

(共二册)



EGA VGA TVGA 高级微机 图形编程指南与实例

(上册)



北京希望电脑公司

(共二册)

EGA VGA TVGA 高级微机 图形编程指南与实例

(上册)

陶 伟	张 玄
马 强	陈 豪

北京希望电脑公司

一九九一年九月

前 言

本书是一本完善的 IBM PC 和 PS/2 系列计算机视屏图形标准的参考书。

图形卡允许用户绘制高分辩率图象，是一种非常重要的计算机外设。EGA 和 VGA 是 IBM PC 为 IBM PC 和 PS/2 开发的图形标准，随后又成为彩色图形的工业标准。

本书阐述了这些图形标准的特色和功能，以便读者学会 EGA 和 VGA 卡的编程技术。本书将教学和参考容为一体，即详细描述显示卡的结构，又介绍了编程技术，本书首先引导读者浏览一些 EGA 和 VGA 卡的特色，然后介绍计算机图形原理，和如何存放地使显示卡的多功能。此外，本书还介绍了一些 TVGA 的技术性能。

本书主要覆盖以下几个方面：

- EGA 和 VGA 寄存器
- 调色板控制
- 字符处理
- 画线和向量
- EGA 和 VGA BIOS
- TVGA 技术性能

本书最后一节给出一套用 C 和汇编语言编制的程序，这些程序涉及文本、字形、图形、适配器颜色段等，这些程序是用户未来开发软件的强大工具。

目 录

第一章 程序员指南简介	1
1.1 计算机图形卡	1
1.2 本书的需求	1
1.3 关于本书	2
1.4 本书的术语和特色	2
1.5 各章一般特色	3
1.6 每章的细节	3
1.7 编程举例	4
1.8 编译器和汇编器	5
第二章 EGA 和 VGA 的特点	6
2.1 EGA/VGA 特点	6
2.2 图形读写模式	10
2.3 EGA 和 VGA 分辨率	14
2.4 连接器和开关	17
2.5 大力神单色图形卡	21
2.6 彩色图形适配器	23
2.7 多色图形陈列	27
2.8 专业图形适配器	27
2.9 8514/A 显示适配器	27
第三章 图形硬件和软件	29
3.1 计算机图形硬件	29
3.2 输入设备	32
3.3 图形适配器	34
3.4 监视器和显示	43
3.5 信号采集	47
3.6 硬拷贝设备	49
3.7 与图形有关软件	53
第四章 PC 机, C 语言和汇编语言	57
4.1 PC 机硬件	57
4.2 C 语言	67
4.3 汇编语言	83

第五章 计算机图形原理	101
5.1 坐标系统	101
5.2 坐标转换	104
5.3 字符	108
5.4 点	112
5.5 线	122
5.6 圆	131
5.7 映像	136
5.8 色调理论	138
第六章 字符处理	141
6.1 EGA 和 VGA 字符处理	141
6.1 字符形状	143
6.3 字符属性	149
6.4 视屏区	152
6.5 字符模式	161
6.6 光标	168
6.7 可下装的字符集	170
第七章 图形处理	179
7.1 字符	179
7.2 图形属性	182
7.3 视屏区	185
7.4 图形显示模式	187
7.5 向视屏区写	205
7.6 从视屏区读	216
7.7 显示时间	220
第八章 调色板和颜色寄存器	226
8.1 调色板	226
8.2 将数据转化为颜色	231
8.3 颜色寄存器	237
第九章 读 EGA 和 VGA 状态	251
9.1 读适配器状态	251
9.2 读适配器显示寄存器	251
9.3 从内存中读 BIOS 表	255
9.4 读 EGA 和 VGA 适配器状态	256
9.5 硬件测试	269

