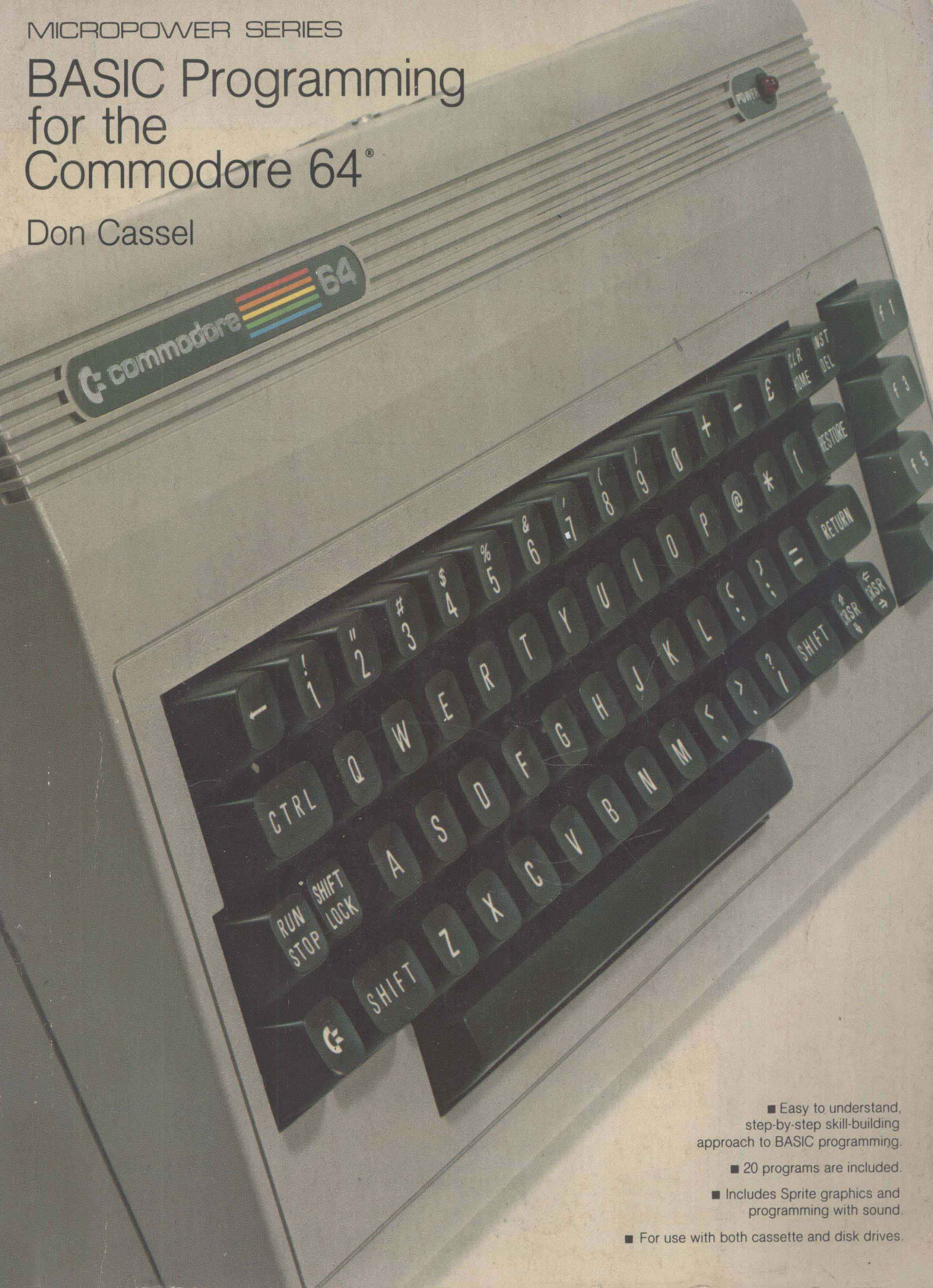


MICROPOWER SERIES

BASIC Programming for the Commodore 64®

Don Cassel



■ Easy to understand, step-by-step skill-building approach to BASIC programming.

■ 20 programs are included.

