

MICROPOWER SERIES

# BASIC Programming for the VIC-20<sup>®</sup>

Don Cassel



- Includes two dozen programs to help you get your VIC-20 up and running quickly.
- For both cassette and disk drives.

MICROPOWER SERIES

# **BASIC Programming for the VIC-20<sup>®</sup>**

**Don Cassel**  
*Humber College*

**wcb**

Wm. C. Brown Publishers  
Dubuque, Iowa

Micropower Series

Cover photo by Bob Coyle.

Figures 1.1, 2.1, 2.2, 2.3, 2.4, and Appendix F are provided 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—062702

ISBN 0—697—09993—8

2—09993—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 VIC-20<sup>®</sup>**

# Preface

---

**T**his book is an introduction to the BASIC language used on the Commodore VIC-20 microcomputer. If you are a beginner at programming in the BASIC language, this material will provide you with a gradual introduction to BASIC, stressing a hands-on approach with your VIC.

The BASIC used in the book is BASIC 2.0, released by Commodore since 1981. This language is virtually identical with the BASIC used on Commodore PET and CBM computers, with a few slight differences where disk applications are concerned. A major difference between the VIC and PET is the VIC's 22 characters per line versus 40 on the PET. Other differences are the color graphics and music capabilities that the VIC provides.

The first chapter is a very brief introduction to the VIC-20 hardware with BASIC and DOS for the floppy disk. Chapter 2 discusses the use of the keyboard, tape, and floppy disk. If you have made use of the VIC before, you may skip over these chapters or read them through quickly.

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

Chapter 3 ends with six complete programs that may be run on the VIC. In these six programs, two things are emphasized. One is the need for planning a program. The approach taken is to show how the input and output needs to 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 concept of flowcharts is 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 that require the use of the BASIC statements you have just studied.

Subsequent chapters follow the same pattern as chapter 3 but go into more depth in the language as indicated in the 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 develop the program logic.

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

Next we look at graphics, animation, and sound in chapter 8. 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 so that the concepts are made clear.

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

As I mentioned at the beginning, this is an introduction to BASIC programming on the VIC. The book does not pretend to be an exhaustive treatment of programming and there is much more to be learned about the VIC beyond this level. However, I hope you will find this book instructive and helpful as you learn to program, and I trust it will provide the necessary foundation for you to move on to more advanced programming on your VIC-20.

Don Cassel

# Contents

---

*Preface xi*

## 1

*Introduction to the VIC-20 1*

**VIC-20 HARDWARE 1**  
**MEMORY 2**  
**BASIC 3**  
**DOS (DISK OPERATING SYSTEM) 3**  
**REVIEW QUESTIONS 4**

## 2

*Getting Started on the VIC-20 5*

**TURNING ON THE VIC-20 6**  
**KEYBOARD CHARACTERS 6**  
*VIC Graphics Keyboard 6*  
**USING THE CASSETTE TAPE 7**  
*Inserting a Cassette 8*  
*Tape Controls 8*  
*Loading a Program 8*  
*Saving a Program 9*  
**USING THE FLOPPY DISK DRIVE 10**  
*Loading a Program 10*  
*Saving a Program 11*  
*Replacing a Program 11*  
*Directories 11*  
*Formatting a Diskette 11*  
*Backups 12*  
**REVIEW QUESTIONS 12**

# 3

## Elementary BASIC Programming 13

<b>WHAT IS BASIC?</b>	13
<b>CALCULATOR MODE</b>	13
<b>HOW TO USE CURSOR CONTROLS</b>	14
Insert/Delete	15
Clear/Home	15
<b>RUN/STOP</b>	15
<b>LIST COMMAND</b>	16
<b>BASIC NUMBERS</b>	16
Integers	17
Real Numbers	17
Scientific Notation	19
<b>STRINGS</b>	20
<b>VARIABLES</b>	20
<b>MULTIPLE STATEMENTS PER LINE</b>	22
<b>USING COLOR</b>	22
<b>ARITHMETIC STATEMENTS</b>	23
Add (+)	24
Subtract (-)	24
Multiply (*)	24
Divide (/)	24
Exponentiation (↑)	25
<b>HIERARCHY AND PARENTHESES</b>	25
<b>PRINT</b>	25
<b>INPUT</b>	28
<b>GOTO</b>	29
<b>REM</b>	30
<b>SIMPLE CALCULATION PROGRAM</b>	30
<b>FAHRENHEIT—CELSIUS PROGRAM</b>	32
<b>IMPROVING YOUR SOLUTION</b>	33
<b>ELEMENTARY GRAPHICS—SAILBOAT</b>	34
<b>WEIGHTED AVERAGE PROGRAM</b>	34
<b>COMPUTING LOAN PAYMENTS</b>	37
<b>REVIEW QUESTIONS</b>	38

