

POCKET COMPUTING POWER!

For the TRS-80 PC2
and Sharp PC1500

**BASIC Programming
and Applications**

**Henry Mullish
Richard Kestenbaum**

POCKET COMPUTING POWER!

Henry Mullish
Richard Kestenbaum

Harper & Row, Publishers New York
Cambridge, Hagerstown, Philadelphia, San Francisco
London, Mexico City, São Paulo, Sydney

The names of all computer programs and computers included herein are registered trademarks of their makers

Sponsoring Editor: John Willig

Project Editor: Mary E. Kennedy

Designer: C. Linda Dingle

Production: Marion Palen, Delia Tedoff

Compositor: Com-Com, a division of Haddon Craftsmen, Inc.

Cover design by: Steve Sullivan

All rights reserved. Printed in the United States of America. No part of this book may be used or reproduced in any manner whatsoever without written permission, except in the case of brief quotations embodied in critical articles and reviews. For information address Harper & Row, Publishers, Inc., 10 East 53d Street, New York, NY 10022.

Copyright © 1984 by Harper & Row Publishers, Inc.

Library of Congress Cataloging in Publication Data

Mullish, Henry.

Pocket computing power.

Portale computers—Programming. I Kestenbaum,

Richard. II Title

QA76.6.M83 1984 001.64'2 83-18524

ISBN 0-06-044659-5

POCKET COMPUTING POWER!

For Izy and Frieda

—Richard Kestenbaum

For Inna

—Henry Mullish

Contents

1	The Evolution of the Pocket Computer	1
2	Manual Mode	9
3	Reserve Mode	31
4	Some Fundamental Concepts of Pocket Computer BASIC	35
5	Getting a Little More Involved with Your Pocket Computer	53
6	The FOR/NEXT Loop	71
7	Arrays	77
8	Character String Manipulation	93
9	Subroutines	109
10	Debugging Programs	121
11	The Cassette Commands	125
12	The Text-Printing Commands	129
13	Printer Graphics	135
14	Some Other Pocket Computers	145
	Answers to Review Questions	147
	Appendix	147
	Glossary	153
	Index	171

The Evolution of the Pocket Computer

Welcome aboard! In this chapter you will be presented with a bird's-eye view of the exciting history of computers. In particular you will learn about:

- the cost of the main-frame computers of a generation ago as compared with today's prices for equally powerful, miniature machines
- the transition from vacuum tubes to transistor technology
- the development of the minicomputer
- integrated circuitry and the development of the "chip"
- the development of the pocket calculator
- the development of high-level languages
- translators and compilers
- the FORTRAN language
- the COBOL language
- the development of the BASIC language
- the birth of the pocket computer
- the successor to the original pocket computer
- the national concern for academic excellence
- the pocket computer as a possible solution to the decline of mathematics and science in education

A BRIEF OVERVIEW OF THE HISTORY OF COMPUTERS

In order to appreciate the technological breakthrough that the pocket computer represents, it is only necessary to know that it is a more powerful computer than most universities or corporate offices had a mere 20 years ago. Moreover, those

massive computers that became available just a generation ago cost in the neighborhood of several hundred thousand dollars each. The cost of the electrical power alone that those computers consumed in one day could buy several dozen pocket computers at today's prices. From the time the transistor edged out the vacuum tube in computer circuitry, we began to see a considerable reduction in the size of computers, leading to faster and more reliable computing ability. Additionally, transistors are not as fragile as vacuum tubes, are considerably more compact, consume far less electricity, and are cheaper to produce. As a direct result of the more modern transistor technology came the evolution of a broad range of minicomputers. These minicomputers cost only a fraction of the price of earlier large-scale computers, are less expensive to maintain, and, as a result, have found their way into businesses that would otherwise not have been able to take advantage of the new technology.

