Advanced Graphics with the BBC Model B Microcomputer

Macmillan Microcomputer Books

General Editor: Ian Birnbaum

Advanced Graphics with the BBC Model B Microcomputer
Ian O. Angell and Brian J. Jones
Assembly Language Programming for the BBC Microcomputer
lan Birnbaum

Also from Macmillan

Advanced Graphics with the Sinclair ZX Spectrum
Ian O. Angell and Brian J. Jones
Advanced Programming for the 16K ZX81 Mike Costello
Beginning BASIC Peter Gosling
Continuing BASIC Peter Gosling
Practical BASIC Programming Peter Gosling
Program Your Microcomputer in BASIC Peter Gosling
Codes for Computers and Microprocessors P. Gosling and Q. Laarhoven
Microprocessors and Microcomputers - their use and programming
Eric Huggins
The Sinclair ZX81 - Programming for Real Applications
Randle Hurley
More Real Applications for the ZX81 and ZX Spectrum Randle Hurley
Z80 Assembly Language Programming for Students Roger Hutty
Digital Techniques Noel Morris
Microprocessor and Microcomputer Technology Noel Morris
The Alien, Numbereater, and Other Programs for Personal Computers - with notes on how they were written John Race
Understanding Microprocessors B. S. Walker
Assembly Language Assembled - for the Sinclair ZX81 Anthony Woods

Advanced Graphics with the BBC Model B Microcomputer

Ian O. Angell and Brian J. Jones

Department of Statistics and Computer Science
Royal Holloway College
University of London
Egham, Surrey

MACMILLAN PRESS
LONDON

First published by
THE MACMILLAN PRESS LTD
London and Basingstoke
Companies and representatives
throughout the world

Typeset by RDL Artset Ltd, Sutton, Surrey

Printed in England by Pindar Print Limited, Scarborough, North Yorkshire.

ISBN 0 333 35052 9 (book)
0 333 35053 7 (cassette 1)
0 333 36141 5 (cassette 2)

Contents

Preface

With the rapid advance of computer technology has come a substantial reduction in the price of computer hardware. In the coming years the price of peripheral devices will also tumble. This means that users with a limited budget, who previously had access only to the most elementary computing devices, will soon be able to afford the most sophisticated computers. They will also be able to escape from the limitation of tabular numerical output and buy microprocessor attachments for television monitors or inexpensive special-purpose colour graphics devices. Software, however, does not appear to be getting cheaper.

Because ofthe enormous capital expenditure that was required to set up graphical output in the past, both for machines and for software, the subject of computer graphics has so far been the preserve of large research groups. This inaccessibility has led to a mystique growing up around the subject and it has thus achieved a false reputation for difficulty. This book is an attempt to lay the ghost of complexity; it will also show that complicated (and hence expensive) software packages, which are naturally of great value in research organisations, need not frighten away the average computer user. For most purposes these packages are unnecessary. This book, as well as being an introduction to computer graphics, may be considered a (very inexpensive) software package: it is a lot cheaper than commercially available packages! Naturally, because of this fundamental approach, users have to achieve a reasonable understanding of their graphics device before pictures, other than those provided, can be drawn. This need not be a disadvantage; the amount of groundwork required will be seen to be very limited and, as a direct result, the user' s knowledge grows along with th, package and he is far less likely to misinterpret any of the graphical procedures. References are given and relevant further reading material is also recommended in order to expand the reader' s horizons in the subject.

It is assumed that the reader has an elementary knowledge of Cartesian coordinate geometry (the authors recommend books detailed in Cohn, 1961, Coxeter, 1974 and McCrae, 1953 - see the references) and also the BASIC programming language (see the BBC User Guide - page numbers are not given because this excellent handbook is constantly being updated as the BBC micro is being extended). Many interesting programming exercises are proposed, and these should raise the standard of the reader' s BASIC expertise. BASIC is a universally popular language that is available (in various guises) on all types of microcomputer, so the programs can be easily adjusted to run on micros other

than the Model B: it is also a good medium for transmitting the algorithms that are used in computer graphics, so enabling readers to translate these ideas readily into any other computer language of their choice.