# 4

## Not So Basic BASIC 41

<b>DIM</b>	41
Storing Values in an Array	42
Multidimensional Arrays	43
<b>IF—THEN</b>	44
<b>FOR—NEXT</b>	46
<b>GOSUB—RETURN</b>	48
<b>STOP</b>	49
<b>GENERATING RANDOM NUMBERS</b>	50
<b>NUMBER GUESSING GAME</b>	51
<b>TIME DELAYS</b>	52
<b>IMPROVED WEIGHTED AVERAGE PROGRAM</b>	54
<b>ESTIMATING TRIP COSTS</b>	55
<b>REVIEW QUESTIONS</b>	59

# 5

## *More on Input and Output* 61

- READ—DATA** 61
- CREATING A BAR CHART** 62
- WEIGHTED AVERAGE WITH DATA** 63
- RESTORE** 64
- MORE ABOUT PRINT** 65
  - Cursor Controls* 65
  - TAB** 67
  - SPC** 67
- METRIC CONVERSION PROGRAM** 68
- SIMPLE PAYROLL PROGRAM** 76
- REVIEW QUESTIONS** 79

# 6

## *Advanced BASIC* 81

- GET** 81
- ON—GOSUB** 82
- ON—GOTO** 84
- POKE** 85
- PEEK** 87
- TI AND TI\$ FUNCTIONS** 88
  - TI or TIme Function* 88
  - Reaction Timer Program* 88
  - TI\$ or TIme\$* 89
- DEF FN** 89
- ARITHMETIC FUNCTIONS** 92
  - ABS** 92
  - ATN** 92
  - COS** 92
  - EXP** 93
  - INT** 93
  - LOG** 93
  - RND** 93
    - Creating a Specific Set of Random Numbers* 94
  - SGN** 94
  - SIN** 94
  - SQR** 95
  - TAN** 95
    - Converting Radians to Degrees* 95
- STRING FUNCTIONS** 95
  - ASC** 95
  - CHR\$** 96
  - LEFT\$** 96
  - LEN** 97
  - MID\$** 97
  - RIGHT\$** 99
  - STR\$** 99
  - VAL** 99
    - Concatenation* 100



<b>PLOTTING GRAPHS</b>	100
<b>CONTROLLING DECIMAL POSITIONS</b>	104
<i>Driver Routines</i>	105
<b>CAI CHESS AND PROGRAM GENERALIZATION</b>	107
<i>The Data</i>	107
<i>The Program</i>	109
<b>REVIEW QUESTIONS</b>	114

## 7

### *Interacting with the User of Your Program* 115

<b>USER LEVEL</b>	115
<i>Casual</i>	115
<i>Trained</i>	115
<i>Programming Skills</i>	116
<b>USER DIALOGUES</b>	116
<i>Prompting</i>	116
<b>DEFAULT RESPONSES</b>	118
<b>MENUS</b>	119
<b>MULTILEVEL MENUS</b>	120
<b>FORM FILLING</b>	121
<b>COMMAND LANGUAGES</b>	121
<b>REVIEW QUESTIONS</b>	122

## 8

### *Graphics, Animation, Sound, and Music* 123

<b>GRAPHIC CHARACTER SET</b>	123
<b>REACTION TIMER WITH GRAPHICS</b>	126
<b>USING POKE TO PRODUCE A GRAPHIC</b>	128
<i>Lunar Lander</i>	128
<i>Chessboard</i>	129
<b>ANIMATION</b>	130
<i>Rocket 1</i>	130
<i>Rocket 2</i>	131
<i>Egg Timer</i>	132
<i>Train—Animating Multiple Objects</i>	134
<b>HIGH RESOLUTION ON THE VIC</b>	135
<i>Defining a Set of Chess Pieces</i>	137
<b>GENERATING SOUNDS</b>	138
<i>VIC's Voices</i>	138
<i>Playing a Steady Tone</i>	138
<i>Controlling Note Duration</i>	139
<i>Sliding Tones</i>	140
<b>MUSIC PLAYER</b>	141
<b>THREE-PART HARMONY</b>	142
<b>REVIEW QUESTIONS</b>	144

# 9

## *Tape Files Extend Your Reach* 145

- CONCEPTS** 145
- OPEN AND CLOSE** 146
- PRINT#** 147
- WRITING BUDGET NAMES ON TAPE** 147
- INPUT#** 148
- READING THE BUDGET NAMES TAPE** 149
- HOW TO HANDLE RECORDS WITH MULTIPLE FIELDS** 150
- UPDATING A TAPE FILE** 151
- THE CASH-FLOW PROBLEM** 153
  - New File* 155
  - Old File* 155
  - Enter/Review Data* 155
  - Display Balance* 156
- SUMMARY OF TAPE FILES** 162
- REVIEW QUESTIONS** 163