The transistor eventually gave way to the integrated circuit, or "chip," a sliver of silicon on which thousands of electronic elements are etched. The chip spawned development of pocket calculators, also a revolution of a sort in its own way. Pocket calculator technology was actually a spin-off from work done in the U.S. space program in which it was necessary to have lightweight but complex, reliable, electronic circuitry. In early 1971 a pocket calculator cost in the region of \$400 and was capable of performing only the four most basic arithmetic operations: addition, subtraction, multiplication, and division. But by the end of the year the price of calculators had plummeted due to innovative marketing practices and improved manufacturing processes. Subsequent to 1971, an estimated one million pocket calculators had been sold at a fraction of the original cost.

By 1975, more sophisticated pocket calculators had become very popular and both the cost of their production and their prices were greatly reduced. By that time many programmable models had been developed and met eager acceptance by a burgeoning market of computer and calculator enthusiasts. The first such pocket programmable calculator, Hewlett-Packard's HP-65, was considered such a unique and worthwhile technological achievement that in the summer of that year, an HP-65 programmable pocket calculator was taken aboard an American *Apollo* spacecraft as a backup for the on-board computers needed to calculate two critical midcourse correction maneuvers just prior to the docking of the U.S. *Apollo* and Russian *Soyuz* spacecrafts. In the event that the on-board computer failed, the crew would have had to rely entirely upon the HP-65, since, at that stage of the mission, the spacecraft would not have

been in communication with any of the monitoring earth stations. Scientists at the National Aeronautics and Space Administration (NASA) had previously written programs of up to 1,000 steps for the HP-65 to execute in the event its computational ability was needed to complete critical maneuvers.

Meanwhile, back on earth, technology was developed that made the task of communicating with a computer considerably easier for most people, hastening the day when computers would be as common as wristwatches. This feat was accomplished through the development of computer languages that more closely resemble the English language. These languages have become known as "high-level" languages. Programs written in these high-level languages are translated automatically by a translator, or "compiler," into individual instructions in a computer-comprehensible language, so that the computer can execute the steps and produce the desired results.

The first high-level language was one designed for engineers and scientists called FORTRAN, an acronym for *FOR*mula *TRAN*slation, written by IBM in 1957. The creation of FORTRAN was a gargantuan piece of work, and it is estimated that 100 man-years of effort were required to produce the first version. Although the language has undergone considerable improvement since its first release, the original version of FORTRAN is a milestone in the development of the man-to-machine relationship.

In view of the enormous enthusiasm with which FORTRAN was received, it did not take long before general business and industry saw the need for a high-level language of their own. So in 1960 a computer language named COBOL was completed. The name COBOL is also an acronym; it stands for *CO*mmon *B*usiness *O*riented *L*anguage, and COBOL became the standard language of programs written for business applications.

By 1956 Professors John Kemeny and Tom Kurtz of Dartmouth College realized that every college student would eventually need to be conversant in computer science. What they felt was necessary was a language that was fairly simple to learn, versatile in its operational capability, and interactive. They therefore set out to develop BASIC, an acronym too, standing for *B*eginners *A*ll-purpose *S*ymbolic *I*nstruction *C*ode. The language they wrote was extremely successful, and the computer manufacturing industry was quick to see how easily it can be learned. Complete novices were able to write fairly substantial programs in a matter of only a few hours. Industry therefore zeroed in on BASIC as both a teaching and an operational language with the result that every personal computer manufactured today supports BASIC.

More recently, the development of even more compact “Large Scale Integrated” chips (LSI technology) has spawned a new industry of smaller computers that have come to be known as microcomputers. The reduction in both the price and the physical size of computers has made them adaptable for consumer use, and the name “microcomputer” has become almost synonymous with the phrase personal or home computer. One can now buy a powerful home computer “off the shelf,” as if buying a television—and what is more remarkable, they are both about the same price. Not only are personal computers a useful and innovative product, but coupled with the proliferation of retail computer stores, computer manufacturers have been able to bring the small computer into the hands of tens of thousands of people, many of them non-professionals.