■ Includes Sprite graphics and programming with sound.

■ For use with both cassette and disk drives.

MICROPOWER SERIES

BASIC Programming for the Commodore 64[®]

Don Cassel

Humber College

wcb

Wm. C. Brown Publishers
Dubuque, Iowa

Cover photo is provided courtesy of Bob Coyle

Figures 1.1, 2.1, 2.2, 2.3, 2.4, and Appendix F are reproduced courtesy of Commodore Business Machines.

Edouard J. Desautels, University of Wisconsin-Madison
Consulting Editor

Copyright © 1984 by Wm. C. Brown Publishers. All rights reserved.

Library of Congress Catalog Card Number: 83—73414

ISBN 0—697—09912—1

2—09912—01

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher.

Printed in the United States of America.

BASIC Programming for the Commodore 64®

Preface

This book introduces the BASIC language used on the Commodore 64 microcomputer. If you are a beginner at programming in BASIC, this material will give you a gradual introduction to the language, stressing a hands-on approach with your Commodore 64.

The BASIC used here is BASIC 2.0, released by Commodore since 1981. It is virtually identical to the BASIC used on Commodore PET and CBM computers and the Commodore VIC-20, with a few slight differences where disk applications are concerned. Major differences between the VIC and C-64 are the color graphics and music capabilities, which are far more extensive on the Commodore 64.

The first chapter is a brief introduction to the C-64 hardware with BASIC and DOS for the floppy disk. Chapter 2 discusses the use of the keyboard, tape, and floppy disk. If you have used the C-64 before, you may skip these chapters or read them quickly.

Chapter 3 begins with an elementary introduction to BASIC, stressing a hands-on approach. By entering the program code on your C-64 as you read, you will get immediate feedback from your computer. Enough BASIC statements are introduced in this chapter for you to begin to write some useful programs at this early stage.

Chapter 3 ends with six complete programs that may be run on the C-64. In these six programs, two things are emphasized.

One is the need for planning a program. The approach taken shows how input and output must be defined before the program is written and then how to make use of an English code (Pseudo code) to develop the general program logic. Later the flowcharts concept is also discussed.

The second emphasis in the sample programs is how to apply the language statements just discussed in the chapter to a variety of situations. Each program is designed to expose you to realistic situations requiring the use of the BASIC statements you have just read about.

Subsequent chapters follow the same pattern as chapter 3 but go into more depth in the language, as indicated in the table of contents. Each chapter has numerous examples and ends with several programs that apply the new features of the language. Each program has been completely developed using the techniques of program design to establish the program logic.

Chapter 7 deviates slightly from this pattern to discuss ways we can communicate with our program users. Since you will sooner or later write programs for other people, this chapter considers effective interaction with users of your program and how to implement these concepts on the C-64.

Next, in chapter 8, we look at graphics, animation, and sound. These topics will be of particular interest since these are powerful capabilities for such an inexpensive computer. Chapter 9 covers the use of sequential files on tape, and chapter 10 explains sequential files on disk. Again, there are examples and programs to try as we make the concepts clear.

Finally, chapter 11 discusses the procedure for debugging your programs. As you will no doubt discover very early, programs don't just work immediately after you have written them. In fact, a program that works perfectly the first time is the exception rather than the rule. So this chapter discusses some techniques for finding your bugs and correcting them.

Remember, this is an introduction to BASIC programming on the Commodore 64. The book does not pretend to be an exhaustive treatment of programming, and there is much more to be learned about the C-64 beyond this level. However, I hope you will find the book instructive and helpful in your quest for learning to program, and I trust it will provide the necessary foundation for you to move on to more advanced programming on your Commodore 64.

