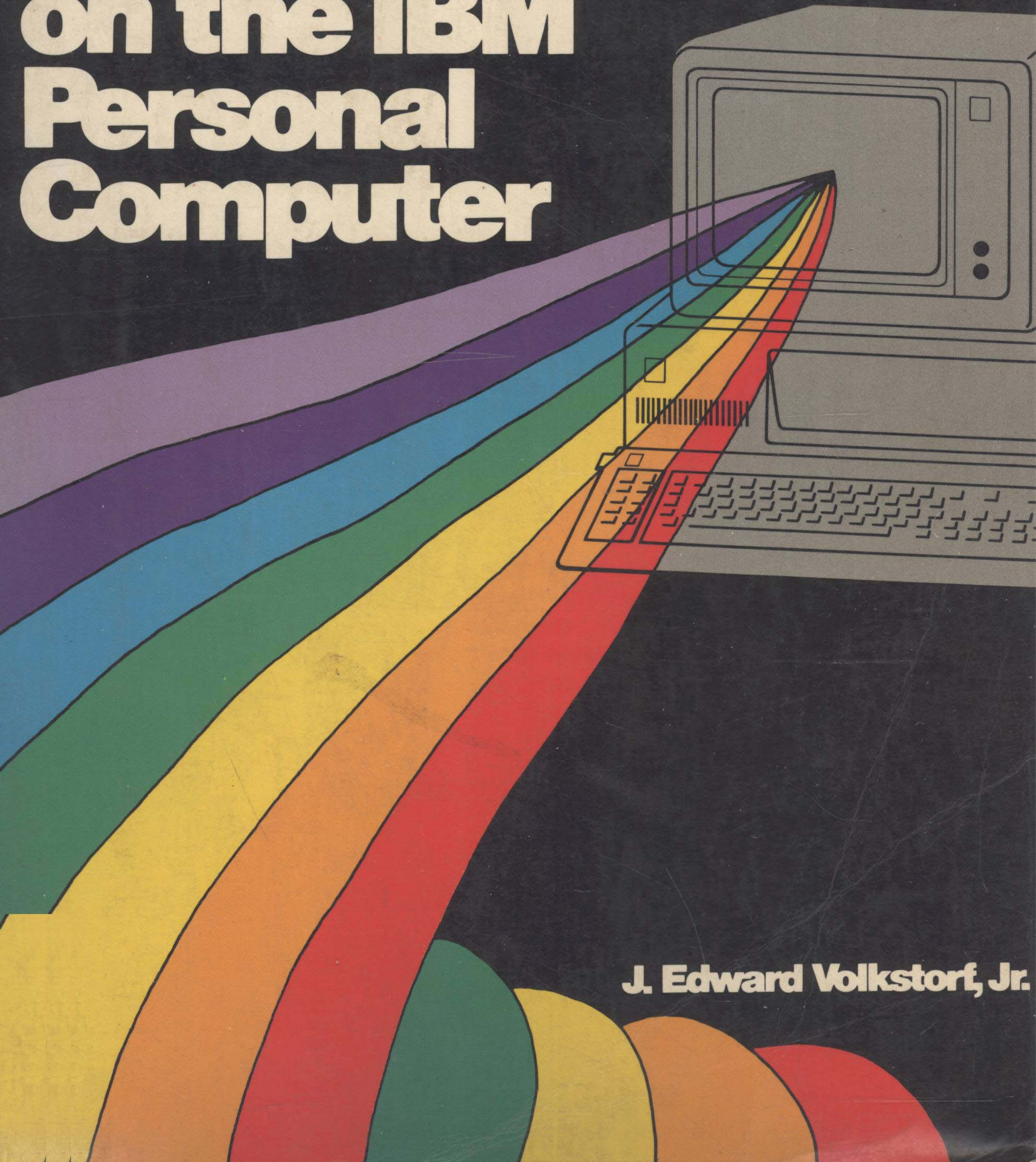
A Spectrum Book

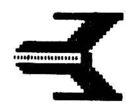
Graphics Programming on the IBM Personal Computer