In mid-1980 the Radio Shack TRS-80 Pocket Computer was introduced to the market at \$249. It was unique in that it was the first pocket-sized computer capable of being programmed in BASIC. It measures only $5/8$ by $2\ 3/4$ by $6\ 7/8$ inches, weighs a mere six ounces, and can be carried in a shirt pocket without any difficulty. Even when the power is switched off the pocket computer retains programs and data in its memory. In other words, the memory is “nonvolatile.” The TRS-80 Pocket Computer is identical to the Sharp PC-1211 (see figure 1.1).

In mid-1982, Radio Shack introduced the highly improved TRS-80 PC-2 to succeed their pocket computer, now referred to as the PC-1. The PC-2 is all but identical to the Sharp PC-1500 (see figure 1.2). The PC-2 is destined to fill a great need and become extremely popular among students, financiers, accountants, scientists, and just about anyone who has the need to solve problems quickly and accurately. Despite its small size, the PC-2 has a large liquid crystal display that is $4\ 1/2$ inches wide, enabling 26 characters to be displayed at a time. The version of BASIC that is available on the PC-2 is a close relative of the level I BASIC available on the TRS-80 home computer, except that the pocket version has been enriched with extra features. The programmable memory is 2.6K bytes of RAM. This means that there are about 2,600 locations of memory that may be randomly accessed by the programmer (*K* is the letter that signifies one thousand and RAM stands for random access memory). The PC-2 is capable of expanding its RAM capacity to either 6.6K or even 10.6K by means of plug-in modules that may be purchased at any computer store. The PC-2 also has 16K bytes of ROM (read-only memory), which provides a version of Extended BASIC that is often not seen even on much larger and more

