 卡太贵,很快由兼容产品厂商在 EGA 功能基础上扩展了性能,把价格降到可竞争的水平。

EGA 目前已成为基本的显示标准之一。随后的视频图形阵列 VGA(Video Graphics Array)克服了 EGA 的一些弱点,把分辨率提

高到 640×480 像素, 色彩仍为 16 种。

正当 IBM PC 的图形系统不断改进的时候, 其它公司开发了把图形当作系统基本单元的计算机。苹果公司推出 Mac, 显示为 512×342 像素分辨率、黑白式, 并在 ROM 中固化了标准图形程序。Amiga 1000 从 320×200 和 640×200 像素分辨率图形开始, 加上改进图形性能的硬件支持。Atari ST 也包括了图形, 随后加上了硬件支持。事情清晰地表明, 图形已扩展到非 IBM PC 用户。

图形显示装置也发生了很大的变化, 四种主要的显示技术在激烈竞争, 它们是阴极射线管 CRT (Cathode Ray Tube)、液晶显示 LCD (Liquid Crystal Display)、等离子体 (Gas-plasma) 显示和场致发光 EL (Electro Luminescent) 显示。在屏幕尺寸、分辨率、色彩等方面努力发展。

CRT 已有 80 多年的历史, 即使这样仍有很大的发展前途。CRT 的高可靠性及性能价格比使它成为目前唯一能产生大幅全彩色图象的相对廉价的系统。世界上有数以百万计的阴极射线管投入使用。高分辨率多频彩色监视器的发展也说明 CRT 不可能在短期内退出历史舞台。

液晶显示 (LCD) 近年来异军突起, 广泛应用于膝上计算机 (Laptop) 和笔记本式计算机。LCD 有许多优点: 尺寸小、重量轻、耗电省、分辨率良好 (通常 640×480 像素)。但要制造和 CRT 一样的显示尺寸和效果则相当困难。最有希望的是近来开发的有源阵 (Active-matrix) 或薄膜晶体管 TFT (Thin-Film Transistor) LCD。

1.3 图形显示面临的问题

虽然从今天的标准来看, 最初的 PC 图形太原始了, 但是它打开了计算机进入崭新的一类应用的窗口。然而, 在第一批图形应用程序可用不久, 问题就暴露出来了。速度问题特别突出, 特别是在

4.77MHz PC 上。CGA 有限的彩色和低分辨率也严重限制了显示效果。

EGA 克服了 CGA 在彩色和分辨率方面的一些限制,同时也加剧了速度问题。CGA 的 32KB 视频内存变成了 EGA 的 112KB。EGA 需要专门的显示刷新,这就降低了对视频内存的存取速度。此外,EGA 需要 CPU 依次访问四个显示平面(表达 16 种颜色),因此速度更受影响。

除了速度问题之外,EGA 还有编程要求。软件开发必须支持三种主要的显示系统(CGA、Hercules 和 EGA)。三种显示需要用不同的技巧来获得最高的速度。图形显示速度非常依赖编码质量,因此所有图形支持是用汇编语言做的,编程的难度显而易见。

EGA 有一些寄存器是只写的,一旦设置完毕,随后不能读回来观察状态如何。这就是说,程序必须在低地址内存保存寄存器所含有的信息。但是并不是所有软件使用或更新这些信息,这样就产生了不兼容性。尤其是用驻留内存 TSR(Terminate-and-Stay-Resident)程序时更是这样。另一个问题是,有些程序利用 EGA 垂直回扫中断来触发事件,由于有错误而不能正确处理。

正当软件开发者大伤脑筋时,用户需要更高的速度和分辨率。这使图形显示更诱人更有用,而软件开发工作更加困难。上述 EGA 的问题已经不少,到了 VGA 情况更加复杂,除了 640×480 像素分辨率尚能标准处理,其他高于它的各种分辨率系统由于兼容厂家不同,设计使用各异,困难是可想而知的。

所有这些归结到标准的窘境。不同的委员会和厂家开发图形描述和显示的标准,而后是图形交换的标准。标准不是有利于工业生产的一、二种,各种因素影响标准:

为了有一种标准,需要大量产品来支持它;而为了有产品,必需有标准。

为了有效地推动 CGA、EGA 及 Hercules 等,要有针对硬件的标准;为了支持高效率编程,要有针对软件的标准。

但是,这些大公司付出兼容性及标准推广的代价,而促成自己的标准,达到市场开发的阶段,此时很难收起自己的一套,支持竞争对手来实现一种标准。

这样,可以毫不夸张地说,已经有几十种图形标准,从单个产品标准到 IEEE 计算机图形元文件,五花八门。这种情况始于 1988 年,目前已有 XGA、TIGA 和 8514/A 等标准,情况还在继续。

因此,应用计算机图形的用户,仍然需要对图形系统进行一番研究,有的还不得不从图形的基本软件进行开发来适应自己的需要。

1.4 图形处理器

目前,人们利用具有更快速的 CPU(如 Intel 80386 80486)的计算机,当图形分辨率、色彩数增加时,用户对图形显示的要求又进一步提高了。这不仅仅是速度问题,好的图形同样需要大量编程。幸运的是,现在有些基本图形功能已经建得很好,它们已成为称作图形处理器 GP(Graphics Processor)这一类新处理器的硬件核心。

GP 具有基本图形运算功能,以硬件方式执行。一个处理器周期所做的工作顶得上软件执行几条指令、花费 20 个或更多的周期。GP 也加有一些很有用的功能,特别是位块传送(BitBlt)。这种操作可以把内存中任意地方的任意位块搬移到任何地方,而无需受字节或字边界的限制。因为图形象素是位确定的单元,BitBlt 能快速简便地在屏幕上移动图形项。

有两个因素有助于降低图形显示的成本。一个是技术方面的,大规模生产可以降低芯片和显示的费用,增加半导体集成度,减少元件数,此外还增加了功能,改善了性能。另一个是用户方面的,一系列管理研究表明,图形及快速响应提高了用户生产率。

由于费用的减少和管理水平的提高,图形显示不再是昂贵的了,它已成为一种关键工具,促使在软硬件方面持续地迅速发展。

IBM 的 PGA(Professional Graphics Adapter)卡是初期 PC GP 的一种,采用 Intel 80286 处理器,提供 640×480 象素分辨率,可在 4096 色调色板中取 256 色。虽然想法对头,但价格和性能不敢恭维,PGA 并不成功。然而局面终于打破了,利用处理器的其他控制器纷纷出笼。早期的另一种控制器使用 Hitachi(日立)7220 芯片,比 PGA 快,但是与通用处理器相比其局限性大。

由于采用处理器,又产生了新的要求。每一种新的 GP 与主计算机 HP(Host Processor)的软件接口不同。为了使 GP 做更多的事,应用程序需要 HP 和 GP 之间的软件协议,使 HP 规定 GP 做些什么。这就是说,在 GP 画一条线时,HP 必须以 GP 认识的形式传递线的起点、终点、颜色、宽度及线型。这就迫使每个图形应用程序要有使用不同接口的各显示控制器的程序编码。

当 GP 控制器推出市场时,GP 和 CGA、EGA 及 Hercules 等的组合使得软件问题十分突出。紧跟着不兼容产品、不兼容软件接口、多重图形软件“标准”的爆炸性发展,一般用户普遍感到困惑。只要开发或使用图形,就要付出相当的努力。

由于 Microsoft Window 2.0 及 OS/2 Presentation Manager 为应用软件建立一系列规则,达到单一接口与虚拟图形设备相连,图形环境开始稳定下来。而在这之前 Window 1.0 版设计时不考虑 GP 的能力,用 GP 时并不见性能明显增强。

除此之外,象 Turbo Pascal 4.0 这样的产品已加入了图形支持,软件工具开发倾向支持 Microsoft Window 和其他窗口接口。IEEE 完成一系列图形标准,它们有广泛的超派别的支持。

人们或多或少认识到,所有产品和标准应该和图形处理器有一种有效的接口。这种想法在 1988 年已广泛传播,可以相信以后的产品会有效地支持 GP 接口。

1.5 图形软件标准

微型计算机的图形软件标准由于历史的原因比较纷乱多样。相比之下,图形工作站就有条理多了。图形工作站相对来说是一种图形性能很强的昂贵设备,它大量应用在计算机辅助设计等方面,程序量很大。由于图形输入输出设备种类繁多,原理不同,性能参数差别大,并且图形应用程序种类也越来越多,开发成本越来越高。为了降低开发各类应用软件成本,并使程序有良好的可移植性,软件的标准便首当其冲。因此,在工作站上从一开始就实施严格的标准化措施。

图形软件标准是指系统中各界面之间进行数据传递和通信的接口标准以及供 CAD 等应用程序调用的子程序功能及格式标准,前者称数据界面标准,后者称子程序界面标准。

对于交互图形系统而言,可移植性可以表达为:

- 应用程序在不同系统间的可移植性。
- 应用程序与图形设备的无关性(独立性)。
- 图形数据的可移植性。
- 编程员的适应性,即编程员可以方便地为不同的系统编制图形程序。

为了实现上述要求,交互图形系统有四个主要环节、或称四级层次上需要实现标准化,见图 1-1。

(1)应用程序与它所处理的几何数据及非几何数据之间的数据接口。

这个接口标准化有利于不同 CAD/CAM 系统间的信息交换。其中最有影响的是初始图形交换规范 IGES (Initial Graphical Exchange Specification),其它还有法国宇航公司的 SET 标准和德国的 VDAFS 标准等。这层标准是系统四级层次的最外层。

(2) 应用程序与图形软件包之间的接口。

这个接口标准化可以实现不同系统间应用程序在源程序级的可移植性。自国际信息处理协会 IFIP 在 1974 年 8 月召开 5.2 组工作会议以来,德

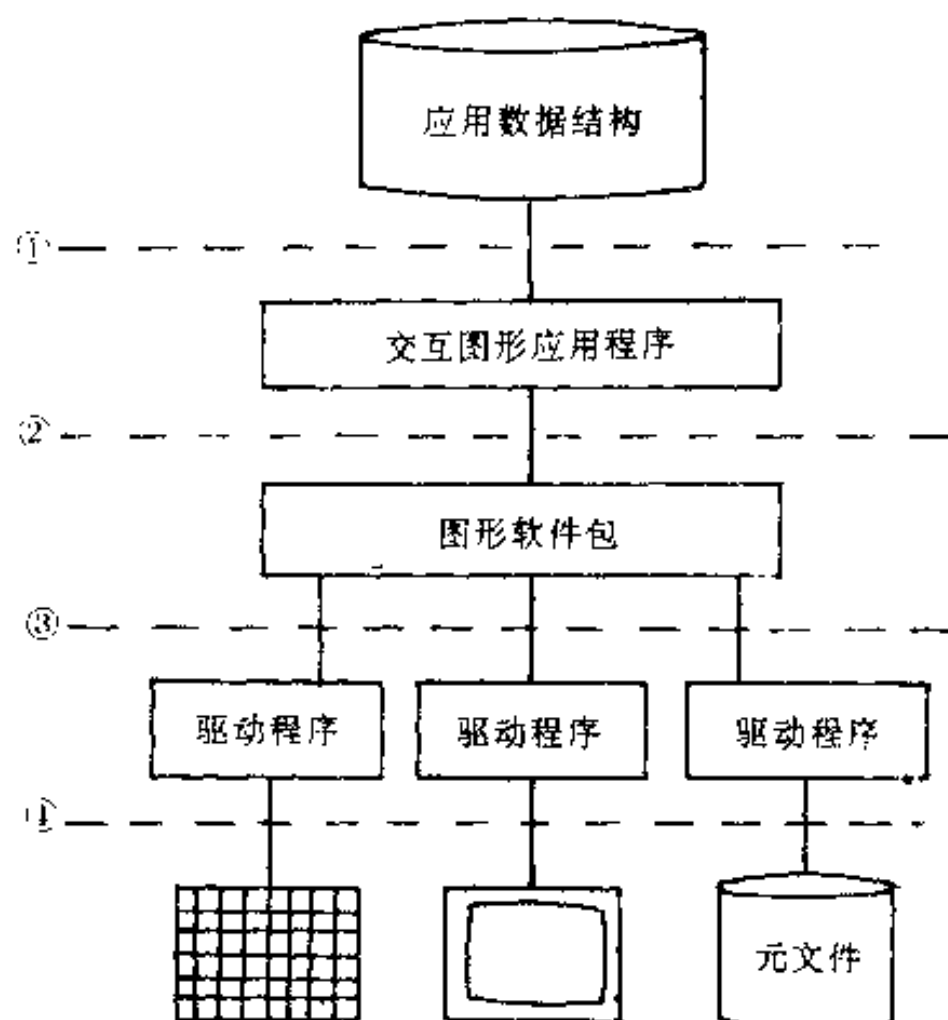


图 1-1 图形界面关系

国的图形核心系统 GKS (Graphical Kernel System) 和美国 ANSI X3H31 开发的程序员级分层交互图形系统 PHIGS (Programmer's Hierarchical Interactive Graphics System) 先后采纳为国际标准。

(3) 图形软件包与图形硬件之间的接口。

这个接口标准化的目的是提供一种与硬设备物理参数无关的图形硬件的控制接口。它是一种低层程序接口,称为虚拟设备接口 VDI (Virtual Device Interface), 现改称为 CGI (Computer Graphics Interface)

(4) 数据文件接口。

数据文件中记录图形信息的标准化允许程序与程序之间,或者系统与系统之间交换图形数据,实现图形数据的通信。这个接口称为虚拟设备元文件 VDM (Virtual Device Metafile), 是一种低层数

据接口标准,现由国际标准化组织改称为计算机图形元文件 CGM (Computer Graphics Metafile)。

以上四种标准接口在交互图形系统中的位置及层次关系如图 1-2 所示。

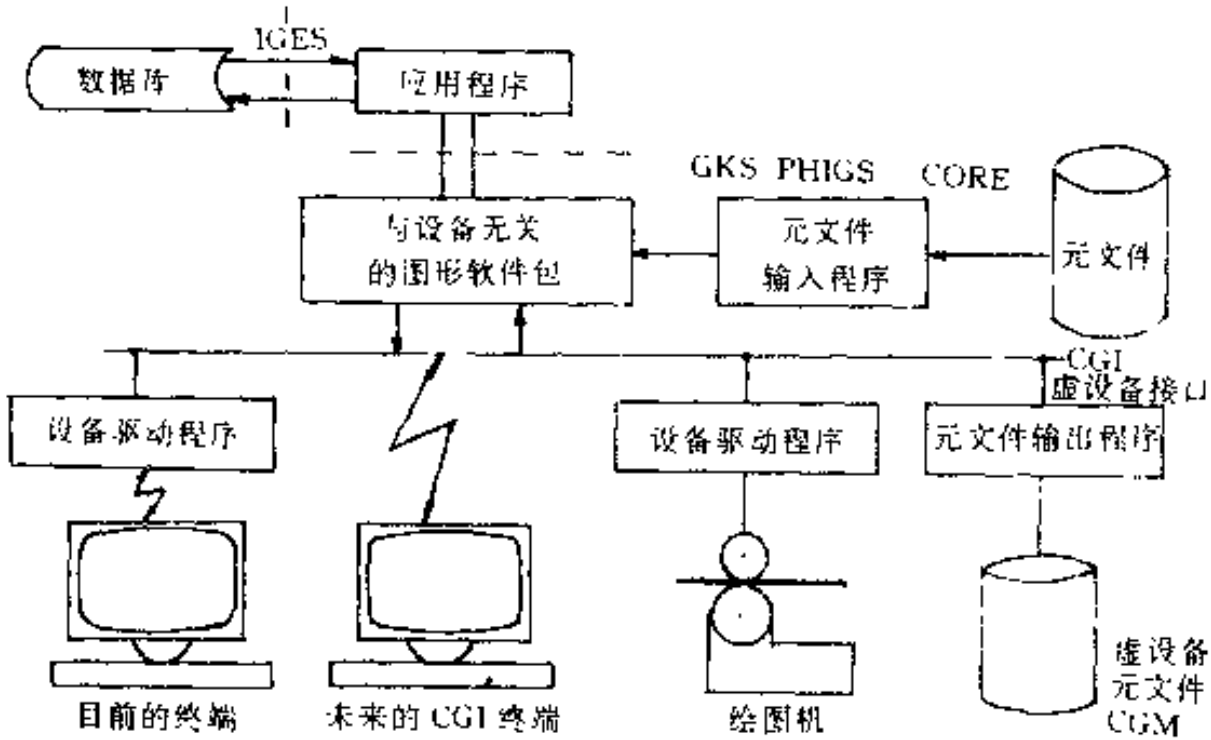


图 1-2 标准接口示意图

第二章 图形显示装置

图形显示装置涉及非常广阔的知识领域,由于消费类电器,特别是电视(再进一步是高清晰度电视)及计算机需求的强力推动,图形显示装置发展相当迅速,而且势头越来越猛。

前些年人们对计算机监视器的认识相当局限,实际上也没有什么可以选择的。当时人们大都和数字、字符打交道,有比较清晰悦目的监视器就满足了。而今,各种各样的监视器层出不穷,它们的性能、要求各不相同。如果说,以前监视器和计算机只需要简单的信号电缆连接就能投入工作的话,现在选择和连接监视器就不那么简单了,对做图形图象工作的用户更是这样。

应该说,图形图象显示装置的发展速度比不上计算机本身的发展和半导体集成电路的发展速度。不过就目前发展趋势来看,显示装置必将为计算机带来全新的面孔,其意义是不可限量的。

图形显示技术种类繁多,主要有阴极射线管显示、液晶显示、等离子体显示、场致发光显示、还有真空荧光显示 VFD(Vacuum fluorescet display)、电泳显示(electrophoretic display)和发光二极管 LED 显示等等。本章主要讨论阴极射线管、液晶、等离子及场致发光等四种显示装置。

2.1 阴极射线管装置

2.1.1 CRT 显示原理与发展

阴极射线管简称 CRT,是一种人们熟知的显示装置,人们从黑

白电视机、彩色电视机和示波器中早就认识了它。它是一种电真空器件,由阴极发射的电子束定向轰击涂在管屏玻璃内面上的荧光物质发光来显示图象。不同荧光物质的余辉不同,因此需要根据要求选择合适的余辉来保证图象质量。

CRT 电子枪发射的带负电的电子束由高压正电加速,经聚焦后由偏转线圈控制水平、垂直扫描,而控制栅上的视频信号控制通过它的电子束强弱、通断,最后在屏上产生图象。CRT 的管颈长,使用它装成的设备的机箱就相当

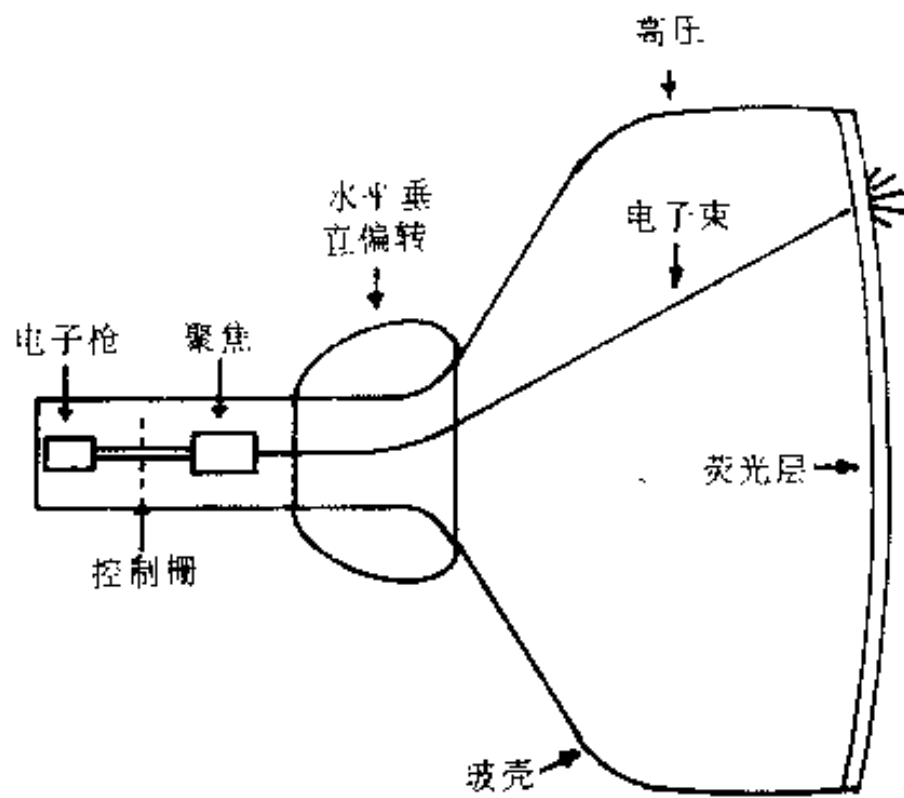


图 2-1 单色 CRT 示意图

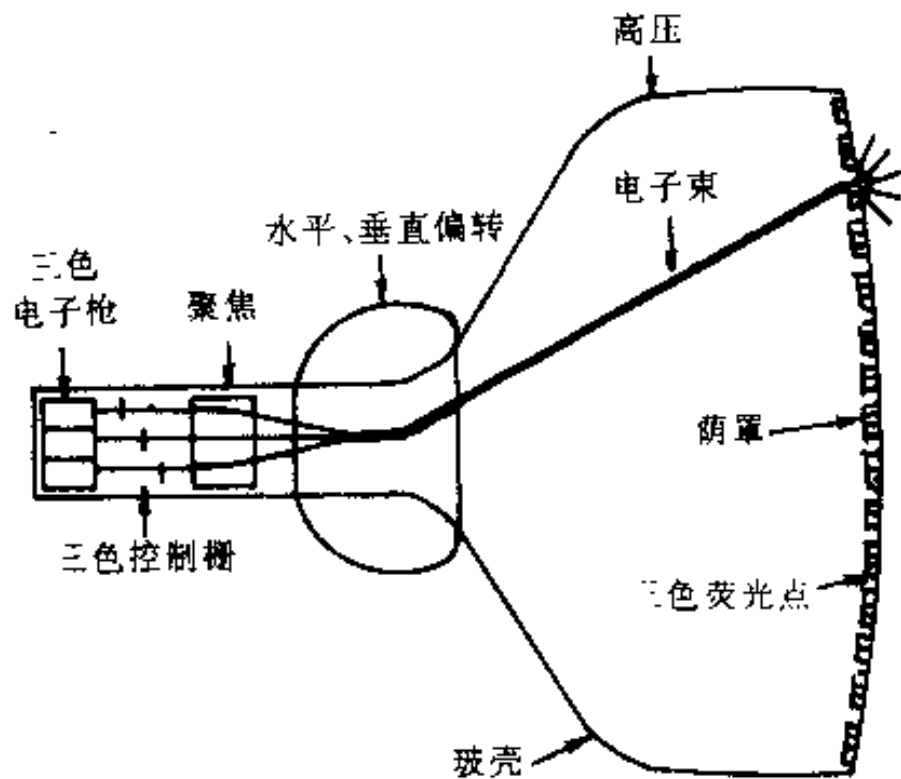


图 2-2 彩色 CRT 示意图

厚。要缩短管颈,需要解决大角度偏转和聚焦的问题。

单色 CRT 的结构如图 2-1 所示。

彩色 CRT 则要复杂得多,通常有 RGB 三根电子枪,管屏内面密布细微的红、绿、蓝三色荧光小点,并有一个荫罩。见图 2-2。

荫罩是由金属制成,上面布满小孔,它们通常是圆形孔,与荧光三色圆点精确对应,避免电子束轰击其它三色点。荧光三色点间底色是黑色,以提高清晰度。见图 2-3。

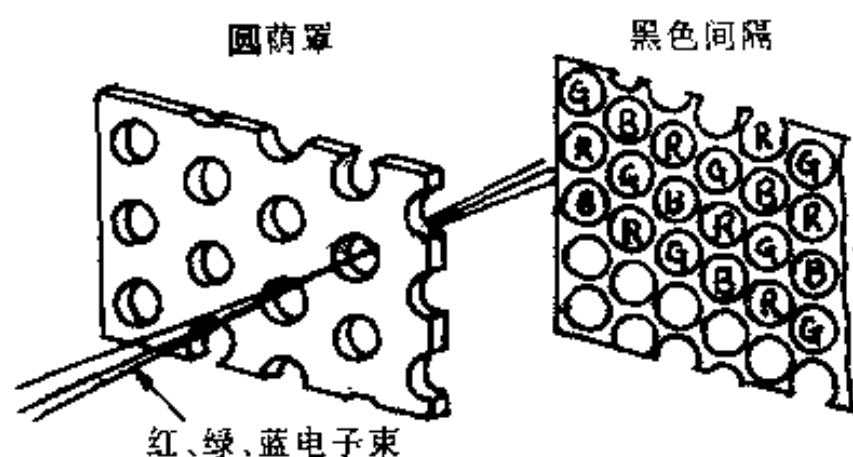


图 2-3 圆孔荫罩及三色荧光点

有些 CRT 的荫罩是用长方孔对应红绿蓝三色荧光小条,控制电子束轰击相应荧光条来形成全彩色。条间底色也是黑色,其作用也是提高清晰度。见图 2-4。

近年来,在 CRT 监视器方面有重大变化,由 NEC 的 Multi-sync 系列为先导,多频扫描 CRT 监视器很快普及开来。早先计算机监视器

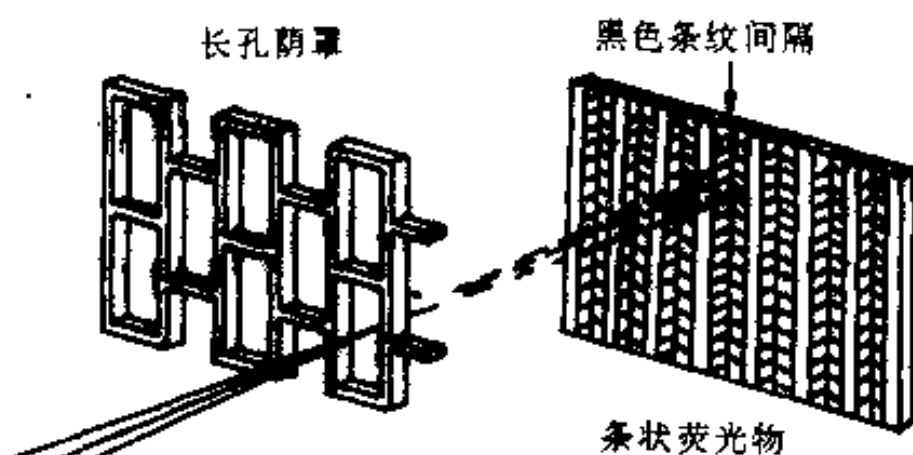


图 2-4 长孔荫罩及三色荧光条

制造厂采用和电视相同的水平及垂直扫描频率,美国标准为:水平 15.750Hz,垂直 60Hz。为了改进亮度(在同一时间间隔中更频繁地轰击荧光点),允许更快地对屏幕刷新,减少闪烁,许多监视器及其

控制器现在可以在一个很宽的频率范围内工作。

这种灵活的扫描能力使监视器可以适应不同的图形扫描标准。一般来说,扫描频率增加,图象质量变好。不过设计在专一频率上工作的系统能比适应广阔频率范围工作的系统有更好的图象质量。

无论对单色还是彩色,需求不断趋向高分辨率大屏幕。在CAD、图象处理、图形系统等方面要求特别突出。就是文字处理,由于各种窗口、菜单等等不仅对高分辨率大屏幕有要求,对色彩也有要求。对单色的CRT当然更要求高分辨率。

要实现高分辨率,遇到的两个老问题是:亮度不足和过热。在单色监视器中,发热是一个顽固的但不是不可克服的问题,它没有荫罩吸收、浪费能量。美国的Mega Scan Technology公司拥有偏转线圈热量消散技术的专利,并用于48cm(19英寸)单色监视器中。它提供 4096×3300 象素分辨率,电子束斑尺寸为0.0127cm.(0.005英寸)。这种极高的分辨率特别适用于医学图象和桌上印刷系统中的字体检查。

显象管形状也影响显示质量。如果管屏为球形,电子束按其半径扫描,一旦聚焦正确,只要符合球面尺寸,则不论在何处都和管屏曲面垂直,聚焦质量相同。由于制造原因和偏转方法的制约,保持电子束圆斑聚焦质量是不容易的,更不要说现在的趋向是屏面越来越平即平面直角!

当聚焦成圆形的电子束在“平面型”管中偏离中心时,半径增长,聚焦改变,它以一定角度轰击管屏,形成一个椭圆。这种畸变的电子束降低了图象质量。不同厂家用不同的方法来解决该问题,通常采用某种电子束成形(beam-shaping)电路,根据在管屏上的位置来调整扫描中的电子束形状。

为了得到更高的彩色显象管分辨率,荫罩孔及三色荧光点需做得更小。此时荧光物少了,亮度随之减弱,加上荫罩孔小了,若电子束强度不变,只有较少的电子束轰击在荧光点上,亮度会进一步

减弱。

如果在给定时间内增加电子束轰击荧光点次数或增强电子束强度,都会使荧光发光强度增加;但不管是哪种方法都会增加轰击荫罩的机会,荫罩会因发热而变形,影响荫罩和荧光三色点的准确对应关系,降低图象质量。一家 CRT 厂说,只有 $1/3$ 电子束能量真正到达荧光点,也就是说,有 $2/3$ 的电子束隔在荫罩一边,其能量转换成了不希望的热量。

要稳定荫罩,使其不变形,方法有多种。其中之一是把普通铁荫罩换成热稳定性更好的一种称为 Invar 的铁镍合金;另一种是 Zenith 公司采用的,它把荫罩处于张力之下,防止发热时偏移。

据 Zenith 公司发言人称,平张力荫罩 FTM(Flat-Tension Mask)是一种紧绷着的钢质荫罩,它比普通荫罩薄,这种荫罩中被吸收的电子束较少,张力还能防止发热时变形和错位。这样改善了亮度和对比度(分别为 50% , 70%),也改善了色纯度。这种 CRT 荧光三色点间距离(即点距)是一个平均值 0.28mm ,管中央为 0.27mm ,边缘为 0.30mm 。FTM 很有效,但也昂贵。

Mitsubishi(三菱公司)展示的 48cm (19 英寸) 2000×2000 彩色监视器原型点距为 0.21mm ,水平扫描频率为 128kHz 。Toshiba(东芝公司)加了 Invar 荫罩,改进了电子枪、聚焦部件和偏转系统,亮度增加了 50% 。

目标还在不断延伸,近年来 1280×1024 的荫罩已经很普遍。此外,多频扫描技术使 CRT 更具有竞争力,因为在改变分辨率时,用不着更换监视器。而固定格式的等离子体显示或 LCD 显示则不行,当然,对较低分辨率且是对最高分辨率进行偶除的例外,比如说对 640×400 象素的,可以实现 640×200 或 320×200 象素显示。

2.1.2 CRT 术语解释

- 点距(tridot pitch):RGB 三色荧光点组间距离。点距愈小,可表达的文本和图形愈细腻。

- 水平扫描频率(horizontal scan frequency):电子束横扫屏幕的频率,也称行频。对标准图形控制器来说,CGA 15.6kHz,EGA 21.8kHz,PGC(Professional Graphics Controller)30.4 kHz,VGA 31.5kHz,Macintosh II 35kHz。

- 垂直扫描频率(vertical scan frequency):电子束竖向扫描的频率,也称帧频。监视器的帧频大都在45Hz~75Hz,甚至更高,可以适应各种标准的图形卡。过低的帧频会使显示闪烁。

- 带宽(bandwidth):电子束通断的频率上限,带宽值越大,每行中可含的独立像素点越多,分辨率越高。

VGA需要的带宽约为34MHz,NEC Multisync XL为65MHz。带宽值大可以适应非IBM标准的更高分辨率的显示控制器。比如,Multisync XL配上NEC MVA-1024控制卡可提供 1024×768 像素分辨率。

在图形控制器中,带宽也称点频(dot rate)或像素频率(pixel rate),它由称为点时钟(dot clock)的晶体振荡器产生。

- 分辨率(resolution):分水平分辨率和垂直分辨率。水平分辨率指每行中独立可分的像素数最大值,垂直分辨率指每屏显示的像素行数最大值。分辨率一般以 640×480 的形式表示,前者为水平分辨率,后者为垂直分辨率。分辨率和带宽、水平扫描频率及垂直扫描频率有关。当然密布在CRT屏内面上的荧光点是分辨率的基础。

在扫描行频和帧频已定、形成扫描光栅的情况下,影响分辨率的是带宽。如果电子束通断的频率上限较低。则每行中可分的独立像素点就较少,水平分辨率就下降。

如果帧频较低,则每帧可容纳较多的扫描行,垂直分辨率似乎可以提高,但此时显示产生闪烁,所以不可能使用过低的帧频。

- 表面处理(face treatment):减弱反射光,改善对比度的方法。为了减弱室内光的影响,普遍采用两个方法。一是对CRT屏幕表面进行蚀刻,使其表面对环境光形成漫射,减弱反射光。二是对

玻璃上色来改善对比度。蚀刻的屏幕易于辨认,反射边缘柔和,但它会对象素产生一些畸变。有些监视器屏幕表面是抛光的,象素畸变小了但反光强。

- 数字信号输入(digital signal input):图形卡送来的是离散的数字信号,可以再现有限彩色。比如 CGA 数字信号为四种:RGBI 分别代表红、绿、蓝和光强。这些信号的取值只有两种(有或无,强或弱),因此只能产生 16 种颜色。EGA 的红、绿、蓝都分别增加一种,用 R,r,G,g,B,b 表示,可以产生 64 色。

- 模拟信号输入(analog signal input):图形卡送来的是连续变化的模拟 RGB 信号,理论上可以再现无数色彩,实际上色彩数受图形卡限制。比如 VGA 每种基色存 6 位,可产生 2^{18} 即 256 000 多种彩色,Macintosh II 每色存 8 位则超过 16M 种彩色。在图形卡上有 D/A 转换器,为 CRT 提供模拟信号。

- 兼容性(compatibility):CRT 监视器是和图形控制卡配套的,是否适应同类规格的其他控制卡而正常工作,这就是兼容性。举例说,用一台 IBM PS/2 80 型计算机和一台 IBM 8513 31cm(12 英寸)模拟式彩色监视器连结,西格玛公司(一家兼容厂)的 Sigma VGA 诊断程序在 80 型机的内置 VGA 线路上运行得很好(当然诊断 Sigma VGA 自身对 VGA 标准扩充了的性能除外)。但 Sigma VGA 图形卡接上 8513 监视器却不能工作,更不用说测试该图形卡对 8513 的效果。

- 电磁干扰(electromagnetic interference):CRT 监视器产生电磁辐射,引起互相干扰,简称为 EMI。以 PS/2 系统为例,被测的监视器离 PS/2 机 55.88cm(2 英尺)、离 PS/2 监视器 30.48cm(1 英尺)。试验证明,许多种监视器都受附近的 PS/2 监视器的电磁干扰而产生不同程度的不稳定,Electrohome 的 ECM1310 是个例外。当 PS/2 监视器关断后,干扰问题消除了。当然还有一些监视器不受 PS/2 监视器的干扰影响,但种类不算多。如果计划使用两台监视器,或两台微型计算机靠近工作,必须考虑选择适当的监视器,它

应该具有抗电磁干扰的良好屏蔽性能。

• 隔行(interlaced)及逐行(non-interlaced)扫描:隔行扫描的每帧分成两场,一场由奇数扫描线组成,另一场由偶数扫描线组成,两场构成完整的全扫描线帧,见图 2-5。

而逐行扫描是一次完成一幅全扫描线帧。隔行扫描对带宽及扫描频率的要求比同样分辨率的逐行扫描要低。隔行扫描在垂直扫描频率和逐行相同时,水平扫描频率和带宽都只需逐行的一半。

在实验室条件下,还可以测试 CRT 的一些低层的性能,比如电子束光斑大小(spot size)及其随亮度增加发生扩散变化的情况,在屏幕中心和边缘处的差异;还有电子束位置随时间的不确定性所产生的闪烁和漂移,低频的漂移大了,就难以完成细致的图形或 CAD 工作,较高频的晃动易使眼睛疲劳。错聚(misconvergence)表示电子束和荧光点对准的误差。错聚严重时,白色文本看起来象有彩色的镶边。屏幕中心错聚情况最小,左上角最差。屏幕上的图形情况还和监视器内部电源的稳压性能有关。屏幕黑背景上的一条直线的位置在背景变白时不应移动,而实际上这受设备负载能力所影响。

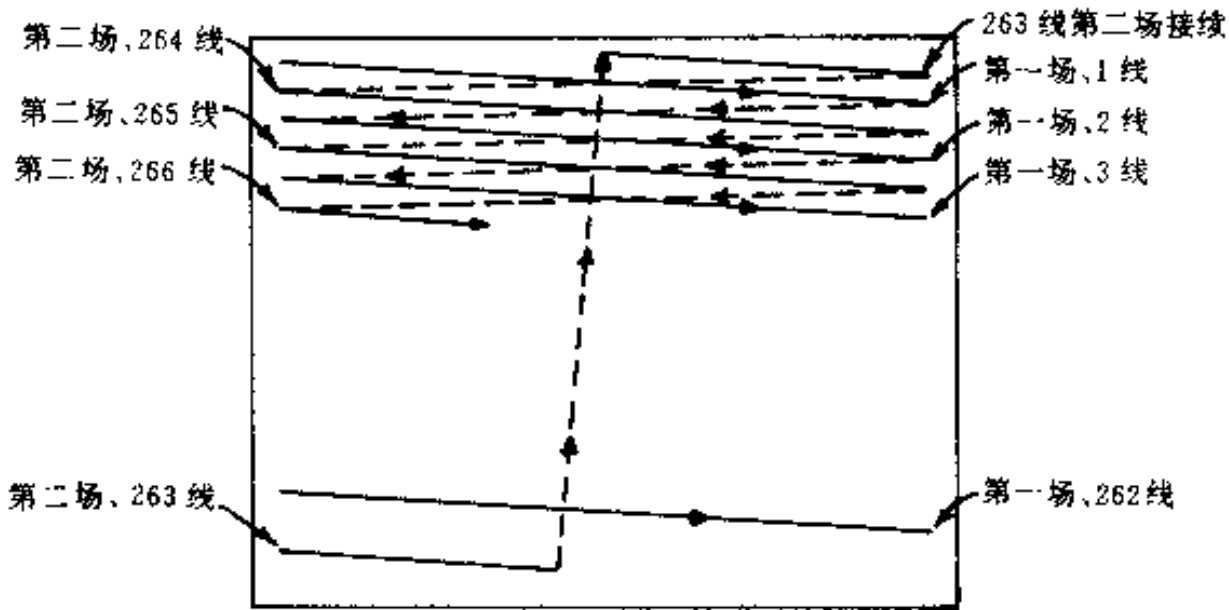


图 2-5 隔行扫描

2.1.3 投影式 CRT 大屏幕显示

在需要大面积图形图象显示方面,投影式 CRT 技术依然是一种相当有效的手段。

投影监视器的基础是采用三支不同颜色的单色 CRT,即作为三基色的红、绿、蓝显象管,通过同步机构来完成彩色图象的显示。所有三支 CRT 通过光学系统投射到同一屏幕上。投射方式可分两种:一种是前投式(front-throw),这时 CRT 发出的光向前投到球面屏幕上,经屏幕反射后为人眼所接收;另一种是背投式(rear-throw),这时 CRT 发出的光投射到屏幕背面,透过屏幕为人们所见。背投式 CRT 和屏幕制成一体,而前投式 CRT 和屏幕是分离的。

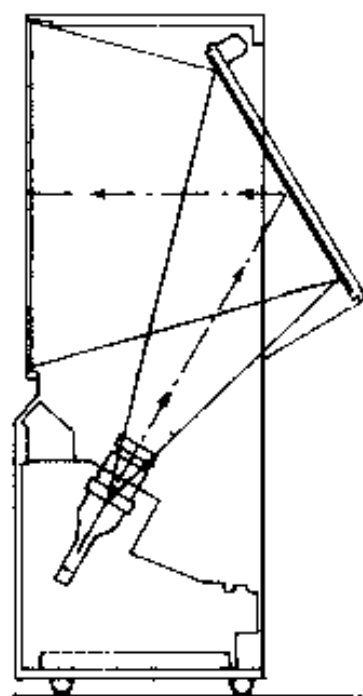


图 2-6 背投式投影机

背投式装置内部结构示意图如图 2-6 所示。

投影装置的核心是高亮度、高清晰度、冷却良好的投影管,配上精密的光学系统和功能完善的控制系统及丰富的接口,可以适应各种应用场合。

以比利时巴可(BARCO)公司的 BARCOGRAPHICS 800IRIS 为例,它是一种智能化重合接口系统(Intelligent Registration Interface System)。可以自动进行快速和精确的会聚;自动锁相电路范围广;行频在 15~90kHz,帧频在 45~120Hz;825lm 的高亮度输出,屏幕宽度可在 1.2~6m 范围内选择,改进型的 20.32cm(8 英寸)液体冷却投影管,带有恒压仓,能产生高质量的图象;光学清晰度为

10lp/mm, F1.06 的混合镜头, 具有提高对比度的完善彩色校正: 在非标准投影角度时, Scheimflug 校正可进行光学聚焦无级调整; 即使对初次使用和无技术经验的用户, 也可以按组织良好的屏幕菜单用红外遥控器进行方便的调整, 可编程口令可以保护投影机的设置以免遭他人弄乱; 通过 RS232/422 接口, 巴可的“Control 800”软件可在 IBM PS/2, Apple MAC 或 UNIX 机上运行, 进行所有的调整工作。

为了和各种计算机顺利连结, 接口多种多样, 有 RGB 模拟 (ANA) 接口、数字 (TTL) 接口; CGA/EGA 兼容、PS/2 接口、VGA 兼容、MAC 接口等。

巴可公司的背投式投影系统可以在几乎任意的环境照明条件下在很宽的视角内清晰可见。它的 RETRO 800 投影机可以象搭积木似的很容易地组成显示墙屏。方法是把两台投影机迭起来放, 上面的一台底朝上, 这样可以形成长墙屏, 比如在 2×2 这样的结构时, 尺寸为 $2.8 \times 2\text{m}$ 。用于过程控制和监视方面是很理想的。

以 RETROGRAPHICS 800 为例, 它的主要技术规格如下:

水平扫描频率 15~90kHz, 垂直扫描频率 37~140Hz, 两者都是自动锁相的。

最小水平回扫时间小于 $2.6\mu\text{s}$, 最小垂直回扫时间小于 $200\mu\text{s}$ 。

RGB 带宽大于 60MHz。

亮度在 10% 白峰值时为 825lm。

高亮度、高清晰度、液体冷却 20.32cm CRT (荧光面积为 16cm (6.3 英寸), 指对角线尺寸)。

高清晰度、全彩色校正、液体耦合、F1.1 混合透镜。

光学分辨率在全屏为 10lp/mm。

投射屏对角线尺寸 1.7m, 在 50% 亮度时, 水平视角为 $\pm 45^\circ$, 垂直视角为 $\pm 12^\circ$ 。

2.1.4 平板式 CRT 显示

多少年来, 人们一直在为实现平板式 CRT 而努力, 一方面是

因为常规 CRT 的厚度太大,和屏幕尺寸相比约为 1:1,另一方面是因为想实现常规 CRT 所不能完成的性能。

这个过程中困难很多,但研究始终没有停止。当许多研究人员把注意力集中在固态器件时,不少公司研制平板式 CRT。平板式 CRT 比其他平板式显示的潜在优势是

- 已有的高效彩色材料的技术
- 表现力强
- 高发光效率荧光材料
- 高亮度
- 良好的多路传送性能
- 容易获得灰度
- 简化的电子线路
- 灵活的格式和尺寸
- 可供利用的已有熟知的技术
- 快速响应能力

开发平板式 CRT 集中在常规 CRT 比较薄弱的地方,即超小型(小于 13cm 即 5 英寸)和超大型(大于 77cm 即 30 英寸),这里的尺寸都是指屏幕对角线尺寸。

虽然许多人认为液晶显示是小尺寸单色显示的最佳方案,因为它们特别省电。但要获得价廉、自然多色彩的液晶显示是件不容易的事。此外,液晶显示尚有开关速度和视角范围有限等方面的问题,这些都限制了它的应用性。

在大尺寸的屏幕显示方面,目前大都由投影 CRT 显示充当重任。投影 CRT 显示的弱点是,费用昂贵、笨重的封装要求及视角范围有限。

平板式 CRT 的研制基础针对阴极、电子束定位和调制、由于阴极不能提供足够电子束流以产生适当亮度的亮度增强技术、荧光物质和真空封装。

在这里，平板式 CRT 定义限定在利用电子束轰击荧光物产生光的一种设备。因此产生电子束的阴极是很重要的。平板式 CRT

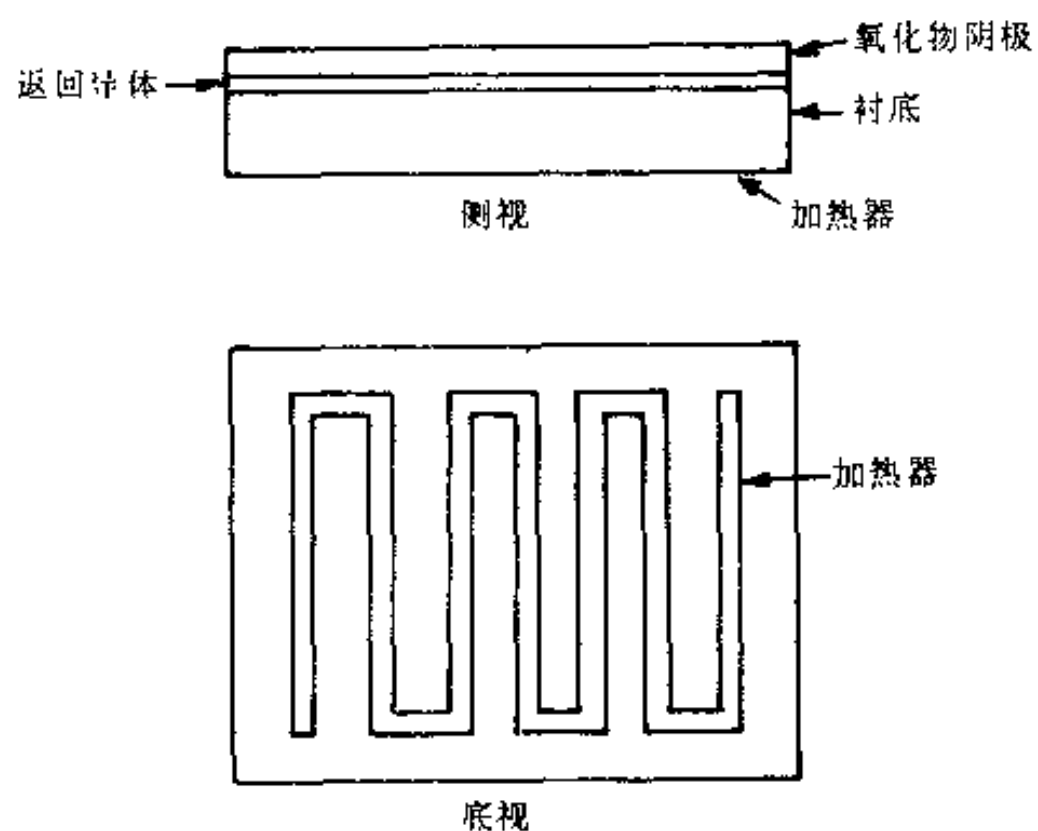


图 2-7 大面积低温阴极

阴极可分单点 (single-point)、大面积热氧化物涂复 (large-area thermionic-oxide-coated)、大面积非热 (large-area nonthermionic) 阴极等。单点阴极和常规 CRT 类似，只是电子束发射方向和荧光屏平行，而最后偏转 90°(实际上偏转角要小得多)轰击荧光屏。大面积的阴极有多种，比如大面积低温阴极，见图 2-7。

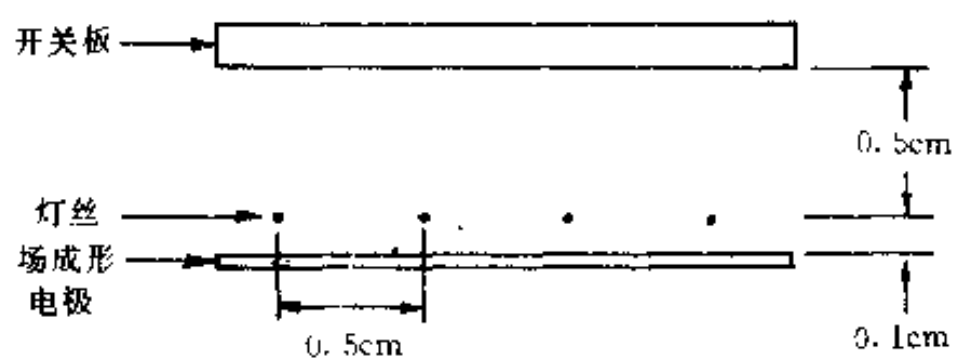


图 2-8 多灯丝阴极

多灯丝阴极，见图 2-8

场发射阴极 (spinal-type field emission cathode)，见图 2-9。等等。

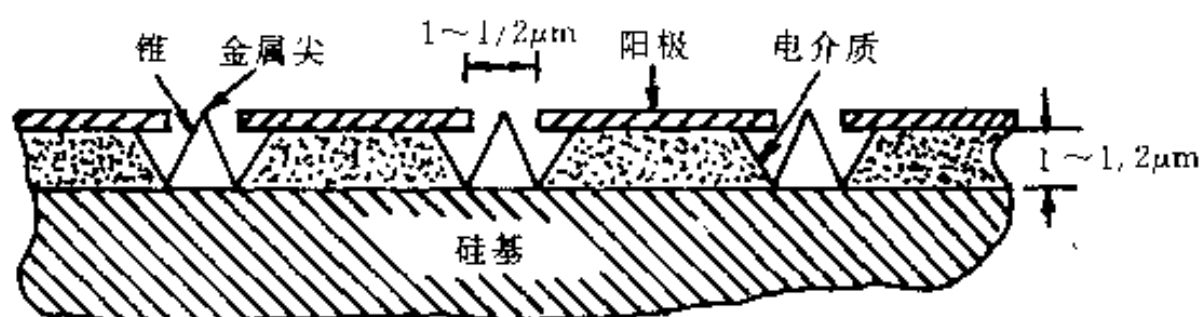


图 2-9 场发射阴极

单色阴极和大面积阴极的 CRT 结构可从图 2-10 和图 2-11 中看出一个大概情况。当然这是许多种平板 CRT 中的两个特例,不能代表所有的情况。

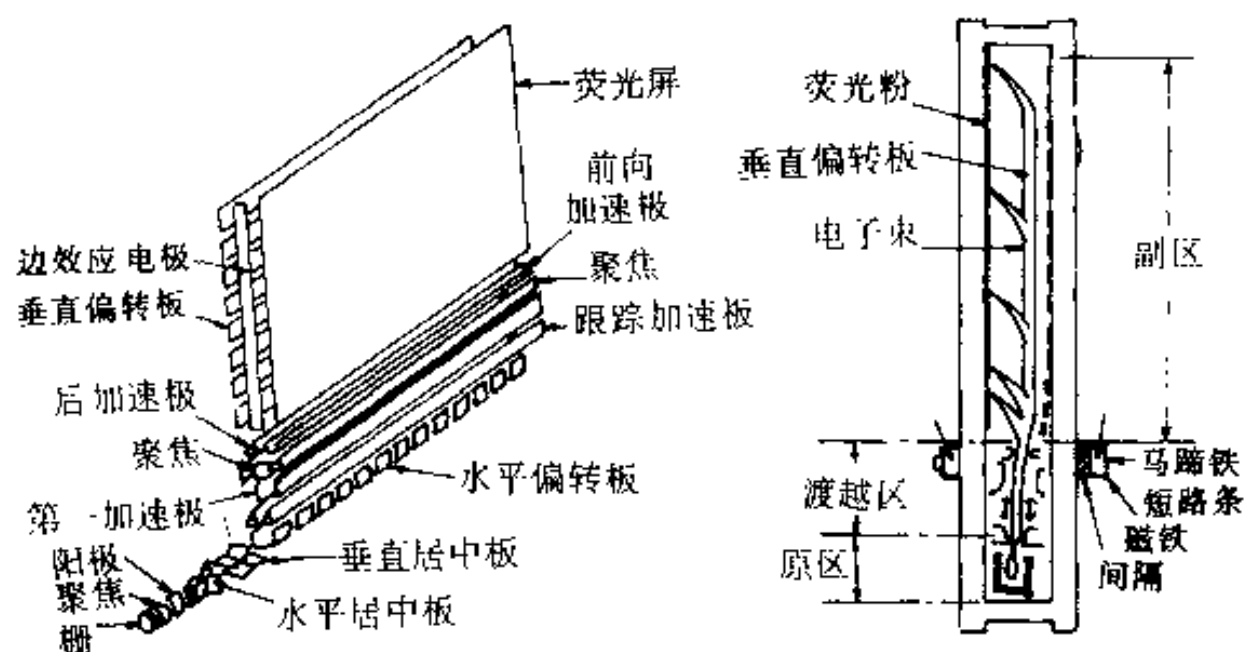


图 2-10 Aiken 管结构

实际上,在研制平板型 CRT 的同时,常规 CRT 发展更快,所以目前平板型 CRT 在市场上远不能和常规 CRT 匹敌。但是平板 CRT 研制中的经验及教训对进一步的研究是有益的,特别是场发射显示 FED(Field-Emission Display)可能是最有意义的。

场发射显示具有 CRT 基本规律的优点。它和目前常规 CRT 最

主要的差别是没有庞大的高压“热阴极”电子枪,而采用微米级大小圆锥状,称为“冷阴极”或“场发射阴极”的结构。该结构中电子受激电压比传统的 CRT 电子枪低得多,产生的热量也低得多,于是常称冷阴极。冷阴极 FED 的典型工作电压约为 1000V,而常规 CRT 用 25 000V~30 000V。

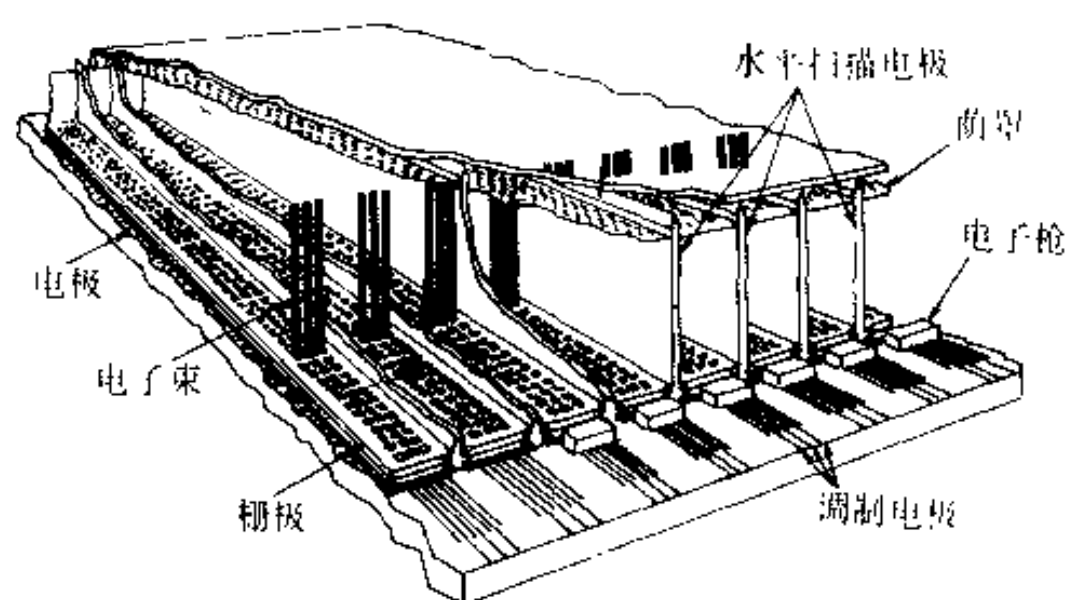


图 2-11 RCA 平板显示

冷阴极很小,所以产品很薄,扁平的 CRT 采用的场发射阴极按 X,Y 阵定址排列。斯坦福研究所(Stanford Research Institute)在 FED 研究开发方面做了大量的工作,产品开发的领先者是美国加州的科罗雷(Coloray in Fremont California)。

Coloray 目前开发的 FED 由两片玻璃组成,中间隔以真空(压力为 10^{-6} torr, torr 为 $1/760$ 个大气压)。面板含荧光屏,象常规 CRT 一样,背板(基板)含冷阴极阵场发射极及行列驱动芯片,见图 2-12。

图 2-13 所示截面图上,阴极发射极做在基板行、列电极上。

每个行列电极交叉处有 10~100 个发射极尖,以保证冗余度、发射极一致性,于是也就保证了高的产品成品率。场发射极尖通过

基板（直径 1 微米的孔发射电子，电子束轰击面板上荧光点的方式和常规 CRT 相同。

显示分辨率由面板上的荧光点设计控制（在基板上阴极的密度超过了 CRT 的最高分辨率要求） 根据 Coloray 提供的资料，典型 PC 显示器的基板和面板厚 1.1mm，中间隔开 0.1mm 的空间。

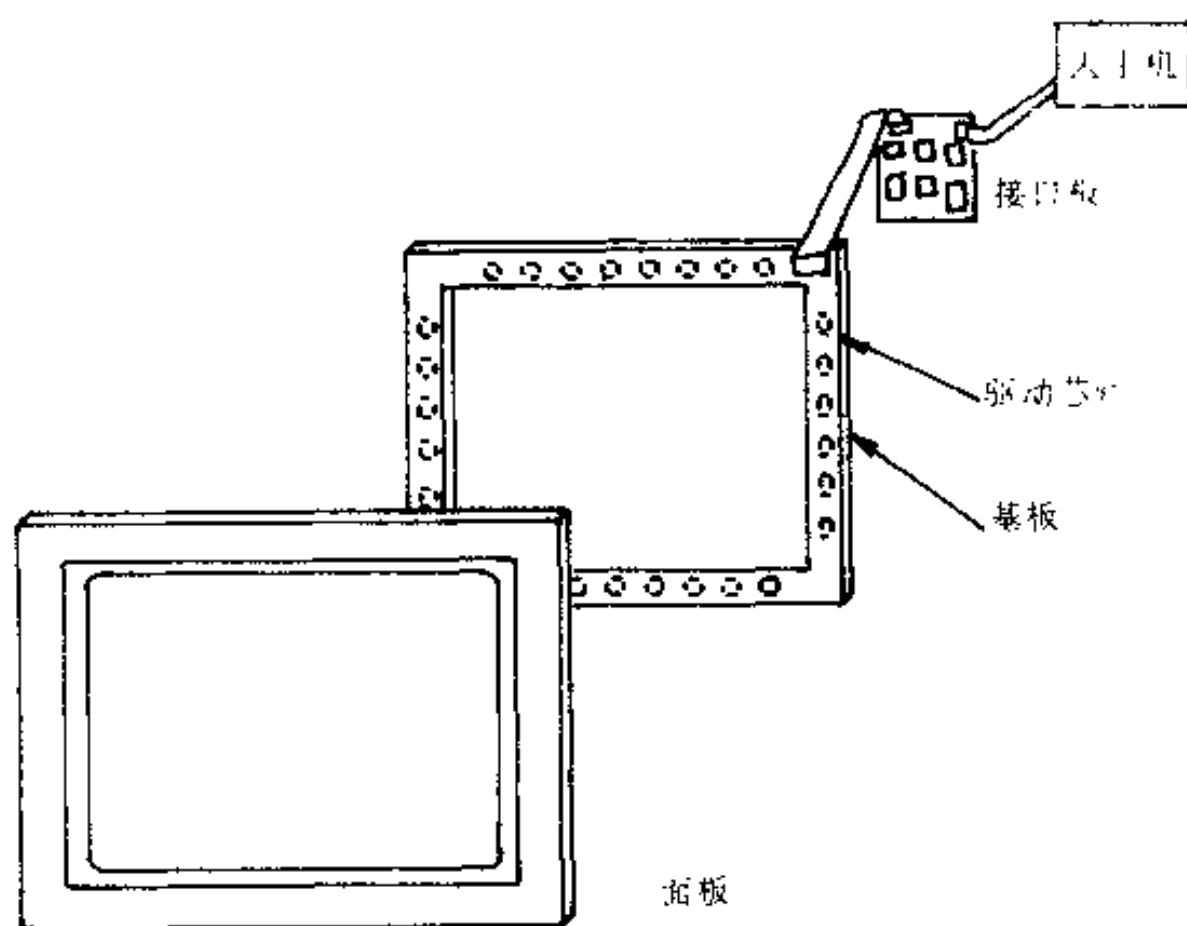


图 2-12 场发射显示结构示意图

FED 技术的支持者认为它是 CRT 和平板显示领域中最出色的技术。FED 的亮度为目前背光照明 LCD 的三倍，成品率和可靠性都高。Coloray 进行的项目为 26cm(10 英寸)FED 监视器，可以显示 16M 种色彩，只需耗电 2.4W。

FED 是一种非常优雅，令人瞩目的技术，不过在近期内推出商品仍非易事。法国的一家研究实验室 LETI 演示了 16cm(6 英寸)FED 监视器原型，已做的时间测试超过了 3000h；FED 的基本技术性能已都演示了。下一步不是技术问题而是商业计划问题。

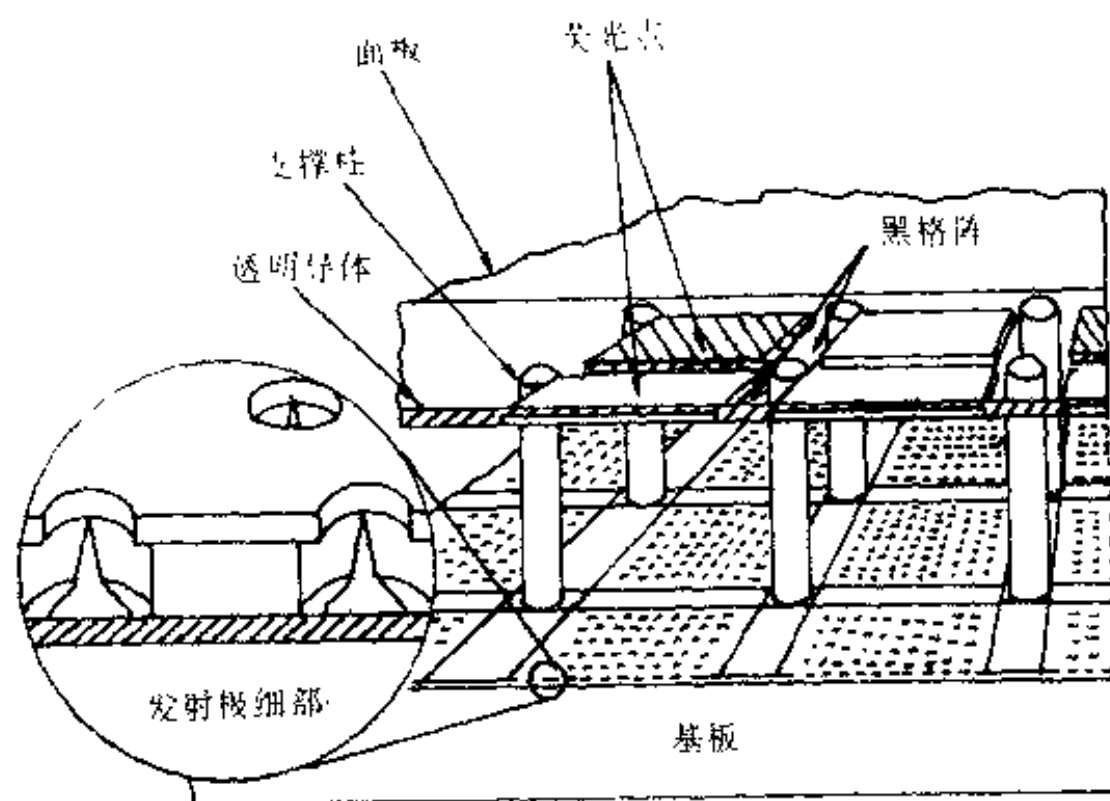


图 2-13 场发射显示横截面图

2.2 液晶显示装置

2.2.1 LCD 原理及发展

液晶显示 (Liquid-Crystal Display) 是一种非发光性的显示器件。就是说,它显示信息的原理不是象 CRT 那样靠器件本身发光来实现,而是依赖对环境光反射或是对外加光源(背投光)的透射加以控制来实现的。它的基础是液晶。

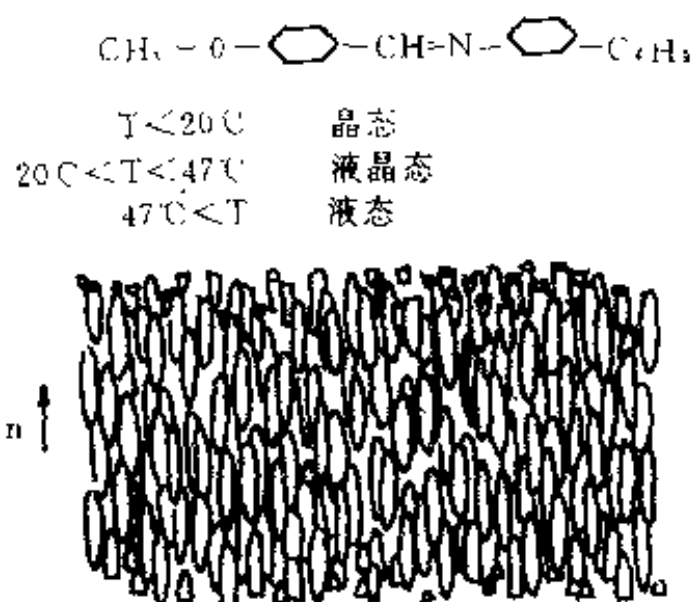


图 2-14 向列相液晶

液晶虽然是一种液态似的物质,但它和普通液体不同。液体没有大范围的次序,它是各向同性的;而液晶有大范围的次序,它是各向异性的。以 MBBA (Methoxy Benzyldiene Butyl-Analyne, 一种甲氧基苯叉丁基化合物) 为例,这种由长扁分子组成的液晶为向列相(nematic phase)。图 2-14

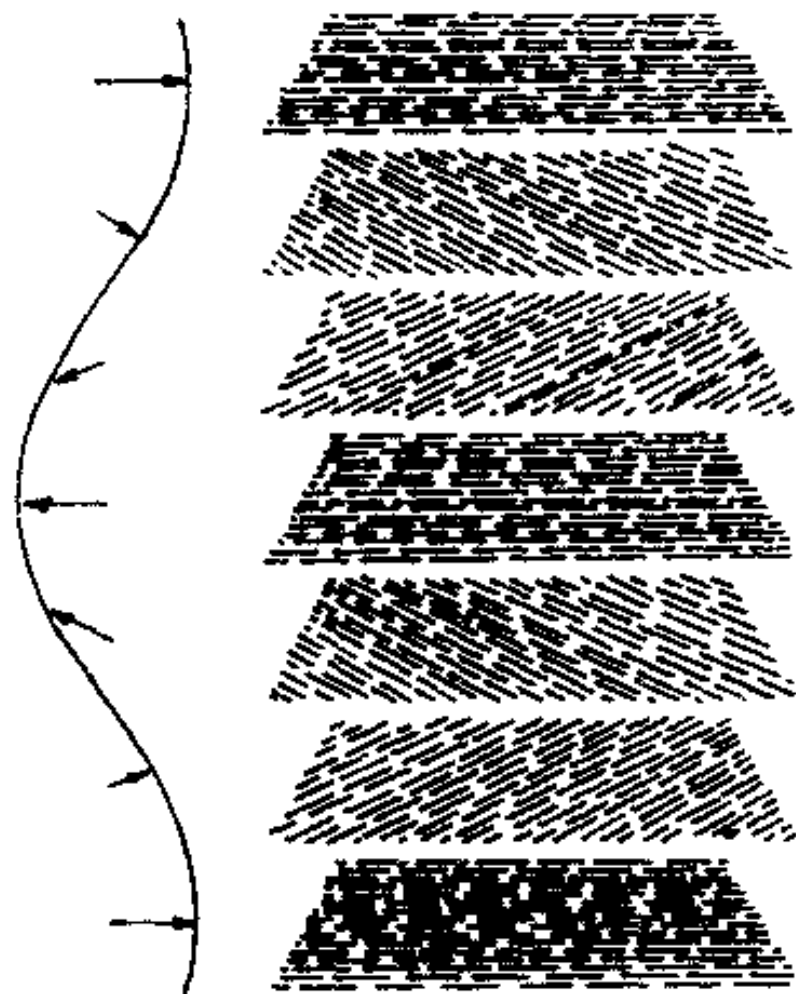


图 2-15 胆甾醇相液晶

中椭圆表示分子,明显可见分子排列次序。

除了向列相之外,还有胆甾醇相(cholesteric phase)和层列相(smectic phase)。图 2-15 是胆甾醇相的三维结构示意图,每层有向列指向,方向各不相同,从而构成螺旋梯级结构。

层列相如图 2-16 所示,液晶分子呈层状分布,层间可相对自由移动。层中指向可有几种形式,因此有几种不同的层列相。

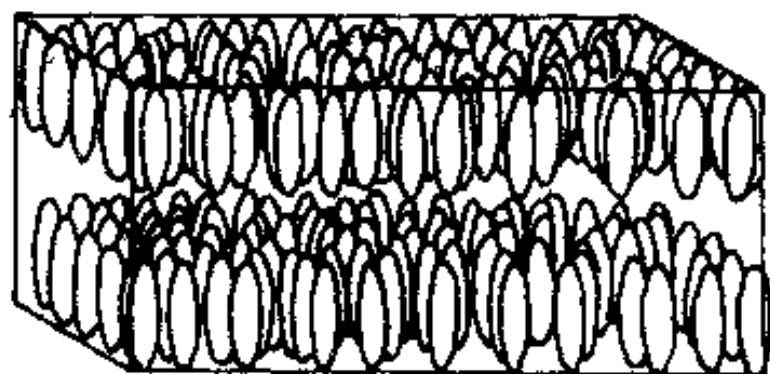


图 2-16 层列相液晶

把液晶用于显示是依靠了液晶的几个重要特性。液晶有三种变形结构：斜向(splay)、扭向(twist)和曲向(bend)。而液晶材料总是反抗这些畸变，其扭矩和形变的关系是线性的，它的比例常数相应为三个弹性常数。这些常数都相当小(约 10^{-12}N)，就是说，只要很小扭矩就能使向列相液晶变形。

其他两个重要特性是液晶的低频和高频电极化率，即电介质常数和折射率。特别是，电介质常数的各向异性导致一种扭矩，这种扭矩使加有电场的液晶重新定向，使最大的电介质常数是顺电场方向的。这种扭矩对抗液晶的弹性扭矩而产生变形。这时高频极化率起作用了，形变的双折射可由人眼观察到，也就是说有了某种显示。

LCD 之所以成功还在于液晶材料比结晶体有更大的双折射，典型值在 0.1~0.3 数量级。这就意味着液晶可产生较大的光调制，显示有良好的对比度。

图 2-17 简要表示了典型的扭向列(twisted nematic)液晶单色显示的原理。当一个 LCD 象素处于关状态的时候，偏振光从液晶材料均匀反射，没有对比度；当该象素受激时，偏振光透过它射在后偏振器上后被吸收，该象素就变暗，形成显示。

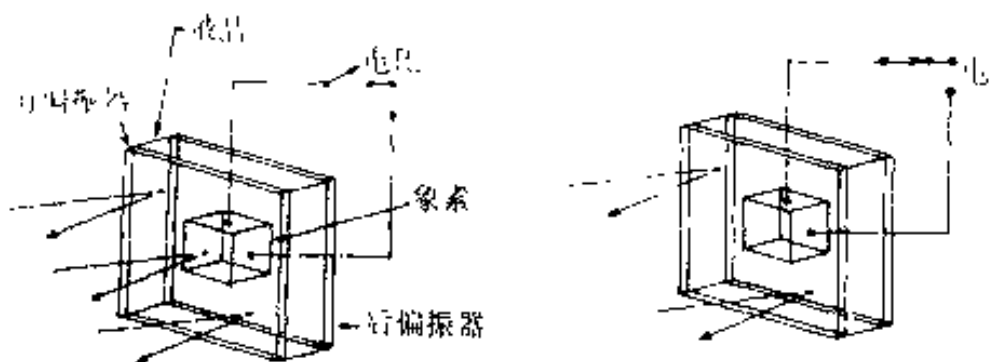


图 2-17 LCD 象素的开关状态

这种向列液晶单色显示已得到广泛使用。LCD 板由通过行列定址的扭晶象素阵组成，晶体在两块极板之间，不加电时，晶体改变了入射光的极性，把大部分光反射回去，因此 LCD 板呈现人们

熟悉的均匀灰色。

加上电压后,入射光通过受激的晶体象素照到后偏振板上被吸收,形成暗象。可惜的是,入射光被 LCD 板各层散射,降低了象的对比度。

早期低对比度及不易读识的问题大体由背投光(rear-lighting, backlit)解决。随着背投光式 LCD 的发展,以前反射式的 LCD 已衰落。

事实上,超扭(supertwist)LCD 需要背投光来发挥它已增强的对比度的优势,由于成为发光显示而更受人们欢迎。彩色 LCD 也需要背投光。不过,原先无源 LCD 耗电很省,而背投光光源耗电通常要比控制液晶本身的耗电大许多倍。与此有关的是把 LCD 板和计算机相连,放在投影机上去代替投影胶片,达到大屏幕显示的效果。

LCD 有许多优点,尺寸小、重量轻、省电、分辨率良好,由于平板技术线性好,象素可以直接定址,没有电子束偏转和畸变的问题。不过,要制造和 CRT 一样的显示尺寸和质量还是相当困难。

最有希望的 LCD 是近来开发的有源阵(active-matrix)或薄膜晶体管 TFT(Thin-Film Transistor)LCD。有源阵 LCD 与标准多路选择技术(即分时象素电子驱动电路)不同,它包括组成各象素的薄膜晶体管在内,这些晶体管作为开关控制各个象素。

TFT 法消除了因多路选择引起的时差(当显示行多时,对比度下降),并允许直接访问每一象素。至于彩色,则用有机滤光器(organic filter)和背投光来实现。

有源阵 LCD 已用于全彩色 9cm(3.5 英寸)袖珍电视机,1988 年 5 月在信息显示协会 SID(Society for Information Display)举行的会议上,几家厂商展示了 23cm(9 英寸)有源阵 LCD 屏。

生产 23cm 或更大屏面的 LCD 的主要障碍是成品率低(一般为 20%)。不过,LCD 已确实成为一个志在取代 CRT 的咄咄逼人的对手。

2.2.2 扭向列 LCD

1970 年开发成的扭向列 (twisted nematic) 液晶是 LCD 技术的重大成功。如其名称所述,扭向列液晶在向列材料中有一个 90° 的扭转,见图 2-18。

由两表面上均匀边界条件引导的扭转对每一表面都成 90° 交角。扭转是一致的,因此扭转的双折射材料结构对液晶来说是唯一的。边界是均匀的,有一微小倾斜来保证分子在门限值上都以相同方向偏扭。这种非均匀的双折射材料的电磁模式是相当复杂的,不过对特定的扭转距离和双折射范围来说,可见光以唯一方式通过该材料。

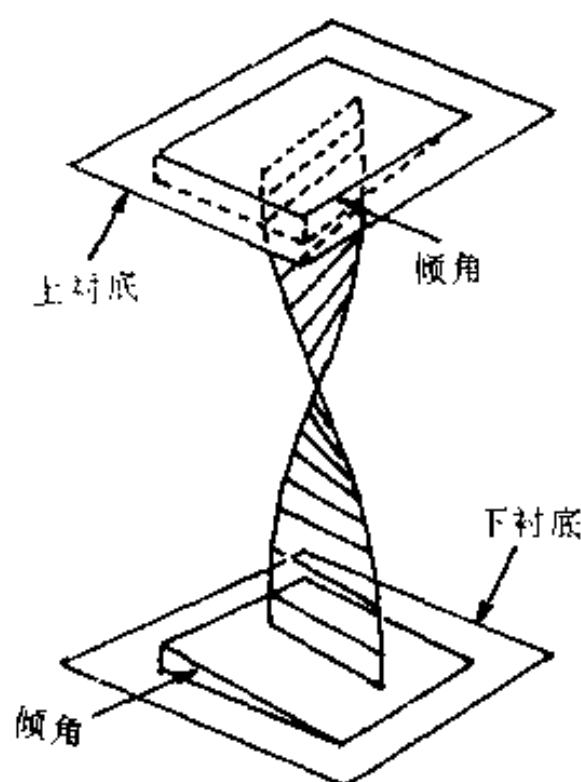


图 2-18 液晶扭向结构

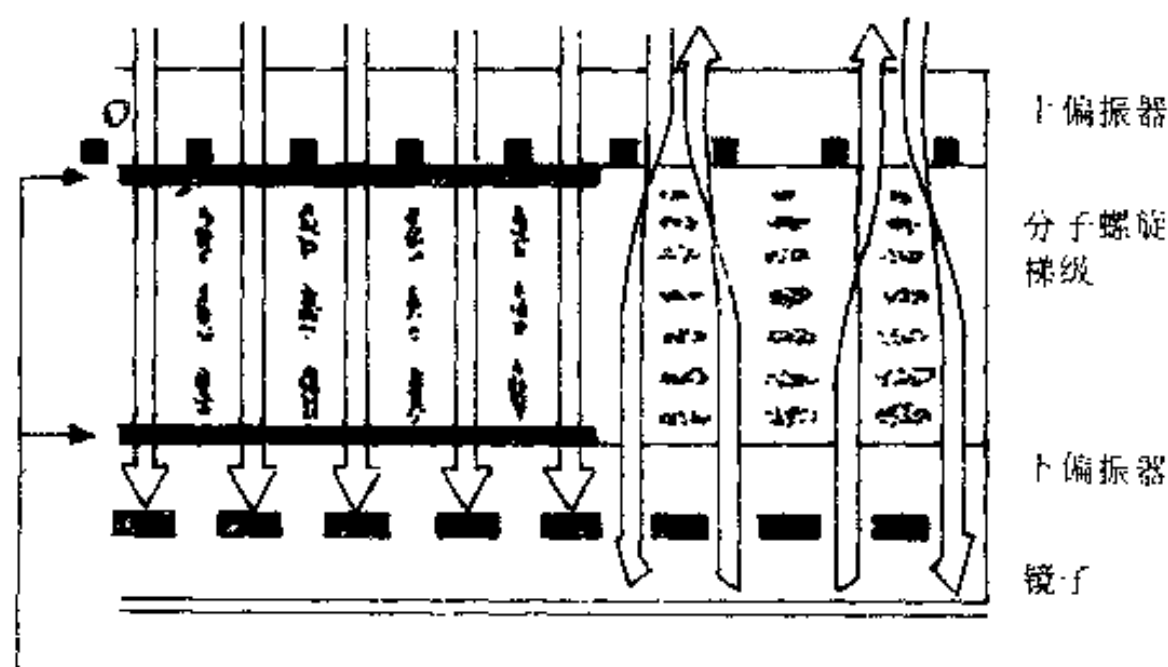


图 2-19 扭向 LCD 截面图

显示器包含有互为交叉的两层偏振膜,见图 2-19。入射到显示器的环境光被第一层偏振膜偏振以后由扭向列液晶转过 90° , 和第二层偏振膜相遇时正好透过,然后由底镜反射回,再次通过整个装置,其间衰减很小。这种状态是液晶未加电时的关状态,液晶屏看上去是灰色的。当加电并超过门限电压时,液晶偶极子向上倾斜,于是破坏了扭向结构。由第一层偏振膜偏振后进入液晶的光线不再转向,结果由第二层(底层)偏振膜吸收。这时液晶屏显示暗色。因此可以通过控制液晶像素上电压施加与否来获得金属灰色为底色的深颜色图象显示。

这种显示方式利用了向列场效应,因此工作电压很低,功耗极省。液晶显示一般可清晰观察的视角范围小,而扭向列液晶显示范围比动态散射方式 DSM (Dynamic Scattering Mode, 由 Heilmeyer, Zanoni 和 Barton 发明,1968 年发布)大得多,其对比度和二色性染料(dichroic dye 1968 年由美国 RCA 发布)液晶显示相当或更好,因此扭向列液晶得到了广泛应用。

此后,LCD 屏技术持续发展,所谓的超扭(supertwist)LCD 显示更上一层楼。以 Zenith Data System 的荧光灯背投式双层超扭 LCD 显示屏为例,它一改过去背景为浅蓝、字符图形图象为深蓝的老面孔;成为白底、黑(或灰)字符图形图象的人们习惯的形式。它的对比度约为 $20:1$,在这以前仅为 $12:1$ 左右。

Zenith 的这种新液晶显示屏用在 80386 膝上型计算机中,采用新型背投光和附加层来获得清晰的显示效果,显示屏的结构见图 2-20。

以往的背投光采用场致发光板(electroluminescent panel),它产生白光,透过 LCD 偏振层后呈淡蓝色,图象等则为深蓝色。Zenith 选择荧光灯管取代场致发光板。荧光灯管产生的光要明亮得多,但是一致性差。于是采用反射器和漫射板结合的办法既保证高亮度,又使光线均匀。荧光灯管是专门设计的,功率为 2.5W ,和一般荧光灯的区别是:光强稳定,可以适应 $10\ 000$ 周以上的开关速度,可

以调节光强,可瞬时启动。

光经过漫射板后依然是白色、非偏振的。紧靠漫射板的是后偏振层,它只让垂直偏振的光通过。然后就是超扭 LCD,这种液晶材料对通过它的光扭转得厉害,每毫米长度上光得扭转几千度。Zenith 的材料每毫米扭转 $30\,000^\circ$,而石英每毫米仅扭转 20° 。

