

ADAM™

Archive.Org

Downloaded from:

<http://adamarchive.org>

ADAM

USER'S GUIDE



A. J. Bell
E. Q. Bell

ADAM

User's Guide

Publishing Director: David Culverwell
Acquisitions Editor: Terrell Anderson
Production Editor/Text Design: Karen Zack
Art Director/Cover Design: Don Sellers
Assistant Art Director: Bernard Vervin
Manufacturing Director: John Komsa

Typesetter: Alexander Typesetting, Indianapolis, IN
Printer: Fairfield Graphics, Fairfield, PA
Typefaces: Folio (text), Palatino (display), OCR-B (programs)

ADAM

User's Guide

A. J. Bell & E. Q. Bell

Brady Communications Company, Inc., Bowie, MD 20715
A Prentice-Hall Publishing Company

ADAM User's Guide

Copyright © 1984 by Brady Communications Company, Inc. All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the publisher. For information, address Brady Communications Company, Inc., Bowie, Maryland 20715.

Library of Congress Cataloging in Publication Data

Bell, A. J., 1951-
ADAM user's guide.

Includes index.

I. Adam (Computer) I. Bell E. Q., 1944-
II. Title. III. Title: A.D.A.M. user's guide.
QA76.8.A153B45 1984 001.64 84-3036

ISBN 0-89303-300-6

Prentice-Hall International, Inc., London
Prentice-Hall Canada, Inc., Scarborough, Ontario
Prentice-Hall of Australia, Pty., Ltd., Sydney
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, Petone, New Zealand
Editora Prentice-Hall Do Brasil LTDA., Rio de Janeiro

Printed in the United States of America

84 85 86 87 88 89 90 91 92 93 94 1 2 3 4 5 6 7 8 9 10

Contents

Part I Your ADAM Computer 1

- 1 **What You've Got** 3
Congratulations on your smart purchase! For one low price, you just bought an entire computer system—a keyboard, Memory Console, printer, game controllers, a game, SmartBASIC, and some ancillary items. Find out all about these in Chapter 1.
- 2 **Hooking it Up** 13
If you've never set up a computer before, you'll appreciate the step-by-step directions and illustrations for hooking up your ADAM Family Computer System.
- 3 **Using Your ADAM Computer** 25
Find out how this computer can be used as an electronic typewriter or a word processor . . . or to write programs, play games, or run ready-to-use software such as home management and educational programs.
- 4 **Maintaining the Computer** 35
Changing ribbons and printwheels will be your biggest "maintenance" tasks. You'll also need to know how to do some smaller tasks—like cleaning the screen—to keep your computer in good operating condition.

Part II Electronic Typing Lessons 43

- 1 **Using ADAM as a Typewriter** 45
If you can type, you can use ADAM as a typewriter just by turning it on. To make typing easier, ADAM has Smart Keys which guide you through the steps for changing margins, tab stops, and line spacing. Four short lessons introduce you to these features, as you type two letters or anything else of your choice.

Part III SmartWRITER Word Processing 53

- 1 **Typing Your First Letter** 55
In just 10 short lessons, you'll be an expert with SmartWRITER Word Processing. You'll learn how to change

SmartWRITER's color combinations and sound, correct errors, use the "oops" commands, store text, print it, and get it back.