Contents

Preface xi

1

Introduction to the Commodore 64 1

COMMODORE 64 HARDWARE 1
MEMORY 2
BASIC 2
DOS (DISK OPERATING SYSTEM) 3
REVIEW QUESTIONS 3

2

Getting Started on the Commodore 64 5

TURNING ON THE COMMODORE 64 5
KEYBOARD CHARACTERS 6
Commodore 64 Graphics Keyboard 6
USING THE CASSETTE TAPE 7
Inserting a Cassette 7
Tape Controls 8
Loading a Program 8
Saving a Program 8
USING THE FLOPPY DISK DRIVE 9
Loading a Program 9
Saving a Program 10
Replacing a Program 10
Directories 10
Formatting a Diskette 10
Backups 11
REVIEW QUESTIONS 11

3

Elementary BASIC Programming 13

WHAT IS BASIC? 13
CALCULATOR MODE 13

HOW TO USE CURSOR CONTROLS	14
INSERT/DELETE	15
CLEAR/HOME	15
RUN/STOP	15
LIST COMMAND	16
BASIC NUMBERS	16
Integers	17
Real Numbers	17
Scientific Notation	19
STRINGS	20
VARIABLES	20
MULTIPLE STATEMENTS PER LINE	22
USING COLOR	22
ARITHMETIC STATEMENTS	23
Add (+)	23
Subtract (−)	24
Multiply (*)	24
Divide (/)	24
Exponentiation (↑)	24
HIERARCHY AND PARENTHESES	24
PRINT	25
INPUT	27
GOTO	28
REM	29
SIMPLE CALCULATION PROGRAM	29
FAHRENHEIT—CELSIUS PROGRAM	30
IMPROVING YOUR SOLUTION	32
ELEMENTARY GRAPHICS—SAILBOAT	33
WEIGHTED AVERAGE PROGRAM	33
COMPUTING LOAN PAYMENTS	36
REVIEW QUESTIONS	37

4 **Not So Basic BASIC** 39

DIM	39
Storing Values in an Array	40
Multidimensional Arrays	41
IF—THEN	42
FOR—NEXT	44
GOSUB—RETURN	45
STOP	47
GENERATING RANDOM NUMBERS	47
NUMBER GUESSING GAME	48
TIME DELAYS	50
IMPROVED WEIGHTED AVERAGE PROGRAM	52
ESTIMATING TRIP COSTS	53
REVIEW QUESTIONS	57

5 **More on Input and Output** 59

READ—DATA	59
CREATING A BAR GRAPH	60
WEIGHTED AVERAGE WITH DATA	61

RESTORE 62
MORE ABOUT PRINT 63
 Cursor Controls 63
 TAB 65
 SPC 65
METRIC CONVERSION PROGRAM 66
SIMPLE PAYROLL PROGRAM 74
REVIEW QUESTIONS 77

6

Advanced BASIC 79

GET 79
ON—GOSUB 80
ON—GOTO 81
POKE 83
PEEK 85
RANDOM STARS 86
TI AND TI\$ FUNCTIONS 86
 TI or Time Function 86
 Reaction Timer Program 86
 TI\$ or Time\$ 87
DEF FN 88
ARITHMETIC FUNCTIONS 90
 ABS 90
 ATN 90
 COS 90
 EXP 91
 INT 91
 LOG 91
 RND 91
 Creating a Specific Set of Random Numbers 92
 SGN 92
 SIN 92
 SQR 93
 TAN 93
 Converting Radians to Degrees 93
STRING FUNCTIONS 93
 ASC 93
 CHR\$ 94
 LEFT\$ 94
 LEN 95
 MID\$ 95
 RIGHT\$ 97
 STR\$ 97
 VAL 97
CONCATENATION 97
PLOTTING GRAPHS 98
CONTROLLING DECIMAL POSITIONS 102
 Driver Routines 104
CAI CHESS AND PROGRAM GENERALIZATION 105
 The Data 105
 The Program 107
REVIEW QUESTIONS 112

7

Interacting with the User of Your Program 113

USER LEVEL	113
1. Casual	113
2. Trained	113
3. Programming Skills	113
USER DIALOGUES	114
Prompting	114
DEFAULT RESPONSES	115
MENUS	116
MULTILEVEL MENUS	117
FORM FILLING	119
COMMAND LANGUAGES	119
REVIEW QUESTIONS	120