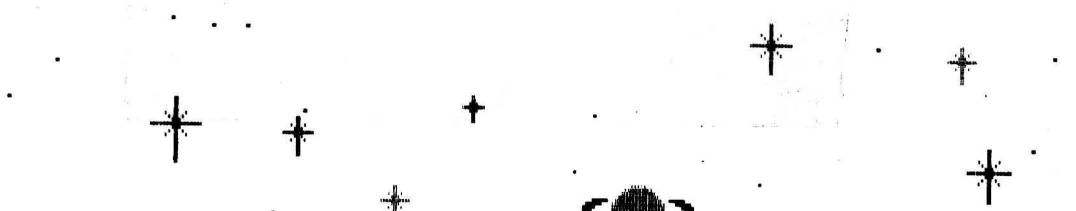


J. Edward Volkstorf, Jr.

J. Edward Volkstorf, Jr.

GRAPHICS PROGRAMMING ON THE IBM PERSONAL COMPUTER





A SPECTRUM BOOK

Prentice-Hall, Inc., Englewood Cliffs, New Jersey 07632

Library of Congress Cataloging in Publication Data.

Volkstorf, J. Edward.
Graphics programming on the IBM personal computer.

"A Spectrum Book."
Includes bibliographical references and indexes.

1. Computer graphics. 2. IBM Personal Computer—Programming. 3. Basic (Computer program language) I. Title. II. Title: Graphics programming on the I.B.M. personal computer.

T385.V65 1983 001.64'43 83-19069

ISBN 0-13-363201-6
ISBN 0-13-363193-1 (pbk.)
ISBN 0-13-363219-9 (book/software)

A Spectrum Book. Printed in the United States of America.

10 9 8 7 6 5 4 3 2 1

ISBN 0-13-363201-6

ISBN 0-13-363193-1 {PBK.}

ISBN 0-13-363219-9 {PBK./SOFTWARE}

Cover design © 1983 by Jeannette Jacobs
Editorial/production supervision by Peter Jordan
Manufacturing buyers: Joyce Levatino and Christine Johnston

This book is available at a special discount when ordered in bulk quantities. Contact Prentice-Hall, Inc., General Publishing Division, Special Sales, Englewood Cliffs, N.J. 07632.

Prentice-Hall International, Inc., *London*
Prentice-Hall of Australia Pty. Limited, *Sydney*
Prentice-Hall Canada Inc., *Toronto*
Prentice-Hall of India Private Limited, *New Delhi*
Prentice-Hall of Japan, Inc., *Tokyo*
Prentice-Hall of Southeast Asia Pte. Ltd., *Singapore*
Whitehall Books Limited, *Wellington, New Zealand*
Editora Prentice-Hall do Brasil Ltda., *Rio de Janeiro*

Preface

Welcome to the world of computer graphics! Its an exciting place full of action, color, and beautiful imagery. As an IBM Personal Computer (PC) owner, you are probably well aware of your computer's potential for graphics. If you are using some graphics software now on your PC, you are most likely interested in learning more about how it works. This book is a step in that direction, for it explains many of the techniques that can turn your PC into a graphics machine.

The book provides a thorough coverage of graphics using the PC's BASIC language. It describes all of the BASIC statements for graphics in a logical and progressive sequence. Along the way many statement samples, short BASIC code segments, and full-length examples are included. After all of the graphics statements (and then some) are examined, they are combined and used in some popular and useful applications.

Take a look at the program example index. These are the 63 "full-length" examples that go beyond the short coding segments sprinkled throughout the text. They include computer-art displays, business graphics, manipulation of the character set, cartography, educational exercises, animation examples, and some graphics-oriented fun and games.

Within the BASIC source code of each of these programs are numerous REMarks and comments; these are provided as an aid and guide when you decide to make changes to the programs for your own special needs. With a little patience, and notes plus the descriptions of them in the book, you should be able to turn many of them into some very sophisticated graphics applications.

The programs require a PC with a minimum of 48K and one disk drive. Systems with a color/graphics adapter can run all programs. Systems with only the monochrome adapter and display can use and run about half of them. Wherever a character mode graphics example is given, it will be compatible with both types of adapters. The programs that require the color adapter have checks in them to determine if an adapter is attached, so feel free to run all of the examples.

In the event that your PC has both adapters attached, just use the "adapterswitch" program code illustrated in Chapter 4. It will turn on your system for the color/graphics adapter when you want to run those programs that require it.

So much for what the book *is*. It is *not* an introduction to BASIC programming on the PC. It is assumed that the use of IF/THEN, FOR/NEXT, GOSUB/RETURN, GOTO, and PRINT statements are reasonably well understood by the reader. Also, the examples are just that—examples. There has been no attempt to optimize the code within them with rigorous and well-refined logic. Each example provides a complete framework for illustrating one or more concepts and also serves as a starting point for future developments.