这种高扭转特性使得采用这种材料的液晶显示屏特别薄,光通过它纯扭转为 90° ,当然实际上是多个 360° 加一个 90° 。液晶屏特别薄也带来一些工程问题,面向液晶的玻璃制造必须特别精密,保证液晶厚度一致。微细的差别就会使某部分液晶区域扭转失准,形成污迹

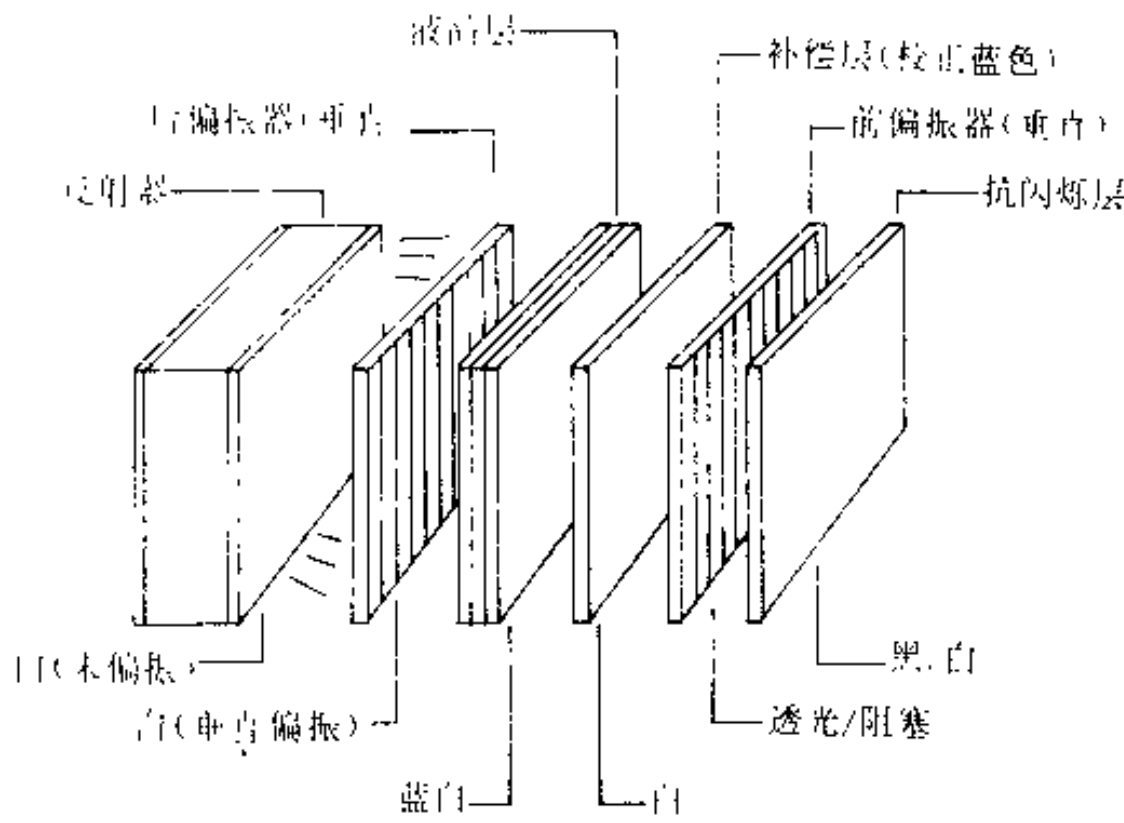


图 2-20 超扭 LCD 屏结构

当在这种 LCD 材料上加上电压时,通常扭曲的液晶就变成非扭曲的了,偏振光通过它们时也不变化。如果光线进入是垂直偏振的,出现时依然是那样。

液晶偏振材料处理蓝光和白光中的其他分量不同,根据材料不同,对蓝光或多扭转了或少扭转了,结果都是较多的蓝光从液晶

中显现出来。Zenith 通过在液晶后加一层光学活化层来抵消蓝光,该层以相反方向等量扭转来达到抵消多余蓝色,起到补偿的作用。因此 Zenith LCD 屏是黑白效果的。由于背投光强,虽然有损耗,亮度依然很强,于是对比度加强了,灰色级也好多了。

目前超扭 LCD 屏价格依然昂贵,用在较高品质的膝上计算机中,为 VGA 分辨率 640×480 像素。日本厂家已生产出了单色超扭 LCD,分辨率已达 1024×768 像素。

2.2.3 有源阵 LCD

有源阵(active-matrix)或薄膜晶体管 TFT(Thin-Film Transister)LCD 比超扭 LCD 有更强的性能,特别在彩色应用场合时更是这样。常规超扭 LCD 采用 x,y 方向的电极条为屏幕像素共享;TFT 显示屏的每个像素有自己独立的晶体管,允许对每个像素直接寻址,比用常规 LCD 屏性能好得多,对比度也好得多。所以说 TFT LCD 实质上是在 LCD 驱动及寻址上的重大进展。TFT 作为半导体开关用于低功耗场合再好不过,用于 LCD 的典型设计是一种 MOS 场效应晶体管,见图 2-21。

晶体管阵通过薄膜工艺做在玻璃衬底上,每个有源单元(晶体管)对应一个 LCD 像素,用 X、Y 阵对晶体管寻址而不是以前的对液晶寻址。当晶体管门打开时,电荷传到液晶电容上,同时也传到一个附加的存储电容里。这些电荷在一幅图的整个时间内支撑显示。因为液晶是一种有损耗的电介质,所以需要附加电容上的电荷来补充,使得晶体管门关断、像素和驱动电路隔离时像素仍保持显示。

TFT LCD 设计思想是漂亮的,不过要把它付诸实施远不那么简单。早期在 TFT 的工作集中在硒化镉(CdSe)上,因为它具有高流动性以便获得薄膜。后来用非晶(amorphous)、多晶(polycrystalline)和激光再结晶(laser recrystallized)等形式构成硅 TFT,因为在硅处理方面人们已有了大量的经验。

80 年代早期 TFT 已经可以做得特性一致,问题是怎样以足够

低的成本来获得统一开关特性的大尺寸晶体管阵。工艺困难之一是利用低温处理获得高机电品质的介电层,其二是制造高质量的流动性一致的半导体材料。另外一个问题是,液晶像素空间密度高时(比如大于每厘米 11.8 线,即每英寸 30 线),它的阻抗很高。半导体开关的电阻性和电抗性阻抗特性这时变得愈发重要。1982 年,用于实验室中的 CdSe 阵,大小为 $13 \times 13\text{cm}$,阵列为 250×250 像素。当时的硅阵大小还要小些。

最终的 TFT 要包含移位寄存器,对 N^2 像素阵的 $2N$ 条引线寻址,这样和 TFT LCD 显示屏的连线就大量减少,这在大信息量、合理成本下是必须的。

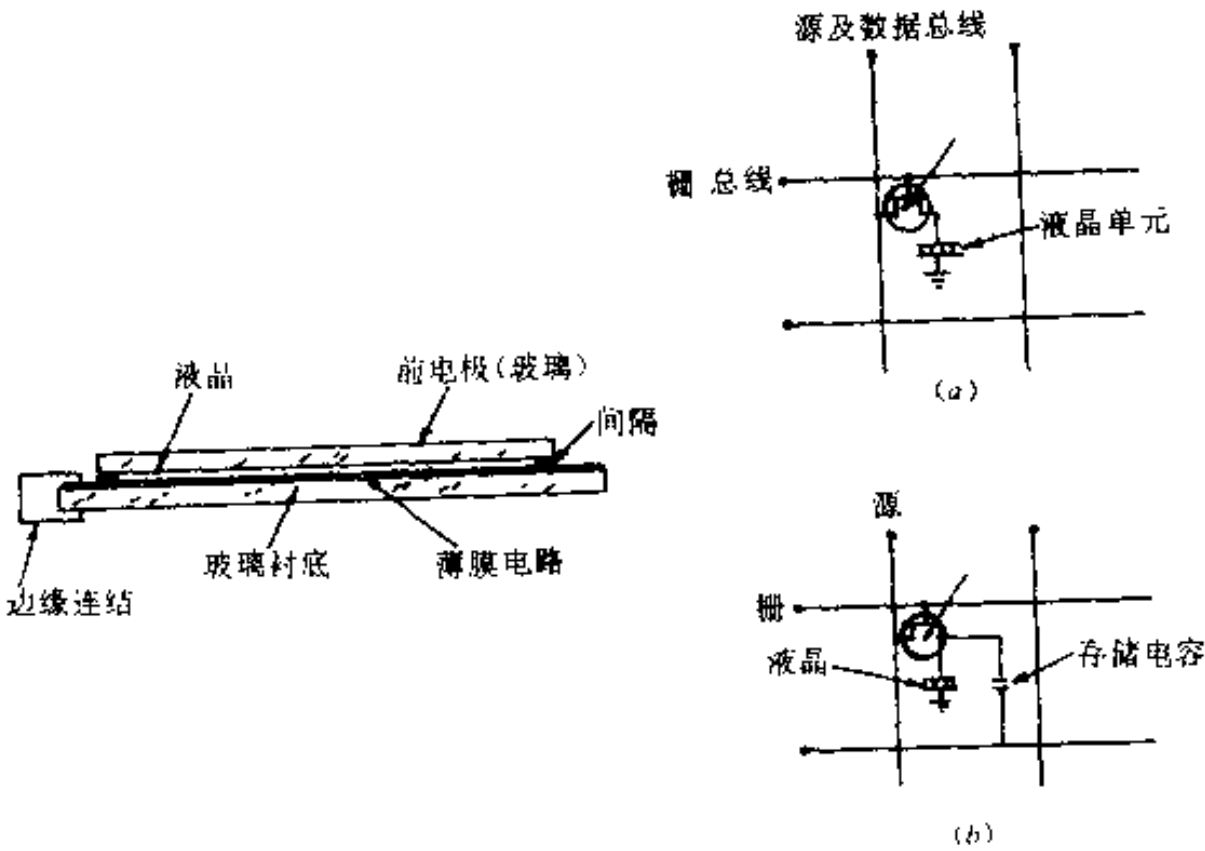


图 2-21 TFT LCD 结构及原理

彩色 TFT 显示屏中,每个像素有三个晶体管来确定红、绿、蓝各分量。此外,TFT 还加有一个彩色滤光层。

彩色滤光层由交替的红、绿滤光材料条组成,这些滤光条可以是水平方向,也可以是垂直方向,复盖在控制像素红、绿、蓝色彩特

性的晶体管上。

要使某一像素形成某种色彩,驱动电路驱动该色彩相应的晶体管,它解除(或是说破坏)液晶中向列材料的扭转状态,使背投光通过液晶到滤色层,然后光就取相应滤色层的颜色在屏幕表面显现出来。由于光通过偏振滤色层,只在一个方向上辐照,因此产生的图象比常规 CRT 技术的更清晰。

每个像素三种基色可产生八种颜色,视像素三个晶体管的状态而定。进一步开发的显示器会有更多色彩,可以通过改变透过滤色层各彩色的光强来达到。

因为每个像素需要三个晶体管,彩色显示在水平或竖直方向的分辨率视采用的滤色条是水平还是垂直的而定,只是单色的 1/3。也可以使彩色滤色层不工作,生成较高分辨率的单色显示。

彩色 TFT LCD 比单色 TFT 复杂得多,制造也愈发困难,目前它们可用的品质较低,价格昂贵。

单色和彩色 TFT LCD 都需要背投光。美国苹果公司的麦金塔什便携机(Macintosh Portable)采用单色 TFT 液晶显示,耗电 1W,用的是反光式而不是背投光式,因此功耗低,但是可读性较差,受环境照明条件限制。反光式在单色显示中尚可行,在彩色显示中普遍采用背投光照明来充分表达色彩效果。

需要背投光或许是 LCD 显示装置工业面临的最大障碍,比如 Mitsubishi 彩色 TFT,16 色、 640×480 像素、26cm LCD 显示需要 135W, NEC 彩色 TFT, 640×400 像素、18cm(7 英寸)监视器需要 90W。因此,还有很长的路要走,在 TFT LCD 方面日本公司拥有明显的优势。

TFT 彩色 LCD 目前支配了膝上计算机市场,相竞争的其他技术会使 TFT 面临剧烈的竞争。

2.2.4 迭层式无源阵 LCD

迭层式无源阵(stacked-panel passive-matrix) LCD 系统由两层或多层 LCD 板堆迭起来,光通过该迭层时,其光谱各部分受到不

同的削弱,这样整块板产生全范围真彩色。

也就是说,迭层式无源阵不象 TFT 显示那样用独立的晶体管对每个象素进行寻址,而是用常规的 LCD 板对准迭放在一起。每块板削减一种原色:青、品红及黄。当所有象素关闭时,背投光提供白色背景,只要 LCD 板中有象素开通,它在白光背景上减掉相应的黄、品、青而构成其他色彩。

美国俄勒冈州的 In Focus System 公司领导迭层式无源阵 LCD 技术的新潮流,它原先专门生产 LCD 投影板,在 1990 年 11 月的 Comdex 展览上推出 26cm 全彩色显示屏,由三层超扭 LCD 板构成。监视器为 VGA 640×480 象素分辨率,色彩可达 4913 种,采用 50W 白炽灯背投光,重约 2.7kg。

由于迭层式无源阵采用常规 LCD 板,没有目前阻碍 TFT 技术的制造问题,这种显示技术大有希望。另一方面,这种无源阵显示系统需要有含镜子和反射器的复杂光学系统来实现背投光照明,这就增加了体积和功耗。

这种显示装置对台式机来说有相当的潜力,要用在膝上机中尚需进一步细化。迭层式无源阵显示还有响应时间方面的问题,目前大约为 250ms,对显示移动的图象尚不够快,有待改进。

2.3 气体等离子体显示装置

2.3.1 气体等离子体显示原理及发展

气体等离子体(gas-plasma)显示通常对氖或氩氖混合气体施加电压激发实现。这种显示有交流和直流两种。其行、列电极由其间的象素孔板中的气体隔开,象素可由行列电极定址。在选定的行列电极上加上足够电压,行列电极交叉处的象素中的气体受激,发出桔红光。直流型的气体等离子体显示装置较易制作,但是通常有固有的背景辉光,原因是刷新需要持续的直流电压。交流型的气体

等离子体显示装置要复杂得多,它没有背景辉光,也没有图象刷新要求。气体等离子体显示的原理见图 2-22。

等离子体显示应用范围很宽,小到七段数码显示,大到数百万个象素,屏幕对角线超过 1m,效果良好。到 1983 年初,等离子体显示支配了大面积平板显示技术市场,成为达到实用阵列寻址的最早的技术之一。这些都有赖于等离子体显示的如下一些主要特点:

- 非线性强。要获得象素阵列可寻址的特性,显示单元的非线性特性越强,可显示的线数越多。气体放电的非线性非常强,在激发电压以下,实际上没有光发射。

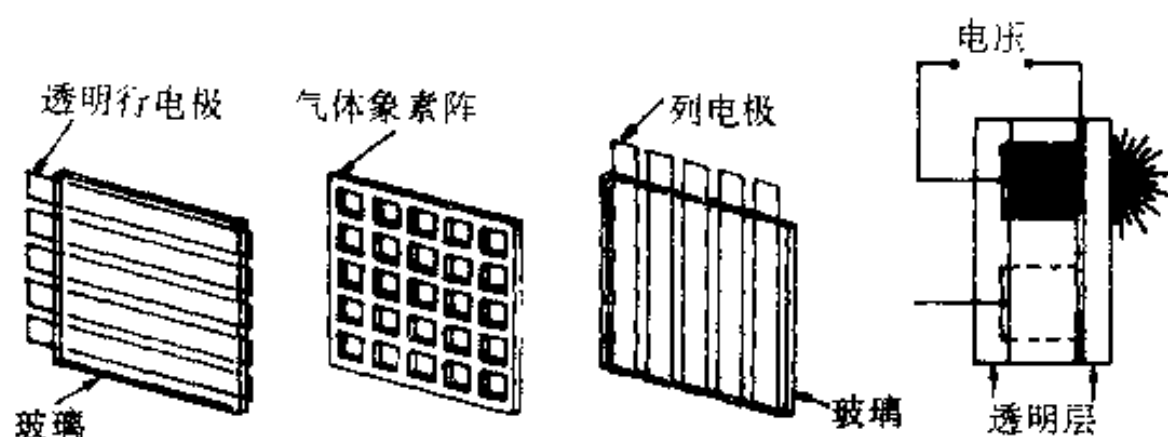


图 2-22 等离子体显示

- 亮度和发光效率良好。商用显示最大亮度在 0.01~0.5 朗伯,发光效率以氦基气体为例在 0.1~0.5lm/W,具体视混合气体成分而定。虽说这些指标和别的显示技术比不见得更好,但是,由于显示介质透明,环境光大都由衬底上的金属电极反射,这种反射可以很方便地用圆偏振滤除掉,因此,即使环境光很强,对比度仍然很出色。实验室中等离子体单元已获得峰值亮度在 100 朗伯,发光效率超过 5lm/W。

- 结构简便、可靠。等离子体显示通常由玻璃和金属构成,结构简便,适应大量装配,材料成本低。由于结构一体化,就象一块玻璃那样耐震,可以用于冲击性强、振动频繁的诸如军用那样的环境。

• 寿命长。等离子体显示寿命一般超过 50 000h。说它失效不是完全不能工作,而是指显示质量下降,越是经常放电的像素会越早变得暗弱。

• 分辨率高,尺寸大。商品等离子显示屏的分辨率为 39.4 行/cm(每英寸 100 线),有的更高,对角线 1m(39 英寸)的显示屏在 1981 年就已推出市场。

• 介质透明。等离子体显示利用气体,它不仅透明,折射系数为 1,避免了其他大于 1 介质所引起的损失,而且可以和投影合用。比如投影一幅地图,而用等离子体显示来指示地图上不同区域的活动。如果在—块绿色厚膜交流场致发光板前放上透明的桔色等离子体显示,彩色对比是相当好的。

• 阻抗特性合理。由于气体的介电常数非常接近于 1,和其他阵列显示技术,比如薄膜场致发光相比,等离子体的显示电容小,因此驱动电流也可小些。工作电压高,一方面是缺点,另一方面电流小,只需薄膜导体就行,这又是优点。

• 平板显示。等离子体显示具有平板显示的所有优点:体积小、形状薄没有显示畸变。

因为气体等离子体显示是一种发光设备,它不需要背投光,但它比有背投光的 LCD 费电。由于工作电压高,不能直接用在以电池为电源的计算机上,如果采用转换电路把用电池作为电源的电压升高(直流约为 200V),又增加了功耗、重量和复杂性。

最初气体等离子技术仅提供亮暗开关控制,现在在最亮和最暗之间可以有若干级(比如 4 级、16 级)桔红色“灰”度。

气体等离子显示面临许多问题。它缺乏全彩色能力,相对来说寿命短、价格昂贵。1985 年,IBM 公司的 14cm(17 英寸)气体等离子体显示屏要 3300 美元,无法与 CRT 匹敌。另一家公司从 IBM 公司购买了交流等离子体显示技术并加以改进,用减少驱动器数及功耗来明显降低成本。

在 1988 年 5 月的 SID 展览上,日本 NHK 实验室对它的全彩

色 51cm(20 英寸)直流等离子体显示屏(准备用于高清晰度电视)展开了讨论。Fujitsu(富士通)也发布了 21cm 及 39cm(15 英寸)的彩色等离子体屏,它们的象素分辨率分别为 76 000 和 256 000。更重要的是其寿命已和 CRT 相仿。和 LCD 一样,人们正在竭尽全力从事大尺寸、全彩色气体等离子体显示屏商品化的工作。

2.3.2 直流等离子体显示

等离子体显示可以有两种工作方式,一种是存储器方式,另一种是刷新方式。

存储器方式的优点是显示亮度比刷新方式强。不过,每个象素需要一位存储器,同时在直流显示时各象素必须有一个限流电阻。这个电阻范围在 $10\text{k}\Omega \sim 1\text{M}\Omega$,这种阻值的小电阻既要耐高压、又要便于和每个等离子体显示象素相串联,制造起来非常困难,因此直流存储器式等离子体显示实用化很困难。

最早的实用气体放电显示器件是 50 年代开发的电子管式辉光数码管 NIXIE,它有十个阴极、一个阳极。阴极形状为数字,阳极通过一个 $10\text{k}\Omega$ 电阻接在 170V 电源上。阴极接地时,它被辉光包围,形成相应的发光数字。其他的阴极有 40V 的偏压,不发光。

而后的分段显示克服了辉光管 NIXIE 又大又难组装的缺点。尤其在需要大量数字时,NIXIE 是不实用的。分段直流显示大多数用在相对来说较小区域的显示,特别大的显示屏可由各个小屏拼成。这时显示内容为数字和字符,不能显示图象。

因此,大量厂家致力于生产直流点阵显示。所谓的点阵(dot matrix)显示是指用一组 X,Y 电极(不采用移位寄存器的原理)控制点状象素阵进行显示,采用移位寄存器的显示将在后面阐述。

大多数点阵显示的基本操作如图 2-23 所示。每一列象素单元与在同一垂直电极相连通到一个扫描开关上,水平行上的象素单元与同一水平电极相连接到数据开关上。

图中黑点表示当前扫描开关和数据开关闭合的情况,导致字母 A 的起头竖直笔划中的相应象素发光。打叉的诸点表示不同扫

描时刻需要亮的象素(以便显示 PLA)。扫描速度必须很快,以便消除显示闪烁现象。注意,这里的限流电阻和数据开关串联。实际应用时,和开路的扫描开关、数据开关相连的电极并不浮空,而是接在一定的偏置电平上。

点阵显示的电极之间距离很近,为了避免相邻电极的误放电,采用各种隔离方法。可是隔离产生了副作用,象素单元启辉困难,因此需要辅助启辉。

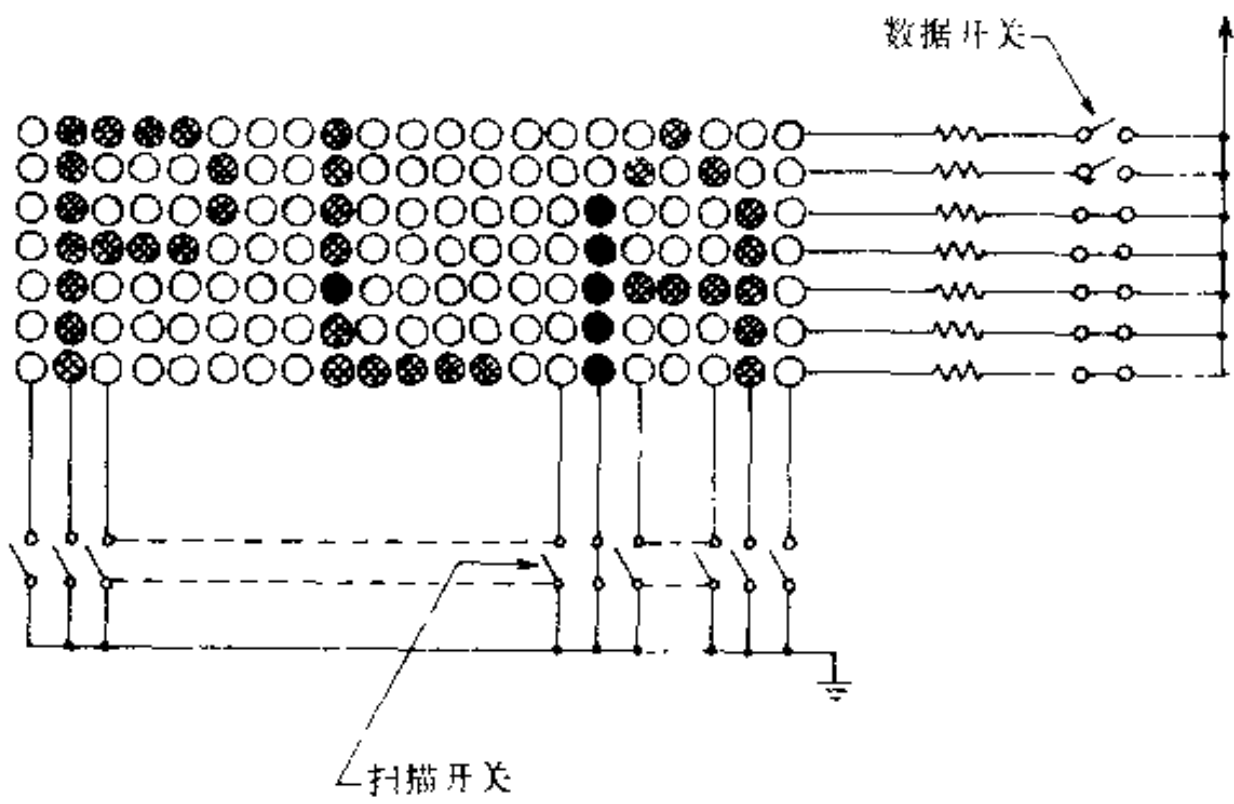


图 2-23 点阵显示扫描技术

所有点阵显示的一个主要问题是驱动电路数量多(因为电极多),因此成本高。伯尔罗夫斯(Burroughs)开发的称作“自扫描”(Self-Scan)显示通过大大削减外部电极数来降低成本。这时扫描开关按在显示屏内部的逻辑来执行,仅需很少外部电极(4~8条)。表 2-1 为点阵显示和自扫描显示在所需驱动器方面的对比。

图 2-24 中可以看出每个显示象素分成两个隔开的单元:一个是用户可见的显示单元;另一个是在显示单元后面、用户见不着的

扫描单元。分隔这两个单元的是带有小孔的阴极,小孔可允许启辉粒子通过。

表 2-1 两种显示所需驱动器对比

5×7 字符数	点阵	行驱动器数	列驱动器数	驱动器总数	非 Self-Scan 时总数
1 行 16—16	7×96	7	4	11	103
1 行 40—40	7×283	7	12	19	290
6 行 40—210	42×283	42	12	54	325
12 行 10—480	84×283	84	12	96	367

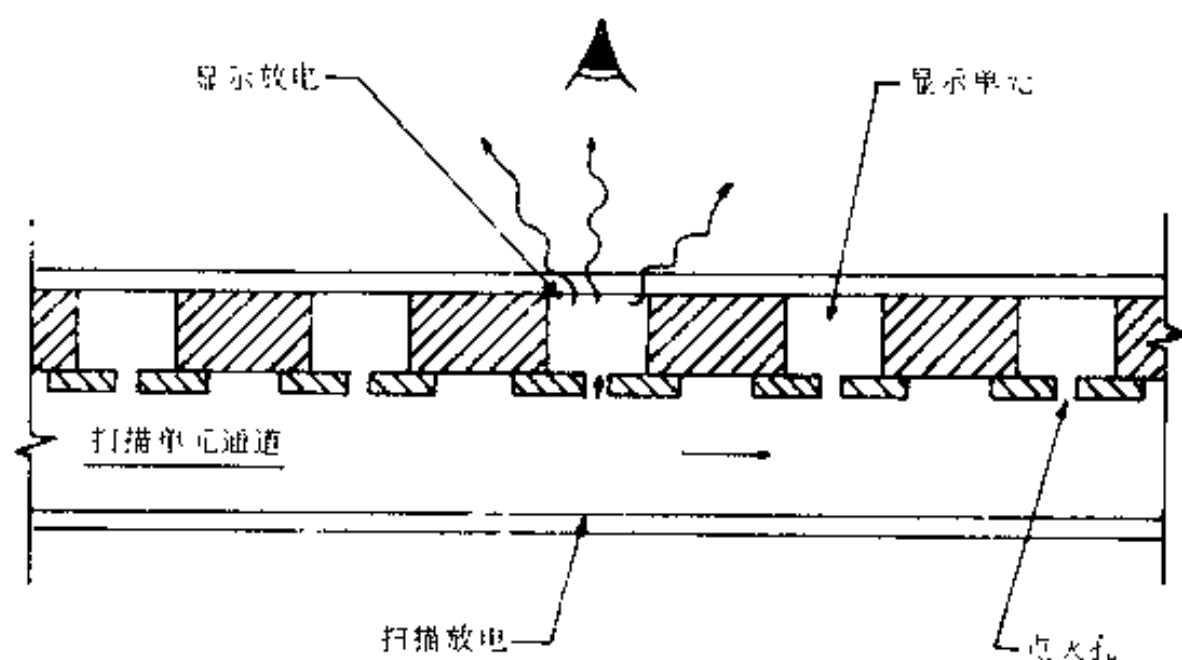


图 2-24 显示单元和扫描单元通道

图 2-25 说明了扫描放电是如何依次从一个象素移向下一个象素的。图中只画出了和扫描放电正常工作必需的电极。请注意,除了复位阴极外,第三阴极均连在共同的总线上。这种连结可以在显示屏内部完成,所以该一整行的扫描放电只需 5 条外部连线来控制:三条阴极总线、复位阴极和扫描阳极。

加在阴极上的信号在图 2-25 的右边。放电次序由复位阴极 Φ_R 、接地开始,这时启动复位阴极和扫描阳极间放电。复位后, Φ_R 电压升回到 +85V,在阴极 1、4、7……上的电压 Φ_i 降到零。由于各

单元这时都加有 250V 电压,有人以为它们全都会放电,其实不然,因为:

1. 所有扫描单元有一共同的阴极限流电阻,一旦有单元开始导通,该电阻上的压降使其它显示单元上的电压下降而阻止它们放电。

2. 接到最多的启辉粒子的单元会首先导通,这些粒子是最近放电的单元中最强的。

以图 2-25 为例,在 Φ_1 脉冲 1 时刻,只能在阴极 1 下有放电,因为启辉粒子从邻近的复位放电而来,而单元 4、7……远离复位阴极,一旦阴极 1 放电,加在它们上面的电压降低而没有机会放电。同样道理,下一步是 Φ_2 脉冲 2 使受阴极 1 放电产生启辉影响的阴极 2 放电……。当放电抵达扫描通道最右端时,扫描次序又回到复位阴极。

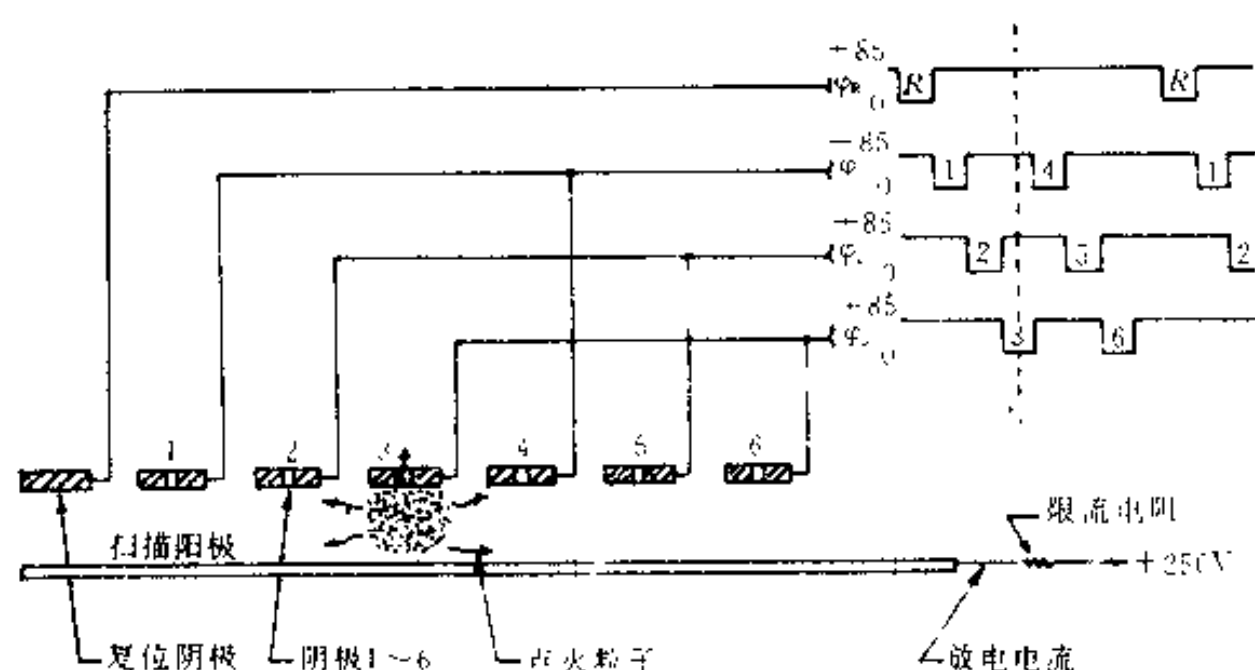


图 2-25 扫描放电示意图

DC(直流)等离子体显示有多种方案,包括不同的扫描技术,也包括采用荧光材料来实现彩色等等。

2.3.3 交流等离子体显示

交流等离子体显示的基本原理的特点是利用电容夹作为限流元件。

交流等离子体显示方式也有两种：存储器方式和刷新方式。采用交流限流的等离子体显示，其存储器特性是独特的。这是因为电荷可以积累在电介质壁上，一旦写入图象，它能保存下去。这样不仅消除了闪烁，而且无需外部存储器。不过，最大的好处还是：内部存储器可以使各像素在整个工作周期工作。而扫描显示的工作周期为扫描线数所瓜分，由于气体放电的饱和效应，工作周期越短亮度也就越暗，当刷新显示扫描行越多，显示也就必定越弱。拥有内在存储器的显示，不论显示屏多大，亮度不变，所以最大的等离子体显示装置是采用交流、存储器式显示的。

1983年初，美国佛托尼格斯(Photonics)公司推出显示区 $60 \times 80\text{cm}$ (对角线 1m) 的等离子体显示屏，装置仅厚 12cm ，分辨率每厘米 19.7 线(每英寸 50 线)，构成 1212×596 象素阵，差不多为 2×10^6 个象素点。它是透明的，把地图放在显示屏后，清晰可见。

它的结构为欧文斯-伊利诺斯(Owens-Illinois)，特别适用于交流存储器式点阵显示。在这之前伊利诺斯大学发明的交流等离子体显示结构如图 2-26 所示。

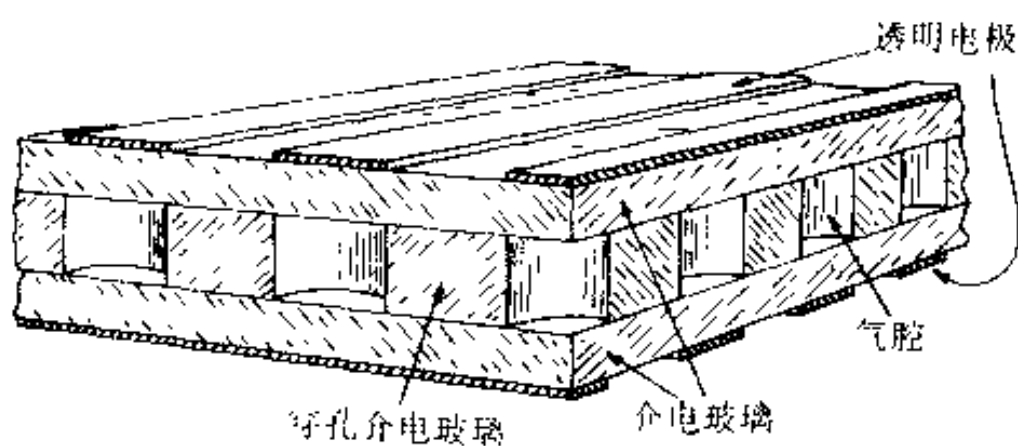


图 2-26 伊利诺斯屏结构

它为大家提供了很多可学习的东西,但是薄玻璃特别易碎,而且气体必须用环氧物来密封,极易混杂。因此这种器件寿命很短。结构中的中间层玻璃上布满小孔,那就是显示单元。从分隔外层玻璃形成气隙来说是必须的,而各单元不必互相隔离。

欧文斯-伊利诺斯完成了对伊利诺斯屏的重大改进,如图 2-27 所示

衬底玻璃通常厚为 6.3mm, 不易破碎,采用全玻璃封装,可在高温下驱除杂质保证长寿。中间层被取消,衬底有足够刚度保持气隙空间。各显示单元由电极产生的电场规定,电极交叉点就是放电像素点。另一个改进是,欧文斯-伊利诺斯结构中的介电玻璃比较薄,加在气体上的电

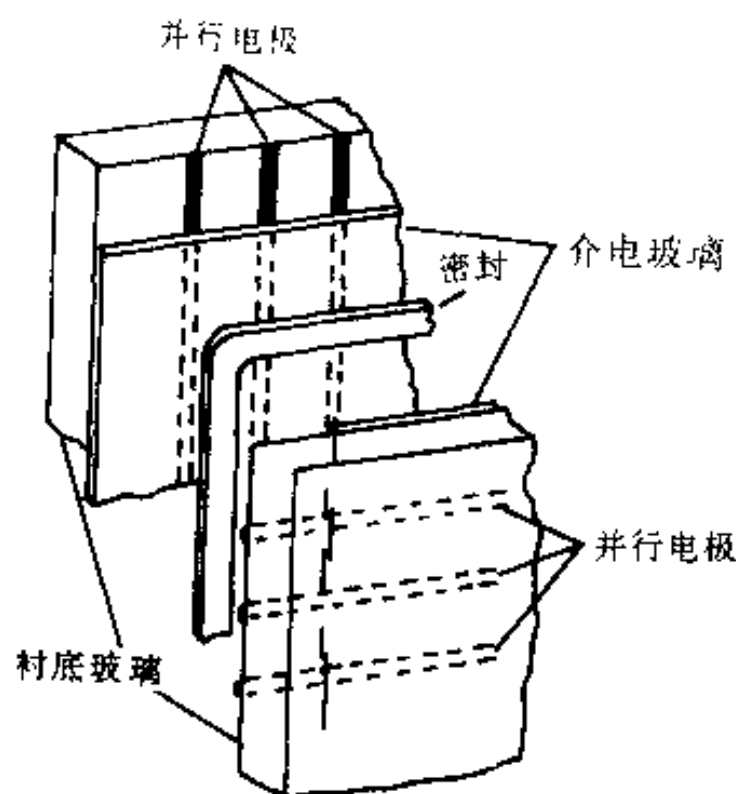


图 2-27 欧文斯-伊利诺斯屏结构

极电压比例就大,因此显示屏的工作电压可以降低在 $\pm 90 \sim 100\text{V}$ 。

图 2-28 所示为典型交流点阵存储器式等离子体显示屏的结构。

衬底为普通玻璃,电极薄膜金、铬-铜-铬或铝,当把焊接玻璃涂复在衬底上作为电介质时,它们能经受超过 600°C 的峰值温度。有些显示屏用在相对比较低的温度上蒸发的薄膜电介质来代替。气体用普通的潘宁混合气(Penning mixture, 氖加很少一点氩,比如 0.1% 的混合气体),工作电压低。气压比大气压低,避免衬底

外胀破坏气隙距离,但又要尽可能高,形成尽可能小的气隙来获得尽可能高的分辨率。

该器件最讲究的部分是 200nm(纳米)的氧化镁薄膜层,它复盖在介电玻璃上,其作用有两点。一是,氧化镁(MgO)是高 γ 材料, (γ 是二次电子系数,小于 1)因此点火电压低,而且该 γ 值长期稳定,使得该器件寿命长。再就是氧化镁是一种耐火材料抗溅射。器件不用直流等离子体显示所必需的水银,工作寿命很长。

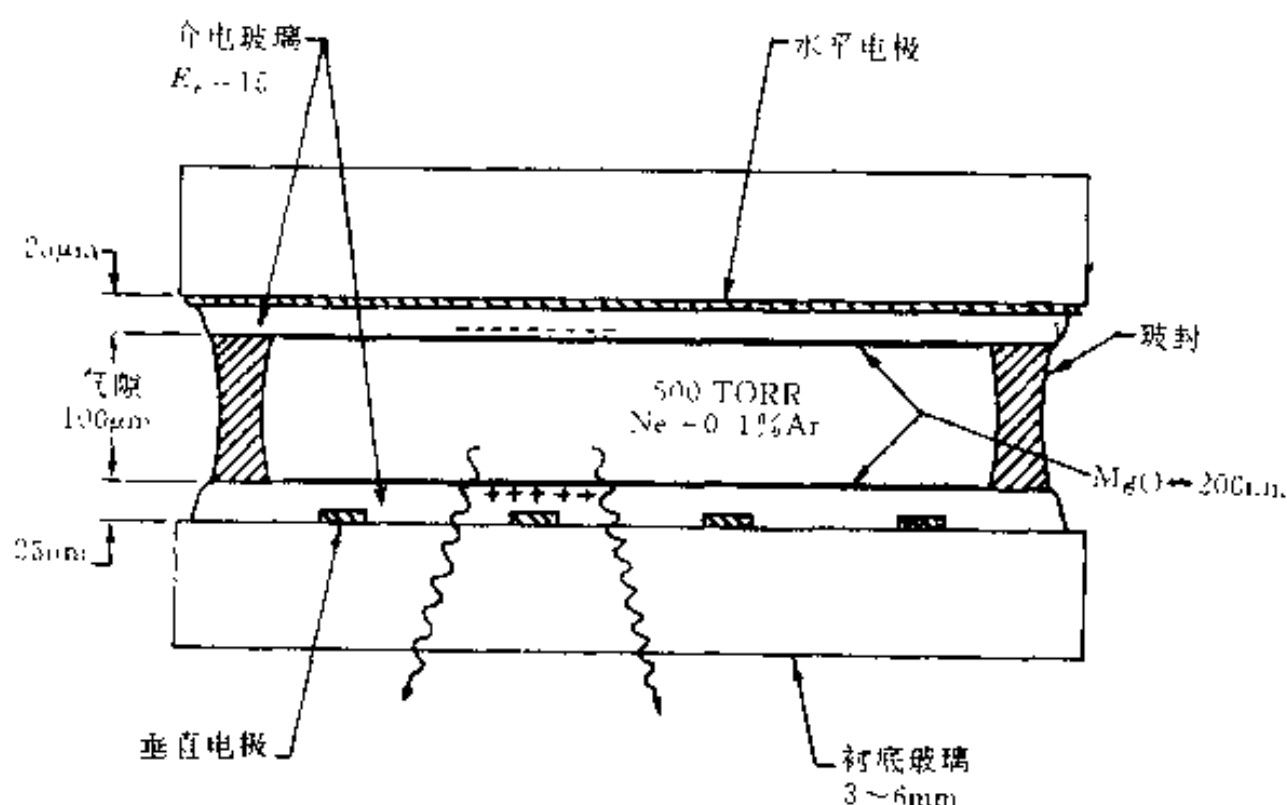


图 2-28 典型交流等离子体显示器结构

交流等离子体显示的电模型如图 2-29 所示。

交流等离子体显示必须加有称作保持电压(sustain voltage)的交流电压来维持放电。电模型中的 C_d 为介电玻璃电容, C_g 为气隙电容。气体放电由非独立的电流源代表,它受多个参数比如加在气隙上的电压等的影响。根据前图中标的尺寸和介电常数, C_d 比 C_g 大 60 倍,可以认为介电玻璃壁上无电荷,全部保持电压加在点 a 和 b 间的气隙上。

当气隙上电压超过点火电压时，气体放电电流开始流动，它把电荷堆积在介电玻璃壁上，这种对电容的充(或放)电将使气隙上的电压降低而使气体放电熄灭，但当保持电压反向时，刚才堆积的电荷形成的电压和它迭加形成高电压又使气体放电，这就是象素的开状态。如果象素处在关状态，介电玻璃壁上无电荷，则保持电压不会使它放电，保持关状态不变。

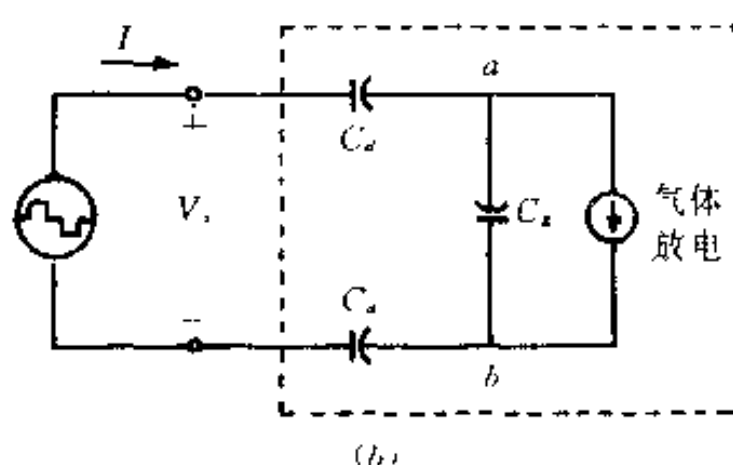
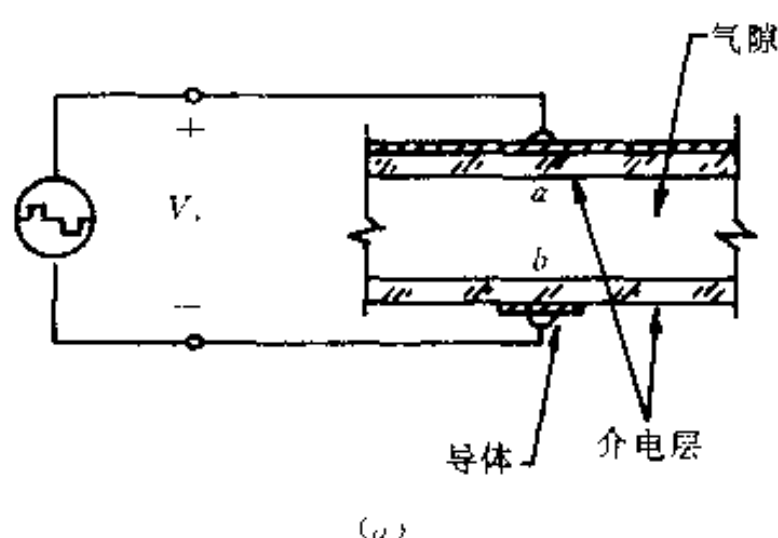


图 2-29 交流等离子体显示的电模型

保持电压的脉冲宽度通常为 $5 \sim 10 \mu s$ ，频率为 $50 kHz$ ，脉冲峰值为 $90 \sim 100 V$ 。由于频率高，由放电发出的光对人眼来说是稳定的。保持电压、介质壁电压和光输出的关系如图 2-30 所示。

产生保持电压波形的线路通常如图 2-31 所示。

保持电压电路一般和地址驱动器分隔开。这主要是经济原因。因为每条显示电极必须有一地址脉冲发生器，因此它是系统的一种主要成本。于是地址脉冲的幅值要调得尽可能小来降低成本。保持脉冲时常明显高于地址脉冲，前者典型值为 $90 \sim 100 V$ ，后者为 $60 \sim 70 V$ 。由于幅值不同，为了经济起见采用分别不同的电路，也就是说用一个保持电压电路来满足整个显示的需要，而地址脉冲发生器是每行一个。也有用两个保持电压电路的，一个用于 X 轴，

一个用于 Y 轴。

由于显示行数多，一般都希望地址驱动器集成化，得克萨斯仪器公司 (Texas Instruments) 已经开发了 SN 75500 正地址驱动器和 SN 75501 负地址驱动器。

交流等离子体显示也有采用刷新方式的，其中有的是分段显示，有的是点阵显示，有的采用移位技术以减少电路成本 (大多数情况下，电路费用超过显示屏本身)。

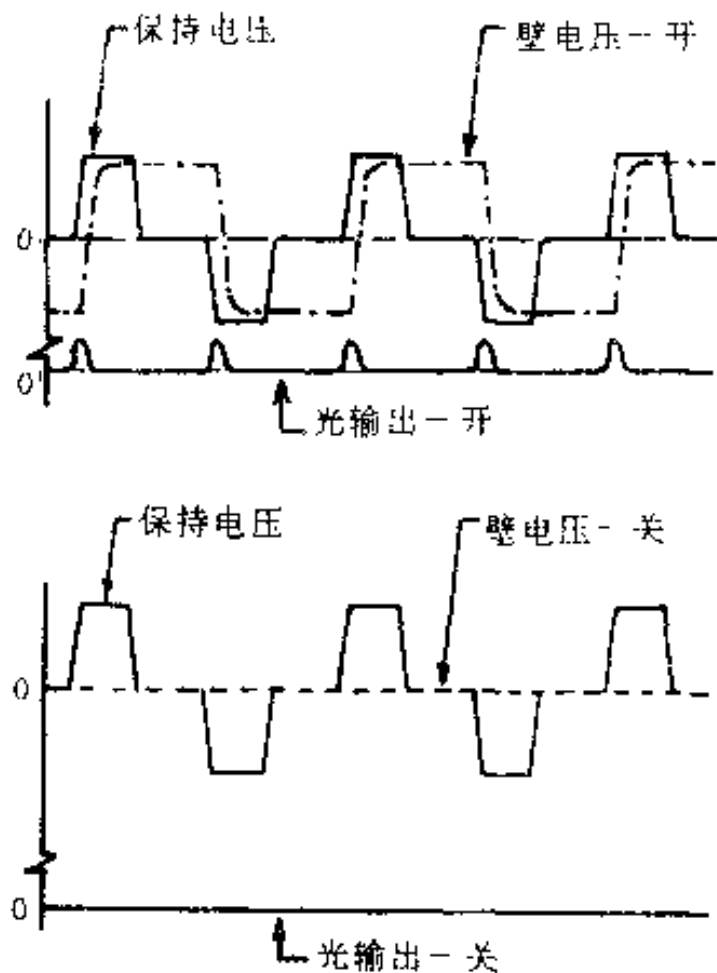


图 2-30 电压和光输出的关系

交流七段显示比直流对手有明显优势。首先，笼罩在电极上的辉光强弱一致性好，原因是用电容限流，而电容分布在电极面上，辉光也是分布状的。在直流分段显示中，直流电极的所有区域互相竞争从单个限流电阻来的电流，因此有的区域电流大些因而亮些，有的就差些。交流显示的这种优点使得显示单元可有 $1 \times 3\text{cm}$ 这样的大尺寸，利用这种分布状辉光特性，加上高频保持电压，所产生的高亮度在阳光下依然可读。

其次，交流显示比直流显示有更宽的工作温度范围。基本原因是，直流器件需要水银而交流器件不用。低温时水银蒸汽压力太低，不能起到抗溅射的作用。当直流器件工作在低温时会缩短寿命。NCR 公司规定气体放电工作温度范围在 $-40 \sim +85^{\circ}\text{C}$ ，低温端对交流器件来说再延伸到更低的温度也不会有什么问题的。由于不采用存储器方式工作，器件对电压变化要求很低，由温度引起的参数变化都不紧要。因此交流器件的温度限制常取决于驱动电路

的温度限制。

交流器件的显示亮度和驱动频率成正比,比例因子大约为 $3\text{fL}\cdot\text{kHz}$, 500fL 的亮度需要 167kHz 的驱动频率。

点阵显示和分段显示不同,分段显示单元少,点阵显示在扫描工作状态时交直流器件遇到同一个问题,这就是亮度问题。为了提高亮度,采用纯氖气(它比潘宁气发光亮),不过这时工作电压要高一些,但还是值得的。其他措施还有面板上采用透明电极,提高保持电压频率等。其中提高保持电压频率的方法最见效果。存储器式交流显示其保持电压频率不能超过 50kHz ,而刷新方式不受此限, 500kHz 的保持电压频率可提高亮度 10 倍。

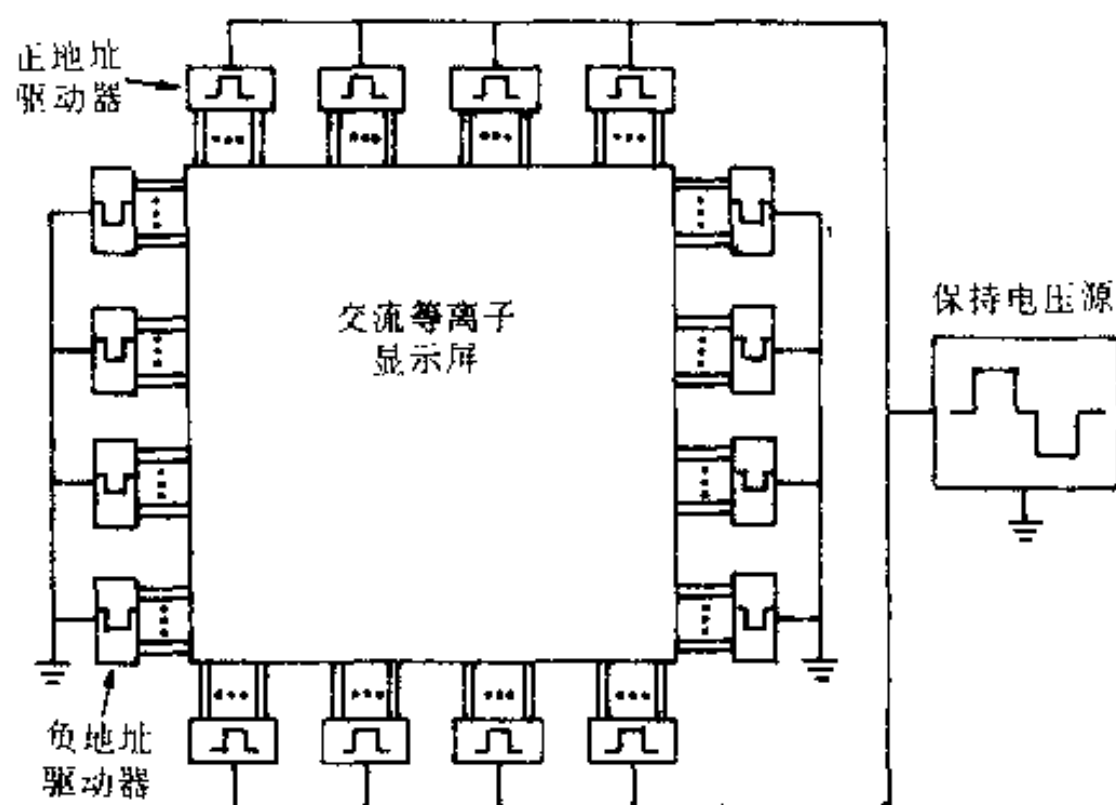


图 2-31 典型驱动电路

2.3.4 交直流混合等离子体显示

为了利用交流等离子体显示和直流等离子体显示两者的优点来制造更高性能的显示装置,设计者同时采用交流和直流限流技术,这时的显示装置中有的气体放电是交流式的,有的是直流式

的。

以 2.3.2 直流等离子体显示中的标准“自扫描”显示为例,它由于内部气体放电逻辑的聪颖设计,所需驱动电路数量少;但是它采用刷新显示、没有存储器,因此亮度及每行中能含象素数的最大值都受限制。通过加入存储器的方法构成自扫描存储器式显示就排除了这些限制。

自扫描存储器式显示装置结构可以看成前面有交流存储器式显示的直流自扫描显示。用户只能看到交流部分。象素寻址由直流自扫描部分完成,象素存储器功能是交流部分的事。存储器功能使得显示亮度高,不受显示尺寸限制,而且没有闪烁现象。

该装置的结构如图 2-32 所示。

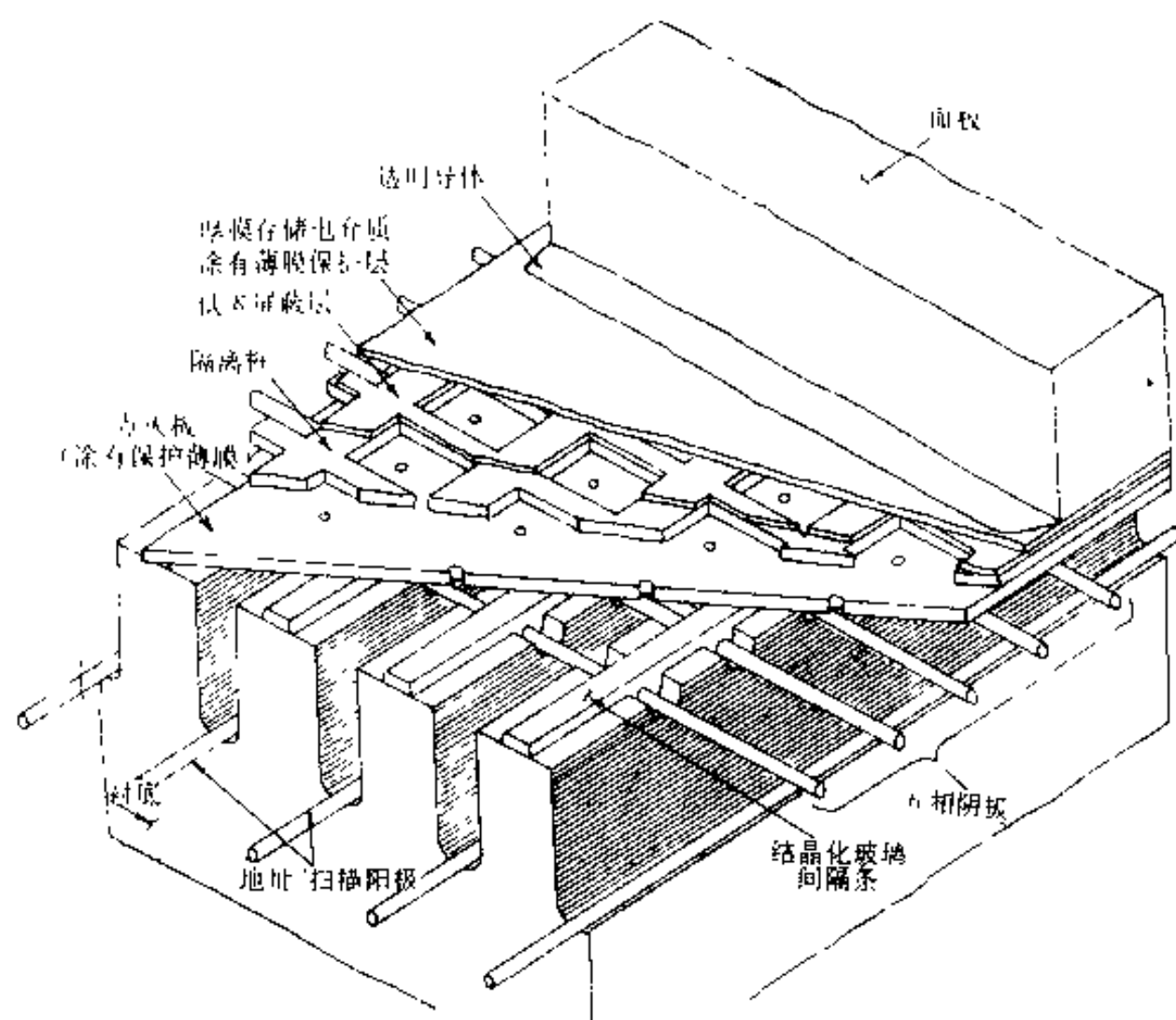


图 2-32 存储器式显示器结构

显示的交流部分在启辉板上面,启辉板和面板上的透明电极构成两个交流电极,它们都覆盖整个显示屏区域,由介质辉光隔离层把它们分开并构成像素。透明导体涂复有标准交流介电玻璃和诸如 MgO 这样的薄膜保护层。启辉板上没有介电玻璃,只是薄膜保护层。和其他交流显示不同,它使用只对一个电极的介质电容来限流。这对交流限流来说是一种完善的技术。交流保持电压加在上述两个电极上,维持像素的开或关状态。

直流部分在启辉板下面。它的工作和标准自扫描显示的扫描放电部分类似。一个重要的区别是,扫描不是不停工作的,而是需要寻址时才做。同样,只有选中的扫描阳极才放电,这也要看寻址的是显示屏的哪一区域。这种技术节省功率、延长显示寿命。阴极线卧在闷光玻璃(devitrified glass)隔离条下,防止和启辉板短接。

对该显示屏进行写像素操作的步骤是这样的:首先,有合适的扫描放电沿着扫描通道移向选中地址的像素下面,如图 2-33 所示的时刻 1 状态。当扫描放电在该像素下发生时,扫描阳极加负脉冲 $1\mu\text{s}$ 而保持电压信号在正半周,于是放电中断,扫描辉光中的电子从启辉板小孔中流进显示屏的交流部分,见图 2-33 的时刻 2。这些电子改变了交流部分介质壁上的电压,完成像素显示的写操作。

要擦抹一个像素,方式是类似的。当保持电压信号在中值并当交流部分介质充上正电时扫描辉光中断。扫描辉光中的电子从启辉板孔中穿过,中和正电荷,使交流部分回到关状态。

2.3.5 图象显示

至此为止,所讨论的等离子体显示都是表现字符、图表类的,像素只有两种状态:开或关。图象显示则要求有灰度和色彩,这是很有挑战性的。目前等离子体显示尚不能和具有霸主地位的 CRT 相比拟,尤其在图象显示方面。但开发研究(包括混合技术)还在不断发展。

交流和直流等离子显示都可以获得灰度。方法可以是放电电流的模拟调制(analog modulation),也可以是对一定的放电电流采

用占空度调制(duty cycle modulation)。

交流等离子体显示在图象显示方面的工作要晚一点,原因可能是人们有交流显示不易调制的错觉。交流显示只有两级亮度:亮或暗,人们在这方面取得过三级亮度的成功,但离实用很远。因此人们的注意力集中在占空度调制和空间调制(spatial modulation)上。

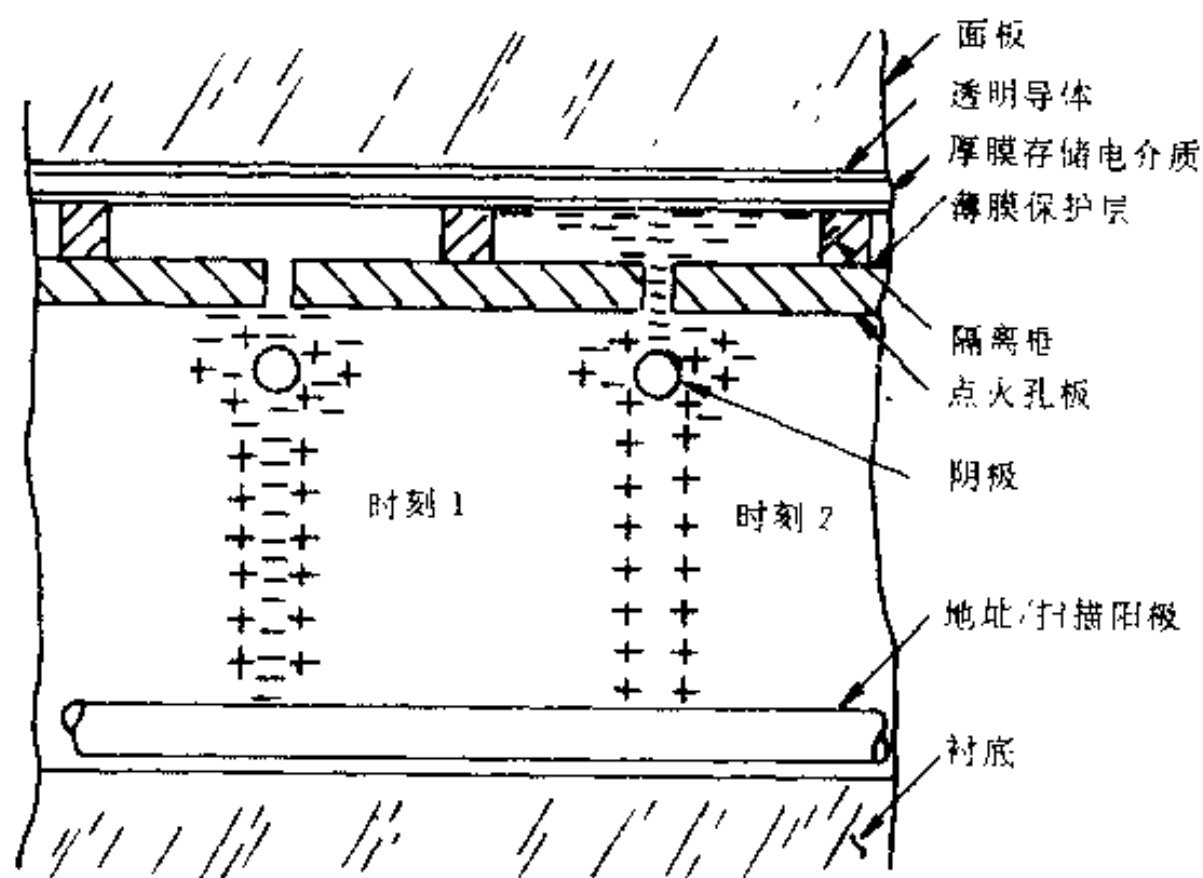


图 2-33 存储器式显示器侧视图

交流等离子体显示分辨率高,因此采用空间调制来实现灰度是很诱人的。这种所谓的空间调制技术和印刷工业中采用的网目铜板(halftone)图象技术类似。这种技术对任何具有足够分辨率但只有两级的显示都可获得灰度效果。它的中心思想是以一定小区域上黑白成分所占面积的多少来表示该区域的灰度。这种技术的主要优点是算法很简便,只需要小量硬件或软件。用它来传真图象是很实用的。由于看起来有灰度效果,而实质上仍是黑白二值的,所以也称伪灰度(pseudo gray scale)。这方面还有其他技术,诸如有

序高频颤动(ordered dither)技术等等,复杂的算法可以改进效果,不过硬件或软件的开销也大。

目前在市场上可见到的等离子体显示装置几乎都是氖气的橙色,为了获得彩色效果可以采用不同的混合气体或荧光物质。

为了获得其他颜色最早采用的办法是改变混合气体。问题是其他惰性气体混合气的发光效率远不如氖气。这就是说要获得与氖气一样的亮度,气体放电至少需要 10 倍的驱动电流或占空度,这样功耗、成本骤增而寿命骤减。

水银基的氙气混合气是一个例外,它发蓝光,效率和亮度可和氖气相比拟,但是它的效率及亮度和水银气压力紧密相关,因此特别依赖于温度。在 $20\sim 60\text{ }^{\circ}\text{C}$ 之间,它的亮度变化为五倍。

后来人们寻找高亮度、高效率的混合气体时,突破了惰性气体的范围,比如氙、卤素或氧等。这些气体已在用于高压气体激光器时深入地研究过了。 Ne-Xe-Cl_2 混合气发蓝光的效率可和 Ne-Xe 潘宁混合气相比,但氯气化学活性太强。 Xe-O_2 混合气发绿光,它有可能和等离子体显示的其他材料配合。

采用加荧光物质的方法可以获得良好的色彩效果,有些荧光物质可由紫外光激发,也可由电子激发。在电子激发时,由于电子能量很低,低于 CRT 很多数量级,因此大多数常用的荧光物质是不适用的。已知适用的荧光物质是 ZnO:Zn (氧化锌及锌)它发绿光。

直流显示时需要荧光物质具有导电性,否则它被充满电后,电子再也轰击不了。在 CRT 里通常采用复盖在荧光物质上的铝导电膜来解决,但在等离子体显示时不行,低能电子根本穿透不了那怕是最薄的导电膜。 ZnO:Zn 是导电的。

ZnO:Zn 有个有趣的特性,当氖气的橙光和它的绿光结合后形成白光,因此可以做黑白图象显示。

人们对光致发光荧光物质进行了详细研究,以便用在等离子体显示上。这些荧光物质由气体放电产生的紫外光激发。在许多

情况下,荧光系统的效率和亮度比氖气系统好得多。表 2-2 为用于彩色等离子体显示的紫外光(UV)激发的一些荧光物质。

表 2-2 UV 激发的荧光物质

荧光物质			色坐标		相对亮度
红	绿	蓝	u	v	
$Y_2O_3:Eu$	$Zn_2SiO_4:Mn$	$CaWO_4:pb$.18	.35	0.58
$Y_2O_3:Eu$	$Zn_2SiO_4:Mn$	$Y_2SiO_5:Ce$.18	.33	1.0
$YPb_{12}V_{15}O_{41}:Eu$	$Zn_2SiO_4:Mn$	$YPb_{12}V_{15}O_{41}$.18	.33	.83
$(Y,Gd)BO_3:Eu$	$BaAl_2O_{19}:Mn$	$BaMgAl_{11}O_{17}:Eu$.19	.31	1.7

2.4 场致发光显示装置

2.4.1 EL 显示原理及发展

场致发光 EL(Electro Luminescence)是指在交变电场下荧光物质发光。EL 显示是真正的固态结构,很结实。在军事及工业等要求在恶劣环境下可靠工作又便于携带的场合,EL 显示的应用越来越普遍。

典型的 EL 显示由一层薄膜状的 EL 荧光物质夹在两层薄膜绝缘介质中组成,见图 2-34。

前面的薄膜绝缘介质是透明的,后面的那层薄膜反光。含锰的硫化锌荧光物质在加有 200V 交流电时发出明亮的黄光。电压是通过很薄的栅状行列电极加上的,每个象素可单独开关,用不同荧光物质可产生不同的色彩。

EL 的开发开始于 40 年代后期,50 年代时最初的想法是用它作照明。先前荧光灯、阴极射线管这些采用类似荧光物质的器件已获成功,也鼓舞了 EL 的开发。而字母数字的显示完全可以用分段显示的方法来完成,每一段只是一个扁平 EL 灯。不过在平板电视

方面的努力由于解决不了以下三个技术问题而在 60 年代后期宣告放弃：

- 高亮度
- 长寿命
- 阵列寻址能力

技术的发展使 EL 显示重返众人关注的舞台。对 EL 显示推动作用最大的两项技术是：

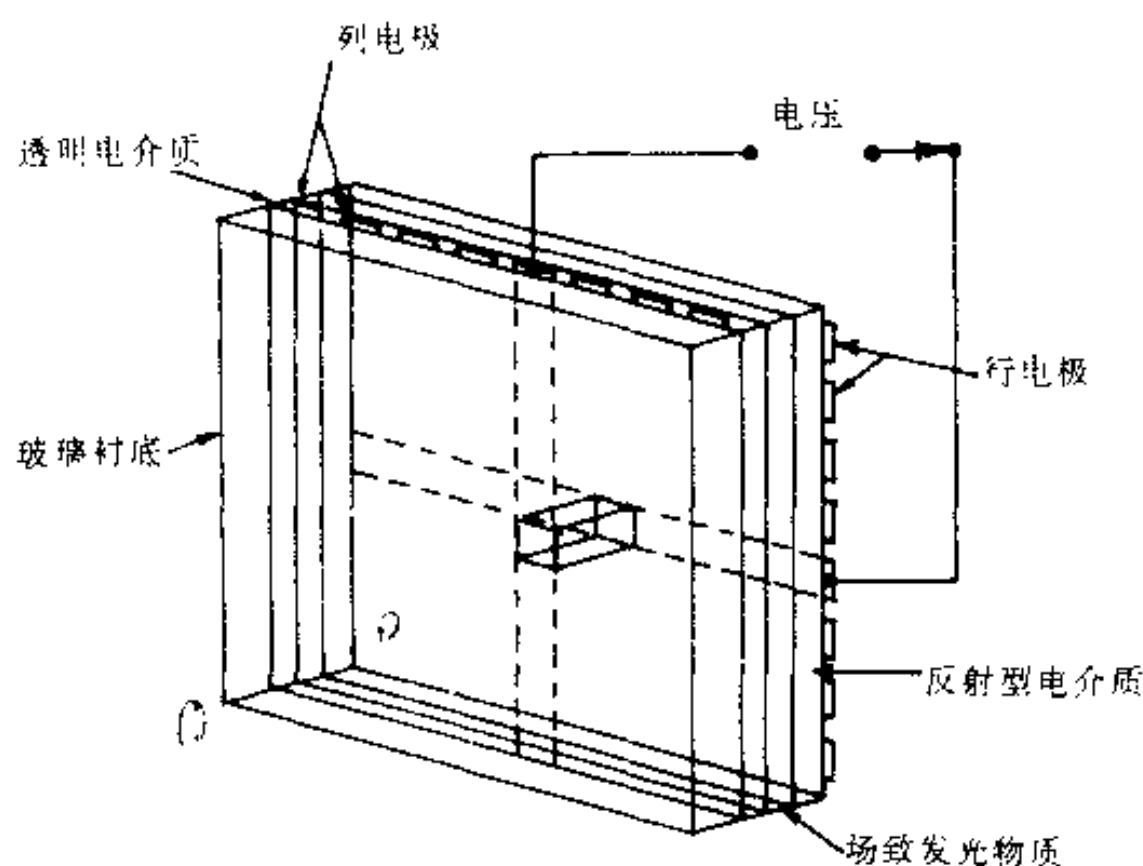


图 2-34 EL 显示原理

大规模集成双扩散金属氧化物半导体 LSI DMOS (Large Scale Integrated Double Diffusion Metal Oxide Semiconductor) 的高压驱动电路技术及阵列寻址的交流薄膜荧光夹层显示屏技术。

最初选择的荧光物质是用锰活化的高纯度硫化锌。这种荧光物质具有很强的非线性，鉴别率 (discrimination ratio) 达 10^4 ，它在高亮度下寿命很长，在阳光下依然可读。

DMOS 场效应管具有很高的电压性能，它和同一衬底上的低

压 CMOS(互补金属氧化物半导体)逻辑配合形成很高集成度的专用芯片来驱动大量电极线。

EL 显示范围尺寸一般从 $6 \times 6\text{cm}$ 到 $31 \times 36\text{cm}$ (12×14 英寸), 分辨率从 320×128 到 640×400 象素。和气体等离子体显示类似, 利用控制器芯片来获得灰度。

EL 显示比气体等离子体显示和 LCD 显示提供更好的对比度、更宽的视角。它的功耗比 LCD 大, 比等离子体显示小, 当然比 CRT 要省得多。用电池供电时, 因为要高压, 它和等离子体显示有同样的弱点。

EL 除了比 CRT 紧凑得多、重量轻外, 还可靠得多, 其平均无故障时间为 10 000h, 而 CRT 通常为 10 000h。

EL 显示器坚固、对比度好, 但价格比较昂贵, 为 CRT 的数倍。它一方面没能大量销售, 价格降不下来, 另一方面价格降不下来又不可能大量销售。但许多工作站厂商看好 EL 显示器。

EL 大多数依然为单色显示, 全彩色 EL 显示器由于很难找到理想的蓝色荧光物质而受阻。

从总体上看, EL 显示器是市场上的高档品, 彩色问题是广泛应用的另一个障碍。厂家会采用较低价的气体等离子体显示, 而 EL 用于军事目的及贵重的工作站等。

2.4.2 交流薄膜 EL

交流薄膜 EL 显示的基本象素结构如图 2-35 所示。

由少于 1 摩尔(mole)%的锰活化的硫化锌在鉴别率、发光效率、寿命和亮度方面是最好的。发光颜色是中心在 585nm 的橙黄色, 鉴别率为 10^6 , 发光效率为 6lm/W , 寿命超过 10 000h, 亮度适应在阳光下依然可读的要求。

前电极必须越透明越好, 尽少阻挡 EL 发出的光。光只从前后两电极重迭的区域发出。最好的前电极材料为 ITO(Indium-Tin Oxide, 铟-氧化锡)。这是一种过渡金属氧化物半导体, 厚度为 600nm 时透光率为 90%。铟在 10 摩尔%时最好。

实验确定的最好荧光物质是硫化钙,激活剂主要是锰,辅助激活剂铜、氯或镓浓度要低得多,只是稍微改进一些性能。比如铜可增加一点余辉,镓可降低一点工作电压等。锰的浓度对荧光物质硫化钙的影响很大。如果希望有滞后效应用于存储式平板显示,有光视效应用于光笔读写,则锰的浓度必须大于1摩尔%。较高浓度的锰会降低亮度和效率,不过这时器件有了存储能力,100%的占空度可用来重新得到较高的平均亮度。

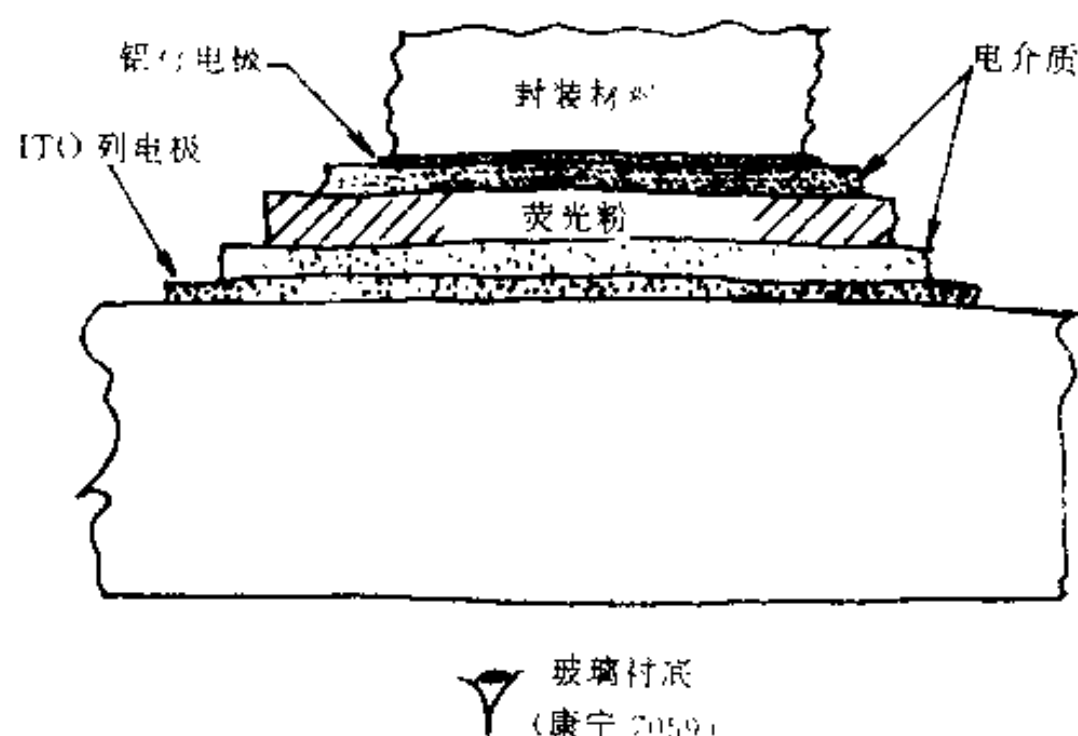


图 2 35 EL 像素基本结构

作为薄膜荧光物质外夹层的电介质,其选择原则是:首先有介电高强度,避免薄膜迭层的电击穿;还要有高介电常数,使电场的大部分集中在荧光物层上。电介质的选择影响着显示屏的效率、亮度和电压性能。

最常用的电介质有 Y_2O_3 、 Al_2O_3 、 Si_3N_4 和非晶态的 $BaTiO_3$ 。电介质层可以是不同物质的双层结构,采取这种措施可以尽量减少薄膜中的针眼。在 EL 薄膜迭层结构中最薄弱层是电介质层,完全消除其中的针眼电击穿是不可能的。Sharp(夏普)公司在它们的产

品中紧靠 ZnS:Mn 荧光层的是氮氧化硅层。一般来说,非晶态膜比晶态膜缺陷少,对混杂和化学计量也不太敏感。不过经过 550°C 荧光物退火后,电介质一般也不能保留在非晶态。在选择最好的电介质方面并不象在选择荧光物时有统一的一致意见。

显示屏可用几种方式进行封装,荧光层必须和水汽隔开。迭层结构中的任何水分子在电场影响下会水解致使薄膜分层。Sharp 公司在第二层玻璃衬底和有机嵌条封装中装入硅油电介质液体。

常用的衬底是低碱硼硅酸盐玻璃,低碱是为了尽量减少碱离子扩散进荧光物而产生掺杂。硼硅酸盐玻璃具有高温稳定性,可以适应荧光物退火温度、通常是 550°C 的考验。日本保谷公司(Hoya Electronics Co.)的 LE 30 玻璃和美国考宁公司(Corning Glass Works)的 7059 玻璃用于 EL 显示。

后电极几乎一致采用铝来完成,其原因是铝电极比其他电极经用,每当有针眼击穿它就紧贴着熔化使击穿开路。锡氧化锡排在其次,用它做后电极其熔断性能不如铝,如果铝薄到 $20\sim 50\text{nm}$,则熔断性能更好。

每个 EL 象素在每次刷新顿时发两个光脉冲,使设计者有更多灵活性来控制闪烁和亮度。在刷新时,用一种极性的阵列寻址信号对显示屏每次定址一行。在该刷新帧结束时,相当于 CRT 回扫时间开始,全屏由一个相反极性的脉冲复位,完成交流周期。在第一个脉冲时选中的象素在两个脉冲时都发光。根据电压幅度的不同,在象素开或关的转换中有一个一帧或两帧的暂态。

EL 亮度和所加交流电压峰值有关。随着电压增高,亮度随电压变化的曲线有一个拐点,该点靠近最佳发光效率和标称工作点。

显示亮度可以直接由刷新频率控制,有些显示屏也可通过脉宽调制来控制。

1972 年就有报导称,交流膜 EL 显示实验屏在峰值电压约 500V , 20kHz 的正弦激励下,亮度达 3800fL 。

用后背反射器可以提高几乎一倍亮度,但是一般不采用,因为

它把环境光也反射了回来。环境光的问题可用黑后背吸收层或前圆偏振器等方法来解决。这些方法也影响了有效的亮度,但提高了可读性。

阵列可寻址性是很重要的,阵列可寻址就是说只需要行和列驱动器。

阵列可寻址性是鉴别率(discrimination ratio)的函数。鉴别率为像素在两种情况下的发光亮度之比:一种是像素作为非阵列单元,对应所加“开”电压时所发光的亮度;另一种是同一像素处于阵列中,对应“关”时(由于阵列寻址显示此时依然有电压),像素的亮度。