8

Graphics, Animation, Sound, and Music 121

GRAPHIC CHARACTER SET	121
REACTION TIMER WITH GRAPHICS	124
USING POKE TO PRODUCE A GRAPHIC	126
Lunar Lander	126
Chessboard	127
ANIMATION	128
Rocket 1	128
Rocket 2	129
Egg Timer	130
Train—Animating Multiple Objects	131
HIGH RESOLUTION ON THE COMMODORE 64	132
SPRITE GRAPHICS	135
Sprite Design	135
Sprite Registers	136
Sprite Movement	137
Moving a Sprite Horizontally	138
Moving a Sprite Vertically	138
Sprite Size	139
Sprite Color	139
GENERATING SOUND AND MUSIC	139
Volume	140
Attack/Decay Setting	140
Sustain/Release Setting	141
Frequency	141
Waveform Settings	142
Playing a Single Note	143
Take Me Out to the Ballgame	143
Note Duration	143
Sound Characteristics	144
REVIEW QUESTIONS	145

9

Tape Files Extend Your Reach 147

- CONCEPTS 147**
- OPEN AND CLOSE 148**
- PRINT# 149**
- WRITING BUDGET NAMES ON TAPE 149**
- INPUT# 150**
- READING THE BUDGET NAMES TAPE 151**
- HOW TO HANDLE RECORDS WITH MULTIPLE FIELDS 152**
- UPDATING A TAPE FILE 154**
- THE CASH-FLOW PROBLEM 156**
 - New File 157**
 - Old File 157**
 - Enter/Review Data 157**
 - Display Balance 158**
- SUMMARY OF TAPE FILES 164**
- REVIEW QUESTIONS 164**

10

Disk Files 165

- SEQUENTIAL FILES 166**
- OPEN AND CLOSE 166**
- PRINT# AND INPUT# 168**
- WRITING BUDGET NAMES ON DISK 168**
- READING BUDGET NAMES FROM DISK 169**
- HOW TO HANDLE MULTIPLE FIELDS ON DISK FILES 169**
- DETECTING DISK ERRORS 171**
- MAINTAINING THE CHECKBOOK 172**
- REPLACING A CURRENT FILE 172**
- USING VARIABLE FILENAMES 172**
- ENGLISH CODE 173**
- DISPLAY SCREEN PAGE 174**
- DELETING TABLE ENTRIES 175**
- REVIEW QUESTIONS 179**

11

How to Debug Your Programs 181

- DESK CHECKING 182**
- SYNTAX ERRORS 182**
- TEST DATA PREPARATION 184**
- IMMEDIATE MODE DEBUGGING 184**
- TRACING PROGRAM LOGIC 185**
 - Manual Tracing 185**
 - Automatic Program Tracing 188**
- REVIEW QUESTIONS 194**

Appendices 195

- A Basic Operating Commands: CLR, LIST, LOAD, NEW, RUN, SAVE, VERIFY 195**
- B Reserved Words 198**
- C Abbreviations 199**
- D Disk Error Messages 200**
- E C-64 ASCII and PEEK/POKE Codes 201**
- F ASCII and CHR\$ Codes 202**

Index 205

1

Introduction to the Commodore 64

BASIC is the language the majority of microcomputers use, so by learning it on your C-64 you will be prepared to apply programming to a wide variety of micros.

Possibly you purchased your C-64 to play some of the exciting games available on cartridge or tape but now you want to explore some of the C-64's further capabilities. Many people have used the C-64 for a wide range of solutions to problems, running from home finance, record keeping, and investment to music skills and child education. The list is endless. However, before you can apply the computer to your interests, you must learn to use the tool effectively. After studying the contents of this book you will be equipped to solve many of your programming applications with BASIC. But first let's look at the hardware.

COMMODORE 64 HARDWARE

Figure 1.1 shows the C-64 microcomputer with the floppy disk drive, printer, and tape used with the Commodore computer. Hidden inside the C-64's case are the electronic components, a microprocessor and a memory, necessary for its operation.

To understand C-64 basics, it is useful to compare the computer to parts of the human body. For example, when you want to learn something, you might read a book, as you are doing now. The new information becomes input to your brain, where it is stored in memory. Similarly, you can enter new information into the computer through the keyboard and then store it temporarily in the C-64's electronic memory or storage.