A suggested approach for reading the book is to study a section on software in its entirety before testing the code or examples. Once a section has been read and you're reasonably comfortable with the concepts introduced, return to the examples and try them out. Experiment with them on your PC as you review the text's explanation of how they work. Make changes and write notes on the effects. Many unusual graphics images can be produced quite by accident.

Contents

INTRODUCTION

Continuing improvements in technology have caused one more bastion to fall: Computer graphics are no longer solely in the domain of the large computer with expensive interfaces and display devices. Graphics are now found in virtually all areas and levels of computing. Many of the newer personal computers have sophisticated color graphics as a standard feature. Even the tried-and-true black and white CRT used throughout the data processing community is increasingly being replaced by multicolor terminals, many with graphic display capabilities.

Graphics applications in such areas as business reporting, medicine, design engineering, art, architecture, education, gaming, and training simulations are commonplace. Cartooning is beginning to receive assistance from experimental computer-aided animation processes. Other medias are now routinely being enhanced with lifelike computer-generated images and action sequences, as in television commercials, news, weather and sports reports, and major motion pictures.

At the leading edge of graphics technology are new developments that will continue and accelerate the growth of graphics to even more spectacular frontiers. Already display devices are being tested that create true three-dimensional images. Video laser and etched disks allow the maintenance of enormous amounts of data and enable the merging of video images with computer-generated graphics.

In the future a data base inquiry will be responded to not only with some lines of printed text but also a detailed picture. Imagine a parts inventory file that displays a paragraph describing the item, along with a drawing of it for detailed examination. Looking back at today's character-text-only terminals will be like comparing the cars of today with Ford's Model T.

On the home front, most personal computers provide some form of graphics, either as part of their basic equipment or through add-on options. Noteworthy of these systems is IBM's Personal Computer, or the PC. Announced in August of 1981, the PC virtually overnight became a top seller in personal computers. Of course the IBM name alone causes immediate acceptance of the system, but the PC has become a potent force in its own right for many other reasons.

The PC's overall design includes as standard many features not typically found on similar computers:

1. The CPU is a 16-bit microprocessor, the INTEL 8088.
2. It has a memory-addressing capability of over 1 million bytes.
3. A high-level BASIC interpreter and sophisticated screen editor are included in the system's ROM.

4. Its set of 256 characters includes business graphics, gaming, mathematical, and many other types of symbols.
5. Alternate processing modes allow the system to function in three levels of point-addressable graphics.
6. The system is attractively styled and packaged as a set of solid and highly integratable components.
7. System user guides and reference manuals are very comprehensive and thorough in their coverage of material.
8. Contrary to IBM's usual "closed-box" policies regarding information about it's equipment, technical specifications about the internals of the PC are well documented, readily available, and open to those interested.

The almost immediate appearance of the many add-on equipment vendors has been fostered by IBM's openness for the product. Within a year of the introduction of the PC there were hundreds of suppliers of PC-compatible memory, disks, I/O adapters, and other hardware products.

The PC's documentation is setting new standards for personal computer systems. The BASIC reference manual is an excellent, comprehensive, and well-written guide for the novice as well as the experienced BASIC programmer. PC BASIC is a version of Microsoft Inc. BASIC and is fairly standard as far as BASICs go. For the user who desires more introductory or advanced BASIC programming information, many other books and aids are available.

But learning more about the PC's graphics capabilities is a different matter. Due to the lack of generally accepted language standards in the area of computer graphics, most books provide only a cursory overview of the field, or else a detailed description about only one particular system. The IBM PC's BASIC programming statements for graphics are similar to those found on some other personal computers, but it's confusing at best to try learning more about one computer by studying another.

It is the goal of this book to provide a supplementary text for the IBM PC user who wants to learn more about the system's graphics capabilities using the BASIC programming language. This is *not* an introductory BASIC programming guide. The reader should have an understanding of the fundamentals of BASIC. Only the graphics language aspects of PC BASIC will be covered, with the assumption that the rest of BASIC is reasonably well understood by the reader.