不考虑环境光,我们有近似关系式:

$$\text{对比度} = (\text{鉴别率} / \text{像素行数}) + 1$$

就是说,如果对比度为 2,像素行数和鉴别率相等。这当然是不现实的,因为没有考虑环境光。因此认为像素行数为鉴别率的十分之一更可取。

由于交流薄膜 EL 的鉴别率达 1×10^6 ,在确定良好的对比度后,依然允许有较宽范围的行寻址能力。

对大部分显示技术,包括 EL 粉末技术在内,可用的鉴别率限制了像素行数,但对交流薄膜 EL 来说,起限制作用的是帧频和占空因数。对每行像素进行电激励是需要时间的,因此对一定亮度必须确保相应的最小占空因数。举例说,如果需要 50fL 的亮度,在帧频为 240Hz 时,约为 $20\mu\text{s}$ 。这时,实际上只能允许有 199 行像素,因为 $20\mu\text{s} \times 200$ (留一行时间给复位脉冲)等于 4ms,略小于一帧时间(或者说允许的最大帧频为 250Hz),可以有足够时间完成刷新。

EL 屏的物理尺寸还受透明 ITO 电极的电阻限制。屏尺寸大时,电极电阻也增加。电极线作为一根传输线,每个像素加上一些线电容,像素间的 ITO 加上一点线电阻。电极线宽窄并不重要,因为线宽了电阻下降但相应的电容按比例增加,所以 RC 时间常数保持不变。

ZnS : Mn 交流薄膜的发光特性与电气特性对温度很不敏感,从 $-29 \sim +77^{\circ}\text{C}$, 只有少量变化。

通过适当制作,采用 ZnS : Mn 荧光物的交流薄膜 EL 有亮度对电压的滞后特性,再通过适当的寻址,该滞后特性可用来赋予交流薄膜 EL 屏类似交流等离子体显示屏那样的存储能力。通过适当寻址,该滞后特性也可用来对一帧视频图象进行定格。所定格的图象可以读出送到存储设备中去。同时,滞后特性的屏是近紫外光敏的。紫外光降低了滞后特性的开门电压,可以用于光学写、抹一幅电维持的画面。

EL 屏也可由象 CRT 中的电子束来进行写操作。EL 存储模式的一个很吸引人的应用是作为存储式 CRT 的面屏。只要持续电压存在,图象就一直保留着。IBM(国际商用机器公司)和 Tektronix(泰克公司)在这方面做了大量工作。

用稀土和金属作为彩色激活剂是交流薄膜 EL 屏获得彩色效果的一种手段。荧光物质、激活剂及辅助激活剂的组合几乎是无限的。有的稀土激活剂,比如 Tb(铽),只要浓度一变,颜色就随着变。

由于薄膜 EL 是透明的,因此容易制作多色显示,把它们集成为一个整体可以消除视差。

Tektronix 的一种双色集成薄膜 EL 屏采用这种原则,它用 ZnS : TbF₃ 为绿色荧光物, ZnS : Mn 为橙黄荧光物。

三色显示可用两片集成的 EL 层共用行电极,如图 2-36 所示。

列电极宽度可以调整一下,使每一个象素在各种颜色时亮度一致。比如最宽的列电极用于蓝色。红色和绿色共享同一主体荧光物,它被交替激活为红和绿。每一列电极再进行分割来提供颜色的灰度。

2.4.3 交流粉末 EL

交流粉末 EL 由硫化锌和硒化锌制成,由超饱和的硫化铜激活,并加有其他辅助激活剂如锰、氯和银用于彩色。其中少量超量的硫化铜保留为分离状态,不溶于 ZnS 晶格,它有时也称为戴斯

特里奥型(Destriauteur)EL。今天的粉末 EL 在本质上和 50 年代时的没有什么不同,主要的改进是在粉末颗粒大小、品质控制、电介质和装配技术等方面。

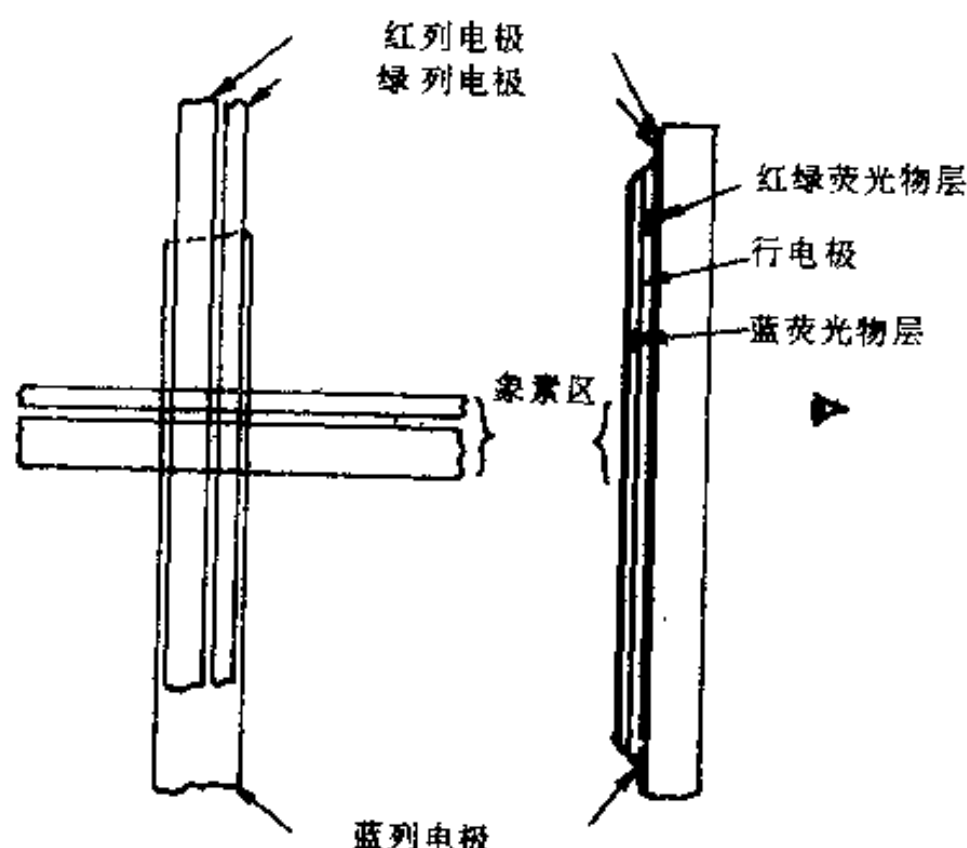


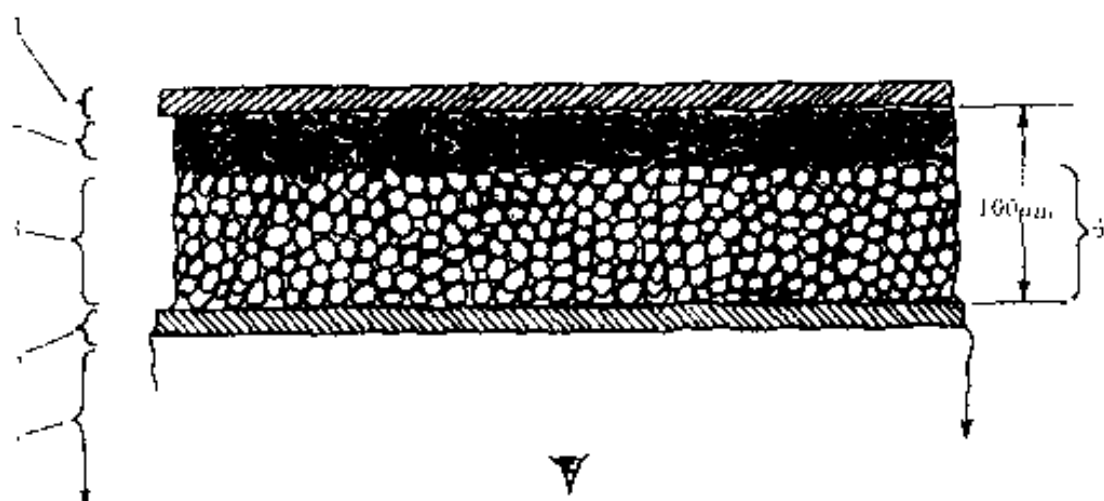
图 2-36 三色 EL 集成显示

硫化锌主体中的关键成分是硫化铜。交流粉末 EL 由许多不同主体混合物和辅助激活剂组成实现不同彩色,不过 $\text{ZnS}:\text{Cu}_2\text{S}$ 是必不可少的。硫化铜有相当好的导电性,它和 Cu_2S 都是良导体。 Cu_2S 被嵌在 ZnS 颗粒中的空位、缝隙和错位中, ZnS 是良电介质,因此所加的电场使导电的 Cu_2S 嵌入“针”大大变形,那里的局部电场强度可以很容易地比平均电场高 1000 倍。

ZnS 的高电场引起电子和空穴从 Cu_2S 针的相对端注入,而后电子激发 Zn 、 Cu 或辅助激活剂发光。空穴由 Cu 复合中心捕获,在电场反转时,发射的电子就可能和捕获的空穴再结合而发光。

为了避免器件闪弧和热损坏,粉末被嵌入电介质内,夹在电极

之中,如图 2-37 所示。



1. 铝电极 2. 有机介电层 3. 荧光层
4. ITO 电极 5. 塑料衬底 6. 发光区

图 2-37 交流粉末 EL 灯基本结构

AC 粉末 EL 器件主要是用来做照明的,用于表盘、键盘、交流等离子体显示屏和扭向列液晶显示屏照明等等。它们可以用于一切需要持续低亮度发光薄膜的地方。

自从 1980 年以来,交流粉末 EL 灯的用途日益增加,原因是:

- 已有驱动 EL 灯的低成本固态精巧电源逆变器可用,比如把 12V 直流逆变为 750Hz/70V 交流。逆变器可以直接装在灯上,用不着象原先那样仔细布置高压走线。

- 一些新产品,诸如液晶显示和薄膜开关需要照明(或背投光),以便暗处或夜间使用。

- 汽车、飞机、船只等所用的固态电子输入输出设备不论在阳光下还是在夜间都要求清晰可读。

- 设备小型化的推动。

不过交流粉末 EL 器件在信息显示方面的应用是十分有限的,因为:

- 鉴别率低

- 在中等亮度到高亮度时,寿命短

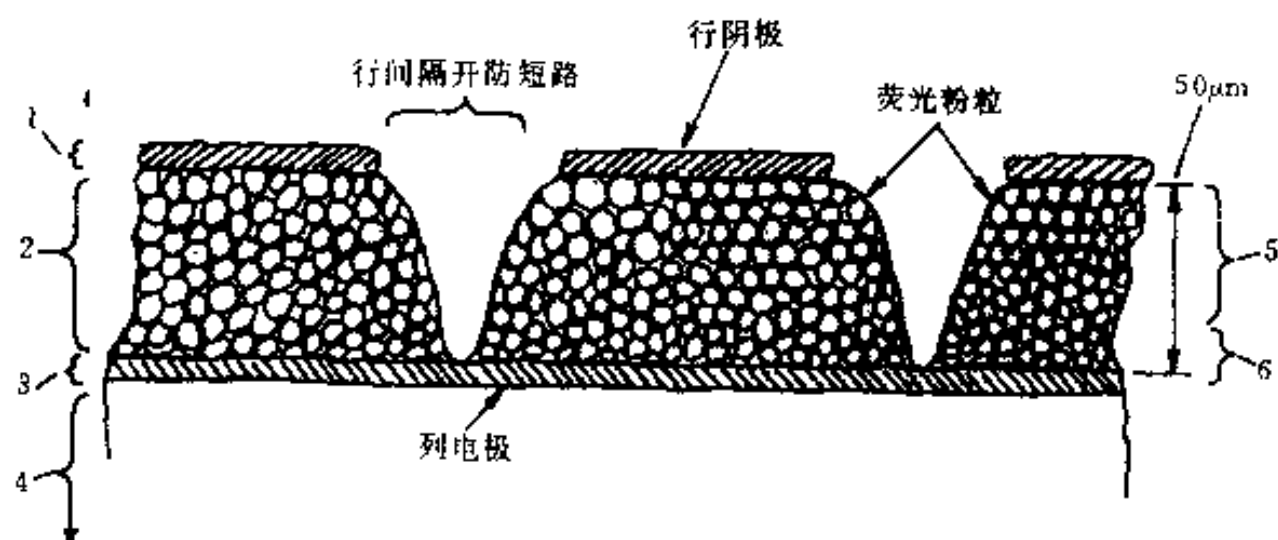
- 对比度低

低鉴别率是由它的发光机制所定的。高亮度时寿命缩短已为实验所证实,原因是较强亮度时需要较强电场,这时暴露在强电场中的铜原子发生迁移而影响了寿命。在中等到强环境光下,对比度低的问题是因荧光粉本身为白色,有很强的反射能力。采用滤光的方法可以得到一些改善。

2.4.4 直流粉末 EL

直流粉末 EL 的基础依然是荧光物质。最好的荧光物质还是硫化锌,硫化锌微粒含激活剂锰,外层由 Cu_2S 包裹,因此是导电的。用碱土激活剂代替锰可以获得很宽范围的色彩。微米大小的荧光粉粒由诸如氮纤维这样的电介质粘合起来。

直流荧光粉是导电的,为了避免行间短路,荧光粉必须做成行条形状来保证行间电绝缘。直流粉末 EL 阵列显示的基本结构如图 2-38 所示。



1. 铝电极 2. 荧光粉层 3. ITO 电极 4. 衬底玻璃
5. 涂 Cu_2S 的 ZnS 粒无光区 6. 无 Cu_2S 的发光区

图 2-38 直流粉末 EL 阵列显示基本结构

光是从紧靠阳极的一薄层荧光粉发出的。阳极是透明导体做成的,位于观察一方。阴极在后背,由铝、金或银蒸发形成。

直流粉末 EL 显示需要经过成形处理后才能发光。在成形过程中, Cu_xS 导体从阳极迁移走, 留下一薄层无 Cu_xS 的荧光粉, 光是从这一层发出的。起初, 每个荧光粉末粒是包着 Cu_xS 的, 它们可以表示为 PNP 半导体夹层结构:

阳极 | P- Cu_xS / N- ZnS:Mn / P- Cu_xS | 阴极

这里 Cu_xS 是 P 型半导体, 荧光粉是 N 型导体。电介质把这些粉末粒粘在一起处于阳极和阴极之间。当电压加在原始器件上时, 由于电阻低, 结果有大电流流过。 Cu_xS 是主要的导体, 锰激活剂也没有激发。当成形继续进行时, 因为阳极附近铜的离子移动, 致使电阻增加。在数小时成形处理并不断增加电压后, 显示器开始发光。这时显示横截面可以表示如下:

阳极 | (N- ZnS:Mn) | | (P- Cu_xS / N- ZnS:Mn / P- Cu_xS) | 阴极

即紧靠阳极的是 N 型、高电阻的薄层发光区: (N- ZnS:Mn); 而右边括号内的只是包着 Cu_xS 的 ZnS 粉末粒, 可以看成是 P 型、低电阻的延伸阴极。

如果把电压极性反转, 刚才的过程并不能逆转, 倒是在先前的阴极(现在的阳极)前形成一薄层新的发光区。这样, 总的亮度下降了, 因为所加的电压不得不分加在两层局部性的高电阻层上。

粉末 EL 迭层器件中, 直流式的比交流式的更能进行阵列寻址, 鉴别率也更高, 在低脉冲占空比时亮度也更大。比如说, 当显示屏由 120V 占空比为 0.5% 的 $3\mu\text{s}$ 脉冲激发时, 亮度为 50fL。对交流粉末 EL 来说, 亮度大约和电压的三次方成正比, 而直流粉末 EL 是六次方。这就使得在阵列寻址显示屏方面直流式的对比度要比交流式的好得多。

1973 年制成并报导的直流粉末 EL 平板电视, 对角线尺寸 34cm (13 英寸), 由 200V 脉冲驱动, 采用一次一行的阵列寻址技术, 帧频 60Hz, 不隔行。显示屏为 224 行 \times 224 列, 在低环境光下亮度为 10fL 时对比度为 10:1。由于所有粉末荧光物的散射、反射特性, 这种亮度下的对比度在通常办公室光照环境下是不实用的。

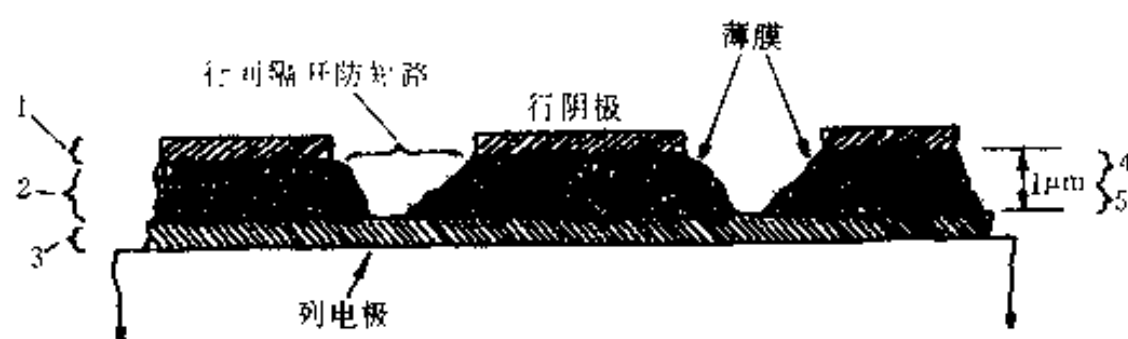
这种显示的总效率低,在半亮度时的寿命为 300h。

采用 $\text{ZnS}:\text{Mn}$ 直流粉末 EL 已经成为商品用于汽车表盘等类似场合的 80 和 256 字符读出显示。通过采用稀土荧光粉,该类器件有不同颜色。

2.4.5 直流薄膜 EL

直流薄膜 EL 是所有 EL 结构中最简单的。它也是除交流粉末 EL 外最早被研究的,可以追溯到 1959 年,而成果最少。

在高电场中的荧光薄膜容易突然打火而形成永久击穿。EL 工作点在电容清除点之上,可以通过在薄膜上敷铜的方法来减低。这样的薄膜是导电的,形成类似直流粉末 EL 那样的结构。由于直流导电必须要求像素间荧光物质互相绝缘,以便象在粉末 EL 中那样进行阵列寻址。直流薄膜 EL 基本结构如图 2-39 所示。



1. 铝电极 2. 荧光层 3. ITO 电极
4. $\text{ZnS}:\text{Cu}_x\text{S}, \text{Mn}$ 无光区 5. 无 Cu_xS 发光区

图 2-39 直流薄膜 EL 阵列显示基本结构

该图和直流粉末 EL 十分相似,明显的区别有:厚度不同,薄膜型荧光物厚 $1\mu\text{m}$ 而粉末型厚 $50\mu\text{m}$;薄膜型中 ZnS 和 Cu_xS 是混合的,而粉末型中 Cu_xS 是包裹在 ZnS 颗粒外面的。

1980 年报导的法国的一种小型阵列可寻址屏(20 行乘 20 列)由 $\text{ZnS}:\text{Mn}, \text{Cu}$ 做成,每个像素的荧光点是通过一种铝掩膜完成

的。在 1kHz 重复频率、占空比为 0.5%、45V 的电压下工作,测试所得寿命为 7300h。最初亮度是 10fL,最终亮度是 7.1fL。

直流薄膜 EL 的主要吸引力是它的工作电压相对较低,结构简单。不过,它的结构比交流薄膜 EL 可能要更复杂一些。因为,直流薄膜需要把每行或每个象素绝缘开来。而由于高压 DMOS 驱动器大规模集成电路和低电压 CMOS 逻辑电路已随手可用,所以低电压的那点好处也消失了。

2.5 显示装置展望

还有其他各种显示,比如真空荧光显示 VFD(Vacuum Fluorescent Display)、电泳显示(electrophoretic display)、发光二极管 LED(Light-Emitting Diode)显示等等。限于篇幅不在这里讨论。

值得提一下的是,白炽灯灰度方式显示在巨大尺寸时具有良好的画面效果。哈尔滨在白炽灯显示屏方面已有相当的研制历史,最初称为记分牌,用于体育场馆,还曾经援助外国。哈尔滨市计算机研究所制造的 16 级灰度白炽灯点阵显示屏在哈尔滨冰上运动基地投入使用多年,其效果之好非亲临现场是体会不到的。他们采用普通白炽灯,降压使用,画面是悦目的橙色。由于有 16 级灰度,表达能力还是十分丰富的。由于是白炽灯,亮度高,在大场面的图形图象显示中特点明显。它可以和计算机、电视摄象机、录象机等视频源连用、功能强。它的寿命长(降压使用灯泡)、价格较低,不过功耗大、单色。不管怎样,它还是无愧于占领显示技术的一席之地。

展望显示装置的未来,考虑形体扁薄、重量轻、固态结构等等优点,平板式显示屏终将取代 CRT。不过 CRT 显示尺寸大、全彩色、廉价等等优点,使它依然有广阔的市场。

谁知道什么样的技术突破会引起又一种显示技术的崛起而成为现在所有显示手段的强有力的竞争者呢?物理、化学和制造技术

的进步会使显示不断发展,将来会向三维、高亮度、全彩色发展。

无论这几年来显示技术进步多大,信息大都是以传统的二维平面形式来表达的。对大多数文本显示应用来说,这已足够了。但是,人类习惯以三维来观察事物,因此三维显示是很有意义的。

由于三维显示方面的实验又走回到用于静止照片到电影、电视的立体观视镜。双眼视觉使人们在相距不远位置上观察到两幅图、从而产生深度感。早先探索用红、蓝眼镜来对应红、蓝图象形成具有一定深度感的图象,而后又有采用偏振光一类的技术来替代。

一种用于所谓虚幻环境(virtual environment)的立体镜技术正在迅猛发展。想象一下用 CAD 软件生成的三维图形,人们不是从二维屏幕上观察,而是戴上头盔式立体视镜、进入三维空间观察模型。你可以围绕着模型转着看,也可以进入模型,从各处观察,寻找错误和不足,确认所建立的模型是预想的,并可实时解决问题。

三维虚幻环境正在成为一种非常诱人的界面。建筑师在规划阶段游览他们设计的建筑物,可以很容易发现问题,以很小的代价来解决。医生可以实习精巧的外科手术而无需在病人身上冒险……其中的技术基础之一,自然是立体显示。

在现实世界中,人们之所以有深度感、距离感是因为用双眼观察,每只眼睛所看到的情况有所不同。为了把这种功能转移到虚幻环境,感观差别必须保持下来。因此需要两个屏幕,每屏显示物体稍有不同的两幅视图。每只眼睛只能看相应的屏幕。头盔式立体视镜就是这样的一种显示装置。

这方面研究的先驱之一是美国国家航空航天局(NASA)的 Ames 研究中心。想法是为空间程序提供一种多媒体或多感觉(multisensory)的、交互的三维接口。Ames 的科学家已开发了 VIEW (Virtual Interface Environment Workstation),它是一种广角、头盔式立体显示系统,由操作者用其声音、体位和手势来控制。

最初的头镜装在摩托车头盔中,对应双眼的是两片 100×100 象素的 LCD。比后来的头镜轻多了,远不是那样幽闭恐怖。除了三

维显示之外,还加入了头跟踪装置,一个微音器用于连续讲话识别,一副耳机用于三维声音提示。

进入虚幻世界的感觉由高分辨率(1000×1000 象素)广角图象所强化。图象在你周围呈现,头跟踪技术(6 个自由度)使你只要转动头部就可改变视点。图象以实时速度更新(高至每秒 30 帧),两个显示器紧密同步给眼睛提供协调的图象,眼罩用来避免环境光的干扰。

超广角视野的光学性能是显示的关键。要感到进入虚幻世界,需要视场接近人类的双眼视野,只采用一个小窗口来进入是不够的。显示必须全部进入视野,给人们一种处身于虚幻环境中的真实感觉。NASA 采用广角放大镜,自然还需解决透镜畸变(用生成图形时的反向畸变来校正)。

头镜方案有一些缺点。其中之一是人们一戴上头镜,就被固定在那种环境中,如果每次需要摘掉头镜来看看虚幻世界之外的某物,在不同任务间切换会变得很复杂。因此需要有一种分离式的立体观察器来补充,并在某些场合不取代头镜。

平衡式 CRT 立体观察装置 CCSV (Counter balanced CRT-based Stereoscopic Viewer) 是一种虚幻环境显示装置,它比头镜更容易和台式环境配合。

虽然 CCSV 不是头戴式的,但它和头的运动耦合,即随头运动。装置安放在平衡的动态连杆机构上,允许在固定点 89cm (35 英寸) 范围内无约束地运动,装置包含具有广角光学效果的双 CRT 立体观察装置。

CCSV 提供广阔的视场, 360° 自由度,强烈的三维幻觉,人们可以方便地进入和离开虚幻世界。

CCSV 的光学结构引入了一种色差,比在 LCD 观察装置中的还明显,在图象边缘处尤其明显。也有和头镜中类似的枕形失真。

CCSV 的另一个问题是,在一组人员共同工作时,传递装置的同时,视点也改变了。因此需要一种“视点冻结”钮,在传递装置给

别人前,先锁定,这样人家很容易看到你所注视的三维图象。

三维显示也已开始采用激光全息技术,它所产生的影响是深远的。

除此之外,对传统技术的改进也出现了一些新的气象。以 Private Eye(雅眼)为例,这是一种采用反射技术的计算机监视器,可以象台式 CRT 显示一样显示字符和图形,但非常小巧,只重 64 克,可以很容易地放在手掌上。

雅眼采用 280 个 LED(发光二极管),1 片可调透镜,1 片振动镜。它利用视觉暂留现象,在 6cm^2 左右的视窗中显示 720×280 像素。它用 9×11 (而不是通常的 8×8)像素阵表达 CGA 正文,并显示标准的 640×200 像素 CGA 图形。

常规的 CRT 显示一次“刷涂”一个像素,自左上至右下地显示。而雅眼的工作方式为每次一整列。当它接在微型计算机上时,它从专用的显示适配器卡中取得每帧显示并存入缓冲器中。然后用该显示帧的第一列的数据激发其 280 元 LED,由一反射镜把该输出射往一片透镜,在观察者眼上聚焦。

下一步,显示帧的第二列激发 LED,镜子稍变角度把第二列反射到第一列的右边。如此反复做 720 列算一次,每秒 50 次。由于视觉暂留,人们以为同时看到了所有 720 列,即整个 720×280 显示。

由于雅眼采用透镜而不是屏幕,人们可以把它聚焦在眼前或是远处。由于它是按一只眼睛设计的,为了避免眼睛疲劳,双眼可以先汇聚在同一物体上,然后对雅眼调焦,使它看来和那物体一样远,这时的输出就被迭加在背景上。

由于人们并不经常改变显示与背景之间的聚焦,雅眼可让人们看重要数据的同时做其他任务。如果把它加在头镜上,可以空出双手。比如说,外科医生只要向上瞥一眼就可得病人的信息,而不会分散注意。

雅眼这种很有意思的显示革新技术可使计算机在更广的领域

中使用,也增强了目前没有全尺寸显示的产品用途,比如说手持传真机等

第三章 图形控制器

一个完整的图形显示系统应包括:图形显示装置及其对应的图形控制器、图形显示驱动程序和实用程序、可供使用、维护和开发的技术文件。

目前,计算机图形显示主要以 CRT 为显示装置。CRT 监视器和在计算机内的图形卡组成图形显示的硬件系统。在微型计算机中这种结构是普遍采用的,在工作站上情况也很类似,而在小型机等主机系统中则采用图形终端的形式。

图形控制器和 CRT 监视器一起迅速发展,同时适应了日益提高的对计算机图形的要求。在微型计算机的发展过程中可以清晰地看出这种历程。

IBM 微机及其兼容机的图形子系统的结构最初受 IBM PC 的制约。它有两大显示类型,一种是字符显示,另一种是图形显示。字符显示的特点是:事先在图形卡上做好称作字符发生器的字符集点阵 ROM(只读存储器),由图形卡直接调用显示,当然显示的可控最小单元是一个字符。图形显示的特点是:显示屏上的像素和图形卡上的显示缓冲器的内容相对应,显示的可控最小单元是一个像素。图形卡有显示缓冲器,与此对应的是主机中的视频内存。在分辨率较低、色彩较少时视频内存可容纳整个一屏的图象内容。在分辨率增高、色彩增加的情况下,图形卡上的显示缓冲器随之增加,而视频内存的大小是固定的,所以视频内存已容纳不下整个一屏的图象内容,需要分批把图象内容送入图形卡的显示缓冲器中去,由此派生出了不同图形标准和兼容或变形模式。

3.1 单色显示和 CGA

3.1.1 单色显示

IBM 公司在 1981 年推出 PC, 可安装单色显示 MDA (只显示字符) 和 CGA (很有限的彩色图形显示)。1982 年, Hercules (赫尔克里士, 大力神) 计算机技术公司推出与 MDA 兼容的字符显示, 实际上, 显示单色图形的 HGC (Hercules Graphics Card) 卡, 取代了 MDA, 成为单色显示的标准。

这种显示在文本模式时, 整屏显示 80 列 \times 25 行, 共计 2000 个字符。屏上一个字符, 显示卡上对应有两个字节缓冲区, 一个字节为字符的 ASCII 码, 另一个字节存放字符属性。因此, MDA 显示卡上有 4KB 缓冲器, 对应主机视频内存地址为 B0000H~B0FFFH。

图形模式时, 显示两页图形, 每页 32KB, 共 64KB。也就是说 HGC 显示卡上有 64KB 显示缓冲区 (MDA 只有 4K, 所以它只能显示字符), 对应视频内存地址为 B0000H~BFFFFH。有关 MDA、HGC、CGA、EGA 对应的视频内存地址图参见图 3-1。

HGC 在图形显示时占视频内存地址为 B0000H~BFFFFH, 文本显示时占用 B0000H~B0FFFH, 因此图形显示时的第 0 页复盖了文本显示区, 或者说这两种显示共用了 4KB 视频内存, 所以两种模式互相转换时, 首先要清除这 4KB 中的内容, 否则会使显示变乱。

3.1.2 字符发生器

早先的单色显示控制器与并行打印机控制器合做在一块印制板上, 单色显示的主要性能指标为:

- 每屏显示 80 列 \times 25 行字符
- 每个字符块大小为 9 \times 14 点阵
- 字符块中的字符由 7 \times 9 点阵组成

- 能显示 8 位 256 种不同编码的字符输出

- 每个输出字符有各自显示属性,如加亮、闪烁、负象(白底黑字)、下横线等。

字符发生器是一个容量为 8KB 的只读存储器,其中含有三套不同字体的字符集(各为 256 种不同的字符)点阵。256 种字符可分为如下几类:

- 16 个专用游戏符号

- 15 个用于文字处理编辑操作的符号

- 96 个常用 ASCII 字符

- 48 个外语字符

- 48 个商用图形符号

- 16 个常用希腊字母

- 15 个常用科学符号

图 3-2 为 7×9、5×7、7×7 三种字体点阵形式及在 ROM 中的地址安排。字体选择是通过线路板上跨接线来完成的。

以字符块 9×14 点阵、80 列×25 行计算,屏幕的分辨率为 720×348 象素。HGC 能以 720×348 象素和 640×400 象素分辨率单色显示图形。

从 MDA 显示控制器和 CRT 监视器来看,控制器通过 9 芯 D 型插座与监视器相连,其中 1、2 脚接地,3、4、5 脚未用,6 脚为亮

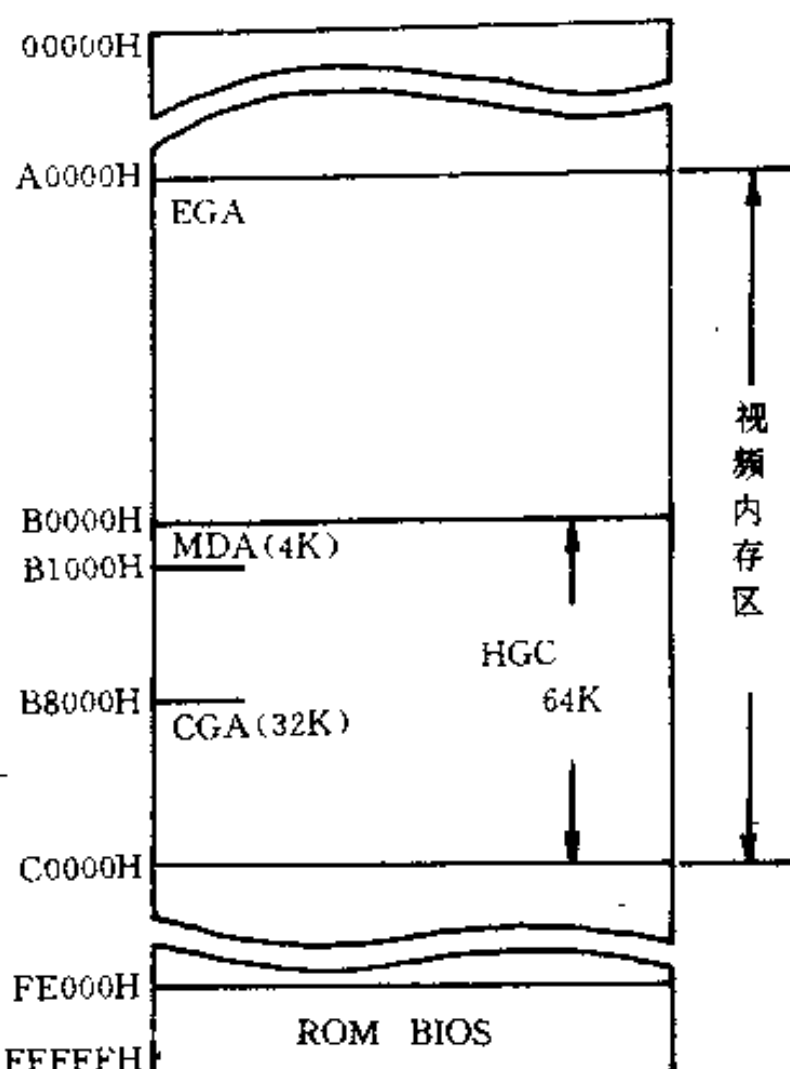


图 3-1 视频内存地址图

度,7 脚为视频信号,8 脚为水平同步,9 脚为垂直同步。

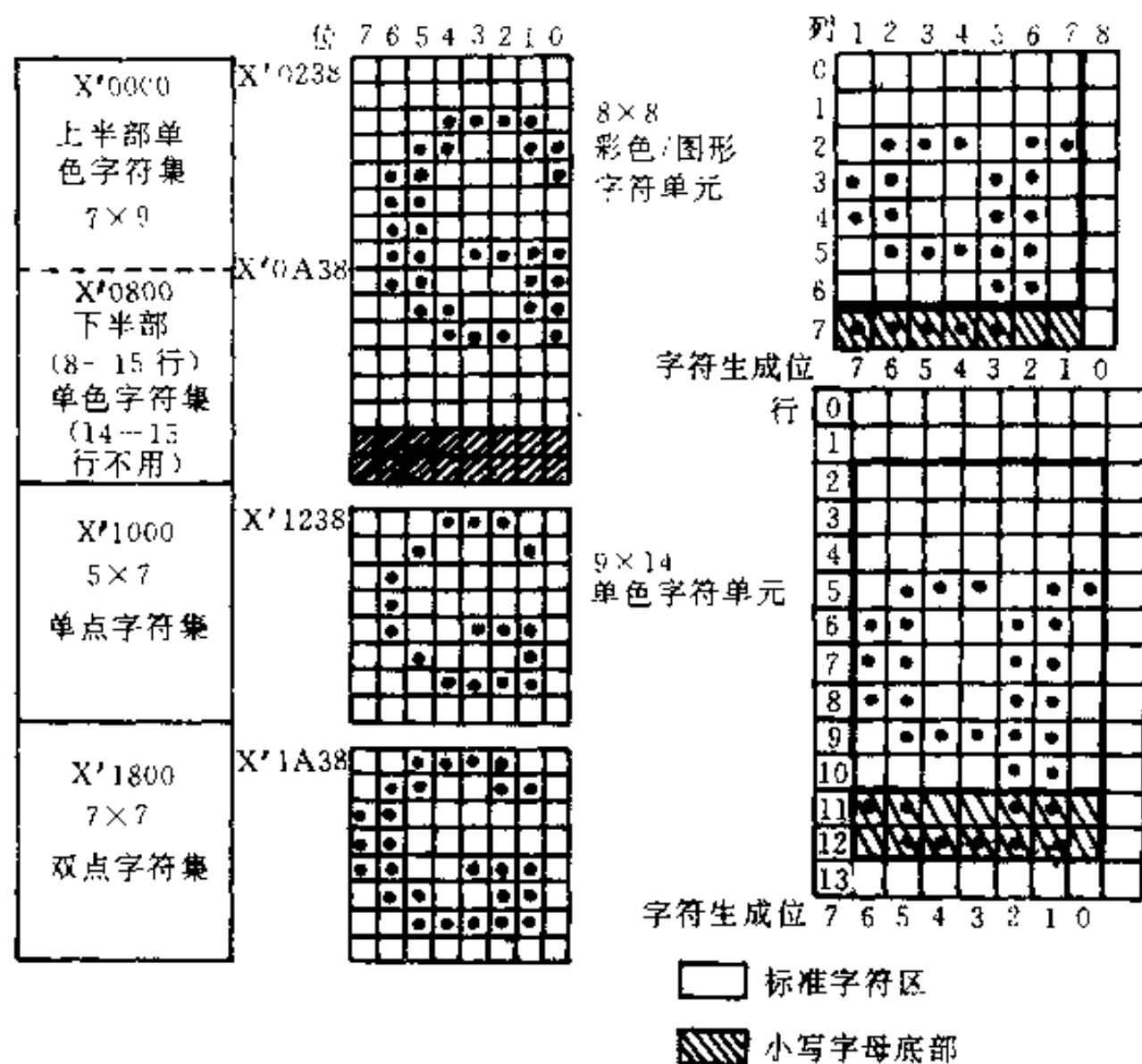
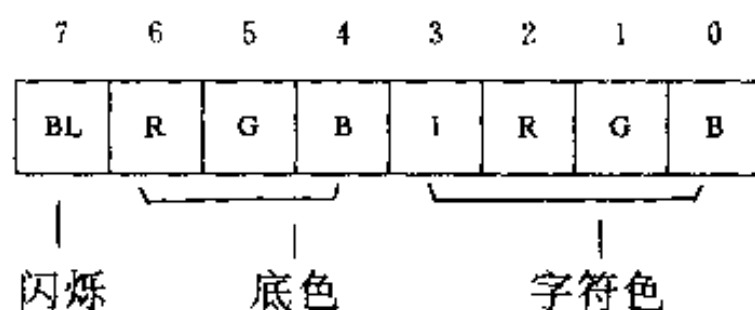


图 3-2 字符发生器字符点阵及地址安排

视频输出带宽 16.27MHz,行频 18.432kHz,帧频 50Hz,每帧 350 线,每线 720 点。

3.1.3 显示控制

MDA 单色字符显示比较简单,它显示 80 列×25 行字符,从视频内存 B0000H 开始偶数字节为所显字符的 ASCII 码,奇数字节为其属性。举例说,B0000H 单元放入 41H(A 的 ASCII 码),B0001H 放入 4FH,则屏幕左上角出现一个闪烁的、加亮的、黑底白字的字母 A。属性字节的格式定义如下:



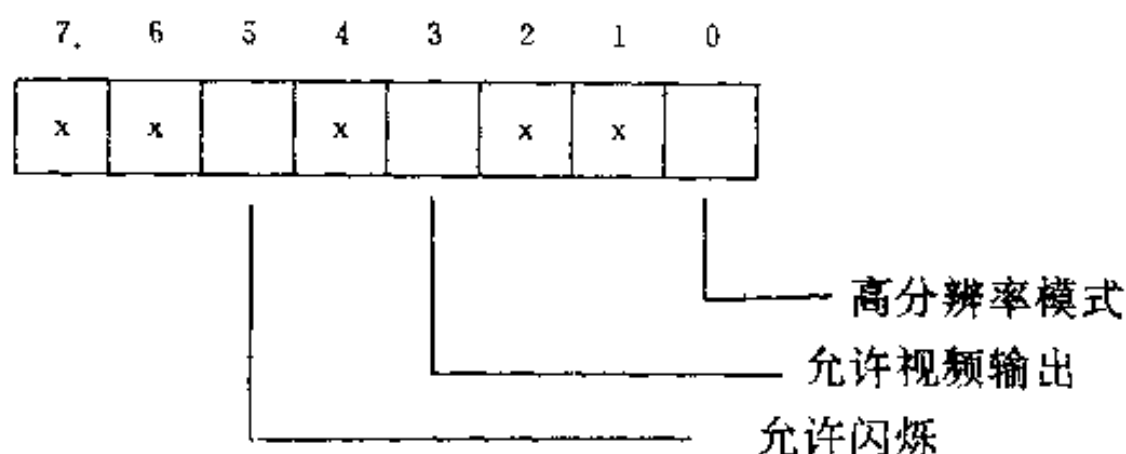
字符显示色和底色在单色显示中的组合有：

底 色 (RGB)	字符色 (RGB)	效 果
000	000	不显示
000	001	加下横杠
000	111	黑底白字
111	000	白底黑字(负象)

如此推理, B0FFEh 是第 2000 个字符; 即屏幕右下角字符的 ASCII 码, 而 B0FFh 是它的属性。

在系统的低层编程时, 显示控制器的工作模式和参数是 CPU 通过输出指令来控制 and 设置的。系统分配给单色显示控制器的输入输出端口地址有四个:

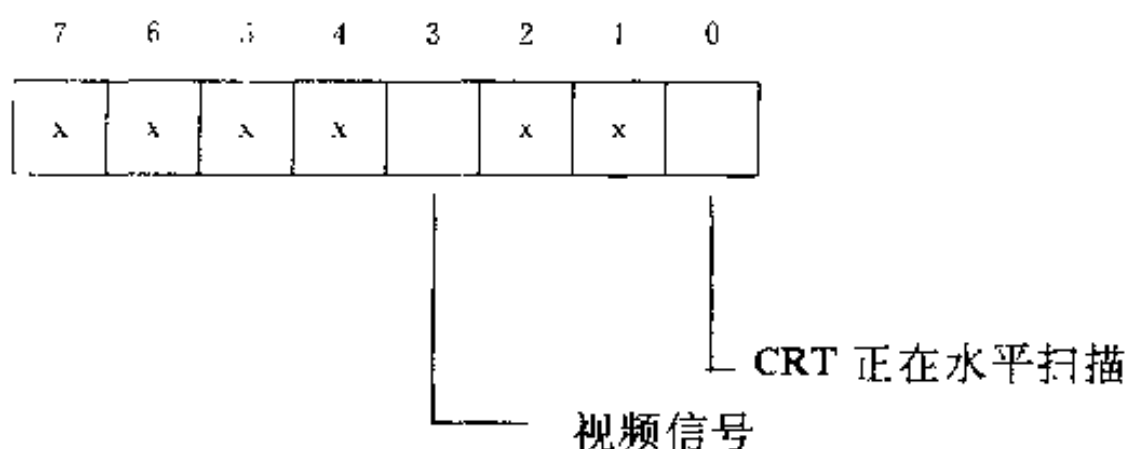
CRT 控制寄存器(I/O 地址 3B8H)



其中 X 为不使用的位。系统加电后向单色显示控制器发出的

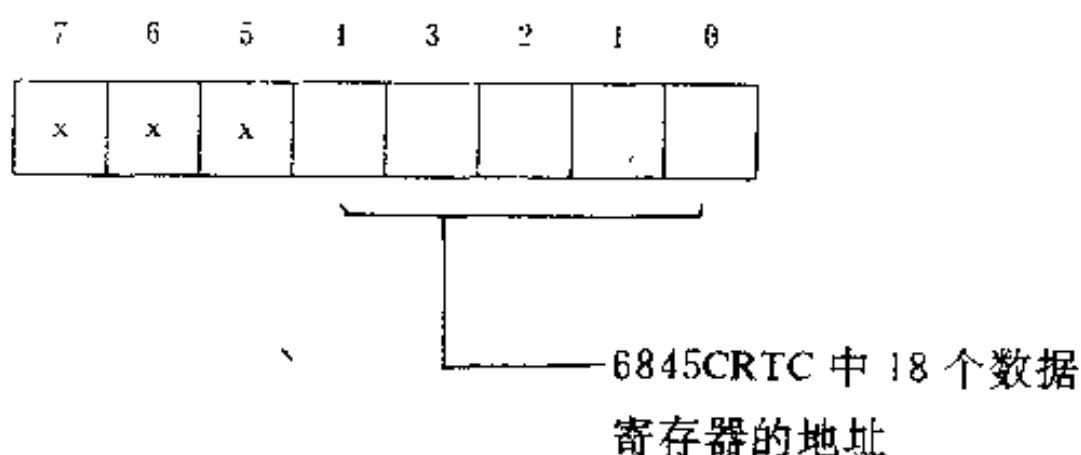
第 1 条命令必需置高分辨率模式位为 1, 否则 CPU 将一直处于等待状态。

• CRT 状态寄存器(I/O 地址 3BAH)



每当 CPU 需要输出显示一个字符时, 必须先读 CRT 状态, 判断它是否处于回扫(此时状态字节最低位为 0)。如果是 0, 则可向缓冲器输出。

• 6845 CRTC 索引寄存器(I/O 地址 3B4H)



显示控制逻辑是围绕着 Motorola 6845 CRT 控制器芯片设计的。它内部有 19 个寄存器, 为了少占用 I/O 端口地址, 用索引寄存器来指明其他 18 个数据寄存器的某一个。

• 6845 CRTC 数据寄存器(I/O 地址 3B5H)单色显示初始化参数情况如下:

它们的物理意义如图 3-3 所示。

R_0 、 R_1 、 R_2 和 R_3 控制水平同步。 R_0 为一次水平扫描的总时间

(包括回扫在内), R_1 为每行的字符数, R_2 决定水平同步信号在水平扫描线上的位置、调节显示画面在屏幕上的左右位置, R_3 用作水平扫描宽度的补偿。

表 3-1 单色显示 6845 初始化参数

寄存器序号	寄存器名称	参数单位	80×25 单色模式参数
R_0	水平扫描总时间	字符	61H
R_1	每行字符数	字符	50H
R_2	水平同步位置	字符	52H
R_3	水平同步宽度	字符	0FH
R_4	垂直扫描总时间	字符行	19H
R_5	垂直总调节	扫描行	06H
R_6	每帧显示行数	字符行	19H
R_7	垂直同步位置	字符行	19H
R_8	隔行扫描模式	——	02H
R_9	最大扫描线地址	扫描线	0DH
R_{10}	光标开始	扫描线	0BH
R_{11}	光标结束	扫描线	0CH
R_{12}	起始地址(高位)	——	00H
R_{13}	起始地址(低位)	——	00H
R_{14}	光标(高位)	——	00H
R_{15}	光标(低位)	——	00H
R_{16}	未使用	——	——
R_{17}	未使用	——	——

R_4 、 R_5 、 R_6 、 R_7 、 R_8 和 R_9 控制垂直同步。 R_4 和 R_5 决定一次垂直扫描的时间, R_5 起调节作用,使垂直扫描时间与额定的帧频相适应。 R_6 是每帧显示的字符行数, R_7 控制垂直同步信号在垂直扫描线上的位置, R_8 用于选择隔行扫描(当 $R_8 = 02$ 时,表示逐行扫描), R_9 为每个字符行中扫描线数减 1。

R_{10} 和 R_{11} 控制光标显示、闪烁、闪烁周期、光标形状和大小等。 R_{14} 和 R_{15} 是只读寄存器、共 14 位,存储光标现行位置。

R_{12} 和 R_{13} 是只写寄存器,用来决定每次垂直扫描开始时,刷新存储器的首地址。

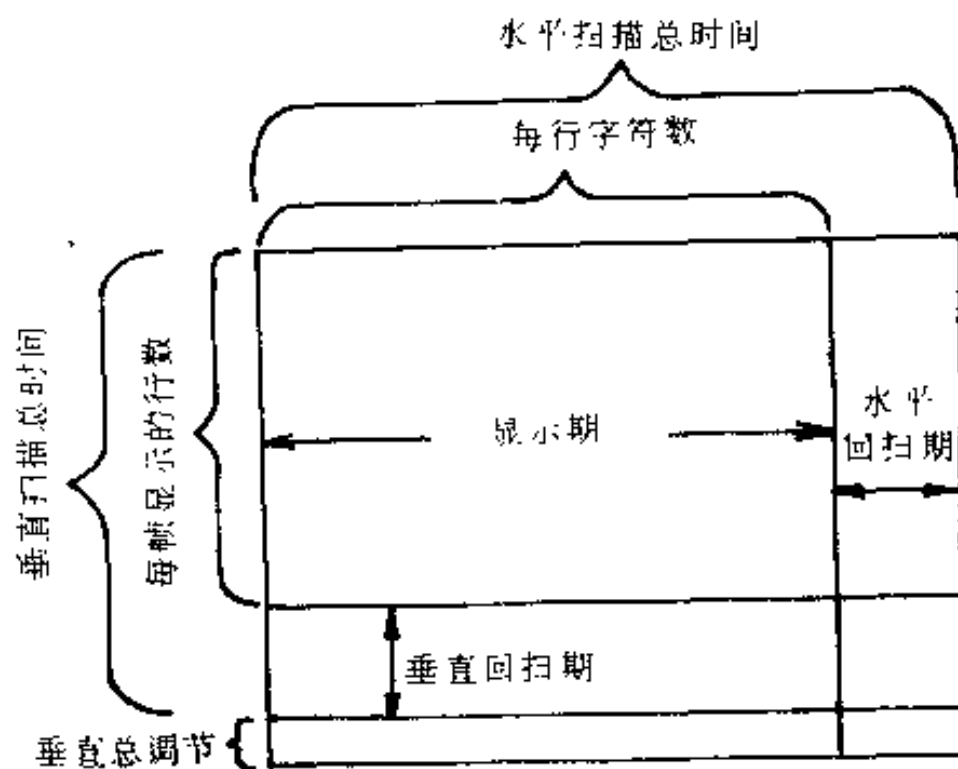


图 3-3 6845 扫描参数物理意义

3.1.4 CGA

CGA 是 IBM PC 最早的彩色图形卡,有两种基本工作模式:

- 字符模式,一屏可显示 40×25 个字符或 80×25 个字符。每个字符块大小为 8×8 象素点阵,其中字符为 5×7 或 7×7 点阵构成。与单色显示一样,可以显示和处理 8 位 256 种字符代码,每个字符带一属性字节。选择黑/白显示模式的属性和单色显示时相同,选择彩色显示模式时,每个字符均有 16 种底色和 16 种显示色,且仍有闪烁属性。

- 图形显示模式,这时显示器屏幕上的每个象素均可由程序控制其亮度或颜色。该模式可取高、中、低三种不同分辨率: 640×200 象素、黑白两色, 320×200 象素、四色, 160×100 象素、16 色。BIOS 驱动程序不支持最低分辨率显示模式,需要时可自行开发。

CGA 控制逻辑结构和 MDA 类似,也采用 6845 CRTC 芯片为核心。图形卡上有 16KB RAM 作为显示缓冲器,对应主机视频内存地址 B8000H~BBFFFH。在图形显示模式时,缓冲器中每 1、2 或

4 个二进位, 分别与高、中和低分辨率的各像素对应。图 3-4 表示屏幕像素位置和显示缓冲器中信息的对应关系(低分辨率时仅用缓冲器的前一半)。图 3-5 为高、中分辨率模式时, 缓冲器中各字节信息内容与屏幕像素位置的对应关系。

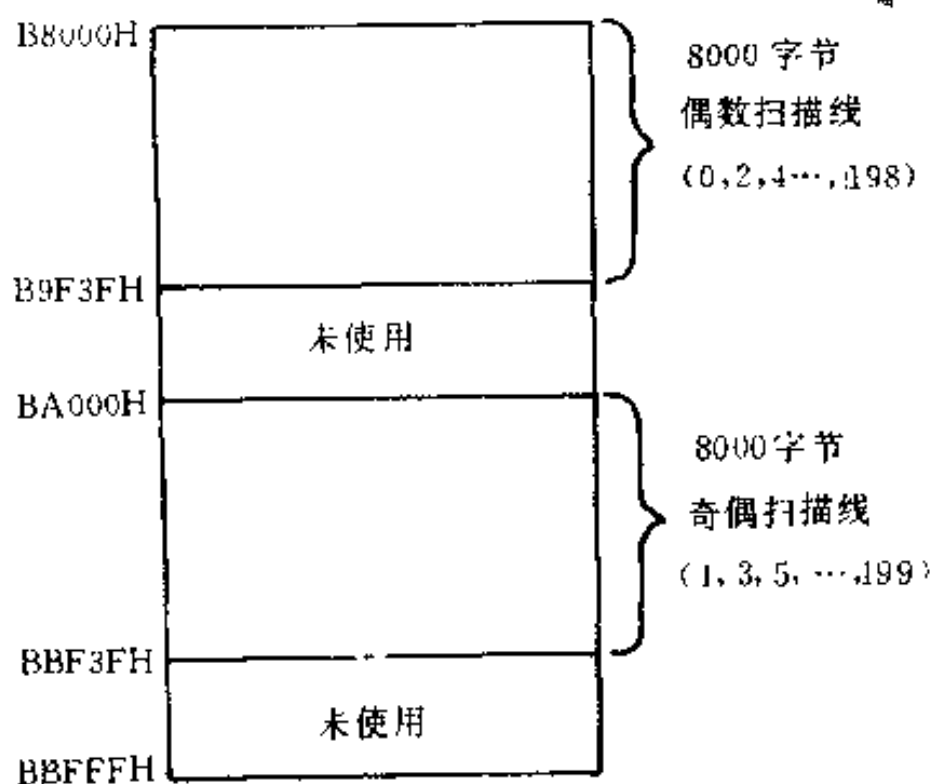


图 3-4 显示缓冲器与屏幕扫描线的关系

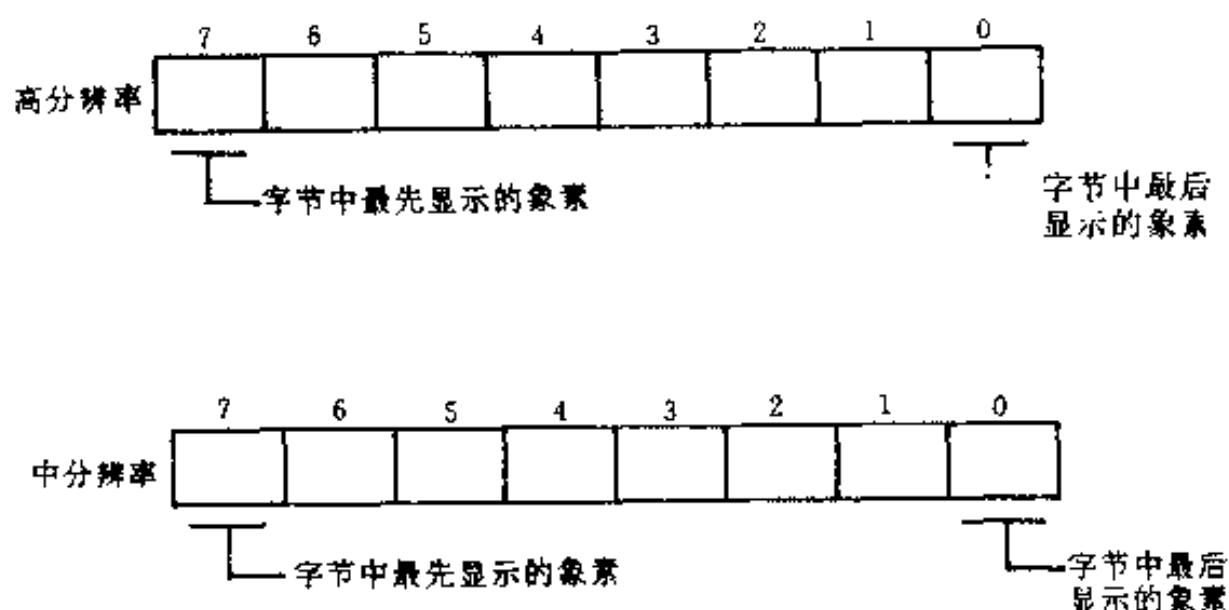


图 3-5 缓冲器字节内容与像素位置的关系

在字符显示模式时, 每个显示字符附有一个属性字节(与单色

显示同,前者为偶数地址,后者为奇数地址),每屏显示 40×25 个字符时,16KB 容量的缓冲器可容纳 8 屏(页)显示信息,页号 0~7;每屏显示 80×25 时,可容纳 4 页,页号 0~3。

CGA 有和 MDA 相同的标准 8KB 字符发生器(MK 36000 ROM),其中有三套不同字体字符,分别为 7×9 、 7×7 和 5×7 点阵。对 CGA 来说,由于受分辨率限制,选择后两种字体,一般使用 7×7 点阵字体。

CGA 有三种接口方式与显示器连接。一种是 D 型 9 芯插头座连结:1、2 脚为地,3、4 和 5 脚分别为红、绿、蓝(RGB)色彩信号,6 脚为亮度,7 脚备用,8、9 脚分别为水平同步和垂直同步信号。这种连接称“直接驱动”。

当使用低分辨率监视器时,可用同轴电缆连接,称“复合信号”方式。

如果采用家用电视机作为显示器时,在复合输出信号之后再加一个射频调制器,把调制后的输出信号连接到电视机的天线端口上。

对 CGA 的控制是通过输入输出指令来进行的。CGA 占用 16 个 I/O 端口地址(3D0H~3DFH),其中具体定义的有:

3D0H 6845 CRTC 索引寄存器

3D1H 6845 CRTC 数据寄存器

3D8H 操作模式控制寄存器

3D9H 彩色选择寄存器

3DAH 状态寄存器

3DBH 清除光笔锁存器

3DCH 预置光笔锁存器

对 CGA 编程设计的一般步骤如下:

- 选定操作模式

- 把模式控制寄存器(3D8H)中第 3 位置 0,暂不允许视频信号输出

- 把所选模式对应的一组参数置入 6845 CRTC 寄存器中
- 按选定的操作模式设置模式控制寄存器
- 设置彩色选择寄存器
- 向显示缓冲器写入需要显示的字符代码或图形信息。

I/O 端口地址 3D8H 的模式选择寄存器的各位定义如图 3-6 所示。

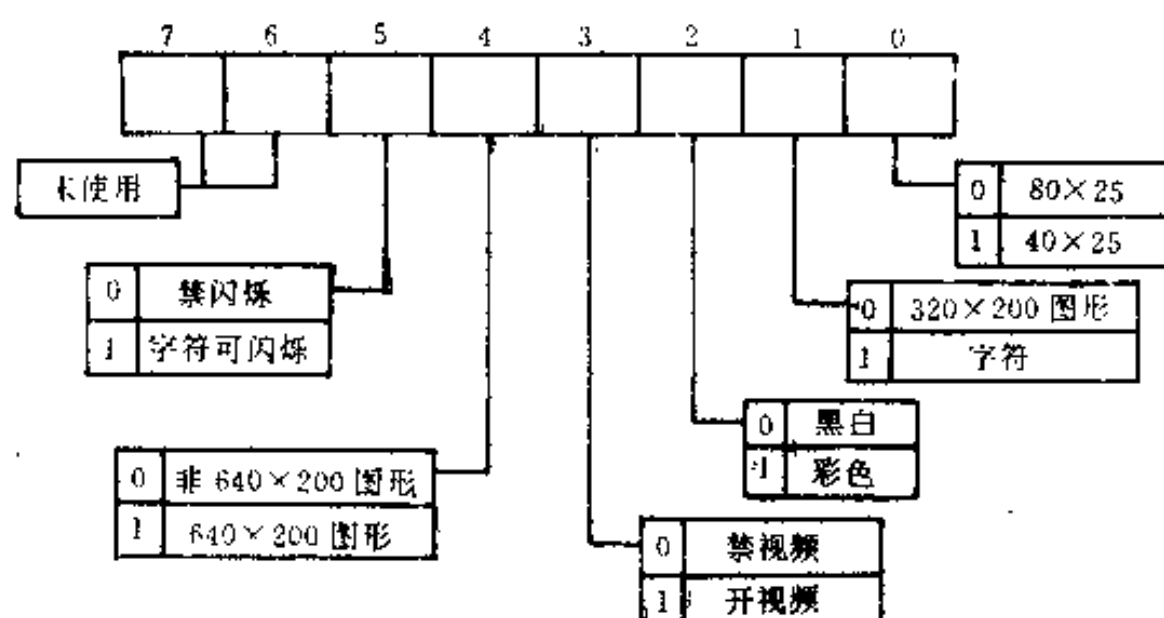


图 3-6 模式选择寄存器定义

由此可见该寄存器的各位组合并不是任意的，常用的见表 3-2。

不同的操作模式要求 6845 CRTC 产生不同的信号，因此送入参数也不同，表 3-3 列出了 6845 CRTC 在三种常用情况下各寄存器应赋予的控制参数。

6845 CRTC 中的每一个参数的设置需要使用两条 OUT 指令。首先是向 3D0(或 3D4)端口送出寄存器索引值(序号)，然后再向 3D1(或 3D5)送出参数值。

表 3-2 CGA 常用操作模式

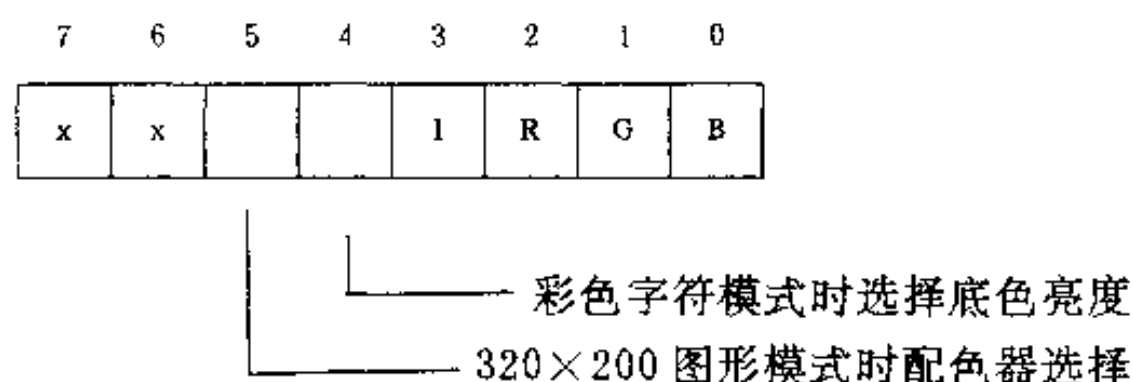
模式选择寄存器								操作模式	
7	6	5	4	3	2	1	0		
X	X	1	0	1	1	0	0	10×25 黑白字符	
X	X	1	0	1	0	0	0	10×25 彩色字符	
X	X	1	0	1	1	0	1	80×25 黑白字符	
X	X	1	0	1	0	0	1	80×25 彩色字符	
X	X	X	0	1	1	1	0	320×200 黑白字符	
X	X	X	0	1	0	1	0	320×200 彩色图形	
X	X	X	1	1	1	1	0	640×200 黑白图形	

表 3-3 6845 寄存器参数表

序号	寄存器号	寄存器名称	I/Q 特性	参数单位	10×25 字符	80×25 字符	图形
0	R ₀	水平扫描总时间	只写	字符	38	71	38
1	R ₁	每行字符数	只写	字符	28	50	28
2	R ₂	水平同步位置	只写	字符	2D	5A	2D
3	R ₃	水平同步宽度	只写	字符	0A	0A	0A
4	R ₄	垂直扫描总时间	只写	字符行	1E	1E	7F
5	R ₅	垂直总调节	只写	扫描线	06	06	06
6	R ₆	每帧显示行数	只写	字符行	19	19	64
7	R ₇	垂直同步位置	只写	字符行	1C	1C	70
8	R ₈	隔行扫描模式	只写	—	02	02	02
9	R ₉	最大扫描线地址	只写	扫描线	07	07	01
A	R ₁₀	光标开始	只写	扫描线	06	06	06
B	R ₁₁	光标结束	只写	扫描线	07	07	07
C	R ₁₂	起始地址(高位)	只写	—	00	00	00
D	R ₁₃	起始地址(低位)	只写	—	00	00	00
E	R ₁₄	光标(高位)	读写	—	xx	xx	xx
F	R ₁₅	光标(低位)	读写	—	xx	xx	xx
10	R ₁₆	光笔(高位)	只读	—	xx	xx	xx
11	R ₁₇	光笔(低位)	只读	—	xx	xx	xx

在黑白字符显示模式时,无论 40×25 还是 80×25 字符,除 16KB 显示缓冲器中可以分别容纳八页或四页外,程序设计和单色显示 MDA 兼容。

I/O 端口地址 3DAH 的彩色选择寄存器的各位定义如下:



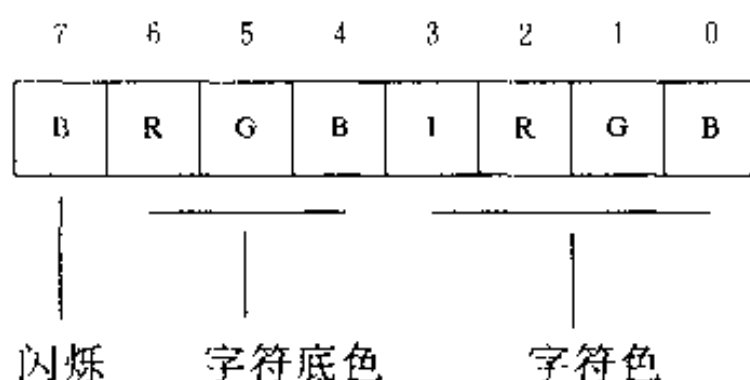
第 0、1、2 和 3 分别代表蓝、绿、红色和加亮,在 40×25 彩色字符模式下代表屏幕上矩形显示区外部边界的颜色;在 320×200 彩色图形显示模式下控制屏幕的底色。I、R、G、B 可以组合成 16 种颜色。见表 3-4。

表 3-4 彩色编码表

I	R	G	B	彩色	I	R	G	B	彩色
0	0	0	0	黑	1	0	0	0	深灰
0	0	0	1	蓝	1	0	0	1	浅蓝
0	0	1	0	绿	1	0	1	0	浅绿
0	0	1	1	青	1	0	1	1	浅青
0	1	0	0	红	1	1	0	0	浅红
0	1	0	1	品红	1	1	0	1	浅品红
0	1	1	0	棕	1	1	1	0	黄
0	1	1	1	淡灰	1	1	1	1	白

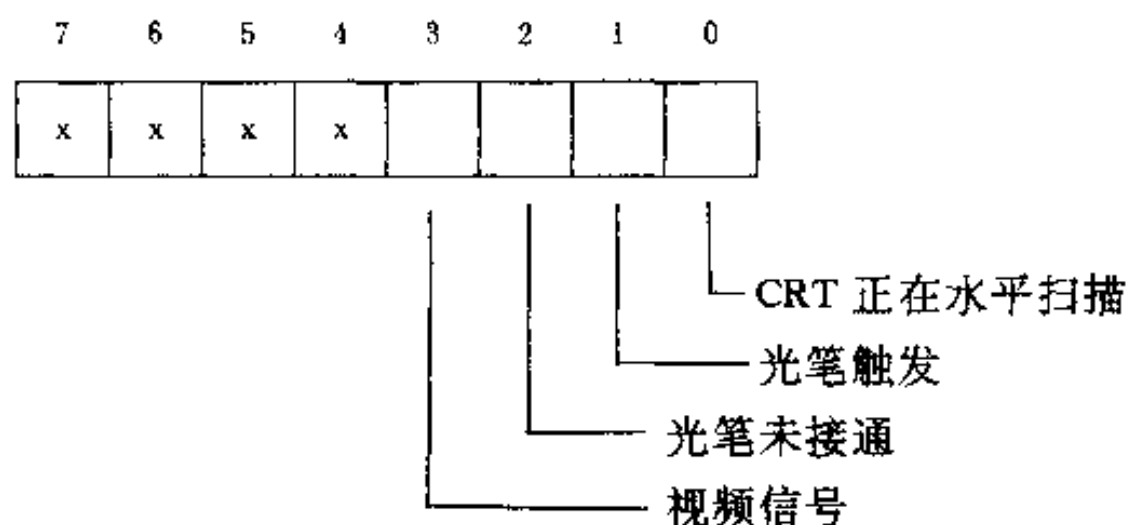
需要注意的是,字符本身的显示颜色及其底色(8×8 方块)不

是由彩色选择器中的 I、R、G、B 决定,而是由字符的属性字节内容决定的。字符属性的定义如下:



彩色选择器中的第 5 位只在 320×200 的彩色图形显示模式时有用,说明象素颜色采用 0 号配色器还是 1 号配色器。 320×200 彩色图形显示模式中每个象素对应显示缓冲器中的两位(C_1 和 C_0),因此各象素可有四种取值、即四种颜色,见表 3-5。

I/O 端口地址 3DAH 的状态寄存器的各位定义如下:



第 0 位为 0 时表示 CRT 在水平或垂直回扫期,可以向显示缓冲器送入新的显示信息而不影响屏幕画面。第 3 位是字符显示时视频输出信号的瞬间状态,利于系统故障诊断和分析。

3.1.5 单色高分辨率显示

对于象 Hercules 这样具有 64KB 显示缓冲器的单色卡,参照 ROM BIOS 中的视频 I/O 驱动程序 VIDEO-IO 重新编写一个驱动程序,让它常驻内存。再修改 INT 10H 的入口地址,形成新的

表 3-5 象素颜色定义

彩色选择器第 5 位	象素 C_1 C_2 值	象素颜色
0	0 0	与底色同
0	0 1	绿
0	1 0	红
0	1 1	黄
1	0 0	与底色同
1	0 1	青
1	1 0	品红
1	1 1	白

视频 I/O 驱动机理,就可以实现单色显示的 640×400 象素或 720×348 象素图形显示。

这种修改程序的中心思想是改变显示方式,调整 6845 CRTC 的初始化参数。当然还要考虑显示缓冲器地址及其分配等等。

以 640×400 象素图形显示为例,它取彩色图形方式,而其定时参数是以字符为单位的。设每屏显示 100 行字符,每字符行只有四条扫描线,每行 40 个彩色字符。由于显示一个彩色字符需要两个字节的的信息,用于黑白方式时,可以在一个字符的定时信号中显示 16 个象素点,即每行 640 个象素点。每屏 100 行,每行四条扫描线,即每屏垂直显示 400 线。因此得 640×400 象素单色高分辨率显示。初始化参数见表 3-6

其中一些参数可以根据监视器参数略作调整,比如有的监视器给出回扫时间参数,则各水平扫描参数可由监视器扫描行频、水平回扫时间、每行显示字符数来确定。如果没有详细监视器参数,应用上述初始化参数后有问题,可以根据寄存器参数的含义进行实验调整。

当以彩色图形方式编程时, R_0 的原参数为 01,即每字符行为两条扫描线。这时屏幕上的象素位置与显示缓冲器地址的对应关系如图 3-4 所示。VIDEO-IO 程序按此关系把象素位置转换成对应的存贮地址,实现图形信息的写入或读出操作,来显示或擦除图形

及文字信息。

表 3-6 640×400 单色图形显示初始化参数

寄存器	原参数	现参数	单位	现参数含义
R ₁₀	38	36	字符	水平扫描总长 55 字符,880 像素
R ₁	28	28	字符	水平显示 40 字符,640 像素
R ₂	2D	2F	字符	水平同步从第 46 字符起
R ₃	0A	07	字符	水平同步脉宽 7 个字符
R ₄	74	72	字符行	垂直扫描总长 115 字符,460 线
R ₅	06	02	扫描线	垂直总调节 2 线
R ₆	64	64	字符行	垂直显示 100 字符行,400 线
R ₇	70	64	字符行	垂直同步从第 100 行起,宽度固定为 16 线
R ₈	02	02	--	逐行扫描
R ₉	01	03	扫描线	每字符行 4 线

现在 R₉ 参数设为 03,即每字符行为四条扫描线,所以 32KB 显示缓冲应分成四块。块中像素点位置在视频内存中的对应地址关系为:

块 0: B000:0000-1F3F,8000 字节,存 0、4、8、…396 行像素;

块 1: B200:0000-1F3F,8000 字节,存 1、5、9…397 行像素;

块 2: B400:0000-1F3F,8000 字节,存 2、6、10、…398 行像素;

块 3: B600:0000-1F3F,8000 字节,存 3、7、11…399 行像素。

以彩色图形方式编程时,像素点位置和视频内存对应地址的转换公式为:

$$\begin{aligned} \text{偶行像素的地址} = & \text{INT}(\text{行数}/2) \times 80 + \text{INT}(\text{列数}/8) \\ & + \text{B8000H} \end{aligned}$$

$$\begin{aligned} \text{奇行像素的地址} = & \text{INT}(\text{行数}/2) \times 80 + \text{INT}(\text{列数}/8) \\ & + \text{B800DH} + 2000\text{H} \end{aligned}$$

而单色 640×400 像素显示的相应转换公式为:

$$\begin{aligned} \text{像素地址} = & \text{INT}(\text{行数}/2) \times 80 + \text{INT}(\text{列数}/8) \\ & + ((\text{行数}) \text{MOD} 4) \times 2000\text{H} + \text{B0000H} \end{aligned}$$

这里 INT 表示取整, MOD 取余数。因此, (行数) MOD 4 得块号 0~3; INT(行数/4) 得本块中的行号 0~99; INT(列数/8) 为行内字节号 0~79。算得地址后, 再用下式算出象素点在该字节中的位号:

$$\text{位号} = (\text{列数}) \text{MOD} 8$$

例如, 需要在 4 行 20 列处显示一点, 它对应的视频内存地址和位号是:

$$\begin{aligned} \text{象素(字节)地址} &= \text{INT}(4/4) \times 80 + \text{INT}(20/8) \\ &\quad + (4 \text{MOD} 4) \times 2000\text{H} + \text{B0000H} = \text{B0052H} \end{aligned}$$

$$\text{位号} = 20 \text{MOD} 8 = 4$$

请注意公式中十进制和十六进制(用 H 表示)混用, 容易出错。

此外, 在图形上卷和下卷子程序中有关的地址增量要作相应修改(+4000H), 移动一行和擦除一行子程序由于显示缓冲器分成四块, 指令需要重复执行来完成。

IBM PC 在图形方式下显示字符不是由图形卡上的字符发生器形成的, 而是通过查找 ROM BIOS 中位于 FA6EH 处字符点阵表, 然后将点阵写入视频内存指定地址来完成字符显示。由于改动的驱动程序被装在系统内存某个区域, 其段址与 ROM BIOS 的不同, 因此需要修改以保证对字符点阵表正确寻址。

由于分辨率从 640×200 提高到 640×400, 每帧将显示 50 行×80 个字符, 字符会缩小一倍, 用作图形注解尚可, 若显示文本则嫌小一些。恢复成 25 行×80 列字符的显示格式的办法是将原来送入偶行的点阵字节同时存入块 0 和块 1, 送入奇行的字节同时存入块 2 和块 3, 类似于隔行扫描的效果。而读出时只要对块 0 和块 2 进行即可。

对于 720×348 单色图形显示来说, 做法类似。6845 CRTC 寄存器初始化参数如下:

$$R_0 = 35\text{H}, R_1 = 2\text{DH}, R_2 = 2\text{EH}, R_3 = 7, R_4 = 5\text{BH}, R_5 = 2, R_6 =$$

57H, $R_7 = 57H$, $R_8 = 2$, $R_9 = 3$, $R_{10} = 0$, $R_{11} = 1$ 。

显示缓冲区从 B000H:0000 开始,共分四页(块),每页大小为 2000H 到 B000:7FFF 为止。

设屏幕从第 0 行开始,至第 347 行结束。第 0 页放的是 0、4、8、……344 行象素,第 1 页放的是 1、5、9、……345 行,第 2 页 2、6、10、……346 行,第 3 页 3、7、11、……347 行。每行 90 字节。比如说第 0 页第 0 行从 B000:0000 至 B000:0059H。每个字节对应八个象素点。

3.2 EGA 和 CGE400

3.2.1 EGA 和 CGA 的比较

随着微型计算机图形应用的扩展,最初的彩色图形适配器 CGA 的局限性也就越来越明显。CGA 设计时考虑能和电视机及组合监视器兼容,使垂直分辨率限于 200 行,四色模式时最高水平分辨率也只有 320 象素。

增强型图形适配器 EGA,通过保持与 CGA 兼容,扩充显示缓冲器,增加 BIOS 对高分辨率的支持而成为广泛应用的一种图形标准。它提供 640×350 象素分辨率,64 选 16 的色彩性能。

从 EGA 对 INT 10H 中视频 BIOS 例程的处理中可以看到 EGA 兼容性的扩展:EGA 把原来的 BIOS 例程放在 INT 42H,而将自己的 BIOS 例程放在 INT 10H。完成某些功能时 EGA BIOS 调用原来的程序,因此 EGA 是把 BIOS 做了扩展,而不是替代,这样可以完全支持原来的 BIOS 调用,新调用的操作也与原来保持一致。

EGA 和 CGA 的结构是完全不同的。CGA 是基于 Motorola(莫托罗拉)6845 CRTC 芯片的,EGA 使用 IBM 专用芯片(不少厂家开发出了功能基本一致的芯片)。其中有一些寄存器模拟 6845 的功能,因此 EGA 可以和 CGA 硬件兼容。

EGA 最明显的结构改进是其存储器组织。CGA 用连续的数据位描述各彩色像素。这种方法编程容易,但浪费了视频内存地址空间(颜色数加倍时要求存储映象也加倍)并大大降低了使用多种色彩的图形应用程序的速度(颜色数加倍使得存储器写操作时间也加倍)。EGA 采用位面(bit plane)技术,同一像素点在各位面上占同一地址,而不同位面上同一像素地址中的内容形成色彩组合。

EGA 对不少用户来说,已经满足大多数应用的要求了,不过人们的要求随着时代的发展也在不断变化。以往认为满意的效果,在现在看来又落伍了,有的应用,比如 CAD、台式印刷系统本来对显示就有很高的要求,所以图形显示在继续迅猛发展。

EGA BIOS 提供 20 个用于显示的基本子程序,功能编号为 0~13H。要调用某个功能,先将功能号放在 AH 中,然后使用 INT 10H。大多数例程还需要通过 AL、BX、CX 和 DX 等传递参数。

EGA 有五种字符模式和七种图形模式,见表 3-7。

表 3-7 EGA 显示模式

模式号	显示类型	兼容情况
0	40×25 单色字符(320×200)	CGA
1	40×25 彩色字符(320×200)	CGA
2	80×25 单色字符(640×200)	CGA
3	80×25 彩色字符(640×200)	CGA
4	320×200 4 彩色图形	CGA
5	320×200 单色图形	CGA
6	640×200 单色图形	CGA
7	80×25 单色字符(720×350)	MDA
0DH	320×200 16 彩色图形	
0EH	640×200 16 彩色图形	
0FH	640×350 4 彩色图形	
10H	640×350 16 彩色图形	

CGA 的显示缓冲器的首地址总是 B800:0000H,EGA 显示缓冲器的首地址可以通过编程设置,共有三种:B000:0000H(用于模拟单色显示)、B800:0000H(用于模拟 CGA)和 A000:0000H(用于

EGA 特有模式)。

EGA 卡上有 256KB 显示缓冲器,占系统视频内存空间 64KB,不仅满足 640×350 象素分辨率、16 色、二页显示,且有剩余。

3.2.2 位面技术

CGA 的显示缓冲器中要用连续的 4 位或 2 位来表示彩色低分辨率或中分辨率的一个象素(参见 3.1.4 CGA)。只有在单色高分辨率(640×200)时,显示缓冲器中的 1 位和一个象素一一对应。就是说,象素及其色彩是用若干连续位表示的,颜色越丰富,连续位数就越长。而视频内存空间是有限的,以 640×350 象素分辨率、16 色计算: $640 \times 350 \times 4/8 = 112000$ 字节,即大约 110KB,若再增加分辨率或色彩则很难表达。此外,这种方法也影响执行速度。

所谓位面(bit plane)技术是指把显示缓冲器分成若干色平面,各平面上相同位置的一位和同一个象素对应、具有相同的视频内存映射地址。色平面越多,可表达象素的色彩越丰富。

EGA 的显示缓冲器为 256KB,占视频内存 64KB,2 页 640×350 象素、16 色。实际上一页大约需 120KB,分成四个平面,分别代表红、绿、蓝和加亮(RGBI);因此占视频内存地址空间仅为 $640 \times 350/8 = 28\,000$ 字节,不到 28KB,而能有 640×350 象素分辨率,16 种色彩。见图 3-7。

由此可见,EGA 的映射关系很简单,但写不同颜色时需要选择不同的色平面(通过色面开关)来实现。也就是说,视频内存的每一位和显示缓冲器各平面的一一对应,最终对应一个屏幕象素,而该象素的颜色则是 16 选一。它的颜色是由显示缓冲器四个不同位面上同一地址的四位决定的。

这种位面组织下,增加一个位面、色彩就可以增加一倍,而存储器写操作程序无需重新计算新地址,程序兼容性好。

对于位面这种组织,修改只有两种色彩的模式的内存和修改无论多少种色彩的模式的内存,所花的时间是一样的。因为视频内存和屏幕象素一一对应,修改视频内存内容就等于修改了屏幕象

素。至于色彩是由色平面控制的,在色平面控制一定的情况下,修改视频内存内容也就等于修改对应的显示缓冲器存储体(以位面为平面,以各色平面为深度)的内容。

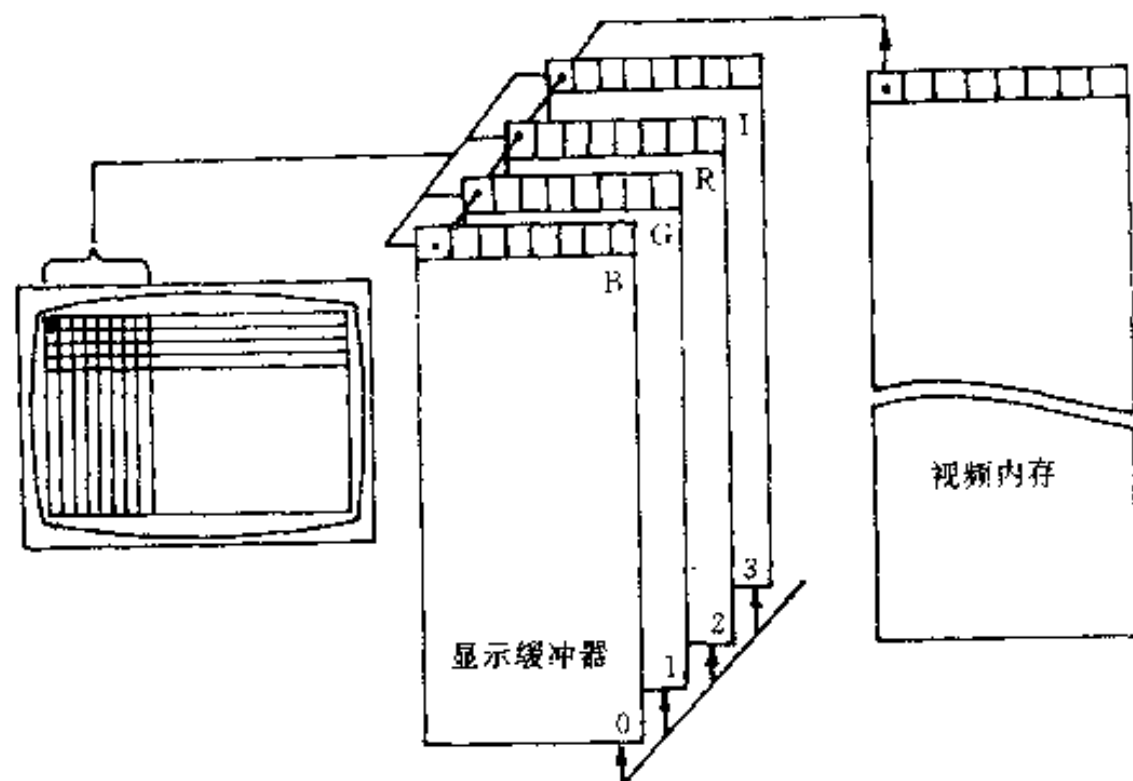


图 3-7 EGA 屏幕象素与显示缓冲器及视频内存的映射关系

色平面的控制是通过映象屏蔽寄存器(map mask register)来实现的,它确定哪一个位面内容将会变化。所以,它相当是位面屏蔽。该寄存器是只写的,端口地址是 3C5H。送数据前应先将 02H 送入 3C5H 对该端口初始化,然后通过索引指向该寄存器,它的 0~3 位对应于位面 0~3,高四位未用。哪一位置 0,对应的位平面将不受视频内存数据的影响,也就是说该位面保持原来已有的内容不变。哪一位置 1,则对应的位面内容就可以修改。

显然,映象屏蔽寄存器只是确定哪一个(或哪一些)位面内容将改变,而内容的修改则需要通过锁存器来完成。CPU 不能直接访问 EGA 卡上的显示缓冲器,而是把数据送往对应四个位面的四个八位锁存器,当然数据能不能送到各锁存器,还要看映象屏蔽寄

寄存器选中哪个(些)位面而定。

实际情况还要复杂一些,CPU 把数据送入锁存器之前,优先考虑使能设置寄存器。使能寄存器选中的位面将全部添上置位/复位寄存器中相应的数据。比如,为了写入一纯颜色,首先应把位面全写成零,可以这样来进行:首先把置位/复位寄存器置零,然后把映象屏蔽的逻辑非值置入使能寄存器(如,映象屏蔽值为 0101H,则使能寄存器值为 1010H)就可。

由于 CPU 以字节为单元传送数据,所以每次会改变八个像素。这种方式对于图形模式下写字符可能很方便(字符为八位宽的点阵),因为一次就可以完成八个像素点的设置,但不能控制单一像素。位屏蔽寄存器(bit mask register)就是用来控制单一像素的。当位屏蔽值的某位为 1 时,则允许改变对应的单一像素。和面屏蔽类似,位屏蔽本身并不能设置像素,像素的开闭仍然由 CPU 传送的数据字节中的相应位是 1 或 0 决定,只是像素的状态变化与否受位屏蔽的控制。

IBM 微机端口空间的限制增加了硬件的复杂性。以上提及的这些寄存器都是通过地址寄存器间接选择的。正如把位面看作是存储器分层,这也可以看作寄存器分层,只不过一次只能修改一个被索引的寄存器(位面可有多个同时被修改)。

3.2.3 CGE 400

CGE400(Color Graphics Enhancer 400)彩色图形增强卡是 IBM 微机彩色图形卡的一个增强型,它的控制芯片依然是 M6845。

CGE 400 在中分辨率及文本方式下与 CGA 和 EGA 兼容,并提供 640×400 像素分辨率、16 色的高分辨率彩色图形方式。CGE400 的显示缓冲器为 128KB(相当 EGA 的一页),占视频内存地址 B8000H~BFFFFH 计 32KB 空间。它和 EGA 类似,显示缓冲器也分为四个色平面(每个平面 32KB)。

CGE400 的扫描方式类似 CGA,只是把 32KB 地址分为四个 8KB 大小的区域,各区域存储所谓的分域扫描映象分别对应 100

条扫描线,采用隔行方式(EGA 是逐行方式,因此计算象素点的存储位置很简便)。当 CGE 400 工作在 640×200 图形方式时,关闭后两个区域,用前两个存储区作标准 IBM PC 的显示方式。在 640×400 图形方式下,视频内存的这四个区域中的每一位直接对应于屏幕上一个象素的开关状态。屏幕象素的水平 X 坐标范围为 $0 \sim 639$,垂直 Y 坐标范围为 $0 \sim 399$ 。每组连续的 80 个字节(640 位)对应一条扫描线,每个字节中的一位对应一个象素,最高位对应最左端象素(指一个字节对应八个象素,这八个象素中的最左端象素)。因此这方面和 EGA 也是一致的。不过因为是隔行扫描,象素和其在视频内存中的位置的关系则要复杂一些,参见图 3-8。

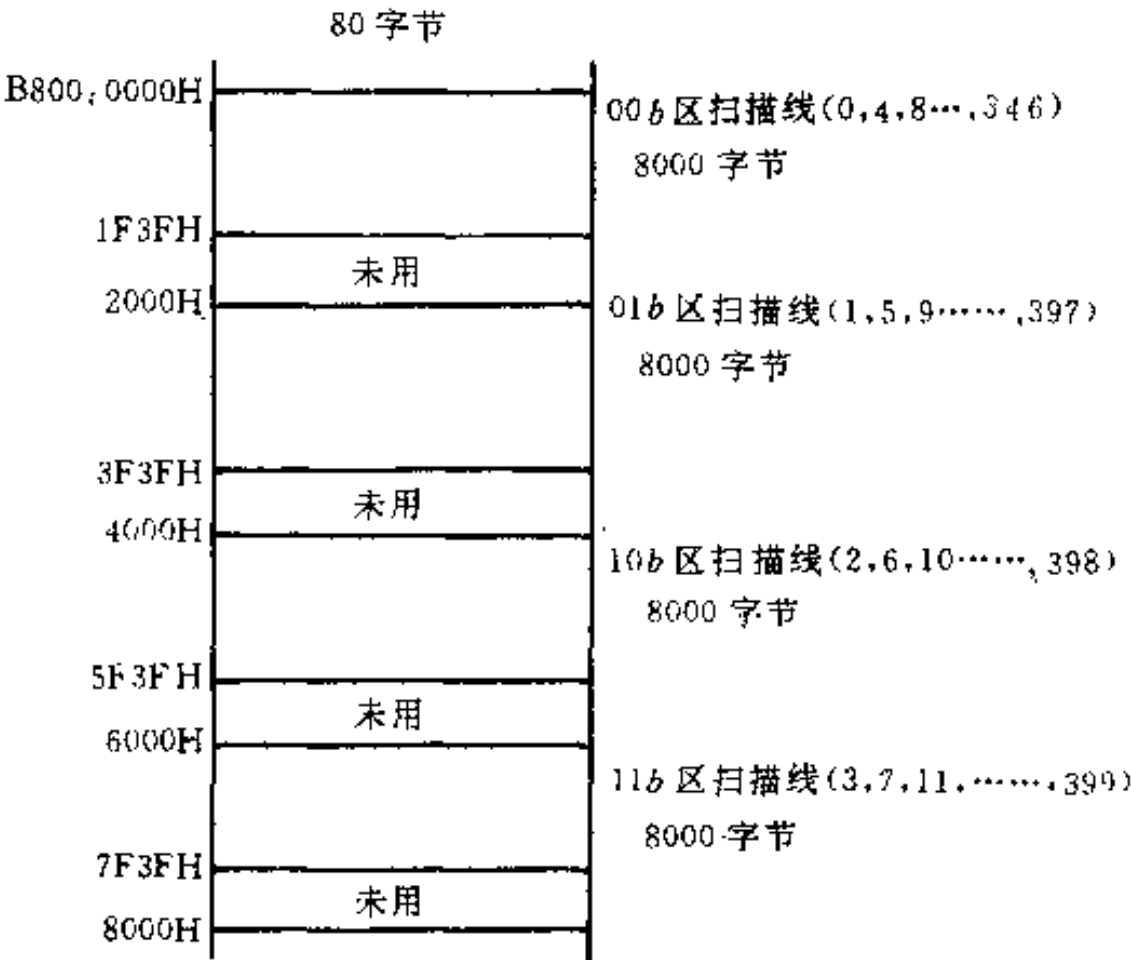


图 3-8 CGE 400 视频内存图

设象素屏幕位置为 (X,Y) , $0 \leq X \leq 639, 0 \leq Y \leq 399$, 求该象素在视频内存中对应的地址。

首先确定在哪一区域:

区号 = $Y \text{MOD} 4$;

然后求得在该区内是第几条完整扫描线之后:

线数 = $\text{INT}(Y/4)$;

再算它在该扫描线之后是第几个字节:

字节数 = $\text{INT}(X/8)$;

以及在该字节中是第几位:

位数 = $X \text{MOD} 8$.