After you have read something, you may think about it by processing it mentally. Then you might discuss your conclusions with someone or maybe simply repeat verbally what you have read. Your voice is then output from the brain. When information has been processed by the computer's processor (brain?), that same information or a new arrangement of it may be displayed on the TV screen, which is the C-64's output device.

These physical components of the computer—display, keyboard, processor, and so on—are called the hardware. The program used to describe the processing steps is called the software. The program, residing in the computer's memory when it is being used, describes in precise steps how the computer is to process the data.

If your brain is anything like mine you will not likely remember everything you have read. One solution to this is to make notes, another kind of output. Although the computer never forgets anything entered into its memory, it does have limited capacity. Therefore another form of output is often



Figure 1.1 C-64 hardware

used to record or store information outside the computer. One of these is the tape cassette. Another is the floppy disk or diskette. Information recorded on the tape or disk can be stored and read back into the computer at a later time when needed, just as you can read your notes to jog your memory. Tape and disk are commonly used to store programs as well as other data.

MEMORY

When you type something on the keyboard or read from tape or disk, the information goes into the C-64's memory or RAM (Random Access Memory) as shown in figure 1.2. RAM is a solid-state memory device that is a part of the C-64's circuitry, and programs or data may be read or written into it. Normally only one program at a time goes into memory; when a new program is read from tape or disk it replaces the previous program in memory.

Memory in the C-64, as the name implies, is 64K or 64,000 storage positions. Each storage position may hold only one character (i.e., a letter or a number), so a 64K Commodore 64 can store 64,000 characters, or bytes in computer lingo. (Technically, 1K bytes is 1,024 bytes, so 64K is actually 65,536 bytes.) BASIC programmers have access to only 38,911 bytes since the computer uses the remaining memory. But even 38,911 bytes will give you lots of room for some sophisticated programs.

Also shown in figure 1.2 is a component called ROM, for Read Only Memory. As the name suggests, ROM may be read only by the computer. The most significant part of ROM is the BASIC interpreter it contains. This interpreter is a program that has been prerecorded in the C-64's ROM, which the computer uses to run your BASIC programs.

BASIC

BASIC is the primary language for programming the C-64, the means by which you give instructions to the computer to solve a particular problem. BASIC, like human languages, has rules that must be followed to use the language effectively. Fortunately, as the name BASIC suggests, it is not nearly as complicated as English or French. In fact, BASIC is one of the easiest of all computer languages to learn.