第十章 EGA 和 VGA 寄存器	275
10.1 EGA/VGA 寄存器	275
10.2 通用和外部寄存器	277
10.3 时序器寄存器	282
10.4 CRT 控制器寄存器	289
10.5 图形控制器寄存器	310
10.6 属性控制器寄存器	323
10.7 颜色寄存器	334
第十一章 EGA 和 VGA 的 BIOS	337
11.1 EGA/VGA BIOS 描述	337
11.2 显示模式	337
11.3 光标控制	340
11.4 光笔定位	343
11.5 显示页选择	343
11.6 上卷屏幕	344
11.7 读/写字符	346
11.8 调色板	350
11.9 读/写一个像素	351
11.10 调色板寄存器	354
11.11 颜色寄存器	360
11.12 字符产生	365
11.13 交替选择	378
11.14 写一字符串	386
11.15 返回 VGA 状态	387
第十二章 TVGA 简介	390
12.1 TVGA 技术性能简介	390
12.2 TVGA 一般技术信息	391
12.3 TVGA 的组织结构	398
12.4 操作原理	434
12.5 模式表	443
12.6 寄存器表	445
12.7 存贮器映射图	451
12.8 TVGA 扩充模式下的存贮组织	481
12.9 扩充模式下的库切换	482
12.10 把 TVGA 设置成 EGA,CGA 和 MDA 的方法	489
12.11 TVGA 的 BIOS 调用	492

第十三章 编程举例	504
13.1 程序列表	504
13.2 介绍	508
13.3 命名协定	509
13.4 宏扩展	509
13.5 控制和读显示类型	511
13.6 判断 VGA 的状态	513
13.7 字符	516
13.8 光标控制	524
13.9 卷页	525
13.10 画一个点	526
13.11 画一条线	527
13.12 画一个圆	529
13.13 画一个图象	552
13.14 数据编码	561
13.15 修剪	564
13.16 清除显示内存	567
13.17 下装字符集表	574
13.18 调色板	590
13.19 颜色寄存器	596
13.20 专用特性	599
13.21 写 EGA/VGA 寄存器	601

第一章 内容概要

1.1 计算机图形卡

随着计算机图形卡的发展，人们在个人计算机上编制图形软件成为可能。无论你编制什么样的图形软件，功能强大的图形卡会帮助你在显示屏上创造出精美的图象。

一个有效的图形系统只有和各种图形软件，图形外设，计算机存储器，DOS 操作系统相配合才能发挥作用。个人计算机利用图形卡在显示屏上绘制的点，直线，曲线，字符来显示内部信息。单显卡，诸如大力神卡（Hercules Card），是通过黑白两种颜色协调来显示；彩色图形卡将数据以多种颜色显示出来。这些图形卡的编程艺术被人们称为计算机图形学（computer graphics）。

本书将帮助您掌握两种较流行的彩色图形卡的编程技术，即 EGA 卡（Enhanced Graphics Adaptor Card）和 VGA 卡（Video Graphics Array Card）。这两种图形卡都遵循良好的图形标准。这些标准有助于使基于其上的软件有兼容的硬件基础。这意味着您编制的软件可以在成千上万台配制 EGA 和 VGA 的计算机上运行，享受较长的软件生命周期。您现在学习的关于这些图形卡的编程技术，在较长时间内不会过时，仍可以使用。当新的图形卡发表之后，您只需了解一下其在 EGA/VGA 标准上所扩展的功能就行了。

IBM 公司为基于 PC（Personal Computer）和 PS/2（Personal System/2）系列计算机开发了 EGA 和 VGA 图形标准。这是因为早期发表的 CGA（Color Graphics Adaptor）分辨率太低，颜色太少。1984 年开发的 EGA 标准及其适配器板解决了上述问题。从此，EGA 变成了 PC 上广为流行的彩色图形标准。制造商们已生产出多种 EGA 产品，几乎所有的应用软件都得益于这种强大的图形标准。

IBM 公司于 1987 年发表了 VGA。它安装在 PS/2 的 50，60，80 型的母板上。该图形卡标准向下兼容 EGA，但增加了许多新的功能。其中最重要的是适配器寄存器的可读性，它既提高了分辨率，又获得了多至 256 种颜色的模式。