从而得含该象素的字节在视频内存中的位置距 B800:0000H 的偏移地址是:

区号 $\times 8192$ + 线数 $\times 80$ + 字节数

在该地址字节的第 $X \text{MOD} 8$ 位内容控制了屏幕象素(X,Y)的开关。

系统在每一时刻只能对一个色平面进行访问,这一点 CGE 400 和 EGA 不同,因此 CGE 的速度要受到影响。以 I/O 口地址为 2DEH 的色平面寄存器的低 2 位来切换所要访问的色平面。其中平面 0 为蓝,平面 1 为绿,平面 2 为红,平面 3 为加亮(这种安排与 EGA 相同)。参见图 3-9。

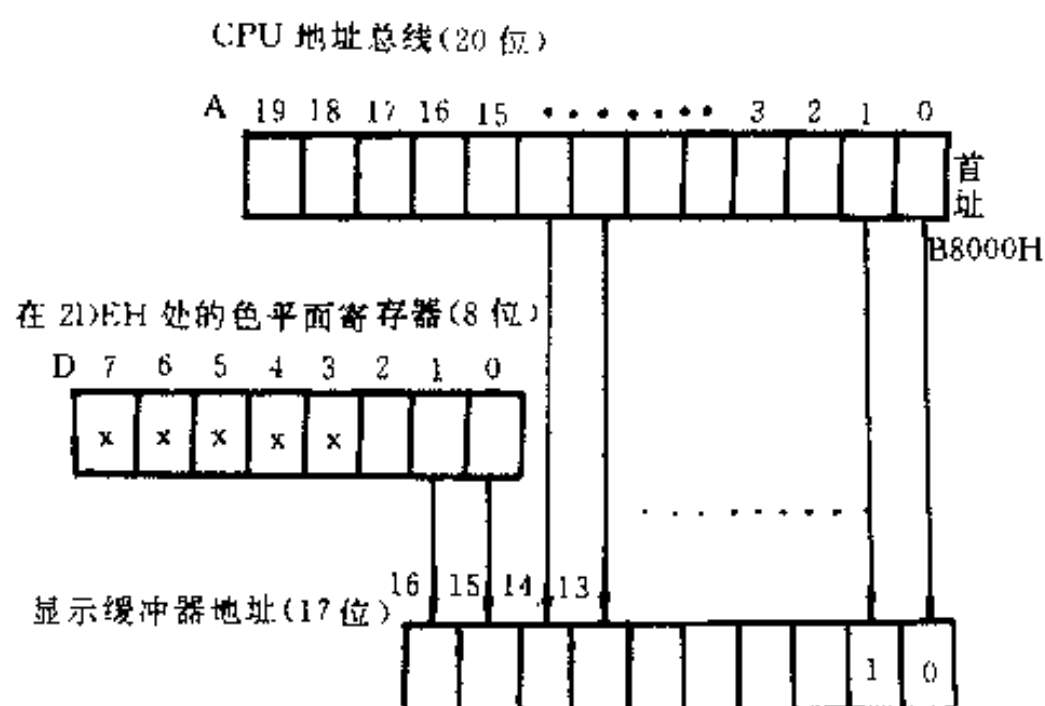


图 3-9 CPU 对显示缓冲器的访问

3.3 MCGA 和 VGA

3.3.1 MCGA 和 VGA 概述

VGA (Video Graphics Array) 是制作在 IBM PS/2 系列机上的视频子系统,同时引入的还有 MCGA (Multi-Color Graphics Array)。这两种视频子系统是装在 PS/2 主板上的,PS/2 30 用 MCGA,50、60 和 80 用 VGA,它们不是独立的插卡。

从兼容性考虑,IBM 也提供 VGA 适配器卡用于 XT、AT 和 PS/2 30 来实现 VGA 功能。事实上,众多的兼容产品已把 VGA 搞得五花八门。除了在 640×480 象素分辨率上尚保持象样的兼容性之外,对该分辨率以上的模式,兼容性通常总是个问题。

MCGA 和 VGA 和以前的 IBM 视频显示控制器(适配卡)不同,它们采用模拟信号监视器,取代了以前的数字信号监视器。数字信号监视器接收从图形适配卡送来的 RGB 彩色数字信号是以通或断的形式出现的,因此能表现的色彩非常有限。比如 EGA 产生的六种信号只能显示总数为 $64(2^6)$ 种不同的颜色。

与此对应的 PS/2 监视器用的 RGB 信号不是简单的通、断,而是连续变化的电平。因为彩色显示亮度和彩色驱动信号的电平有关,所以模拟式监视器可以显示多得多的彩色,也不用象数字信号监视器那样通过增加信号线来增加色彩表现能力。

从功能上来看,VGA 实质上是 EGA 的一种适度的改版,但是 VGA 在显示的视频模式分辨率方面要比 EGA 灵活得多,在使用多频监视器时尤其明显。

而 MCGA 是 CGA 的扩充,在 CGA 上运行的程序可以不加修改地在 MCGA 上运行。即使这些程序绕过视频 BIOS、对硬件直接编程也没有关系。PS/2 30 有和 PC 兼容的总线,因此单色显示卡或 Hercules 显示卡可以和 MCGA 同时使用。

VGA 和 EGA 在编程接口方面是类似的,它的控制口及缓冲寻址和 EGA 兼容,因此在 EGA 上运行的程序一般来说也可以在 VGA 上运行。VGA 和 EGA 在硬件一级兼容性很好,在 VGA 上可运行直接访问 EGA 寄存器的程序。

从编制程序的角度来看,MCGA 和 VGA 之间并没有多大相似之处。MCGA 和 VGA 规定的 I/O 口明显不同,它们的视频内存情况也不一样。绕过视频 BIOS、直接控制硬件的程序一般不能保证在 MCGA 和 VGA 上都可以运行。

3.3.2 MCGA

MCGA 电路的主要部分是两种特有的门阵列,一种是内存控制器门阵列,它用来配合 CRT 控制器;另一种是视频模式控制门阵列,它控制视频模式选择及彩色属性解码。

通过在 I/O 口地址 3D4H 和 3D5H 的一组八位的寄存器,人们可以对内存控制器编程。见表 3-8。

和 CGA 一样,访问寄存器首先得向端口 3D4H 写入该寄存器号,然后对在 3D5H 的指定寄存器进行读写。不同的是所有内存控制寄存器都可读写。这对调试程序是很方便的,而对和 CGA 保持兼容不利。

内存控制寄存器中的前 16 个(0~0F)和 CGA 用的 CRT 控制器芯片 M6845 的相仿。因此直接访问这些寄存器的 CGA 程序可以在 MCGA 上运行。由于 MCGA 用的 CRT 水平及垂直时序参数的缺省值和 CGA 的不同,需要对前七个寄存器实行写保护。这样,CGA 程序想更新这些寄存器时不会因疏忽而瓦解事关紧要的 CRT 时序信号。寄存器 10H 是模式控制寄存器,它的最高位是有关时序参数寄存器的写保护位。

余下的内存控制寄存器控制视频模式选择及字母数字字符发生器。这些在 CGA 上是没有的。它们提供的功能和 VGA 的类似,有附加的图形模式和可装入内存的字母数字字符集。

表 3-8 MCGA 内存控制寄存器

寄存器号	功 能	备 注
0	水平扫描总量	和 CGA CRT 控制器同
1	水平显示数	
2	水平同步位置	
3	水平同步宽度	
4	垂直扫描总量	
5	垂直总调节	
6	垂直显示数	
7	垂直同步位置	
8	(保 留)	
9	每字符扫描行数	
0A	光标开始	
0B	光标结束	
0C	起始地址高位	
0D	起始地址低位	
0E	光标地址高位	
0F	光标地址低位	
10	模式控制	
11	中断控制	
12	字符发生器,同步极性	
13	字符发生器指针	
14	字符发生器计数	

视频模式控制器支持三个 CGA 兼容的控制寄存器: I/O 口地址 3D8H 的模式控制寄存器, 它控制显示模式选择; 3D9H 的色彩控制寄存器, 它控制调色板及图形模式背景色的选择; 3DAH 的状态寄存器是只读的, 其内容指示 CRT 水平及垂直时序信号状态。

除了 CGA 支持的 6 种视频显示模式之外, MCGA 提供 640×480 两色图形模式(视频 BIOS 模式 11H)及 320×200、256 色图形模式(BIOS. 13H)可用 INT 10H 功能 0 来设置模式:

```

mov     ah,0                ; AH = INT 10h function 7
mov     si,video ModeNumber ; AH = 11h (640×480 two-color)
                        ; 12h (640×480 16 color)
                        ; 13h (320×200 256 color)
int     0h                  ; Call video BIOS

```

清单 3-1 设置视频模式

MCGA 有两个特点对编程来说很有意思。一是字母数字及图形模式的垂直分辨率都比以往 IBM 视频显示适配器高,再就是 MCGA 可以从 262 144(256K)种色彩中选择,同时显示 256 色。

缺省 BIOS 视频模式的垂直分辨率见表 3-9。

表 3-9 MCGA 和 VGA 新 BIOS 视频模式

模式号	模式意义	MCGA	VGA
0	40×25 16 色字母数字 (320×400 分辨率)	✓	
0	40×25 16 色字母数字 (350×400 分辨率)		✓
2	80×25 16 色字母数字 (640×400 分辨率)	✓	
2	80×25 16 色字母数字 (720×400 分辨率)		✓
11H	640×480 双色图形	✓	✓
12H	640×480 16 色图形		✓
13H	320×200 256 色图形	✓	✓

在字母数字字符模式,垂直分辨率为 400 线,是 CGA 的两倍,比 EGA350 线“增强”模式也高。因为在该模式中 BIOS 依然显示 25 行字符,字符垂直方向为 16 线,高分辨率的字符清晰易读。

当 MCGA 模仿 CGA 图形模式(640×200 二色及 320×200 四

色)时,象素扫描线在垂直方向加倍,虽然分辨率标为 200,显示仍是 400 线,MCGA 显示效果比 CGA 清晰。

MCGA 和 VGA 的另一特色是彩色显示能力扩展了。基础是采用数模转换器(DAC)产生模拟 RGB 信号来驱动 PS/2 单色及彩色监视器(单色监视器只响应绿信号)。

视频数模转换器利用一组 256 个 18 位内部寄存器,每一个规定一种 RGB 组合。18 位彩色寄存器中的每六位分配给三原色中的一色。数模转换器把这六位值转换为相应的模拟电压信号输出给监视器。由于对彩色寄存器中任何一种原色都可由视频数模转换器产生 64 种色彩,从而最终可生成 256K(64^3)种彩色组合。因为共有 256 个视频数模转换彩色寄存器,该视频子系统一次可显示 256K 彩色组合中的任何一组 256 色。

3.3.3 MCGA 编程

1. 视频 BIOS

PS/2 30 上的视频 BIOS 提供 PC 主板 ROM BIOS 一样的功能,编程接口也相同;通过 CPU 寄存器传递参数,利用 INT 10H 调用 BIOS 程序来实现所有视频 BIOS 功能。

当然 30 型和 PS/2 系列其他型还有几个 INT 10H 新功能,参见表 3-10。

IBM 扩充了 INT 10H 的功能 10H 以便访问视频 DAC 彩色寄存器。功能 12H 有几个新子功能用于改变其他 BIOS 程序的动作。比如当视频模式改变时,设 BL 寄存器为 31H、调用 INT 10H 功能 12H 来允许或禁止装载缺省调色板。

INT 10H 功能 1AH 和 1BH 是 PS/2 系列机的新功能,可以通过调用它们来确定视频子系统的状态。功能 1AH 回送显示组合码,它表示所使用的监视器型号。功能 1BH 回送一张表,其内容描述视频 BIOS 的当前状态:当前视频模式、有效视频页、可用视频内存量等等。这些功能在多种视频模式下运行的程序中是很有用的。当然,在必须确定当前视频状态、产生适当的视频输出的上弹式内

存驻留程序中也很需要。

表 3-10 MCGA 和 VGA 新 INT 10H 功能

功能 10H:	彩色——调色板接口
	AL=3: 开关字母数字强度/闪烁状态
	7: 读单独调色板寄存器(只是 VGA 有)
	8: 读过扫(边界色)寄存器(只是 VGA 有)
	9: 读所有的调色板寄存器和过扫寄存器(只是 VGA 有)
	10h: 设单独视频 DAC 彩色寄存器
	12h: 设视频 DAC 彩色寄存器
	13h: 选视频 DAC 彩色页(只是 VGA 有)
	15h: 读单独视频 DAC 彩色寄存器
	17h: 读视频 DAC 彩色寄存器组
	1Ah: 读视频 DAC 彩色页状态(只是 VGA 有)
	1Bh: 灰度总计
功能 11H:	字符发生器接口
	AL=4: 装载 8×16 字母数字字符
	14h: 设 8×16 字母数字字符模式
	24h: 装载 8×16 图形字符
功能 12H:	替代选择
	BL=30h: 选择字母数字模式的垂直分辨率(只是 VGA 有)
	31h: 允许/禁止装载缺省调色板
	32h: 允许/禁止视频寻址
	33h: 允许/禁止缺省灰度总计
	34h: 允许/禁止字母数字光标仿真(只是 VGA 有)
	35h: 显示开关接口
功能 1AH:	显示组合码
	AL=0: 读显示组合码
	1: 写显示组合码
功能 1BH:	功能/状态信息
功能 1CH:	保存/恢复视频状态(只是 VGA 有)
	AL=0: 回送状态缓冲区大小
	1: 保存视频状态
	2: 恢复视频状态

2. 字母数字模式编程

字母数字模式编程与 MCGA 改进的分辨率无关,实质上和在 CGA 时一样。重要的不同之处在于 MCGA 可以显示用户定义的字符(EGA、VGA、Hercules Graphics Card Plus 等也有该功能)。MCGA 字母数字字符发生器可显示视频内存中定义的四张 256 字符表中的任何字符。

为了使 MCGA 的字符发生器显示这些字符集中的一个,首先把定义字符的位样图(bit pattern)装入视频内存;然后对字符发生器编程把位样图从视频内存拷贝到内部的两张字符定义表的一张中去(IBM 技术文件中称这些字符定义表为字型页,font pages)。

用 MCGA 内存可装字符集来显示小于缺省(16 线)的字符是很容易的,参看清单 3-2。

程序调用视频 BIOS 来装入缺省的图形模式字符定义(字符只有八线高),由字母数字字符发生器使用。对 MCGA 内存控制器重新编程,由于字符只有八线高,所以一屏为 50 行,每行 80 个字符。

除了内存可装字符集之外,MCGA 重复了几乎所有的 CGA 功能。不过 MCGA 不受字母数字模式中显示干扰问题的影响。在 CGA 时,必须仔细同步协调在显示刷新周期中 CPU 访问视频内存及水平与垂直回扫间隔。不如此的话,每当程序访问视频缓冲区时,会在屏幕上看到干扰的随机图样或雪花点。MCGA 的硬件设计避免了这种显示干扰。

在 CGA 字母数字模式中,可以对彩色选择寄存器(I/O 口地址 3D9H)编程来选择 16 种彩色中的一种来显示边界。在 MCGA,可以依然对该寄存器编程,但并不显示边界。

3. 图形模式编程

在 CGA 兼容的 640×200 二色及 320×200 四色模式中,MCGA 对 CGA 进行仿真。和在 CGA 中一样,系统以隔行方式把像素映射在 B800:0000H 处的视频内存。在 640×480 二色和 320×200 、256 色模式时,视频内存寻址是不同的。

在 640×480 二色和 320×200 、256 色模式时, 视频内存区从 $A000:0000H$ 开始。这两种模式在视频内存的映象和在屏幕上的图是线性关系。在 640×480 二色模式中, 每一位代表一个象素, 所以视频内存中的一个字节代表八个象素, 80 字节表示屏幕上的 10 行象素, 参见图 3-10。

; 把视频 BIOS 8×8 字符装入字母数字字符发生器

```
mov ax, 1102h    ; AH=INT 10h 功能号
                  ; AL=8\8 字符集装载
mov bx, 0        ; BX=装载块
int 10h          ; 把 8\8 字符装入内存
mov ax, 1103h    ; AH=INT 10h 功能号
                  ; AL=字符发生器装载
mov bx, 0        ; BX=装载块
int 10h          ; 把 8\8 字符装入字符发生器
```

; 对 CRT 控制器编程, 显示 8×8 字符

```
mov dx, 3D4h     ; DX=MCGA I/O 口地址
mov ax, 3D9h     ; AL=9(寄存器号)
                  ; AH=3(寄存器值)
out dx, ax       ; 更新扫描线寄存器
mov al, 0Ah      ; AL=0Ah(寄存器号)
out dx, ax       ; 更新光标开始寄存器
mov al, 0Bh      ; AL=0Bh(寄存器号)
out dx, ax       ; 更新光标结束寄存器
```

; 更新视频 BIOS 数据段的状态变量

```
mov ax, 40h
mov ds, ax        ; DS→视频 BIOS 数据段
mov word ptr ds:[4Ch], 80 * 50 * 2 ; 更新 CRT-LEN
mov byte ptr ds:[84h], 19          ; 更新 ROWS
mov word ptr ds:[85h], 8            ; 更新 POINTS
```

清单 3-2 MCGA 80×50 字母数字模式

在 320×200 、256 色模式中, 每一象素用一个字节代表 200 显示行, 每行 320 字节, 参见图 3-11。

设在该模式中象素点坐标为(X,Y),则用如下程序可计算得它在视频内存的映象地址(DS:BX)。

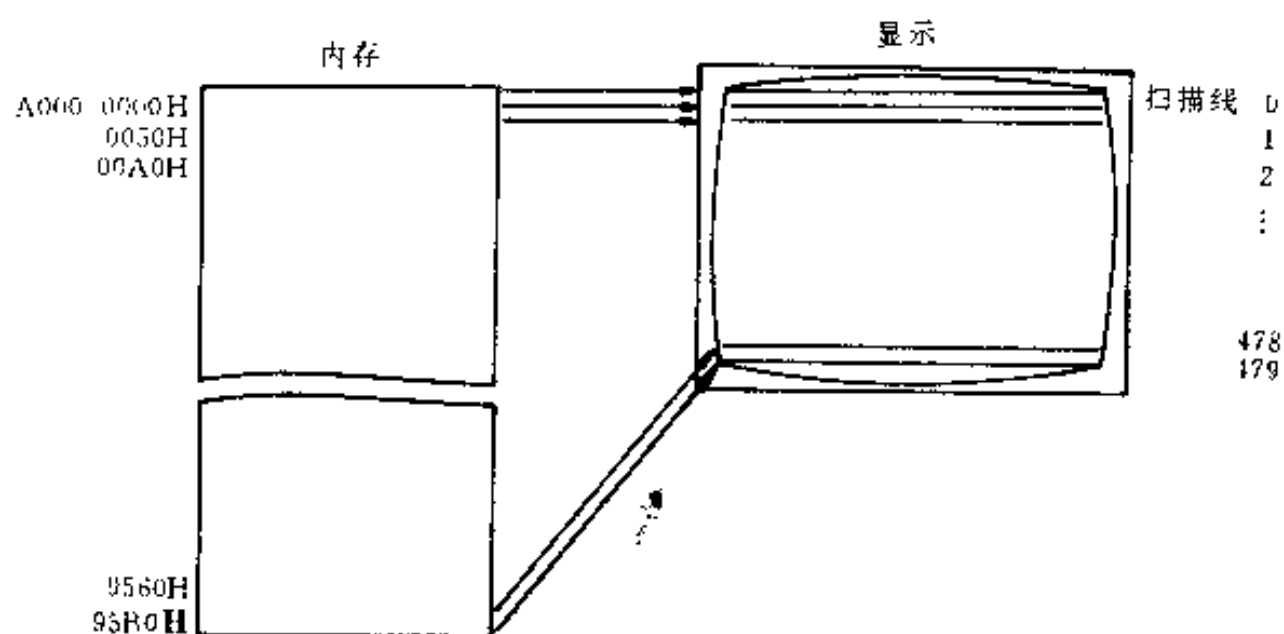


图 3-10 640×480 二色图形模式视频内存映射图

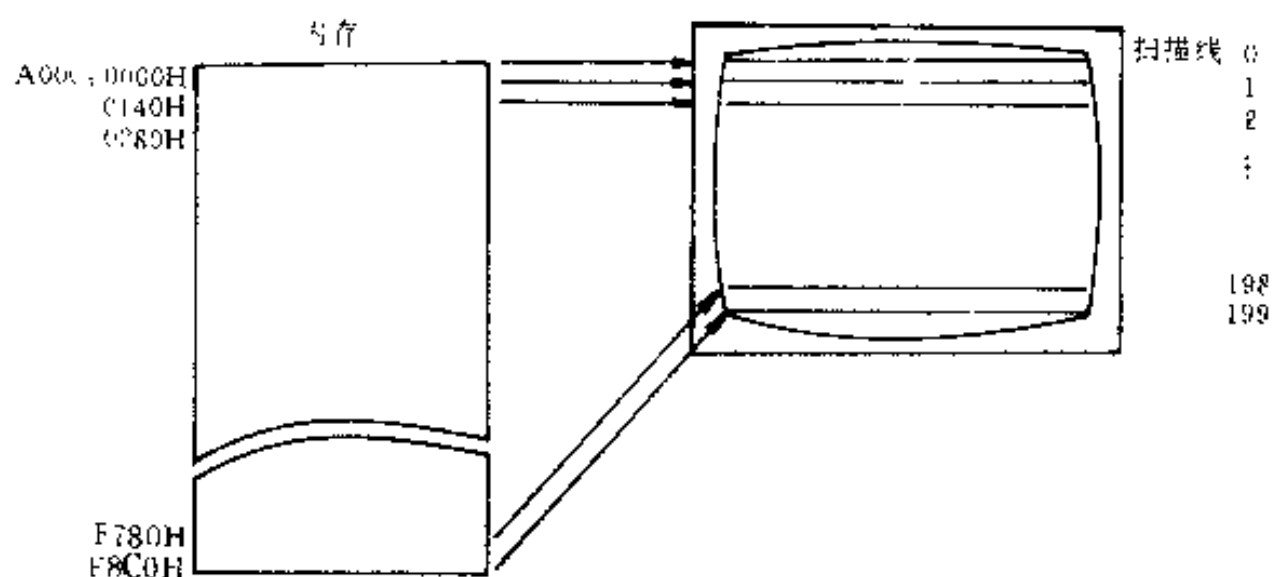


图 3-11 320×200 的 256 色图形模式视频内存映射图

```

SP1          PROC      near
;call with;    AX == Y coordinate
;              BX == X
;              CL == Pixel value
;compute address of pixel in video buffer
        mov     dx,320
        mul     dx
        add     bx,ax      ;BX == X + 320 * y
        mov     ax,0A000h
        mov     ds,ax      ;DS:BX -> pixel in video buffer
;update the pixel value
        mov     [bx],cl
        ret
SP1      ENDP

```

清单3-3 320×200 256色模式

。该程序根据每行像素数(320)乘上像素 Y 坐标再加上它的 X 坐标,计算出映象在视频内存中的地址,然后把存在 CL 中的像素值转存在该地址单元中,这时为320×200、256色模式。

MCGA 的640×480二色图形模式中它的水平及垂直方向单位尺寸中像素数相同,即像素是正方形的。因此,在该模式中作图不需考虑像素在水平、垂直方向尺寸不同而需调整的问题。

4. 视频 DAC 编程

利用视频 BIOS 对 MCGA 视频 DAC 进行编程,过程是很简明的。除320×200、256色图形模式之外,其他所有模式只用前16个视频 DAC 寄存器。视频 BIOS 以缺省方式装入一组 CGA 兼容的彩色数值。当然人们可以更新这些彩色寄存器中的数值,选用256K 可用彩色组合中的任何一种。

举例来说,以下程序显示如何来改变视频 DAC 寄存器7中的彩色数值。该彩色寄存器值大小为18位,红、绿、蓝各占6位。所定的数值越大,相应的原色显示强度越大。

```

mov  ah, 10h          ; AH = INT 10h function #
mov  bx, 7             ; BX = 7 (register #)
mov  dh, RedValue      ; DH, CH, CL = 6bit RGB Values
mov  ch, GreenValue
mov  cl, BlueValue
int  10h              ; Call video BIOS

```

清单3-1 更新视频 DAC 彩色寄存器

如果 MCGA 接单色显示的话,在视频 BIOS 把彩色数值装入指定的视频 DAC 彩色寄存器前,先执行灰度计算。这是一种加权平均计算,30%红加59%绿加11%蓝,它的结果是一种灰度值,对应指定彩色组合的综合强度。

3.3.4 VGA

VGA 和 EGA 一样有可编程的控制部件: CRT 控制器 (CRTC)、定序器、图形控制器及属性控制器,它们集成在一块芯片上。参见图3-12。

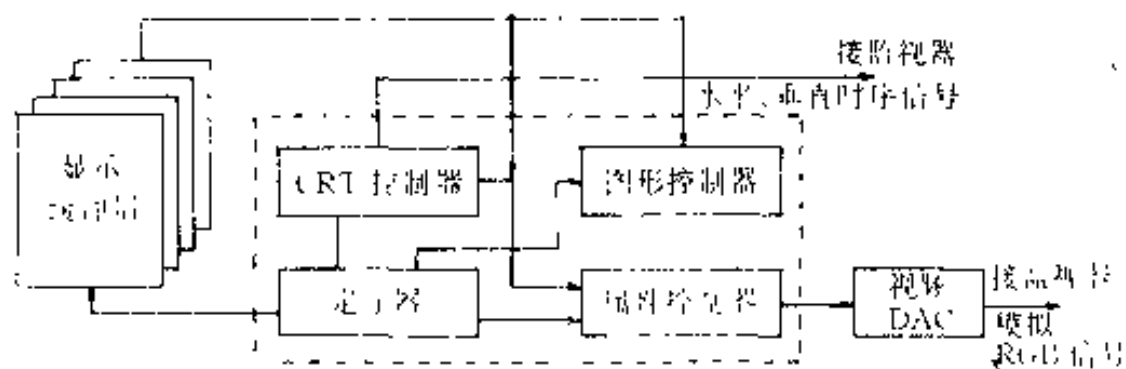


图3-12 VGA 子系统

每个部件有一组控制寄存器,通过一对 I/O 八位端口来管理。就象 MCGA 那样,先对一个 I/O 口写入需访问的寄存器号来指定,然后通过另一个 I/O 口对该指定的寄存器进行读写。

CRT 控制器有25个寄存器,地址设在3D4H 和3D5H;定序器有五个寄存器,地址为3C4H 和3C5H;图形控制器有九个寄存器,地址为3CEH 和3CFH;属性控制器有21个寄存器,包括16个调色板寄存器,地址为3C0H 和3C1H。

几乎所有的 VGA 控制寄存器有和 EGA 相同的功能。VGA 支持 EGA 所有可用的视频模式,也支持 MCGA 640×480 二色及 320×200、256 色图形模式。VGA 所独有的是 640×480、16 色图形模式 (BIOS 模式 12H),它和 EGA 兼容的 640×350、16 色模式类似,只是垂直分辨率更高。

和 MCGA 一样,VGA 缺省的字母数字模式时垂直分辨率为 400 线。不同的是,VGA 的 CRT 控制器可设置成显示 200 线或 350 线字母数字字符以便和 CGA 和 EGA 兼容。

3.3.5 VGA 视频模式

VGA 的 ROM BIOS 含有一系列子程序,可以通过调用软中断 INT 10H 的功能 0 来对 VGA 控制器编程,共有 24 种不同的视频模式。

每一种 VGA 控制器有一组寄存器,它们的内容确定了相应的功能。VGA ROM BIOS 中有数表,它装有针对每一种视频模式的适当寄存器值。因此,多数程序员只需调用 ROM BIOS 来选择视频模式,不用直接修改寄存器。不过,如果想要生成一种该 BIOS 不支持的视频模式,则还是需要知道在这些寄存器中装些什么值才能获得所需的视频模式。

这种情况通常出现在获得比通常 ROM BIOS 可以提供的更高分辨率图形或更多的正文显示。某些广泛应用的商品软件,比如 Microsoft Word 和 Lotus 1-2-3 可以实现上述要求。用户若想在己的应用软件上使用更高分辨率,则需要自己做一些编程工作。

1. 视频模式控制

视频模式可以用如下几种方式来区分:

- 垂直分辨率:屏幕显示的象素行数(扫描线数)
- 水平分辨率:每行的字符数或象素数
- 视频内存中的数据表示方法
- 属性解码:彩色、闪烁等等

在对 VGA 编程时,可以对垂直及水平分辨率进行多种控制。

由于硬件设计的原因,在数据表示和属性解码方面的灵活性就小多了。这样,设置替代的 VGA 视频模式最简便的方法莫过于利用 ROM BIOS 模式设置程序来建立基本视频模式,然后修改水平及垂直分辨率来产生一种新的视频模式。

2. 视频显示时序

控制显示图象的分辨率是一种时序问题。水平及垂直分辨率和视频监视器中控制电子束的 VGA 输出信号的时序有关。当然显示屏上的图象不是静止的,它由监视器电子束周期性扫描产生。见图3-13。屏幕图象每秒刷新50至70次。

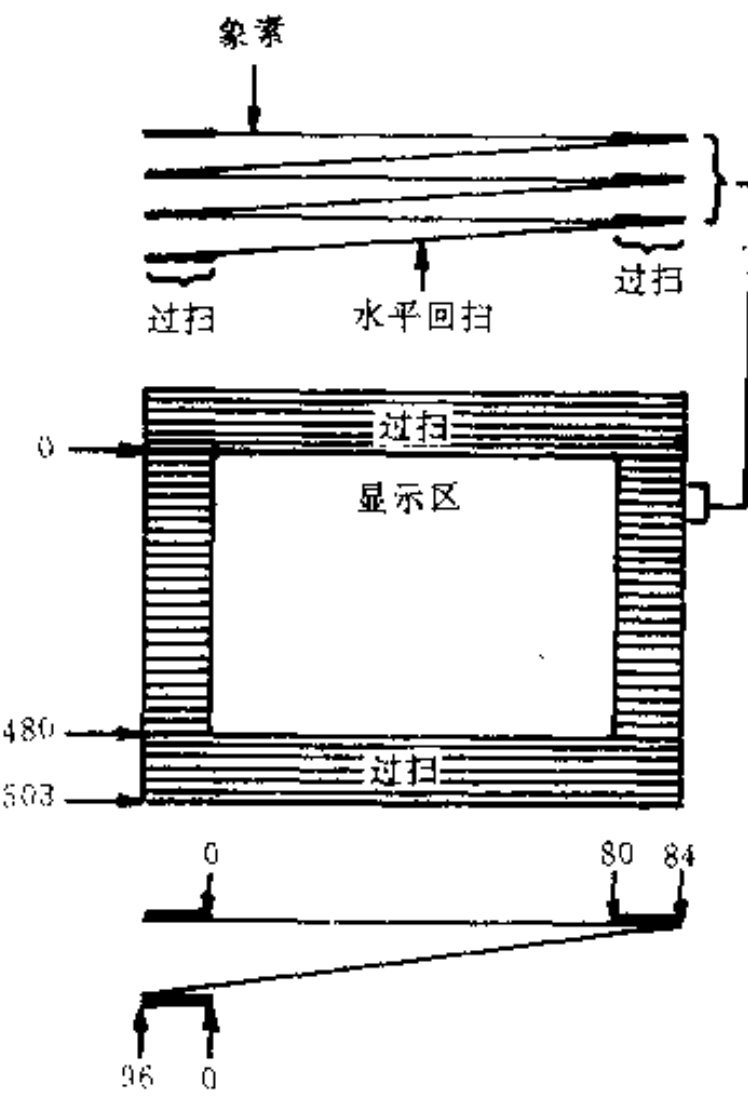


图3-13 显示屏扫描示意

扫描线上的像素由 VGA 产生的信号调制的电子束形成。在彩色监视器中,有三束对应原色的电子束,从视频显示时序的观点来

看,它们可看成一束。VGA 产生水平同步和垂直同步信号控制水平和垂直回扫。

把视频内存的数据显示出来所需时间比电子束水平及垂直扫描所需总时间短。扫描超出的这部分时间化费在水平及垂直过扫描上(overscan)。可以把屏幕过扫描区(也称边界区)赋于一种颜色而提供所显示图象的可视边框。当然过扫描区的基本作用是把屏幕图象确定在中央位置。

用户可以通过更改适当的 CRTC 寄存器,控制 VGA 的 CRT 控制器产生的水平时序信号。存在这些寄存器中的时序值由“字符时钟数”(character clock)计量。一个字符时钟对应 VGA 图形方式中的1个像素,或字母数字方式中的8/9个像素。可以设想字符时钟是一种单位时间(即在屏幕上显示一个字符所需时间)。

控制水平时序的关键参数如下:

- 水平总量。显示每条扫描线化费的总时间,包括水平回扫的时间在内。

- 水平显示时间。每条扫描线数据从视频显示缓冲区取出并显示所需字符时钟数。水平总量和水平显示时间的差就是水平过扫描量。

- 水平同步时间。水平同步脉冲开始所处的字符时钟。

控制显示图象垂直大小和屏上位置的时序参数与水平的类似。垂直时序参数通常以扫描线数来说明。象字符时钟一样,可把它看成一种单位时间(即在屏上扫描一行并回扫到下一行开端所花费的时间)。

在建立 VGA 视频模式时需要的垂直时序参数是:

- 垂直总量。一个完整刷新周期内的扫描线总数。

- 垂直显示时间。屏上数据显示的扫描线数。垂直总量和垂直显示时间的差就是垂直过扫描量。

- 垂直同步时间。垂直同步脉冲开始所处的扫描线。

3. VGA 时序关系

为了使用上述通用的时序参数来对 VGA 编程, 需要知道所用 VGA 及监视器的基本时序频率。有三种不同的控制信号频率需要考虑: 象素显示频率, 行扫描频率及整屏图象刷新频率, 它们一般称为点频、行频和帧频。

- 点频。视频子系统显示象素的频率, 也称视频带宽。该频率由称作点时钟(dot clock)的高频晶体振荡器产生。VGA 含有两种晶体振荡器(25.175MHz 和 28.322MHz), 也可以从 PS/2 50、60 或 80 系统板上的辅助视频连接器选择第三个振荡源。

- 行频。行频就是每秒显示的扫描线数。对 VGA 编程时, 可以间接用每条扫描线所含象素总数来标明行频。如果把点时钟频率除以每条扫描线的象素总数, 就得行频。

- 帧频。帧频是每秒屏幕刷新的次数。对 VGA 编程, 在每个刷新周期内显示规定的扫描线数, 也就决定了帧频。把行频除以每帧的扫描线数得到帧频。

要建立一定的 VGA 视频模式的关键在于能否对 VGA 进行编程, 使产生的时序信号在所用监视器的频率范围内。就是说, 对 VGA 编程设置视频模式时, 所选择的点频, 行频和帧频必须在所用监视器的容差内, 见表 3-11

表 3-11 一些 VGA 兼容监视器参数

监视器	点频 (MHz)	最高水平扫描频率 (kHz)	垂直扫描频率 (Hz)
IBM 8503(单色)	28	31.5	50-70
IBM 8513(彩色)	28	31.5	50-70
NEC Multisync	30	35	56-65
NFC Multisync plus	30	45	56-80
Electrohome LCM J310	30	34	47-85
Sony Mutiscan CPD J302	25	31	50-100
Princeton Ultrasync	30	35	45-120

比如说, 缺省的 ROM BIOS 视频模式 12H 是 640×480 象素 16 色。它设计用于 IBM PS/2 兼容监视器, 需要行频 31.5kHz。BIOS 在

该模式时使用25.175MHz点频,以下是时序关系的计算.

把点频除以行频得每扫描行的像素数,然后除以8(每字符时钟的像素数)得每扫描行的字符时钟数,即水平总量:

$$\text{水平总量} = (25\,175\,000 / 31\,500) / 8 = 100 \text{ 字符时钟}$$

因为每扫描行有640像素,水平显示参数为640/8即80.超出的20个字符时钟表示水平过扫和回扫所用的时间.为了把显示的图象置于屏幕中央,水平同步信号在第84个字符时钟处开始,脉冲宽12个字符时钟.这样扫描线在两端各有4个字符时钟的过扫,见图3-13。

ROM BIOS 垂直扫描参数的计算类似.BIOS 设置该模式每秒显示60帧.这在 IBM PS/2视频监视器容差的中央.以扫描线计量的垂直总量是实际行频(每秒扫描行数)对所需每秒60帧的帧频的商:

$$\text{垂直总量} = (25\,175\,000 / (100 \times 8)) / 60 = 524 \text{ 扫描行}$$

垂直显示参数是480,即该模式中像素显示的行数.相差的44行为垂直过扫和回扫的时间.BIOS 在第503扫描行后开始垂直回扫,并规定垂直同步脉冲宽为两扫描行.这样480行显示,两行回扫,42行过扫,总共524行。

3.3.6 VGA BIOS 编程

对于 VGA 常规模式,编程通过调用 BIOS 功能实现;对于非常规模式,则需要对 CRTC 编程。

1. 视频 BIOS

和 MCGA 一样,VGA 视频 BIOS 提供对所有 CGA 和 EGA 兼容的 INT 10H 功能的支持.VGA BIOS 还支持 INT 10H 功能 1AH 和 1BH,它们回送的信息是关于硬件配置和视频 BIOS 状态的,这和 MCGA 也是一致的。

VGA 视频 BIOS 通过 INT 10H 功能 1CH 提供视频状态保存/恢复能力.该功能利用用户规定地址的缓冲区保存和恢复所有控制寄存器,视频 DAC 寄存器以及从 BIOS 的 RAM 数据区得到的有

关视频的信息。这种保存 VGA 状态并依次恢复的能力可以自由切换视频模式,或对调色板和视频 DAC 寄存器编程而不会丢失先前建立的视频状态的内容。

2. 字母数字模式编程

CGA 兼容的字母数字模式编程在 VGA 和在 MCGA 上是一样的。视频内存地址首址是 B800:0000H,交替放入字符编码及其属性。视频 BIOS 字符 I/O 能力和在 IBM 其他视频子系统中一样。

与 EGA 和 MCGA 一样,VGA 可以构成显示用户定义的字母字符集。也可以对 VGA CRT 控制器编程来显示垂直尺寸不同的字符,就是说可以显示比缺省的25行更多行的字母数字字符。清单3-5简要说明如何调用视频 BIOS,利用 8×8 字符(BIOS ROM 中有它的定义)设置 80×50 字母数字模式。

;建立VGA 80×50 字母数字模式

```
mov ax,1202h      ;AH=12h(INT 10h 功能号)
                  ;AL=2(选择400线模式)
mov bl,30h        ;BL=字母数字扫描线选择
int 10h
mov ax,3          ;调用视频 BIOS 选择模式
int 10h
```

;把视频 BIOS 8×8 字符装入字母数字字符发生器

```
mov ax,1112h      ;AH=INT 10h 功能号
                  ;AL= $8 \times 8$ 字符集装载
mov bl,0          ;BL=装载块
int 10h           ;把 $8 \times 8$ 字符装入内存
```

清单3-5 VGA 80×50 字母数字模式

3. 图形模式编程

在和 CGA、MCGA 兼容的图形模式中,读写象素的程序一样能运行。在 EGA 图形模式(320×200 、16色, 640×200 、16色 及 640×350 、16色)中运行的程序也可在 VGA 相应的模式上运行。VGA 独有的是 640×480 、16色模式,不过它和 EGA 640×350 、16色模式几乎一样。

在和 EGA 兼容的模式中,显示缓冲区有四个位面,共享从 A000:0000H 开始的相同地址范围。每当 CPU 执行读写指令时,数据字节对位面是并行传送的。

这种有限的并行处理是由 VGA 图形控制器实现的,它有一组四个八位的内部寄存器或锁存器。每当 CPU 执行一条指令从显示缓冲区一个地址中读取时,图形控制器把每个位面的该指定地址的内容复制到锁存器中。比如,CPU 执行 MOV reg,mem 指令,图形控制器从位面复制四字节数据到锁存器中。

当 CPU 执行写指令时,过程就与上述相反。在图形模式中更新显示缓冲区的操作顺序是 CPU 先读后写。这可以是两条 MOV 指令,也可以用诸如 MOVS 这样的单指令。

象素由各位面上同一地址的一组对应位表示。因为有四个位面,所以一个象素由四位表示,也就是说能显示的不同色彩数为16色。需要注意的是,图形控制器的锁存器中的内容是8个相邻的象素值(一个象素一位,一共四个位面,所以迭成四位)。

VGA 和 EGA 相同的还有,图形模式编程的关键是控制图形控制器处理读自/写往位面的数据字节(象素值)的方法。在 CGA 上,两种图形控制器读模式和四种写模式决定了 CPU 读写时图形控制器的操作。

4. 图形控制器读模式

两种图形控制器读模式和在 EGA 上执行的相同。在读模式0中,每当由 CPU 读操作装载锁存器时,四个中的一个锁存器中的值被复制到 CPU 中。在模式1时,锁存器中八个象素值和存储在图形控制器的彩色比较寄存器的值进行比较,相同的话,该位置1,形成一个字节结果返回给 CPU。读模式0在把数据从位面传到系统内存时很有用,因为可以分别访问每个位面。利用读模式1进行图形操作,比如区域填充也很方便。因为这时得扫描显示缓冲区,让象素符合事先确定的值。

5. 图形控制器写模式

图形控制器有四种写模式,其中每一种都是用来简化一定种类的编程任务的。写模式0是视频 BIOS 程序用得最多的一种。在写模式0时,图形控制器把八个已锁存的象素值和 CPU 写的数据字节或在图形控制器置位/复位寄存器中存贮象素值组合起来。该图形控制器可以对象素值进行与、或及异或操作,也可以用 CPU 数据置换或置位/复位。在操作中,图形控制器位屏蔽寄存器可以控制八个锁存的象素值中的哪一个需要更新,哪一个保持不变。

清单3.6说明如何利用写模式0更新640×480、16色模式中一个象素的值。

程序首先计算象素所占视频内存的地址以及位屏蔽寄存器的位屏蔽值。然后设置图形控制器的各个寄存器,选择写模式0,把所需象素值存入置位/复位寄存器,允许对所有四个位面置位/复位,把位屏蔽值放入位屏蔽寄存器。此后 OR 指令更新位面。最后图形控制器的各寄存器根据视频 BIOS 缺省值修改,使随后的视频 BIOS 程序正常运行。

很明显,大部分操作是有关设置图形控制器的,只有一条指令真正是更新象素的。注意执行 OR 指令时所发生的一连串事件:首先,CPU 读操作,锁存器装入在指定地址上的八个象素值。然后 CPU 执行寄存器和从图形控制器读来的值做逻辑或,用结果进行 CPU 写操作。

CPU 写操作要通过图形控制器的置位/复位寄存器中的象素值来更新锁存器。位屏蔽寄存器的值规定了在 CPU 写操作、锁存数值复制到位面时,八位被锁存的象素值中的哪一位用置位/复位寄存器的相应位替代。

在写模式1时,四个锁存器的内容直接复制到各位面上。因此写模式1在填充一定彩色或花样时很有用。

在写模式2时,锁存器中的各个象素用 CPU 数据字节而不是用置位/复位寄存器中规定的象素值来更新。因此,模式2可以象模式0一样容易地更新显示缓冲器中单个象素的值。

SP2 PROC near ;call with;	AX=Y	坐标
:	BX=X	坐标
:	CL	象素值
;计算象素在视频内存中的地址		
push cx	;压入象素值	
mov cx,bxand		
mov ch,10000000b		
shr ch,cl	;CH=象素位屏散	
mov dx,80		
mul dx	;AX=Y*80	
mov cl,3		
shr bx,cl	;BX=X/8	
add bx,ax	;BX=Y*80+X/8	
mov ax,0A000h		
mov ds,ax	;DS:BX→象素地址	
;设置图形控制器		
mov dx,3CEh		
mov ax,0005		
out dx,ax	;设写模式0	
pop ax	;弹出象素值	
mov ah,al		
mov al,0		
out dx,ax	;设置位/复位寄存器	
mov ax,0F01h		
out dx,ax	;设置允许置位/复位寄存器	
mov ah,ch		
mov ah,8		
out dx,ax	;设位屏蔽寄存器	
;更新象素		
or [bx],al	;CPU 读时更新锁存器	
	;CPU 写时更新位平面	

;恢复缺省图形控制器寄存器数值

```
mov     ax,0000
```

```
out     dx,ax      ;缺省置位/复位数值
```

```
mov     ax,0001
```

```
out     dx,ax      ;缺省允许置位/复位数值
```

```
mov     ax,0FF08h
```

```
out     dx,ax      ;缺省位屏蔽数值
```

```
ret
```

```
SP2     ENDP
```

清单3-6 640×480 16色模式

写模式3和模式0相仿,只是其位屏蔽是 CPU 写的数据字节和位屏蔽寄存器的值进行逻辑与得来的。这样无需对位屏蔽寄存器编程就能改变位屏蔽格式。EGA 不支持写模式3,如果设计的程序在 EGA 和 VGA 都要能使用,则应避免使用写模式3。

6. 视频 DAC 编程

用 VGA 视频数模转换器(DAC)要比 MCGA 复杂,因为 VGA 的属性控制器控制对视频 DAC 的访问。VGA 在字母数字模式和16色图形模式中并不限制只使用前16个视频 DAC 彩色寄存器,对属性控制器编程可以对16组共计256个视频 DAC 彩色寄存器进行寻址。

在 VGA 上,每四位属性值(字母数字模式时)或四位像素值(图形模式时)由属性控制器处理,它利用该值来选择16个调色板寄存器中的一个(见图3-14)

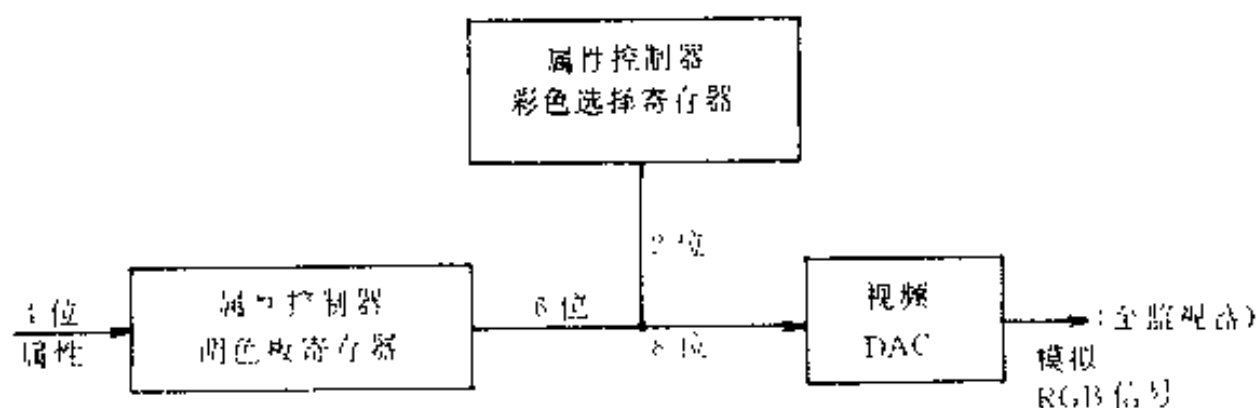


图3-14 VGA 彩色控制

每个调色板寄存器含有一个六位值,它和属性控制器的彩色选择寄存器一起构成一个八位值送到视频 DAC。视频 DAC 然后用该八位值来选择 256 彩色寄存器中的一个。每个寄存器有 18 位 RGB 值。这样视频子系统把四位属性解码为一组模拟 RGB 三色值输出给监视器。

在缺省情况下,视频 BIOS 保持 EGA 兼容性。方法是在初始化时把属性控制器调色板寄存器置成和 EGA 上一样的六位值,并把前 64 视频 DAC 寄存器置成和 EGA 同样的 RGB 值,以便产生可在 EGA 上使用的相同 64 彩色。

通过对属性控制器彩色选择寄存器的编程,可以使用余下的三组 64 视频 DAC 寄存器。视频 BIOS INT 10H 功能 10H 支持该项工作(在 IBM 技术文件中视频 DAC 彩色寄存组称为色彩页 color page)。

清单 3-7 中,用了视频 BIOS 把第一组 64 视频 DAC 寄存器中的内容复制到第二组中。然后调用 BIOS 灰度功能把第二组彩色寄存器的值换成它们灰度等价值。此时,可以再一次调用 INT 10H 的功能 10H 或是再次选择第一组 64 彩色寄存器组中的缺省彩色值,或是第二组中的灰度值。

3.3.7 VGA 寄存器编程

通过 I/O 端口(见表 3-12)可以对 VGA 控制器直接编程。

表 3-12 用于 VGA 控制的 I/O 端口

端 口	功 能	存取特性
3C0	属性控制器	读/写
3C2	杂项输出	只写
3C4, 3C5	定序器	读/写
3CC	杂项输出	只读
3CE/3CF	图形控制器	读/写
3D4, 3D5*	CR1 控制器	读/写

* 在模式 7 和 011H 时为 3B4/3B5

```

Buffer    db      3 * 64 dup(?)      ;彩色寄存器值缓冲区
;把第  组64视频 DAC 彩色寄存器内容复制到二组64寄存器中
        mov     ax,1017h              ;AH=10h(视频 BIOS 功能号)
                                           ;AL=17H(读彩色寄存器组)
        mov     bx,0                  ;BX=第一个寄存器
        mov     cx,64                 ;CX=要读的寄存器数
        mov     dx,segBuffer
        mov     es,dx
        mov     dx,offsetBuffer      ;ES;DX→缓冲区地址
        int     10h                  ;把寄存器内容读入缓冲区
        mov     ax,1012h              ;AH=10h
                                           ;AL=12h(设定彩色寄存器组)
        mov     bx,64                 ;BX=设定的第一个寄存器
        mov     cx,64                 ;CX=设定寄存器的个数
        mov     dx,segBuffer
        mov     es,dx
        mov     dx,offset Buffer      ;ES;DX→缓冲区地址
        int     10h                  ;设定寄存器
;灰度计算
        mov     ax,101Bh              ;AH=10h
                                           ;AL=1Bh(灰度计算)
        mov     bx,64                 ;BX=第一个做灰度计算的寄存器
        mov     cx,64                 ;CX=寄存器数
        int     10h
;使用灰度值
        mov     ax,1013h              ;AH=10h
                                           ;AL=13h(选择视频 DAC 彩色页)
        mov     bx,101h              ;BL=1(选择指定彩色页)
        int     10h                  ;BH=1(彩色页指定值)
;使用缺省彩色值
        mov     ax,1013h
        mov     bx,001h              ;BH=0(彩色页指定为0页)
        int     10h

```

清单3-7 视频 DAC 彩色寄存器的使用

访问这些端口需要通过汇编语言中的 IN 和 OUT 指令或高级语言中相应的语句。访问视频 BIOS 需要执行中断 10H, 这可以直接用汇编语言, 或通过高级语言比如 Microsoft C 中的 INT86() 功能。

1. CRTC 编程

VGA 对水平及垂直时序参数的大部分控制是通过 CRTC 实现的。CRTC 控制送往监视器水平及垂直时序信号, 并把时序信号和从显示缓冲区取出、经显示电路处理的数据配合起来。对 CRTC 相应寄存器内容进行编程, 就可以控制上述功能。CRTC 各寄存器见表 3-13。

要更新一个 CRTC 寄存器时, 首先向 I/O 口 3D4h 写该寄存器的号, 然后再向 3D5h 写该寄存器的新值, 见清单 3-8。

```
cli                      ; 禁中断
mov     al, RegNumber    ; AL = 寄存器号
mov     dx, 3D4h
out     dx, al           ; 选择寄存器
mov     al, RegValue     ; AL = 寄存器新值
inc     dx
out     dx, al           ; 向 3D5H 口写新值
sti                      ; 开中断
```

清单 3-8 更新 CRTC 寄存器

利用一些技巧, 可以用单一的 16 位口写操作来实现同样的目的, 见清单 3-9。

```
mov     al, RegNumber    ; AL = 寄存器号
mov     ah, RegValue     ; AH = 寄存器新值
mov     dx, 3D4h
out     dx, ax           ; 对 13D4h/3D5h 写操作
```

清单 3-9 更新 CRTC 的 16 位操作

要注意清中断, 以免在端口写操作之间因硬件中断而打断程序。此外, 在用视频 BIOS 模式 7 或 0Fh 时, 要用端口 3B4/3B5。它们模拟单色显示适配器的端口, 这样在同一计算机中既可工作在

VGA,也可工作在其它彩色视频子系统。

表3-13 用于视频模式编程的 VGA CRT 控制器中的各寄存器

寄存器号	名 称	内 容
0	水平总量	水平总量(-5)
1	水平显示使能结束	所显数据的字符时钟数(-1)
2	水平空白开始	水平空白开始时的字符时钟
3	水平空白结束	水平空白脉冲结束(仅是位4~6)
4	水平回扫开始	水平同步开始时的字符时钟
5	水平回扫结束	水平同步脉冲结束(仅是位4~6)
6	垂直总量	垂直总量(10位值的位7~0)
7	输出	位0:垂直总量位8 1:垂直显示使能结束位8 2:垂直回扫开始位8 3:垂直空白开始位8 5:垂直总量位9 6:垂直显示使能结束位9 7:垂直回扫开始位9
9	最大扫描行	位1-0:每个字符的扫描行数(-1) 位5:垂直空白开始位9
10h	垂直回扫开始	垂直同步脉冲开始时的扫描行 (10位值的位7~0)
11h	垂直回扫结束	位7:写保护(CRTC寄存器0~7) 位3-0:垂直同步结束时的扫描行
12h	垂直显示使能结束	显示视频数据的扫描行数 (10位值的位7~0)
13h	偏移量	以字计的視頻缓冲区数据的逻辑 行宽度
15h	垂直空白开始	垂直空白开始时的扫描行(-1) (10位值的位7~0)
16h	垂直空白结束	垂直空白结束时的扫描行 (位7~0)

和以往的 IBM 视频子系统 中的控制寄存器不同,VGA 的控制寄存器均可读写,这使得在修改 CRTC 寄存器前就可以把它的当

前状态保存起来。清单3-10说明了VGA CRTC寄存器可以方便地读写。

```
cli                ; 禁中断
mov     al,RegNumber ; AL = 寄存器号
mov     dx,3D4h
out     dx,al       ; 向3D4h写
inc     dx
in      al,dx        ; AL = 寄存器值
shl
```

清单3-10 VGA CRTC 寄存器可读可写

2. 定序器编程

VGA 定序器通过编程可以确定字符时钟是每8个还是每9个点时钟显示一个字符。也就是说,根据对定序器的编程,视频显示缓冲区中的数据每一字节可以显示成八个或九个水平像素。

在缺省的VGA字母数字模式中,屏幕上每个字符水平显示为九个像素。在VGA兼容的350线字母数字模式和图形模式中,系统把定序器编程为每个字符水平显示八个像素。这第九个像素可增加所显示正文的清晰度,而省掉它则可以在屏幕上显示更多的字符。

访问定序器寄存器需要通过I/O端口3C4h和3C5h,见表3-14。

表3-14 VGA 定序器寄存器

寄存器号	名称	内容
0	复位	位1:同步复位
1	时钟模式	位0:1=8像素字符 0=9像素字符

方法和访问CRTC类似,需要注意的是,当选择一个新的点时钟或更改字符像素数时,必须把复位寄存器的位1置0使定序器暂时复位,事后恢复。

字符形式。

```
cli                      ;禁中断
mov     dx,3C4h
mov     ax,0100h        ;AH—复位寄存器值,位1=0,位0=0
out     dx,ax           ;定序器同步复位
mov     al,1            ;AL=时钟模式寄存器号
out     dx,al
inc     dx
in      al,dx           ;AL—时钟模式寄存器值
dec     dx
or      al,1
mov     ah,al           ;AH—时钟模式寄存器新值
mov     al,1
out     dx,ax           ;更新时钟模式寄存器
mov     ax,0300h        ;AH—复位寄存器的值,位1=1,位0=1
out     dx,ax
sti
```

清单3-11 定序器编程

3. 杂项输出寄存器编程

杂项输出寄存器的位2和位3都是0时,规定使用的点时钟频率是25.175MHz;位2是0而位3是1时,则为28.322MHz。要修改该寄存器内容时,先从I/O口3CCh读入当前值,把位2和3加以屏蔽,然后从3C2h写出。做该项工作时,应该象前一个程序例子中那样使定序器暂时复位。

在缺省的字母数字模式中,视频BIOS把VGA置成25行显示。比如说在400线字母数字模式(加电缺省模式)中,每个字符高为16扫描线。如果修改字符扫描线数,字符扫描线数减少、则每屏容纳的字符行就比25行多。

CRTC寄存器09是最大扫描线寄存器,它的位0到位4控制字符显示的高度。该位段中的值比字符高度的扫描线数少1。因此在缺省的字母数字模式中位0到4的值是01111B(0FH)。如果把该值改成00111B(07H),CRTC显示只有8扫描线高的字符,这样可得到

每屏50行字符的视频模式。

虽然可以直接修改扫描行寄存器,只要有可能最好的办法还是通过 BIOS 来做。BIOS 提供相当的灵活性来设置字母数字模式中字符显示高度,同时还可选择适当的字符集。比如:

```
mov ax,1112h    ;AH=11h(BIOS 功能号)
                ;AL=12h(子功能号)

mov bl,0
int 10h
```

同样实现高为8扫描线的字符显示。BIOS 把字符集装入硬件字符发生器,完成对 CRTC 的编程,并以新的字符行数来修改全局数据区。

视频 BIOS 把字符行数记录在全局数据区0040:0081h 处,但 PC-DOS 的清屏命令 CLS 总以为清除的是25行数据。如果把 CRTC 编程显示50行数据 CLS 仅清屏幕的上半部,因此需要编写自己的清屏命令。参见清单3-12。

IBM 用在 VGA 子系统集中的集成电路是专用的。IBM 的竞争对手被迫对 VGA 硬件做反工程的工作,用他们自己的产品发挥同样的能力。这就是说 VGA 兼容产品并不一定要和 IBM VGA 硬件完全一样。它们可能在两方面有所不同:控制寄存器中的数值和可用的点时钟频率。

以上叙述的寄存器编程技术不能适用所有的 VGA 兼容系统。比如,当对 Video Seven 的 VEGA VGA 上的 CRTC 编程时,会发现许多 CRTC 寄存器需要和 IBM VGA 不同的值。包括 Paradise VGA Plus 在内的其他适配器,寄存器值和 IBM VGA 一致,因此对它们编程要容易得多。

VGA 兼容产品的厂家通常实行更高的分辨率,采用更高频率的点时钟的非 IBM 视频模式。比如,Paradise VGA Plus 在132列字母数字模式及800×600像素图形模式时采用36.0MHz 点时钟。由于采用更高频率的点时钟,导致比 IBM VGA 在28.322MHz 点时

钟时较高的水平和垂直扫描频率。扫描频率接近大部分监视器容差的中间,由于垂直扫描频率增高,闪烁减弱。

```

CodeSeg  SEGMENT    byte
            Assume    CS:CodeSeg    SS:Stackseg
VGACLS    PROC        far
            mov     ah,0Fn           ;AH=INT 10H 功能号
            int     10h             ;取视频状态
; AH 现为所显示字符的列数,AL 现为视频模式,BH 现为当前视频页
            push    bx              ;保存 BX,AX
            push    ax
            mov     ax,1130h        ;AH=11H(INT10H 功能号)
                                     ;AL=30H(子功能号)
            int     10h             ;取字符发生器信息:
                                     ;DL=显示字符行数-1
            pop     ax              ;恢复 AX
            mov     dh,dl           ;DH=屏上最后一行
            mov     dl,ah
            dec     dl              ;DL=屏上最后一列
            sub     cx,cx           ;CL=0(屏上第一行)
                                     ;CL=0(屏上第一列)
            mov     bh,7            ;BH=缺省字母数字属性
            cmp     al,7
            jnc     L01             ;如是字母数字模式则转移
            xor     bh,bh           ;BH=缺省图形属性
L01        mov     ax,0600h        ;AH=6(INT 10h 功能号)
                                     ;AL=0(滚动的行数)
            int     10h             ;L 卷(清屏)
            pop     bx              ;BH=当前视频页
            xor     dx,dx           ;DH,DL=0(新光标位置)
            mov     ah,2
            int     10h             ;设光标位置为左上角
            mov     ax,1C00h
            int     21h             ;调用 DOS 结束程序
VGACLS    ENDP

```

Code Seg	LNDS	
Stack Seg	SEGMENT	stack
Stack Seg	DB	800h dup(?)
	ENDS	VGACLS
	END	

清单3-12 VGA 清屏

第四章 图形标准的进展

VGA 是 MS-DOS 和 OS/2 为基础的计算机的较新图形标准,支持 640×480 象素的分辨率,可用 256 种颜色。这比先前的图形标准强多了,但依然满足不了日益增长的需要。

对高分辨率的要求,一部分来自 CAD 及电子排版等专门应用,不过大部分是因为不断增长的对窗口界面的要求而产生的。比如 OS/2 下的 Presentation Manager 和 MS-DOS 下的 Microsoft Window。虽然它们可以在较低分辨率监视器上使用,但在高分辨率下效果显然好得多。

在提高分辨率和色彩等图形功能时,标准方面的问题日益突出。几乎所有的图形适配器制造厂家都加入了自己特有的正文和图形模式(通过硬件和 BIOS 扩充)。从正文方面来说,有些模式已达 132×60 字符。从图形方面来说,有 800×600 象素 16 色和 256 色选择,甚至 1024×768 象素以及更高分辨率。

结果,软件公司的产品图形软件只能有限地支持一些个别图形卡的专门扩展模式。图形卡的制造厂不得不扩大他们自己的编程和技术支持,为每一个图形软件包配软件驱动程序以便解决用户的安装问题。

视频电子标准协会 VESA (Video Electronics Standards Association) 推荐标准的 BIOS 和编程接口用于扩展的图形模式。VESA 的 VGA BIOS 扩展提案 2.0 版称为 Super VGA,在 1989 年 4 月发表,而后 IBM 则推出了 XGA。

为了实现高性能的图形功能,由专门的图形协处理器来完成已成为潮流。因此迫切需要界面标准化。1989 年完成的高分辨率

图形(1024×768 象素及以上)标准有 TIGA 和 8514/A。

德克萨斯仪器公司(TI)支持 TIGA(Texas Instruments Graphics Architecture)作为用 Intel 微处理器的计算机和使用 TI340X0 系列图形协处理器芯片的图形卡之间的标准接口。TI 开放软件标准,促进并销售软件开发包,帮助 TIGA 兼容产品产出并进入市场。

IBM 把 8514/A 设计成一个封闭的硬件产品,从未发表其硬件规格说明。而兼容产品公司把他们的产品做得和 IBM 版本寄存器级兼容,西方数字公司等(Western Digital, Headland Technologies, Chip & Technologies)推行他们 8514/A 图形适配器的各自版本。

TIGA 和 8514/A 都想作为软件标准,但是 8514/A 芯片组的第三方制造商认为至少有一些软件开发者要直接和硬件打交道,因此他们考虑保持和 8514/A 寄存器级兼容。TIGA 则在设计开发时就确定程序员不和硬件打交道。

4.1 Super VGA

4.1.1 Super VGA 标准

Super VGA 是标准 IBM VGA 的超集,它将隔离硬件的差别,以与硬件无关的方式提供信息。这就是说,最后将可以编写图形软件而无需专门的代码用于各种特殊的芯片,也不用识别所用的是什么适配器。

VESA 提出标准来确定视频环境、编程支持兼容性、模式号及 BIOS 功能。该视频模式号为二进制 15 位长,而 VGA 为 7 位,从 00h~13h。厂家已建立的扩展模式从 14h~7Fh。由于位 7 是清除视频内存的标志,所以 80h~FFh 不用。除了模式 6Ah 外,所有其它 VESA 模式和原先等同或另加 100h。参见表 4-1

表 4-1 VESA 模式

模式号	分辨率	色彩数
6Ah	800×600	16
100h	640×480	256
101h	800×600	256

扩展的 BIOS 支持使用 INT 10H 的扩展功能 4Fh。功能调用把状态返送回 AX 寄存器,表明支持与否、成功与否。参见表 4-2。

表 4-2 VESA 功能

功能号	目的
00h	返送 Super VGA 信息。该功能返送一个缓冲区的指针,其中装有所支持的 Super VGA 性能和其他指针。
01h	返送 Super VGA 模式信息。该功能返送一个详情表的指针,其中有大小,属性,特定 Super VGA 模式的分辨率。
02h	设置 Super VGA 视频模式。该功能设置可用的所需模式,否则保持原环境不变。
03h	返送当前视频模式。
04h	保存或恢复 Super VGA 视频状态。
05h	设置 CPU 视频内存窗口。该功能允许直接访问视频内存的硬件分页寄存器。

所涉及的视频适配器,除了这些功能之外,其它的方面有:

- 扩展视频内存映射,通过尽量和硬件细节隔离,简化窗口软件。
- 外部调色板控制 6~8 位 D/A 转换,产生与 256 000 色或 16M 色调色板统一的接口。
- CRT 可变的起始地址,这对动画技术很有意义。
- 标准显示时序参数,这样的监视器产品不用调节就可同步,并提供各种分辨率合适的图象尺寸。

对于最终用户来说,人们见不到种种技术处理,反映到表面上来的是:扩展图形模式需要较长时间,各特定适配器系列采用各自驱动程序。

一旦接受了 VESA 标准,人们看到的软件驱动程序更少但功能更强。此外,切换到一种新的视频显示适配器时,通常不用一整套各种各样的驱动程序,因为以 VESA 标准写的驱动程序可用于各种 VESA 适配器。

VESA 标准可能解决不兼容的问题,但是较高分辨率又会产生其它问题。分辨率提高意味着象素更多,更新一屏的操作量更大,因此响应也更慢。

4. 77MHz 在支持 EGA 图形时速度嫌慢。而对基本的 8MHz AT 来说,640×480 象素的 VGA 或许是可接受图形性能的最好分辨率。12MHz AT 可支持 800×600 象素 VGA,而 1024×768 象素则需要 16MHz AT 或 80386 机才可能有象样的图形模式性能。

不过,这并不是说一切限制都得归咎于处理器,16 位 VGA 比 8 位 VGA 快,它提供的硬件和软件可以处理 16 位图形存取。

由于大量运行的图形软件在 640×480 象素,处理器处理的象素是 CGA 的 2.4 倍,更高分辨率的处理速度会明显受到影响。见表 4-3

表 4-3 分辨率对图形处理的影响

视频模式	分辨率	象素数	CGA 百分数
CGA	640×200	128 000	100
EGA	640×350	224 000	175
VGA	640×480	307 200	240
Super VGA	800×600	480 000	375
8514/A	1024×768	786 432	614

除了象素数之外,影响处理速度的另外两个因素是图形适配

器存储器访问时间,改变单一像素所需操作次数。

EGA 在访问存储器这一点上缺点突出。6 个时钟的存储器访问周期中的五个时钟必须专门用来刷新显示,只有一个时钟周期用于更新,这是 EGA 显示慢的一个主要原因。

VGA 的问题不同,它在访问存储器时速度加快了,但是较大的像素阵列需要五个 I/O 命令来完成一次更新。原因是要以 64KB 地址空间对 256KB 或 512KB 分段进行访问。在 16 色模式,VGA 能一次更新两个像素,但在 256 色模式,必须分别访问各像素。

许多 VGA 适配器并没有用通往 AT 总线的 16 位接口,依然采用 8 位寄存器,再加上较慢的 BIOS 访问,这就更加影响了速度。

在 800×600 像素以上的高分辨率领域,需要更进一步的性能,需要专用图形处理器硬件的帮助,比如 Intel 的 82786、TI 的 34010、34020 以及支持 IBM 8514/A 的芯片等等。与此相应的是需要改写图形驱动程序,或许要重新设计接口。在处理画线或填充重复性的操作时,图形处理器的能力强,主处理器和图形处理器的关系也比较简单——只是传递参数,对复杂形状图形处理器可能比主 CPU 处理得要慢,这时处理的情况要复杂一些。

4.1.2 TVGA

TVGA 是由 Trident 公司开发的图形卡,它提供与 VGA 寄存器级兼容能力,而且大大增加了图形、文本模式,支持 $256K \times 4$ DRAM,等离子体显示控制,它可以模拟 VGA、EGA、CGA 和 MDA。在图形方式时,VGA 在 256 色情况下仅支持 320×200 像素分辨率,而 TVGA 则支持 256 色下的 640×400 、 640×480 、 800×600 和 1024×768 模式。

1. TVGA 结构

TVGA 主要由:定序器、CRT 控制器、图形控制器,属性控制器、DAC 支持逻辑、主总线接口及显示寄存器接口等七部分组成,它们之间的逻辑关系如图 1-1 所示。

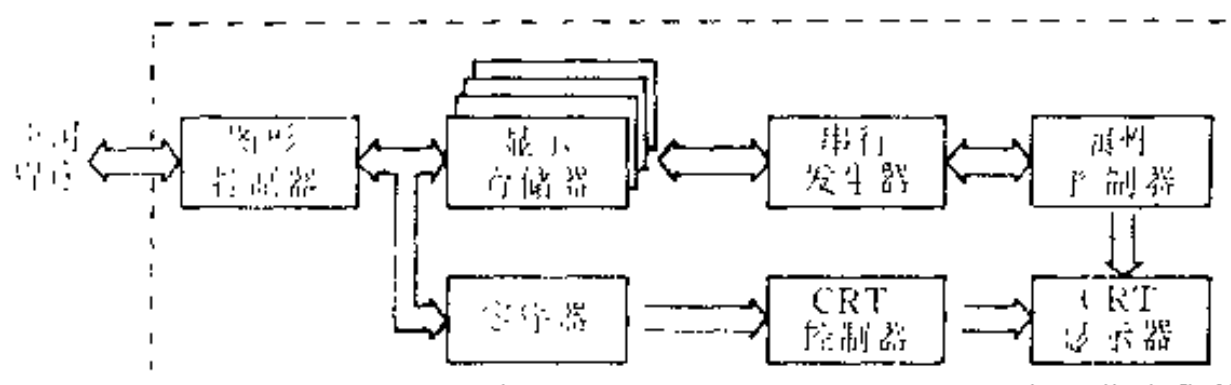


图 4-1 TVGA 框图

• 定序器

定序器为 DRAM 接口提供基本的存储器时序,为 CRT 提供字符时钟,并执行某些存储器地址解码。它通过五个可以读写的寄存器进行控制,它们有两个 I/O 地址(3C4 和 3C5)。3C4 是序号寄存器,用于选择工作寄存器。数据则输出到 I/O 地址为 3C5 的寄存器。表 4-4 列出了定序器寄存器和它们的序号。

表 4-4 定序器寄存器

序号(3C4)	定序器寄存器(3C5)
00	重置寄存器
01	时钟模式
02	彩色页面写允许
03	字符发生器选择
04	存储模式

• CRT 控制器

CRT 控制器通过产生显示器光栅的同步信号等来控制,它也可以定义屏幕上显示数据的格式。CRT 控制器使用两个 I/O 地址,一个是序号寄存器,用于选择 24 个 CRT 控制器中的内部工作寄存器;另一个用于选定寄存器数据的读写。

• 图形控制器

图形控制器位于显示存储器与系统处理器之间的数据通道中。在缺省状态下,图形控制器是透明的:数据在处理器与显示存储器之间直接通过。其它情况下,图形控制器可以对处理器读写的数据进行逻辑操作,为图形算法提供硬件支持。