# 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** 170
- 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

<b>A</b>	<b>BASIC OPERATING COMMANDS</b>	195
	<b>CLR</b>	195
	<b>LIST</b>	195
	<b>LOAD</b>	196
	<b>NEW</b>	196
	<b>RUN</b>	196
	<b>SAVE</b>	196
	<b>VERIFY</b>	197
<b>B</b>	<b>RESERVED WORDS</b>	198
<b>C</b>	<b>ABBREVIATIONS</b>	199
<b>D</b>	<b>DISK ERROR MESSAGES</b>	200
<b>E</b>	<b>VIC ASCII AND PEEK/POKE CODES</b>	201
<b>F</b>	<b>ASCII AND CHR\$ CODES</b>	202

**Index** 205

# 1

---

---

## ***Introduction to the VIC-20***

---

---

**T**his book is about programming the VIC-20 microcomputer using the BASIC language. BASIC is the language used by the majority of microcomputers, so by learning it on your VIC you will be prepared to learn programming for a variety of micros.

Possibly you purchased your VIC-20 to play some of the exciting games available on cartridge or tape, but now you want to explore some of the further capabilities of the VIC. Many people have used the VIC successfully for solutions to problems ranging 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 the BASIC language. But first let's look at the hardware.

### ***VIC-20 HARDWARE***

Figure 1.1 shows the VIC-20 microcomputer with a TV set for the display screen. Also in this figure is the floppy disk drive, printer, and modem used with the Commodore computer. Hidden inside the VIC's case are the electronic components—a microprocessor and memory—necessary for its operation.

To understand the basics of the VIC, it is useful to compare the computer to parts of the human body. Of course, the similarity is only coincidental. For example, when you want to learn something new, you often read a book—such as you are doing right now. The new information passes through your eyes and becomes input to the brain, where it is stored in memory. Similarly, new information can be entered into the computer through the keyboard and then stored temporarily in the VIC's electronic memory or storage.

After you have read something you may think about it by processing it mentally in your conscious brain. 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 then be displayed on the TV screen, which is the VIC's output device.

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



Figure 1.1 VIC-20 hardware

If your brain is anything like mine you will not likely remember everything you have read. One solution to this is to make notes by writing or typing. This process is 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 used to record or store information outside of the computer. One form of output 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 it is 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 VIC'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 VIC's circuitry. Programs or data may be read or written into RAM. Normally only one program at a time goes into memory and when a new program is read from tape or disk it will replace the previous program in memory.

Memory in the VIC starts at 5K or 5,000 storage positions and may be expanded up to 32K. Each storage position may hold only one character (i.e., a letter or a number), so a 32K VIC can store 32,000 characters, or in computer lingo, bytes.<sup>1</sup> The memory size of your computer will determine the size of program it is capable of running. If you have 5K of memory, the computer will not have room for a 8K or 16K program. The VIC's memory may be expanded by plugging in a memory expansion module.