The approach of the book is fairly straightforward. Each chapter builds on the previous one as topics are gradually introduced and expanded upon. Some fundamentals regarding PC hardware are introduced first, followed by a review of each of the various BASIC software statements related to graphics. The two fundamental graphics modes, alphanumeric character text, and all-points-addressable graphics are each discussed. Application examples are given first from the standpoint of static displays such as business graphics, education, and gaming; animation and action graphics follow.

Programming examples that illustrate graphics in the BASIC language are an integral part of the book. These examples fall into three classes:

1. Statement samples that illustrate the many forms the PC's BASIC graphics statements can take.
2. BASIC coding segments (short programs of 3 to 10 lines) that show the simple use of a statement in the context of other BASIC code.
3. Complete, self-contained, and fully documented BASIC programs using one or more of the BASIC statements and graphics concepts that are being illustrated.

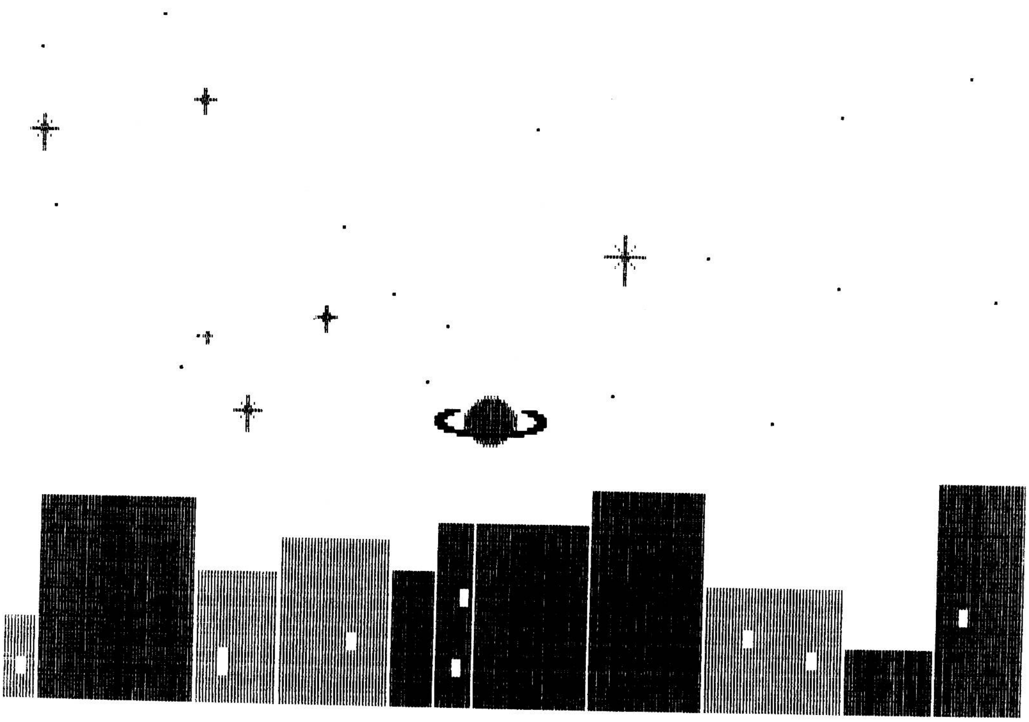
The program examples range from about 20 up to 100 or more lines. A standard format is followed in that each is begun with a series of remarks explaining the nature and purpose of the program. Additional remarks follow throughout the code as an aid to reading and using the program. Standard variable names are used in all programs, such as R for row and C for column. Each program is named and indexed by chapter and a sequence number within that chapter. The logic in each program has been kept fairly simple. No attempt has been made to optimize their performance and operation. These and other changes are left as exercises to the reader. A diskette is available from the publisher containing each of the example programs, plus a menu that aids in selecting a particular program of interest.

Suggestions are offered for making improvements in all programs as well as changes to illustrate additional concepts. Although complete in most respects, the programs should also be considered a starting point, since many of them could easily be expanded to sophisticated applications.

These last points are important. By dabbling with the BASIC code, making changes (for better or worse), and seeing their effect, you will gain a far greater understanding and appreciation for PC graphics than by just reading paragraph after paragraph of text. Furthermore, computer graphics can be very exhilarating. As a graphics image unveils itself on the screen a certain pride of authorship comes over you which is not the same as when a printed screen rolls by. So by all means tinker, play, and enjoy. Your PC has a tremendous potential for graphics which you are now on the road to discovering.