同样,图形控制器有两个 I/O 地址:3CE 和 3CF,前者是序号寄存器,用于选择图形控制器各工作寄存器,后者用于选定寄存器的数据读写。

• 属性控制寄存器

属性控制寄存器是用来控制诸如闪烁、下划线这样的显示属性的。在有些模式下,只需要控制颜色。

属性控制器内含有 20 个输出,使用一个 I/O 地址。序号和数据寄存器共用 3C0。

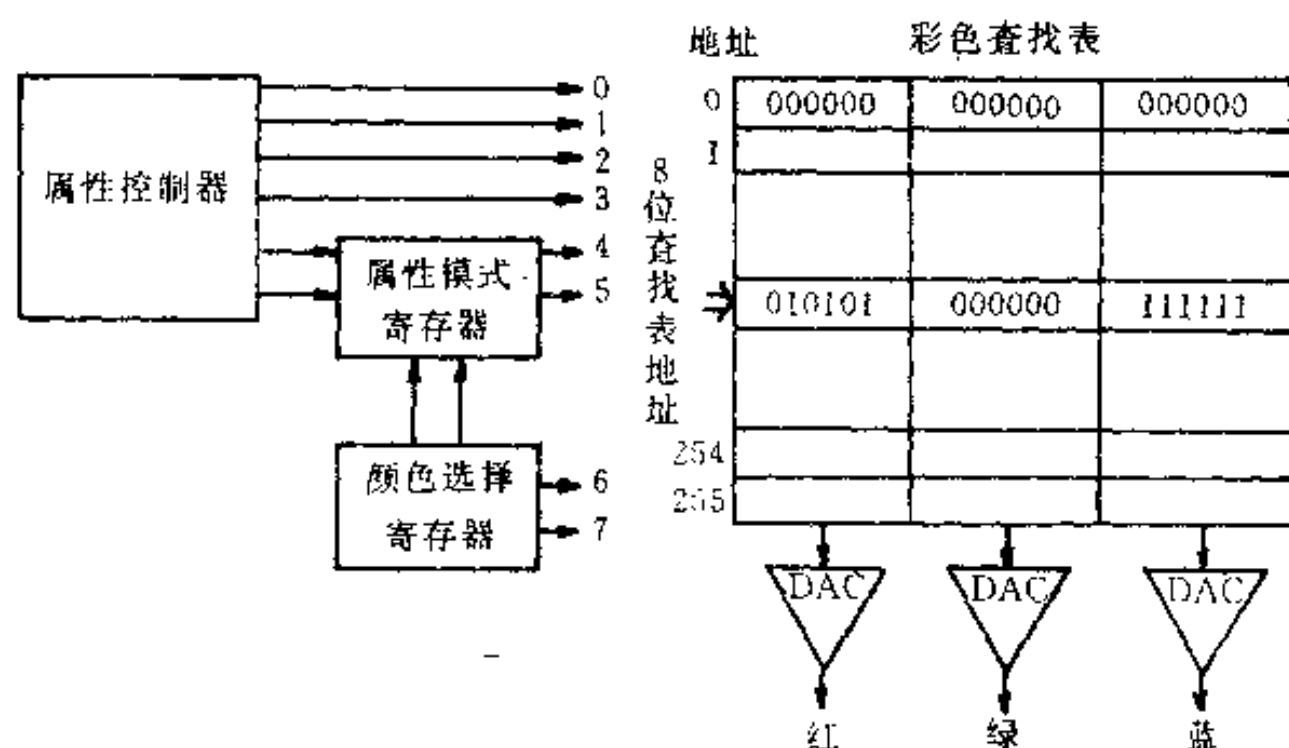


图 4-2 色彩形成示意图

在图形模式下,属性控制器把内存中的数据转化成彩色查找表的内址,并由视频 DAC 转换成模拟电平来驱动 TVGA 模拟显示器。

• 视频 DAC

TVGA 内部有一张彩色查找表,编号从 0 到 255。它相当于一块彩色调色板,其编号就是色彩编号。

TVGA 256 色模式时,显示缓冲区中每一字节都存放着相应彩色查找表编号(色彩号)。查找表中的每一项由 18 位组成,分成三部分、各六位,分别代表红、绿、蓝的值。查找表中每一项是可以设置的,因此可有 2^{18} 种颜色可以选择,由于查找表最多只有 256 项,所以同时显示的颜色只有 256 种。视频 DAC 和彩色查找表的逻辑关系见图 4-2

用于访问视频 DAC 的寄存器有五个,见表 4-5

表 4-5 与 DAC 有关的寄存器

I/O 地址	寄存器
3C6	点屏蔽寄存器
3C7	DAC 状态寄存器
3C7	查找表读序号(只读)
3C8	查找表写序号(只写)
3C9	查找表数据寄存器

选择查找表中 256 个内部色彩寄存器用了两个独立的寄存器,读序号寄存器仅用于读查找表数据,写序号寄存器仅用于写查找表数据。彩色寄存器宽 18 位,把 8 位序号写入 3C8 写序号寄存器,然后写三个 6 位数到数据寄存器(3C9),这就是彩色寄存器值。每写三个字节序号寄存器自动加一,因此不必重复设置序号便可给一组彩色寄存器赋值。同样,向读序号寄存器(3C7)写入 8 位序号,然后从 3C9 数据寄存器中读三个 6 位值,便得一个彩色寄存器值。每读三个字节,序号寄存器自动加 1,因此不必重置序号便可读出一组彩色寄存器值。

读 DAC 状态寄存器(3C7)可以知道彩色查找表当前可以读还是写。 D_0, D_1 位上的 00 值表示处于读模式,01 值表示处于写模式。

• TVGA 寄存器

TVGA 有很多寄存器,大部分是可读的,分为内部控制寄存器和外部控制寄存器。

2. TVGA 存储模式

TVGA 共有 41 种显示模式,其中 0~13 是和 VGA 兼容的,50~62 是 TVGA 扩展的。这里介绍 TVGA256 色图形模式,包括模式 5C、5D、5E 和 62 等。在这几种模式中,调色板的设置和 VGA 的 13 模式相同,都是通过对端口 3C7、3C8 和 3C9 的读写达到重置彩色查找表的目的。这几种显示模式显示缓冲区的起始地址都是 A0000,而所需缓存容量都大于 VGA 基本的 64KB,因此 TVGA 在控制卡上增加了一部缓存,它们通过 3C4、3C5 来存取。

• TVGA 存储器映

射图

在 256 色状态下,存储器中每个字节对应屏幕上的一个象素点,其中的值代表该点在彩色查找表中的索引号。但一幅 800×600 象素的画面大约需要 300KB 的缓存。原 VGA 中用 20 位来表示象素点在缓存中的地址已不够用了(这 20 位地址范围为 A0000~AFFFF 只有 64KB),需要增加三位,相当于设

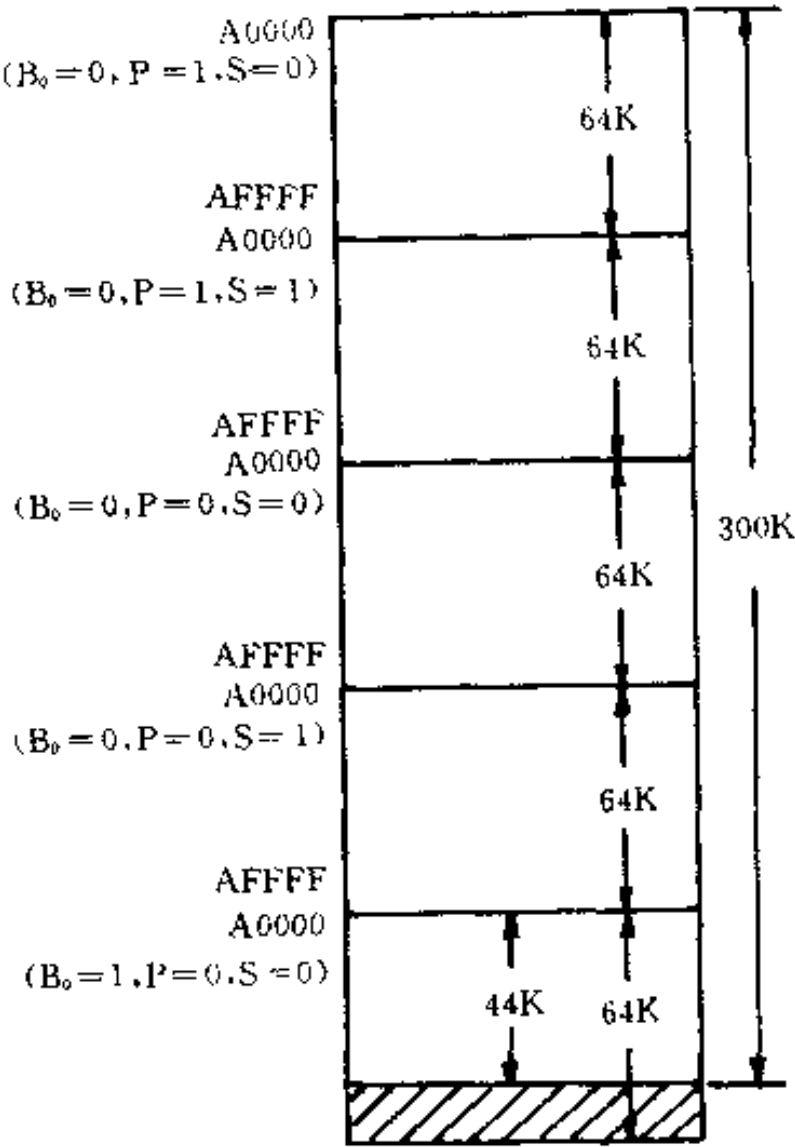


图 4-3 模式 5E(800×600×256)显示缓冲区

置三个开关,分别控制数据传送到不同的几个 64KB 缓存中去,见图 4-3

其中最上面的 64K 由原 VGA 提供,下面的各块由 TVGA 扩充, B_0 、 P 、 S 代表三位开关,这是 800×600 像素 256 色的模式 5E 情况,其他 256 色模式的显示缓存情况类似,只是缓存小于该模式时,用的开关位少;大于该模式时,用的开关位多(模式 62 用 B_1 、 B_0 、 P 、 S 四开关位)。

• TVGA 像素寻址

图形显示模式下 TVGA 使用自然坐标系对其存储器进行寻址,各像素按坐标在屏幕定位,原点位于屏幕左上角。坐标最大点 (X_{\max}, Y_{\max}) 位于右下角。

屏幕像素地址为 20 位(不包括 A0000 基地址)格式为:

19	18	17	16	15	0
B	P	S	偏	移	地 址

位 18~19 选择 0~3 号存储块,这两位相应于 3C4 的 0E 寄存器的第 2、3 位。该 2、3 位的值就是存储器块号。

位 17 选择存储器的页面 0 或 1,位 17 相应于 3C4 的 0E 寄存器的位 1,0 表示页面 1,1 表示页面 0。

位 16 选择页面中某段,相应于 3C4 的 0E 寄存器的位 0,0 表示 0 段,1 表示 1 段。

位 0~15 为段中的偏移地址。

假设在 1024×768 像素 256 色模式中取一像素 (X, Y) , $X = 245$, $Y = 546$,此时 $X_{\max} = 1023$, $Y_{\max} = 767$,求该像素 20 位地址表示:

a. 求每条扫描行中的字节数(Y_{box})

$$Y_{\text{box}} = (X_{\max} + 1) / (\text{每字节像素数}) \\ = (1023 + 1) / (8/8) = 1024 \text{ 字节}$$

b. 计算像素所在扫描行字节序号(B_n)

$$B_n = Y \cdot Y_{\text{box}} = 546 \times 1024 = 559\,104 \text{ 字节}$$

c. 计算象素所在扫描行中的偏移量(B_0)

$$B_0 = (X + 1) / (\text{每字节象素数}) = 246 \text{ 字节}$$

d. B_n 加上 B_0 得象素所在字节 B_{box} , 转换成 20 位二进制地址

$$B_{\text{box}} = B_n + B_0 = 559350 \text{ 字节}$$

即, $(10001001000011110110)_2$ 该二进制地址的解释如前所述。

3. TVGA 编程

下面用 C 语言程序的几个简短例子来说明 TVGA 编程:

• 调色板的读写

用视频 DAC 中五个寄存器, 通过端口 3C6、3C7、3C8 和 3C9 来读写调色板寄存器。

读调色板程序

```
void readpallet (int pallet [256][3])
{
    int i,j;
    for(i = 0;i<256;i++) {
        outp(0x3C7,1) /* 设置查找表读序号 */
        for(j= 0;j<3;j++) {
            (pallet[i][j])=(inp)(0x3C9) /* 从端口 3C9 中连续读 3 个数(r,
            g,b)并转换为整形 */
        } /* 大循环,每次读一种颜色 */
    }
}
```

设置调色板

```
void setpallet (int pallet [256][3])
{
    int i,j;
    outp(0x3C8,0); /* 置查找表写序号 */
    for(i= 0;i<256;i++) {
        for(j= 0;j<3;j++) {
            outp(0x3C9,(char)pallet[i][j]); /* 向端口 3C9 写数据 */
        } /* 小循环,每次写一个颜色中 r,g,b 之一值 */
    }
}
```

```

        /* 大循环,每次写一种颜色 */
    }
}

```

• 模式设置

用 BIOS 中的 INT 10H 可实现模式设置,下面只讨论 256 色图形模式的设定。在每次模式设定前,还要设置缓冲区的存取模式。

缓冲区存取模式设置

64K 模式

```

void mode64K( )
{
    outp(0x3c4,0x0b);
    inp(0x3c4);
    outpw(0x3ce,0x506);
}

```

128K 模式

```

void mode 128K( )
{
    outp(0x3c5,0x0b)
    outpw(0x3c5,0x106)
}

```

显示模式的设置

```

set-mode(int mode)
{
    union REGS regs;
    regs.h.al=(char)mode /* 模式号 */
    regs.h.ah=0; /* 中断分量 */
    int86(0x10,&regs,&regs); /* BIOS 调用 */
}

```

• 画点程序

用前述象素寻址方法,设屏幕某点地址为 ix,iy,送到相应的显示缓冲区,即可画出一个点。

```

tygapoint (int ix,int iy,int color)

```

```

union address
{
    long int offset;
    struct bit0-31 {
        int b0-15;
        char b16-23;
        char b24-31;
    } bit;
}; ar;

char out;
char temp;
ar * offset = (long)iy * 800L + (long)ix; /* 设屏幕模向 800 象
素 */
base = (char *)VIDEO + (ar * offset & 0x0000ffffL);
/* VIDEO 定义为 a0000000L, base 为绝对地址 */
out = ar * bit * b16-23; /* 得到高 8 位 */
out = out ^ 0x02; /* 高 8 位中高 2 位取反 */
outp(0x3c4, 0x0E); /* 设置索引寄存器 0E */
temp = inp(0x3c5); /* 取 3C5 中数据 */
temp = temp ^ 1;
temp = temp ^ 2; /* 高 2 位取反 */
outp(0x3c5, temp); /* 送至 3c5 */
outp(0x3c4, 0x0E); /* 设置索引寄存器 0E */
outp(0x3c5, out); /* out 送至 3C5 */
* base = color; /* 把色彩号送至相应地址 */

```

4.2 TIGA

4.2.1 TIGA 原语与结构

许多公司采用 TI34010 协处理器开发了各种图形卡,但是它们和应用程序的接口很少相同,软件公司几乎得为每块卡写驱动程序。1989 年 4 月 TI 公司宣布了 TIGA,它是一种软件接口,目的就是在系统软件和应用软件及基于 340X0 的图形卡之间建立一种标准接口。

34010 和 34020 是快速、强有力的 32 位图形协处理器,它们和 8514/A 不同(它只具备基本图形功能),具有很强功能的图形操作指令集。

34010 有 15 个通用寄存器,256 字节指令高速缓冲器,32 位 ALU(算术逻辑部件),桶式移位寄存器(barrel shifter),32 位内部数据通道,16 位 I/O。和 TI 的数字信号处理器相同,34010 采用 Harvard 结构,内存访问和指令执行并行进行以提高速度。34020 则更先进,它有 32 位 I/O 及其他性能。

除了通用性能以外,34010 有专用性能用来处理图形,包括可编程 CRT 控制,DRAM 和 VRAM(可以同时读写的双口内存芯片)的直接接口。

TIGA 是具有近 150 种功能和原语的完整编程语言。原语有三种:核心原语(core primitives),它始终可用;扩展原语(extended primitives),它保存在库中,当程序需要时,在初始化时装入;用户扩展原语(user-extended primitives),它可由程序员编制、存入库中作为扩展原语。从功能上来说,扩展原语和用户扩展原语之间没有区别,TI 称它们在开销和速度上也没有区别。

一般来说,核心原语考虑基本环境处理(例如,清屏、返送前台和后台颜色、设置光标形状等等)。大部分绘图命令是扩展原语,比

如画线命令；而象素块搬移(BitBlit)这样的阵列功能也是扩展原语。如果需要的话，程序员可以很容易把它们用自己的用户程序来替代。

虽然 TIGA 和 340X0 系列协处理器相关，但它和屏幕分辨率、色彩数及图形限制无关。为了成为一种通用的接口，它要采用新的图形设备和新的 340X0 协处理器。

TIGA 由三部分组成：应用程序接口 AI(Application Interface)，通信驱动程序 CD(Communication Driver)，图形管理程序 GM(Graphics Manager)。

AI 由标题文件和应用程序在编译和连接时使用的库组成。用 TIGA 原语编写 AI 是应用程序编制者的任务。TI 销售驱动程序开发包，可以直接访问标准的 TIGA 环境；还有软件开发包用于编写可下载到 TIGA 的扩展程序，其中包括扩展原语。

AI 和 CD 相连，CD 是一种内在驻留(TSR)程序。它可以在 PC 主机上运行，是专门针对图形卡的。CD 从 AI 取过功能调用，在图形卡和主机之间来回传递信息。

CD 再和 34010 卡上的 GM 通信。GM 和 CD 一样是专门针对具体图形卡的，它由厂家提供。TI 有软件开发包可用于和 TIGA 兼容产品的开发。GM 包括处理图形卡方面通信的命令执行程序，核心原语库，在初始化时下载的扩展原语和用户扩展原语。

4.2.2 TIGA 编程

用 TIGA 可以直接编写图形功能，不过最简便的方法是用主机程序调用的方式。TIGA 的 C 编译程序 GSP(图形系统处理器用)C 是由 TI 公司提供的软件包的一部分，它和 Microsoft C 兼容，因此调用方便。

在编写程序时，程序员需要知道 TIGA 系统特有的并行性能来获得更高的速度，尽可能把程序设计成由图形卡独自工作而不用经常和主机打交道。

为了提高通度，TIGA 从主机向图形卡传递参数有两种方法。

最简单灵活的方法是用 C-信息包(C-packet)功能(注意不要和 C 编程语言混淆,这里 C 是指堆栈)。C-信息包功能用的参数由主机一边接受,由 CD 传递给图形卡存储器的通信缓冲器中。从此,参数被压入处理器堆栈。这样,对主机来说,该功能就象是本地调用似的。

为了进行该项,GM 必须要理解功能的自变量。在 C-信息包方式中,每个自变量是一个具有自己标题的独立信息包。为了使编写标题更容易一些,TIGA·H 这个包含文件中含有额外的定义来表示不同的数据类型。

C-信息包功能的一般格式如下:

ENTRY-POINT-NAME (CMD-ID, numpackets, packet1, ..., packetn)这是一个三子程序进入点的项目,依赖于功能所定的功能调用类型。其中一个进入点用于无需返送数据的功能,一个用于需要返送数据的功能,第三个是向由该功能间接修改的数据传递指针的功能。这里 CMD-ID 为该功能标识,告诉 GM 信息包的数量和种类,随后是信息包本身。

填充矩形的功能如下:

```
#define fillrect (w,h,x,y)
cp __cmd(USER __cp(CMD __ID),4,WORD(w),WORD(h),
WORD(x),WORD(y))
```

按照上述解释可知 cp __cmd 项中,CMD __ID 为功能命令号,功能有四个自变量,数据类型是字。

C-信息包简单灵活,但需要开销来处理信息包格式,并把信息压入堆栈。称为直接方式的另一种参数传递方式中,可以避免这种开销。

直接方式更快,但较难写。它是把原始数据直接送入图形卡上 GSP 存储器的通信缓冲区中。放在堆栈上的唯一参数是一个指针,它指示通信缓冲区中数据所存的地方。该功能必须按照预先定好的格式从该缓冲区中取数据。

直接方式中,参数清单长度以 16 位字计量。所用各项决定参数的规定格式及通信缓冲区如何接受这些参数。

举例来说,最普遍的直接方式项是 `dm __cmd`,它作为标准命令项。TIGA 核心原语 `poke-breg` 就是利用该项的一种典型功能,它传送一个 16 位寄存器号和一个 32 位值,把值装入该寄存器:

```
#define poke-breg(regno,value)
dm-cmd(POKE __BREG,3,(short)(regno),(long)(value);
```

这里用 3 是因为参数值是 32 位值(两个字)。为了能调用该原语,参数指针 `date __ptr` 要放在堆栈上指向第一个参数 `regno`(寄存器号)的位置。

TI 希望大多数功能采用 C-信息包方式,只有对时间有特别要求的功能采用直接方式,就是它们也可能先写成 C-信息包方式而后再修改。除了容易编写之外,C-信息包方式也较安全。在直接方式的最快捷的形式时,没有设置数据量大小的检查,缓冲区很容易溢出。先用 C-信息包方式确定正确无误不失为一种较好的方法。

4.2.3 TIGA 的扩展性

TIGA 所确定的发展路线也是它的优点。TIGA 并不事先局限于屏幕分辨率、色彩数或所用的图形协处理器,因此扩展性很好。TI 的 34020 比 34010 大约强五倍,但依然是 TIGA 兼容性的。

要建立标准的软件图形接口都会遇到两个主要问题。其中一个和软件开发者有关,另一个和硬件制造厂家有关。和软件开发有关的问题是速度。图形接口必须快速,否则应用程序开发者会绕过它,甚至不惜为专门的图形卡写自己的驱动程序。

硬件生产厂家的问题是市场占有率。除非组成象 VGA 那样的工业标准,图形卡制造商不会把自己限死在一种方式上。他们会在其中增加一些功能使他们的产品工作得更快更好。事实上,就是在有标准时,图形卡制造商也喜欢提供额外的功能。大部分 EGA 和 VGA 卡提供比 IBM 产品要多的功能。于是出现了增加功能而不顾标准接口的情况。

TIGA 通过提供扩展命令集来实现额外功能的愿望。制造商和用户可以借助 TIGA 开发包用一种兼容的 C 编译或汇编来编写扩展程序并把扩展部分连入 TIGA 库。所以,在编制应用程序时,可以编写新的命令,并把它连入 TIGA。

相对来说,速度问题更困难。TI 花了许多时间优化原语和接口,使 TIGA 工作得尽可能快。

通过增加扩展原语,就可以为专门的应用程序实现 TIGA 优化。比如,TI 采用一组扩展原语编写窗口驱动程序,它是窗口软件快速的原因之一。

TI 可提供库的源码。程序员若需要专门功能来加速特定的任务,可以研究库中的功能,采用有关信息或编写全新程序。

发挥图形协处理器速度的另一种方法是利用系统固有的并行处理能力。只要仔细地进行编程,就可以实现图形处理器和 CPU 并行工作。此时,CPU 通过 TIGA 启动图形协处理器进行图形处理的工作,而 CPU 在该处理完成前,一直处理别的任务。

4.3 8514/A

4.3.1 8514/A 结构

8514/A 是由 IBM 公司作为 PS/2 微机的高分辨率图形标准随 VGA 一起于 1987 年推出的。兼容制造厂商处理 8514/A 就象 VGA、EGA 及其它较早时 IBM 图形标准的手段类似。他们为了抢占市场,一方面在寄存器级实现他们硬件的标准化,另一方面提供兼容扩展来增强他们的产品。VESA 已有委员会对 8514/A 进行工作。

在 IBM 图形产品中,8514/A 独树一帜。它不是从别的 IBM 图形标准演变而来,(VGA 是从 EGA 发展而来的)也不和以前的各种标准兼容。

在 IBM 的实施方案中,没有任何理由要 8514/A 兼容。8514/A 是一种 PS/2 产品,而 PS/2 系列机在母板上已有 VGA。在 PS/2 中,8514/A 适配器插入视频总线,在 VGA 工作时它不起作用。IBM 没有发布在 AT 总线上运行的版本,8514/A 只为 PS/2 系列机所用。

IBM 一反传统,不发表 8514/A 硬件规格说明,而是让程序员使用和应用程序接驳的功能调用。这就是说,IBM 并不打算在将来采用 AI 的产品中维持硬件或寄存器兼容性。

结果,要想保持和 8514/A 一致,任务比 VGA 难得多。从 IBM 公司在 1987 年 4 月宣布 8514/A,并在 7 月发售的两年多以后才有第一套寄存器兼容的芯片组推出。由于原始资料少,8514/A 牵涉的反工程要多得多。

8514/A 物理上是一种两芯片为一组的 VLSI,其中有一片主芯片称像素地址管理芯片和一片像素片称像素数据管理芯片。Chips & Technologies 的兼容产品是一整片。

主片包括与 PC 总线的接口、显示控制器及图形处理器。像素片对图象的像素进行数据处理。

显示控制器控制显示同步信号,比如水平同步及垂直同步;可以生成地址用于显示存储器串行传送和接口控制;并对显示存储器进行刷新。

从程序员的角度来说,图形处理器是主片中最有意思的部分。在创建图形中,它完成大部分工作,并且还生成坐标地址、画线,在显示存储器中任意复制矩形、向主机传送数据等。

像素处理器的任务比较有限。它处理显示存储器中的像素。它可以对源和目的像素一起执行 16 种逻辑操作,16 种算术操作。它有两个色彩寄存器用于交叉面像素处理,还有八种色彩比较功能允许只选择确定的像素进行处理。它也有读、写屏蔽来限制对给定的显示平面的处理。

像素处理器的另一个重要特点是桶式移位寄存器。在数据块

复制时,芯片可以非常快地对准源象素,因此 8514/A 的 BitBlt 速度很快。

除了两块芯片外,8514/A 板还包括显示存储器(通常为 1M 字节 VRAM),一片象素串行转换器芯片,一片 RAMDAC (RAM D/A 转换器)及加电自检 ROM。象素串行转换器的作用是,从显示存储器中一次取一个字节数据,把它转换为一连串的串行位,适应视频显示。RAMDAC (INMOS IMS171 或兼容芯片)把从串行转换器接收得的数字象素信息转换为 8514/A 监视器所需的模拟 RGB 信号。

8514/A 的视频显示特点是隔行扫描。虽然隔行扫描并不新鲜,广播电视就是用的隔行扫描,不过在 MS-DOS 系统中它并不多见。

隔行扫描的好处是垂直分辨率加倍时,监视器扫描频率不用加倍。因为监视器价格随扫描频率增加(不是线性关系),这样就降低了成本。

在 8514/A 的情况下,这是一笔可观的费用。在标准宣布前,一系列多频监视器已接近 8514/A 的 33kHz。若 8514/A 不用隔行扫描,则当时还没有合适的监视器(因为需要 60kHz,超过了当时监视器的最高水平)。

在设计 8514/A 时,IBM 很清楚地使其监视器扫描频率低一些。逻辑显示分辨率为 1024×960 象素,实际为 1024×768 。较低的垂直分辨率可以降低对扫描频率的要求,更可以避免闪烁。隔行扫描把屏幕刷新频率减半,分成两次扫描来完成整幅屏幕的刷新。计算机监视器典型的垂直扫描频率为每秒 60 次,这时看不出闪烁。把垂直分辨率加倍,刷新频率减到每秒 30 次,闪烁出现。把垂直分辨率定在 768 象素,可用每秒 44 次的扫描频率,能明显地削弱闪烁,此外,8514/A 监视器采用长余辉管,显示质量得到保证。

4.3.2 8514/A 编程

与 340X0 不同,8514/A 并不是一种通用的图形处理器。它可

以接受命令,但因为板上没有存储程序的芯片,所以可编程能力不行。它的命令集简单而且不能扩展。

8514/A 命令所完成的任务要比 VGA 时复杂得多。用 VGA 时,命令实质上是改换图形系统状态。8514/A 用单条命令完成不少任务,不过执行过程更象是把命令装载进一个队列而不是对微处理器编程。主片上 PC 总线接口的任务之一就是维持命令队列,包括探知队列什么时候满等情况。

IBM 希望程序员采用以 8514/A 为基础的应用程序接口 AI,这是一种利用驻留内存程序 HDILOAD.EXE 的功能调用系统,由它对图形卡进行访问。

对比之下,寄存器编程是通过 I/O 口做的,每个寄存器有其相应的 I/O 口地址。8514/A 中的 56 个寄存器中,大部分是字寄存器。调色寄存器是个例外,它们全是 8 位。

8514/A 芯片组(图形处理器、显示控制器、象素处理器)有相应的一系列寄存器。比如,象素处理器的寄存器有:背景色寄存器、前景色寄存器、写屏蔽寄存器、读屏蔽寄存器、色彩比较寄存器、背景混合寄存器、前景混合寄存器、短划向量(short stroke vector)传递寄存器及象素数据传递寄存器。显示控制器的一系列寄存器的共同特点是只写性质。

对 8514/A 编程比 TIGA 板容易得多。可以采用应用程序接口(AI)下合适的功能调用,或在寄存器中放入适当的数值。这两种情况下,它的命令集都比 TIGA 的简单,容易记忆和交往。

但是用 8514/A 编写实用程序并不那么简单,就是用 AI,程序员也必须在很底层的水平上和 8514/A 打交道。因为功能有限,程序员可能不得不花更多的时间来构思如何做某事。有人把对该图形卡的编程和对绘图机的编程相提并论。

比如,用 BitBlt 命令,程序首先执行一条命令告诉 8514/A 需要何种传递(即,从主机到显示或显示到主机)。第二条命令给出读或写的地址。

MS-DOS 图形通常的巧妙之处在于程序员与硬件直接打交道来使显示最快。如果通过软件接口,工作速度明显较慢。许多厂家认为用 8514/A 这方面问题少一些。

8514/A 与以前的标准有一些明显的差别。其一,8514/A 就是通过 AI,速度也很快。对许多应用程序来说,它的速度足够快,开发者用不着撇开软件接口。

其二,8514/A 设计用来在 OS/2 这样的新环境中工作。IBM 版本只在 PS/2 上可用,它们是 OS/2 的 IBM 平台。在 OS/2 下比在 MS-DOS 下更难掌握牵涉硬件的技巧。

程序执行速度和编制难度是一对矛盾,大部分应用程序的开发者不想直接与硬件打交道。不过系统软件开发者可能很需要对寄存器编程来为应用程序获得尽可能快的速度。这就是 Microsoft 公司为窗口软件 Windows 开发 8514/A 驱动程序的原由。

8514/A 的设计思想是性能适可而止,实用而廉价。它不是具有内置功能扩展库的全功能图形协处理器,而是一种具有数量有限、硬线连接式指令的相对简单的处理器。

理论上说,图形协处理器越强越好,这也是 34010、80860 和大部分其他图形协处理器的思想。它们昂贵,不过易于通过编程来发挥强大功能。

8514/A 指令集有限,在一旦决定如何去做某事、方法又直接时,编程很容易。如果工作复杂,需要勾画出整个从要求到编程策略时,是很费脑筋的。

8514/A 明显缺乏画弧原语、多边形填充原语。在做多边形填充时,要用 Begin Filled Area 和 End Filled Area 命令来完成。

虽然命令集固定在硬件中,芯片组仍有相当潜力在分辨率和色彩方面扩展。此外,地址芯片可以处理多达四个图形芯片。这样使 8514/A 做 32 位色彩处理和八位色彩速度一样。

4.3.3 8514 A 的现状

一些兼容厂商宣布了他们自己的 8514/A 芯片组,它们可以

低到在寄存器一级上兼容。和许多兼容 VGA 卡相仿,这些芯片组设计得比 IBM 的功能更多、性能更好。

西方数字公司(Western Digital)称它的芯片组为个人工作站图形阵 1 号 PWGA1(Personal Workstation Graphics Array 1)。它包含一系列 8514/A 所没有的功能,其中有支持高达 1280×1024 像素分辨率的隔行与逐行扫描这样的功能。芯片组的时钟频率为 60MHz,比 IBM 的快 30%,因此它可以加速诸如 BitBlt 及矩形填充这样的操作。PWGA1 存储器传送时间较短,因此可以进一步加速。

此外,PWGA 1 有 turbo BitBlt 方式,可以在 16 色图象时将速度加倍。在这种方式时,从图形卡上 VRAM 传送来的八位数据作为两个四位传送处理,即每次传送可读两个像素。四位可定义 16 色,八位可定义 256 色。如果应用程序用 16 色或少于 16 色,PWGA 1 把两次四位传送合为一次八位传送,这样可节省一半时间。

8514/A 不仅打破了 IBM 兼容图形标准的技术基础,而且要冒由于和以前产品不兼容而产生的市场风险。因为 IBM 不宣布硬件详细说明,谁也不能阻止 IBM 推出一种全新的产品,它遵从软件规定。

另一方面,IBM 要把 8514/A 装入 PS/2 新型号的母板上,因此 8514/A 有几年时间进行扩充。

用 8514/A 来进行更新,对用户来说有点进退两难。它不向后兼容,而且也并非所有软件都有 8514/A 驱动程序。

解决问题的方法有多种。其一是把 8514/A 适配器和 VGA 卡装在一起,利用作为 VGA 一部分的标准通道连接器连接起来。这种方法可行,但要多用一个扩展槽。此外,在一些 VGA 执行程序中,有内存冲突的问题。兼容厂家在 VGA 卡上增加了性能和模式,他们的 BIOS 比 IBM 的大,因此他们的执行程序会在 8514/A 存储空间运行而形成冲突。

另一种方法是把 8514/A 和 VGA 做在同一块电路板上。这会

使价格上升,但用户使用方便,可靠。

TIGA 和 8514/A 在技术上都代表了 VGA 之上的一大进步,它们仍在经受市场的考验。

4.4 XGA

4.4.1 引言

1987 年 VGA 和 8514/A 作为图形标准随新推出的 PS/2 系统出现在世人面前,而三年半后 IBM 最终揭示了它的新一代 PS/2 图形硬件:扩展图形阵列 XGA(Extended Graphics Array)。

XGA 是随 IBM 最新的 PS/2 90XP486 型和 95XP486 型的缺省图形显示平台推出的。在台式 90 型上,XGA 做在母板上;在立式 95 型上,它是做在独立的微通道(Micro channel)结构电路板上。XGA 显示卡/A 适用于其他 386 和 i486 为基础的 PS/2 上。

IBM 把 VGA 换成 XGA 作为缺省的图形平台是令人瞩目的。几年前,传闻四起,说 IBM 要在 PS/2 母板上实现 8514/A 高级图形技术。但是 8514/A 缺乏一个主要的性能,而它又是回避不了的,这就是与以前的兼容性。XGA 和 VGA 硬件兼容,消除了这个问题,因此适合在母板上实现。

表 4-6 XGA 和其他标准的比较

平 台	分辨率	1024×768 隔行扫描	基本 兼容性	图形协 处理器	映射 存储器	286-/8086 兼容性	RAM 类型
PS/2 VGA	640×480×16	否	VGA	否	是	是	DRAM
	320×200×256						
Super VGA	800×600×16/256	否	VGA	否	是	是	DRAM
	1024×768×16/256						
8514/A	640×480×16/256	是	无	是	否	是	VRAM
	1024×768×16/256	注 3	注 4				
XGA	640×480×2/4/16/256/65536	是	VGA	是	是	否	VRAM
	1024×768×2/4/16/256						

注 1:有些 Super VGA 1024×768 隔行,但大多数可在隔行或逐行之间转换

注 2:有几种 Super VGA 用 VRAM

注 3:8514/A 系列支持 1024×768 逐行扫描

注 4:VGA 兼容性是通过 8514/A 和它之外的独立 VGA 相连实现的

从某些角度来看,XGA 是 VGA 和 8514/A 图形平台之间的结合物。表 4-6 表明了 XGA 与其它标准的一些比较。

XGA 是在英国的 IBM 赫尔斯莱实验室(Hursley Lab)中开发的,8514/A 和图象适配器/A 也是这样。因此在 XGA 的设计中保持了许多 8514/A 的特性,当然它们是以一种不同的风格出现的。

有些新的特性为的是利用微通道结构总线的优点,比如总线支配(bus mastering);有的是为了给已有的设计以更大的灵活性,比如存储器映射帧缓冲器(memory-mapped frame buffer)和硬件光标,这样可以减轻软件开发者的负担。其他特点还有可定义位图、状态保存等。另一个好消息是,IBM 公开了 XGA 全部寄存器的性能指标,这一点和 8514/A 绝然不同。8514/A 因缺乏这方面的文件,在为公众接受上大受伤害。XGA 适配器接口支持和 8514/A 应用程序兼容。

从技术观点来看,XGA 几乎解决了 8514/A 的一切问题,只是隔行显示和简化画线依然保留下来。

4.4.2 XGA 模式及寄存器

XGA 有三种不同的模式:VGA 兼容模式,132 列 VGA 兼容正文模式和扩展图形模式。扩展图形模式是最令人感兴趣的,它提供了较高的分辨率和具有实质性的图形加速能力。

除了保持与最初建立的 VGA 标准的完全兼容外,IBM 也从大量的 VGA 兼容产品中学到了一些东西,并实现了较宽的数据通路。VGA 模式时内部仍只有 8 位数据通路,但支持 32 位宽的总线。此外,内部的写操作高速缓冲可使芯片对总线数据进行分段写操作,不需要用等待状态来暂停系统的工作。

根据 IBM 公司的说明文件,当 XGA 在 VGA 模式时,在 DOS

下比原 VGA 可快 90%，在窗口环境下可快 50%。除了性能改进之外，在该模式中 VGA 功能没有变化。

值得注意的是，当在一个系统中采用多达 8 个 XGA（结构设置软件支持至多 6 个）时，任何时候只能有一个 VGA 有效。因此，如果要把 XGA 转换为 VGA 模式时，必须确认系统中无其他有效工作的 VGA，否则系统会因 I/O 冲突而崩溃。

采用 132 列正文模式（VGA 的一种扩展）时，人们可以在屏幕上显示并处理每行 132 个字符的正文。字符宽为八个象素，有效水平分辨率 1056。字符高度与所用字体有关，整屏可有 132×43 、 132×50 或 132×60 个字符。

目前，只能通过人工处理 XGA 寄存器来使用 132 列正文模式，不过最终将完成由视频模式 14h 来调用。从实用角度来说，132 列正文模式是一种 VGA 模式，因此 VGA 各种提示也是适用的。

扩展图形模式有许多令人振奋的特性，比如 65536 种色彩，总线支配，画图加速及硬件光标等。虽然该模式的有些特性在其他模式时也可用，大多数 XGA 寄存器及功能是专用于扩展图形模式的。

图形协处理器和显示控制器是 XGA 的核心。图形协处理器控制 VGA 兼容性，画图功能及存储器管理。显示控制器包含有彩色对照表的 RAM D/A 转换器（RAMDAC）、CRT 控制器、硬件光标支撑和 VRAM 串行转换器（把数据从 VRAM 中提取出来形成串行信号用于显示）。

访问 XGA 是通过两组寄存器来实现的，第一组映射在系统的 I/O 空间，另一组映射到存储器。由于 XGA 配置可变，这些寄存器的地址也可变化，因此同一系统中允许有多个 XGA。

I/O 寄存器映射在 I/O 地址 $21x0h$ ，这里 x 是 XGA 的一种选择。根据 IBM 提供的材料，系统只有一个 XGA 时，它通常为 6，因此 I/O 基底地址是 $2160h$ 。

存储器映象寄存器占 128 个字节，在 8KB 块的最后 1KB 中。

该块驻留在 PC 地址 0C0000h~0DFFFFh 的范围内,以 8KB 为一区。要有 8KB 块的目的只是在 XGA 显示适配器/A 时,在该块前 7KB 装 ROM 数据。XGA 的母板形式不需要独自の ROM,主板上的 ROM 已包含所有必要的信息,比如 XGA 初始化代码。XGA 的上述选择 x 决定了该 128 字节在 8K 块中的位置。

I/O 寄存器事先规定适合 XGA 的显示控制器。而存储器映象寄存器主要和图形协处理器有关。XGA 加电自检程序设置了这两组寄存器的基址,也可通过检查 PS/2 的 POST 各寄存器来确定这些地址。

许多存储器映象寄存器大小是 32 位,因为 XGA 适合 32 位环境,比如 Intel386 和 i486。由于软件支持是 IBM 为 XGA 开发的,它只适用于 386 或 386 为基础的 PS/2 机(包括 386SX 为基础的 PS/2)。XGA 也提供 Motorola 格式寻址,可使 Motorola 68000 一类的处理器共享 XGA。

4.4.3 操作特点

初始化是使用图形设备的必要步骤。在 XGA 的情况下,初始化主要通过操作模式寄存器把 XGA 设置在扩展图形模式,然后针对所需分辨率确定 CRT 控制寄存器数据。可选择分辨率为 640×480 及 1024×768 象素。

图象质量不仅和分辨率有关,而且依赖于象素色彩表现力。如果每个象素用 16 位来表示,即 65536 种色彩,则图象几乎和完美的照片一样。象素 16 位的分配是:5 位红,6 位绿及 5 位蓝(5-6-5)。换句话说,每个象素可有 32 种红色调,64 种绿色调,32 种蓝色调。

这种配色和 PC 标准 TARGA 格式 5-5-5(1 位用于重迭),i860 格式(6-6-4)各不相同。采用 5-6-5 方案是因为 IBM 不同机型上早已采用了,此外,眼睛对绿色调中的变化比起其他两种原色来更敏感。

XGA 的显示控制器不仅用于初始化,也用于色彩对照表和标

块(复盖屏幕的 64×64 象素块)的处理。

人们可以利用对照表把 1-, 2-, 4-, 或 8-位象素值转换为适当的 RGB 值。过程是:先把象素值作为对照表中的变址,所得的结果为 RGB 值。RGB 值通过内置的 DAC 从数字量转换为模拟电压。

象 VGA 和 8514/A 一样,XGA 的对照表支持 64 级(二进制 6 位)原色,这样一共可有 262 144 种色彩组合。因此在 256 色模式中,人们可从 262 144 种色彩的调色板上选 256 种颜色。

在标块中,每个象素有四个可能值:标块色彩 0,标块色彩 1,透明和补色。标块色彩由专用寄存器定义,人们可以为它的色彩规定 RGB 值。这些 RGB 值直接送往 DAC,这样就允许应用程序完全修改本地调色板而不必保存光标的有关项,也不用担心光标经过不同显示区时改变颜色。

在设置标块透明时,可以允许光标比所定义的 64×64 象素块小。如果希望在任何背景时总可以看到光标,则可以采用补色设置。

在建立一个多任务环境时,系统软件开发人员最头痛的是保存图形硬件当前状态,允许另一应用软件使用图形设备。这种状态保存必须考虑硬件正在操作之中、调色板正在更新之中这样的可能性。XGA 在设计时就考虑了任务切换问题,它有额外的机构来保存和恢复硬件状态;包括被中断的操作。

XGA 采用目前 PC 图形标准中独树一帜的位图(bit map),在执行画图功能前,它们必须事先定义。位图和以前提到的位面不同。位面中的每一位和屏幕上的一个象素对应,由若干个位面来决定象素的色彩;而位图中是每个字节(或更多位,看象素色彩数而定)来对应一个象素。

位图是存储器的线性区域,以象素为单位定义宽和高,所谓的深度是指象素包含的位数。比如一幅 8 位象素的位图,如果宽为 10,高为 6 则需要 60 字节的存储器区域。在存储器中,位图中一行的最后一个象素/字节直接挨着下一行的第一个象素/字节。

位图的突出优点在于,它们可以放在系统地址和存储器空间的任何地方。如果人们在自己的程序数据区中定义了一幅位图,XGA 可以对它进行读写操作,用不着通过系统处理器把数据拷贝到 XGA 或从中拷贝出来。

XGA VRAM 映射到系统地址空间,通常位于 386 的 4GB(gigabyte)地址空间的最高端。要把它作为位图用时,直接使用相应地址就行,由于地址空间极大,不会有存储器冲突的问题。

4.4.4 总线支配和画图基础

当 XGA 访问位图时,需要确定是访问本地的 VRAM,还是远地的系统。对远地访问,由 XGA 仲裁总线并开始访问系统内存。这就是 XGA 总线支配(bus-mastering)能力的作用。在使用系统存储器为位图时比使用本地 VRAM 的开销要大些。不过当系统处理器干别的工作时,由 XGA 处理器来处理系统存储器,明显有益于提高性能。

这种总线支配存在一些潜在的问题。386 和 i486 处理器通过内部页面转换表(page-mapping table)支持虚拟存储器。该页面转换表允许诸如扩充内存管理、DOS 扩展程序和高级操作系统这样的控制应用程序每次建立 4KB(或一页)虚拟 PC 地址。这样,当软件应用程序以为它把数据写往地址 WWWWW,而页面转换表可能把它转换成物理地址 QQQQQ。

在许多情况下,应用程序不可能知道它所使用的地址并非物理地址。为了对总线进行正确的控制,XGA 需要一个物理地址,否则它可能对一个错误的地址进行数据拷贝工作。要解决这个问题,需要一些控制软件。

如果一个应用软件要访问控制软件的页面转换表,它把该信息传递给 XGA,然后使用页面转换表、完成自己的虚拟——物理地址转换。可惜的是,许多控制程序和操作系统并不提供对页面转换表的访问。不管怎样,一般的 DOS 应用程序最有可能正确使用总线支配,因为内存的前 640KB 通常是最不可能虚拟化的。

一般情况下,XGA 的 VRAM 总是映射在内存高地址,不过也可以利用软件把 XGA 映射在 A0000h 或 B0000h 处的 64KB 区,这是标准 PC 视频内存区。通过这样的方法访问 XGA 的 VRAM 中不同的 64KB,只需要往口径变址(aperture index)寄存器中写入一个变址值。这种分组机构在系统处理器实模式(real-mode)应用软件中是很方便的。与此相应,整个 1MB VRAM 可以全部映射到系统存储器前 16MB 中没有存储器冲突的某个地方。这种形式的转换在系统处理器保护模式(protected-mode)应用软件中很有用。

画图命令需要一种或多种位图:源、目的和图案。源位图包含需要复制或作为拼花的数据;目的位图是指画图或复制数据等操作工作的对象;图案位图包含一幅单色图案,用于某一区域的花样。

XGA 还支持一种附加的位图,称为屏蔽图。屏蔽图是单色位图用于执行任意剪取(arbitrary clipping),这是一种非规则几何形状的剪取方法。在屏蔽图有效时,图中每个 0 位表示在画图时该像素不变,而 1 表示该像素可在画图时变化。如果屏蔽图小于所给定的操作目的图,屏蔽图的外缘定义为剪取矩形。

在窗口环境中,窗口重迭时屏蔽图显得特别有用。这是因为只要定义屏蔽图允许只在所定窗口的裸露区画图,那么就可以在重迭窗口的某一层上画图,而不必事先剪取上层窗口中的内容。1024×768 像素的全屏幕屏蔽图只占 96KB 存储器。人们也可以部分使屏蔽图有效,这样它只作为剪取矩形而不作任意逐点剪取。

上述四种位图由 5 像素图寄存器定义。位图的第一个像素在显示位图的左上角,它的座标为(0,0)。屏蔽图原点寄存器定义屏蔽图在目的图上的位置。所有的图限制在高和宽为 4096×4096 像素范围内。

XGA 提供的画图操作有:画线、短向量、填充矩形、BitBlt 和区域填充。分别解释如下:

画线并不象想象中的那么简单,以为只是一对 X,Y 坐标的问

题。XGA 采用 Bresenham 画线算法,不过首先得计算初始 Bresenham 参数。在画线较短时,这个过程的开销是明显的。计算这些参数的方法和在 8514/A 上的实质相同。

XGA 短向量(short vector)和 8514/A 的短划向量(short stroke vector)类似。这些向量在长度上可达 15 像素,它们的方向为 45°的倍数(水平、垂直、对角)。XGA 短向量的好处是,每一向量的定义只消一个字节,每次可传 4 个字节,允许快速数据传送,因此画得快。

填充矩形的意义不解自明,只要标明矩形的宽、高和位置就行。BitBlt 和填充矩形类似,只是必须规定源图和目的图。XGA 还有能力执行简单的彩色扩展 BitBlt,其中有一种源是单色的,凡是 0 位转换成一种颜色而 1 位转换成另一种颜色。目的图的深度可以是 1 到 8 位中的任意一个数。为了在色彩丰富的屏幕上显示字体,彩色扩展是很有用的。

区域填充是一种修改了的矩形填充。XGA 图协处理器采用图案图作为样张用于扫描转换。这种填充利用一位标志,在逐行扫描图案位图时开关填充状态。最初,每条扫描线的填充状态是关。当到 1 位时,图案中随后的像素(或位)被填充,直到遇到下一个 1 位为止。

线、短向量和填充矩形也可以使用源图和图案图,这样利用图案图来形成不同花色的线和区域,利用源图来装饰某一区域(在填充矩形时)或提供彩色花样的线和短向量。

除了上述五类画图操作之外,还有四种画图修改。它们是画色、混和、色比较和像素位屏蔽。

画色很容易用,前景色表示正常画入的颜色,而背景色规定了打算在彩色扩展中(对 0 位)采用的颜色。

混和在其他平台上称为光栅操作,它提供一种机制,用它来完成目的像素和源像素的混合操作。比较典型的混和是 XOR(异或)操作,用于光标和需要着重显示的地方。

色比较用于正常象素更新。目前，象素值(或色彩)和目的色彩比较值寄存器进行比较，如果比较的结果为真，象素就不需要更新。实际一共有8种比较条件：真、假、大于、小于、等于、不等于、小于等于及大于等于。在需要保护一定区域，比如图景中的背景和前景物体，不让它们的色彩更新，这种色比较就很有用。

象素位屏蔽可以控制象素的哪些位可以修改，哪些位不可以。它的最大用处是保护二进制色彩和色平面。

第五章 图形处理器

为了得到理想的计算机图形,人们在显示设备和图形算法方面做出了很大的努力。随着高质量图形的获得,人们的兴奋又转成了不满足:图形太消耗资源,太费时间,有时甚至不可忍受。图形处理器是解决图形品质和速度的有效方法。本章主要介绍 Intel 82786、TMS 34010 和 34020,以及密切相关的视频 RAM。

计算机硬件,包括主处理器和图形处理器的不断发展,为计算机图形奠定了良好的基础。如今还是高档的性能指标、常人不敢问津的昂贵代价,明天就可能成为普通的设备,而新一轮竞赛又早已开始。

5.1 Intel 82786 图形协处理器

Intel 82786 图形协处理器的目标是解决更高分辨率、更多色彩和更快的显示速度。它的出现改进了目前诸如 Microsoft Window 或 Digital Research 的 GEM 这样程序的性能,使用户有更大的灵活性。由于提高了显示速度和分辨率,使软件开发者有更多的余地设计新系统。

Intel 把 82786 窗口定义为位图的一部分,由显示处理器输出。82786 可以同时显示很多窗口,在任一条扫描线上可多至 16 个水平窗口段(window segment),而垂直窗口段数最多可和一屏的扫描线数相同。

在系统时钟为 10MHz,视频时钟为 25MHz 情况下时,82786

的速度是:画线为每秒 2.5 兆像素(2.5MPS),画圆为 2MPS(Million Pixels per Second,每秒兆像素),填充区域为每秒 30 兆位。

图形协处理器总线接口完成总线块传送 BBLT(Bus BLock Transfers)为 24Mbps,每秒画 25 000 字符与字体或指向无关。内部硬件处理字符字体速度达 16×16 像素。较大的字体需要软硬件两者配合或 BBLT 传送。

82786 可以立即完成垂直和水平卷滚(scrolling),无需额外硬件,并可在水平和垂直方向上放大至 64 倍(此时 zoom 缩放功能和像素复制 pixel replication 功能等同)。

82786 内部图形处理器和显示处理器均用 22 位地址位,产生 4MB 地址范围,由主处理器和图形显示处理器共享。82786 可同时显示多达 1024 种颜色,它支持的分辨率从 640×480 像素 8 位到 1024×1024 像素 2 位,60Hz 逐行刷新。

若再使用视频 RAM,还可将分辨率提高到 2048×1536 像素 8 位,最高为 4096×4096 像素 1 位。此外,可以将多片 82786 同步起来构成具有更高分辨率、更多色彩、更宽带宽、性能更强的系统。

5.1.1 图形协处理器结构

82786 图形协处理器结构的主要部件是图形处理器、显示处理器、总线接口、RAM 刷新单元和图形存储器(见图 5-1)。图形处理器执行主 CPU 放在 RAM 中的指令,并为显示处理器更新图形存储器中的位图。

显示处理器把由图形处理器生成的位图转换成系统显示所需的光栅信号序列(raster sequences)。总线接口有 4 个逻辑接口用于图形处理器对外部存储器访问、主处理器访问处理器寄存器和图形存储器、图形处理器访问图形存储器、视频 RAM 刷新请求。

1. 图形处理器

图形处理器通过总线接口逻辑访问,来执行驻留在主处理器存储器中的图形命令联结表,主处理器首先建立图形指令表及其参数,然后主处理器将图形处理器初始化,方法是把连接地址高字

节和低字节写入图形处理器的控制寄存器并设置连接命令位(link-command bit)。

每次复位后,图形处理器以访问模式(poll mode)启动,并检查其指令寄存器位 0 的值(指令表结束标志)。当主处理器装载指令时,就清除了位 0,图形处理器通过一个连接地址,在所指的存储器中指令表的起始点处开始执行指令。

图形处理器一旦由主处理器启动,它执行指令就无需和主处理器或显示处理器同步。主处理器可以在图形处理器 4MB 地址空间的任何地方写图形处理器指令。

画像素只在图形存储器中完成。图形处理器可以更改图形存储器中任一位图的各位,无需顾及显示处理器是否正在对该位图进行显示操作。

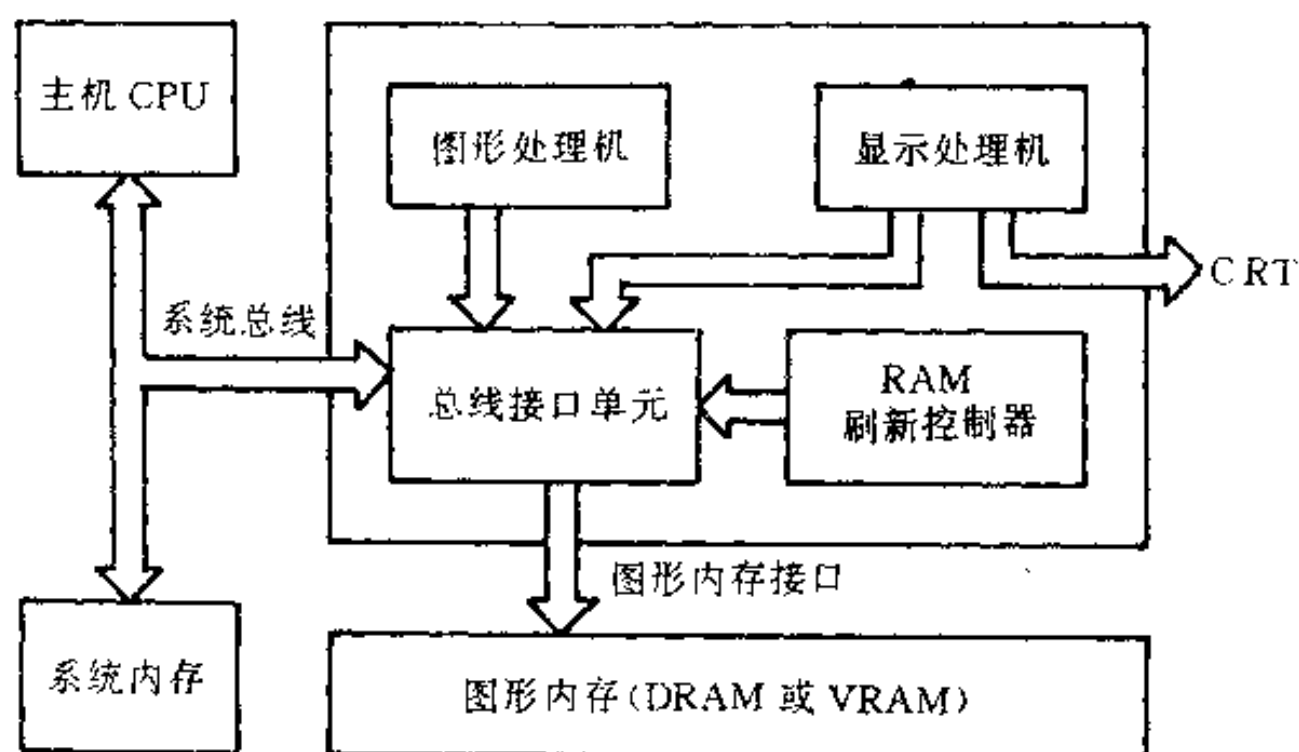


图 5-1 82786 结构示意图

图形处理器也能以检取模式(pick mode)工作,此时图形处理器不是更改图形存储器,而是执行指令并检查所定址的像素是否在窗口描述符块(window descriptor block)中定义的剪取(clipping)窗口中。如果物体在剪取窗中,设置标志加速选择屏上物体。

2. 显示处理器

注：
C = 光标开
D = 显示开
IL = 隔行
W = 窗口状态使能
S = 同步从模式
B = 空从模式
AA = 加速视频
S = 光标大小
X = 十字标模式
T = 透明模式
CS = 光标状态
CSC = 光标状态控制

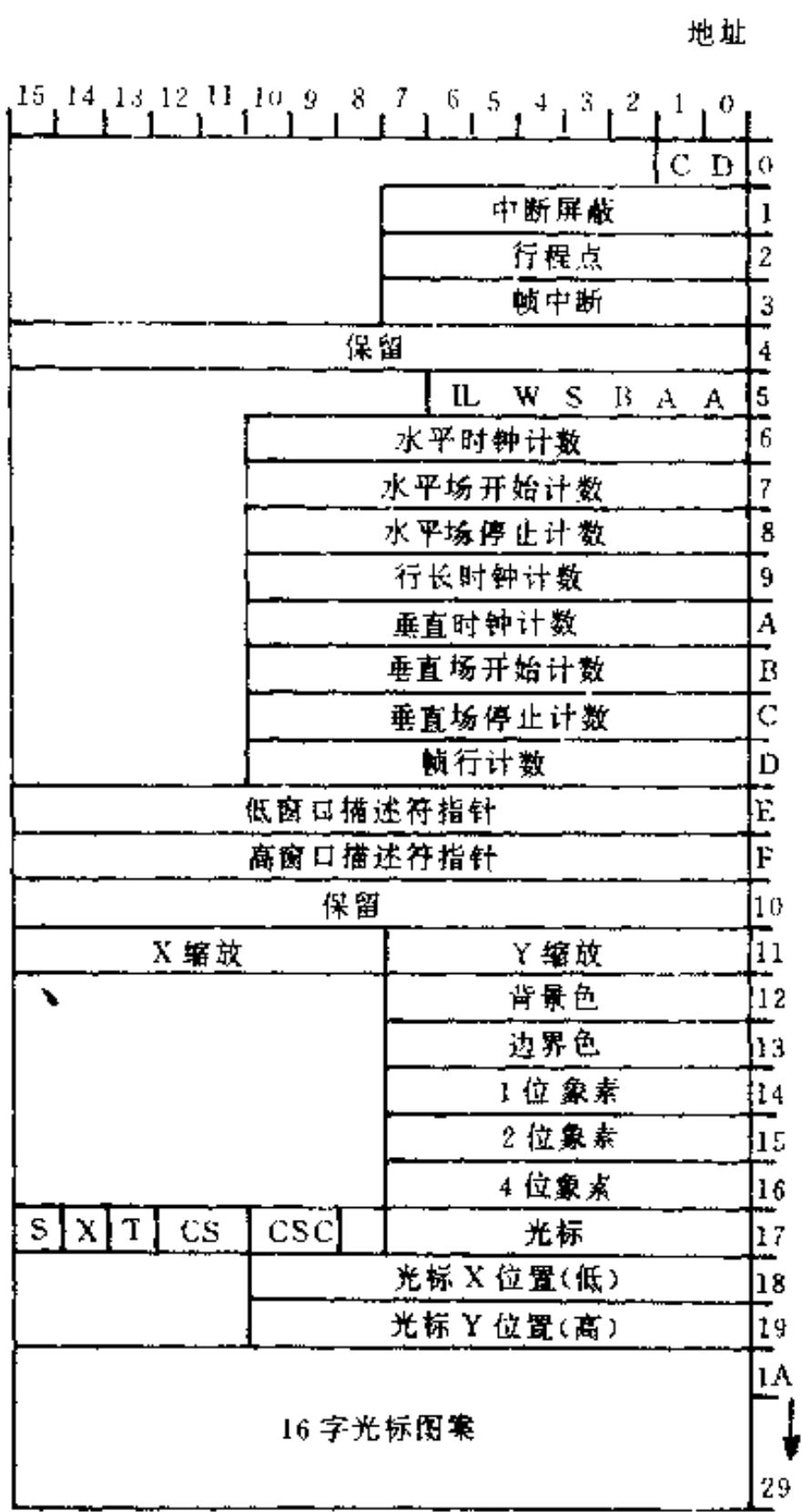


图 5-2 图形协处理器显示控制块

显示处理器控制象素输出给 CRT、激光打印机或其它光栅输出设备。显示处理器把由图形处理器生成的位图转换成象素序列

和所接设备需要的控制信号。显示处理器经过优化,数据以像素为单位的位图形式,而在窗口中可切换到 IBM 的 CGA 模式和字节交换寻址(byte-swapped addressing),人们也可以把显示处理器和外部视频源同步起来。

程序指令也可以驱动显示处理器,形式是一样的。显示处理器把一条新指令装入它的寄存器,以便在下一帧执行,每帧一次。这种高度复杂的指令规定了除屏幕窗口之外的一切。指令(见图 5—2)指向另一块存储器,由它规定各窗口。程序参数规定了一系列功能,诸如:硬件缩放、光标显示、窗口、视频像素速率(video pixel rates)及 CRT 时序信号。

显示处理器可以在水平和垂直方向上以 1 到 64 的倍率放大显示屏幕窗口(这是一种像素复制功能,与重新计算终点和重画线相对应)。在通常 25MHz 视频、8 位像素的情况下,人们可以用单一像素分辨率在屏幕上确定各窗口,也可以用单一像素分辨率把光标显示为 8×8 或 16×16 像素。

光标可以是不透明的、透明的方块或是全屏幕十字线(full-screen cross hairs)。人们可以通过编程确定光标颜色、花样、闪烁,也可以开关光标。CRT 时序信号以单一像素分辨率编程,允许高达 4096×4096 像素显示。

改变显示处理器的模式也就改变了一个像素中所含的位数,像素的绘制速度也就不同。基本情况是,每像素 8 位时为 25MHz,4 位时为 50MHz,2 位时为 100MHz,1 位时为 200MHz。因此在较高速率时,水平分辨率会降低到以 2、4、8 像素为单位。

显示处理器在各扫描间的回扫期,通过装载移位寄存器来控制视频 RAM。在扫描行装入 VRAM 移位寄存器前需要对含窗口或光标显示的每条显示线进行附加处理。

3. 图形存储器

82786 主要高性能图形特性之一是其动态存储器接口。该接口使 82786 可以利用双组(dual-bank)和页面两种模式动态 RAM

的全部优点,使其最高速度达每秒 10MB。

图形处理器 1MB 线性地址空间从图形存储器十六进制的 0 地址开始,如果使用的图形存储器空间少于 4MB,那么剩余的地址可对系统存储器存取。通过设置外部地址位的最高两位就可使系统地址指向主处理器 4MB 空间的任何地方。

82786 的设计支持更宽范围的 DRAM 配置:用于隔行扫描的 1~2 组存储体,每组 1-4 行:1、4、8 位宽的 16K 位到 1M 位的 RAM;性能更高的页面或高速页面模式(高速页面模式 RAM 访问周期为 100ns,而页面模式 RAM 需要 200ns)。

· 存储器接口支持 32 片 DRAM,而无需外部逻辑,可寻址范围为 1MB 存储空间。

82786 在其存储器中也可使用视频 RAM。VRAM 包括一个移位寄存器和一个允许视频数据高速移位又不重复访问存储器的串行输出装置。图形存储器利用 VRAM,大大减少了开销,使更多存储周期用于主机和图形处理器。

82786 能直接缩放 VRAM 支持的垂直显示,但对水平缩放需外加电路,因为一旦位图装入移位寄存器,82786 将不能访问该位图来复制这些像素。

DRAM 访问可以是单周期或多周期,单周期是一个 16 位字,而多周期则是 2 个或多个字。在 10MHz 频率下单周期读写访问需 300ns,读改写 RMW(Read-Modify-Write)需要 400ns。多周期访问数据其速率对逐行页面模式为每秒 10MB,对隔行快速页面模式可高达每秒 10MB(为获得最小处理开销,显示处理器只要可能,总使用多周期访问形式)。图形处理器使用单 RMW 周期来处理像素的更新,而将多 RMW 周期用在块的移动中,主处理器仅使用单周期。

5.1.2 82786 功能特点

1. 存储器访问优先级

总线接口逻辑执行四种逻辑接口间的仲裁:82786 对外部主

存储器访问、主处理器对 82786 内部寄存器和图形存储器访问、82786 访问图形存储器、最重要的是 RAM 刷新。

刷新单元产生对总线接口的 RAM 刷新请求信号,并对它们进行排队,以高优先级执行。主处理器、图形处理器和显示处理器按 3 位优先级系统竞争访问存储器。

优先级进一步分解为首次访问 FA (First Access) 和随后访问 SA (Subsequent Access)。缺省的优先级(自高至低)为 RAM 刷新、显示处理器、图形处理器、主处理器。RAM 刷新总是最高优先级,其他可以通过程序控制来调整。这样允许用户通过编程来利用各种性能以达到在不同应用情况下取得最佳效果。

复位后的缺省优先级为主机(FA)7、显示(FA)6、图形(FA)5、显示(SA)3、图形(SA)2。通常图形处理器、显示处理器访问的长度超过一个字,总线接口在访问连续字时提供最大的吞吐率(throughput)。然而,如果顺序访问排斥了从另一处理器来的首次访问的话,就可能会影响整体性能。

这种可调优先级能通过软件编程发挥 82786 的最佳性能,方法是平衡主处理器、图形处理器和显示处理器的访问优先级和它们对首次及顺序访问需求的关系。

虽然由总线接口单元执行实际刷新周期,但仍需软件在启动时设置刷新控制器参数来规定 DRAM 刷新时序。因为对图形存储器访问是有竞争的,可在内部把三个刷新请求进行排队。没有刷新且能接受的最长延时在 10MHz 时为 225 个周期(大约 45ms)。

2. 处理位图

图形处理器把主处理器的图形命令转换成专门的位表示,它由图形处理器指令中所包含的画图参数来规定。指令执行过程从装入指向指令表头的图形处理器连接地址(link address)寄存器开始。图形处理器执行指令过程一直进行到指令字中的标志(位 0)置 1 时才停止。

图形处理器把位图写入图形存储器,人们可以把位图想象为

显示像素的矩形阵列(大小可高达 $32K \times 32K$ 像素),其坐标(0,0)在显示屏幕的左上角,像素在位图中是线性组合的

系统以 4 位一个像素来存储,因而每 16 位字可分为 4 个像素。每行由像素数除以 4 个字组成,各行顺序存储。因此对给定的显示大小和每个像素的位数,位图只需最小存储器就可确定了。

图形处理器的性能很大程度上依赖于对图形存储器访问的频繁程度。若忽略指令的开销,每更改一个字需要一个图形存储器的读改写(RMW)周期,对每 8 位一个像素的分辨率,需要用一个字来存取两个像素。

两个事先确定的影响性能的因素是图形存储器带宽和一次操作所需的显示处理器开销量。如果采用 VRAM,图形处理器能用到可用存储器总线带宽的 99%。如果用 DRAM,图形处理器可用存储器的带宽依赖于对显示处理器所需的访问,此时为 50~90%。

5.1.3 编程操作

通过软件设置人们可以控制 82786 芯片,主机软件可对图形处理器和显示处理器编程,也可为总线接口控制设定参数。主机软件是通过一个 128 字节的地址块来访问 82786 的,这个地址块在 4MB 地址空间内的任何位置,它可以是存储器映射的,也可以是 I/O 映射的。这 128 个字节空间又分成三个区域。分别用于图形处理器,显示处理器和总线接口。在这 128 个字节空间中 Intel 公司保留了一些未用的字以便将来开发。

总线接口控制对系统和图形存储器的访问。8 位总线接口寄存器中包含了 82786 操作的绝大部分关键性信息。这些寄存器规定了所有 82786 控制寄存器的中断和它们的状态,DRAM 刷新和配置以及当访问图形存储器时显示处理器、图形处理器和主处理器的相对优先权。

主处理器在使用 82786 前,必须首先确定存储器的再定位和映射、配置图形存储器,以及设定刷新参数和图形处理器的优先权。在对所有 DRAM 进行足够长时间的刷新后,主处理器通过编

程使图形处理器生成多屏信息,再经过编程使显示处理器显示这些信息。

在位图内,图形处理器可以画线、圆、多边形和圆弧,BBLT 能处理任意矩形位块。

字符块传递(CBLT)以四个旋转角度将字符矩阵移入位图内。图形处理器指令的四种分类是非绘图指令、绘图控制指令、几何绘图指令、位及字符块传送指令。

非绘图指令允许访问图形处理寄存器和执行控制指令,以及设置宏指令序列。如 NOP 为空操作,LINK 是使图形处理器指向下一条指令,而 INTR __ GEN 使 82786 产生对主处理器的中断。通过 DUMP __ REG 和 LOAD __ REG 指令保存不直接寻址的图形处理寄存器(寄存器 1~22)的内容,也可以恢复以便程序调动。使用 ENTER __ MACRO 和 EXIT __ MACRO 指令可编制子程序。

每个绘图操作受绘图控制命令设置的参数支配。主要的控制是确定位图和剪取矩形在存储器中的位置,以及进入和退出检取模式。存储器位置和大小确定了所操作位图的边界,剪取矩形限制了对位图的矩形子集像素的修改。检取模式允许执行显示处理器指令而不修改位图。当指令指向现行剪取矩形内的像素时,标志置“1”,这可使在屏幕上选定目标时快速定位。

其它命令控制前景和背景颜色、线型、彩色位屏蔽和光栅操作。

彩色和线型的控制能绘制出不透明或透明的虚点线或虚线,位屏蔽可限制某一子集彩色位的绘制,使在同一屏幕上不同平面的信息可以重迭在一起。

光栅操作在 BBLT 和 CBLT 操作期间定义了各位的逻辑组合,控制命令也定义了字符集、字符指向和字符间空隙。在位图定位中提供相对移动和绝对移动两种命令。

几何命令表述起来最简单,这些指令能画点、线、圆、弧、矩形、多折线(不封闭多段线)、多边形(封闭的多段线)。

扫描线指令用于在主处理器确定边界后进行快速填充操作。块传送指令能在单个位图内为简单移动或逻辑重迭进行传送。BBLT 也能在不同位图间传送,把分别画好的图象结合起来。

字符传送以四个指向中的任何一个方向将字符矩阵装进位图。显示处理器用这些设计参数进行操作,切换窗口或显示的速度为每帧一次;一般为每秒 30~60 次且开销最小。显示处理器不是迅速的执行简单指令,而是以较低的速度执行复杂指令,这种方法很好地匹配了显示过程的特性。

5.2 TMS 34010 图形系统处理器

Texas 仪器公司的 TMS 34010 图形系统处理器(本文以下简称 GSP, Graphics System Processor)在钟频 50MHz 时可以每秒突发(burst rate)执行 6×10^6 条指令。芯片上的硬件生成位图速度比通用微处理器快一个数量级。流水线(pipeline)结构增强了 GSP 处理能力,在一个 160ns 的机器状态时间(machine-state time)内, GSP 能实现下列操作:读两个寄存器、写一个寄存器、解码一条指令、执行当前指令、并初始化或完成本地存储周期。

TMS 34010 芯片能有这样的性能是通过内部并行机构完成的,它由一个 256 字节的指令高速缓冲器(cache)、31 个 32 位的寄存器、可变宽度(variable-width)ALU(算术逻辑部件)、单周期指令、桶形移位器(barrel shifter)和屏蔽一合并(mask-merge)硬件等组成。其内部总线为 32 位,外部总线为 16 位,内置(built-in)的 VRAM/DRAM 接口和显示控制线路简化了接口设计。GSP 可通过 8 或 16 位主口(host port)使其作为主中央处理器或专用的从处理器,提供对 GSP 本地内存空间的间接访问。

5.2.1 TMS 34010 基本结构

GSP 提供独立的接口用于图形显示、主处理器和其自身的本

地内存总线。主机接口决定 GSP 是否独立工作或是作为从处理器配合主处理器工作。在作为从处理器时, GSP 主机接口在主内存和 GSP 的本地内存间传送程序、数据和图形显示单。在作单机(stand-alone)应用时, GSP 通过本地内存总线管理自己的外设。显示刷新和本地内存刷新由本地内存控制器通过对芯片上 I/O 寄存器编程控制所确定的间隔中完成。执行程序时 GSP 所用的全部指令和数据都来自本地内存。

指令通常取自本地内存并存入芯片上的 256 个字节指令高速缓冲器。当实现高速缓冲时, GSP 是在高速缓冲器当前最少使用的部分中进行存、取指令操作。指令先经过解码然后再执行流水线工作。GSP 中所有主要的通路(data path)、寄存器和 ALU 都是 32 位宽。象素大小固定为 2 的幂(1、2、4、8、16), 除象素之外的所有数据可以是 1~32 位的任意大小(字段宽度可编程)。在存入 GSP 寄存器时, 少于 32 位的数据字段(data field)扩展成 32 位。这种扩展性能也是可编程的。数据字段可以跨越字边界, 但是需要多个内存周期。

GSP 所拥有的高速图形操作吞吐率得益于并行处理几个象素。象素大小在编辑后, ALU 便筹划执行多个具有同样大小的象素进行布尔或算术操作。ALU 进位链以象素大小间隔截断、形成一串较小的 ALU。平面屏蔽(plane masking)和透明度与布尔、算术操作一起使用, 增强了图形的并行处理能力。

1. 主机接口

主处理器通过主机接口与 GSP 通讯, 其最大速率可达每秒 5MB。并为在本地内存和主机数据寄存器之间提供双向流水线的 DMA。

接口执行中有四个主机寄存器和一种主机通讯总线参与操作(参见图 5-3)。一个指向 GSP 本地内存的 32 位地址指针, 它是由链接两个主机地址寄存器 HSTADRH 和 HSTADRL 构成。主机数据寄存器(HSTDATA)装有一个 16 位数据, 它取自主机地址寄存

器指向的 GSP 本地内存的位置。主机控制寄存器(HSTCTL)含有两个 3 位信息段(message fields)、一个用于主机到 GSP 的通讯,另一个为 GSP 到主机的通讯;两位控制与主机有关的中断,GSP 暂停位,一位高速缓冲溢出位,两位依照 HSTDATA 读/写控制主机地址寄存器自动增量,一位控制高位有效字节转换(Intel 格式对 Motorola 格式)。

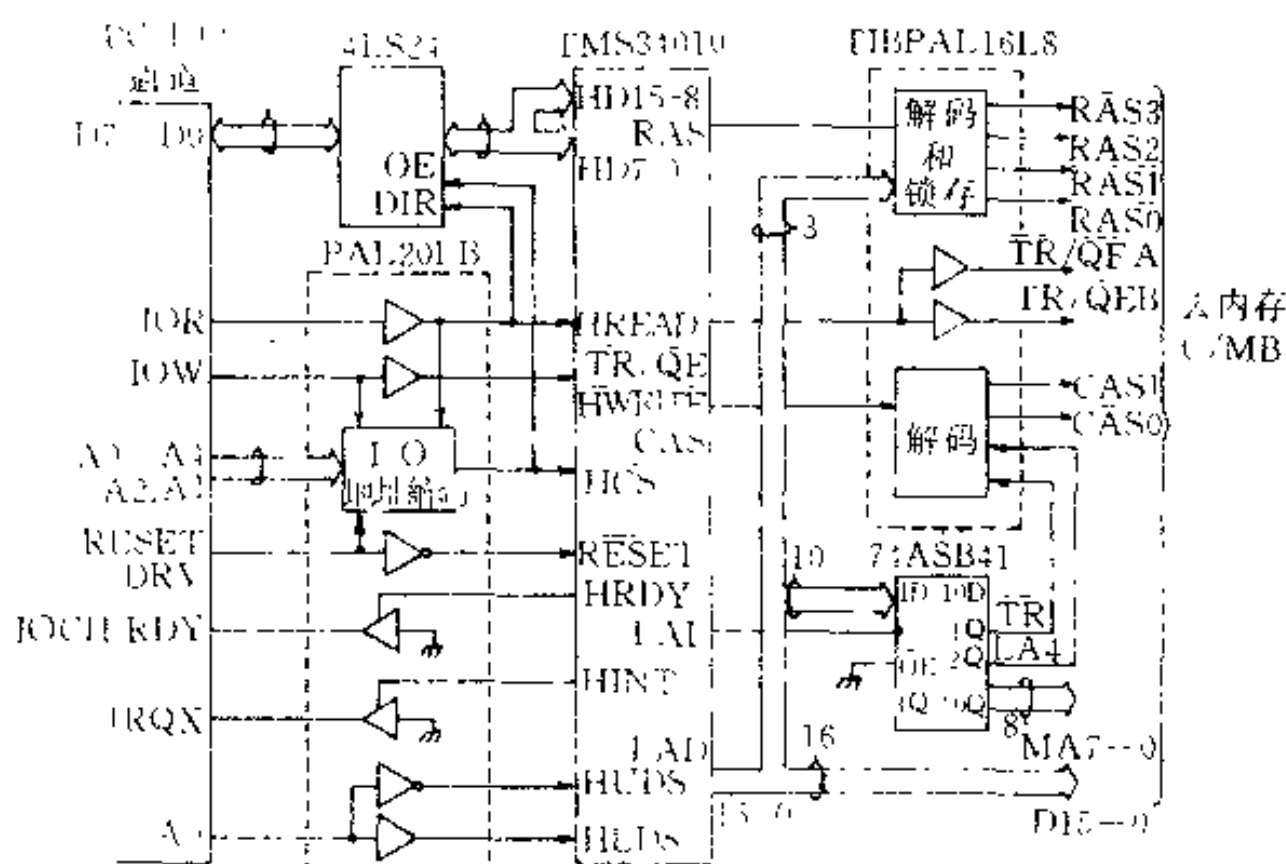


图 5-3 TMS34010 接口

主机通讯总线有 16 条数据线(HD15~0)和 10 条控制线,它有芯片选择(\overline{HCS})、读写选通(\overline{HREAD} 、 \overline{HWRITE})、高低数据字节选通(\overline{HUDS} 、 \overline{HLDS})、对主机交互就绪信号(HRDY)、功能选择(HFS0、HFS1)、对主机的中断(HINT)、GSP 复位(\overline{RESET})。由 \overline{HUDS} 和 \overline{HLDS} 规定的驱动可以使 HD15~8 或 HD7~0 在主机接口访问期间保持三态。

GSP 是通电暂停还是自由运行要看 \overline{RESET} 从低到高的变化中 \overline{HCS} 的状态。在通电暂停时,可以通过主机接口把启动程序装

入,不必采用启动 ROM。GSP 用作主处理器(象在 32 位个人机中那样)时则采用通电自由运行方式。所有主机接口寄存器和 GSP 本地内存空间中的 I/O 寄存器一样可进行存取操作。

2. 本地内存接口

GSP 有直接连接动态和视频 RAM 的接口,这种接口支持每秒 6MB 的指令读取和数据处理。这种本地接口中有:16 条三重混合地址/数据线(LADO~15),一条输入时钟(INCLK),两条输出时钟(LCLK1、LCLK2)和 12 条控制信号。

一种微码(microcode)内存控制器规定 DRAM 和显示刷新、仲裁所有器件对本地总线的访问、产生所有本地内存控制信号。除了管理本地内存接口外,总线控制器改进指令执行时序,把不同宽度数据段的数据对准。

典型的内存周期消耗两个 GSP 机器状态。第一个状态发出内存地址,第二个状态传送数据。在周期的地址部分,行列地址在 LAD 引脚上传播,伴以 $\overline{\text{RAS}}$ 、 $\overline{\text{CAS}}$ 选通。各单独 LAD 引脚上的地址位是乘 8 的偏移量,这有助于直接连结 $64\text{K} \times 1$ 位和 $64\text{K} \times 4$ 位 (256K) DRAM 和 VRAM。附加的选通($\overline{\text{LAL}}$)用来外部锁存列地址。

动态内存接口(微处理器传统上只有静态内存接口)简化了系统设计。只用 $\overline{\text{RAS}}$ 的 DRAM 刷新和 $\overline{\text{CAS}}$ 先于 $\overline{\text{RAS}}$ 的 DRAM 刷新两种内存周期都可用于 GSP 本地内存。显示刷新是由基本内存周期加以变化来完成的,其中用 $\overline{\text{TR}}/\overline{\text{QE}}$ 信号切换来表达 VRAM 移位寄存器传送周期。

在写操作时,把执行单元从本地总线和内存控制器的 32 位写队列解耦开来,以改进指令时序。指令可以把 1~32 位写入队列。控制器完成对本地内存的写操作,它能延迟需要访问本地内存的指令,以便结束前一个写操作。内存控制器自动完成位对准和数据传送操作所需的数据段的插入/取出。