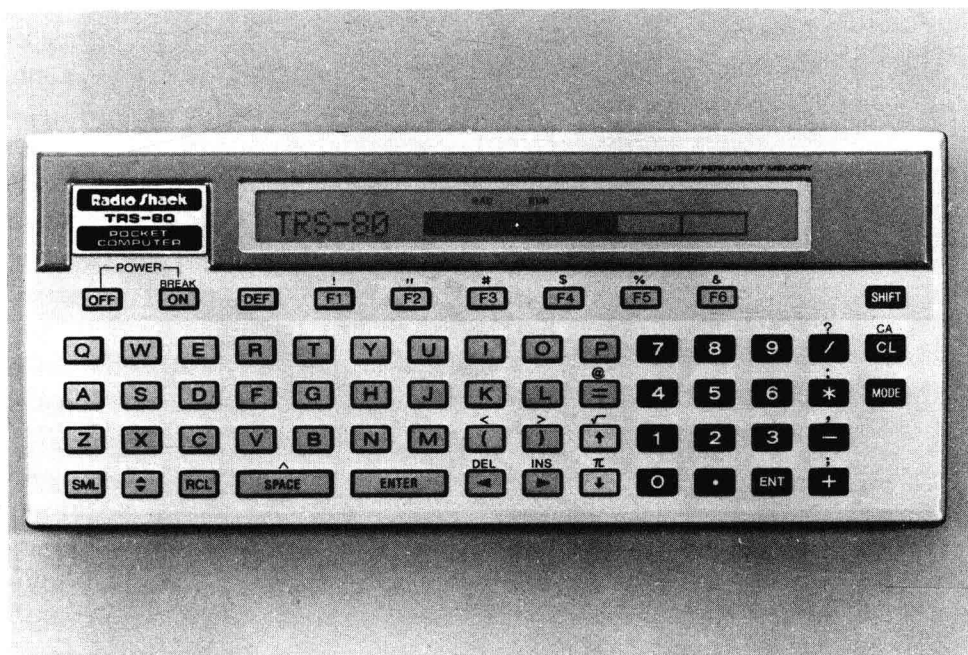


FIG. 1.1 Radio Shack TRS-80 Pocket Computer



FIG. 1.1 Sharp PC-1211 Pocket Computer

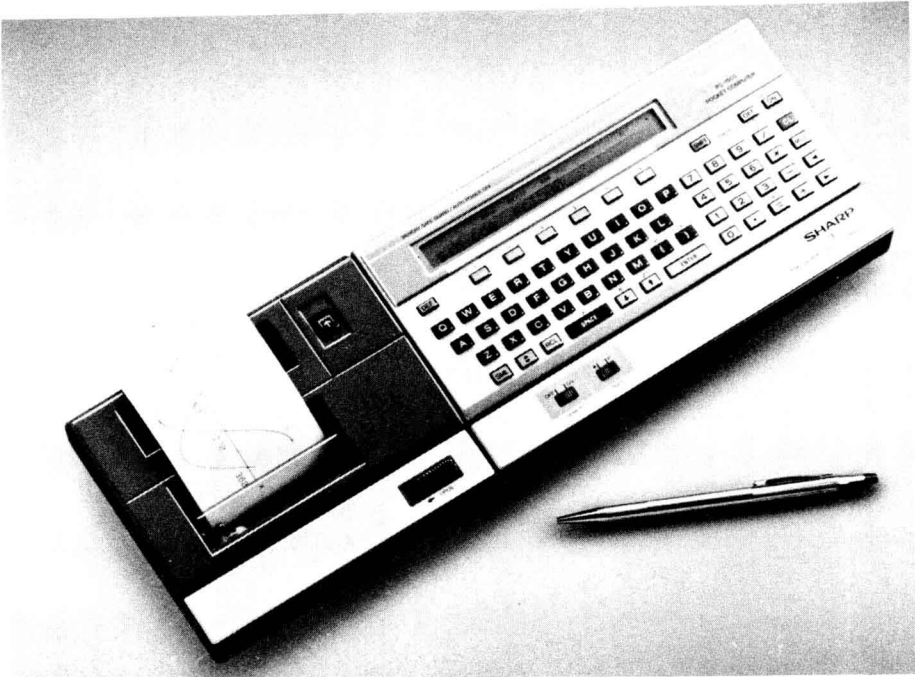


FIG. 1.2 Sharp PC-1500 Pocket Computer

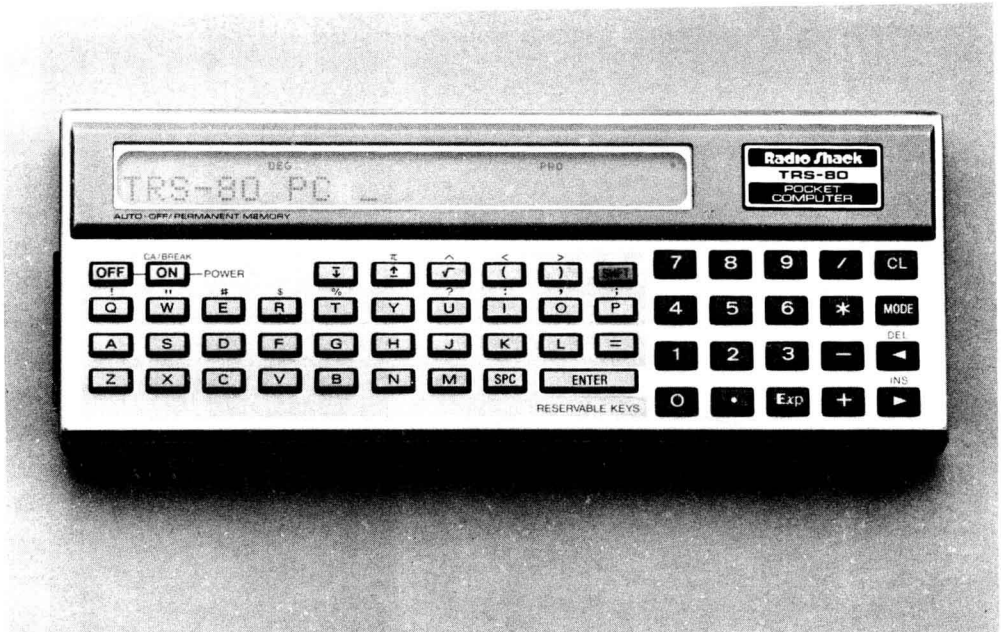


FIG. 1.2 Radio Shack TRS-80 PC-2

expensive desktop models. It can handle arrays, two-dimensional matrices, character strings, program chaining and a host of other advanced features.