- 2 **Getting Organized—File Maintenance** 73
With SmartWRITER you file what you type on digital data packs instead of folders. And to keep your files organized, you can copy any of the text you filed, give the file a new name, or throw out text when it's no longer needed. Complete two lessons and you'll know how to use ADAM's file maintenance procedures.
- 3 **Typing a Telephone List** 79
Want to type a list of all your friends and relatives and their phone numbers? Or a chart or other report information in columns? Then go through the five lessons in this chapter and learn how to use the Moving Window Format, change margins, tabs, and line spacing.
- 4 **Using Your Printing Options** 87
Do you want page numbers printed on each page? Do you want to start numbering pages with 1? Or, perhaps you want to leave your pages unnumbered. Find out what your options are in just one short lesson.
- 5 **Typing Equations, Formulas, and Footnotes** 91
Fancy features like subscripts and superscripts are no problem for ADAM. Scientific charts and assignments with footnotes can be done on ADAM. One short lesson is all it takes to use these features.
- 6 **Revising Text** 93
Learn how easy it is to move text or copy it to different places. You can also search for words or phrases and then replace them with something else. In just two lessons you'll be able to use these features to revise any report, letter, or other document.
- 7 **Applications—Typing Letters, Reports, Lists and Resumes** 97
Sample letters, resumes and other documents are shown here, along with all the settings and selections you need. This "brings together" all the information you've learned for everyday personal and business needs.

Part IV Introducing SmartBASIC 103

- 1 **Getting Started (PRINT and GOTO)** 105
Want to write your own programs? ADAM's special SmartBASIC makes writing programs fun and easy. In

this chapter, you'll learn the two basic rules for writing your own programs. Then, in just two lessons, you will write and "run" your first SmartBASIC program.

- 2 **Making Changes (INPUT and REM)** 113
Learn to use two new keywords in this chapter: REM and INPUT. You'll also find out how to make corrections when typing programs. Finally, you'll write and run another SmartBASIC program.
- 3 **Adding it Up (+, -, *, /)** 125
ADAM is great with numbers! Here you'll learn to use ADAM's special talents to perform addition, subtraction, multiplication, and division problems.
- 4 **Storing Your Program (SAVE and LOAD)** 137
Now that you've learned to write programs, you'll want to save your work and show it to your friends! This chapter explains how to save and load programs stored on ADAM's special digital data packs.
- 5 **Color a Rainbow (GR, COLOR =, PLOT, VLIN, and HVLIN)** 145
Low resolution graphics can add excitement to your SmartBASIC programs. Here you'll learn how to use 15 different colors and draw shapes using three important keywords: PLOT, VLIN, and HVLIN.
- 6 **Try These** 153
You're now ready to begin writing your own programs. Instead of lessons in this chapter, there are sample programs for you to type and use. If you want, you can save these programs and begin building your own library of programs!

Part V **Doing More with SmartBASIC** 163

- 1 **Pick a Number** 165
If you are interested in writing games for your ADAM computer, you will discover that one of the most interesting aspects of SmartBASIC is the ability it gives you to select a random number. This chapter explains how to use RND and INT to select random numbers in a program.
- 2 **Comparison Shopping (IF/THEN)** 173
You ask yourself questions when shopping. Asking questions (for example, "Is Brand X larger or smaller than Brand Y?") helps you compare values and then take action based on your information. Your ADAM

computer can do the same thing using IF/THEN statements. Here you'll learn to use IF/THEN statements in your programs. In addition, you'll enter two game programs that you can save for future use.

- 3 **Take a Trip (GOSUB and RETURN)** 183
Subroutines are like little trips for your computer. In this chapter, you'll learn to use GOSUB and RETURN in programs. You'll also get a chance to enter and save two more programs.
- 4 **Around & Around (FOR/NEXT Loops)** 193
FOR/NEXT loops can save you time and energy when writing programs. Here you'll learn to use FOR/NEXT loops along with STEP.
- 5 **More Rainbows (HGR, HGR2, HCOLOR=, and H PLOT)** 201
Low resolution graphics are just the beginning! With high resolution graphics you can "draw" even more detailed pictures and designs. This chapter explains the difference between HGR and HGR2. You'll also learn how to use H PLOT in high resolution graphics.
- 6 **More Programs to Try** 209
You're becoming a SmartBASIC expert! This chapter contains more programs for you to enter and save. One of the programs in this chapter is designed to be your new cooking assistant. It will offer recipe suggestions based on the main ingredient of your meal. Believe it or not, another program in this chapter will make your ADAM computer write a poem! By the time you finish this chapter, your library of SmartBASIC programs will have really grown!