伴随着 PS/2 系列出现的第二种适配器是 MCGA（Multi-Color Graphics Array）。MCGA 集成在 PS/2 30 型计算机中，向下兼容 CGA（Color Graphics Adaptor）。这种兼容不仅是 BIOS 级，而且在寄存器，视屏区层次上也是兼容的。这种适配器不如 EGA 和 VGA 应用的广泛。因此，本书主要介绍 EGA 和 VGA 卡。

1.2 本书对读者的要求

EGA 或 VGA 卡的操作细节只能从技术手册上获得。但是这些手册上包含着许多甚至对一些有经验的图形程序员也用不着的东西，从中找一点有用的东西是一件令人头痛的工作。此外，任何一本参考手册都不会阐述一位程序员如何使用图形卡，如何在应用中恰如其份地使用一些相关技术。至于一些图形编程的书也只是有选择地提供一些图形子程序，而缺乏编程的必要细节。

随着图形系统的功能和灵活性的增长，图形编程的需求越来越大，工作也越来越复杂。一个程序员无论在使用已存在的图形库还是直接对图形卡编程都有必要了解图形系统的功

能。为了充分利用图形系统，程序员必须详细掌握图形算法，图形系统以及如何将图形系统集成到计算机环境中。只有掌握了图形卡和计算机图形的知识，程序员才能创造出各种各样具有精美界面的图形应用程序。

1.3 关于本书

本书向读者提供有关 EGA 和 VGA 编程的全面的、综合性的指南。本书的宗旨是帮助每一位读者掌握有关图形卡的知识 and 编程技巧。书中的每一章都是独立的，并按一般的学习过程安排内容，这种将传授和参考手册融为一体的特色给读者留下了充分的灵活性。读者可以随便浏览，研究，深入。

本书向读者阐述的一种重要原则是优化程序。EGA/VGA 有许多特色，它们隐藏于控制寄存器和 BIOS 调用之中。在 EGA/VGA 上一种功能的实现可以有多种方法，掌握这些适配器的特点的程序员将能作出最好的选择。

EGA 和 VGA 中有一些特色很难掌握。由于不了解这些特色的底层支持功能，程序员会发现这些特色晦涩难懂，于是就不去使用它。这本书将通过介绍相关图形设备，应用软件和基本原理，向读者展示这些秘密。

自从引入这些图形标准的，许多制造商发表了与 EGA/VGA 兼容的图形卡，其中许多都扩展了 EGA/VGA 的功能。为了简练，这里只讨论 IBM 标准和一些重要的例外。

1.4 本书的术语和风格

VGA 实际上是 EGA 的一个增强版本，尽管它们之间存在着某些差别。为了简练，当我们仅叙述与 VGA 或 EGA 有关的信息时，只用“VGA”或“EGA”，若二者兼有，则用“EGA/VGA”。

数值以十进制或十六进制表示。若是十六进制则后面跟有一个标识：hex，对于缺省的则为十进制。例如：“10”表示数值 10，“10hex”表示数值 16。

EGA/VGA 寄存器在第十章中介绍。每一个区域均在寄存器的图解和描述中指明，包括每一个区域影响哪一种适配器。寄存器图解包括一个代码名，该代码名表明每一个区域所指的位。如果这种代码名前面有一个小方块，则指的是 EGA。如果后面跟一个小方块，则指的是 VGA。

有些示意图中，不同的区域对不同的适配器意义是不同的。这些区域都有用斜线隔开的两个代码。前一个指 EGA，后一个指 VGA。如果代码前后均没有小方块则指 EGA 和 VGA。例如图 1.1 所示的一种寄存器。

十三章中，所有样例程序均遵循 Microsoft C 和 Microsoft 汇编语言规则。虽然本书所提供的程序代码很容易转换成“中模式”编译，但我们对书中所有代码仍使用“大模式”编译。

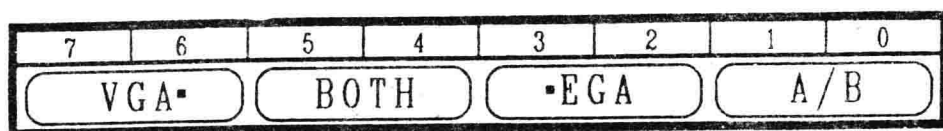


图 1.1 典型的寄存器图示

1.5 本书总体布局

第二章至第五章帮助读者掌握以下内容：EGA/VGA 的特点，EGA/VGA 适配器与其它适配器的关系，PC 机，图形设备，图形软件。为了让初学者加强理解，我们介绍用 C 语言和汇编语言编制的例子。例子中的图形算法在图形指导章节介绍。