As you read through this text, you will be amazed how easy it is to operate these computers and command them to do interesting and functionally useful chores for you. Indeed, they have so much capacity for such incredibly small machines that they might eventually change the definition of a personal computer to one that does not have to be tied to a television screen but may be carried in one's coat pocket.

Pocket computers can be of use not only to professionals but as an educational tool as well. In a 240-page report written in 1980 by the National Science Foundation and the Education Department, the commitment of the United States to academic excellence and achievement in the sciences was severely criticized. The report, titled "Science and Engineering Education for the 1980's and Beyond," was written in response to a memo by then-President Carter in February of that year, inquiring how adequately science and mathematics education were preparing students to tackle the nation's anticipated industrial problems. No doubt the impetus for such concerns stemmed from the decreasing rates of productivity growth and the ability of foreign competitors to undercut American producers even in their own domestic market. The report's conclusions may be summed up by the following two points:

1. Any technical lead the U.S. has is being diminished by rapidly advancing foreign competitors.
2. The current trend toward scientific and technological illiteracy, unless reversed, will result in decisions of great scientific and technological magnitude (crucial to our international position of leadership) being made in a national environment of ignorance and misunderstanding.

According to this sobering report, more students than ever before are dropping out of science and mathematics courses, and the trend shows no signs of abating.

Perhaps one way to stimulate the study of computer science is to introduce the subject of computer programming at the high school level or even earlier. Although a small number of high schools across the country already do provide some access to computers, it is generally true that the vast majority of elementary schools and high schools provide scant exposure to this important discipline. Further compounding the problem is the historically high cost of scientific equipment for use in educational laboratories.

The pocket computer can be a useful tool in working against the disturbing trends just noted. Low in cost, it could serve well as a student's introduction to the fascinating art of computer programming. As teachers of computer science will readily attest to, learning computer programming is, for the great majority of students, sheer fun and entertainment. There is something about programming that just catches the imagination and opens the mind to vast new possibilities, perhaps never before contemplated. In a laboratory equipped with pocket computers, a whole school of students could have direct access to a computer without having to wait in line. Each student could have "hands-on" experience, a most desirable, satisfying, and stimulating way to learn what science and computers are all about.

Whether you are a student, a housewife, a doctor, a baker, a financier, a teacher or a Wall Street banker, the pocket computer is a tool that can be of the greatest use to you. However, you are hereby cautioned that programming a computer can be most addictive—and there is no known cure.

In order to avoid unnecessary duplication and to keep the size of this book to manageable proportions, the body of the text is focused on the TRS-80 PC-2/Sharp PC-1500 pocket computers. For those readers who own the TRS-80 PC-1/Sharp PC-1211 models, chapter 14 describes in detail the differences between the various models and how they are used.

If this is your first encounter with a computer, we wish you every success. Welcome to the wonderful world of pocket computer programming!

Manual Mode

The chances are that you are just itching to write your first program and want to learn how to do so in a hurry. Although such enthusiasm is readily understandable and is to be encouraged, it is a good idea to delay this move until we have covered some basic (excuse the pun!) material. After all, when you are learning to drive an automobile you don't head immediately for the center of town. In this chapter you will be introduced to the computer in its manual mode. You will come to appreciate what an incredible calculator it can be. In particular, you will be learning about:

- the layout of the keyboard of the PC-2
- RUN and PROgram mode
- elementary calculations
- the rules of mathematical precedence
- the use of parentheses
- a mnemonic to remember the rules of precedence
- how to correct a wrongly keyed in value
- finding square roots using the ($\sqrt{\quad}$) symbol and SQR function
- finding the n th root
- the trigonometric functions (SIN, COS, TAN, etc.)
- logarithms (common and natural LOG, LN)
- the exponential function (EXP)
- the constant π and its equivalent, PI
- the INTeger function
- the ABSolute function
- the SiGNum function