Part VI **Appendix** 219

- 1 **SmartWRITER Reference Guide** 221
This reference guide lists each SmartWRITER command alphabetically, describes what each command does, and explains how to use each.
- 2 **Glossary of SmartWRITER Terms** 245
The 40 most important SmartWRITER word processing terms are listed here.
- 3 **SmartBASIC Reference Guide** 253
The most frequently used BASIC keywords are listed alphabetically and their functions are described.

- 4 **SmartBASIC Error Messages** 305
Each SmartBASIC error message is listed here, along with an explanation.
- 5 **Glossary of Computer Terms** 311
The 60 most frequently used computer “buzzwords” are described in plain English.
- 6 **Index** 317

Limits of Liability and Disclaimer of Warranty

The authors and publisher of this book have used their best efforts in preparing this book and the programs contained in it. These efforts include the development, research, and testing of the programs to determine their effectiveness. The authors and publisher make no warranty of any kind, expressed or implied, with regard to these programs, the text, or the documentation contained in this book. The authors and publisher shall not be liable in any event for claims of incidental or consequential damages in connection with or arising out of the furnishing, performance, or use of the text or the programs. The programs contained in this book are intended for the use of the original purchaser-user.

Note to Authors

Do you have a manuscript or a software program related to personal computers? Do you have an idea for developing such a project? If so, we would like to hear from you. The Brady Co. produces a complete range of books and applications software for the personal computer market. We invite you to write to David Culverwell, Publishing Director, Brady Communications Company, Inc., Bowie, MD 20715.

Acknowledgments

This book could not have been written without the assistance and encouragement of many people. We would like to express our thanks to the following people for helping make this work a reality:

At Brady—David Culverwell, Terry Anderson, Karen Zack, and Susan Westberg.

At Coleco—Harvey Johnson, Linda Lightfoot, Paul Klim, and Andy Fried.

Last but not least—Patti Sostoch, Allison Sandberg, and our understanding families and friends.

AJB
EQB

Preface

Welcome!

Computers are everywhere—in grocery stores, banks, factories, businesses, airports—and now, even in your home. With the ADAM Family Computer System, you can do many things that only a short time ago required an enormous investment. You can use it to type school and business reports and letters, to do home management and accounting tasks, and to write your own computer programs. And when you're ready for some fun, you can use it to play the best video games in town.

If you think all this sounds like a lot, you're right. But you can find almost everything you need to know about the ADAM Family Computer System in this book. Even better, everything is explained in plain English, with lots of pictures and examples.

In Part I, you'll learn how to set up your computer and what it can be used for. When you're ready to use ADAM as an electronic typewriter, just turn to Part II. Or go directly to Part III and find out how to use the built-in word processing features. Want to create your own programs in SmartBASIC? Open to Parts IV and V for some of the clearest directions you'll ever see. And if all you want is reference materials, then look at Part VI.

So welcome to your wonderful ADAM Family Computer System. And have fun!

Part I
Your ADAM Computer

Everything You Need In One Package!



1

What You've Got

Whether you've bought the ADAM Family Computer System or Expansion Module #3, which connects to a ColecoVision Video Game System, you now have an entire computer system. In fact, you really have more than a computer system—there are programs that can be used on your system, a switch box to connect the computer to your TV, and two game controllers. Items such as these must usually be purchased in addition to your computer. But not with ADAM. You get everything for one low price.

What comes in the ADAM computer system package? You can see everything that comes with your computer by looking at the picture on the opposite page. Make sure you have all these items by placing a check beside each item as you unpack your package.