Part One

PC HARDWARE FUNDAMENTALS



1.

SOME BASIC CONCEPTS

ADAPTERS AND MONITORS

In order to fully understand graphics on the PC, a review is needed of some of the various hardware options that are available for it. Foremost in graphics are two items: the display adapter and the CRT display monitor. An adapter is simply a hardware module that accepts data from the CPU and converts it to a format acceptable to the display. Included in the adapter are such components as a memory area, a character generator, connections to the main CPU, and a video convertor for the final output to the display unit.

Display Adapters

One of the primary tasks of the adapter is converting alphanumeric character text from its internal binary-coded format to a form that may be displayed. Suppose you type the three letters ABC on your PC. Each character is internally stored as 8 bits of data, based on what is known as the ASCII character set. Each of the three characters may be viewed internally as:

```
01000001 01000010 0100011 in binary
              101  102  103 in octal
                 65  66  67 in decimal
                 41  42  43 in hexadecimal
```

depending on how you want to look at them. Converting these values to the actual patterns and shapes that represent the characters is done by the port character generator (Figure 1.1).

There are two types of adapters available from IBM for the PC: the monochrome display adapter and the color/graphics display adapter. The monochrome adapter comes in two forms, one with an integrated parallel printer adapter and the other without the printer adapter. Both types of adapters are identical in function for display purposes.

Figures 1.2 and 1.3 are simplified block diagrams of each adapter. Each adapter has its own internal memory, called a display buffer, for holding the data it receives from the CPU for display. An address from the CPU also determines where in the buffer to store the data, which in turn affects where it appears on the display screen. The display buffer is physically separate from the CPU's main