The concepts necessary for the study of computer graphics are organised as a combination of theory and worked examples; these are introduced as and when they are needed in the natural progression of the subject. Program listings that form part of the examples may be considered not only as algorithms that describe solutions to fundamental graphical problems, but also as a computer graphics software package in BAS1C, orjust as programs to draw patterns. Alongside the examples are a series of exercises that expand on these ideas. The practical problems that are implicit in programming the various concepts ofcomputer graphics are often more a source of difficulty to the student than the concepts themselves. Therefore it is essential that readers implement many of the program listings given in the book in order to understand the algorithms, as well as attempt a large number of the exercises. As an extra learning aid, two companion audio-cassette tapes are being made available; these contain most of the larger program listings that are given in this book. If readers are frightened by the mathematics they should run the programs first before studying the theory.

This approach to the subject has been used with great success in teaching computer graphics to undergraduates and postgraduates at Royal Holloway College. Quickly producing apparently complex pictures results in the positive feedback of enthusiastic interest. The ability to construct pictures on line-drawing and colour interactive graphics VDUs makes a long-lasting impression on the student; and the step by step approach brings him very quickly to the level of very sophisticated computer graphics. That level is outside the scope of this book, but where necessary the reader will find relevant references to guide him into the more advanced topics.

This book is aimed at those who are competent BASIC programmers but who are complete beginners in graphics. It contains the elementary ideas and basic information about pixel and two-dimensional graphics which must be mastered before attempting the more involved ideas of character and three-dimensional graphics. This is followed by a section relating to character graphics and the display of data (in line drawings and colour) - probably the most important non-specialised, commercial use of computer graphics. Later chapters introduce the reader to the geometry of three-dimensional space, and to a variety of projections of this space on to the two-dimensional space of graphics devices. The related problems of hidden lines and hidden surfaces, as well as the constmction of complex three-dimensional objects, are dealt with in detail. Finally we return to advanced ideas in BASIC programming and give a large worked example of a video game (to be found on cassette 2).

Graphics is one of the most rapidly expanding areas of computer science. It is being used more and more in the fields of Computer Aided Design (C.A.D.), Computer Assisted Management (C.A.M.) and Computer Assisted Learning

(C.A.L.). At one time it was only the big corporations such as aircraft and automobile manufacturers who used these techniques, but now most companies are realising the potential and financial savings of these ideas. What is more, not only is computer graphics profitable, it is fun! The BBC microcomputer is an ideal machine on which to learn the basics of computer graphics, and an excellent springboard up to the most sophisticated (and expensive) graphics devices.

We hope this book will display some of the excitement and enthusiasm for computer graphics experienced by us, our colleagues and students. To demonstrate just how useful computer drawings are for illustrating books and painphlets, all the pictures in the following chapters were drawn by computer specifically for this book.

Ian O. Angell
Brian J. Jones

Introduction

This book may be read at a number of different levels. Firstly, it can be considered as a recipe book of graphics programs for those who simply want to draw complex pictures with their BBC microcomputer. We naturally hope that the reader, haing drawn these figures, will be inspired to delve deeper into the book in order to understand how and why the programs were constructed. Secondly, some of the programs can be used as a package to produce and to label data diagrams (pie-charts, histograms and graphs) for business and laboratory applications. Finally, and the main objective in writing the book, it is an introductory text to computer graphics that leads the reader from the elementary notions of the subject to such advanced topics as character graphics, construction of three-dimensional objects and hidden surface (and line) algorithms.

The complex programs given 1ater in the book are much too involved to be compiled as single listings; furthermore, there is a great deal of repetition in the use of elementary algorithms. Therefore the *top down* or *modular* approach is used in writing and in explaining programs. The solution to each major graphics problem is conceived as a series of solutions to subproblems. These subproblems can be further broken down into a set of problems to be solved (*modules*). Such modules are programmed in the form of BAS1C procedures. Each is given an identifier (in lower case characters) and will solve a particular subtask. Submodules are then combined to solve the major graphics problem. The program listings present the algorithms that are needed for the solution of subtasks, and the naming of the procedures makes an understanding of the algorithms easier. We use lower case characters for procedure identifiers (and groupings of procedures in the text) only: all other program variables are in upper case characters to aoid confusion.

Two cassette tapes are available to accompany the text; they contain all the larger listings in the book, as well as the data for diagrams and character sets used in later programs (which would otherwise hne to be constructed by the readers themselves - a rather time-consuming process). The first cassette consists of the two- and three-dimensional geometrical programs, and the second contains the character graphic manipulation, diagram construction and video games etc.