1. Technically, 1K bytes is 1024 bytes, so 32K is actually 32768 bytes.

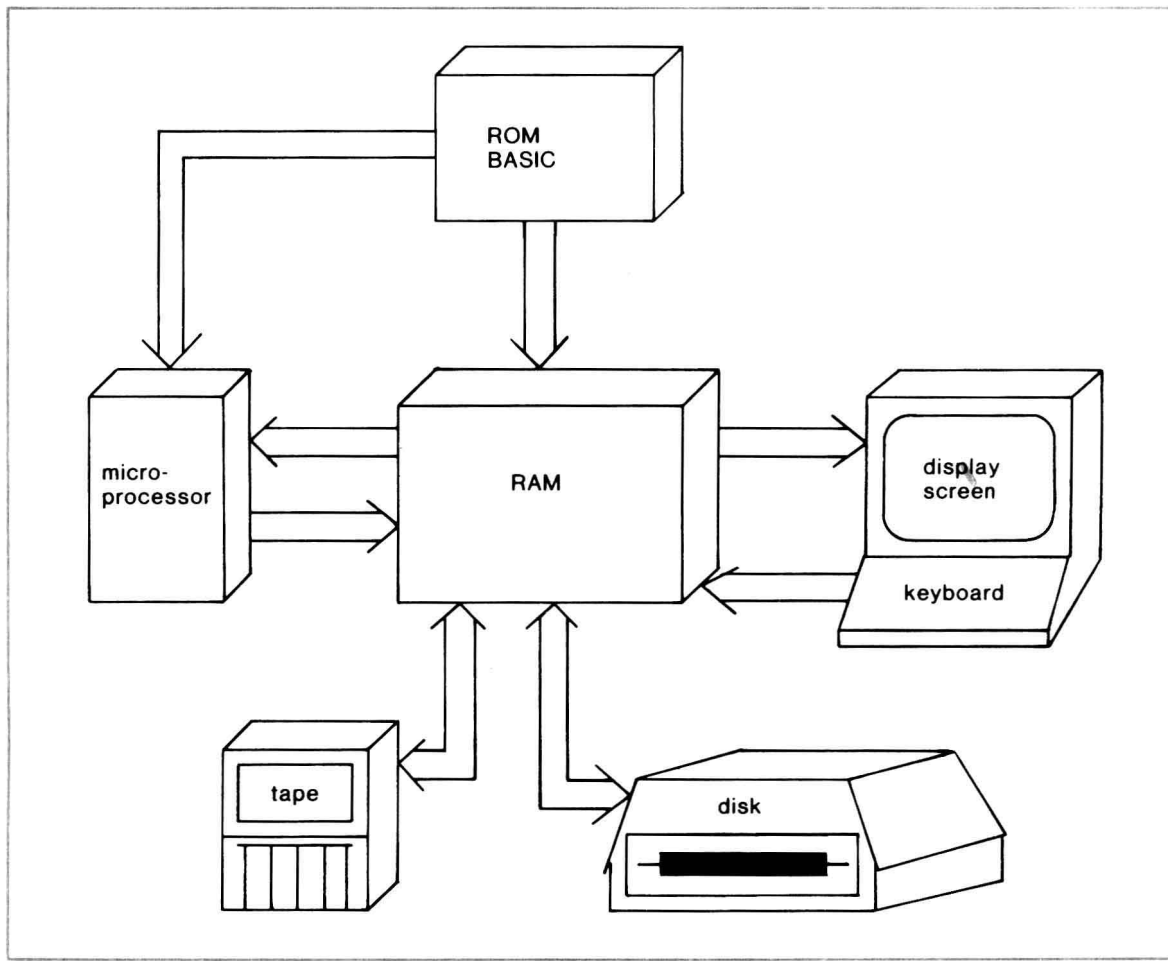


Figure 1.2 Components of a VIC-20 system

Also shown in figure 1.2 is a component called ROM (for Read Only Memory). As the name suggests, ROM may only be *read* 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 VIC's ROM and is used by the computer to run your BASIC programs.

## BASIC

BASIC is the primary language for programming the VIC-20. It is through BASIC that you give instructions to the computer to solve a particular problem. BASIC, like human languages, has rules for its use 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 to learn of all computer languages.

## DOS (DISK OPERATING SYSTEM)

DOS is the program in the floppy disk drive that controls the reading and writing of disk files. Without the use of DOS, working with disk would be terribly complex. Although we will introduce the use of DOS at appropriate times, most of our discussion will make use of 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 VIC-20 microcomputer.
2. Compare the concepts of input, process, and output to the human body. Can you think of any other analogy to the computer?
3. What is hardware? Give some examples.
4. What is software? What purpose does it perform in the microcomputer?
5. What is meant by the term RAM? What sizes of RAM are common for the VIC?
6. Discuss the function of BASIC on the VIC.
7. What device must your computer have if you will need to use DOS?

# 2

---

---

## **Getting Started on the VIC-20**

---

---

**I**f you have already used a VIC, you might wish to skip ahead to chapter 3. Otherwise you should read this chapter to learn the basic operation of the VIC-20.

The VIC (figure 2.1) comes with the fullsize graphics keyboard and is available with 5K to 32K of RAM. A TV is used for the display, which shows 22 characters per line. The most basic configuration for the VIC is with a cassette tape, while the floppy disk and/or printer are potential additions.



**Figure 2.1** VIC-20 microcomputer



## TURNING ON THE VIC-20

First be sure the VIC's power supply is plugged into a wall outlet and the other end of the power supply cord is plugged into the power supply socket on the side of the VIC. A video cable plugs into the back of the VIC, 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. Some modulators will let you select either channel 3 or 4 to get the best picture. Turn on the TV.

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

```
**** CBM BASIC V2 ****
3583 BYTES FREE
READY.
```

## KEYBOARD CHARACTERS

### VIC Graphics Keyboard

The keyboard for the VIC is illustrated in figure 2.2. This keyboard is arranged with 62 keys for alphabetic, numeric, special, and graphic characters. These keys select uppercase alphabetic letters from A to Z and special characters such as \$ % ' ' , ( ) that are mainly used as special programming symbols. At the right of the keyboard are four function keys that may be programmed to perform special duties.

The keyboard contains most of the VIC graphic symbols on the front of each key. To select the graphic symbol on the right of the key, it is necessary to 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 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 blue triangle on the screen.



Figure 2.2 VIC keyboard

At the right of the main keyboard is a RETURN key. When an entry is made to the VIC, such as a BASIC statement or a response to a question from a program, the RETURN key is pressed to enter that line into memory.

On the left of the keyboard, above the Commodore key, is a 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.