Figure 1.1 Conversion of data to its external form

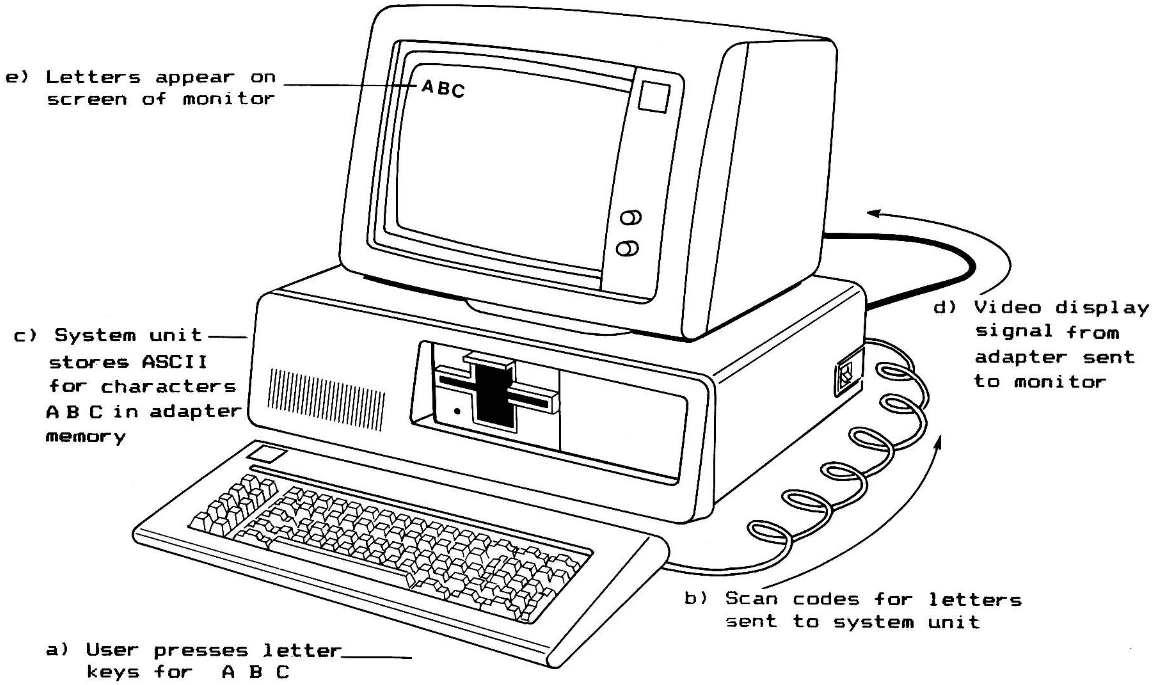


Figure 1.2 Monochrome display adapter block diagram

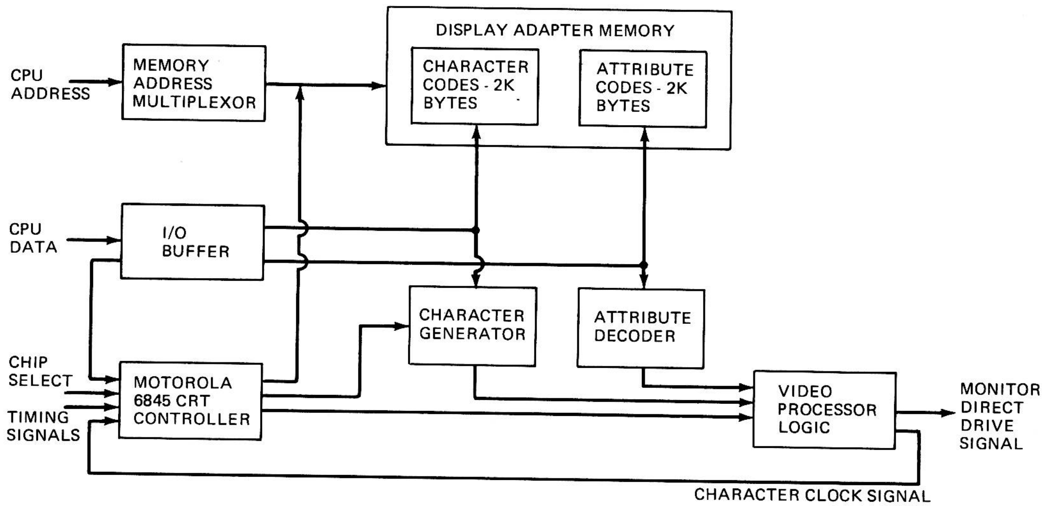
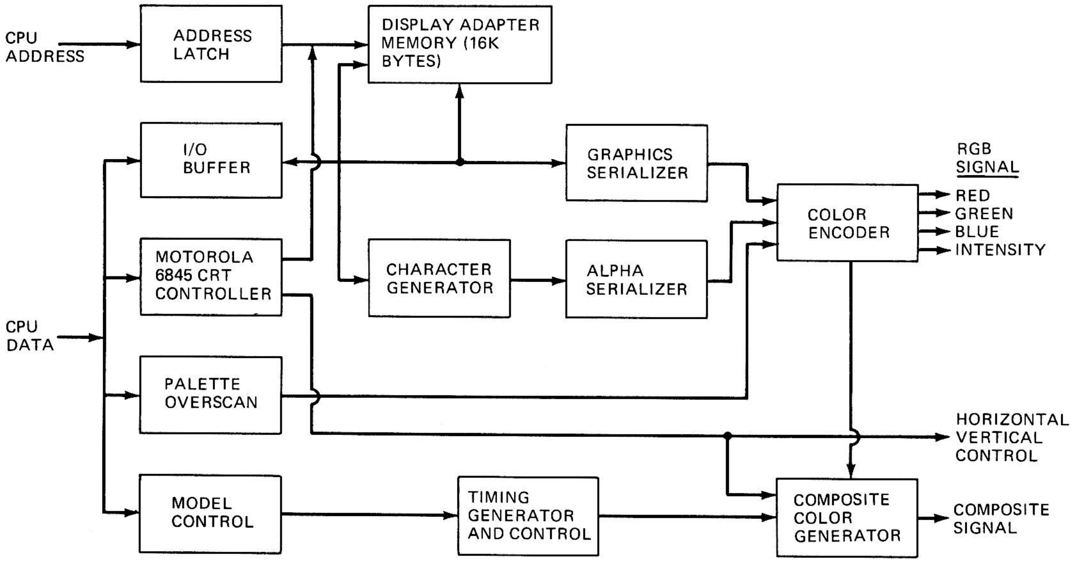


Figure 1.3 Color/graphics display adapter block diagram



memory, but appears as though it were a continuous part of all of memory. Thus programs can be written in BASIC to directly access the display buffer in the same manner as all other memory by using the standard PEEK and POKE commands.

The entire memory address space of the PC is over 1 million bytes. Within that address space are included not only the display memories but also ROM for the BASIC interpreter and the Basic Input/Output System (BIOS), as well as user memory and numerous reserved areas for expansion. Figure 1.4 lists the major memory areas of the PC.