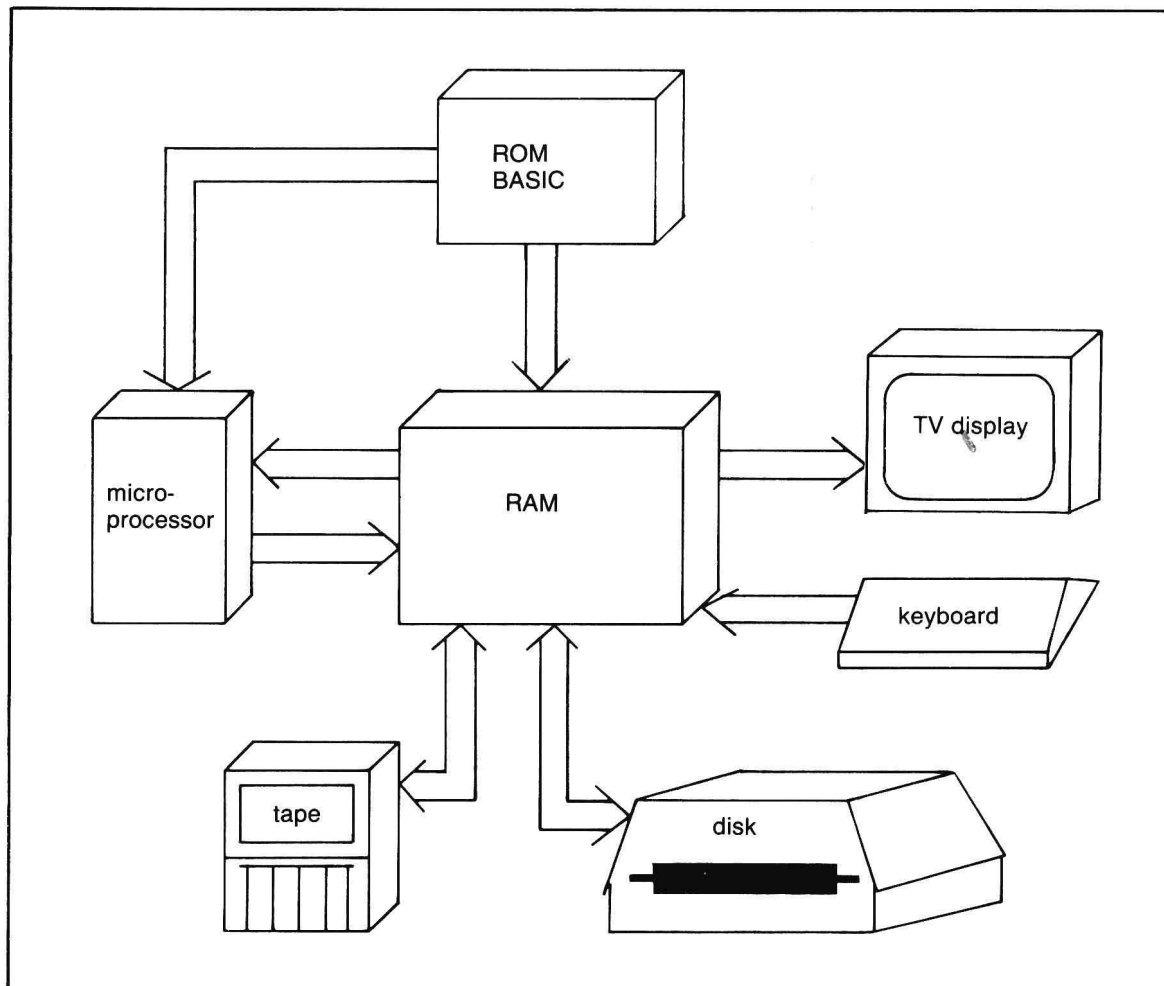


Figure 1.2 Components of a Commodore 64 system

DOS (DISK OPERATING SYSTEM)

DOS is the program in the floppy disk drive that controls the reading and writing of disk files. Without DOS, working with disk would be terribly complex. Although we will introduce the use of DOS at appropriate times, most of our discussion will refer to tape as the primary storage device. So if you don't have a disk drive, don't worry; tape will work equally well.

REVIEW QUESTIONS—CHAPTER 1

1. Name some of the components of the Commodore 64 microcomputer.
2. Compare the concepts of computer input, process, and output to the human body. Can you think of any other analogy?
3. What is hardware? Give some examples.
4. What is software? What purpose does it serve in the microcomputer?
5. What does RAM mean?
6. Discuss BASIC's function on the C-64.
7. What device must your computer have if you need to use DOS?

2

Getting Started on the Commodore 64

If you are familiar with the C-64, you might wish to skip ahead to chapter 3. Otherwise you should read this chapter to learn its basic operation.

The C-64 (figure 2.1) comes with the full-sized graphics keyboard and 64K of RAM. A TV is used for the display, which shows 40 characters per line. The most basic configuration for the C-64 is with a cassette tape; the floppy disk and/or printer are possible additions.