第六章至十三章详细介绍 EGA/VGA 以及 TVGA 卡的功能。这些章节将向读者提供有关显示模式，显示存储器，图形处理器，下装字形库，彩色处理，控制寄存器和 BIOS 调用的精确的、综合性的信息。

所有这些信息均揉合在十三章中的例子中，十三章提供的 70 个样例程序将向读者展示 EGA/VGA 卡的多种强大的功能。

1.6 各章详细内容

第二章向读者展示了 EGA/VGA 的特色，以便读者了解这些适配器的能力，同时也探讨了与显示分辨率及颜色分辨率有关的问题，并回顾了其它图形适配器的特色。

第三章介绍了图形硬件和软件，以及它们对 EGA/VGA 程序员的影响。硬件设备包括人机对话设备，图形适配器，监视器，数据采集，硬拷贝设备。这一章还讨论了基于 EGA/VGA 的流行的图形应用软件对 EGA/VGA 的要求。

第四章分析了 EGA/VGA 和 PC 硬件软件的关系，引入了与 EGA/VGA 性能息息相关的各方面内容，这包括 PC 内存，输入/输出系统，内部 CPU 寄存器，BIOS 调用。此外还向读者介绍了一些 C 语言和汇编语言的编程技巧。这些技巧在十三章的例子中都能用得着。

第五章解释了十三章中的图形软件例子所用的原理，描述了坐标系统和转换，字符，点，直线，形状。本章重点放在画直线，圆，及剪裁算法。

第六章向读者提供有关 EGA/VGA 的文本状态下的编程工具，建立了显示适配器的字符处理能力，描述了字符形状大小，属性。每一种字符显示模式均有解释和图解。提供了各种字符模式的显示存储器使用情况。还讨论了下装字符集，包括字形的组织，读写以及选择字形的技术。

第七章向读者提供在 EGA/VGA 上进行图形编程的必要工具。这里也要讨论字符，字符属性和字形。本章给出了每一种图形显示模式及每一种显示模式的显示存储器组织，以及内部图形处理器的操作细节，这样便可帮助用户深入硬件内部。本章还将探讨如何提高图形操作的速度。最后简介了视屏分时，这对程序员来说是一个有用的概念。

第八章介绍 EGA/VGA 的颜色处理功能。详细介绍了调色板寄存器和颜色寄存器,以及从这些寄存器读写数据的技术。与从一种颜色体系转化到另一种一样,如何将数据转化为颜色的技术也提供给了读者。

第九章重点放在检查图形适配器的状态。有三种方法可以获取这个信息:直接读寄存器, BIOS 调用, RAM 中的 BIOS 数据区。本章以表格形式提供了 EGA/VGA 寄存器的缺省值。这些值在许多情况下对程序员很有用。例如,要了解显示模式的操作或不同显示寄存器的相互作用等。

第十章介绍了 EGA/VGA 卡上的所有寄存器。所有与控制适配器有关的内容均寄存在这些寄存器中。许多寄存器又分为若干区域。一个寄存器可以控制若干个图形功能。寄存器的每个区域在该章中都得到详细介绍。

第十一章介绍了 EGA/VGA 卡上全部的 BIOS 调用。每一种调用的输入和输出均提供给读者,包括每个调用对显示, 宿主机寄存器, EGA/VGA 寄存器以及 BIOS 数据区的影响。

第十二章详细介绍了 TVGA 的组织结构、技术参数以及 TVGA 的 BIOS 调用,并举了一些实例。

第十三章提供了 70 个工作在 EGA/VGA 卡上的样例程序。这些例子涉及文本, 字形, 图形, 适配器颜色选择。每一个程序均有良好的注释, 通俗易懂, 这些程序构成了开发程序的工具箱。