- keyboard
- Memory Console with one digital data drive and built-in SmartWRITER word processing
- SmartWRITER letter-quality printer
- controller holder (which attaches to keyboard)
- pica-10 daisy wheel printer element—already inserted in printer
- multistrike carbon ribbon cartridge—already inserted in printer
- gray coiled cord for keyboard/Memory Console connection
- black video connecting cable
- 3-to-2 prong plug adapter (not provided in Canada)
- SmartBASIC digital data pack
- BUCK ROGERS Planet of Zoom by SEGA super game pack
- blank digital data pack
- 4 manuals (Set-Up Manual, Word Processing Manual, SmartBASIC Manual, and a Game Play Manual)
- Easy Reference Word Processing Instruction Guide
- 2 game controllers (only in the ADAM Family Computer System)
- 2 gray coiled cords for game controller/Memory Console connection (only in the ADAM Family Computer System)
- TV/computer switch box (only in the ADAM Family Computer System)
- system interlock tray (only in Expansion Module #3)

In addition to these items, you will also need a color TV screen or color video monitor.

Your TV, along with the keyboard, Memory Console, and printer are the four essential parts of your computer. If you want to know more about these parts, learn how they provide you with computer power, and learn some basic computer words, then continue reading this chapter. However, if you want to get your computer hooked up right away, turn to Part 1, Chapter 2, "Hooking It Up." You can come back to this section later, if you want more information about computers and how they work.

What's a Computer?

Your ADAM Family Computer System (or Expansion Module #3) uses a keyboard, a Memory Console (with digital data pack drive), a printer, and a TV screen or monitor. These items are shown in Figure 1-1-2.

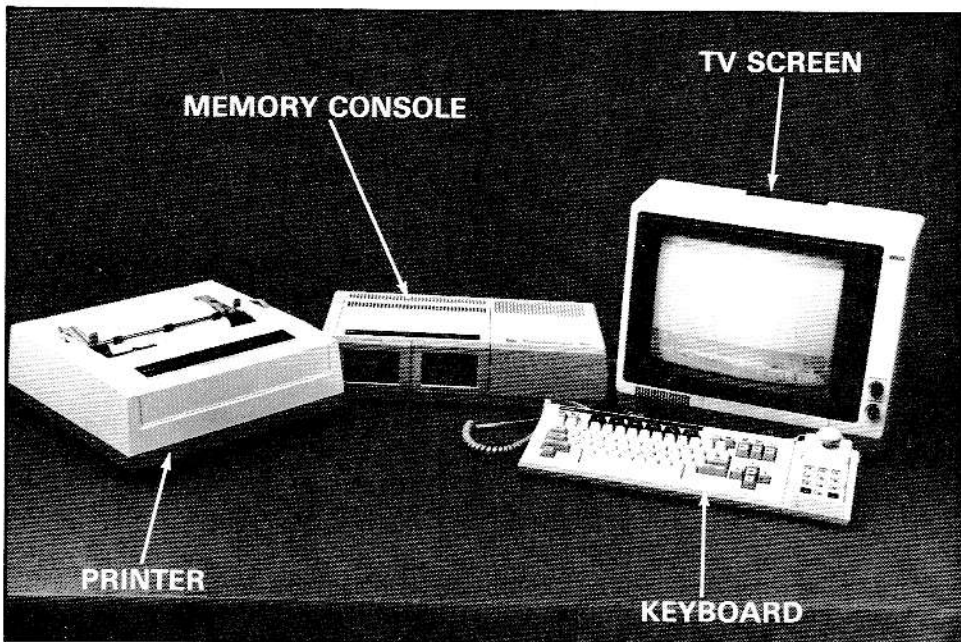


Figure 1-1-2

To use your computer, you type or "enter" information using the keyboard. The information you type appears immediately on the TV screen. You can print this information at any time. It can also be stored on a digital data pack, which is held in the digital data pack drive, for future use.

Everything you type and every instruction you give your computer goes IN to the Memory Console. So what you type is called "input." An input device

is the keyboard. Input can also be provided by game controllers, game cartridges, or information that is stored on digital data packs.

The Memory Console then carries out your instructions and acts upon, or "processes," your information.

Finally, the Memory Console sends the processed information to the right place—the TV screen, the printer, or a digital data pack. Since this information is sent OUT from the Memory Console, computer users call it "output."

In this way, your ADAM Family Computer System performs the 3 functions which are essential to any computer: input, processing, and output.