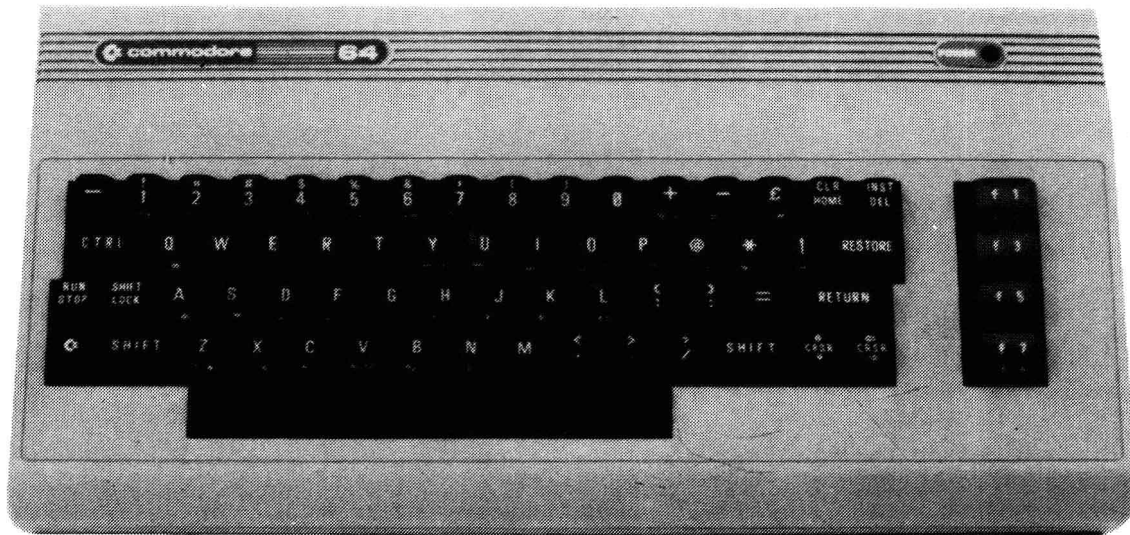


Figure 2.1 C-64 microcomputer

TURNING ON THE COMMODORE 64

First be sure the C-64's power supply is plugged into a wall outlet and that the other end of the power supply cord is plugged into the power supply socket on the side of the C-64. A video cable plugs into the back of the C-64 and the other end connects to the RF modulator. Then attach the modulator wire to the VHF antenna on your TV set and tune the TV to channel 3. Turn on the TV. If channel 3 has interference on your set, try changing the channel selector on the back of the C-64 and set the TV to channel 4.

Now press the power switch on the right side of the C-64 to the ON position. In a few seconds the TV screen will come to life and display the following characters:

```
****COMMODORE 64 BASIC V2****  
64K RAM SYSTEM 38911 BYTES FREE  
READY.
```

KEYBOARD CHARACTERS

Commodore 64 Graphics Keyboard

Figure 2.2 illustrates the C-64's keyboard. It is arranged with 62 keys for alphabetic, numeric, special, and graphic characters. The keys select capital letters from A to Z and such special characters as \$ % ' , and (), which are used mainly as special programming symbols. At the right of the keyboard are four function keys that may be programmed to perform special activities.

The keyboard contains most of the C-64 graphic symbols on the front of each key. To select the graphic symbol on the right of the key, you must hold down one of the two shift keys or the shift/lock while you also press the appropriate graphic key. Try holding the shift while you press the Q key. You should get a solid light blue circle on the screen.

To get the graphics on the left of the keys, hold the Commodore key (the one with the Commodore logo beside the shift key) down while you press the graphic key. Try holding the Commodore key while you press the * key. This time you should get a light blue triangle on the screen.



Figure 2.2 Commodore 64 keyboard

At the right of the main keyboard is the Return key. When you make an entry to the C-64, such as a BASIC statement or a response to a question from a program, pressing the Return key will enter that line into memory.

On the left of the keyboard, above the Commodore key, is the RUN/STOP key. This key may be used to STOP a program or to RUN (by holding shift and pressing RUN) a program from tape.

On the left of the keyboard is the CTRL (control) key. When this key is held down, other keyboard features are activated. Try this: Hold down the CTRL and simultaneously press the 9 key. Now type some characters. Notice that they are now dark blue on a light-blue background. This is called a reverse character, which explains the RVS ON on the 9 key. You have activated the reverse character feature of the C-64. Now hold down the CTRL key and press 0 (zero), which turns the RVS OFF. Now type the same characters and notice the difference.

The CTRL key also may be used to change the color of the characters you type. Hold CTRL and press the RED key. First the cursor, that blinking rectangular symbol flashing on the screen, changes to red. Now any characters you type will also be in red. Holding CTRL and pressing another color will change the cursor once again. Use RUN/STOP and the RESTORE keys simultaneously to get things back to normal.

Below the Return key are two cursor (CRSR) control keys to control cursor movement. Using these keys with the shift key will cause the cursor to move up, down, left, or right.