1.7 编程例题

十三章中的编程例题覆盖图形编程的多个方面。这包括 EGA/VGA 控制, 读适配器状态, 字符处理, 控制图形处理, 颜色控制, 下装字符字库, 以及特殊使用效果。这些例程构成图形工具箱的基本内容。

这些编程例题均用 C 语言和汇编语言写成。之所以选择 C 语言, 在于 C 的广泛应用, 结构化和高速。C 语言拥有许多优秀的编译器和大量的支持工具。本书的 C 语言例程可以被转化为其它高级语言。

在许多例子中, 程序是用汇编语言写的。每一个汇编语言例程都非常短小而且有良好的注释。甚至一名汇编语言新手掌握这些程序也没有多大问题。

我们在应用本书的例程时需要考虑一些问题, 即哪些例程应采用, 例程应该到什么样的复杂程度, 每一例程怎样为更特殊的应用服务, 例程中哪些适配器状态可以去掉等。

本书选择的例程覆盖了 EGA/VGA 的全部领域。这些例程在许多方面对程序员都很有用处, 例如: 图形设置, 字符串处理, 绘制点, 直线, 弧线, 剪裁, 卷动和漫游, 窗口, 下装字符集, 控制颜色等。这些程序展示了本书介绍的一些原理。

本书选择的例程力图精简, 没有必要将完成复杂图形控制的长篇累牍的程序提供给读者。书中每个例程完成一项特殊功能。程序员可以将这些例程有机地组合起来去完成更复杂的任务。然而画线例程出于优化的考虑, 显得有些复杂。但是画线在图形环境中非常重要, 应该选最高速度的例程。此外, 这种优化为程序员提供好几种技巧去开发高速图形。

每一例程的复杂程度取决于应用。通用的例程往往复杂, 低速。而专用的程序又会导致程序员不得不去开发一大堆类似的例程, 从而增加了软件的复杂性。

在十三章中的一些例程有广泛的应用, 例如, 窗口例程可以在任何一种字符或图形模式

下工作。另一方面，其它的一些例程也确实比较特殊。例如，画线例程调用了五个特殊的例程去画一条线。

高效的编程需要在我们采用这些例程时考虑与适配器条件有关的问题。例如，一个特定例程需要设置多个寄存器。如果不将缺省值重置给这些寄存器的话，那么任何对寄存器的修改将导致适配器于不可预知的状态。在例程开始前设置寄存器并在结束时恢复原值是很费时的。如果一个例程被反复调用，但其间并无其他例程使用过这些寄存器，那自然浪费了很多时间。因此，好的办法是在此例程调用前设置寄存器，然后反复调用该例程，最后再恢复寄存器原值。不过在处理完成之前，适配器仍然是不可预测状态。程序员必须认识到这种情况，在寄存器恢复到原来缺省值状态前，不要中断所进行的处理。这里有个折衷办法，由于性能的高低对应着编程复杂性的多少，所以每一个应用对应着解决问题的最好方法。

1.8 编译器和汇编器

本书例程适用于 Microsoft C 4.0 或以上的编译器，汇编器也是 4.0 或 4.0 以上。连接器为 3.0 或 3.0 以上。C 语言例程可以很容易地利用其它 C 编译器编译。汇编语言程序显然没那么简单。其它汇编器的用户也许要修改汇编语言代码。

所有的编程例子遵循 Microsoft C 或 Microsoft 汇编语言规则，编译时选用大模式。汇编器指令及参数地址格式在选择其它模式时需要改变。

第二章 EGA 和 VGA 的特点

2.1 EGA/VGA 的特点

EGA 和 VGA 是通过超大规模集成电路实现的。一些 PC 兼容图形卡使用了 EGA/VGA 集成电路。这些卡有很多优点, 诸如成本低, 高速, 高密度存贮, 轻便, 紧凑, 容易维护等。硬件制造商已经在商品化的图形卡中揉合了 EGA/VGA 标准。此外, 许多计算机, 包括 IBM 的 PS/2 系列, 都将 VGA 合成到 PC 的母板上。