It is the job of BIOS, among other things, to manipulate the characters that appear on the display screen. This includes such functions as displaying a single character or whole line of characters, scrolling up (or down) the display, clearing the screen, and so forth. All character movement takes place in the adapter's display buffer and is under the control of BIOS and the PC's CPU. The adapter, in turn, continually outputs a video signal that represents everything in the buffer for the screen of the display monitor.

The Motorola 6845 CRT controller chip performs the function of monitoring the display buffer and creating this video signal. The controller may be reprogrammed to a certain extent to create other graphics modes. It contains a set of 19 registers that may be changed by using BASIC's OUT statement. For instance, in the PC's Guide to Operations manual is a section on shifting the display image to the left or right for those using a television set for a monitor. The two OUT statements listed select one of the 6845 registers, and then change it to shift the left margin. More details on each of the registers can be found in the PC's Technical Reference manual section on the color/graphics adapter.

Figure 1.4 PC memory map

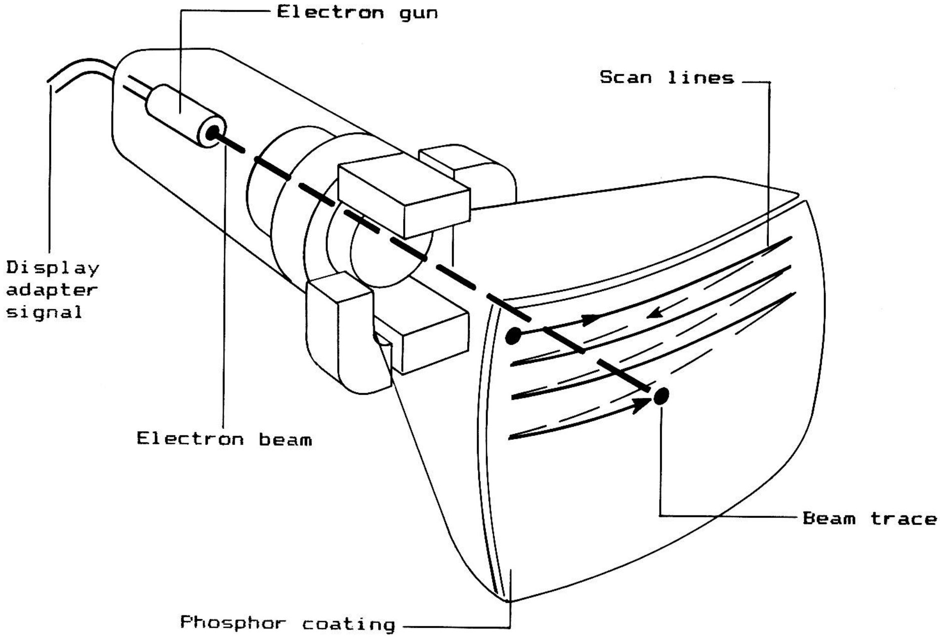
Hex Memory Address	Memory Function	
0	16 to 64 KB system board	
10000	I/O Channel Memory Expansion (maximum 192 KB)	256 KB RAM I/O Channel
40000	Additional memory expansion (maximum 384 KB)	384 KB RAM expansion
A0000	16 KB Reserved Area	
B0000	Monochrome Adapter Buffer memory (4 KB)	128 KB RAM Graphics/Display Buffer Areas
B8000	Color/graphics Adapter Buffer memory (16 KB)	
C0000	192 KB Memory Expansion area	
F0000	16 KB Reserved Area	256 KB ROM System Area
F4000	48 KB BIOS and Cassette BASIC	
FC000		

The color/graphics adapter also contains logic that allows point-by-point color graphics to be displayed. In doing so the adapter examines its buffer on a point-by-point basis instead of its usual line-by-line, character-by-character manner. The output may be in one of two forms: the RGB color encoder or the composite color generator. Either a color or black and white monitor may be used with the color/graphics adapter.