In computer terms, you can describe the parts of your ADAM computer like this:

- The keyboard, game controllers, game cartridges, cartridge slot, digital data packs, and digital data pack drive are "input" devices.
- The Memory Console is your "processor."
- The TV screen or monitor, printer, digital data packs, and digital data pack drive are "output" devices.

Sometimes the keyboard, screen, printer, digital data packs, and digital data pack drives are referred to as "I/O" (eye-oh) devices. This is just another way of saying "input/output" devices. Some of these devices, such as digital data packs (and drives), are used as both input and output devices.

On some computers, you will also find I/O labelled where input and output devices can be connected to the computer. These places are called "I/O ports."

Another way to describe the keyboard, screen, printer, and drives on your ADAM is to call them computer "peripherals." They are called peripherals because they are outside or away from the box containing the microprocessor which is the central part of the computer. (On some other computers, the keyboard, screen, drives, and microprocessor are built together in one piece of equipment and only the printer is a "peripheral.")

The Keyboard

The ADAM keyboard is a professional quality typewriter keyboard with 75 keys. You'll find it easy to use this keyboard because keys which perform similar types of functions are grouped together and color coded. You can see this by looking at Figure 1-1-3.

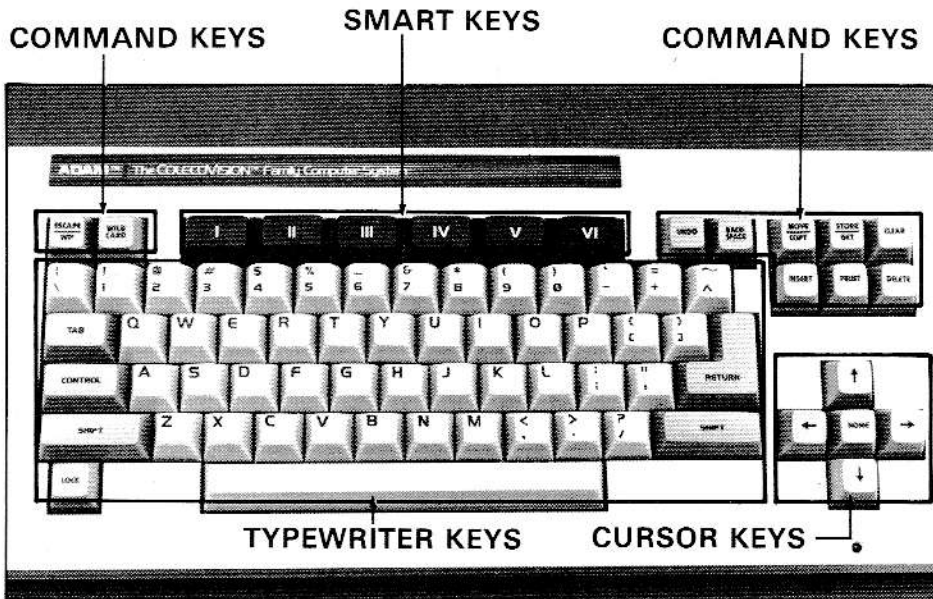


Figure 1-1-3

There are four types of keys on the ADAM keyboard.

1. **Typewriter keys:** These are the same keys that you will find on any standard typewriter. They include letters, numbers, and special symbol keys in the same location as on a standard typewriter. All the white keys on the keyboard are typewriter keys. Five of the gray keys—LOCK, SHIFT, RETURN, TAB, and BACKSPACE—are also typewriter keys.
2. **Cursor Control Keys:** There are a total of five Cursor Control Keys. All of them are located at the lower right of the keyboard and are gray. Four of these keys have arrows on them, and the fifth is labelled HOME. You use Cursor Control Keys to scroll or move text on the screen, to point to a particular file which is listed in the File Directory, and to point to text you want to correct.
3. **Command Keys:** The 10 gray Command Keys are used to tell the computer what you want to do, such as "print" a letter.
 - Four of these keys—ESCAPE/WP, WILD CARD, UNDO, and CONTROL—are located above or to the left of the typing keys.
 - Six more Command Keys—MOVE/COPY, STORE/GET, CLEAR, INSERT, PRINT, and DELETE—are located at the upper right of the keyboard.
4. **Smart Keys:** Smart Keys are the six black keys at the top of the keyboard, labeled I, II, III, IV, V, and VI. They are called "Smart Keys"