EGA/VGA 标准是这些图形卡成功的唯一的重要因素。这种标准允许软件开发商开发满足多种用户的软件。EGA 和 VGA 卡有它们自己的 BIOS。这种 BIOS 对兼容性起着重要作用。BIOS 中的例程消除了硬件实现的差别。几乎所有的标准 PC 的应用软件都和 EGA/VGA 兼容; 包括文字处理器, 数据库管理程序, 电子表格, 实用程序。图形软件包括商业图形, 计算机辅助设计, 桌面印刷系统, 图表软件都采用 EGA 彩色标准。

我们应该认识到虽然 EGA/VGA 标准有明确的定义, 但该标准并没有扮演更多的角色。许多 EGA/VGA 兼容卡在标准的基础之上又提高了分辨率, 提供了更多的颜色选择, 更大的显存, 特殊字符库, 132 列宽屏幕, 算术处理器, 图形硬件缩放功能等。这些增加的功能不遵守 IBM 或任何其它标准, 需用专门的驱动器来将这些功能合并到上述软件应用中去。对于一个使用多种软件包的人来说, 这简直是一件麻烦事, 因为并不是每一个卡都有一个各种软件包的驱动器。此外, 当有一个软件包的驱动程序驻留内存时, 另外一个软件包驱动程序就有可能不能驻留。

EGA/VGA 拥有丰富的、灵活的图形功能, 为程序提供了强大的图形工具。EGA/VGA 可以被配制为多种字符和图形模式中的一种, 并且能驱动单色监视器, 低分辨率和高分辨率数字彩色监视器, 一些 EGA 以及所有的 VGA 都可以驱动模拟监视器。

本章将先介绍 MDA/Hercules 和 CGA 图形适配器, 然后再介绍 EGA/VGA。许多 EGA/VGA 的功能与前二个图形适配器兼容。最后还要简要介绍 MCGA, PGA (Professional Graphics Adaptor) 和 8514/A 图形适配器。

2.1.1 读模式和写模式

各种读模式和写模式是在显示处理中实现的。这些模式增加了宿主机功能, 它们允许许多操作, 诸如在 EGA/VGA 图形处理器中显示读写一个长为 32 位的字这对应于 8 位操作其处理速度提高了四倍。其它写模式允许在不用宿主机处理计算的条件下, 完成逻辑操作。一种读模式提供颜色比较, 仅用一条 PC 指令便可实现与周围 8 个像素点比较一个四位颜色 (4bit)。一旦 EGA 的许多隐藏的、重要的特点被掌握, 图形程序员会发现自己进入了一个新的境地。

2.1.2 显存

程序员可以读写显存, 没有必要等待水平或垂直扫描回扫周期。与其它显示适配器相

比,这大大加速了图形处理。

显存是以一种简单的方式组织成的。一旦掌握了确切地址,程序员可以轻而易举地将线条和图形直接画在屏幕上,而不依靠 BIOS 调用。

显存直接映射到 PC 的内存地址空间上,这样使程序员可以使用汇编语言中的 `mov e` 或 `mov e string` 指令访问显存,实现宿主机内存和显存之间快速数据传递。然而,多种等待状态是共用的。

显存映射到宿主机内存中的为图形存储器保留的地址空间上。EGA 配制为单显模式时可与 CGA 共存,相似地,EGA 配制在彩色模式时可以和 MDA 共存。

显存通过一个 8 位数据总线和宿主机处理器连接而不管宿主机数据总线的宽度。

显存可由多至 256K 字节的存储器组成。这样可以将显示模式组成各种分辨率,如将显存分别配制为几个低分辨率页或一个高分辨率页。

当前显示模式决定显存是以压缩格式组织还是按位面格式组织。

EGA/VGA 提供 32 位数据总线。宿主机可以用一条指令完成从 32 位宿主机内存到显存的数据传输。然而它不是唯一的。数据传输器也可以用一条指令将 32 位数据在显存内部传送。这将是 8 位传输速度的 4 倍。