A list of complete programs is given at the end of each chapter, together with suitable data values, for those who want nothing more than to run these programs. 1n fact it is a good idea for everyone, including the more serious readers, to LOAD the relevant programs from the tape and run them before reading any particular chapter.

There are many REMarks in the program listings, however, and hence some

of the programs approach the storage limits of the BBC micro. 1n these cases you should delete the REMarks before saving the programs. To make the listings easy to read we advise readers to L1STOl the programs in MODE 7. We llave placed a REMark in red before each procedure (on lines with numbers that end in 0) so that they stand out: all other REMarks are in green (on lines with numbers that end in 9). You may find that the latter REMarks take up too much store in which case you should strip them away by typing AUT09, 10 and by holding the RETURN key down. Even then some of the programs are too big to fit into the store, in which case you must LOAD them after setting PAGE=&1100.

As an example ofwhat to expect we give the program that is required to draw figure 1.1, a drawing of a body of revolution in which all the hidden surfaces have been suppressed.

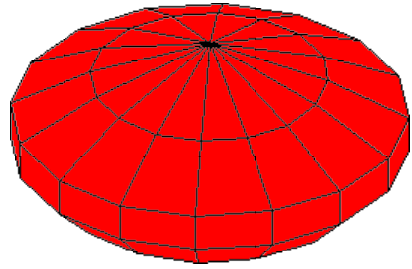

Figure 1.1

The program requires the listings 2.1 (‘start’), 2.2 (two functions FN X and FN Y), 2.3 (‘setorigin’), 2.4 (‘moveto’), 2.5 (‘lineto’) and 2.7 (‘triangle’). This combination of procedures will be called ' libl' , and it was designed for drawing line figures on the television screen.

To ' lib1' must be added listings 3.3 (‘angle’), 8.1 (‘multt3' and ' idR3’), 8.2 (‘tran3’), 8.3 (‘scale3’), 8.4 (‘rot3’), 9.1 (‘1ook3’) and 9.2 (‘main program’). Procedures, which when combined we call ' lib3' , are used for transforming and for observing objects in three-dimensional space.

Listing 10.5 (‘revbod’) is also needed, together with the ' scene3' procedure given in listing 1.1.

Listing 1.1

```
6000 REM scene3 / flying saucer
6010 DEF PROCscene3
6020 LOCAL I%
6030 DIM X(12),Y(12),XD(6),YD(6)
6040 DIM A(4,4),B(4,4),R(4,4)
6050 DATA 0,3, 3,2, 5,1, 5,0, 4,-1, 0,-3
6060 RESTORE
6069 REM INPUT horizontal data
6070 NUMV=5
6080 INPUT"NUMBER OF HORIZONTAL LINES",NUMH
6090 INPUT"INITIAL ROTATION",PHI
6099 REM READ definition set
6100 FOR I%=1 TO NUMV+1
6110 READ XD(I%),YD(I%)
6120 NEXT I%
6130 PROCidR3 : PROClook3
6140 PROCrevbod
6150 ENDPROC
```

Figure 1.1 requires tlie data HORIZ = 12, VERT = 9, EX = 1 , EY = 2, EZ = 3, DX = 0, DY = 0, DZ = 0, number of horizontal lines; NUMH = 16 and initial rotation PHI = 0. Each value has to be typed in individually, when requested by the machine. Run the program with different data values. What happens if HORIZ = 6 and VERT = 4 and the other values stay the same? Set HORIZ = 16, VERT = 12, EX = 1 , EY = -2, EZ = 3, DX = 1, DY = 0 and DZ = 0. Try NUMH = 20, PHI= 0.1. You will have to read up to and including chapter 10 to understand the details of what is happening.

This example illustrates the reasoning behind the layout of this book. Assuming that you are a fast typist, or that you have bought the accompanying cassettes, then a relatively complex three-dimensional picture cal1 be constructed very quickly with a minnimum of effort. Even one-finger typists (like the authors) will have little difficulty in implementing this and the other programs, before they go on to study the book in detail.

We hope that this example will inspire you to implement *all* the programs in this book, to try most of the examples, and then to go on and draw your very own computer graphics pictures.

Now read the rest of our book and we wish you many happy hours with your BBC microcomputer.