3. 显示控制接口

显示控制接口提供对显示设备的直接控制(彩色监视器、激光

打印机等等)并对显示刷新的 VRAM 移位寄存器装载周期进行初始化。显示控制信号中有:时钟输入(VCLK)、水平垂直同步输入/输出(HSYNC、VSYNC),空输出(BLANK)。它们的时序由装入定时器和比较器的值来确定,这些值来自显示控制 I/O 寄存器。作为一种附加的性能,软件可以指定 HSYNC、VSYNC 为输入,这样显示时序可以与其它设备/系统,比如摄像机或多 GSP 系统。

还可以采用显示中断来使图形绘制和显示同步。GSP 内部中断当物体每帧间隔仅移动一次。显示中断是在垂直扫描线数等于显示中断(DPYINT)寄存器中的数值时发生。

显示刷新由 VRAM 移位寄存器装载周期完成,且在 GSP 本地总线上发生。显示刷新周期通常为每条显示扫描线一次,它大约占用本地内存带宽的 1.5%。显示控制逻辑在编程设计的间隔中对为这些周期提出的请求进行初始化。

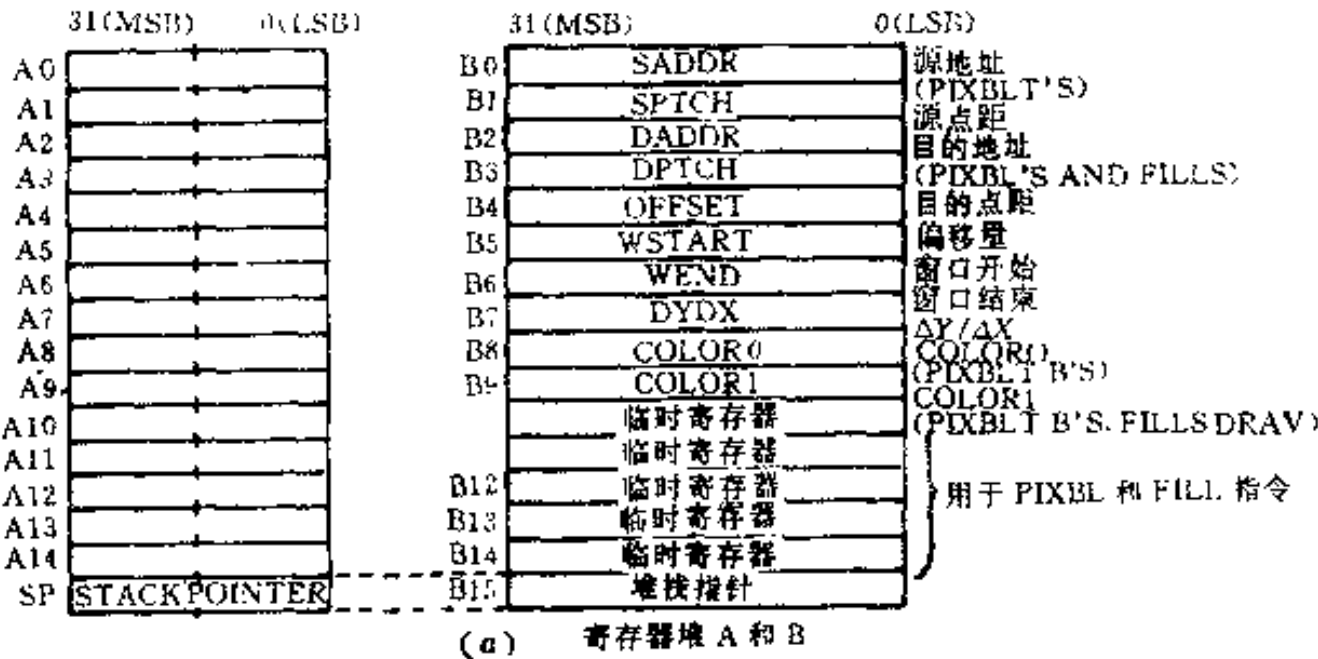
5.2.2 TMS 34010 信息存储特点

1. 指令高速缓冲

GSP 芯片中含有 256 个字节的指令高速缓冲器,它使本地总线以传递数据为主要功能。指令的读取较少与数据访问竞争内存总线带宽,这是因为大部分 GSP 执行时间花在指令高速缓冲器内部的循环中,而高速缓冲器总是可以在单个机器状态(160ns)中存取的。本地内存指令读取需要两个机器状态(320ns)。指令高速缓冲器还可以用另一种方式提高性能,这就是指令解码在前一条指令执行时就同时进行。

该高速缓冲器容纳 128 个 16 位指令字,指令由单字操作码及立即数、立即地址组成。如数据不是指令的一部分,则它总是在本地内存进行存取,而不是在高速缓冲器中。高速缓冲器还能组织成四段 64 字节。每段可含来自不同区内存的一个程序循环。存储在高速缓冲器中的指令事先取出,毫无延时。当高速缓冲器有“空缺”(miss)发生时(高速缓冲器中无下条指令),含下条指令的四字长块从本地内存中取出存入当前最少使用 LRU(Least-Recently-

Used)段中。这四字长指令块称为子段(subsegments)。每段中有8个
子段。高速缓冲控制逻辑对子段装入LRU段加以管理。LRU堆栈
指示当前最少使用的段。



当主机控制高字节
(HSTCTLH)寄存器中的
高速缓冲器清除(CF)位
置1时,软件可以禁止和
清除高速缓冲器。当高速
缓冲器被禁止时,处理器
从本地内存每次取一条指
令来执行;而清除高速缓
冲器则迫使高速缓冲器装
入新指令。当CF复位为0
时高速缓冲器有效,并立
即进行高速缓冲器装入操
作。控制寄存器的高速缓

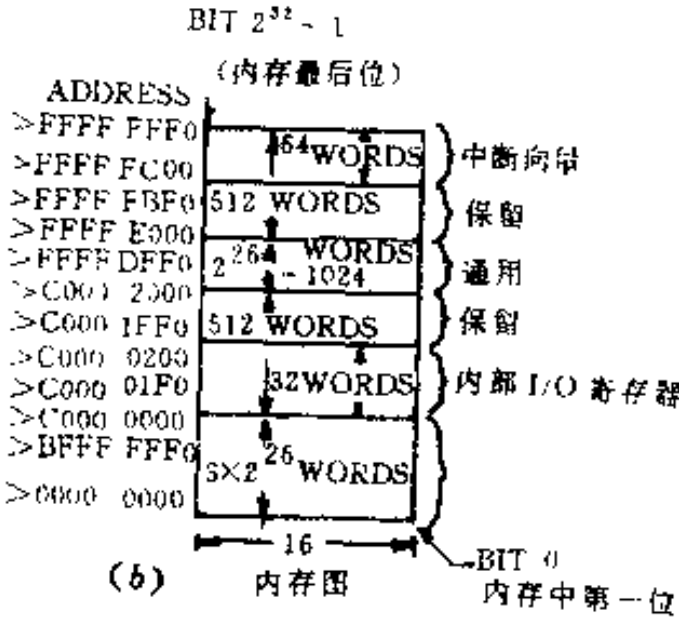
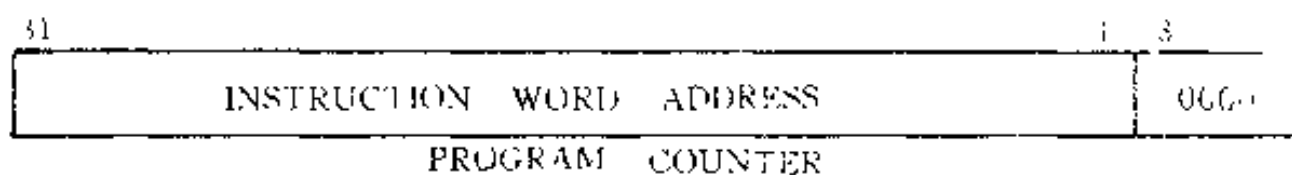


图 5-4 TMS34010 编程信息(a~d)



N- 负

C- 进位

Z- 零

V- 溢出

PBX- 执行 PIXBLT

IE- 中断允许

FE1, FE0-- 操作数小于 32 位段扩展

FS1, FS0-- 需要的段大小(1-32 BITS)

(c) 状态寄存器

缓冲器禁止 CD (Cache Disable) 位置 1 时只是暂时禁止缓冲器。当 CD 复位为 0 时, 保持在缓冲器中的指令再一次可用来执行。主机和 GSP 软件都可以处理 CF 和 CD。

指令高速缓冲器操作对软件是自动和透明的。由于 GSP 结构和指令集的协同作用, 通常不需要为高速缓冲器调整软件。因为图形指令专门化, 加上大型通用寄

ADDRESS	寄存器	
>C000 01F0	REFCNT	DRAM 刷新计数
>C000 01E0	DPYADR	显示地址
>C000 01D0	VCOUNT	垂直计数
>C000 01C0	HCOUNT	水平计数
>C000 01B0	DPYTAP	显示抽头点
>C000 01A0	保留	
>C000 0190		
>C000 0180		
>C000 0170	保留	
>C000 0160		
>C000 0150		
>C000 0140	PMASK	平面屏蔽
>C000 0130	PSIZE	象素大小
>C000 0120	CONVDP	转换(目的点距)
>C000 0110	CONVSP	转换(源点距)
>C000 0100	INTPEND	中断等待
>C000 00F0	INTENB	中断允许
>C000 00E0	HSTCTLH	主机控制(8 高有效位)
>C000 00D0	HSTCTL	主机控制(8 低有效位)
>C000 00C0	HSTADRH	主机地址(16 高有效位)
>C000 00B0	HSTADRL	主机地址(16 低有效位)
>C000 00A0	HSTDATA	主机数据
>C000 0090	CONTROL	控制
>C000 0080	DPYINT	显示中断
>C000 0070	DPYSTRT	显示开始
>C000 0060	DPYCTL	显示控制
>C000 0050	VTOTAL	视频总量
>C000 0040	VSBLNK	垂直开始空
>C000 0030	VEBLNK	垂直结束空
>C000 0020	VESYNC	垂直结束同步
>C000 0010	HTOTAL	水平总量
>C000 0000	HSBLNK	水平开始空
	HEBLNK	水平结束空
	HESYNC	水平结束同步

(d) I/O 寄存器

寄存器堆(general-purpose register files)可把操作数保留在寄存器中,所以编码循环较小。一些图形参数可以放在专用内存位置中(驻片 I/O 寄存器),这也可以免除把它们移入/出寄存器所必需的代码。参见图 5-4。

2. 内存

内存大小用位来规定,这是因为 GSP 是可位寻址(bit-addressable)且段宽度可编程(field width-programmable)的处理器。虽然 GSP 结构支持 1G 位(512MB)地址空间,第一代器件并不提供最高两位(30 位地址对应 1G 位或 128MB)。

在 GSP 内存中,数据的最低有效位总为最低有效内存地址。当访问内存时,GSP 在每个总线周期总是读/写 16 位数据。位级(bit-level)寻址对程序员是透明的。本地内存控制器初始化读改写周期、执行屏蔽-合并操作,并保持可用的下一位的行踪。

GSP 内存组成为:顶部是陷阱和矢量,随后有 I/O 寄存器和底部通用单元(见图 5-4 中的内存图)。GSP 访问 I/O 寄存器为内部解码,它们的地址不在本地总线广播。

3. 可变宽度数据

GSP 不象大多数处理器结构普遍采用的那样把数据固在一定的大小上(如位、字节、字、倍字)。GSP 支持四种类型数据结构:字节、位段(bit field)1 和 2,像素和两维像素阵。

字节大小是固定的,为 8 位,其它数据结构是用软件构造的。位段和字节可从任何位地址开始。数据字(16 位)边界无需对准。像素可以是 1、2、4、8 或 16 位,在一个存储字中相应装入 16、8、4、2、1 个像素。

最常用的数据类型是位段。GSP 同时支持两个独立大小的位段,每一个宽为 1~32 位。因为 GSP 中所有数据通道和寄存器都为 32 位宽,和每一位段大小相关的段扩展位(field-extension bit)补充空缺把数据在内部扩展到 32 位。位段大小和扩展在状态寄存器中规定。

图形由象素数据类型支持,象素大小在 PSIZE 寄存器中规定。GSP 支持所谓的“集装箱象素”(packed pixel)数据格式,把组成一个象素的所有位装在同一存储字(memory word)中,象素在内存中受字边界的约束。与此相反,平面象素组织格式把构成一个象素的各位放在存储器的不同区域(位面)。

4. 寄存器

状态寄存器包括 N(负)、C(进位)、Z(零)和 V(溢出)标志,它们根据算术和逻辑运算结果来置位或复位。还有中断允许位 IE、PBX 标志(提示 LINE、PIXBLT 或 FILL 执行被中断)和两组位段大小和位段扩展位(FS0、FE0;FS1、FE1)。当位段的扩展位设为 0 时,寄存器在装入过程中,位段用 0 扩展;反之,扩展位为 1 时,数据扩展是带符号的。

程序计数器(PC)为 32 位宽,但最低 4 位总是零。指令长度为一个或多个 16 位字,并总是对准偶数、16 位字边界。PC 可装入立即数或改为指令执行的结果(转移、调用等)和中断。PC 总是指向要执行的下一条指令的地址。

堆栈指针(SP)指向系统堆栈的顶部。(系统堆栈向低内存地址发展,因此栈顶处在堆栈的最低地址)。SP 在 A 和 B 寄存器堆中作为寄存器 15 寻址。用户可以用该 30 个通用寄存器中的任意一个来做附加堆栈。

5. 图形寄存器

与接口和图形功能有关的有 28 个 16 位 I/O 寄存器中控制寄存器是最最重要的一个。它含有控制图形处理的各位。把象素透明位(T)设置为 1,就可允许透明。透明可避免新的零值象素复盖较早的象素(在前景物体遮挡背景物体时有意义)。窗口检查(W)位段长两位,支持四种不同窗口操作,包括剪裁和拾取屏幕物体。适当设置 W,可以禁止象素写操作,也可生成中断。PBH 位决定 PIXBLT 和 FILL 是从左至右还是从右至左进行操作,而 PBV 位选择自顶至底或从底到顶来处理。象素处理顺序的控制是提供给那

些源和目的像素阵列复盖场合中使用的。

控制寄存器的像素处理(PP)位段有5位宽,可使程序员在22种像素处理操作中选择。这些操作包括16个双操作数布尔运算和6个双操作数的算术运算。像素处理在使用PIXBLT、FILL、DRAW、PIXT和LINE指令时被调用。

CONVSP和CONVDP这两个寄存器提供给GSP源和目的像素的点距(pitch)信息。点距是指在位图或像素图中两垂直相邻像素之间内存地址的差。点距在计算以X,Y方式访问数据内存地址时很有用。线性存储(连续内存地址)对采用磁盘来说是合适的,对于显示操作处理二维像素阵列以及指定数据位置用相对原点的X,Y偏移量更好。GSP使用CONVSP和CONVDP寄存器支持开窗计算,完成X,Y方式和线性方式的地址转换以及为PIXBLT调整起始角(starting corner)地址。

像素大小寄存器(PSIZE)提供GSP微指令,其中对像素位宽进行处理。这种微指令使用PSIZE的主要目的是在像素之间断开ALU的进位链(carry chain),从而使ALU能同时处理多个像素。如果在平面屏蔽寄存器(PMASK)中置入适当位,就可以在像素处理操作中保护或屏蔽像素位(平面)。平面屏蔽技术可以产生诸如文本面与图形面重迭等许多效果,这种技术是不破坏原来的有用信息在一屏上混合显示文本和图形的一种方法。

CONTROL、PSIZE和PMASK寄存器使GSP图形算法实际上独立于像素大小和处理过程。例如用一种算法画4位像素的异或运算圆和用另一种不同的算法画8位像素的或运算圆已没有什么必要。用单一的强功能画圆算法只要通过修改CONTROL、PSIZE和PMASK寄存器就可以控制像素处理(透明度、屏蔽、布尔运算和算法)和像素大小。

采用寄存器COLOR0和COLOR1减少了存储多色彩字形所需的图表区域。字形或物体的每像素一位的表示和两个彩色属性:一个“0”彩色(在COLOR0中)和一个“1”彩色(在COLOR1中)一起存

入内存中 图 5-5 表示一种 4 位扩展,每像素 1 位的形式变为每像素 4 位。

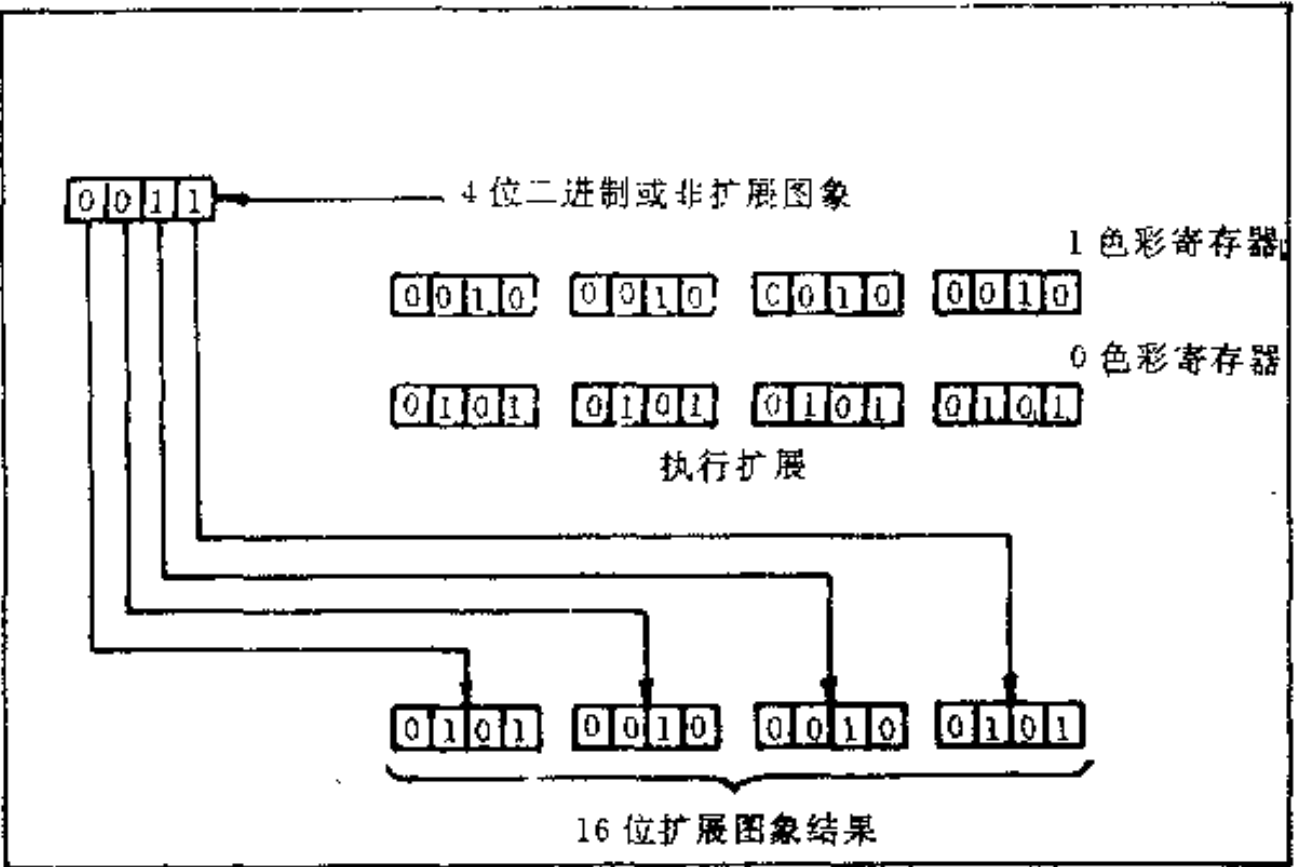


图 5-5 彩色扩展算法

6. 通用寄存器

GSP 共有 30 个 32 位的通用寄存器,且分成 A、B 两组。由于没有规定寄存器的特定功能,因此所有的通用寄存器既可用作地址指针又可作数据存储,在实际编程中非常简单。

具有众多通用寄存器的处理器对绝大多数的应用来说速度是相当快的,这是因为减少了内存和寄存器之间费时的数据交换。(片内寄存器比主内存速度更快)。

5.2.3 TMS 34010 指令特点

34010 具有完备的指令系统,其中包括针对图形操作、功能很强的图形操作指令;通用目的的浮点运算、图形和开窗等 GSP 指令;专用的图形指令和微指令。

1. 图形操作指令

在绘图操作过程中,由于要对许多变量进行处理,因此规定了一些图形操作指令。图形操作指令隐含使用 B 寄存器组来存放操作数和暂存数据。这种约定允许图形微指令直接访问影响图形操作的操作数,这种设计的另一种好处是可以中断重复的图形指令。例如:如果一条 LINE 指令违反窗口边界,则这条指令被中断,经过判断,该指令会重新启动或中止。图形指令能被重新启动的原因是当中断发生时微指令在寄存器 B₁₆到 B₁保存了临时变量。

PSIZE 寄存器使得各个 ALU 把每一个象素都当作单个的实体来处理(一个 4 位象素的第 4 位对邻近象素的第 1 位没有影响)。此外,ALU 为图形操作提供了一个单一周期,32 位桶形移位寄存器和屏蔽 合并硬件用于数据字和寄存器内改变象素位置。

对所有的图形操作指令,象素显示的最低有效地址定在屏幕的原点,由软件指定到屏幕的左上角或左下角。

2. GSP 指令

GSP 具有完整的通用指令集(127 条指令)以加强浮点运算、图形和开窗环境的功能。所有指令的形式遵循下列规则:指令的第一个字(16 位)为操作码和子序列字(subsequent word),包括立即数或立即地址。寻址方式有寄存器寻址、寄存器间接寻址、带符号位移量的寄存器间接寻址、带预减量(predcrement)的寄存器间接寻址、带后置增量的(postincrement)寄存器间接寻址和绝对地址寻址六种方式。

GSP 执行浮点运算指令的速度同使用特殊指令和硬件的浮点协处理的速度相同。浮点软件使用速度很快的(160ns)桶形移位寄存器和屏蔽 合并硬件来抽取浮点数据中的指数和尾数。在乘数和被乘数的指数和尾数被适当处理后,其结果最左边的尾数在一个机器状态周期(160ns)内由指令 LMO(Left Most Ones detect,最左边 1 检验)来检测。最左边 1 的位置将确定规格化处理尾数时桶形移位寄存器将要循环移动多少位。在规格化处理后,调整指数,使用

反序列分割指数和尾数并将它们拼接在一起。

3. PIXBLT 操作指令

PIXBLT(象素块传送)是图形指令中最重要的,因为多数图形微指令都专用于 PIXBLT 指令。PIXBLT 与 BITBLT(位块传送)紧密相关,是用布尔运算处理二维位阵列象素的一种方法。BITBLT 极适合于处理黑白显示,但不适合处理灰度和彩色显示。PIXBLT 是 BITBLT 概念的推广,用来处理二维象素阵列,其算法仍同布尔运算器。

以缩小显示图形的锯齿效果为例,这是一种显示非垂直或水平线时在屏幕上出现的受分辨率限制的现象。通过平滑(anti-aliasing)方法使锯齿边部分有逐渐的亮度变化,这样人眼看起来就自然多了。两条平滑线相交时,仅用布尔运算子来计算两个低亮度象素拼接成的象素亮度值很困难,甚至是不可能的;用 PIXBLT 采用简单算术运算符就很方便。

GSP 的 PIXBLT 提供了 16 种双操作数的布尔指令,除此之外还有加、减、饱和加(add-with-saturate)(箝位到全 1)、饱和减(subtract with-saturate)(箝位到全 0)、取小(把两个象素进行比较,取值小的那个)和取大(与取小相反)。

调用 PIXBLT 前,寄存器 B₆ 到 B₇ 必须先装入原象素阵列的地址和垂直点距、目的象素阵列的地址和垂直点距、屏幕原点位置、窗口的起角和终角坐标以及象素阵列的大小。若调用 PIXBLTB(带二进制扩展的 PIXBLT),寄存器 B₈ 和 B₉ 必须分别装入背景和前景颜色。总之,对于所有的 PIXBLT、CONTROL、CONVSP、CONVDP、PSIZE 和 PMASK 寄存器也必须装入所期望结果的适当信息。

4. 开窗

此外,GSP 还提供了两种类型剪取窗口环境的方法:后剪取(past-clipping)和前剪取(pre-clipping)。两种方法均可防止现行激活窗口外的显示内存被修改,其中以前剪取方法速度更快。

在后剪取运算中,要求对所画像素完成全部计算,然后比较像素的位置是否在可用窗口内,如果像素位于窗口外,则不画此像素。GSP 使用控制寄存器的 W 字段对所有画像素指令实现后剪取。

前剪取方法自动适用于 PIXBLT 和 FILL。前剪取通常应用于画线操作以及去除完全位于窗口之外的线和线段。这种去除功能使得图形机(这里是 GSP)保存了原本要消耗在计算看不见的像素上的所有带宽。GSP 支持 Cohen-Sutherland 算法,用专门指令 CPW 做前剪取。

CPW (Compare Point to Window)意思是和窗口比较点,它生成 Cohen-Sutherland 算法所用的剪取数据。当 CPW 与工作窗口(active window)比较线终点或线段时,GSP 硬件生成 4 位“外部码”。如果线段的两个外部码相与的结果为非零,则说明该线段完全位于工作窗口之外,无需进一步处理。反之,外部码相与结果为零则表示需要更精确的测试。

5.3 TMS 34020 图形系统处理器

TMS 34020 是 TI 公司 TMS340 GSP 族的又一成员。它在 TMS 34010 成功的基础上,进一步考虑了用户对图形处理器速度、灵活、易连接、系统扩充性等多方面更高的要求,是一种内外总线都是 32 位,完整的可编程图形处理器。

TMS 34020 很适合于工作站和微型计算机,用于高级交互用户接口,也适用于激光打印机。

TMS 34020 还适用于数据和图象的压缩,如图文传真和 CD-ROM 等。因为快速位寻址处理器(fast bit-addressable processor)比通用处理器更适合于执行霍夫曼(Huffman type)编码和解码。霍夫曼码是 CCITT FAX 标准所必需的。

5.3.1 TMS 34020 的功能和特点

34020 具有为 TI TMS 340 GSP 族定义的所有结构特点。这些特点包括主机-总线接口、带有半自治存储控制器 (semi-autonomous memory controller) 的本地总线、显示控制接口和寄存器、指令高速缓冲、31 个 32 位寄存器、单周期通用指令及多周期图形指令。(以下 34010 简称 '10, 34020 简称 '20)。

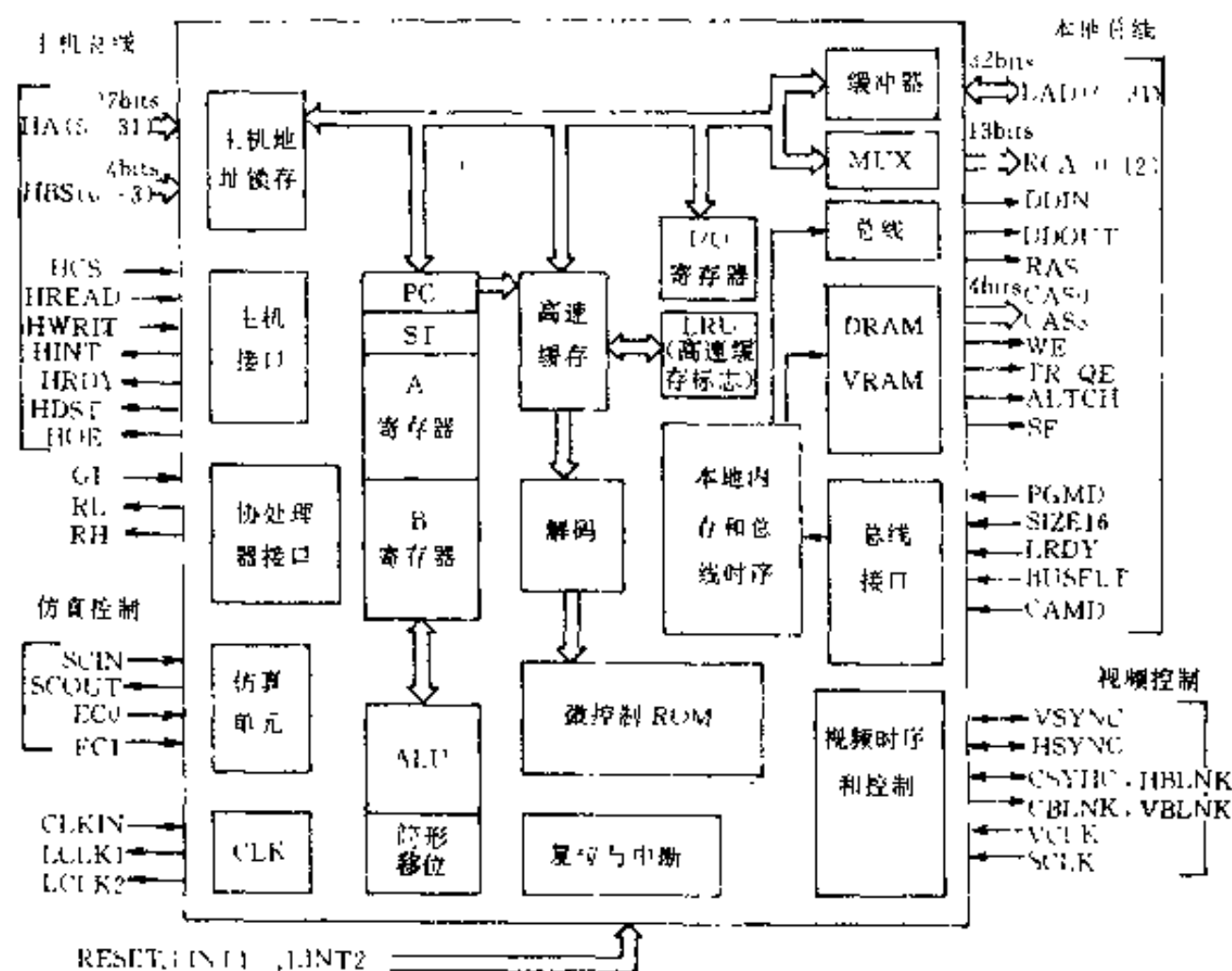


图 5-6 TMS 34020 框图

围绕着一个完整可编程微处理器和图形机 (graphics engine) 的概念展开设计,正是 '10 和其它图形芯片不同的地方,这使得 TMS340 族器件获得了成功。'10 和通常微处理器不同在于特殊的硬件和微代码,支持位映射的图形操作等。它把图形系统集成在一起,把 CRT 的控制功能,动态 RAM (DRAM) 和 VRAM 都封装在一片器件上。该设计中基本的目标是使位映射图形的灵活性和图形

处理器的灵活性相配合。

'20的定义在'10的开发后期就开始了,因此'20的设计更具鲜明的针对性和更强的实用性。首要问题是获得超过'10速度。这很明显:'20从'10的32位内部总线,16位外部总线扩展为全部32位总线。

有了32位像素和32位外部总线的支持,'20上的一些图形功能得到加强。像素格式寄存器(pixel formatted register)也从16位扩展到32位,其中包括彩色扩充(color-expand)寄存器和平面屏蔽寄存器。此外改进处理器周期时间,使用DRAM快速页面方式(fast page mode)。指令高速缓冲器加倍到512个字节,减少高速缓冲空缺,并可装入更大的算法。

34020在以下三个基本方面做了改进,大大提高了系统总线的带宽:较宽的数据总线('20为32位,'10为16位);增强的页面方式对DRAM和VRAM进行访问;较快的时钟速率或状态速率。较宽的数据总线和页面方式访问使得'20在高速缓冲空缺时(cache miss)以较少时间取较多指令。在时钟为40MHZ时,'20一般在900ns中取16条指令,而在50MHz时'10只取了3条指令。由于本地总线带宽的增加,在多周期数据处理指令FILL、PIXBLT和BLOCK MOVE改进最大。比如,'20的填充操作,完成一屏只需'10的1/7时间。如果采用44C251 VRAM的块写能力,可以达到更高的性能,比'10高50倍。对大量数据传送,'20组成一个先进先出的8字区间(每字32位)用来存取源数据进行修改,然后把它送回本地内存中的目的地。这样做进一步改善了'20的整体性能。

此外'20采用了指令混合(instruction mix)技术,使其在主要的图形运算方面比前代'10快6~50倍。利用其片内512个字节的指令高速缓冲器,'20可每秒运行10兆条指令(MIPS)।'20可直接连接第二片'20,也可连接每秒执行40M浮点指令(FLOPS)的TMS 34082图形浮点协处理器FPU(Floating-point coprocessor)।'20具有指令可执行以像素或位对准的块传送,速度达142M位/s。当

采用 TMS 44C2511 M 视频 RAM(44C251 VRAM)时,'20可以高达 1.136G 位/s 的速度填充。

为在复杂的应用程序中,使图形处理器更好地完成在主机内存空间中的操作,'20的主机接口已经完全修改了。显示控制器的 CRT 时序控制更加完善,因此可使用广播品质的 RS170 NTSC(National Television System Committee)时序。新加入的 XY 寻址功能及改进的 VRAM 控制,把帧缓冲器的扫描线合在一起,以便更好地利用存储器。'20 VRAM 功能可以利用 1M 位 VRAM 分裂式移位寄存器(split shift register)的好处,也可以利用较早(较低效率)的 256K 位 VRAM 中的行中再装入(mid-line reloading)功能。

'20有强化的 XY 寻址功能,包括可以直接支持点距(象素所在相应线间距离)而不是 XY 地址空间的二进制数。点距是两个 2 的乘幂之和。这样就使得输出程序到线性内存以及指令所支持的位面形式的存储体屏幕组织的编码更容易。此外,'20的 XY 都看成是有符号的 16 数。

'20的另一特点是 4 位 CAS 输出(见图 5-6),它支持字节写操作,避免 RMW 周期。许多图形和字符串操作需要进行大量的字节处理,可通过'20的 CAS 选通实现这一过程。另外这些选通能使'20的画线机(line-drawing engine)在一秒钟内完成 5×10^6 个象素的显示(RMW 周期会降低性能,每秒少于 3×10^6 个象素)。

此外,通过 SIZE16 输入,'20可以动态地使用 16 位和 32 位内存,同时它可动态改变其列和地址多路转换,来支持同一系统中不同阵列大小的内存且无需附加硬件。

'20的定义已经远远超出自身芯片的范围,它还包括开发极快的图形浮点协处理器 34082,比典型的微机 FPU 快 30~100 倍。

另外,在'20定义的同时就考虑了其外围芯片的设计与连接,因此由'20构成系统不仅简单而且功能很强。

5.3.2 TMS 34020 应用特点

1. 34020 与主机

'20可以象主处理器一样工作,也可以作为图形处理器挂在主机总线上。'20上的主机接口与主机系统是完全透明的。当'20的内存直接映射到主机系统的内存图(memory map)时,'20的功能就象是一种 DRAM 控制器。

当主机访问'20的本地内存时,主机的地址锁存在'20中。该地址随后改格式,出现在 LAD 和 RCA 总线上,然后去'20的本地内存(见图 5-6)。数据通过外部数据缓冲器,由'20直接控制。

38 脚接口支持高达每秒 20MB 的突发块传送或随机访问方式。'20在内部主机控制寄存器中还有选择位,可构成接口流水线以适应于块(读或写)或 RMW 操作,来谋求最高性能。

'20支持多处理机环境,可以直接访问主机内存,在访问主机资源时改进了系统性能。'20具有内置总线故障(built-in bus-fault)及重试(retry)机构处理地址冲突和总线争用(bus-contention)。'20支持数据以降序(数据从最高位开始排列)和升序(数据从最低位开始排列)方式操作,这样去除了数据重新映射的操作。

2. 占线问题

'20的 Hold/Holda 协议不仅能使一个处理器暂停主 GSP 在它本地总线上的操作,而且可以使其它多个 GSP 直接接在同一本地总线上。三条信号线 GI(Grant In 许可进)、RL(Low-priority Request,低优先级请求)和 RH(High-priority Request,高优先级请求)提供了该接口(见图 5-6)的应答机构在处理图形系统和 DRAM 及 VRAM 为基础的系统中的专门问题是必需的。

在典型的占线(hold)情况下,'20在移位寄存器传送(显示刷新)或内存刷新周期等待时必须重新判定总线控制。内部计数器会根据这些事件的潜伏情况,安排提供最佳性能。内部刷新排队需要多达 12 个 DRAM 刷新周期,限制了总线重新仲裁。

两片或更多片'20可以直接相连,来增强系统性能。多个 GSP 同步在同一本地时钟内,可以通过三线接口有效地进行应答,允许总线控制在每一本地存储器周期时在处理器之间传递信息。随着

指令高速缓冲区的增大,指令采集(instruction acquisition)效率的改进,利用该接口可以大大增强系统性能。

3. 指令特点

'20有'10的全部通用指令,并把它们与丰富的应用性图形指令(见表5-1)相结合。'20有足够必要的汇编语言指令集,以直接字段(direct field)支持强有力的C语言。

'20和'10都直接支持内存中对字段的操作。因为是位寻址的处理器,内存中的字段可以在任意位边界开始,指令集进一步支持可变字段宽度(从1~32位)。这在处理C语言的“字段”和成包数组所用内存存储中特别重要。

两操作数的像素块传送指令 PIXBLT,对一对两维像素阵,执行16个二进制运算,6个算术光栅运算。两操作数的 PIXBLT 指令在处理和搬移有边界的目标及单色操作时很有用。

表 5-1 34020 新增指令

图形指令	
CUIP	剪裁 DADDR.DXDY 到窗口
CVSXYL RS,RD	转换源 XY 地址为线性地址
FLINI Z	快速以线性寻址画线
FPXLQ	寻找等同 COLOR0 的像素
FPXLNF	寻找不等同 COLOR0 的像素
GETPS RD	取像素大小
HPFILL XY	水平填充图案
LCOLOR	VRAM
LINIT	线初始化
PIXBLT B,M,L	屏蔽 PLXBLT: 二进制到线性
PIXBLT L,M,L	屏蔽 PLXBLT: 线性到线性
PPROX RS,RD	处理寄存器中像素
RPXL RD	复制像素
SETCDP	依据 DPTCH 寄存器设置 CONVDP I/O 寄存器
SITCMP	依据 MPTCH 寄存器设置 CONVMP I/O 寄存器
SFTCSP	依据 SPTCH 寄存器设置 CONVSP I/O 寄存器
VBLI	VRAM 像素块传送

VFILL VRAM 用已处理像素填充,线性

协处理器指令

CLXLC	S,com,i	协处理器内部操作,短型
CLXLC	S,com,i	协处理器内部操作,长型
CMOVCG	RS,RD,S,com,i	从协处理器移到 34020 寄存器
CMOVCG	RS,com,i	从协处理器移到 34020 寄存器
CMOVCM	*RD,K,S,com,i	从协处理器到存储器间接搬移(先减量)
CMOVCM	*RD=,K,S,com,i	从协处理器到存储器间接搬移(后增量)
CMOVCS	com,i	从协处理器移到 34020 状态寄存器
CMOVCC	RD,com,i	从一个 34020 寄存器移到协处理器
CMOVCC	RS,RD,S,com,i	从一个 34020 寄存器移到协处理器
CMOVMC	*RS,K,S,com,i	间接搬移(先减量);存储器到协处理器
CMOVMC	*RS=,K,S,com,i	间接搬移(后增量);存储器到协处理器
CMOVMC	*RS=,RD,S,com,i	间接搬移(后增量);存储器到协处理器

其中,i=协处理器 ID(0-7)

K 要搬移的参数计数

S 操作值大小(0-32 位,i=64 位)

com=协处理器指令段

搬移指令

BLKMOV 块搬移

通用指令

RMO RS,RD 最右端

程序控制和上下文切换指令

IDLE 空闲;等中断

MWAIT 存储器等待

SWAPE *RS,RD 交换段(读改写段)

TRAPLN 软件中断,长型

用多个 PIXBLT 以及算术光栅运算、二进制扩展运算等可以仿真更精巧的操作。比如采用二进制扩展,字体以位模式存储,构成彩色正文(含透明),双色花式操作可在位图上进行。由于 PIXBLT 指令的灵活性,使人们可以建立象 8514/A 比较运算这样的操作,前景、背景的光栅运算也一样。此外,还可用它支持非矩形目标。

'20扩充了'10的 PIXBLT 指令,加进了第三个操作数,即二进制屏蔽操作数。当'10硬件支持矩形窗口区域和象素块时,许多应用程序已开始要求对非矩形窗口区域进行控制和显示。'20的第三个屏蔽数组支持这种要求。屏蔽是用逐个象素的方式为基础来描述窗口区域。这样描述的区域就特别灵活,它可以是填充的,也可以是凹的、凸的、构轮廓的、有孔的等等。



图 5-7 三操作数 PIXBLT

三操作数 PIXBLT 用二进制屏蔽(binary-mask)数组,对第一操作数,源象素数组进行操作。该屏蔽数组在源数组中选择独特的象素和目的数组结合。对正文来说,源数组可以是任意模式(闪烁的或中间色调的)来描述正文的“颜色”或“影调”(shading)(见图 5-7)。屏蔽数据包含字符的形状或雕像。目的数组是画字符的可见“画布”。

以这种方法,屏蔽中值为 1 的各位选中源数组模式中各象素(1、2、4、8、16 或 32 位深)把它们传送到目的位置。还有附加功能在 22 种光栅运算中取出一种对所选的源或目的象素进行操作,包括一些透明度和平面屏蔽处理。

1. 光栅运算

TMS 340 族的光栅运算包括所有两位的布尔组合和它们的反及算术运算。后者支持象素加法、减法和一对比较运算。首先计算源和目的象素的 \max 值,其次计算最小象素值。这些计算可用于对象素颜色所在各色平面的分层处理。它们已经用于显示一种比如游戏的物体,它在整个屏幕跑动而不破坏背景或前景图象。象素通过二进制或算术光栅运算组合构成结果象素。

'20的光栅运算有几种新的透明模式。透明在有些场合中用来禁止把新的象素数据写入基于光栅运算实际操作数信息的目的数中去。这样,透明对在通常情况下目的象素的写入有影响。

'20有四种透明模式:透明禁止、结果象素透明=0、源象素透明=0、目的数透明=COLOR。其中前两个与'10兼容。

在透明被禁止时,光栅运算如通常一样。当结果象素透明为0时,'20检查所产生的结果象素值,从光栅组合中查得零值,这些零值用初始的目的象素值取代,因此目的象素位置不变。

当源象素透明为0时,在零值象素光栅运算之前检查源象素值。确认每一零值,然后使目的象素保留它的原值。这在使用其它'20光栅运算时保持源象素阵中透明象素时很有用。

当目的数透明为COLOR时,在等于“背景”颜色(存储在32位COLOR寄存器中)的象素作光栅运算前检查目的象素值。在光栅运算中不变。这对维持目的位图中背景象素是很有用的,不用考虑源象素值是多少,也不管采用了22种光栅运算中的哪一种。

5. 画线操作

有关画线方面的操作'20比'10功能更强,这突出表现在,快速画线、线初始化、支持花式线和图案

为了加强画线功能,'20对'10的Bresenham算法加入了快速画线指令FLINE,它执行与'10完全一样的任意角算法,但要假设将预先剪切的终点作为输入。这允许CPU略过窗口检验微码而以存储器带宽速度执行画线算法为核心(光栅运算,平面屏蔽,透明全都有效)。

为了减少线初始化时的开销,LINIT 指令为 LINE 指令做好必要的准备。LINIT 加速并简化了与线初始化有关的操作开销。非常重要的是,LINIT 允许利用 LINE 指令将较大的算法装入到高速缓冲器中,于是增强了画线性能。

由于增强了花式线的绘制功能,'20增加了一种水平填充指令 HFILL,它支持由 32 位内部寄存器描述的图案。这种重复的图案取自寄存器,如果寄存器存的是一个非零图案,象素就被画出,旋转。PATRN 寄存器含有当前绘制位置,这由该指令尾部的位 0 表示。

5.3.3 34020 与 VRAM 44C251

因为'20和 44C251 1M 位 VRAM 是同时制定开发的。因此,'20支持 44C251 的性能和系统特性,这使得两种结构可以配合在一起使用。

这种联合定义的重要性可以最清楚不过地从块写方式 BWM (Block-Write Mode)看出。44C251 有四组 $64K \times 4$ 位内存,内部总线为 16 位宽(见图 5-8)。在通常方式中,地址的最低两位选四组中的一组(每次只能选择一组)并由四条数据线给出数据。在 BWM 中,四条数据线用来允许将数据写入 44C251 四组中的任一单元。需要写时,数据由专门的彩色锁存数据多路转换后提供。'20控制 44C251,在 VRAM 中把 32 位数据通路扩展为 128 位宽(见图 5-9)。

BWM 功能模仿 TMS 340 族的“二进制到色彩扩展”功能。色彩扩展是通过把数据转换为控制信号来选择是否写色彩的。因为每条数据线控制一个多重位(multiple-bit)色彩值,我们采用术语“色彩扩展(color expansion)”。

'20和'10所有灵活的色彩扩展器,把每象素 1 位的形状信息转换成每象素 1、2、4、8 或 16 位。'20 一直扩展到每象素 32 位。这种功能广泛应用于位映射彩色图形中。

虽然 BWM 有潜力给出更强的性能,如提高四倍或四倍以上进行填充和彩色正文显示,但是这需要'20中专门的寻址和控制以

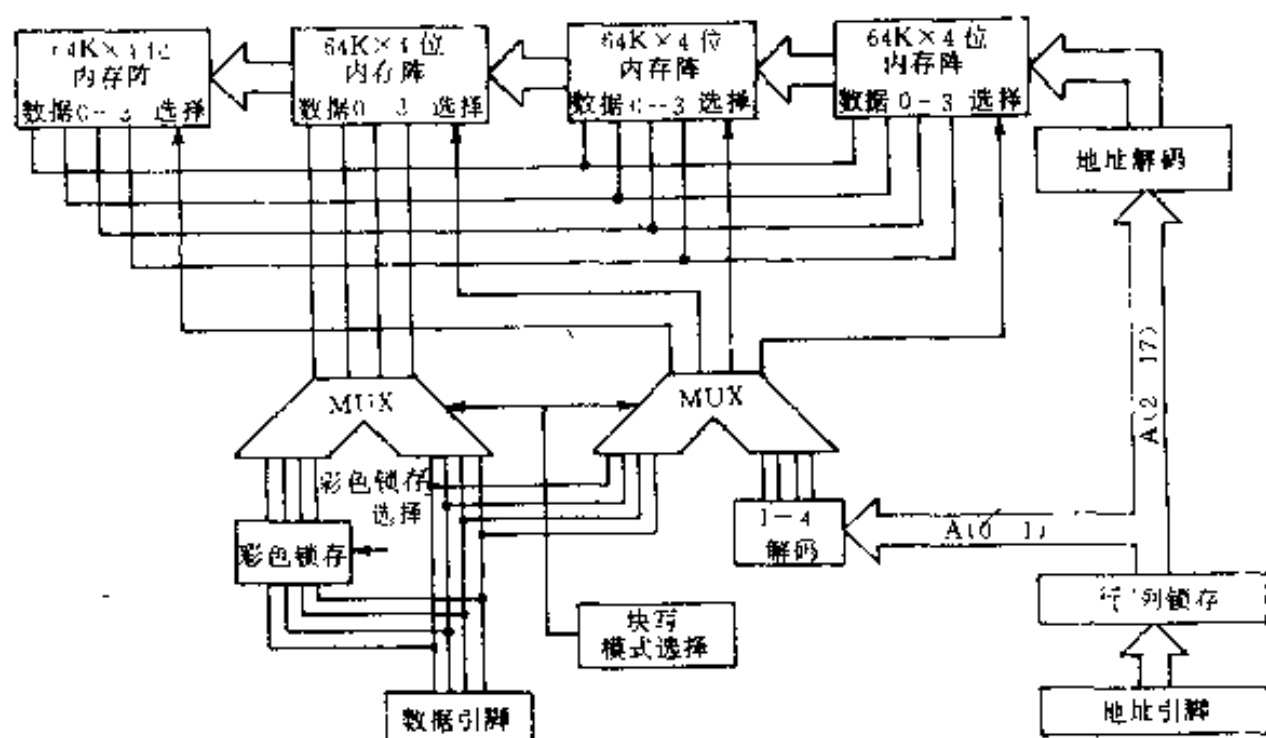


图 5-8 TMS 44C251

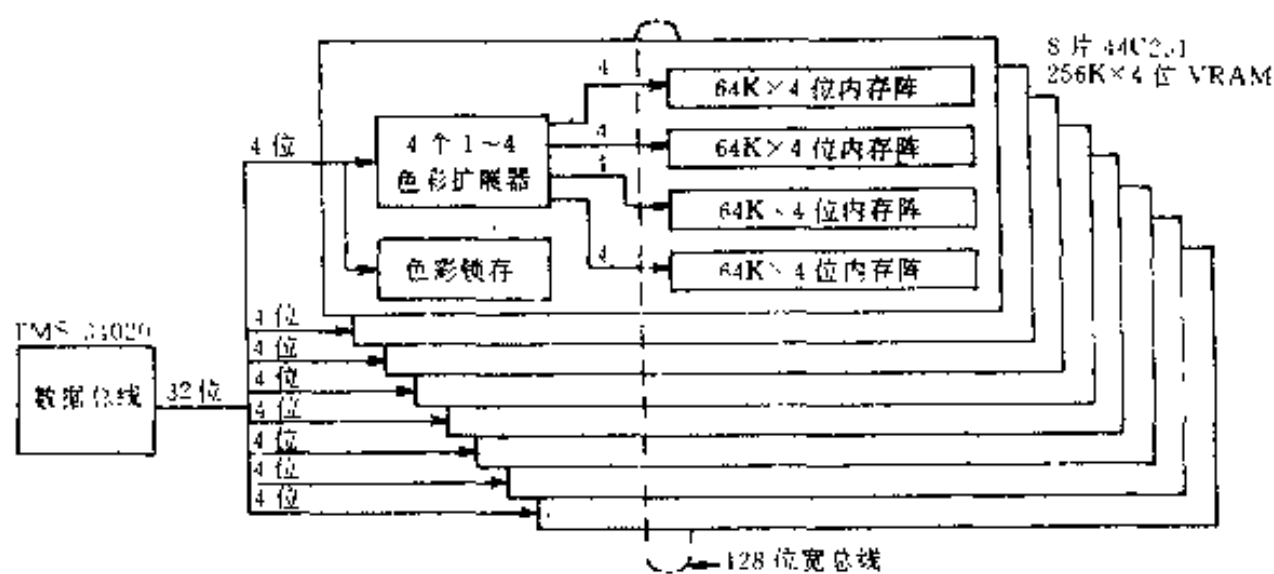


图 5-9 44C25 与 '20 连接图

及数据映射硬件。这种映射硬件使得象素写入内存和无 BWM 时写入内存相一致,用这可以解释块写操作的特性。

当采用 BWM 时,可以想象每个 44C251 VRAM 中有四个 1~4 位固定的色彩扩展器。换句话说,每个内存可把进入的 4 位数据或控制信息扩展为 16 位。把两片 VRAM 组合起来(得到 8 位宽色彩锁存器和扩展器)就是四个 1~8 位扩展器。

采用'20的色彩扩展硬件和芯片上附加的硬件产生与 VRAM 接口所必需的控制信号,'20的 32 位数据总线在 BWM 周期时可被有效地扩展为 128 位。无 BWM 时,'20依然支持内存色彩扩展功能,这仍比无色彩扩展硬件的微处理器改善大约 10 倍。

'20中加入两条专用指令以便利用 BWM 的优势。VFILL 用来快速实心填充(fast solid filling)。VBLT 指令读出二进制形状式信息并控制 VRAM 进行正文及图案色彩扩展。

还有一种改进是有关平面屏蔽功能的。44C251 的“每位持续写”加速了该功能。'20不必做读改写(RMW)周期来执行屏蔽操作。

把较宽数据总线、快速页面方式、较快的处理器周期和'20上 BWM 支持功能结合起来,在写彩色正文时,速度比'10 高 50 倍。而'10的内部色彩扩展硬件已经使它比通用处理器在上述功能情况下好了许多。

44C251 的分裂式移位寄存器大大简化了当移位寄存器仍处在移位时,对它进行装载的时序要求。'20直接支持这种专门的控制和时序,解决了系统设计者为之头痛的对时间要求特别紧的实时重装问题。

5.3.4 34020 与图形浮点协处理器 34082

34082 是专为 34020 开发的高性能图形浮点协处理器。它集成了微定序器(mirosequencer)、寄存器堆、浮点算术逻辑部件和乘法器(见图 5-10),支持高级图形指令,执行浮点运算高达 40MFLOPS。

'82上的寄存器堆由 22 个 64 位寄存器组成。它有 10 个 A 堆寄存器(A-file register),10 个 B 堆寄存器及 2 个 C 堆寄存器用于

反馈。这个大型寄存器堆可使复杂算法在内部存储中间结果,缩短了为进一步计算再装这些操作数所需要的存储器周期。该器件也组成有本地总线用于存储高达 64K 32 位字,以及对数据和微代码进行扩充。这种扩充存储使用户开发自己的功能或把数据在'82本地存储。

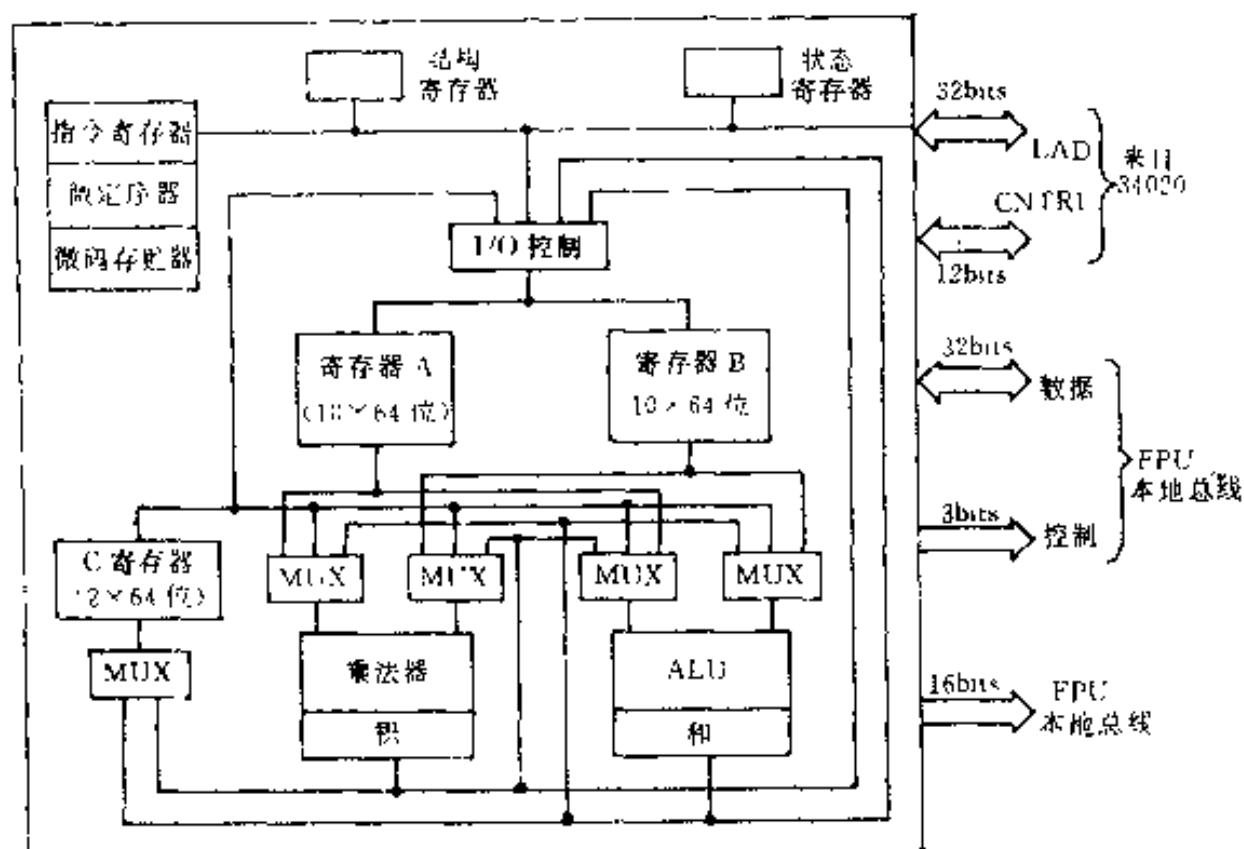


图 5-10 TMS 34082 浮点协处理器框图。

利用'20的协处理器接口,'82可和'20直接接口不需要“连接”逻辑。可以在系统中留有'82的插座以便日后补充。该接口允许把四片'82和'20相连。'20提供一个通用的协处理器接口允许在'20片外实现高级功能以改进系统整体性能。'82正是利用该接口提高性能的一个协处理的例子。该接口由五个基本周期组成,用于数据通信和控制:

1. 把数据从'20寄存器搬移到协处理器。
2. 把数据从协处理器搬移到'20寄存器。
3. 把数据从存储器搬移到协处理器(间接'20寄存器搬移)。
4. 把数据从协处理器搬移到存储器(间接'20寄存器搬移)。

5. 执行协处理器命令。

'82不仅支持完全兼容的 IEEE 标准 754-1985 加法、乘法、除法、平方根和比较,它还有为复杂图形计算所用的指令。表 5-2 给出了全部指令清单,此外'82 的扩充能力允许用户创建任意多的自定义功能。

表 5-2 34082 FPU 指令集功能

一操作数操作	
Absolute Value 绝对值	1's Complement 二进制反码
Square root 平方根	2's complement 二进制补码
二操作数操作	
Add 加法	Divide 除法
Subtract 减法	Compare 比较
Multiply 乘法	
约定(常规)	
Integer to single 整数到单型	Single to integer 单型到整数
Integer to double 整数到双型	
Double to integer 双型到整数	Single to double 单型到双型
Double to single 双型到单型	
矩阵运算	
4×4, 4×4 乘法	3×3, 3×3 乘法
1×4, 4×1 乘法	1×3, 3×3 乘法
图形操作	
Backface testing	背面测试
Polygon clipping	多边形剪取
Polygon elimination	多边形消元
Viewport scaling and conversion	视图比例和转化
2-D Linear interpolation	二维线性内插
2-D Window compare	二维窗口比较
2-Plane Clipping(X, Y, Z)	二平面剪取
2-D Cubic spline	二维三次样条
3-D linear interpolation	三维线性内插
3-D Volume compare	三维体积比较
2-plane color clipping(R, G, B, I)	2 平面彩色剪取
3-D Cubic spline	三维三次样条

图象处理

3×3 卷积

链接操作

Polynomial expansion 多项式扩展

1-D minimum / maximum 一维极小/极大

Multiply / accumulate 乘法/累加

2-D minimum / maximum 二维极小/极大

矢量操作

Add 加 Dot product 点积

Subtract 减 Cross product 叉积

Magnitude 幅值 Normalization 正规化

Scaling 定标 Reflection 反射

5.4 视 频 RAM

视频 RAM (VRAM) 是一种改型动态 RAM (DRAM), 它从存储器阵向一个独立的内部串行移位寄存器进行大量的位传送。在 VRAM 之前, 常规帧缓冲器 (围绕 DRAM 设计的) 的可用带宽, 其主要部分用于显示刷新。这样一来处理器实际上已没有时间更新显示数据。

在 VRAM 的位传送之后, 移位寄存器的内容独立移出并送给显示。结果, VRAM 的随机端口 (random port) 可以不受阻碍地对图形处理器进行访问。随着图形显示分辨率和存储器件密度的增高, VRAM 对帧缓冲器设计来说是必不可少的。

最初的 64K 位 VRAM 组织成 $64K \times 1$ 位器件, 并有单一的 256 位移位寄存器, 它有四个抽头点 (移位寄存器的开始列位置), 存储器阵至移位寄存器传送周期只能在水平回扫、移位寄存器的串行时钟脉冲停止时进行。

第二代 VRAM 是 256K 位器件, 把四个 64K 位 VRAM 的功能结合起来做在一块芯片上。这种器件组织成 $64K \times 4$ 位, 有四个

256 位“串行移位”寄存器。一个静态 RAM 和一个模仿前代移位寄存器功能的计数器,允许移位从任何抽头点开始。

改变移位寄存器至存储器传送的时序可以实现一行中间的传送。这种“行中再装”允许设计利用较少 VRAM 完成一定大小的帧缓冲器;但是在这种情况下时序非常严格,很少有系统可以利用它。

最近的 VRAM 是 1M 位器件,这种器件在结构上做了若干增加以改进其性能。它的组织是 $256K \times 4$ 位,四个 512 位移位寄存器。有些厂家计划一种 $128K \times 8$ 位 VRAM 替代品。以下情况必须引起重视:当图形结构采用宽于 128 位的数据总线时,有些存储器件会只用其中的一部分,因为它们需要额外的器件只是为了达到所需数据总线的宽度,但由于乘 4 存储器的组织“较深”,为了补偿存储器深度的增长,1M 位 VRAM 包含有改进性能而无需更宽总线的特性。

大部分 1M 位 VRAM 支持块写方式(BWM),它使主要图形操作如填充、彩色正文生成等性能提高四倍或更多。快速页面方式访问(在 1M 位 DRAM 中也普遍使用)进一步改善了存储器带宽。有些 1M 位 VRAM 支持“每位持续写”(persistent write per bit),它使彩色屏蔽锁定在 VRAM 上,减少了读改写(RMW)周期。1M 位 VRAM 还支持“分裂式移位寄存器传送”,为串行移位寄存器实时再装简化时序和必要的控制电路。分裂式移位寄存器在非 2 的幂显示分辨率时可节省存储空间。

BWM 是 1M 位 VRAM 的新结构特点。在 $256K \times 4$ 位 VRAM 上,一个内部 4 位色彩寄存器在专门写周期时从 4 位宽的数据总线上装入。在每个像素 4 位时,该锁存器含有送写的色彩值;在每像素 8 位时,两个 VRAM 连在一起构成一个 8 位寄存器。

当 VRAM 控制信号选择 BWM 时,地址的最低两位被忽略,4 条数据引脚控制把色彩寄存器的内容写入内存中的任意一个位置或全部四个连续位置(即一块)。数据线上的“1”启动对色彩寄存器

值的写操作,而“0”则阻止向块内对应位置进行写操作。四条数据线控制 4 位存储单元,净效应是使写带宽增加四倍。在 BWM 中支持单独像素控制的能力,以用来显示彩色正文和区域填充。

第六章 图形工作站

工作站是用于专业目的,可以单机工作,但特别适合于连网工作,具有强大计算能力和图形能力的计算机系统。它拥有技术人员所需的各种计算机工具,包括 CAD、大规模数学建模等专门应用程序;也包括通用工具,用于编程、写作;并共享局域网和广域网上的信息。

工作站提供真正的多任务操作,用户不用花费很多时间等待计算机完成一种任务。当有较大任务运行时,用户可以插入运行另一个任务而不至明显减慢系统的响应时间。

从表面上看,高档微机和低档工作站似乎已经重迭。在硬件规格性能上高档微机似乎毫不逊色。但是 CPU 时钟频率、处理器的 MIPS 以及诸如 Dhrystones 的测试结果并不能说明什么问题。这是因为工作站性能是按仔细平衡 CPU 和操作系统吞吐率优化的。系统中每个部件按其极限运行。由于把设备和进程的速度和数据传输率集合在一起,提高了系统总体的吞吐率。在多任务处理多进程操作时,工作站的优势愈发明显。所以离开多任务和连网环境来评价工作站是毫无意义的。

不过,微机使用方便,有少数应用程序超过工作站。比如电子表格就是一个很好的例子。诸如 Lotus 1-2-3 这样的 MS-DOS 程序得到广泛应用,在技术界留下深刻印象。

现在有的工作站提供微机仿真,使管理人员和其它个人机用户也可以使用工作站,也使留恋微机软件传统的工作站用户得到一点安慰。

6.1 图形系统的发展

6.1.1 微机与工作站

要想预测微机的前景,看看工作站的情况就行.现在微机中普遍拥有的特性,如窗口、网络、32 位 CPU 等曾是以前标准工作站特性的一部分。

从工作站转移到微机的下一种主要的技术是复杂、通用的图形硬件。比如说,微机采用大容量帧缓冲器、专用的图形变换处理器。和工作站相比较,微机仍然缺乏三维显示方面的标准和支持。

虽然并非所有系统都需要快速显示三维物体的能力,不过象 CAD(计算机辅助设计)这样的场合,如果能很快看到三维设计,工作效率就会大大提高。快速的三维显示系统可以使设计者从任何方向、距离来观看所设计的模型,也可以对它进行旋转、剖切等处理,还可以改变色彩和光照条件等等。目前微机在有限的基础上也可以实现类似效果,但它通常用专用硬件、专门的非标准图形调用来实现。与此形成鲜明对照的是,工作站采用工业标准,诸如 PHIGS 或 GKS。

图形工作站和微机在实现三维功能方面可以明显看出结构上的区别。在显示三维模型时,首先要把有关物体的信息转换成屏幕显示用的二维表达形式。计算机系统(微机或工作站)中若无专门的三维用硬件,这种转换就得由 CPU 来完成,这是非常费时的。

图形工作站通常包含专门的浮点处理器,称转换机(transform engine),见图 6-1,它可大大加速转换速度。

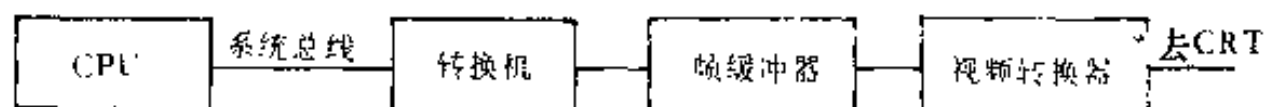


图 6-1 典型工作站图形子系统

工作站系统主 CPU 把物体有关信息传递到转换机,该专用浮点处理器把传来的信息转换为二维屏幕图象并存入图形子系统的内存一帧缓冲器中,然后由视频转换器读入帧缓冲器中的内容,最后生成视频扫描信息,送往 CRT。

图形处理流水线(pipeline)的设计随系统而变。在多处理器结构中,如果 CPU 速度足够,转换可由系统主处理器完成而不用辅助的专用机。由于微机应用的局限性,也就不需要广泛采用图形转换硬件。实际上,并没有多少应用软件需要它。不过,与专用三维硬件的情况相反,微机中显示存储器则日见庞大。

微机 VGA 支持 640×480 象素分辨率,即整屏大约为 3×10^5 个象素。高档图形工作站则至少提供 1×10^6 个象素,典型的分辨率为 1280×1024 。高于 VGA 的微机显示系统(如 Super VGA, TIGA)不仅昂贵,而且缺乏标准,结果只有很少应用程序可以利用这些高档微机显示系统的优点。

图形工作站不仅有较高的分辨率,而且常有不显示的屏幕内存区,称为隐内存(hidden memory)。它作为一种显示内存暂存器,有助于快速形成位映射正文(即区别于字符发生器产生的字符、图象字符)、纹理及图案拼花。这是因为这些操作效果依赖于存储块传送速度,从屏幕内存的一部分传到另一部分当然比从主内存到屏幕内存要快得多。除了屏幕内存本身存取速度比主内存快以外,主要原因是无需把主内存使用的位格式转换成屏幕内存的位格式。

目前使用较大的隐内存组的做法越来越普遍,事实上,有些系统允许用户选择屏幕内存的某一部分为可见的内存区。当用户使用很大的位图时,可以通过依次选择可见的内存区来实现比系统显示分辨率还要大的图象的显示。

6.1.2 帧缓冲存储器组织

工作站的帧缓冲存储器组织是目前微机不能比拟的,也是微机发展的前景。

1. 屏图深度

屏图深度(screen depth)是指象素所包含的信息位数。表示屏图深度的通常方法是标明可用的平面数。每个象素含有 8 位的系统称八平面系统。近年来,工作站中使用多平面系统发展很快,有些系统每个象素含的位数以十计。见图 6-2。

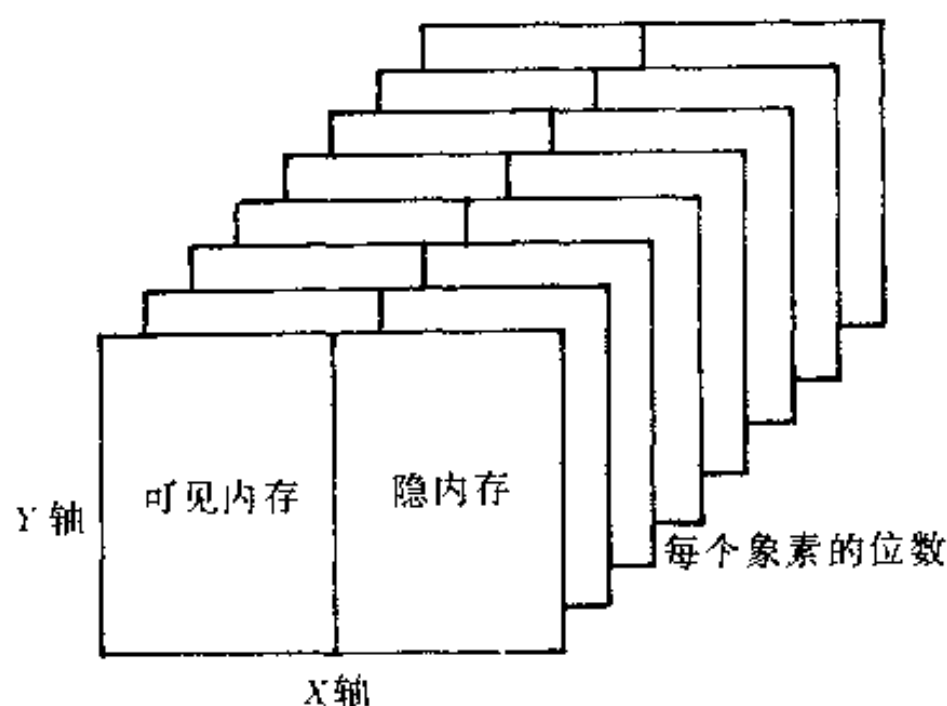


图 6-2 多平面帧缓冲器

单平面的图形显示只能是单色图。这里所示系统每个象素含 8 位,可以提供 256 种选择(256 色),不显示的那一部分隐内存可以用于快速生成位映射正文、纹理和图案。至此,这样的结构形成的彩色通常称为伪彩色。因为肉眼可以很容易地区分出调色板中相邻色的不同。目前 24 平面系统可以精确地标出整个可见光范围的色彩,这样的系统称为真彩色系统,可以产生 1600×10^4 种色彩。

帧缓冲器并不总是固定在一种格式上,比如 640×480 象素 24 平面的帧缓冲器与 1024×900 象素八平面的内存量是一样的。

一些图形系统可以在多个模式下工作。微机上具有 24 平面的很罕见,并且因为无工业标准,很少软件可以利用这些产品的长处。

2. 双缓冲

双缓冲(double buffering)是又一种需要附加屏图内存的技术,它被用来平滑快速变化图象的动画效果。即使在具有快速硬件的条件下,就是表现简单模型,由于不断重画物体也会引起明显的闪烁。这种闪烁的原因是系统开始在下一个位置画物体时需要清屏。为了消除闪烁,双缓冲让系统在不显示的屏图内存中作图,然后将主副图象缓冲区加以切换,就可以立刻把该物体在新位置上重现出来,避免由于清屏及重新开始象素依次显示形成的闪烁。

这种双缓冲技术需要一些平面(副平面)作为隐内存。如果是八平面的伪彩色系统,则采用双缓冲时需要 16 平面;如果是真彩色系统,则需要 48 平面。只有当第二图完成后,才能通知视频系统切换可视内存区来显示该图。这种技术并不是为了多画几帧,而是为了形成真实的运动感。理论上在微机上也可以产生双缓冲的伪彩色图,而实际几乎没有软件加以支持。

3.2 缓冲

为了理解 Z 缓冲是怎么回事,首先简要地说明一下图形工作站是如何建造三维物体的。

一个三维物体通常以相邻多边形拼接造型,构成实体(solid)的。即使是平滑的曲面(如球面),其内部表达也为某种多面体。曲面细分得越小,看起来越平滑。在确定其它因素后,显示程序利用专门的光照和阴影算法来消除小平面(facet)效果。当然这里还没有牵涉哪些象素或多边形被别的阻挡了、因而实际上是见不着的。

因此 Z 缓冲实际是以屏图深度为条件,确定象素是否可见的技术。

转换机把模型的三维坐标系统转换为二维屏幕坐标,因此每个象素除 X、Y 坐标和色彩外,还有深度值(或 Z 值)。这样视频屏幕上理论平面中的一连串二维多边形重迭在一起。这些多边形需要以深度排序。为实体造型设计的图形工作站,诸如 Apollo DN 1500 DVS、HP9000 系列 835 Turbo SRX 及 Silicon Graphics Personal Iris 具有专用的硬件来解决这个问题。没有相应硬件的系统必须

把该排序工作加入到应用程序中去,这样才能依次画出多边形而避免把远处的多边形画在近处的多边形之上。这种排序有多种算法,由于模型有大量的多边形,处理起来是很费时间的。一种十分简单的方法是,保留图中每个像素的 Z 值,把它放入称为 Z 缓冲器的专用内存中,提供给下一步消除隐藏面的工作。

1. 隐藏面消除技术

消除隐藏面的基础是列出多边形表。实体造型软件经过二维多边形表,处理对应图的每一个像素。如果像素的 Z 值比已有的大,这就是说它在已有的像素后面(相对于工作站前用户而言),因此不用显示。如果 Z 值表明它在已有的像素之前,那么它就应该显示出来。结果是所显示的像素颜色变成刚处理的那个,显示的 Z 值被更新。按这种方式,依次处理,最后结果是和原始模型一致的视觉效果图。

需要多少额外的屏图内存来代表这些深度值呢?如果每个像素有 8 位 Z 缓冲,则可有 256 个深度值,对于简单的模型来说已经足够了。当然它不能代表物体所有的每一个可能的深度。需要考虑所涉及物体的最大和最小 Z 值。如果涉及的三维物体为 5cm 见方,那么在一个 10cm 的深度范围内是安全的。8 位 Z 缓冲器把该深度范围分成 256 份,增量为 $1/3\text{mm}$ 。取舍误差可能产生少量错画的像素,但绝大部分场合下,表达是精确的。

目前高档图形工作站提供大得多的 Z 缓冲,有的高达 32 位,超过 40 亿不同的深度值,足以应付需要。

人们经常错误地认为,较大空间的模型(比如不是 5cm 见方,而是 50m 见方)需要较大的 Z 缓冲。其实在较大模型中,如果都牵涉大尺寸,则 Z 缓冲需求很小。只是当所涉及的模型在深度上既有大尺寸、又有小尺寸时,问题才严重起来。

假设有一个模型由两个小物体用一根 50m 的圆杆连结起来,对该小物体要用 $1/2\text{mm}$ 的深度分辨率,则对 50m 米说超过了 100 000 不同的 Z 值,或至少是需要 17 位的 Z 缓冲器。当考虑某些

较大模型时(如空间站的部件小到螺栓,大到太阳能集电板),大深度的 Z 缓冲是很重要的。好在用双缓冲时,并不需要把 Z 缓冲加倍。一旦图在一个缓冲器中画成后,Z 缓冲内存可马上清除,为另一个缓冲器所用。

5. α 缓冲

Z 缓冲为的是跟踪象素的深度值,而 α 缓冲是用来给象素赋加透明度值,这也称作混合(blending)。8 位 α 缓冲可以确定 256 种透明度,通常已经足够了。 α 值为 255 时表示物体不透明,就是说靠近观察者的象素会彻底遮盖它后面的所有象素。

如果 α 值为 200,此时象素大约为 22%透明,就是说绘出的新象素含 78%的本色,22%的后透色。如果 α 值为 0,物体 100%透明,则无需绘出。

α 缓冲并不真正是透明物体造型的准确方法,因为它并没有涉及反射和折射。比如说,一个高透明度值的透镜状物体并没有放大镜的性能。要达到实际效果,需要采用光线跟踪算法。 α 缓冲技术的有用之处在于既能表现固体内部情况,又不至于大大损失外表的信息。

图形系统可以利用 α 缓冲来平滑直线或边缘线的显示效果。由于屏幕分辨率的限制,一条斜线或边缘线在显示时往往呈锯齿状,利用 α 缓冲使前景和背景之间变化比较平滑,就可以消除斜线或边缘线的锯齿效果。

6. 重迭平面

重迭平面也是屏图内存的附加平面,常用来保持光标位置或注释正文等。重迭平面可以修改而不破坏屏幕的原有图象,就象透明胶片重迭起来似的,非常方便有效。每当图形系统扫描显示内存产生当前图象时,它也完成了重迭平面的信息。重迭平面的显示优先于屏图内存的其余部分,在该平面上凡色彩值为零的象素不予显示。

以光标为例,在没有重迭平面时,每当移动光标越过屏幕面上

已有图象时,系统必须改变光标越过的那些位,而后再把它们恢复原状。从系统性能来说,这样做开销大,而用一个重迭平面来跟踪光标位置显然很方便。完全可以把该重迭平面想象成只含有光标象的透明层。如果希望光标和正文具有多种颜色,可以增加重迭平面数。四个平面可实现 16 种颜色,用于光标和正文,通常这已足够了。

除了重迭面(overlay plane)外,有些系统具有衬垫(underlay)面,主要用于背景图象,它们用得较少。

7. 控制平面

图形系统也可以利用一些位面来跟踪诸如窗口关系等。从现在的标准来看,没有窗口也就无所谓工作站,窗口是工作站必须具有的特性。屏图内存经常需要进行不同的构置来适应不同窗口的各种用途。比如说,一个窗口只需要画线功能,另一个需要完全的三维功能,而第三个需要单色正文,这就需要控制。

系统需要采用附加位面来标志象素的窗口类型,这样可以正确地对它进行处理。用于这些簿记功能的位面称为控制平面。

一个只有 30 平面左右的系统可以实现在一个窗口显示真彩色图象,在另一窗口中用双缓冲演示 12 位伪彩色动画。这两种窗口都使用相同数量的位面。

8. 发展趋势

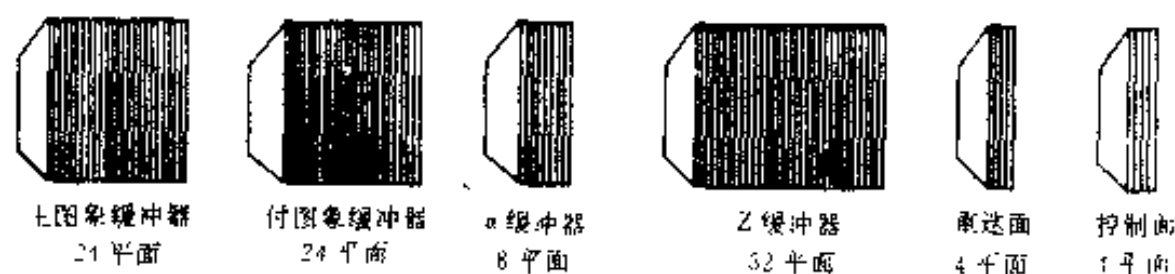


图 6-3 帧缓冲器结构示意图

如图 6-3 所示,帧缓冲器可由许多位面组成,需要巨量内存。图示

结构中总共为 96 平面,设屏幕分辨率 1280×1024 象素,这就是说屏图内存总共为 $125\ 829\ 120$ 位或近 16MB!