表 2.1 字符属性

<i>Mode</i>	<i>Attribute</i>
Mono	Normal, Intensified, Reversed, Blinking, Underlined
Color	16 foreground, 16 background colors
Color	16 foreground, 8 background, blinking
Color	8 foreground, 8 background, 512 characters, blinking

2.1.3 字符模式

显存以一种与 MDA 及 CGA 兼容的方式组织,这提供了完全的向下字符模式兼容能力。

每一个字符在显存中用两个字节表示,ASCII 码字符表示显示的字符是什么,另一个属性字节决定字符的颜色。因为一个字符在显存中只占用两个字节,所以它的屏幕传递速度非常快。

在宿主机内存中,字符和属性字节存储在一个连续的内存空间,EGA/VGA 中一个特定地址模式可以使字符和属性在显存地址空间中按顺序排列。这样程序可以使用串移动指令实现快速字符数据传输。

字符可以用单色或彩色属性显示。典型属性列于表 2.1。

显存的大小决定可存入的正文页数,显存最多一次可以装入 8 页字符正文。

字符模式中可有多种分辨率模式。缺省模式和字符大小列 2.2 表中。

在标准显示模式中可以通过使用 8×8 字符集定义新的字符分辨率。一些基于彩色显示模式的例子列在表 2.3 中。

字符集驻留在显存之中,EGA 中同时最多可以驻留四套字符集,VGA 则是 8 套。

表 2.2 缺省字符模式分辨率

<i>Mode</i>	<i>Adapter</i>	<i>Resolution (Characters)</i>	<i>Resolution (Pixels)</i>	<i>Characters (Pixels)</i>
Color	EGA/VGA	40 by 25	320 by 200	8 by 8
Color	EGA/VGA	40 by 25	320 by 350	8 by 14
Color	VGA	40 by 25	360 by 400	9 by 16
Color	EGA/VGA	80 by 25	640 by 200	8 by 8
Color	EGA/VGA	80 by 25	640 by 350	8 by 14
Color	VGA	80 by 25	720 by 400	9 by 16
Mono	EGA/VGA	80 by 25	720 by 350	9 by 14
Mono	VGA	80 by 25	720 by 400	9 by 16

表 2.3 非标准字符模式分辨率

<i>Mode</i>	<i>Adapter</i>	<i>Resolution (Characters)</i>	<i>Resolution (Pixels)</i>	<i>Characters (Pixels)</i>
Color	EGA/VGA	80 by 25	640 by 200	8 by 8
Color	EGA/VGA	80 by 43	640 by 350	8 by 8
Color	VGA	80 by 50	640 by 400	8 by 8

2.1.4 图形模式

显存组织方式使用几种模式与 CGA 显存兼容，这使 CGA 图形具有向下的兼容性。

单显模式使用两位或一位表示一个像素。当程序在仿真大力神卡时，一个像素只需两位就可以表达单显的各种属性。其它模式叫双色模式，也可以被当成单显模式，因为它们可以提供两种颜色。这两种颜色没有必要是仅黑白二色。

彩色模式使用每像素 2 位，4 位，8 位。“每像素 2 位”模式仿真 CGA，使一幅图象同时可以有最多四种颜色。“每像素 4 位”模式允许同时有 16 种颜色，“每像素 8 位”模式允许同时有 256 种颜色。

显存有多种配制，这取决于当前显示模式。标准分辨率列在表 2.4 中。

字符集驻留在宿主机内存空间，或是在 BIOS ROM 中，或是在 RAM 中。

表 2.4 标准图形分辨率

<i>Mode</i>	<i>Adapter</i>	<i>Resolution (Pixels)</i>	<i>Number of Colors</i>	<i>Typical Characters (Pixels)</i>
Mono	EGA/VGA	640 by 200	2	8 by 8
Mono	EGA/VGA	640 by 350	4	8 by 14
Mono	VGA	640 by 480	2	8 by 16
Color	VGA	320 by 200	256	8 by 8
Color	EGA/VGA	640 by 200	16	8 by 8
Color	EGA/VGA	640 by 350	16	8 by 8
Color	VGA	640 by 480	16	8 by 16
Color	EGA/VGA	320 by 200	16	8 by 8