- angular measure (DMS, degrees, minutes, and seconds)
- angular modes (DEG, RAD, and GRAD)
- scientific notation
- the six relational operators
- using the computer's memory
- memory mathematics
- memories for storing character strings
- automatically detected errors
- deleting and writing characters
- some editing techniques

In this chapter we will explore the pocket computer's capabilities, not as a computer but rather as a calculator. The pocket computer can, if you wish, function as a superior calculator unlike any other you have ever used. Once familiar with the vast array of functions that the pocket computer offers, you are ready to examine the machine's potential as a remarkable, personable, portable, versatile, economical computer. A little patience at this juncture will be handsomely rewarded, for we are dealing with a complex instrument. A considerable amount of attention to detail will be necessary. As a matter of fact, you will soon learn that to become a good pocket computer programmer, you will have to develop (if you don't already possess) a whole complex of personal characteristics. The list is rather long and includes a little of the following: patience, versatility, creativeness, a sense of logic, dogged persistence, and a capacity to devote meticulous attention to detail without losing sight of the overall problem.

First, let us take a look at the keyboard of the pocket computer (see figure 2.1).

You will notice that along the top edge is a "liquid crystal" display capable of holding 26 characters. Liquid crystal displays have become very popular in recent years because they use very little electrical current. They display in reflected light—the light in the immediate environment—rather than from an internal lamp. The saving in electricity means that the pocket computer may be powered by a battery the size of a shirt button. (It is said—perhaps with tongue in cheek—that way back in the early 1950s when the first mainframe computer was built at the University of Pennsylvania, each time the computer was switched on, all the lights in Philadelphia dimmed momentarily.) Characters are displayed via a 7×156 dot matrix that permits each character to be seen in an unusually clear and distinct fashion. Certain characters are accessed

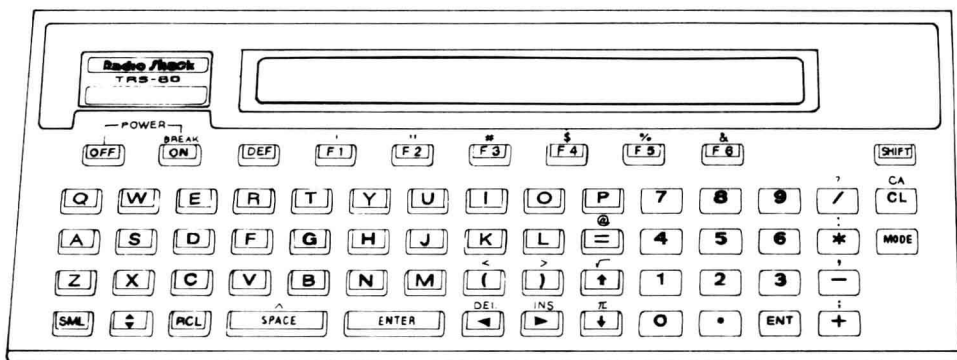


Figure 2.1

by pressing two keys in succession. For example, the comma, which is uppercase (—), is accessed by first pressing the yellow shift key, SHIFT, followed by the (—) key. Thus we have a rather full alphabetic typewriter-style keyboard of the familiar “QWERTY” type.

On the right-hand side of the keyboard are the numeric keys 0 through 9, plus the four arithmetic keys: “/” for division, “*” for multiplication, “—” for subtraction, and “+” for addition. The key marked CL clears the display, while CA, Clear All, clears all the arithmetic registers that are used by the computer whenever any arithmetic is done. Let us now try our hand at some calculations. Before we can, however, we must first switch the computer on by pressing the key on the top right of the keyboard, the one marked ON. What happens after the ON key is pressed depends what was done with the computer the last time it was switched on. The dot that appears on the extreme upper right of the display indicates that the batteries are operating, and a prompter with the sign “>” appears on the extreme left. If the word PRO appears, this means that the computer is in its PROgram mode. If the word DEG appears on the screen, that means that it was last used in DEGree mode and any angles that are displayed will be represented as degrees. It is possible that a Roman