上述各种存储技术中,Z 缓冲和 α 缓冲专用于三维显示,其他的则用于几乎所有的应用场合。值得指出的是,带宽是整个图形流水线中最紧要的。流水线式的绘画过程的各个步骤都要能互相衔接上,否则局部的速度和性能的改进毫无意义。

随着新技术的发展,存储器费用不断下降,所有计算机的图形能力将会建立在不断增大的帧缓冲器上。

6.2 “PC 兼容”工作站

6.2.1 概 述

IBM 微机的巨大成功,对工作站的影响也是明显的。不少厂家推出的工作站系统可以在 IBM PC 仿真模式下运行 MS-DOS 应用程序。这种仿真应用实现了在微机和工作站之间共享数据,并且可以把软盘作为传输媒介。当然,作为工作站,它有自己的特点,其主要特点如下。

1. 计算能力

无论把工作站用在什么专门场合:工程、印刷或科学分析等等,它需要一定的硬件性能来保证合适的计算能力。

首先是 CPU,工作站普遍需要 32 位地址空间,执行速度至少每秒 1×10^6 条指令(1MIPS)。此外还有可供选择的浮点处理器、阵列和图形处理硬件等来进一步提高性能。

Apollo Domain Series 3000、Sun 3/160 和 Xerox 6085 这三个系统都是具有 IBM PC 仿真模式的所谓“PC 兼容”工作站。Apollo 和 Sun 系统采用 Motorola 68020 微处理器,系统速度相应为 1.3MIPS 和 2MIPS;Xerox 6085 CPU 采用 Mesa,速度为 2MIPS。

多任务操作环境需要数兆字节主存,基于磁盘存储的虚拟存

诸系统及网络资源。硬盘存储容量从 20MB 开始, 多的超过 1GB (gigabyte)。

2. 图形

大屏幕、高分辨率、快速的图形显示是工作站最明显的特点。特别是图形显示速度把工作站和微机的性能明显区分开来。工作站的图形可以实现实时转动多色多面体, 并带有在选定的光照条件下逼真的光影效果。这还不是全部, 可以设定它在一个窗口内显示, 而在第二个窗口进行高速数据传送, 在第三个窗口上翻动正文文件。

显示屏幕大都用 48cm (对角线尺寸), 允许同时显示几个窗口。屏幕上可以同时并排显示两页 22cm (8.5 英寸) × 28cm (11 英寸) 的窗口内容。

对工作站来说, 屏幕分辨率最低为 1024 × 800, 有的可高达 1280 × 1024。彩色系统至少能从多于 4096 色调的调色板中同时显示 256 种不同色。这种色彩能力在工作站表现光影, 天然色和质感等方面是必需的。

图形速度的一种测量方法是二维向量绘制速度。工作站的速度从基本系统的每秒 5000 条到加有加速装置的每秒 40 000 条, 目前速度还在不断提高。

为了提供一个灵活多样的平台, 以便运行各种商品图形软件, 工作站通常支持多种图形标准, 诸如 ANSI CGI (Computer Graphics Interface)、GKS (Graphical Kernel System)。这种标准灵活、扩展性好, 足以在各种系统上运行不同应用程序。

3. 用户界面

工作站功能强大, 系统复杂, 使用起来当然也不简单, 部分原因是 UNIX 命令比较复杂, 键盘也增强了。

微机中广泛应用的鼠标/窗口界面是源于工作站的, 不过它们在质量和功能上依然有差别。工作站首先采用光鼠标, 更重要的是工作站用户界面灵活。人们可以通过键盘命令或用鼠标调用图标

(icon)来完成大多数功能。而作为基于鼠标微机的主要代表 Macintosh(苹果公司的麦金塔什)主要依靠鼠标操作。

4. UNIX

大多数工作站采用 UNIX 操作系统,它由 C 语言写成,被认为是软件开发的理想环境。它提供强有力的程序工具,用于编程和文件处理。

UNIX 特别适用于工作站。作为工作站心脏的 32 位微处理器,尤其是 Motorola 68000 系列,非常适合 UNIX。首先,线性地址空间大(32 位地址),不需要管理大量存储段指针来处理进程映象。其次,这些芯片的双操作状态(dual operating states)可以方便地实现 UNIX 的用户(user)模式和内核(kernel)模式。

此外,微处理器的通用寄存器可以从主存装入经处理后存回,简化了在 UNIX 上下文转移时操作系统状态的变化。而且在寄存器处理数据时无需消耗存储器周期,加速了 UNIX 上下文转移的进程。

工作站微处理器不包含内存管理指令,因此实现操作系统的余地很大,在这样的硬件环境下当然希望象 UNIX 这样灵活、紧凑的系统。

5. 连网

工作站的一个本质特点是连网工作,进行通信和共享数据。其工作量很大,微机上很少见。工程的“工作群”(work group)概念是指有多个单机分别针对一个大项目的不同方面协同工作,因此高性能的网络是高效工作站的一个本质特点。

工作在局域网的通信速率可达每秒 10~12M 位,和每秒 230K 位的 AppleTalk 网相比,差别是很明显的。正因为有较高的速率,就实现了透过网络交通共享存储器资源,把它作为虚拟存储的一部分,给工作站每个进程虚存空间 64~256MB。连网的另一个用途是把处理工作分配给网上的其他系统。比如说,在网上找出空闲的系统,远地登录,然后开始执行任务。

CAD 和 CAE 图形应用程序所需的巨大数据库只能通过这种虚拟的无限的存储器和处理能力方能正常工作。相比之下,PC 网络常局限于简单的数据传送和不太频繁的文件共享。

6. PC 仿真

工作站通常是用硬件和软件结合的方法来实现 PC 仿真的。一般都包含有 80186 或 80286 微处理器的线路板,以及 RAM、360K 或 1.2MB 软盘驱动器等。

IBM 键盘、屏幕图形适配器及 I/O 通道由软件仿真。Apollo Domain Series 3000 的 PC 仿真比较彻底,它包括 AT 兼容总线:7 个 16 位 AT 兼容插槽和 1 个 8 位 XT 兼容插槽。系统模仿 IBM 单色和 CGA 显示系统,Xerox 6085 只模仿 Hercules 卡。

从软件方面来讲,兼容性是很有意义的,可以运行各种应用程序,包括 Microsoft 公司的 Multiplan 和 GW-BASIC 以及 Xyquest 公司的 XyWrite 等。

6.2.2 Apollo Domain Series 3000

Apollo DS 3000 以装备 IBM PC AT 兼容总线及扩展插槽为特点。这种低档工作站速度为 1.2MIPS,采用三重总线内部结构,基于 12MHz Motorola 68020 CPU 和 68881 数学协处理器。外设有 1.2 MB 软盘驱动器,86MB(未格式化)ESDI(Enhanced Small Device Interface)硬盘,39cm 和 48cm 彩色和单色高分辨率显示器,Domain 及 Ethernet 网络硬件和软件。

1. 总线结构

DS 3000 的 CPU、数学协处理器和存储器管理单元(MMU)都挂在 32 位内部“虚拟总线”上。CPU 生成 32 位虚拟地址和 3 位功能码。虚拟总线在位 26 处“捆扎”起来(27~31 位不予考虑)。虚拟地址一旦到了总线上,或转换、或通过 MMU 传送到 68881 浮点协处理器或中断向量生成电路产生 68020CPU 所需的中断响应周期。

系统的 CPU 控制寄存器和状态寄存器、中央处理器、请求分

页的 MMU、浮点单元、DMA 控制器和页面寄存器都在一种专用的、或物理的内部总线上。中断控制器、串行 I/O 口、系统时钟日历、网络识别 PROM 和启动 PROM 以及主存(多至 8MB 节)也是这样。该专用总线支持 32 位数据(CPU 控制、状态、和页面奇偶寄存器只有 16 位宽)和 16 位地址。DMA 控制器及页面寄存器、中断控制器、串 I/O 口、系统定时器和日历、网络识别 PROM 和启动 PROM 均为 8 位器件。

2. AT 兼容总线

所有外设控制器(磁盘、图形、网络)和 IBM PC AT 选项插槽都挂在 AT 兼容总线上(含 6 个 AT 兼容插槽,1 个 XT 兼容插槽)。微机协处理器 PCC(Personal Computer Coprocessor)支持两种地址空间: Intel 定义的 16 位 I/O 地址空间(64KB)和 1MB 存储器地址空间。为了适应 DS 3000 16 位地址并为 AT 总线上 I/O 设备提供每页存取校验(per-page access checking)和保护,把 AT 兼容 I/O 地址重新映射到 DS 3000 自己的地址范围。操作系统把每组 8 字节地址映射到不同的但连续的 1024 字节页的首个 8 字节。AT 兼容总线支持所有采用 20 或 24 位物理地址的兼容设备。

分立的 MMU(DMMU)是 Motorola 68851 分页存储器管理单元的一个兼容子单元。68020 CPU 发送 32 位虚拟地址,而 DMMU 只转换 26 位。在映射或虚拟模式中,DMMU 执行地址校验并转换 26 位地址为 24 位物理地址,这样 CPU 可以利用它们访问物理存储器或设备寄存器。在非映射模式,虚拟地址的低 24 位成为物理地址。DMA 页面寄存器提供 DMA 控制器所需的地址位来访问 24 位地址空间。DMA 控制器可把数据传送到主存或 AT 兼容总线上现存的存储设备。

所有 Domain 设备映射在 64MB 虚拟地址空间的高 4MB 区,也称“全局管理”区或“全局 B”。在全局 B,平常使用的页面可由过程共享,避免每个进程页面的重复。

3. AT 兼容协处理器

Domain PCC 由 8MHZ Intel 80286 CPU, 可选的 80287 浮点加速器, 1MB 驻板内存构成, 做在 IBM PC AT 总线兼容的电路板上。图形能力包括 IBM 单色监视器仿真、IBM CGA、Hercules 图形卡。在 PCC 窗口中运行的应用程序可以对 DS 3000 1.2M 软盘驱动器进行读写, 格式和 IBM PC 的相同。此外, 也可以使用其他所有 DS 3000 资源, 包括文件、网络打印机等。

4. 图形支持

图形系统包含 39cm 和 18cm 彩色、单色监视器。图象内存为 512KB 用于存储位图, 由四平面彩色显示控制器使用。图象内存中每个字代表某一平面中的 16 个水平像素。

虽然图象内存含有 1024×1024 像素, 显示分辨率实际只有 1024×800 像素。没有用于实际显示的图象内存用于存放字体和其他图形元素。与此相仿, 单色图象内存 (256KB) 可含 2048×1024 像素, 但屏幕分辨率为 1280×1024 像素。额外的图象内存用于其他图形功能。

DS 3000 图形资源还包括: 对一些公用图形标准, 比如 GKS 的支持; Apollo GSR (Graphics Service Routines) 是一系列图形子程序和支撑; Tektronix 4014 图形终端仿真程序; GMR (Graphics Metafile Resources) 为一系列二维及三维子程序及图形数据库。

5. 操作系统

DS 3000 有 Aegis 和 Domain/IX 二种操作系统。

表面上看, Domain/IX 操作系统象是 Berkeley 4.2 和 AT&T System V UNIX 操作系统两者的结合。实际上, 它是一个面向目标的操作系统, 它的内核称为核 (nucleus)。该核是一种较高层的管理程序, 其任务是页面替换、页面故障控制、较低层进程之间位流传送的控制。在这种情况下, Berkeley 4.2 外壳、AT&T System V 外壳、Apollo Aegis 操作系统及象语言解释程序、显示管理程序这样的应用程序都被处理成核控制下的“目标”。数据可以在目标之间传送而核不需要知道数据类型。

6. 语言系统

Domain 语言系统包括 FORTRAN77、ISO Pascal、Domain/C 和 Domain/LISP 编程语言,以及辅助程序员工作的实用程序。C 和 LISP 编译程序包括重要的扩展,以便利用 Domain 系统的优点。

程序员可以按照具体问题选择不同语言来编写大程序中的不同部分,然后再结合成单一的应用程序。

其他软件工具包括软件工程环境和用于用户和网络通信的电子邮件程序。

6.2.3 Sun-3

Sun-3/160 是当时性能价格比适中的工作站,速度为 2MIPS,采用 16.67MHz Motorola 68020 CPU 和 16.67MHz 68881 数学协处理器。

工作站围绕 Motorola VME 总线结构设计,它有 16 插槽用于 VME 总线兼容线路板,规格是三高度、四深度 Eurocard 格式。3/160 的硬件结构有两种内部总线。一种是高速、同步 32 位 P2 总线,它处理对主存和浮点加速器的存取。32 位 VME 总线处理所有的 I/O 操作。

系统每个进程可用高达 16MB 主存和 256MB 虚拟存储器。而图形缓冲存储器不算系统主存,它不能象主存一样寻址,而由图形处理器控制。

3/160 装有 8MB RAM。中央处理器、图形处理器、Sun IPC 处理器、同步磁盘控制器和 RAM 占了总数为 16 的插槽中的 8 个。

Sun 3/160 图形处理器和可供选择的图形缓冲器支持 48cm 单色有灰度的和彩色的监视器,分辨率为 1152×900 象素。图形处理器可以在任何窗口从 16×10^6 种色彩的调色板中选择同时显示 256 色。

1. 磁盘

3/160 有一个四通道 SCSI 和 ESMD 兼容的同步磁盘控制器。硬盘容量从 71MB 到 1.1GB。280MB ESMD 盘的平均存取时间为

20ms,数据传输率为每秒2.4M位。575MB盘速度更快,平均存取时间为18ms。系统一般装有360K、1.2M软盘驱动器,60M、0.635cm(0.25英寸)磁带备份系统。

2. 操作系统

Sun 工作站采用 Sun OS 操作系统,它是 UNIX 的一种版本,它在 Berkeley 4.2 BSD 的基础上作了一些改进。它把 System V 功能加入到内核库和实用程序中。

Berkeley 4.2 和 System V 是 UNIX 操作系统的两大代表,它们的大部分功能可以结合成单一的操作系统,不过它们之间的差别依然是明显的。由于一些功能不能组合, System V 版本以分离的目录或库的形式提供。在 Sun 工作站上开发的程序和 Berkeley 及 System V 都兼容,可以利用两种操作系统外壳功能的优点。这种双功能的特性使得在 System V 上开发的程序可以比较容易地在 Sun 工作站上使用。

Sun OS 支持网络文件系统 NFS(Network File System),它用于远地执行、文件上锁、与网络上其他节点透明文件共享等。由于采用 NFS,虚拟存储器的概念也进一步扩充了,可以允许越过网络请求分页。每个单独的进程可以使用高达256MB的虚拟存储器。

Sun 工作站支持 Ethernet(以太网)软件和硬件及很多通信协议,诸如:OSI、SNA3270、BSC3270及 BSC RJE 和广域网络协议包括 X.25和其他网间发送程序。

3. 用户界面

Sun 采用工作站普遍使用的窗口和鼠标环境。3/160工作站使用三键光鼠标。左键选图标,中键调整或修改选择,右键激活菜单。

Sun 的界面颇有特点,人们可以用鼠标选择命令,把它们拷贝到命令行中去。这样,当人们熟悉系统后,只要少量击键就可执行通常不敢想象的、复杂的 UNIX 命令。

4. 协处理器硬件

Sun 把它的 AT 兼容协处理器称为 Sun IPC(Integrated Personal

Computer for Sun workstation),即为 Sun 工作站提供的一体化微机。IPC 硬件包括10MHz Intel 80286 CPU、8MHz Intel 80287数学协处理器、一个并行口、1MB 双口本地存储器(存取时间100ns)可用于 IPC 软件、MS-DOS 3.1和 PC 应用程序(640K 用于运行 PC 应用软件)。

IPC 有两个仿真串行口以每秒2400位的速率工作,一个逻辑硬盘(动态调整大小),两个仿真打印机口(LPT1和 LPT3)和仿真的 EMS(Lotus, Intel 和 Microsoft 定义的扩展内存规范)。仿真的打印机口实际上输出到装在 Sun IPC 板上的串行口。象 Sun-3/160这样的连网系统,所有软硬件包括 Sun IPC 板在内,都可通过网络共享。

Sun IPC 运行的窗口可以由键盘命令 Pctool 或图标来选择。一台 3/160至多支持四块 IPC 电路板。在本地系统(UNIX)命令行中键入 pctool 就可用第一块板。如果想用别的板,只要指定板的 ID (标识)就可。人们可以通过在 Pctool 窗口命令行键入节点名和 IPC 板标识,对远地 Sun IPC 进行访问。

Sun IPC 在单用户或多路存取结构中都可使用。多路存取结构可以是本地存取,也可以是远地存取,用户可以无限,不过只能是一次一个。实际上还有一些限制,比如说,数据不能在 IPC 窗口之间传递。因为每个 IPC 窗口把文件存在自己的逻辑盘上。要解决这个问题,可以把 IPC 文件传到网络服务器上,由网络管理程序来安装含有所需 IPC 文件的 NFS。这样两个 IPC 都可以共享该文件系统。

5. IPC 外设

Sun IPC 外设包括单个或两个软盘驱动器(360KB 的和1.2MB 的)、NFS、仿真 Microsoft 并行 PC 鼠标、并行口(在 IPC 板上)、仿真的三种图形显示适配器——IBM 80×25单色、720×348 CGA 及 Hercules 720×348单色。

Sun 键盘通过键盘映射技术来模仿 IBM PC AT。Sun 随 Sun

IPC 提供一幅图来指明 Sun 本机键盘和 IBM PC AT 键盘的对照。

Sun IPC 支持可直接连在 IPC 板上并行口的 IBM PC AT 外设,也支持 Sun 工作站的仿真串行口和逻辑设备 LPT1、LPT2 和 LPT3。Sun IPC 没有硬件的 AT 总线,这样,通过 AT 总线连系的 PC AT 外设就不能和 IPC 一起使用。

不过 Sun 工作站支持三种打印机作为逻辑设备 LPT1、LPT2 和 LPT3。LPT1 是 Sun IPC 的缺省打印机,它可以是 Sun 激光打印机或其他高速打印机,人们可以把 ASCII 文件送往 LPT1 打印。

LPT2 是应连在 Sun 工作站并行口上的并行打印机。LPT3 是工作站的缺省打印机。LPT3 的缺省打印机是 Sun 激光打印机,它会自动地把 Epson 格式的文件转换成 PostScript 格式。这样 LPT3 既可处理图形,也可处理正文。LPT3 的文件转换对用户来说是透明的。

象 DS 3000 一样,Sun 系统的用户是专业性的。可用的语言包括 C、FORTRAN77、Pascal 和汇编。Sun 的另一个特点是说明文件很出色,对没有多少 UNIX 经验的用户是很有吸引力的。

Sun386i 是少数采用 Intel 80386 一类芯片的工作站之一。它的突出特点是可同时在不同窗口执行 MS-DOS 程序。它那复杂的监控程序允许这些程序使用工作站的硬盘和内存而没有冲突。除此之外,Sun386i 依然和 Sun 的其他机型一样,是一种 Unix 工作站,它的操作系统采用 SunOS4.0 (BSD4.2/4.3 Unix 和 System V.3 Unix 的结合产物),它提供 NFS 能力,基于窗口的用户接口 Sun-View,及便于使用的诸如 Sun Organizer 和 Color Editor 等实用程序。

在 Sun386i 的 DOS 窗口中能可靠地运行 XyWrite II plus、Lotus 1-2-3 2.01、dBASE II Plus 1.01、Designer 1.0 等等,但 Sun 386i 终究是 Unix 工作站,运行 PC 程序是它的特点,而不是特长。

以 Sun386i/250 工作站为例,开一个 DOS 窗口时,性能比 compaq386/20 强;开两个 DOS 窗口时,性能稍低于 IBM PS/2 80;开三个 DOS 窗口时,性能和 IBMPC AT 相近。虽然 SunOS 并不限制开

窗数,不过同时使用三个窗口大概是到顶了。

Sun 386i 中有 PC 扩展插槽,当然要有 UNIX 经验的人来做系统软件配置,让它接受 PC 扩展卡。

6.3 RISC 工作站

6.3.1 概 述

1980年专业人员面对的是低性能的微型计算机和昂贵的小型计算机。当时的微机没有足够的 CPU 功能、内存和共享资源,显示分辨率也远不能满足处理工程问题。小型机倒是具备这些性能,除了价格昂贵外,还缺乏交互特性和个人系统的独立性。

1981~1982年,Apollo Computer 和 Sun Microsystems 公司填补了这个空档,它们的产品就是通用的工作站 Domain DN 100和 Sun-1。这些系统具有高速 CPU、大量 RAM、高分辨率显示、大容量硬盘还有网络连接不同工程工作站和共享绘图机、打印机及文件服务器等资源。

1983~1986年,Apollo 和 Sun 得到迅速发展。它们把系统卖给 OEM 公司,这些公司把系统配上软件构成专用的工程系统用于电子工程、航空航天设计、机械 CAD、人工智能、计算辅助软件工程、出版、图象、三维实体造型、甚至财务等。当然,自1981年开始,许多公司也进入了工作站市场。

当时,不同公司所研制的低档工作站非常相象,通常有32位 CPU、4~8MB RAM、40~80M 硬盘、1M 象素显示、10M 波特连网、操作系统具有请求分页、多任务虚拟存储。

后来的工作站大部分是围绕 Motorola 68000系列微处理器构作的,Apollo、Sun 和 HP 公司都是这样,而 DEC 公司的 VAXstation、IBM 的 RT PC 等则采用专用芯片,此外,还有少量基于80386的系统。

各家公司都想方设法提高速度。最简单的方法是采用较高时钟频率的微处理器,比如把16MHz 68020系统更新为25MHz 68020系统。第二种方法是采用较高档的芯片,比如把68020系统改进为68030系统。最后,工作站 CPU 朝专用芯片发展,通常采用精简指令集计算机 RISC(Reduced Instruction Set Computer)原则。比如 Sun-4 (SPARC)、IBM RT PC (RISC ROMP)、HP 9000 系列 800 (HP Precision Architecture RISC)等都是这样。就是 DEC 公司也不例外,它的 VAXstation 采用 Micro VAX 芯片,而后也采用 RISC 技术。采用 RISC 技术的工作站的速度有明显的提高,目前的工作站速度为几十个 MIPS,比以前几个 MIPS 提高了一个数量级。

Sun-4 采用一种新的 RISC 微处理器,它是 SPARC (Scaleable Processor ARChitecture),含有 CPU 和 FPU,它和两片 Weitek 公司的 FPP 芯片一起以 16.67MHz 运行,速度在 10MIPS 和 1.6MFLOPS (Millions of Floating-point Operations Per Second),结构中还有高速缓冲和控制器提供 1GB 虚存,128MB RAM。

80860 是 Intel 推出的第一个真正的单片 RISC 处理器。它的集成度是 1×10^6 个晶体管,是突破百万大关的首批处理器。它含有一个 RISC 内核、一个具有向量能力的 64 位 FPU、4KB 和 8KB 数据高速缓冲器,并有硬件支持三维图形原语。所有这些聚集在一块芯片上的功能是通过芯片内快速、宽阔的总线(高达 128 位)紧耦合在一起,提供比其他多芯片 RISC 结构更快的浮点性能。

i860 在巅峰状态时可达每秒 66M 浮点运算,在运行优化的 LINPACK 时为 17MFLOPS,在大多数非优化应用程序时为 10MFLOPS。直至如今,80860 依然是太新了,它还缺乏软硬件的支持。

Intel 80486 和 Motorola 68040 的推出,使得复杂指令集计算机 CISC (Complex Instruction Set Computer)可以和目前的 RISC 相抗衡,CISC 和 RISC 的抗争烽火再起。

Intel 80486 有一个高度优化的和 80386 兼容的整数部件、一个

和80387兼容的 FPU、完整的虚拟存储器管理和保护系统、8KB 程序和数 据高速缓存、总线窥探(bus snooping)和其他多道处理机构, 还有少量系统结构和指令集的扩充。它的片芯尺寸 1.57×1.05 cm, 采用 1 μ m 高速 CMOS 技术, 内含 1.18 M 晶体管, 打破了 80860 1M 的短暂记录。

Intel 公司声称80486在25MHz 时性能为:Dhrystones 37 000、 Whetstones(双精度)6.1M, 或以 VAX11/780标准为15~20VAX MIPS。当然33MHz、40MHz 的80486的性能更强。

CISC 处理器取得了以往只有 RISC 才能获得的性能水平, CISC 和 RISC 的竞争在继续。可以说, 不管怎样, 超级计算机的能力, 小型机的价格正在变成现实, 计算机图形能力日益雄厚。

6.3.2 DECstation

以小型机著名的 DEC 公司, 在工作站上也不遗余力, 推出 DECstation。这是一种低价高性能的 Unix 工作站, 也是 DEC 公司设计成只在 Unix 下运行的第一种机型、同时采用其他公司 CPU 的第一种工作站。因为 DEC 公司通常使用 VMS 操作系统和自己的 CPU。

DECstation 2100和3100除了 CPU 和存储器速度不同外, 其他是一样的。DECstation 的中心是 MIPS 公司的 R2000 RISC CPU、R 2010数学协处理器、R2020写缓冲器。DECstation 2100上的 CPU 为 12.5MHz, 而3100上的为16.67MHz, 它们运行速度分别为10MIPS 和11MIPS。FPU 在双精度 LINPACK 时, 2100为1.2MFLOPS, 3100 为1.6MFLOPS。它们和采用 Motorola 公司88000 RISC CPU 的 DG (Data General)、Sun 公司的 SPARCstation 1是同一档次的系统。

和 Sun SPARCstation 1不同的是, DECstation 的32位存储器总线不支持直接存储器存取, 也没有扩展插槽。

系统存储器由120ns DRAM 模块组成, 以4M 增量, 最多为 24MB。还有64KB 数据高速缓存、64KB 指令高速缓存。2100采用 35ns 静态 RAM(SRAM)芯片, 而3100采用23ns 的 SRAM。

系统内可装两个104MB的SCSI硬盘,平均存取时间为33ms。DEC公司推荐把它们用于网络文件系统(NFS)环境中的页面交换,当然也可存储数据。系统有SCSI接口用于扩充设备,比如外接硬盘、数据流磁带机、CD-ROM光盘机等。

系统使用39cm和48cm监视器,8位彩色系统使用1MB帧缓冲器,在16.7M种色彩中选择,可同时显示256种色彩。在单色系统中帧缓冲器为256KB。

系统除SCSI插口外,还有标准的以太网接口,15针D型视频口,鼠标和键盘口,两个串行口(用于打印机、调制解调器)。遗憾的是,系统没有并行口(因此不能使用众多的打印机),串行口用专门的六线插口(有点象RS-442),这样就需要转换插口和电缆来连接非DEC串行设备。

系统的键盘是非标准的105键键盘,它源于VAXstation,有些键是针对DEC的VMS操作系统的,在Ultrix(DEC的Unix版本)下毫无作用。

DECstation的Ultrix工作站软件包括Ultrix-32、Ultrix/DEC Window、C编译器、X窗口系统11、X用户界面(XUI)和TCP/IP、NFS。

DECstation的Ultrix-32操作系统把BSD Unix和System V.3结合在一起,它有自己的/bin/sh和/bin/csh版本。DEC提供标准的Unix外壳,不过必须注意是哪种外壳。具有两种版本的好处是功能强,缺点是不同机器间移植难。

DEC的RISC机所用的Ultrix包括用于优化程序执行的专用命令。比如pixie是分析程序块使用的程序,cond2获取pixie产生的信息并重新安排基本块以完成更好的缓存映射。其他还有许多实用程序用于电子邮件、网络邮件和报文处理。

XUI的组成是:窗口管理程序;会话特性管理程序(session property manager),这是一种提供文件和任务管理的用户工作环境;开发指南;图形用户接口编译器及资源管理程序,它们可以简

化应用程序用户界面的开发。

DECstation 在存储器映射字节次序中采用与 VAX 机相同的低字节在先,而大多数工作站,象 Sun 的 SPARCStation 1 则采用相反次序,高字节在先。

DECstation 2100 3100 不适宜于作为计算服务器或文件服务器使用,原因是它作大量磁盘输入输出时性能有限。它主要用于办公室和工程,特别是在工程上具有很强的竞争力。

DECstation 5000 是新型的工作站,功能比 2100 和 3100 要强得多。它采用 MIPS R3000 RISC 微处理器,可以配置 120MB RAM、21GB 磁盘。它的 SCSI 插口可连结七台设备。增强的图形功能有二维向量加速复杂的三维流水线。它采用新的 Turbo channel 32 位总线结构。这种内部 I/O 通道通过三个 44 脚插槽连结扩展电路板的数据。总线可在 12.5~25MHz 间工作,DEC 宣称它的峰值 DMA 性能为每秒 93MB。

DEC 为程序员提供了几乎一切所需的 X 软件。除了建 DEC 窗口应用软件所需的库外,DEC 还提供 GKS、OSF/Motif 和 PEX (PHIGS 的 X 扩展)。PEX 支持三维物体的创建和绘制。此外还支持应用程序的 PostScript 能力。

6.3.3 SPARCstation

Sun 公司的 SPARC 芯片组是目前用的最多的 RISC 处理器,它首先用在 Sun-4 工作站和 SPARCstation 1 中。而 Sun-3 系列用的是 Motorola 68020、68030。

SPARCstation 1 在 12MIPS,而 SPARCstation 300 用的是 25MHz SRARC 处理器,功能更强。

从外表来看,SPARCstation 1 和 Sun-3/80 (CPU 为 68030,速度 3MIPS)是一样的。系统主机尺寸很小,长和宽为 41cm (16 英寸),而高仅 6.35cm (2.5 英寸)。系统有 85W 电源,装置一个 1.44MB、8.89cm (3.5 英寸)软盘机、两个 100MB、8.89cm 硬盘机。硬盘机平均存取时间为 23ms,吞吐率约为每秒 1.2MB。系统的扩展箱可以装

327MB、13.34cm(5.25英寸)硬盘及150MB 磁带备份装置。

SPARCstation 带一个光鼠标和 IBMPC AT 兼容键盘(称为4型键盘),它已成为 Sun 产品系列的标准。

工作站用的监视器有多种。有44cm、48cm 单色灰度监视器,也有41cm、48cm8位彩色监视器。所有监视器的分辨率为1152×900象素。

SPARCstation 1 的 SPARC CPU 钟频为20MHz,它的 FPU 可以实现1.5MFLOPS,几乎是 Motorola 68882 FPU (Mac II、NeXT 和 Sun-3/80都采用68882)的10倍。

SPARCstation 从4MB RAM 开始,可一直加到16MB。主机板含两个串行口,一个以太网口、一个 SCSI 口和三个扩展插槽,其新的专用总线称为 S-BUS。

1. S 总线

S 总线是 SPARCStation 的一大特色,它是32位同步总线,以主 CPU 的钟频(即20MHz)工作。它具有 DMA 能力,允许 CPU 摆脱许多 I/O 处理任务,把 S 总线上的 DMA 通道分隔开来。

S 总线不光是一种扩展插槽总线,它包括各独立的通道用于 SCSI、以太网和串行口,它们都可以通过 S 总线实现 DMA 和直接虚拟存储器存取 DVMA(Direct Virtual Memory Access)。CPU 可以对64KB 高速缓存进行访问,因此在大多数 I/O 操作时可以全速工作。由六片 VLSI 组成的芯片群控制高速缓存、存储器管理、DMA 和 DVMA、时钟、键盘鼠标和外设口(SCSI、以太网和串行口)。

S 总线扩展插槽主要用于视频卡、帧缓冲、图形加速器或是第二块以太网卡。需要注意的是,S 总线是一种多主总线,就是说除了 CPU 外,S 总线的扩展卡(比如以太网控制器)可以暂时控制总线进行更快的处理。SPARCstation 上的三插槽 S 总线之中两个为主插槽,一个为从插槽。

S 总线以每秒60MB 的突发数据传输率工作,这比其他 Sun 工作站和许多其他型号的 Unix 工作站所用的 VME 总线快几倍。

SPARCstation 还有单通道音频接口芯片和内置扬声器及外接插座。系统软件包括每秒能处理8000个8位采样的 Sound Tool(声音处理工具),不过比起苹果公司的声音处理芯片和 NeXT 机的声音处理能力来算是很基本的了。SPARCstation 的实际强项是图形和数字处理。

2. 图形

Sun 公司为其工作站配备各种档次的图形功能,从基本的单色帧缓冲器到复杂昂贵的32位图形加速器都有。后者带 Z 缓冲,用于三维图形应用。较低档的工作站通常装备单色或彩色帧缓冲器(8位或24位彩色)或 GX 8位彩色或灰度图形加速板。

GX 板支持二维和三维坐标变换,并含有帧缓冲控制器。坐标变换机构可以高速执行矩阵数学运算,处理坐标平移和旋转、定标、全景和缩放等。它还控制 32×32 象素的硬件光标,去除了常规帧缓冲存储器中光标重迭面。坐标变换机构的结果直接送到帧缓冲控制器来处理锯齿显示效果平滑、光栅操作、剪取和位块传送(bitblt)等。板上有1MB 视频 RAM,通过 D/A 转换和1M 象素显示器连通。

GX 板的性能和主 CPU 及 FPU 的处理功能直接相关。GX 执行坐标变换的理论限度为51MFLOPS。它大大提高了速度,最高可为常规帧缓冲性能的50倍。GX 板可以用于 Sun-3/80或 SPARCstation(P4或 S 总线结构)。在 SPARCstation 中 GX 占两个扩展插槽。GX 板对于二维和三维制图及线框图是很理想的,对台式印刷系统也很有用。

3. 软件

Sun SPARCstation 采用的操作系统是 SunOS 4.0版。这种 Sun 版 Unix 的特点是共享库,它允许多个应用程序共享存储器中的库程序。系统软件包括 Sun 的网络文件系统 NFS、系统网络管理程序 SNAP(System Network Administration Program)。后者可以处理网上文件自动备份和维护。此外还有方便的信息帮助提示和文书管理。

在 SunOS 的顶层系统软件包括 X11/News 窗口系统、X/View 开发包(用于开发窗口应用软件)及 Open Look 性能指标和设计指导(用于设计 Open Look 用户界面)。这个软件包称作 Open Windows(开放式窗口),Sun 还提供基于 Open Look 的三种应用程序: SunWrite、SunPaint 和 SunDraw。

总的来说,这一层次的工作站缺乏强有力而容易使用的图形界面,因此它适合专业人员使用,而一般用户使用起来会对 Unix 命令等感到困难。

SPARCstation 除了往高档延伸之外,它也有廉价但性能依然很好的低档系统。比如 SPARCstation SLC 是一种无盘工作站,它的 SPARC 处理器速度为12MIPS,并含8MB 内存、以太网插口、高分辨率44cm 单色监视器等。和 SPARCstation 1不同的是,它没有薄薄的主机箱,整个系统装在监视器壳内。SLC 系统板上的 SPARC 处理器速度为 20MHz, FPU 为 1.0MFLOPS (SPARCstation 为 1.5 MFLOPS)。SPARCstation 1 后来用 25MHz 处理器,速度提高到 15MIPS 而价格不变。SLC 的单色监视器分辨率为1152×900像素,每厘米39.4个像素(每英寸100个像素)。SLC 没有风扇,使用起来很安静。

不过 SLC 也有局限性,它是按网络环境中的一个节点设计的;尽管它能单独使用,但在网络环境中更有意义。其原因是:第一,SLC 是单色系统,即使想外接彩色监视器,由于没有彩色图形加速板所需的 S 总线插槽也不行;第二,SLC 的磁盘存储传输率有限,虽然可通过 SCSI 口外接磁盘、磁带或光盘,数据传输率只有每秒1.5MB (SPARCstation 1 用 S 总线可达80MBps),第三,SLC 无软盘驱动器。而在网络环境下,SLC 比同样配置的 SPARCstation 1 便宜得多。

SLC 比 X 窗口系统终端有明显的优越性。后者没有独立的处理能力,当有很多用户同时运行应用程序时,基于“分时”的系统性能会迅速下降。而 SLC 有独立的处理能力,比 X 窗口系统终端也

贵不了多少。在构成网络系统时,采用一些无盘工作站是值得认真考虑的。

Sun 公司还推出了更小型化的工作站,SPARCstation IPC。主机箱尺寸为 $267 \times 240 \times 121\text{mm}$,但却有24MB 内存、1.44MB 软盘驱动器、207MB 内装硬盘、两个串行口、粗缆以太网口、外接 SCSI 插口、音频 I/O 插口、彩色帧缓冲器板、S 总线插槽。

IPC 有 SunOS 4.1 和 Open Windows。Open Windows 是一种应用程序环境,允许 X 窗口系统、Open Look、SunView 和 News(也可选择 DOS)程序共享同一个显示器。News 把 PostScript 的灵活性和 X 窗口 11.4 的袖珍性揉合在一起提供开发需要的尺寸可变字体和图形性能。IPC 具有很高的性能价格比,从软件来说微型计算机更是不可比拟。这种类型的机器将成为日用的 Unix 工作站。

6.3.4 RISC 工作站之战

HP/APOLLO 最近推出 9000 系列 700 的两个新机型:710 和 705。它们采用 PA-RISC 处理器,这是目前最快的 RISC 处理器。710 运行在 50MHz,综合性能为 SPARCstation IPC 的四倍多。705 运行在 35MHz,为 IPC 的三倍多。

机器被称为低价个人工作站,主机箱尺寸为 $425 \times 375 \times 76\text{mm}$,其中有粗缆及细缆以太网连接器、并行口、两个串行口和 SCSI-2 插口。710 有 8 位帧缓冲器,支持 48cm (1280×1024 像素、72Hz)或 41cm (1024×768 像素、75Hz)256 色(同时)显示。公司也提供 48cm 灰度监视器。

最小系统是无盘的,有 16MB 80ns RAM、32KB 指令高速缓存,64KB 数据高速缓存。RAM 可装至 64MB。最大的内置硬盘驱动器容量为 840MB,最大的外接盘容量为 9.45GB。

RISC 工作站之战正在不断升级。Sun 的 SPARC 设计主宰了工作站市场数年,IBM 的 RISC System/6000 一改它的 IBM RT 的可笑地位而登上工作站性能最佳的宝座。不过 IBM 并没能支配工作站市场,而 1993 年随着 HP/Apollo 推出 720,IBM 的性能最佳记录即

将打破。

其他著名的 RISC 设计有 Motorola 的 88000、MIPS 处理器和最新的 Sony 产品。最新的 MIPS R4000 是全 64 位处理器。

HP/Apollo 710 和 705 工作站的目标是技术及商用市场。它们是很有用的计算机辅助绘图和设计工作站,对于需要高分辨率和高性能的电子印刷、信息管理和针对成组工作和项目的连网场合尤其适用。

Lotus 1-2-3、WordPerfect、Wingz、Oracle、Informix、Ingres、Sybase 及普遍的电子印刷和多媒体程序包都能在 710、705 上运行。

HP/Apollo 称 710 可以每秒产生 950 000 三维向量,相当于同一价格其他工作站相应性能的二倍。在三维曲面显示时也没有问题。在用 Iris Indigo 时,完成三维图形操作不用图形流水线协处理器。

750 有更大的高速缓存, RAM 可达 384MB,内置硬盘可达 2.6 GB,处理图形的能力则更强。如果只需要更大的盘空间和 PA-RISC 处理器,则可选用廉价的 9000 系列 700 服务器。

第七章 图形软件接口标准

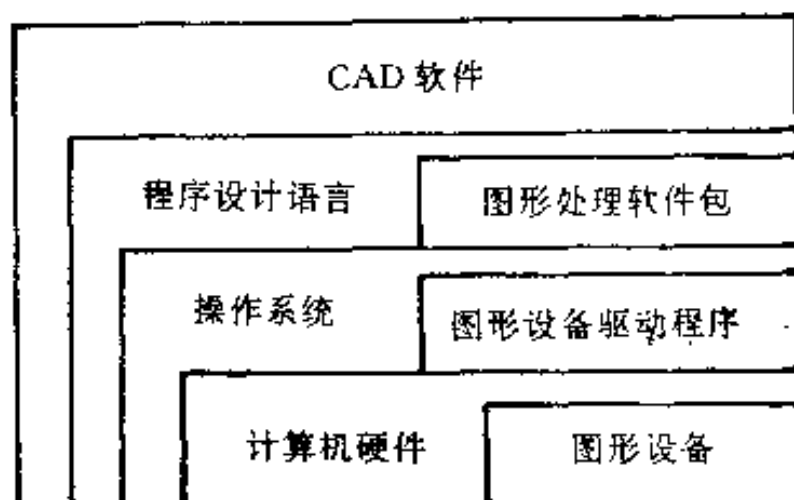
工作站和微型计算机的另一个明显区别是,工作站软件系统具有一系列接口标准使图形应用软件摆脱对硬件设备的依赖,允许很方便地在不同系统间进行移植。

以 CAD 系统为例,目前有许多大型的 CAD 软件可以在不同型号的工作站上运行,这也是微型计算机系统不能比拟的。CAD 系统软件的层次结构如图7-1所示。

目前,计算机图形是非常活跃的领域,在诸多方面都有长足发展:

- 图形支撑软件的标准化,采用国际标准 GKS、PHIGS、GKS-3D 等。

- 应用软件开发环境通用化和标准化,采用用户界面管理系统 UIMS,窗口管理系统如 Microsoft 公司的 Window、Digital Research 公司的



GEM、MIT 的 X Window 等。 图7-1 CAD 系统软件层次结构

- 算法的可靠性和高效率,快速生成高度真实感彩色图象,对体素拼合中曲面求交和拓扑处理等。

- 并行处理提高真实感图象的生成速度,采用多处理器并行计算的方法来解决庞大的计算量。

- 模拟自然景象,采用纹理映射、分形(fractal)技术、圆球造型

等方法获得逼真的图形。

工作越是深入,图形软件标准就越重要。各种接口标准相互关系参见图7-2

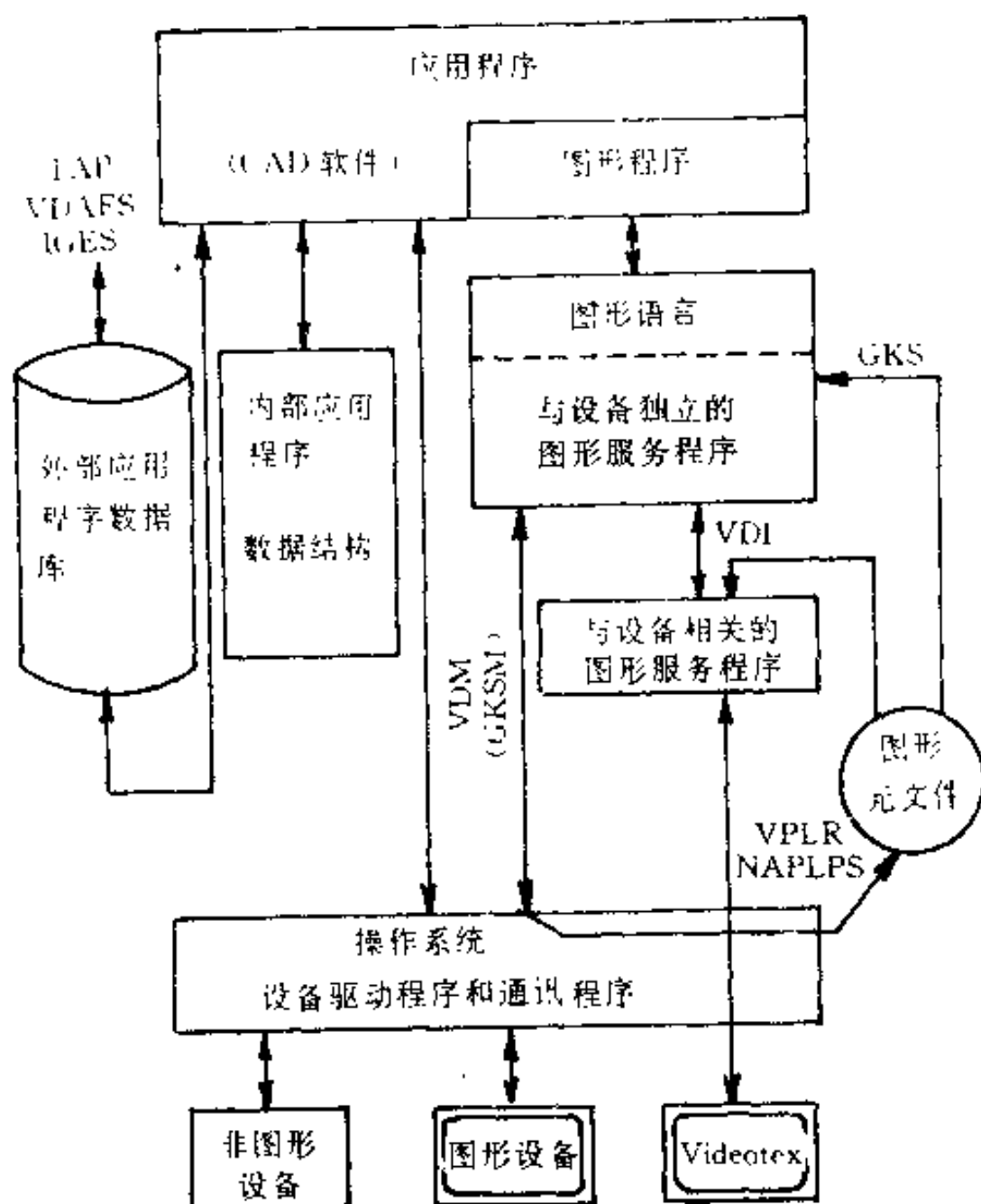


图7-2 图形系统中各种标准和接口

7.1 图形程序设计标准

7.1.1 GKS

GKS 是国际标准化组织批准的在应用程序和图形输入输出设备之间的功能接口。它只定义了一个独立于语言的图形系统的核心。在具体应用中,必须以符合所使用语言的约定方式把 GKS 嵌入到相应的语言中。GKS 在图形应用中的地位如图7-3所示。

图中每个层次可以调用下一层次的功能。对于应用程序员来说,通常使用面向应用层、语言联编层以及操作系统等资源。

1. 图形工作站

GKS 的基础是图形工作站,这里的工作站是指一种抽象的物理设备。一个抽象的工作站可以具有如下性能:

- 有一个可按地址访问的具有一定分辨率的显示面。
- 允许使用一个规则的显示空间,该显示空间不能由几个分离部分组成。
- 允许在显示空间中指定和使用一个较小的显示空间,工作站有裁剪功能,可使指定显示空间外的图形不予显示。
- 提供线型、字符字体和字体大小,使输出的图形元素可以有不同的属性。
- 允许有一个或多个逻辑输入设备,用于实现不同种类的输入功能和输入模式。
- 工作站可用于存储图形段,为修改和管理图形段提供方便。

对于一个具体的工作站来说,往往并不具备所有如上六方面的全部功能。例如绘图机是一个只具备有限输出功能的工作站。GKS 工作站有以下六类:

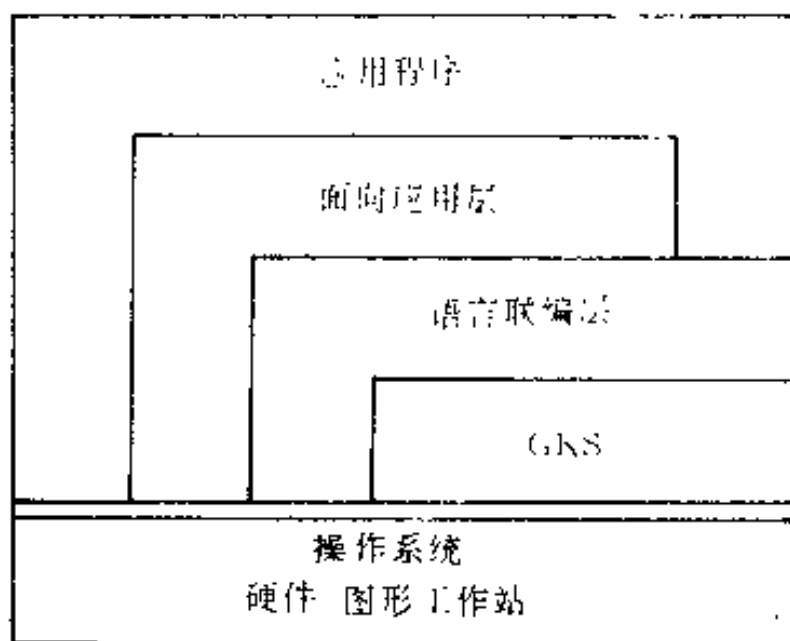


图7-3 GKS 在图形系统中的位置

- 输入工作站,它至少有一个输入设备,比如数字化仪。
- 输出工作站,它有一个显示面,用于显示输出图形元素,例如绘图机。
- 输入输出工作站,它有一个显示面和至少有一个输入设备。
- 与工作站独立的图段存储 WISS (Workstation Independent Segment Storage)。
- GKS 元文件 GKSM (GKS Metafile) 输出工作站。
- GKSM 输入工作站。

2. 输出功能

GKS 的核心是输出功能。应用程序的要生成的各种图形几乎都可以由一些基本的几何元素来构成。这种基本元素在 GKS 中称为图形元或图元 (graphical primitive)。应用程序利用图元进行输出时,不仅指出每个图元的位置、形状和大小,还要通过其属性指出外貌特征,从而构造出各种多姿多彩的图画来。表7-1是 GKS 六种输出图元及其属性的简要概括。

为了便于对被显示的图形的某一部分实现位置、形状或大小等各种变化,也为了使图形中的某些部分易于多次重复生成,GKS 中引入了图段 (segment) 的概念。所谓图段,指的是能够作为一个整体进行处理的一组图元。通常一幅图形总是若干图段组成的,每个图段具有各不相同的图段标识。

图段有五种属性:

- 可见性 (visibility),表示图段在输出设备上是否生成可见图形。
- 醒目性 (highlighting),表示图段是否以比较醒目的方式(如加亮、闪烁、加框等)显示。
- 次序性 (priority),表示图段与观察者距离的远近。
- 可检取性 (detectability),表示该图段能否用检取设备 (pick device) 进行选择。
- 图段变换 (segment transformation),表示图段在送往物理设

备之前,它所包含的全部图元必须进行一次二维几何变换,借此达到动态改变图段位置、形状或大小的目的。

表7-1 GKS 的输出图元及其属性

名 称	含 义	属 性
折线 polyline	一组相互连接的线段	线型、线宽因子、色彩号
多点标记 poly marker	组相互离散的点标记	标记类型,标记尺寸因子,色彩号
填充区 fill area	内部涂以指定颜色,图案或阴影线的一个多边形围成的闭合区域	内部装饰类型,图案或阴影的类型号 图案尺寸,图案参考点,色彩号
正文 text	在指定位置处输出的一串字符	字符高度,垂直向量,路径,对准,字符展宽因子、间距,字体及精度,色彩号
单元阵 cell array	由若干行和列构成的矩形区域,其中每个单元可以具有各自的颜色或灰度	除检取标识符外无其他属性
广义图元 GDP	其它图元,如圆、椭圆、样条曲线,它们由具体的工作站决定。	借用性质类似的图元属性控制其外貌

此外,GKS 把不同类型的输入设备分成六种不同的逻辑设备,即:定位设备、笔画设备、检取设备、选择设备、数值输入和字符串输入设备。

3. GKS 语言联编

GKS 作为一种功能描述,它既和设备独立又和各种高级语言独立。这样,程序语言可灵活选取,而 GKS 功能变成某种高级语言的子程序是通过语言联编层来解决的。现以 GKS 的功能 SET WORKSTATION VIEWPORT 为例来说明。

SET WORKSTATION VIEWPORT 的参数为:

工作站标识符(输入) N

工作站视口限界(输入) XMIN XMAX, YMIN, YMAX DC 4XR

其中“DC”表示设备坐标,“N”表示 GKS 数据类型“名称”,“R”为数据类型“实数”。

在 FORTRAN 语言联编中,该功能取如下形式:

SUBROUTINE GSWKVP(WKID,XMIN,XMAX,YMIN,YMAX)

输入参数:

INTEGER WKID 工作站标识符

REAL XMIN,XMAX,YMIN,YMAX 工作站设备坐标的视口限界

注意 GKS 数据类型“N”转换为“INTEGER”。

在和 Pascal 语言联编时为:

Procedure GSetWsViewport

(Wsid:GTWsid;bound:GRbound);

该程序采用的 Pascal GKS 数据结构为:

Const GTWsid=INTEGER;

GRbound = record

LeftBound,RightBound,

LowerBound,Upper Bound;REAL

end;

对于不同类型的应用程序,在 GKS 上要加上一些所需的特殊功能,如绘制工程图需要剖面线、尺寸线等功能,它们可以在面向应用层中完成。应用程序可以调用面向应用层、语言联编层及操作系统等。

4. GKS 元文件

为了保存和交换图形信息,GKS 提供一个顺序文件的接口,该文件称 GKS 的元文件 GKSM(GKS Metafile)。它的作用是:

- 统一的文件格式便于在不同图形输出设备上输出图形。
- 和设备无关的存储格式便于图形保存和日后调用。
- 图形信息可以通过通讯线路或介质(如磁带)传送或交换。
- 便于把各种软件包及其他信息源产生的图形信息统一成一种格式。

• 便于图形编辑。

7.1.2 GKS 的扩展

GKS 原本是一种二维图形系统,它可以满足很多应用要求,不过,对于 CAD 这样的应用领域,三维系统就是不可缺少的了。这样的一种三维图形系统称为 3D-GKS。

3D-GKS 是 2D-GKS 的扩展,所有的 2D 功能保持不变,所有的 2D GKS 程序可以在 3D-GKS 系统中运行并产生同样的结果。3D 功能是 2D 功能的普遍化,所增加的 3D 功能规定了 3D 输出原语、3D 变换、3D 到 2D 的视见变换以及 3D 坐标输入。还有隐藏线、隐藏面处理等。

此外,GKS Output level 3, GSPC Core 和 PHIGS 都是对 GKS 在维数和结构方面的扩展,它们的地位如图 7-4 所示。

由图可见,PHIGS 对 GKS 的扩展最为广泛。

PHIGS 和 3D-GKS 从应用角度来看,都采用与设备独立的原则,都是三维图形支撑系统。在两个系统中,很多概念具有同等含义。这是因为 PHIGS 设计的初衷就是尽量和 GKS 保持兼容。这样移植的困难减少,不

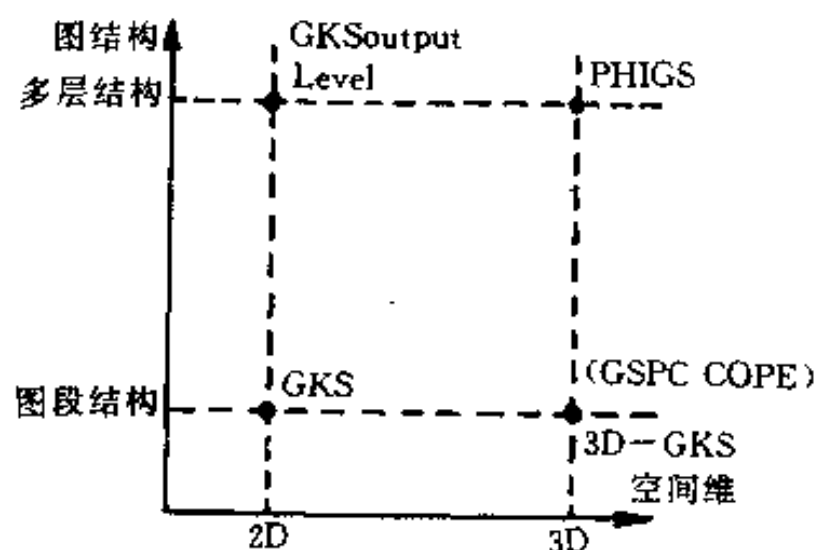


图 7-4 图形标准分类

过它们毕竟是两个不同的标准,PHIGS 比 GKS-3D 具有更多的优点和特色。

PHIGS 不但在细小处做了扩充(如增加图元类型),更重要的是采用了层次模型的思想。这是 PHIGS 和 GKS-3D 的最大区别。GKS 是通过图段来构造物体的,图段由若干图元组成,但图段不

能嵌套。为了表达复杂物体或构造一个复杂图段,或构造多个相对简单的图段再把它们分别显示出来,这些对应用开发者来说都是很不方便的。PHIGS 采用适合人们思维逻辑的层次模型,把复杂物体自顶向下逐级分解,再用结构功能自底向上逐级构造堆积,大大方便了应用程序的开发和维护。

PHIGS 比 3D-GKS 更接近工作站硬件,许多高性能工作站的功能得到了充分发挥。

7.1.3 PHIGS

PHIGS 即“程序员层次化交互式图形系统,”它提供基本的三维绘图元,并有一个层次化的数据结构。应用程序可以此为基础建立自己具有层次化逻辑结构的实体模型,并可以交互式地修改和操纵该模型(如视见变换等)。

PHIGS 的主要功能如下:

- 产生和编辑层次化应用模型
- 产生和设备独立的二维或三维计算机图形
- 强有力的三维视见变换功能
- 图形交互功能
- 实体模型和其图形的实时更新操作
- 图形和模型的脱机存储

从结构上看,PHIGS 可以分为四个部分:核心为控制中心,其外层为数据定义系统、数据显示系统和交互系统。

控制中心控制整个系统的运行状态,监视应用程序对其它各子系统的访问。数据定义系统控制“结构”的建立、编辑、修改和归档等,以便构造用户需要的物体模型。数据显示系统负责把图形数据送到输出设备(图形终端或绘图仪等)。图形交互系统是 PHIGS 中最复杂的一部分,它负责整个系统和外部的交互。

1. 控制中心

PHIGS 控制中心通过设置一些系统的状态参量来维护系统的运行状态。比如用户要启动 PHIGS 必须把它“打开”,某个子系统

要用某一输入或输出工作站也要做“打开”和“关闭”的操作。PHIGS 把系统的图形设备(如终端、绘图机等)甚至图形文件都称做工作站。

数据定义子系统中为了建立“结构”,也需要先打开一个“结构”,建成后再关闭。对于这些状态的控制,都由 PHIGS 的控制中心完成,它用 open 或 close 命令来实现打开或关闭。

2. 数据定义系统

数据定义系统是 PHIGS 的最重要的一部份,是构造三维实体模型的工具。应用程序利用它把图形数据交给 PHIGS,再由 PHIGS 完成视见变换和裁剪等,最后由数据显示系统送到输出工作站上。PHIGS 提供了多种构成三维实体的基本单元——图元(primitive),若干图元组合为一个结构(structure),若干简单结构组合成一个复杂结构。每个结构都可以动态编辑和修改。这些通过结构构造的模型数据存放在 PHIGS 图形数据库——中心结构存储器(CSS)中。

①基本几何图元

PHIGS 提供如下八种图元,每个图元定义在三维模型坐标系中,应用程序可以把 Z 坐标设为零来定义二维图形。

- 多点折线(polyline):通过一系列点相连的直线段。
- 多点标记(polymarker):选择所需点进行标记。
- 正文(text):在模型空间任选一平面输出一串字符。
- 注释正文(annotation text):在模型坐标系任一指定位置输出一串字符,在规范化空间中转换到 X-Y 平面上。
- 填充区(fill area):模型空间的一个多边形区域,可以填充,也可以中空。
- 填充区组(fill area set):一系列多边形区域,但看成是一个实体。
- 单元阵列(cell array):一系列指定色彩的相同图形单元的二维重复排列。
- 广义绘图元(generalized drawing primitive):由 PHIGS 开发工

具提供的在某工作站上实现的专用特殊图元。

这些图元和 GKS 基本类似,不过增加了注释正文,填充区组,此外所有图元都定义在三维模型坐标系中。

② 属性

图元一共有四种属性:几何属性、非几何属性、视见属性和识别属性。图元可以有零或多个属性。

几何属性决定图元大小和形状,如正文字符高度、线段的长度等。非几何属性影响图元外观形态,如颜色、线宽等。视见属性决定图元在三维空间中被观察的角度和方向。识别属性标识图元以便被输入设备检取。这些属性都在结构定义中给出,而在结构遍历(显示输出)中影响图形状态。

③ 物理模型的构造

PHIGS 一个最大的特色是支持分层的数据结构。一个复杂的物体可以逐级分解为一个简单的小单元。每个单元可定义成一个结构,类似于 GKS 中的图段。

一个结构可由若干个结构元素(structure element)组成,它们可以是基本图元或属性定义,也可以是其它结构。以若干简单结构生成一个复杂结构的方法,不断进行下去可以构造非常复杂的结构来表示复杂的物体模型。PHIGS 中的层次结构不允许递归引用。

每个结构有自己特定的名字,一旦结构被定义,应用程序可对它方便地进行动态修改和编辑。为此,先用 open-structure 命令打开该结构。这时有一个内部指针指向结构的最后一个元素;应用程序可以移动指针来修改指定的元素。为了便于指针定位,PHIGS 提供称为“标号”的特殊结构。它不影响输出,只是标识一下结构中元素的位置。

结构被编辑和修改后,将由显示系统自动把更新后的内容送到显示工作站上。结构中的数据放在 CSS 中,也可以送到文档文件中,供其他应用程序引用。文档文件可以看成是 PHIGS 的一个输出工作站。

3. 数据显示系统

PHIGS 把数据定义和显示完全分隔开来。数据定义系统把实体模型数据以结构的方式放在 CSS 中,显示系统则通过遍历 CSS 中某一结构中所有元素来实现数据的显示。一旦控制中心检测到数据定义系统中已经作过修改,就自动启动显示系统重新遍历结构的内容以保证 PHIGS 内部和输出工作站上的一致性。由于 PHIGS 中图元是被定义在许多不同的模型坐标系下的,它的视见变换流水线(viewing pipeline)和 GKS 有所不同。

1. 交互处理系统

PHIGS 交互处理和 GKS 类似,提供六类逻辑输入设备:

- 定位(locator):输入一个(X,Y)值
- 拾取(pick):标识一个显示目标
- 选择(choice):从一组可选项中选择一项
- 定值(valuator):返回一个值
- 字符串(string):输入一串字符
- 笔画(stroke):输入一系列点

它们很容易被映射到一些典型的物理设备上。

应用以上六种输入设备,应用程序可以控制以下几种交互输入方式:

- 请求(request)方式:要求操作者响应输入请求,否则程序处于等待状态。

- 采样(sample)方式:操作者和外设无交互动作,由应用程序在需要时在指定设备上获得当前值。

- 事件(event)方式:输入设备产生的输入值放在应用程序可读的 FIFO 队列中。输入动作不中断应用程序的进行。

在 CAD 实际应用中,由曲线或曲面组成的复杂实体是很常见的,此外人们希望在图形终端上显示真实感图形,而目前许多高性能图形工作站都提供了快速浓淡处理的功能。为了弥补不足,PHIGS-plus 对 PHIGS 做了以下扩充:

- 增加曲线和曲面图元,包括三角形网(triangle net)、四边形网格(quadrilateral mesh)以及非均匀有理 B-样条(NURB)曲线和曲面。

- 应用 Gouroud 和 Phong 的光照模型,产生高度真实感图形。

PHIGS-plus 的曲线和曲面功能提供了构造平滑物体的手段,如汽车和飞机外形等,大大加强了实体构造功能。浓淡图的处理大大减少了应用程序开发的负担。

由于 PHIGS 独特的层次结构模型,给应用程序构造复杂物体提供了方便手段,越来越多的应用把它作为开发平台。然而在许多实际应用中,如三维实体造型系统,应用程序为了从已知物体通过布尔运算构造新的物体,需要知道原物体的几何信息及拓扑信息。假设原物体已由 PHIGS 的数据定义功能存放在 CSS 中,则对该物体的检索不会返回任何对拼合运算有用的信息,只有一个结构内部的标识及结构元素个数。可见,PHIGS 的数据结构除了方便显示和视见变换之外,对应用开发并没有太大的帮助。对于上述实体造型系统,必须选用适当的方法(如 B-rep)构造由应用程序自己保持的数据结构,只是在新物体生成后,再把数据传给 CSS,从指定设备上显示出来。所以,许多高级应用场合不能完全依赖 PHIGS 提供的数据结构,而需要有自己的数据结构。

另外,在制定 PHIGS 的初期,工作站上基于窗口的系统没有象今天那样有影响,所以在标准中并没有说明它和窗口系统的关系,而现今大多数 PHIGS 支撑程序都是在这样的系统中运行的。还有,PHIGS 并没有对消除隐藏线和隐藏面功能提供支持,这些都是有待改进的地方。

当然,尽管 PHIGS 不能解决应用中所有的问题,但它确实对应用程序的开发做了迄今为止最强有力的支持。

7.2 图形元文件

7.2.1 GKSM

图形元文件(Graphics Metafile)是一种具有标准格式的文件,用于图形的存储与传送,它不仅和设备、也和应用程序独立。这样,它以标准方式把各种图形设备及图形系统连结起来。

GKSM 是 GKS 的元文件,它是一种顺序文件,可由 GKS 读或写,用于长期保存和交换。GKSM 的读写参见图7-5。

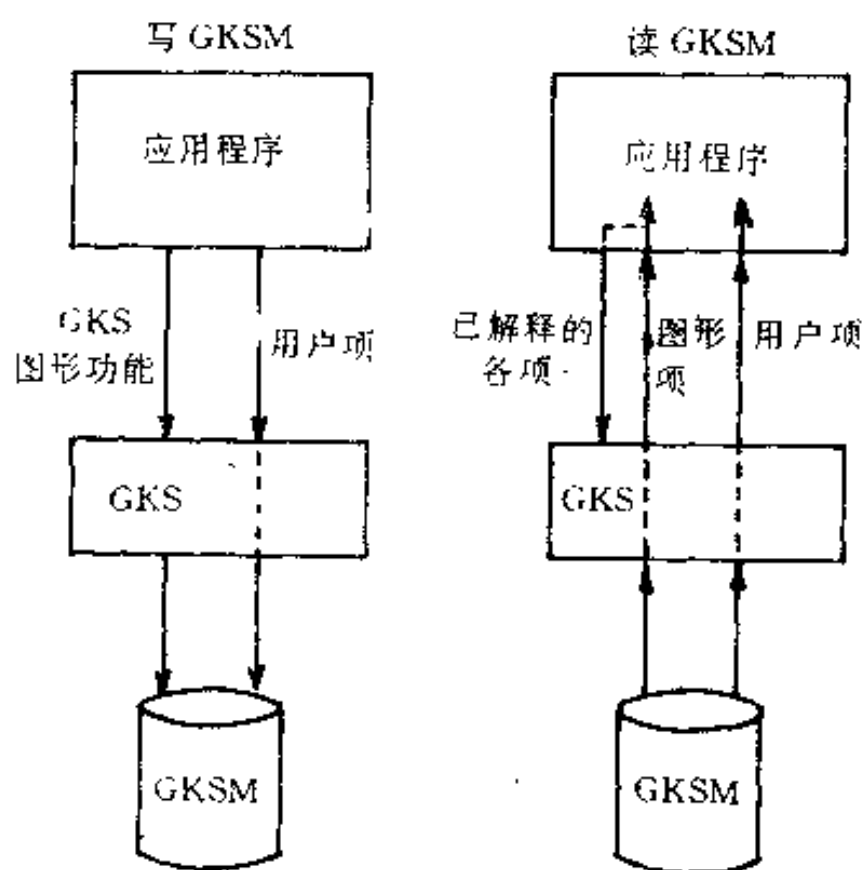


图7-5 GKSM 读写

GKSM 包含二维图,它是由调用的 GKS 功能产生的数据记录(“项”)表示的,内容为:

- 文件头
- 图头

- 记录尾
- 输出图形
- 属性
- 非图型、与应用有关的数据

GKSM 由逻辑上长度可变的数据记录串组成。GKS 把 GKSM 作为一个工作站来寻址,因此该元文件有两种工作站类型: GKSM 输出工作站和 GKSM 输入工作站。

7.2.2 VDM 和 CGM

虚拟设备元文件 VDM (Virtual Device Metafile) 是一种数据接口,用来规定在数据文件中记录图形信息的标准,这种标准允许程序之间、系统之间交换图形数据。它规定了如下一系列元素的格式(相当句法)和功能行

为(相当语义):

- 描述符元素
- 控制元素
- 图描述符元素
- 图形元素
- 属性元素
- 转移元素
- 外部元素

VDM 就是上述这些元素的一种集合。描述符元素提供给 VDM 解释程序足够的数据来解释元文件元素并决定显示所需资源。

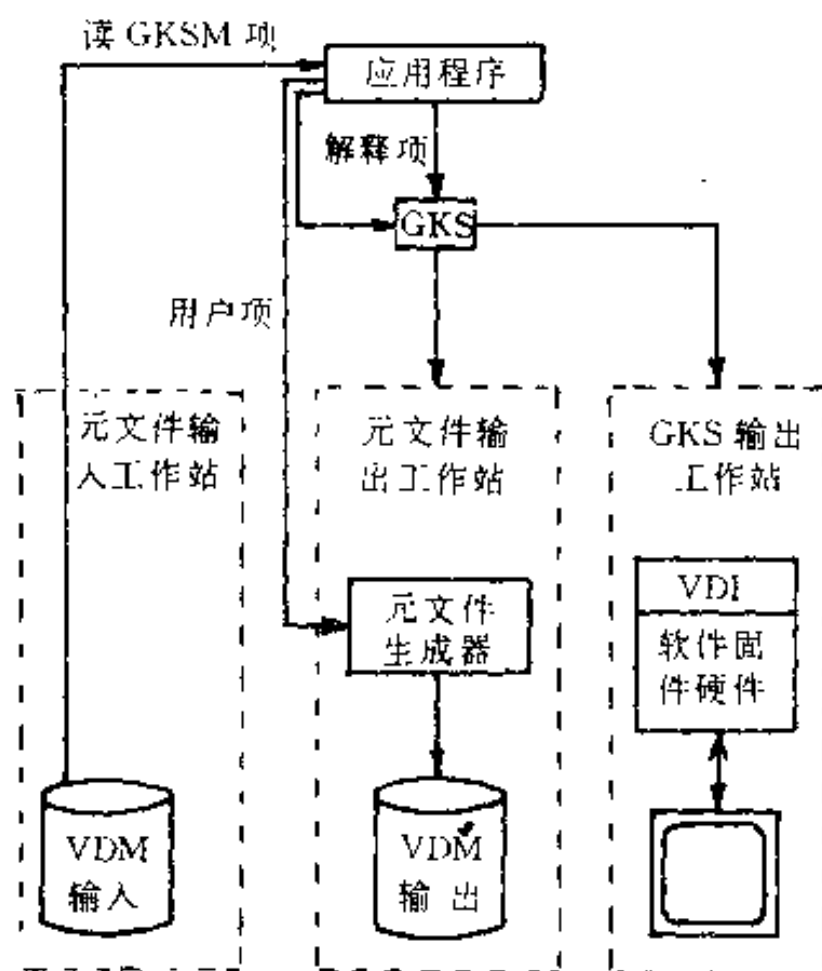


图7-6 VDM 和 GKS 的关系

VDM 和 GKS 的关系如图7-6所示。

因此可以把 GKSM 看成是一种 VDM,它实质上是一种低层数

据接口标准,现由 ISO 改称为计算机图形元文件 CGM(Computer Graphics Metafile)。

CGM 是为系统和系统开发而设计的,它能广泛地用于各种设备及应用程序系统。设计 CGM 的主要目的是:

- 提供图形存档的数据格式。
- 为图形设备接口标准化创造条件。
- 提供假脱机绘图的图形协议。
- 便于检查图形中的错误,保证图形质量。
- 提供了把不同图形系统所产生的图形集合到一起的手段。

7.3 工作站和设备的接口

7.3.1 VDI 和 WSI

在 GKS 和图形工作站之间存在一种界面,由它隔离图形系统的设备无关部分及设备有关部分。对于具有高度智能的设备来说,该界面将包括大部分 GKS 功能,因此是一种高层的界面。对于缺乏本机智能的设备来说,设备的功能就会很有限,那些没有的功能只能靠仿真来实现。可能的最高层界面称为工作站接口 WSI (Work Station Interface),而和虚拟图形设备 VGD(Virtual Graphic Device)衔接的是虚拟设备接口 VDI(Virtual Device Interface)。由于 WSI 有标准的固定的功能,因此 VDI 的功能就和设备有关,参见图7-7。

WSI 从功能上分隔了 GKS 核和 GKS 工作站。它允许 GKS 在分布式系统中的操作,并在核与工作站之间实现信息交换。这种不同站主的分布并不对 GKS 规定的功能有任何限制。界面的结构参见图7-8。

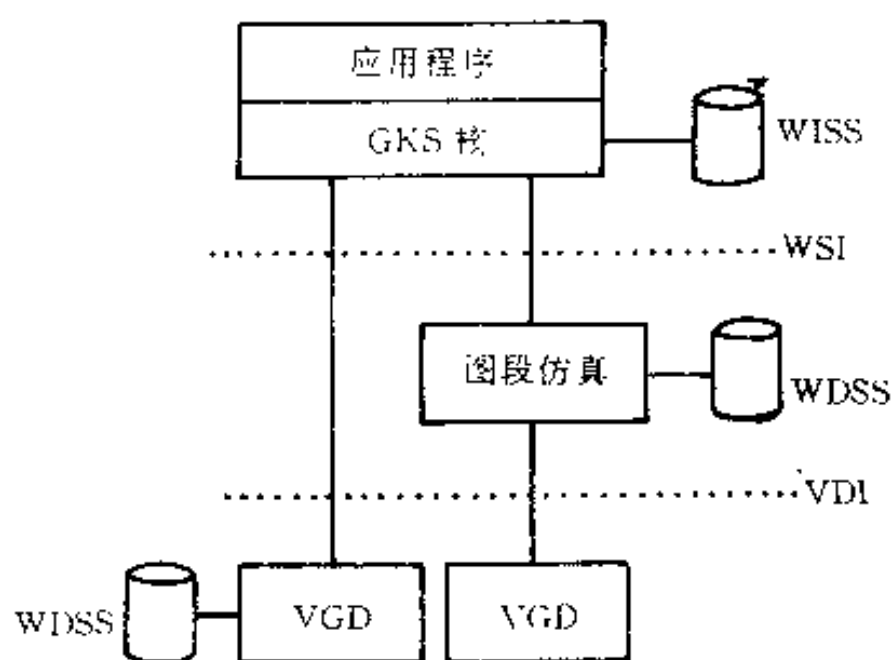


图7-7 GKS 工作站界面

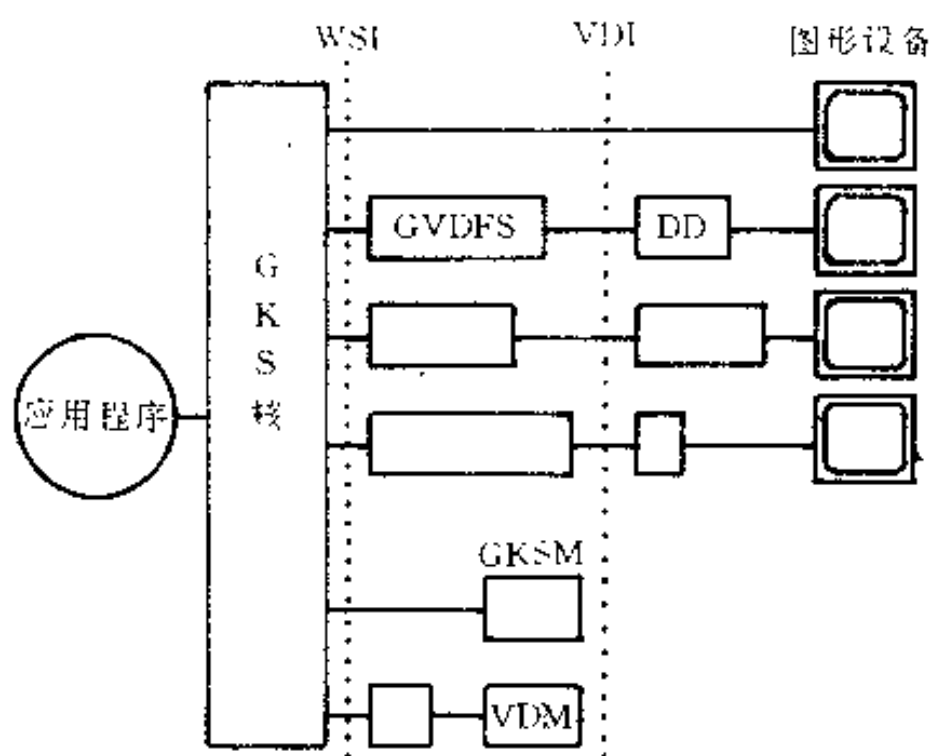


图7-8 界面结构

可见,有些设备,包括 GKSM 和0级 VDM 在内,VDI 的功能集和 WSI 的是一致的。对大多数设备来说,到 VDI 前尚未具备的功能必须要做仿真。这是由通用虚拟设备功能集 GVDFS (Generic Virtual Device Function Set) 来完成的。它包含从 WSI 处的功能转换到 VDI 处的功能所需的标准转换程序。

VDI 是一种标准化的功能及句法规范,用于与设备独立的图形软件及依赖设备的图形设备驱动程序之间的控制及数据交换。图形设备驱动程序把从 VDI 来的命令和数据转换为特定的输入输出设备所需要的形式。VDI 是最接近物理设备而依然与设备独立的最后一层界面。

7.3.2 CGI

计算机图形设备接口 CGI(Computer Graphics Interface)取代原先的老名称 VDI,是 ISO 提出的图形设备接口标准。

CGI 是与设备无关的计算机图形软件与一个或多个设备相关的图形设备驱动程序之间进行控制和数据交换的标准。它既可以以子程序的形式直接提供给用户使用,也可以作为隐含的标准支持软件实现 GKS、PHIGS 等高层图形标准。

CGI 具有初始化设备以及终止设备执行、重新设置设备属性和控制其约定状态的功能;CGI 也为用户提供了从图形缓存输出图形及不断修改图形的手段。CGI 还具有光栅图形处理功能。光栅操作的基本单元是像素和位图。位图操作可以把虚拟设备坐标空间 VDC 中指定区域的图象映射到当前设备坐标空间 DC 中。一旦产生了位图,它就可以作为图形输出的内容,并可以对图形中一部分重新定义和命名。

7.4 应用程序接口

7.4.1 概述

和硬设备这样低层次联系密切的接口,一般用户是不接触的。对于众多的应用程序用户来说,很关心不同系统之间信息共享的问题。以 CAD/CAM 系统来说,各种 CAD/CAM 软件设计者考虑到软件的应用目的,分别采用了不同的图形数据存储管理方式。为了节省存储空间和加快图形数据处理速度,都采用十分紧凑的格式

来存储数据。产品的设计和生产信息通常都以计算机可读的形式存储在 CAD/CAM 系统中。一般而言,一个 CAD/CAM 系统的内部数据表示和其它系统是不兼容的。这对于受不同硬件环境支持的 CAD/CAM 系统来说是常见的。

为了充分发挥现有 CAD/CAM 系统内在潜力,解决不同 CAD/CAM 系统间数据共享及通信问题,实现产品设计、分析与制造一体化,迫切需要解决 CAD/CAM 系统间数据传输问题。即一个 CAD/CAM 系统的图形数据信息,经过格式和代码的转换后,能被另一个 CAD/CAM 系统所接受,成为本系统图形数据库的内容。要实现这种转换,主要方法有两种:直接翻译和中性数据文件翻译。

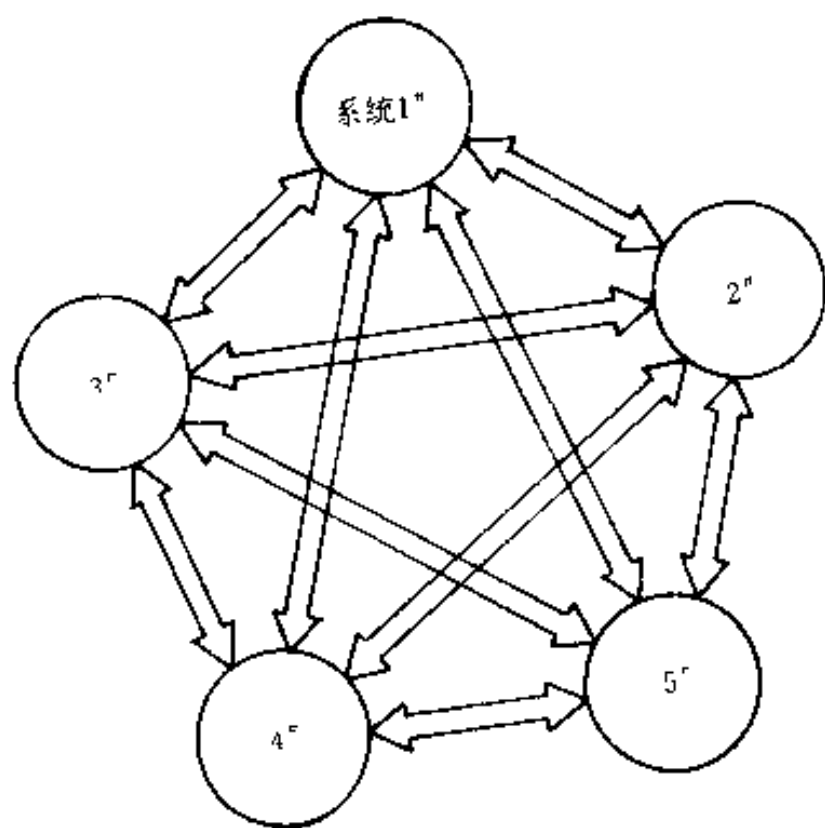


图 7-9 直接翻译示意

一些 CAD/CAM 用户曾试图编制专门的翻译程序将各个系统成对地连接起来,以克服各个系统单独操作的状况。这种情况只能

在局部条件下做一些有限的工作以解燃眉之急。在那些子系统很多的大系统中,所需翻译程序的数量增加很快。例如:在四个不同系统间使用直接翻译程序,需要 12 个;增加第五个系统时需要额外增加八个翻译程序,参见图 7-9。

以此类推:对 n 个系统来说,需 $n(n-1)$ 个直接翻译程序,每增加一个系统,需增添 $(2n+1)$ 个翻译程序。可见翻译程序的数量随系统数量的增加而剧增。如果考虑所编制的翻译程序仅在系统指定版本中有效,假设其中有一个系统发生变动,有关的直接翻译程序都得随之而变。因此直接翻译作为通用手段是行不通的。

和系统中其他模块化结构的思想一样,将系统和图形数据文件存储格式都转换成一种中性文件格式(不依赖系统),以它为共同枢纽来解决数据共享和通信是很实际的,这就是中性数据文件翻译程序的概念。对于四个系统而言,用中性数据文件转换,只需八个翻译程序;每增加一个系统,只需要增加二个翻译程序。参见图 7-10。

初始图形交换规范 IGES (Initial Graphical Exchange Specification) 就是在 CAD/CAM 中应用很广泛的一种中间数据文件格式。从理论上讲,任何 CAD/CAM 系统都可以把自身的图形实体转换为 IGES 格式,反之亦然。因此从理论上来说可以实现不同 CAD/CAM 系统间的数据交换。

7.4.2 IGES 的历史与未来

IGES 是 70 年代末在美国国家标准局 NBS 领导下研制成功的,其宗旨就是提供一种不同 CAD/CAM 系统间传送产品定义数据的中性文件格式。它的发展得到美国政府、空军 ICAM (Integrated Computer Aided Manufacturing) 工程的财政援助。NBS 下属的技术委员会与通用电气公司、波音公司及 CAD/CAM 制造厂家协助研制成功后,在 1980 年成为 NBS 标准。

1981 年,IGES1.0 版作为 ANSI 标准发表,它是产品数据交换领域中第一个国家标准。IGES1.0 版的功能包括:具有二维坐标,

或用平面、曲面三维线框表示的几何模型,这些都是 70 年代末 CAD 系统中的典型模型;注解,包括一系列标准尺寸标注格式;结构单元,提供实体和相应定义的属性联系。IGES 1.0 版没有为传送三维实体造型(solid modeling)完整信息提供手段。IGES 是一种 ASCII 码文件,其中 80 个字符为一记录,进行数据交换。