Display Monitors

A typical black and white raster-scan monitor tube is illustrated in Figure 1.5. An electron gun emits a beam of electrons which strikes a phosphorous coating inside the face of the screen. This causes it to glow in patterns of dots that

Figure 1.5 Typical black and white raster-scan monitor tube

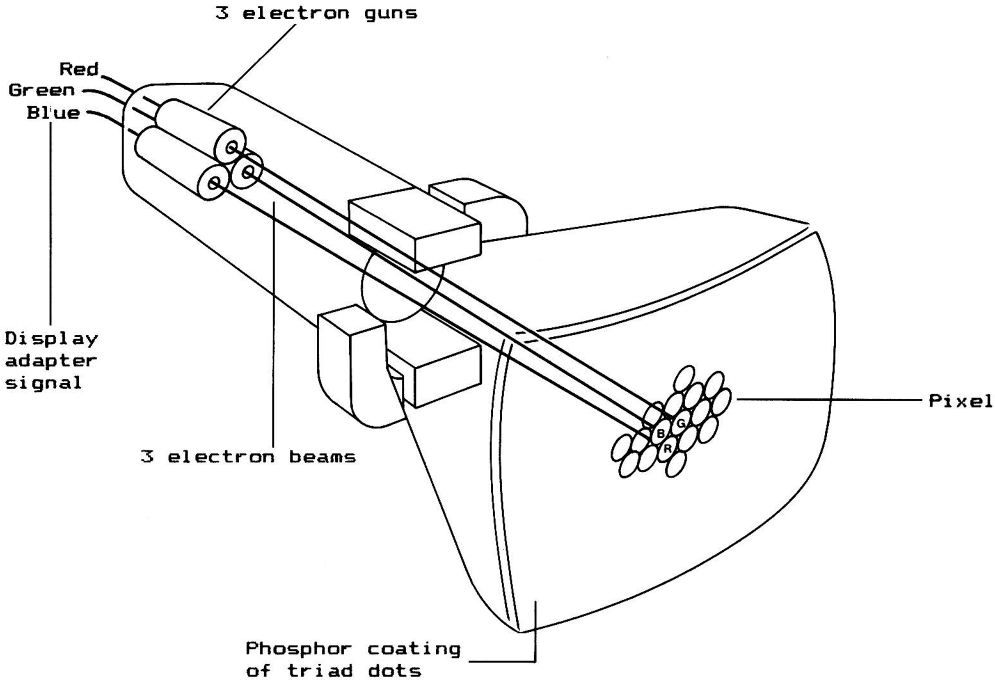


represent the data being displayed. The term "raster-scan" refers to the manner in which the electron beam scans the face of the screen. It criss-crosses back and forth across the screen so fast that the glowing dots are refreshed by a new pass of the beam before the glow from the previous pass can fade away. An electromagnetic field within and about the neck of the monitor tube causes the actual beam movement. The brightness of an image is controlled by the intensity of the beam.

In principle, a color monitor is like a black and white monitor, but with three electron guns and a different type of phosphorous coating inside the face of the tube. Each of the three electron guns emit a single beam that controls a single color (either red, green, or blue) by hitting a separate dot in the phosphorous coating. This coating consists of many thousands of three-dot triads of phosphor that glow either red, green, or blue when struck by an electron beam. Each of the three beams causes only its particular phosphor to glow. Based on each beam's intensity, all three colors are essentially mixed by glowing together to produce all other colors. Figure 1.6 illustrates the overall organization of a color monitor tube.

Common to both black and white and color monitors is a measurement known as resolution. This describes the level of detail in terms of horizontal lines and vertical points the monitor is capable of displaying. The greater the resolution, the better the clarity of the image being displayed. But resolution also depends on the ability of the adapter to display a comparable number of points.

Figure 1.6 Typical color monitor tube



Thus a high-resolution monitor does not gain that much in detail if a lower-resolution mode is used by the adapter attached to it. It is best to match as closely as possible the monitor's resolution with the resolution of the mode that will be most used by your system.

Types of Color Monitors

There are essentially two types of color monitors: the RGB monitor and the composite monitor. An RGB monitor accepts three video signals as input, one for each color. The IBM color/graphics adapter adds to this by providing a fourth signal that controls what it calls intensity. Thus the color/graphics adapter produces 16 colors. The first eight are based on combinations of each of the three primary colors, and the second eight consist of "intensified" versions of the first eight, that is, the fourth monitor signal is turned on. Some RGB monitors do not recognize the intensity signal, so only eight colors can be displayed. Both groups of eight colors will appear the same. Most manufacturers have added kits or patches to their equipment to handle the extra signal.

Composite monitors accept a video color signal that has the information about color and its intensity encoded differently. This encoding difference is