2.1.5 下装字符集

每一种显示模式都有一套缺省字符集，缺省字符集驻留在 BIOS ROM 中。

EGA/VGA 能够使用用户定义的可下装字符集，任何一种格式的字符集都能被使用。字符集最多可以达 256 个字符。每一个字符最多 9 个像素宽，32 条扫描线高。

EGA BIOS ROM 含有三套字符集，大小分别为 8×8，8×14，9×14，其中 9×14 字符集是附加的，它仅包含那些与 8×14 字符集形状不同的字符。

VGA BIOS ROM 含有五套字符集，大小分别为 8×8，8×14，9×14，8×16，9×16，其中 9×14 和 9×16 字符集是附加的。它们仅包含那些与 8×14 及 8×16 字符集形状不同的字符。

EGA 显存中最多可同时驻留 4 套字符集，每套 256 个字符长。其中任何两套可被激活，向程序员同时提供 512 个字符。

VGA 显存中最多可同时驻留 8 套字符集，每套 256 个字符长。其中任何两套可以被激活，向程序员同时提供 512 个字符。

2.1.6 控制寄存器

EGA/VGA 最复杂的方面是掌握和了解控制寄存器。这些寄存器被分成组（见表 2.5）。

每一个寄存器为一个字节宽。有几个寄存器又再分成若干位段（fields）。寄存器中每一个位通常控制完全不同的功能。有些功能要求其控制位段不止出现在一个寄存器上。

这些寄存器通过 I/O 端口地址空间访问。寄存器子集的 I/O 地址变化取决于配制。如果 EGA/VGA 在单显模式，这个子集映象为 MDA 地址。与此相类似，EGA/VGA 在彩色模式下，这个子集映象为 CGA 的地址。

在表 2.5 列出的寄存器组分别在宿主机 I/O 端口地址空间用两个寄存器来表示。EGA/VGA 寄存器通过一个索引或一个地址寄存器和一个数据寄存器由地址分配表索引。

在 EGA 中，绝大多数的寄存器是只写的。这限制了程序访问显示适配器的能力。在 VGA 中，除了属性控制触发器以外，均可以被宿主机读。

表 2.5 EGA/VGA 寄存器组

<i>Number of Registers</i>	<i>Group Name</i>
4	General registers
5	Sequencer registers
28	CRT controller registers
9	Graphics controller registers
22	Attribute controller
5	Color registers

2.1.7 显示模式

EGA/VGA 可以在一系列的标准显示格式中选择配制。这些格式决定显存的配制，显示分辨率，每个象素的位数，缺省字符字形，及显存的首地址。

所有的显示模式分成两大类：字符文本模式和图形模式。图形模式又称为所有点可编址模式（all-point-addressible）。两种类型的模式之间存在着重大差别。

程序员可以通过直接控制寄存器来设计一种新的模式，但一般不鼓励这么做。因为改变一种模式的操作过程非常复杂，一个错误的操作有可能摧毁监视器和显示适配器。用 BIOS 调用来改变显示模式既简单又有效。

文本显示模式实际上是在仿真 MDA 单显系统和 CGA 彩色系统的操作。

2.1.8 EGA/VGA BIOS

EGA/VGA 的 BIOS 包含在 ROM 中，提供有关 EGA/VGA 的所有操作。有许多文本和图形功能可以通过 BIOS 调用来完成。

BIOS 调用一般通过汇编语言调用来完成，在一些情况下，也可以直接用 C 语言调用。

BIOS 调用可以分为若干类，这些类列在表 2.6 中。

对显存的直接操作是 EGA/VGA 最重要的一个方面，这一点却完全被人们所忽视。BIOS 几乎不能向显存写数据或从显存读数据。BIOS 调用仅限于读写一个象素。

2.2 图形读写模式

程序员可以使用四种写模式。这有助于加速写操作的速度。此外，还可以使用两种读模式。一种是直接从显存读数据，另外一种读比较的结果。

2.2.1 写模式 0

程序员可以通过 8 位数据总线直接向显存写。