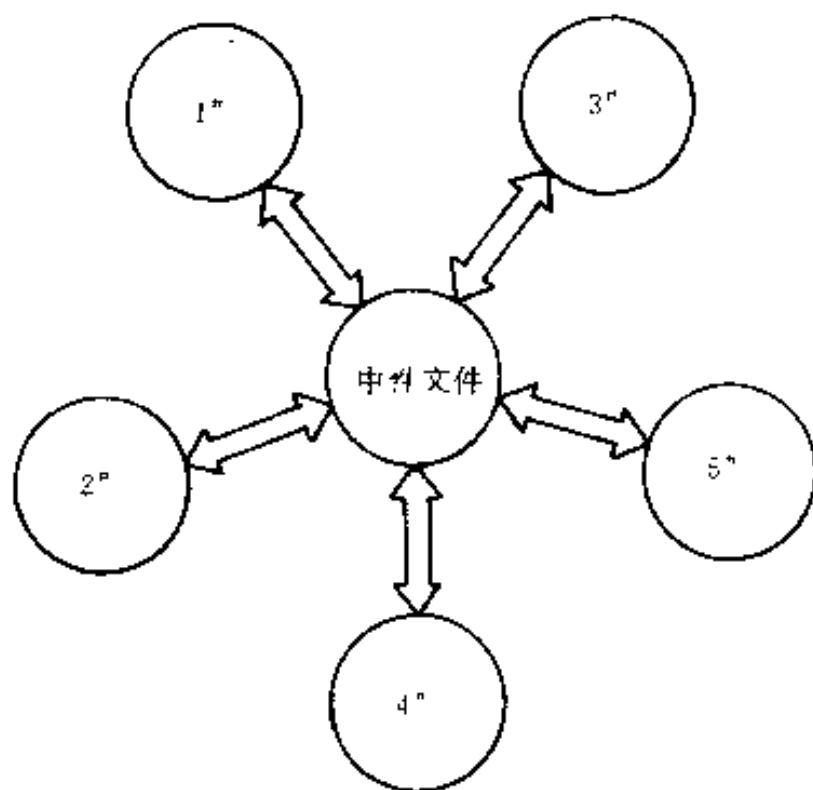


图 7-10 中性文件翻译示意

1982 年发表了 IGES 2.0 版,它是 1.0 版的扩充,新增加的内容是:

- 两个新的几何单元,有理 B 样条曲线和有理 B 样条曲面。
- 直纹(Ruled)、曲面、表列(Tabulated)圆柱和旋转体曲面。
- 更大的字符集。
- 有限元造型数据。

- 电子印制线路板产品数据。
- 二进制文件格式。

IGES 1.0 与 2.0 版除了在字符集单元的变化上不兼容外,其它方面是向上兼容的。由于 IGES 2.0 一直未提交给 ANSI 进行标准化检验,未被列为国家标准。

1987 年 8 月,IGES 3.0 版提交给 ANSI,成为美国国家标准。IGES 3.0 版定义的实体(entity)类型有 54 种,有些实体有多种形式,因此一共有 150 多种实体类型。用户还可以自己定义实体类型。

与此同时,又推出了 IGES 4.0 版。它除了对已有的 IGES 实体与格式做了一些修改和补充(主要是有限元部分)外,增加了对实体模型的建设性的实体几何表示的规定。

IGES 5.0 则主要吸收了实体造型的边界表示。

IGES 的继承者是 IGES 组织研制开发的产品定义交换规范 PDES(Product Definition Exchange Specification)。PDES 不再定义象 IGES 中的线、圆、尺寸标注等实体,而改用更概念化的模型。PDES 中零件以实体为基础,用孔、凸缘(flange)或凸条(rib)等特征来定义,并对要进行制造的零件定义一个公差包。PDES 中包括诸如使用材料、制造工艺、成本等非几何信息。这样它就成为零件的完整计算机模型。PDES 的应用领域主要在机械、电子及建筑工程制图及制造。在发展过程中采用信息建模(information-modeling)技术,找出为所有应用所需的概念数据的最小集合。存储信息的数据表达是一种三层结构:应用层、概念(或逻辑)层、物理层。这样 PDES 的数据交换格式要比 IGES 复杂得多。

PDES 是个雄心勃勃的计划,它从 1987 年开始勾画,很明显是面向未来的文件格式,它甚至可能和人工智能同伍。不管怎样,目前高质量的 IGES 翻译程序仍然是头等重要的。

7.4.3 IGES 的文件构成

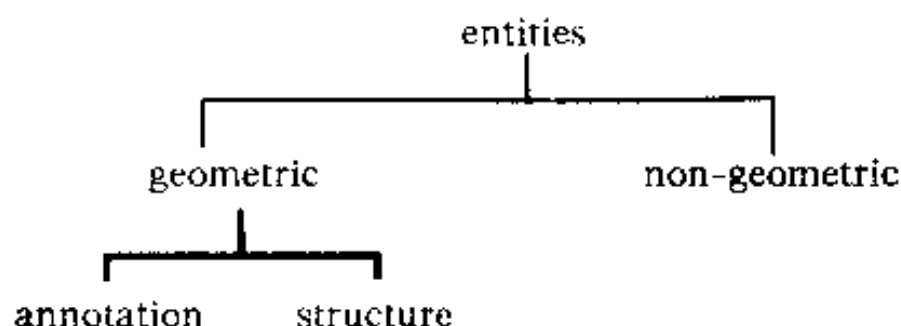
IGES 是用来描述产品设计和产品制造过程中有关信息的数

据格式的一种规范。它既规定了有关产品定义的各种几何数据和非几何数据的表示方法,也规定了使用的描述语言格式和文件结构格式。它既是系统中使用的几何模型的扩展,又独立于系统所使用的具体几何模型类型。

IGES 文件中的基本信息单位是实体(entity)。IGES 定义的实体有:

- 几何实体(geometric entities)
- 标注实体(annotation entities)
- 结构实体(structure entities)

它们之间的关系如下



几何实体定义了构体的基本组成,如点、线、圆弧、曲面等;标注实体给出物体的尺寸标注、文字注释;结构实体为系统组合复杂物体提供手段。以汽车设计为例,汽车轮子可以定义为一个结构实体。它由各种曲线组成,冠以名字后单独存储。在绘汽车图时,四个车轮不用重复画,只需调用结构实体在指定位置上画出就行。

IGES 文件有五个组成部分:起始部分(start section)、全局部分(global section)、目录索引部分(directory section)、参数数据部分(parameter data section)及结束部分(terminate entry section)。

IGES 文件的每部分由若干个记录组成,每个记录长度均为 80 个字符,使用的是 ASCII 字符集,最后八个字符为记录标识。IGES 起始部分是供人们阅读该文件的“序言”,说明文件的用途、性质和使用注意事项等。全局部分包含用来处理该文件所必须了解的有关信息,如分隔符和记录结束符的定义,文件名及文件生成日期等等,详细情况见表 7-2:

表 7-2 全局部分参数

参数	域类型	说 明
1	字符串	分隔符(缺省符为,)
2	字符串	记录结束符(缺省符为:)
3	字符串	发送系统的产品标志(图号)
4	字符串	文件名称
5	字符串	系统标记、卖主、软件版本号
6	字符串	ANSI 标准翻译版本号
7	整型	表示整型数的二进制位数
8	整型	单精度浮点数指数的二进制位数
9	整型	单精度浮点数尾数的二进制位数
10	整型	双精度浮点数指数的二进制位数
11	整型	双精度浮点数尾数的二进制位数
12	字符串	接受系统的产品标志(图号)
13	浮点	模型空间比例
14	整型	单位标志
15	字符串	单位
16	整型	线条深浅色调等级数(1~32768)
17	浮点	线段最大宽度
18	字符串	文件生成日期和时间 13HYMMDD.HHNNSS YY 年份(最后两位数) MM 月份(01~12) DD 日期(01~31) HH 小时(00~23) NN 分钟(00~59) SS 秒(00~59)
19	浮 点	模型的最小分辨率
20	浮 点	模型最大坐标值
21	字符串	作者姓名
22	字符串	组织

注:模型空间比例:模型空间对实际空间的比例

IGES 中无论是几何实体还是非几何实体都是用与应用程序无关的格式进行描述的。文件中可以包含任意个各类型的实体,每个实体有一个目录索引项和至少一个参数数据项。目录索引项集中存放在 IGES 文件的目录索引段,它给出实体的索引(指针)和实体的属性(或属性指针)。目录项的长度固定,共由 160 个字符组成,分成 20 个字段,每个字段八个字符。

目录索引段各个字段代表的意义如表 7-3 所示。

表 7-3 IGES 文件目录索引段说明

	1 8	9 16	17 24	25 32	33 40	41 48	49 56	57 64	65 72	73 80
第 1 行	实体 类型号 # 1	参数收 据指针 ▶ 2	结构 # ▶ 3	线型 # ▶ 4	图层 level # ▶ 5	视图 view 0 ▶ 6	转换 矩阵 0 ▶ 7	标号 ▶ 8	状态号 # 9	顺序号 0 # 10
第 2 行	实体 类型号 # 11	直线 数号 # 12	颜色 # ▶ 13	参数 行数量 # 14	格式号 # 15	未用 16	未用 17	实体 标号 18	实体 子标号 # 19	顺序号 0 4 4 1 20

注：#-数字，▶-指针，# ▶-数字或指针

实体的参数数据集中存放在 IGES 文件的参数数据段。参数以自由格式存放,第一个字段总是实体的类型号,在实体类型号与参数之间用逗号隔开。实体的类型号由 IGES 统一规定,例如圆弧类型号为 100,组合曲线类型号为 102,二次曲线类型号为 101 等等。实体参数的个数、数值、格式等由实体的类型、形状及它们与其他实体的关系决定。例如,一个圆弧实体可以有如表 7-4 所示参数。

IGES 文件的结束段只有一个记录,其中包含了文件中其余四个部分的最后一个记录的序号,它必须是整个文件的最后一个记录。

IGES 文件的目录索引(DE)段和参数数据(PD)段实体数据关系如表 7-5 所示。

表 7-4 圆弧实体参数

参数	值	格式	说明
1	ZT	浮点	圆弧所在平面与 Z 平面的位移量
2 3	X1 Y1	浮点 浮点	圆心坐标
4 5	X2 Y2	浮点 浮点	圆弧起点坐标
6 7	X3 Y3	浮点 浮点	圆弧终点坐标
8	N	整数	有关的其它实体(如标注实体、关系实体)个数
9 ⋮ 9+N-1	DE1 ⋮ DEN	指针 指针	指向相关实体的目录项的指针
9+N	M	整数	特性数
9+N+1 ⋮ 9+N+M	DEN+1 ⋮ DEN+M	指针 指针	指向各特性的指针

表 7-5 DE 和 PD 段实体数据关系

实体类 型号	参数 指针	PD 段记 录行数	矩阵 指针	顺序号	
124	1	1		00020000D	1 DE-Records Transl. Matrix
124		1	MAT	1D	2
100	2	1	1	00000000D	3 DE Records
100	1	2	1	ARC01	ID 4 circular arc
116	3	1	1	00000000D	5 DE Record
116	1	2	1	PT01	ID 6 Point

Rotation and Translation Matrix

124,	1., 0., 0., 0., 1., 0., 0., 0., 0., 1., 0.;	1P 1	PD-Records Transl. Matrix
	Z X Y X _a Y _a X _b Y _b		
100,	0., 25., 43.5, 35., 43.5, 15.72, 39.81;	3P 2	PD Records circular Arc
	X Y Z		
116,	25., 43.5, 0.;	5P 3	PD Records Point

* 注：上数据表达圆心在 PT01，起点在 a，终点 b 的一段弧 ARC01。

7.4.4 IGES 前后处理程序

IGES 只是一种描述产品定义数据的数据文件格式,要想利用它在不同 CAD/CAM 系统间传递产品定义数据,CAD/CAM 厂家及软件开发者还必须编制相应的软件来实现 IGES 的功能。参见图 7-11。

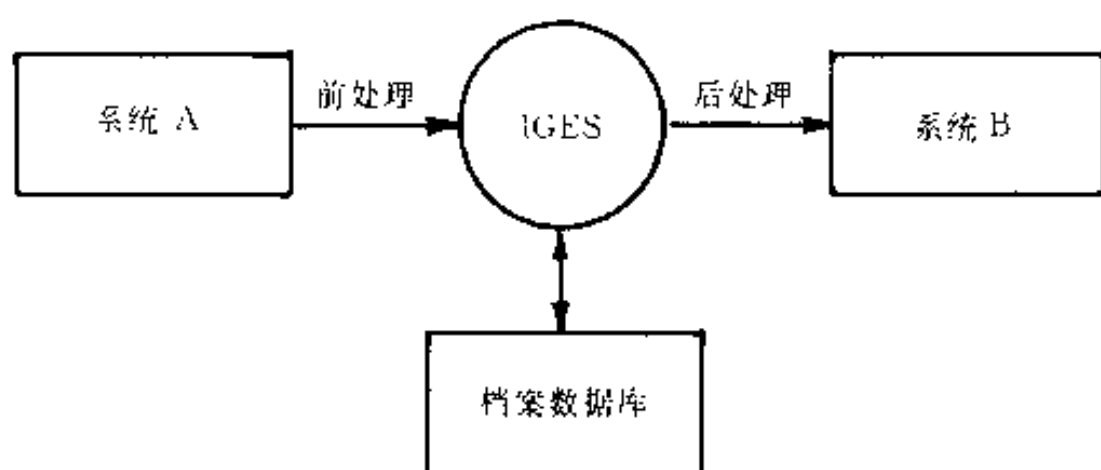


图 7-11 IGES 前后处理程序

对于系统 A 和系统 B,要把系统 A 的图形数据信息传送到系统 B 中,需要借助中性数据文件 IGES。而首先需要进行转换,实现从 CAD/CAM 系统到 IGES 转换的叫前处理程序(pre-processor);实现 IGES 到 CAD/CAM 系统的叫后处理程序(post-processor),前、后处理程序的质量决定了转换的成败。自 IGES 问世以来,有很多 CAD/CAM 软件厂家在 CAD/CAM 软件中加入了对 IGES 的支持。就是说,系统中已经含有前后处理程序。比如 Autodesk 公司的 AutoCAD 中有 IGESIN 和 IGESOUT 功能,更不消说象 SDRC 公司的 I-DEAS 和 Calma 公司的 DDM 这样的大型 CAD 软件。不过问题也并非如此简单。在实际应用中常常出现一些问题。例如两个机械系统间传输图形数据比一个机械系统与一个电子系统之间传输数据的准确性要高。这是由于机械系统和电子系统应用方向不同,各自定义的实体类型及数量不尽相同,传输以后有可能丢失一些

信息,这样产生的图形就会出现缺实体或实体位置、颜色不一致等错误。

第八章 微型计算机视频编程汇集

微型计算机的视频编程可以通过三种层次的方法来实现。一种是最上层的高级语言编程,通过高级语言或实用程序的图形库程序,这种方法实现起来比较方便,不需要对硬件有深入了解,在图形能满足的基础上应首先采用。第二种是中层,通过视频设备驱动程序 BIOS INT 10H 来实现,它可以用汇编语言或 C 语言等来调用;本章汇集就是有关这种方法的资料。第三种是底层,直接用输出输入语句通过 I/O 端口对图形控制卡(适配器)上的寄存器等进行操作;这种方法比较繁琐,对硬件要有深入的了解,但是它能充分发挥图形卡的所有功能,在有特殊需要时,它是唯一的办法,我们在第三、四章中有过介绍。

本章内容分三部分:第一部分是 BIOS INT 10H 功能提要,实质是 BIOS 提供的显示子程序功能;第二部分概括了 IBM 标准视频模式及其兼容显示;第三部分比较详细地说明了视频 BIOS 各个功能。

举例说,我们要设定 VGA 640×480 像素、16 色图形模式。从第一部分提要中可一眼看出功能 0 是模式选择;从第二部分表中可查得模式号为 12H;从第三部分说明中可知编程为:

```
mov AH,0  
mov AL,12H  
int 10H
```

这段程序执行后,模式就已按要求设定。

8.1 视频 BIOS 功能提要

- 功能 0——模式选择
- 功能 1——设置光标的大小
- 功能 2——设置光标的位置
- 功能 3——读取光标的大小和位置
- 功能 4——读取光笔的位置
- 功能 5——选择当前显示页
- 功能 6——文本窗口向上滚动(或消隐文本窗口)
- 功能 7——文本窗口向下滚动(或消隐文本窗口)
- 功能 8——读取光标位置上的字符及属性
- 功能 9——在光标的位置上写字符及属性
- 功能 10(0AH)——在当前光标位置上写字符
- 功能 11(0BH)——设置 CGA 的彩色调色板(模式 4、5、6)
- 功能 12(0CH)——写图形象素
- 功能 13(0DH)——读图形象素
- 功能 14(0EH)——写字符并向前移动光标
- 功能 15(0FH)——读取当前显示模式
- 功能 16(10H)——设置 EGA 调色板寄存器
 - 子功能 0——设置单个调色板寄存器
 - 子功能 1——设置边界颜色
 - 子功能 2——设置所有的调色板寄存器
 - 子功能 3——闪烁/高亮属性控制
 - 子功能 7——读取单个调色板寄存器(仅对 VGA 有效)
 - 子功能 8——读取边界颜色寄存器(仅对 VGA 有效)
 - 子功能 9——读取所有的调色板寄存器(仅对 VGA 有效)
 - 子功能 10H——设置单个 DAC 寄存器(仅对 VGA 有效)

- 子功能 12H——设置一组 DAC 寄存器(仅对 VGA 有效)
- 子功能 13H——选择彩色子集(仅对 VGA 有效)
- 子功能 15H——读单个 DAC 寄存器(仅对 VGA 有效)
- 子功能 17H——读一组 DAC 寄存器(仅对 VGA 有效)
- 子功能 1AH——读子集的状态(仅对 VGA 有效)
- 子功能 1BH——使 DAC 寄存器转成灰度(仅对 VGA 有效)
- 功能 17(11H)——装载字符发生器
 - 子功能 0——装载用户自定义的字符发生器
 - 子功能 1——装载单色字符集
 - 子功能 2——装载 CGA 字符集
 - 子功能 3——选择当前字符集
 - 子功能 4——装载 VGA 16 线字符集
 - 子功能 20H——初始化 INT 1FH 矢量
 - 子功能 21H——设置自定义字符集的图形模式
 - 子功能 22H——设置 EGA 字符集的图形模式
 - 子功能 23H——设置 CGA 字符集的图形模式
 - 子功能 24H——设置 VGA 字符集的图形模式
 - 子功能 30H——读取有关当前字符集的信息
- 功能 18(12H)——EGA 状态/设置替代屏幕打印程序
 - 子功能 10H——返回 EGA/VGA 的配置信息
 - 子功能 20H——选择替代屏幕打印程序
 - 子功能 30H——设置文本模式的扫描线数(仅对 VGA 有效)
 - 子功能 31H——允许/禁止调色板的装载(仅对 VGA 有效)
 - 子功能 32H——允许/禁止 VGA 适配器
 - 子功能 33H——允许/禁止灰度转换
 - 子功能 34H——允许/禁止 CGA 光标仿真
 - 子功能 35H——切换显示
 - 子功能 36H——显示开/关

功能 19(13H)——写文本字符串

功能 26(1AH)——读或写配置信息(仅对 VGA 有效)

子功能 0——读显示配置代码

子功能 1——写显示配置代码

功能 27(1BH)——返回 VGA 的状态信息

功能 28(1CH)——保护/恢复显示适配器状态(仅对 VGA 有效)

子功能 0——返回所需的缓冲器大小

子功能 1——保护显示适配器状态

子功能 2——恢复显示适配器状态

8.2 标准 IBM 视频模式及兼容显示

表 8-1 标准 IBM 视频模式

模式号	类 型	彩色数	分辨率	兼容显示器
0,1	彩色文本	16	40×25 8×8 字符单元	CD、ED、VGA 多频率
0+,1+	彩色文本	16	40×25 8×14 字符单元	ED、VGA 多频率
0+,1+	彩色文本	16	40×25 9×16 字符单元	VGA 多频率
2,3	彩色文本	16	80×25 8×8 字符单元	CD、ED、VGA 多频率
2+,3+	彩色文本	16	80×25 8×14 字符单元	ED、VGA 多频率
2+,3+	彩色文本	16	80×25 9×16 字符单元	VGA 多频率

模式号	类 型	彩色数	分辨率	兼容显示器
4,5	彩色图形	4	320×200	CD,ED,VGA 多频率
6	彩色图形	2	640×200	CD,ED,VGA 多频率
7	单色文本	2	80×25 8×14 字符单元	单色 VGA
7-	单色文本		80×25 9×16 字符单元	只有 VGA
8,9,A	保留			
0DH	彩色图形	16	320×200	CD,ED,VGA 多频率
0EH	彩色图形	16	640×200	CD,ED,VGA 多频率
0FH	单色图形		640×350	单色 VGA
10H	彩色图形	16	640×350	ED,VGA 多频率
11H	彩色图形	2	640×480	VGA 多频率
12H	彩色图形	16	640×480	VGA 多频率
13H	彩色图形	256	320×200	VGA 多频率

注:CD=彩色显示器

#,+表示同一模式的不同情况

ED=增强型彩色显示器

表 8-2 IBM-PC 兼容显示

显示器	兼 容 适配器	彩 色 种 数	文本 分辨率	图 形 分辨率	扫 描 速率
单 色	MDA	2	80×25	640×350	垂直-50Hz 水平-15.8kHz
	Hercules			720×350	
	EGA			720×348	
彩色	CGA	16	40×25	320×200	垂直-60Hz
	EGA		80×25	640×200	水平-15.8kHz
增强 彩色	CGA	64 色中 的 16 色	40×25	320×200	垂直-60Hz
	EGA		80×25	640×200	水平-15.8kHz
					640×350
多频 数字	CGA	64 色中 的 16 色	40×25	320×200	可变
	EGA		80×25	640×200	
多频 模拟	VGA	256K 色中 的 256 色	80×25	640×480	可变
				800×600	
VGA 彩色	VGA	256K 色中 的 256 色	40×25	320×400	垂直-70Hz
			80×25	640×400	水平-31.5kHz
VGA 单色				320×350	
				640×350	
				720×350	
				720×400	
				640×480	

8.3 视频 BIOS 功能说明

8.3.1 功能 0——模式选择

BIOS 功能 0 可用来初始化任何标准操作模式的显示。

若 AL 寄存器 D₇ 位为 0, 显示存储器将随部分模式初始化的操作被清除; 若 D₇ 位为 1, 则显示存储器将保持不变。

EGA 在复位到模式 0、3 或 7 时将被缺省, 这取决于配置开关的设置。VGA 的缺省模式 3 或 7, 取决于将要使用的显示类型。

输入参数: AH=0

AL=模式号(0~13H)

8.3.2 功能 1——置光标尺寸

EGA 光标是一个字符宽, 1~32 个扫描线高的矩形, 其高度不能大于所用到的字符集的高度, BIOS 功能定义的光标高度将出现在一个相应的字符单元中。当使用不同字符集时该功能是产生软件兼容性问题的原因之一(例如当运行 CGA 软件时使用 EGA 增强字符集), 在这种情况下, 光标可能出现在相对于字符单元的错误位置上。

输入参数: AH=1

CH=开始扫描线(0~31)

CL=结束扫描线(0~31)

缺省设置: 单色显示: 开始=11, 结束=12

彩色显示: 开始=6, 结束=7

增强彩色: 开始=11, 结束=12

8.3.3 功能 2——置光标位置

该功能指定显示屏幕上光标的位置。对于允许多页显示的各种模式, 为每页保留一个分离的光标。不论显示页是否为当前活动页, 光标定位功能可访问任何显示页。除了确定光标在屏幕上何处出现外, 当前的光标位置还决定执行 BIOS 字符或字符串输出功能时, 下一个字符将在屏幕上何处出现。

输入参数: AH=2

BH=显示页号

DH=行(0~24)

DL=列(0~79)

8.3.4 功能 3——读光标尺寸与位置

数据返回光标在屏幕上的当前位置和光标的起始与结束扫描线(光标形状)。对任何有效的显示页,无论其是否为当前活动(显示)页,光标都可询知。

输入参数:AH=3

BH=显示页号

返回值:CH=光标起始扫描线

CL=光标结束扫描线

DH=光标行

DL=光标列

8.3.5 功能 4——取光笔位置

返回的信息与光笔是否在使用、是否被触发及其在屏幕的当前位置有关。位置信息以象素位置(对图形模式而言)和字符位置(对文本模式而言)两种形式返回。EGA 的光笔不与 CGA 的兼容,通常,使用光笔的 CGA 软件不能正确地在 EGA 上操作。VGA 适配器不支持光笔。

输入参数:AH=4

返回值:AH=0 指光笔未放下或未被触发

AH=1 指光笔放下或已被触发

CH=象素行(0~348)(总是 2 的倍数)

BX=象素列

(对于 320 列模式为 0~316,总是 4 的倍数)

(对于 640 列模式为 0~632,总是 8 的倍数)

DH=字符行(0~24)

DL=字符列(0~39 或 0~79)

8.3.6 功能 5——选择活动页

若规定的页号有效,则变成活动页并在屏幕上显示。不必为了从某页中读或写数据而选择该页,或访问该页光标。

输入参数:AH=5

AL=显示页号

有效页号为:模式 0、1、2、3、D:0~7 页

模式 E: 0~3 页

模式 F、10: 0~1 页

缺省设置:0 页

8.3.7 功能 6——文本窗口上卷(或空白窗口)

将当前的活动显示页的规定部分(卷动窗口)向上卷动规定的行数,在卷动窗口下部产生的空行用空格填充,该显示窗口外的所有显示区域保留不变,卷过该窗口顶部的数据则消失。

卷动窗口为一矩形,通过该窗口左上角和右下角的屏幕坐标来定义。当窗口的左上角和右下角与屏幕的相重叠时,本功能提供了一种清除屏幕的方便方法(使用卷动 25 或 0 行)。

输入参数:AH=6

AL=卷动的行数(AL=0 把窗口全部变成空格)

BH=填充窗口底部空行所使用的文本属性

CH=窗口左上角的行号(0~24)

CL=窗口左上角的列号(0~79)

DH=窗口右下角的行号(0~24)

DL=窗口右下角的列号(0~79)

8.3.8 功能 7——文本窗口下卷(或空白窗口)

当前的活动显示页的指定部分(卷动窗口)被向下卷动规定的行数,在卷动窗口顶部产生的空行由空格填充,显示窗口外的所有显示域保留不变,卷过该窗口底部的数据则消失。

卷动窗口为矩形,通过规定该窗口左上角和右下角的屏幕坐标来定义。当窗口和屏幕两者的左上、右下角相重叠时,该功能提供了一种清屏的简便方法(使用卷动 25 行或 0 行)。

输入参数:AH=7

AL=卷动的行数(AL=0 将窗口全部变成空格)

BH=填充窗口顶部空行所使用的文本属性

CH=窗口左上角的行号(0~24)

CL=窗口左上角的列号(0~79)

DH=窗口右下角的行号(0~24)

DL=窗口右下角的列号(0~79)

8.3.9 功能 8——在光标位置读字符和属性

该功能在任何显示页的当前光标位置返回 ASCII 字符,以及其字符属性。

输入参数:AH=8

BH=显示页号

返回值:AL=ASCII 字符

AH=字符属性

8.3.10 功能 9——在光标位置写字符和属性

在任何有效显示页的当前光标位置把 ASCII 字符和属性值写入显示存储器中。规定一个重复计数值,在这种情况下,在后续的存储器位置重复进行写操作,直至达到该重复计数值。若使用的重复计数值超过当前字符行尾端,则不能保证结果正确。

输入参数:AH=9

BH=显示页号

AL=ASCII 字符

BL=属性(文本)/彩色值(图形模式)

CX=重复计数(直到当前行的尾部)

8.3.11 功能 10(0AH)——在光标位置只写字符

在任何有效显示页的当前光标位置把 ASCII 字符写入显示存储器中,保存先前的字符属性。规定一个重复计数值,在这种情况下,在后续的存储器位置重复写操作,直至达到重复计数值,若使用一个试图跨越当前字符行的重复计数值,则不保证结果正确。

光标不会自动推移而是保留在当前位置。

输入参数:AH=0AH

AL=ASCII 字符

BH=显示页号

BL=彩色值(图形模式)

CX=重复计数值

若 EGA 以图形模式操作且寄存器 BL 的 D₇ 位等于 1, 则显示存储器中先前的数据和要写入的字符异或。

8.3.12 功能 11(0BH)——置 CGA 调色板(模式 4、5、6)

为了和 CGA 兼容, 本功能调用被包含在 EGA BIOS 中, 功能 0BH 使 EGA 可以模拟两个标准 CGA 图形彩色调色板之一的配置。表 8-3 描述了 CGA 的图形彩色调色板。

表 8-3 模式 4 和 5 的 CGA 彩色

图素值	调色板 0	调色板 1
0	同底色	同底色
1	绿	青
2	红	品红
3	棕	白

输入参数: AH=0BH

若 BH=0;

BL=图形底色(0~15)或文本边缘彩色(0~15)

若 BH=1;

BL=调色板号(0 或 1)

8.3.13 功能 12(0CH)——写图形象素

本功能可在图形模式下任一页画点, 0 列是屏幕最左边, 0 行是屏幕最顶行。表 8-4 概括了各种图形模式的有效值。

输入参数: AH=0CH

AL=象素值(见表 8-4)

BH=页号

CX=象素列号(0~639)

DX=象素行号(0~349)

表 8-4 BIOS 功能 12(0CH)的合法象素值

模式	合法的象素值(色彩)
4,5	0~3
6	0~1
D	0~15
E	0~15
F	0~1
10	0~15

若寄存器 AL 的 D₇ 位置成 1,则新的象素值与现有的底色异或。

8.3.14 功能 13(0DH)——读图形象素

本功能和功能 12 相仿,不过不是画点而是读点。

输入参数:AH=0DH

CX=象素列号(0~639)

BH=页号

DX=象素行号(0~349)

返回值:AL=象素值

8.3.15 功能 14(0EH)——写字符和推进光标

该功能有时被认为是电传机模式,因为当接受一个字符时其行为类似于一个 ASCII 终端。该字符显示在当前的光标位置,并且光标自动地推进到下一个字符位置。在一行的尾端,光标将绕到下一行首部。ASCII 码的振铃、回格、回车和换行被识别出,从而执行它们相应的功能。按要求执行垂直卷动。

若 EGA 在文本模式下操作,则字符属性保留不修改;若 EGA 在图形模式下操作,则在调用时要指定字符颜色。

对于屏幕处理,功能 14 是由标准 MS-DOS 控制台驱动。

输入参数:AH=0EH

AL=ASCII 字符

BH=页号(只在文本状态下)

BL=字符颜色(只在图形状态下)

8.3.16 功能 15(0FH)——取当前显示模式

本功能可用来确定 EGA 当前的操作模式,前面的表 8-2 概括了所有的有效模式。

输入参数:AH=0FH

返回值:AH=显示列号

AL=显示模式(0~13H)

BH=活动的显示页

8.3.17 功能 16(10H)——置 EGA 调色板寄存器

本功能是一种为 EGA 定义彩色或控制属性控制器的其它功能的优选方法,它会被用到四种不同形式中去:

子功能 0——置单个调色板寄存器

子功能 1——置边框彩色寄存器

子功能 2——置全部调色板寄存器

子功能 3——允许前景闪烁属性或底色增强属性。

下列功能仅可用于 VGA:

子功能 7——读单个调色板寄存器

子功能 8——读边框彩色寄存器

子功能 9——读全部调色板寄存器

子功能 10H——置单个 DAC 寄存器

子功能 12H——置多个 DAC 寄存器

子功能 13H——选择彩色子集

子功能 15H——读单个 DAC 寄存器

子功能 17H——读多个 DAC 寄存器

子功能 1AH——读彩色页状态

子功能 1BH——把 DAC 寄存器转变成灰度

上述各子功能分述如下:

子功能 0——置单个调色板寄存器

输入参数:AH=10H

AL=00H

BL=调色板寄存器号(0~F)

BH=彩色数据

子功能 1——置边框彩色

本功能设置属性控制器的边框彩色寄存器(有时称为过扫或边界扫描),使用该寄存器时需极其小心,因为边框彩色在许多 EGA 兼容产品的某些模式中不能正确工作。

输入参数:AH=10H

AL=01H

BH=彩色数据

子功能 2——置全部调色板寄存器

该功能提供一种快速编制所有 EGA 调色板寄存器程序的方法,调色板彩色数据必须放于系统存储器某个 17 字节表中,0 位到 15 字节包含调色板寄存器 0 到 15 号的数据,字节 16 是边框彩色。

输入参数:AH=10H

AL=02H

ES:DX=调色板数据的地址

子功能 3——闪烁/亮度属性控制

本功能提供了一种方便的触发控制位方法来确定是允许闪烁属性还是加亮背景属性。

输入参数:AH=10H

AL=03H

BL=0——允许底色加亮

BL=1——允许前景闪烁

子功能 7——读单个调色板寄存器

由于利用了 VGA 的可读寄存器,本功能返回调色板寄存器的当前内容。

输入参数:AH=10H

AL=7

BL=寄存器号(0~15)

返回值:BH=调色板寄存器值

子功能 8——读边框颜色寄存器

由于利用了 VGA 的可读寄存器,本功能返回边框彩色(边界扫描)寄存器的计数。

输入参数:AH=10H

AL=8

返回值:BH=边框彩色寄存器值

子功能 9——读全部调色板寄存器

输入参数:AH=10H

AL=9

ES:DX=指针指向 17 字节数据表

返回值:存在(ES:DX)的 17 字节

子功能 10H——置单个 DAC 寄存器

设置单个 DAC 寄存器 18 位彩色值。

输入参数:AH=10H

AL=10H

BX=DAC 寄存器号(0~255)

DH=红色亮度级(6 位)

CH=绿色亮度级(6 位)

CL=蓝色亮度级(6 位)

子功能 12H——置 DAC 块寄存器

设置 DAC 寄存器组 18 位彩色值。

输入参数:AH=10H

AL=12H

BX=起始 DAC 寄存器(0~255),典型值为 0。

CX=要设置的寄存器号(1~256),典型值为 64,

256。

ES:DX=彩色表地址

彩色表由每个寄存器 3 字节(红、绿、蓝)组成。

子功能 13H——选择彩色子集

本功能选择 1 至 16 个彩色子集。

输入参数:AH=10H

AL=13H

BL=0:选择模式

BH=0:64 种彩色的 1 个子集

BH=1:16 种彩色的 16 个子集

BL=1:选择子集

BH=子集(0~16)

子功能 15H——读单个 DAC 寄存器

输入参数:AH=10H

AL=15H

BX=DAC 寄存器号(0~255)

返回值:DH=红色亮度(6 位)

CH=绿色亮度(6 位)

CL=蓝色亮度(6 位)

子功能 17H——读 DAC 块寄存器

输入参数:AH=10H

AL=17H

BX=起始 DAC 寄存器号(0~255)

CX=寄存器号(1~256)

ES:DX=寄存器数据的目的地地址

返回值:在目的地地址的寄存器数据(每个寄存器 3 字节)

子功能 1AH——读子集状态

本子功能返回当前的彩色子集号。

输入参数:AH=10H

AL=1AH

返回值: BH=当前彩色子集号

BL=0: 4个可用集

BL=1: 16个可用集

子功能 1BH——DAC 寄存器转换为灰度值

本子功能把一组 DAC 寄存器从彩色值转换为单色的灰度值。对于每个寄存器,彩色数据读出并按加权相加计算(30%红、59%绿和 11%蓝),然后把结果写回寄存器的所有三种彩色成份,而原来的寄存数据则消失。

输入参数: AH=10H

AL=1BH

BX=起始 DAC 寄存器号(0~255)

CX=寄存器号(1~256)

8.3.18 功能 17(11H)——装入字符发生器

本功能介绍一种在文本模式操作时把字符发生器(标准的或常规的)装入显示存储器中的方法,适配器状态将重新初始化,但显示存储器不清除,无论是一部分还是完整的字符发生器都可装入。可使用下列不同形式:

子功能 0 和 16——装入常规字符发生器

子功能 1 和 17——装入标准单色字符集

子功能 2 和 18——装入标准 CGA 字符集

子功能 3 和 19——选择活动的 EGA 字符集(0~3)

子功能 4 和 14——装入 VGA 16 线字符集

子功能 32(20H)——为常规字符集初始化 CGA 图形(设置 INT 1FH 向量)

子功能 33(21H)——常规字符集初始化图形模式以显示文本(设置 INT 43H 向量)

子功能 34(22H)——初始化图形模式以显示标准增强(8×14)文本(设置 INT 43H 向量)

子功能 35(23H)——初始化图形模式以显示标准 CGA 文本
(设置 INT 43H 向量)

子功能 36(24H)——初始化图形模式以显示 VGA 16 线文本
(设置 INT 43H 向量)

子功能 48(30H)——返回有关当前字符集的信息

以上各子功能分述如下:

装入常规字符发生器——0

功能 17 这种形式允许使用程序装入一个常规字符集,或者用常规字符代替部分标准字符集。在调用本功能以前,需把用户字符表装入系统存储器,以后可把它从系统存储器中删去。

输入参数:AH=11H

AL=0

ES:BP=系统 RAM 中字符数据地址

CX=装入的字符数(1~256)

DX=补偿到字符发生器表中的字符

BL=装入 EGA 字符表(0-3)

BH=每个字符的字节数(1-32)

装入单色字符集——1

功能 17 的这种形式是把标准 EGA 单色字符集从 BIOS ROM 中装入显示存储器页面 2。

输入参数:AH=11H

AL=1

BL=装入 EGA 字符表(0-3)

装入 CGA 字符集——2

功能 17 的这种形式是把标准 CGA 8×8 字符集从 BIOS ROM 装入 EGA 显示存储器页面 2。

输入参数:AH=11H

AL=2

BL=装入 EGA 字符表(0~3)

选择活动字符集 --- 3

功能 17 的这种形式适用于选择四种 EGA 内部字符发生器表中的哪一个将被活化。在活化一个表前,必须使用功能 17 其它形式之一把该表和字符数据一同装入。

输入参数:AH=11H

AL=3

BL(D₀,D₁)属性位 3=0 时的文本字符选择活化的字符表

BL(D₂,D₃)属性位 3=1 时的文本字符选择活化的字符表

装入 VGA 16 线字符集 --- 4

把标准的 VGA 8×16 字符集装入显示存储器的页面 2 中。

输入参数:AH=11H

AL=4

BL=装入 VGA 字符表(0~7)

初始化 INT 1FH 向量(模式 4、5、6)——20H

如果要求多于 128 个字符,可在 CGA 兼容图形模式(方式 4、5、6)下使用本功能。向量 INT 1FH 用来指向一个附加的 128 字符表(ASCII 码 128~255),这时必须提供这些附加字符,所有字符必须是 8 像素高。

输入参数:AH=11H

AL=20H

ES:BP=系统 RAM 中装入字符集的地址

置图形模式为显示常规字符集——21H

本功能可以作为文本和图形组合应用来使用。适配器被初始化为图形模式、向量初始化,以使用常规字符集来描绘文本。字符集必须保留在系统存储器中,因为它不能被装入 EGA 显示存储器中。

输入参数:AH=11H

AL=21H

ES:BP—系统 RAM 中常规字符表地址

CX—每个字符字节

BL=显示的字符行数

BL=1:14 字符行

BL=2:25 字符行

BL=3:43 字符行

BL=0:用户自定义的行数,存于 DL 中。

置图形模式为显示增强文本——22H

该功能把适配器初始化为图形模式,并把它配置成显示标准的 EGA 增强字符集。

输入参数:AH=11H

AL=22H

BL=定义屏幕上的字符行数

BL=1:14 字符行

BL=2:25 字符行

初始化图形模式为显示标准 CGA 文本——23H

本功能把适配器初始化为图形模式并把它配置成显示标准的 CGA 字符集。

输入参数:AH=11H

AL=23H

BL=定义屏幕上的字符行数

BL=2:25 字符行

BL=3:43 字符行

初始化图形模式为显示 VGA 文本——24H

本功能将初始化适配器为图形模式并配置它来显示 VGA 16 象素高字符集。

输入参数:AH=11H

AL=24H

BL=定义屏幕上的字符行数

BL=1;14 字符行

BL=2;25 字符行

BL=3;43 字符行

返回有关当前字符集的信息——30H

输入参数:AH=11H

AL=30H

BH=需要的信息类型

BH=0;返回当前 INT 1FH 指针

BH=1;返回当前 INT 43H 指针

BH=2;返回指针到增强的(8×14)字符集

BH=3;返回指针到 CGA (8×8)字符集

BH=4;返回指针到 ROM 8×8 字符集高半部

BH=5;返回指针到备用的 9×14 单色字符

BH=6;返回指针到备用的 8×16 单色字符

BH=7;返回指针到备用的 9×16 单色字符

返回值:CL=字符高度(一个字符中的行数)

DL=屏幕上的字符行

ES:BP=返回指针

8.3.19 功能 18(12H)——EGA 状态/置替代屏幕打印

功能 18 实际上是一组享有同样功能号的独立功能,如下所示:

子功能 10H——关于当前 EGA/VGA 配置的返回信息

子功能 20H——选择替代打印屏幕程序

子功能 30H——置文本模式扫描线 (只对 VGA)

子功能 31H——允许/禁止调色板装入模式选择(只对 VGA)

子功能 32H——允许/禁止 VGA 适配器(只对 VGA)

子功能 33H——允许/禁止灰度转换(只对 VGA)

子功能 34H——允许/禁止光标模拟(只对 VGA)

子功能 35H——转换显示(只对 VGA)

子功能 36H——显示开/关(只对 VGA)

上述子功能说明如下:

关于当前 EGA/VGA 配置的返回信息——10H

除了返回关于显示适配器当前状态的有用信息外,本功能也用来决定:如果 EGA 或 VGA 出现在系统中,对系统就允许用自动配置;如果系统中没用 EGA 或 VGA,本功能调用将保留 BH 和 CH 不作修改,这一点能够用来测试 EGA 或 VGA 是否出现。

输入参数:AH=12H

BL=10H

返回值:BH=1 配器为单色模式

BH=0 适配器为彩色模式

BL=0 EGA 含 64KB 显示存储器

BL=1 EGA 含 128KB 显示存储器

BL=2 EGA 含 192KB 显示存储器

BL=3 EGA 含 256KB 显示存储器

CH=特征控制位

CL=EGA 配置开关

选择替代打印屏幕程序——20H

该功能在 43 行文本模式中有打印屏幕能力。

输入参数:AH=12H

BL=20H

置文本模式扫描线——30H

本功能通过设置屏幕分辨率,在用模式选择进行文本模式初始化时将用这一点来改善 VGA 与 EGA 的兼容性。然后,使用模式选择命令的 CGA 软件也可用于增强文本模式。

输入参数:AH=12H

BL=30H

AL=0;CGA(8×8)文本下二次扫描 200 线模式

AL=1; EGA 增强(8×14)文本下 350 扫描线模式

AL=2; VGA 增强(9×16)文本下 400 扫描线模式

返回值: AL=12H

允许/禁止调色板装入模式选择——31H

该功能可用来防止调色板在作模式选择时缺省数据的初始化,这一点在常规调色板已经被编程的情况下是有用的。

输入参数: AH=12H

BL=31H

AL=0: 允许根据模式选择初始化调色板

AL=1: 禁止根据模式选择初始化调色板

返回值: AL=12H

允许/禁止 VGA 适配器——32H

本功能可以禁止 VGA 响应任何 I/O 或存储器的读或写,显示不受影响。

输入参数: AH=12H

BL=32H

AL=0: 允许 VGA

AL=1: 禁止 VGA

返回值: AL=12H

允许/禁止灰度转换——33H

本功能将允许与禁止 VGA 的灰度转换特性。在允许时,模式选择在 DAC 寄存器装入后或 DAC 寄存器装入功能调用后,来执行灰度转换。

输入参数: AH=12H

BL=33H

AL=0: 允许灰度转换

AL=1: 禁止灰度转换

返回值: AL=12H

允许/禁止 CGA 光标仿真——34H

本功能用来解决 CGA 软件设置光标形状与使用 EGA 或 VGA 增强文本访问光标的兼容问题。由于增强文本用到较大的字符单元,光标将出现在错误字符单元的位置上,因此必须解决 CGA 光标仿真、允许传送不同的光标参数(光标开始和结束)值。

输入参数:AH=12H

BL=34H

AL=0;允许 CGA 光标仿真

AL=1;禁止 CGA 光标仿真

返回值:AL=12H

转换显示——35H

本功能可用来在 PS/2 系统的母板 VGA 和一块插入的显示适配器之间进行转换,而不用担心这两个适配器的存储器和 I/O 空间相冲突。PS/2 将插入适配器初始化,成为主显示器;本功能的四个变量必须以正确的顺序接着执行,以便完成显示转换。

输入参数:AH=12H

BL=35H

ES:DX=指向 128 字节状态保存缓冲

AL=0;关闭插入的显示适配器(必须先做)

AL=1;打开 VGA(必须第二步做)

AL=2;关闭活动显示适配器(对后继转换用)

AL=3;打开非现用的显示适配器(对后继转换用)

返回值:AL=12H

显示开/关——36H

本功能将允许与禁止 VGA 显示。

输入参数:AH=12H

BL=36H

AL=0;允许显示

AL=1;禁止显示

返回值:AL=12H

8.3.20 功能 19(13H)——写文本字符串

本功能允许申请传送一个完整的文本字符到 BIOS。该文本串可以是纯粹的 ASCII 数据,或包含嵌入属性数据。光标可推进到文本末,或保留不变。

输入参数:AH=13H

BH=显示页号

CX=字符计数(串长度)

DH=串开始处行号

DL=串开始处列号

ES;BP=系统 RAM 中源文本串地址

AL=模式

AL=0;BL=对所有字符的属性,光标不修改

AL=1;BL=对所有字符的属性,光标要修改。

AL=2;包含 ASCII 和属性的字符串,光标不修改。

AL=3;包含 ASCII 和属性的字符串,光标要修改。

8.3.21 功能 26(1AH)——读或写配置

本功能被用来读或修改关于系统中当前显示设备配置的有关信息。它由二个子功能组成:

子功能 0——读显示配置代码

输入参数:AH=1AH

AL=0

返回值:AL=1AH

BL=主显示

BH=辅显示

显示信息释意如下:

0=无显示

1=MDA

2=CGA

3=使用 ECD 显示器的 EGA

4=使用 CD 显示器的 EGA
 5=使用单色显示器的 EGA
 6=PGC(专用图形控制器)
 7=使用单色显示器的 VGA
 8=使用彩色显示器的 VGA
 0BH=使用单色显示器的 MCGA
 0CH=使用彩色显示器的 MCGA

子功能 1——写显示配置代码

输入参数:AH=1AH

AL=1

BL=主显示信息

BH=辅显示信息

返回值:AL=1AH

8.3.22 功能 27(1BH)——返回 VGA 状态信息

输入参数:AH=1BH

BX=0

ES:DI=为返回数据指针指向 64 字节缓存

返回值:AL=1BH

返回缓存的信息如下:

字节	长度	说明
0~3	2 个字	指向状态函数表
4	1 字节	当前显示模式
5,6	1 个字	可显示的字符列数
7,8	1 个字	视频数据区的字节数
9,0AH	1 个字	视频数据区的起始地址
0BH—1AH	8 个字	0—7 页的光标位置
1BH	1 字节	光标起始线

1CH	1 字节	光标结束线
1DH	1 字节	当前显示页
1EH	1 个字	CRT 控制器端口地址(3B4/3D4)
22H	1 字节	可显示的字符行数
23H	1 字节	字符像素高度
25H	1 字节	第一 DCC(适配器)
26H	1 字节	第二 DCC(适配器)
27H,28H	1 个字	当前模式的可用彩色数
29H	1 字节	当前模式的显示页面数
2AH	1 字节	当前模式的扫描线数: 0=200,1=350,2=400,3=480
2BH	1 字节	基本字符映象(0~7)
2CH	1 字节	辅助字符映象(0~7)
2DH	1 字节	混合状态信息(按位存): D ₀ =1:所有模式可用 D ₁ =1:允许灰度转换 D ₂ =1:连接单色显示器 D ₃ =1:允许缺省调色板初始化 D ₄ =1:允许 CGA 光标仿真 D ₅ =1:允许闪烁 D ₅ =0:允许底色增强
31H	1 字节	视频存储容量(以 64K 块计,0=64K)

32H 1 字节 存指针状态信息：
 $D_0=1$: 用 512 字符集
 $D_1=1$: 使用调色板保存区
 $D_2=1$: 使用字符模式字符集
 $D_3=1$: 使用用户图形字符集
 $D_4=1$: 使用用户调色板集
 $D_5=1$: 使用 DCC 扩展

状态函数表(16 字节)格式化如下:

字节	长度	说明																														
0—2	3 字节	<p>每种模式支持一位的视频模式表</p> <table border="1"> <thead> <tr> <th>字节 0</th> <th>字节 1</th> <th>字节 2</th> </tr> <tr> <th>位/模式</th> <th>位/模式</th> <th>位/模式</th> </tr> </thead> <tbody> <tr> <td>$D_7—7$</td> <td>$D_7—FH$</td> <td>D_7</td> </tr> <tr> <td>$D_6—6$</td> <td>$D_6—EH$</td> <td>D_6</td> </tr> <tr> <td>$D_5—5$</td> <td>$D_5—DH$</td> <td>D_5</td> </tr> <tr> <td>$D_4—4$</td> <td>$D_4—CH$</td> <td>D_4</td> </tr> <tr> <td>$D_3—3$</td> <td>$D_3—BH$</td> <td>$D_3—13H$</td> </tr> <tr> <td>$D_2—2$</td> <td>$D_2—AH$</td> <td>$D_2—12H$</td> </tr> <tr> <td>$D_1—1$</td> <td>$D_1—9$</td> <td>$D_1—11H$</td> </tr> <tr> <td>$D_0—0$</td> <td>$D_0—8$</td> <td>$D_0—10H$</td> </tr> </tbody> </table>	字节 0	字节 1	字节 2	位/模式	位/模式	位/模式	$D_7—7$	$D_7—FH$	D_7	$D_6—6$	$D_6—EH$	D_6	$D_5—5$	$D_5—DH$	D_5	$D_4—4$	$D_4—CH$	D_4	$D_3—3$	$D_3—BH$	$D_3—13H$	$D_2—2$	$D_2—AH$	$D_2—12H$	$D_1—1$	$D_1—9$	$D_1—11H$	$D_0—0$	$D_0—8$	$D_0—10H$
字节 0	字节 1	字节 2																														
位/模式	位/模式	位/模式																														
$D_7—7$	$D_7—FH$	D_7																														
$D_6—6$	$D_6—EH$	D_6																														
$D_5—5$	$D_5—DH$	D_5																														
$D_4—4$	$D_4—CH$	D_4																														
$D_3—3$	$D_3—BH$	$D_3—13H$																														
$D_2—2$	$D_2—AH$	$D_2—12H$																														
$D_1—1$	$D_1—9$	$D_1—11H$																														
$D_0—0$	$D_0—8$	$D_0—10H$																														
7	1 字节	<p>允许使用的分辨率</p> <p>$D_0=1$: 200 线</p> <p>$D_1=1$: 350 线</p> <p>$D_2=1$: 400 线</p>																														

8	1 字节	文本模式字符映象个数
9	1 字节	活动文本模式字符的最大个数
0AH	1 字节	混合功能(相应位为 1 时): D ₀ :所有模式可用 D ₁ :灰度转换 D ₂ :字符字形装入 D ₃ :缺省调色板装入 D ₄ :CGA 光标仿真 D ₅ :EGA 调色板 D ₆ :DAC 调色板 D ₇ :多重 DAC 彩色表
0BH	1 字节	混合功能(相应位为 1 时) D ₀ :光笔支持 D ₁ :视频状态保存/恢复 D ₂ :允许闪烁 D ₃ :显示配置代码(DCC 表)
0EH	1 字节	保存指针功能(相应位为 1 时): D ₀ :512 字符集 D ₁ :调色板保存区 D ₂ :用户字符模式字符集 D ₃ :用户图形模式字符集 D ₄ :用户调色板集 D ₅ :DCC 扩展

8.3.23 功能 28(1CH)——保存/恢复显示适配器状态

本功能包含三个子功能:

子功能 0——返回所需缓冲区大小

子功能 1——保存显示适配器状态

子功能 2——恢复显示适配器状态

分述如下:

返回所需缓冲区大小——0

本子功能提供为了返回存储状态信息,必须由主机提供所需要的缓冲区容量数。

输入参数:AH=1CH

AL=0

CX=被保存的数据类型:

0=寄存器

1=BIOS 数据区

2=DAC 寄存器

返回值:AL=1CH

BX=需要的缓冲区容量(以 64 字节为单位)

保存显示适配器状态——1

本子功能将适配器状态数据存入由调用程序提供的缓冲区。

输入参数:AH=1CH

AL=1

CX=被保存数据的类型

0=寄存器

1=BIOS 数据区

2=DAC 寄存器

ES;BX=指向缓冲区的指针

返回值:AL=1CH

恢复显示适配器状态——2

输入参数:AH=1CH

AL=2

CX=被恢复的数据类型

0=寄存器

1 = BIOS 数据

2 = DAC 寄存器

ES:BX = 指向缓冲区的指针

返回值: AL = ICH

术语索引

1's complement	二进制反码	5. 3. 4
2's complement	二进制补码	5. 3. 4
active window	工作窗口	5. 3. 2
active matrix	有源阵	2. 2. 3
add-with-saturate	饱和加	5. 2. 3
AI(Application Interface)	应用程序接口	4. 2. 1
ALU(Arithmetic Logic Unit)	算术逻辑部件	4. 2. 1, 5. 2
amorphous	非晶	2. 2. 3
ANA (Analog)	模拟(信号接口)	2. 1. 2
analog modulation	模拟调制	2. 3. 4
analog signal input	模拟信号输入	2. 1. 2
annotation text	注释正文	7. 1. 3
ANSI(American National Standards Institute)	美国国家标准学会	7. 4. 2
anti-aliasing	平滑	5. 2. 3
aperture index	口径变址	4. 4. 4
arbitration	仲裁	5. 1. 2
ASCII(American Standard Code for Information Interchange)	美国信息交换标准码	3. 1. 2
band width	带宽	2. 1. 2
barrel shifter	桶式移位寄存器	4. 2. 1
BBLT(Bus Block Transfer)	总线块传送	5. 1
beam-shaping	电子束成形	2. 1. 1

bend	曲向	2. 2. 1
big endian	降序	5. 3. 2
binary mask	二进制屏蔽	5. 3. 2
BIOS(Basic I/O System)	基本输入输出系统	第八章
bit field	位段	5. 2. 2
bit map	位图	5. 1. 2, 4. 4. 2
bit mask register	位屏蔽寄存器	3. 2. 2
bit plane	位面	3. 2. 2
bit-addressable	可位寻址	5. 2. 2
bitblt	位块传送	4. 3. 1
built-in bus-fault	内置总线故障	5. 3. 2
burst rate	突发	5. 2
bus mastering	总线支配	4. 4. 1, 4. 4. 4
bus-contention	总线争用	5. 3. 2
BWM(Block-Write Mode)	块写方式	5. 3. 3
byte-swapped addressing	字节交换寻址	5. 1. 1
c-packet	c-信息包	4. 2. 2
cache	高速缓冲器	5. 2
cache miss	高速缓冲器空缺	5. 2. 2
CAD(Computer Aided Design)	计算机辅助设计	1. 1
CAM(Computer Aided Manufacturing)	计算机辅助制造	1. 5
carry chain	进位链	5. 2. 2
CAS(Column Address Strobe)	列地址选通	5. 2. 1
CBLT(Character Block Transfer)	字符块传送	5. 1. 3
CCITT (Consultative Committee, International Telegraph and Telephone)	国际电报电话咨询委员会	5. 3
CCSV(Counter balanced CRT-based Stereoscopic Viewer)	配重平衡CRT式立体观察器	2. 5
CD(Communication Driver)	通信驱动程序	4. 2. 1

CD(Cache Disable) 高速缓冲器禁止	5. 2. 2
CD ROM(Compact Disk-ROM) 光盘只读存储器	5. 3
cell array 单元阵列	7. 1. 3
CGA(Color Graphics Adapter) 彩色图形适配器	3. 1. 4
CGE(Color Graphics Enhancer 400) 彩色图形增强 400 型卡	3. 2. 3
CGI(Computer Graphics Interface) 计算机图形接口	7. 3. 2
CGM(Computer Graphics Metafile) 计算机图形元文件	7. 2. 2
character clock 字符时钟	3. 3. 4
cholesteric phase 胆甾醇相	2. 2. 1
CISC(Complex Instruction Set Computer) 复杂指令集计算机	6. 3. 1
clipping 剪裁	5. 2. 2
color page 色彩页	3. 3. 6
color-expand 彩色扩充	5. 3. 1
compatibility 兼容性	2. 1. 2
computer graphics 计算机图形	1. 1
core primitive 核心原语	4. 2. 1
CRT(Cathode Ray Tube) 阴极射线管	2. 1. 1
CRTC(CRT Controller) CRT 控制器	3. 3. 7
CSS(Center Struture Storage) 中心结构存贮器	7. 1. 3
CT(Computer Tomography) 计算机 X 射线断层造影	1. 1
DAC(Digital-Analog Converter) 数模转换器	4. 1. 2
data field 数据字段	5. 2. 1
data path 数据通路	5. 2. 1
DC(Device Coordinate) 设备坐标空间	7. 3. 2
dichroic dye 二色性染料	2. 2. 2
discrimination ratio 鉴别率	2. 4. 1
DMA(Direct Memory Access) 直接存储器存取	5. 2. 1

DMMU (Discrete MMU)	分立 MMU	6. 2. 2
dot clock	点时钟	3. 3. 4
dot matrix	点阵	2. 3. 2
dot rate	点频	2. 1. 2
double buffering	双缓冲	6. 1. 2
DSM (Dynamic Scattering Mode)	动态散射方式	2. 2. 2
dual operating states	双操作状态	6. 2. 1
dual-bank	双组	5. 1. 1
duty cycle modulation	占空度调制	2. 3. 4
DVMA (Direct Virtual Memory Access)	直接虚拟存储器存取	6. 3. 3
EGA (Enhanced Graphics Adaptor)	增强型图形适配器	3. 2. 1
EL (Electro Luminescent)	场致发光	2. 4. 1
electromagnetic interference	电磁干扰	2. 1. 2
electrophoretic display	电泳显示	第二章
entity	实体	7. 4. 3
ESDI (Enhanced Small Device Interface)	增强型小型设备接	6. 2. 2
目		
ethernet	以太网	6. 2. 3
event	事件	7. 1. 3
extended primitive	扩展原语	4. 2. 1
FA (First Access)	首次访问	5. 1. 2
face treatment	表面处理	2. 1. 2
fast bit-addressable processor	快速位寻址处理器	5. 3
fast page mode	快速页面方式	5. 3. 1
fast solid filling	快速实心填充	5. 3. 3
FED (Field-Emission Display)	场发射显示	2. 1. 4
field-extension bit	段扩展位	5. 2. 2

field-width-programmable	段宽可编程	5. 2. 2
FIFO (First In First Out)	先进先出	7. 1. 3
fill area	填充区	7. 1. 3
fill area set	填充区组	7. 1. 3
flange	凸缘	7. 4. 2
font page	字型页	3. 3. 3
FPU (Floating-Point Coprocessor)	浮点协处理器	5. 3. 1
fractal	分形	第七章
front-throw	前投式	2. 1. 3
FTM (Flat Tension Mask)	平张力荫罩	2. 1. 1
full-screen cross hair	全屏幕十字线标	5. 1. 1
gas-plasma	气体等离子体	2. 3. 1
GB (GigaByte)	京字节, 1024 兆字节	4. 4. 3
general-purpose register file	通用寄存器堆	5. 2. 2
generalized drawing primitive	广义绘图元	7. 1. 3
GI (Grant In)	许可进	5. 3. 2
gigabit	京位	5. 2. 2
GKS (Graphical Kernel System)	图形核心系统	7. 1. 1
GKSM (GKS Metafile)	GKS 元文件	7. 2. 1
GM (Graphics Manager)	图形管理程序	4. 2. 1
GMR (Graphics metafile Resources)	图形元文件资源	6. 2. 2
GP (Graphics Processor)	图形处理器	第五章
graphical primitive	图元	7. 1. 1
graphics	图形	第一章
graphics engine	图形机	5. 3. 1
graphics metafile	图形元文件	7. 1. 1
GSP (Graphics System Processor)	图形系统处理器	5. 2
GSR (Graphics Service Routines)	图形服务程序	6. 2. 2
GVDFS (Generic Virtual Device Function Set)	通用虚拟设备功	

能集	7. 3. 1
halftone 网目	2. 3. 5
HDTV (High Definition Television) 高清晰度电视	第一章
HGC(Hercules Graphics Card) 赫尔克里士图形卡	3. 1. 1
horizontal scan frequency 水平扫描频率	2. 1. 2
host port 主口	5. 2
HP (Host Processor) 主处理器	1. 4
Huffman 霍夫曼	5. 3
IGES(Initial Graphical Exchange Specification) 初始图形交换	
规范	7. 4. 2
image 图象	第一章
image processing 图象处理	1. 1
information-modeling 信息建模	7. 4. 2
instruction acquisition 指令采集	5. 3. 2
instruction mix 指令混合	5. 3. 1
intelligent Registration Interface System 智能化重合接口系统	
	2. 1. 3
interlaced 隔行	2. 1. 2
Invar 一种热稳定性高的铁镍合金	2. 1. 1
IPC(Integrated Personal Computer) 集成个人机	6. 2. 3
ISO(International Standard Organization) 国际标准化组织	
	7. 1. 1
ITO(Indium-Tin Oxide) 铟-氧化锡	2. 4. 2
laptop 膝上型计算机	1. 2
large-area thermionic-oxide-coated 大面积热氧化物涂复	
	2. 1. 4
laser recrystallized 激光再结晶	2. 2. 3
LCD (Liquid Crystal Display) 液晶显示	2. 2. 1
LED (Light-Emitting Diode) 发光二极管	第二章

line-drawing engine	画线机	5. 3. 1
link address	连接地址	5. 1. 2
link-command bit	连接命令位	5. 1. 1
little endian	升序	5. 3. 2
LMO(Left Most Ones)	最左端 1	5. 2. 3
LRU (Least-Recently-Used)	当前最少使用	5. 2. 2
LST DMOS (Large Scale Integrated Double Diffusion Metal Oxide Semiconductor)	大规模集成双扩散金属氧化物半导体	2. 4. 1
machine-state time	机器状态时间	5. 2
macintosh portable	麦金塔什便携机(美国苹果电脑公司)	2. 2. 3
map mask register	映象屏蔽寄存器	3. 2. 2
mask-merge	屏蔽一合并	5. 2
MCGA (Multicolor Graphics Array)	多色图形阵列	3. 3. 2
MDA(Monochrome Display Adaptor)	单色显示适配器	3. 1. 1
memory map	内存图	5. 3. 2
memory-mapped frame buffer	存储器映射帧缓冲器	4. 4. 1
message field	信息段	5. 2. 1
MFLOPS (Millions of Floating-point Operation Per second)	百万次浮点运算/秒	6. 3. 1
micro channel	微通道	4. 4. 1
microcode	微码	5. 2. 1
microsequencer	微定序器	5. 3. 4
midline reloading	中行再装入	5. 3. 1
MIPS(Million Instructions Per Second)	百万条指令/秒	5. 2
misconvergence	错聚	2. 1. 2
MMU (Memory Management Unit)	存储器管理单元	6. 2. 2
MPS (Million Pixels per Second)	兆像素/秒	5. 1
multi-media	多媒体	第一章

multisensory	多感觉	2.5
nematic phase	向列相	2.2.1
NFS (Network File System)	网络文件系统	6.2.3
NIXIE	电子管式辉光数码管	2.3.2
non interlaced	逐行	2.1.2
nucleus	核	6.2.2
NURB	非均匀有理 B 样条	7.1.3
ordered dither	有序高频颤动	2.3.5
organic filter	有机滤光器	2.2.1
overlay plane	重迭平面	6.1.2
packed bit-map form	象素为单位的位图形式	5.1.1
packed-pixel	集装象素,把象素所有位装入同一存储单位的象素	5.2.2
page-mapping table	页面转换表	4.4.4
pattern	图案,花式	5.3.2
pattern recognition	模式识别	1.1
PCC(Personal Computer Coprocessor)	个人机协处理器	6.2.2
PDES(Product Definition Exchange Specification)	产品定义交换规范	7.4.2
Penning mixture	潘宁混合气	2.3.3
per-page access checking	每页存取校验	6.2.2
persistent write per bit	每位持续写	5.4
PEX	PHIGS 的 X 扩展	6.3.2
PGA (Professional Graphics Adaptor)	专业型图形适配器	1.4
PHIGS (Programmer's Hierarchical Interactive Graphics System)	程序员级分层交互图形系统	7.1.3
pipe line	流水线	6.1.1, 5.2
pixblt	象素块传送	5.2.3
pixel rate	象素频率	2.1.2

pixel-formatted register	象素格式寄存器	5. 3. 1
pixel-replication	象素复制	5. 1
plane masking	平面屏蔽	5. 2. 1
planar pixel-organized	平面象素组织,线性组织	5. 2. 2
poll mode	访问模式	5. 1. 1
polycrystalline	多晶	2. 2. 3
polyline	折线	7. 1. 3
polymarker	多点标记	7. 1. 3
post clipping	后剪取	5. 2. 3
post-processor	后处理程序	7. 4. 4
postincrement	后置增量	5. 2. 3
pre-clipping	前剪取	5. 2. 3
pre-processor	前处理程序	7. 4. 4
predecrement	预减量	5. 2. 3
program swapping	程序交换调动	5. 1. 3
PROM (Programmable ROM)	可编程 ROM	6. 2. 2
protected-mode	保护模式	4. 4. 4
pseudo gray scale	伪灰度	2. 3. 5
PWGAL (Personal Workstation Graphics Array 1)	个人工作站图形阵列 1, 西方数字公司的芯片组	4. 3. 3
quadrilateral mesh	四边形网格	7. 1. 3
RAM(Random Access Memory)	随机存储器	5. 1. 1
random port	随机端口	5. 4
RAS (Row Address Strobe)	行地址选通	5. 2. 1
raster sequence	光栅信号序列	5. 1. 1
rear-lighting, backlit	背投光	2. 2. 1
rear-throw	背投式	2. 1. 3
reformatted	重新格式	5. 3. 2
request	请求	7. 1. 3

resolution	分辨率	2. 1. 2
RH(High-priority Request)	高优先级请求	5. 3. 2
rib	凸条	7. 4. 2
RISC(Reduced Instruction Set Computer)	精减指令集计算机	6. 3. 1
RL(Low-priority Request)	低优先级请求	5. 3. 2
RMW(Read-Modify-Write)	读改写	5. 1. 1
ROM (Read Only Memory)	只读存储器	第三章
ruled surface	直纹曲面	7. 4. 2
SA (Subsequent Access)	随后访问	5. 1. 2
sample	采样	7. 1. 3
screen depth	屏图深度	6. 1. 2
SCST (Small Computer System Interface)	小型计算机系统接口	6. 3. 2
segment	图段	7. 1. 1
segment transformation	图段变换	7. 1. 1
self-scan	自扫描	2. 3. 2
semi-autonomous memory controller	半自治存储控制器	5. 3. 1
shading	影调、浓淡图	5. 3. 2
short stroke vector	短划向量	4. 3. 2
short vector	短向量	4. 4. 4
SID (Society for Information Display)	信息显示协会	2. 2. 1
single-point	单点	2. 1. 4
slave bus	从总线	5. 2. 1
smectic phase	层列相	2. 2. 1
SNAP (System Network Administration Program)	系统网络管理程序	6. 3. 3
solid modeling	实体造型	7. 4. 2
spatial modulation	空间调制	2. 3. 4

spind type field emission cathode	场发射阴极	2. 1. 4
split shift register	分裂式移位寄存器	5. 3. 1
spot size	光斑大小	2. 1. 2
stacked-panel passive-matrix	迭层式无源阵	2. 2. 4
stand-alone	单机	5. 2. 1
string	字符串	7. 1. 3
stroke	笔划	7. 1. 3
structure	结构	7. 1. 3
structure element	结构元素	7. 1. 3
subsegment	子段	5. 2. 2
subsequent word	子序列字	5. 2. 3
subtract-with saturate	饱和减	5. 2. 3
supertwist	超扭	2. 7. 1
sustain voltage	保持电压	2. 3. 2
tabulated cylinder	表列园柱	7. 4. 2
tap point	抽头点	5. 4
text	正文	7. 1. 3
TFT(Thin-Film Transistor)	薄膜晶体管	2. 2. 3
throughput	吞吐率	5. 1. 2
TIGA (Texas Instruments Graphics Architecture)	得克萨斯仪 器公司图形结构	第四章
tolerance envelop	公差包	7. 4. 2
torr	托, 1/760 大气压	2. 1. 4
transform engine	转换机	6. 1. 1
triangle net	三角形网	7. 1. 3
tridot pitch	三色点距	2. 1. 2
TTL (Transistor-Transistor Logic)	数字(信号接口)	2. 1. 2
twist	扭向	2. 2. 1
twisted nematic	扭向列	2. 2. 2

UGI (User's Graphics Interface)	用户图形界面	第一章
UIMS (User Interface Management System)	用户界面管理系统	第七章
ultrix	DEC 公司的 Unix 版本	6.3.2
underlay plane	衬垫平面	6.1.2
user-extended primitive	用户扩展原语	4.2.1
variable width	可变宽度	5.2
VDC (Virtual Device Coordinate)	虚拟设备坐标空间	7.3.2
VDI (Virtual Device Interface)	虚拟设备接口	7.3.1
VDM (Virtual Device Metafile)	虚拟设备元文件	7.2.2
vertical scan frequency	垂直扫描频率	2.1.2
VESA (Video Electronics Standards Association)	视频电子标准协会	4.1.1
VFD (Vacuum Fluorescent Display)	真空荧光显示	第二章
VGA (Video Graphics Array)	视频图形阵列	3.3.1
VGd (Virtual Graphic Device)	虚拟图形设备	7.3.1
video pixel rate	视频像素速率	5.1.1
VIEW (Virtual Interface Environment Workstation)	广角头盔式立体显示系统	2.5
viewing pipeline	视见变换流水线	7.1.3
VLSI	超大规模集成电路	4.3.1
VRAM	视频 RAM	5.4
window descriptor block	窗口描述符块	5.1.1
window segment	窗口段	5.1
WISS (Workstation Independent Segment Storage)	与工作站独立的图段存储	7.1.1
work group	工作群	6.2.1
WSI (WorkStation Interface)	工作站接口	7.3.1
XGA (extended Graphics Array)	扩展图形阵列	4.4.1

XUI(X User Interface)	X 用户接口	6.3.2
zoom	缩放	5.1.1

参考书目及文献

- 金廷赞. 计算机图形学. 杭州: 浙江大学出版社, 1988.
- 唐荣锡等. 计算机图形学教程. 北京: 科学出版社, 1990.
- 明智. EGA/VGA 应用开发指南. 北京: 清华大学出版社, 1990.
- 陶伟等. EGA、VGA、TVGA 高级微机图形编程指南与实例. 北京: 北京希望电脑公司, 1991.
- 赫克刚等. 计算机图形核心系统 GKS 引论. 西安: 西北大学出版社, 1986.
- 美 Chan S. Park. 交互式微型计算机图形学. 陆吟芳等译. 北京: 清华大学出版社, 1988.
- 美 S · 戴维斯. 计算机数据显示器. 长工译. 北京: 科学出版社, 1977.
- 美 S · Harrington. 计算机图形学. 高福文等编译. 北京: 北京师范大学出版社, 1985.
- 美 T · 帕夫利迪斯. 计算机图形显示和图象处理的算法. 吴成柯译. 北京: 科学出版社, 1987.
- 张福炎. 交互式计算机图形软件标准评述. 计算机研究与发展, No. 8. 1990.
- 王知衍, 刘丹琳. CAD/CAM 系统间数据交换方法的研究, 中国 CAD/CAM 应用与研究. 成都: 电子科技大学出版社, 1992. 8. 123~126
- 王知衍, 王新龙. 图形标准 PHIGS 性能及评价, 中国 CAD/CAM 应用与研究. 成都: 电子科技大学出版社, 1992. 8. 46~50

Apiki, Steve, et al. The brains behind the graphics. Byte, vol. 14, No. 12; 178-198 Nov. 1989.

Arnold, David. Computer graphics interface and CAD applications. CAD vol. 19, No. 8; 444-450 Oct. 1987.

BARCO, Largerear-screen projection in high ambient

Baran, Nick. Sun's low-cost RISC. Byte, vol. 15, No. 6; 136-137. Jun. 1990

Bissell, Don. The father of computer graphics. Byte, vol. 15 No. 6; 380-381. June 1990.

Bloor, M. S. , & Owen, J. CAD/CAM product-data exchange; the next step. CAD. vol. 23, no. 4; 237-243 May 1991.

Bogoch, Sam, et al. Supercomputers get personal. Byte, vol. 15, No. 5; 231-238 May 1990.

Brinkmann, Ron M. 3-D graphics, from alpha to z-buffer. Byte, vol. 15, No. 7; 271-278 July 1990.

Brodlic, K. W. , et al. The new graphical kernel system. CAD. vol. 23, No. 4; 312-318 May 1991.

Cook, Rick. Clash of the graphics titans. Byte, vol. 14, No. 11; 143-156

Covington, Michael A. Smooth views. Byte, vol. 15, No. 5; 279-283 May, 1990.

Duce, D. A. & Hopgood, F. R. A. The graphical kernel system. CAD vol. 19, NO. 8; 396-409 Oct. 1987.

Ellsworth, D. et al. Distributing display lists on a multicomputer. CG. vol. 24, No. 2; 147-154 Mar. 1990.

Encarnacao, J. , et al. Product data interfaces in CAD/CAM applications. Springer-Verlag, Berlin, 1986.

Finnie J. Scot. Dual-page duel; two high-resolution monitors square off. Byte, vol. 15, no. 9; 212-214 Sep. 1990.

Fisher, Scott S. , & Tazelaar, Jane Morrill. Living in a virtual world. Byte; 215-221 Jul. 1990.

Foley, James D. , & Dam, Andries Van. Fundamentals of computer graphics. Addison-Wesley, U. S. A. 1982.

Galyean, T. A. & Hughes J. F. Sculpting; an interactive volumetric modeling technique. CG. vol. 25, No. 1; 267-274 July 1991.

Glass, L. Brett. Digital video interactive. Byte, vol. 14, No. 5; 283-289 May 1989.

Glass, L. Brett. Inside EISA. Byte, vol. 14 No. 12; 417-425 Nov. 1989.

Goldman, G. et al. Computer graphics and architectural design. CG. vol. 25, No. 3; 174-177 Jul. 1991.

Hansen, Wilfred J. Data structures in a bit-mapped text editor. Byte, vol. 12, No. 1; 183-189 Jan. 1987.

Hart, J. C. & DeFanti, T. A. Efficient antialiased rendering of 3-D linear fractals. CG. vol. 25, No. 4; 91-100 July 1991.

Hayes, Frank. Intel's Cray-on a-chip. Byte, vol. 14, No. 5; 113-114 May 1989.

Howard, T. L. J. Evaluating PHIGS for CAD and general graphics application. CAD. vol. 23, No. 4; 244-251 May 1991.

Hsiung, P, et al. T-buffer; fast visualization of relativistic effects in spacetime. CG. vol. 24, No. 2; 83-88 Mar. 1990.

Miastkowski, Stan. Right before your eyes. Byte, Vol. 15, No. 8; 110 Aug. 1990.

Miller, J. V. , et al. Geometrically deformed models; a method for extracting closed geometric models from volume data. CG. vol. 25, No. 4. July 1991.

Mumford, A. M. Computer graphics metafile standard-an update. CAD. vol. 23, No. 23, No. 4; 303-305 May 1991.

Mumford, A. M. Why care about the computer graphics metafile?. CAD, vol. 19, No. 8; 425-430 Oct. 1987.

Muraki, Shigeru. Volumetric shape description of range data using blobby model. CG, vol. 25, No. 4; 227-230 July 1991.

Nicholls, Bill. Graphics; the big picture. Byte, vol. 13, No. 11; 129-138 Nov. 1988.

Nicholls, Bill. High-performance graphics boards. Byte, vol. 13, No. 1; 151-155 Jan. 1988.

Nicholls, Bill. Inside the 82786 graphics chip. Byte, vol. 12, No. 9; 135-141 Aug. 1987.

Nicholls, Bill. Is it really supper. Byte, vol. 14, No. 11; 159-164

Pappas, Chris H. , & Murray, William H. EGA times 12. Byte, vol. 12, No. 1; 313-326 Jan. 1987.

Park, Chan S. Interactive microcomputer graphics. Addison-Wesley, U. S. A. 1985.

Pentland, A. , et al. The thing would modeling system; virtual sculpting by modal forces. CG, vol. 24, No. 2; 143-144 Mar. 1990.

Plachn, martin. PHIGS; programmer's hierachical interactive graphics standard. Byte, vol. 12, No. 13; 275-286 Nov. 1987.

Rash Jr. Wayne. Light, bright, and white. Byte, vol. 13, No. 13; 321-324 Dec. 1988

Robinson, Phillip. A world of workstations. Byte, vol. 12, No. 13; 251-260 Nov. 1987.

Schilling, A. A new simple and efficient antialiasing with subpixel mask. CG, vol. 25, No. 4; 133-141 July 1991.

Shuey, D. PHIGS; a graphics platform for CAD application development. CAD, vol. 19, No. 8; 410-417 Oct. 1987.

Slater, Michael, & Wharton, John H. Revenge of the CISC. Byte, vol. 14, No. 12; 323-328 Nov. 1989.

Smith, Ben. A new workstation standard. Byte, vol. 16, No. 6; 52-54 June 1991.

Sith, Ben. Around the world in text display. Byte, vol. 15, No. 5; 262-268 May 1990.

Smith, Ben, & Mitchell, Rob. DEC's RISC power house. Byte, vol. 14, No. 12; 201-206 Nov. 1989.

Stewart, George A. Multiscan color monitors. Byte, vol. 13, No. 2; 100-115 Feb. 1988.

Tannas, L. E. Flat-panel displays and CRTs, New York; Van Nostrand Reinhold Company, 1985.

Thomson, Tom, Sun's newest workstation; the sun386i. Byte, vol. 13, No. 7; 103-106 Jul. 1988.

Unger, John. The sun386i. Byte, vol. 13, No. 13; 183-188 Dec. 1988.

Vecchiet, K. S. Computer graphics interface; a developer's perspective. CAD, vol. 19, No. 8; 415-455 Oct. 1987.

Velho, Luiz, et al. Digital halftoning with space filling curves. CG, vol. 25, No. 4; 81-90 July 1991.

Wang, J., et al. A real-time optical 3D tracker for head-mounted display systems. CG, vol. 24, No. 2; 205-215 Mar. 1990.

Wang Zhiyan. The design and implementation of a customized graphics system for instrument modeling CAD. Proceedings of the International Conference on Computer in Machinery Industry. Shanghai. 75-78 Sep. 1992.

Wang Zhiyan & Liu Danlin. A fidelity method of data exchange between CAD/CAM systems. Proceedings of the 1992 Harbin International Conference on Electronics and Computer. II -208~ II -211 Aug. 1992.

Ware, C. & Osborne, S. Exploration and virtual camera control in

virtual three dimensional environments. CG. vol. 24, No. 2; 175-183 Mar. 1990.

Weston, Charles D. & Stewart, George A. Workstations. Byte, vol. 12, No. 2; 85-97 Feb. 1987

Whang, Min-Hur & Kua, Joe. Jion the EISA evolution. Byte, vol. 15, No. 5; 241-247 May 1990.

Williams, Lamce. 3D paint. CG. vol. 24, No. 2; 225-233 Mar. 1990.

Wilton, Richard. PS/2 video programming. Byte, vol. 12, No. 12; 67-78

Wilton, Richard. VGA video modes. Byte, vol. 13, No. 11; 187-198 Nov. 1988.

Wu, Xiaolin. An efficient antialiasing technique. CG. vol. 25, No. 4; 143-152 July 1991.

Yager, Tom. DEC's latest RISC. Byte, vol. 15, No. 9; 154-156 Sep. 1990.

录

第一章概述

- 1 . 1 计算机图形图象
- 1 . 2 微型计算机图形显示回顾
- 1 . 3 图形显示面临的问题
- 1 . 4 图形处理器
- 1 . 5 图形软件标准

第二章图形显示装置

- 2 . 1 阴极射线管装置
- 2 . 2 液晶显示装置
- 2 . 3 气体等离子体显示装置
- 2 . 4 场致发光显示装置
- 2 . 5 显示装置展望

第三章图形控制器

- 3 . 1 单色显示和C G A
- 3 . 2 E G A和C G E 4 0 0
- 3 . 3 M C G A和V G A

第四章图形标准的进展

- 4 . 1 S u p e r V G A
- 4 . 2 T I G A
- 4 . 3 8 5 1 4 / A
- 4 . 4 X G A

第五章图形处理器

- 5 . 1 I n t e l 8 2 7 8 6 图形协处理器
- 5 . 2 T M S 3 4 0 1 0 图形系统处理器
- 5 . 3 T M S 3 4 0 2 0 图形系统处理器
- 5 . 4 视频R A M

第六章图形工作站

- 6 . 1 图形系统的发展
- 6 . 2 “ P C 兼容 ” 工作站
- 6 . 3 R 1 S C 工作站

第七章图形软件接口标准

- 7 . 1 图形程序设计标准
- 7 . 2 图形元文件
- 7 . 3 工作站和设备的接口
- 7 . 4 应用程序接口

第八章微型计算机视频编程汇集

- 8 . 1 视频B I O S 功能提要
- 8 . 2 标准I B M 视频模式及兼容显示
- 8 . 3 视频B I O S 功能说明

术语索引

参考书目及文献