because they can be used to do many different things. For example, when you are using word processing, the labels for these keys appear on the bottom of the screen. The labels change automatically to help guide you through procedures.

TV Screen or Monitor

The next component of your computer system, a color TV or color monitor, must be supplied by you. After you hook up your computer to a color television or monitor, you will be able to use a full range of colors with everything you do—from games to word processing. This means you won't be working with white or green letters on a black background, as you must on many computers.

The appearance of the screen will change depending on what you are doing. For example, when you use SmartWRITER Word Processing, you will see ruler lines and special labels on the screen. Information on the appearance of the screen is included in the chapters on typing, word processing, and programming.

The Memory Console

The Memory Console looks a lot like an audio cassette player without the controls. There are some differences between the Memory Console that comes in the ADAM Family Computer System package and the one in Expansion Module #3, which is used with the ColecoVision Video Game System. The latter is shown in Figure 1-1-4.

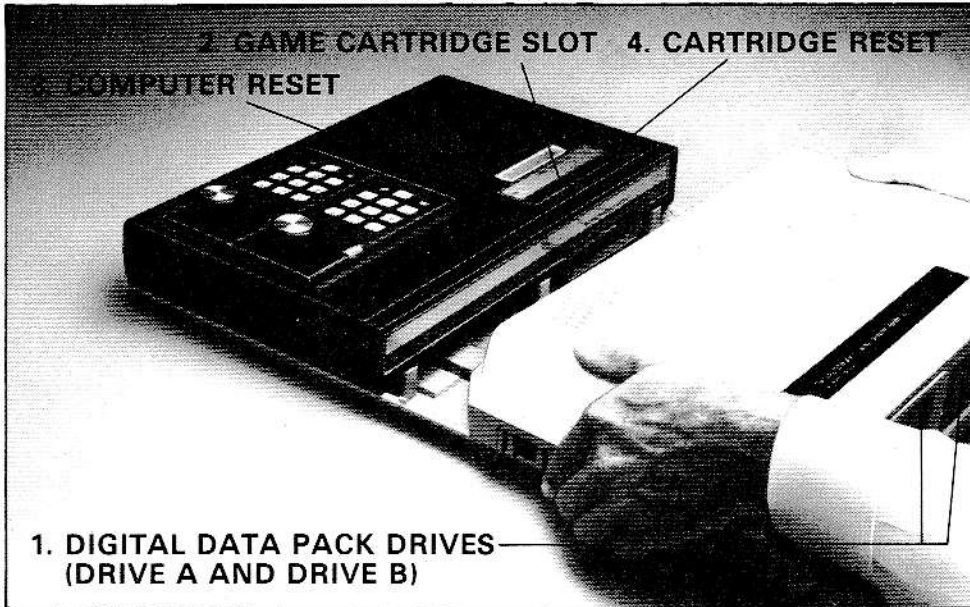


Figure 1-1-4

There are four important items on the Memory Console.

1. **Digital data pack drives:** One digital data pack drive comes with each system. This is called "DRIVE A" and is behind the left drive door. You can purchase a second drive, DRIVE B, at any time and install it behind the right drive door. Drives hold digital data packs, which look a lot like audio cassettes, and are used to store letters, programs, and games.
2. **Game cartridge slot:** Since your ADAM computer can use game cartridges, you can play any of the ColecoVision games which come on cartridges. If you've been using a ColecoVision Video Game System, notice that the game slot is in your game system and not in the Memory Console.
3. **Computer reset:** When you're using ADAM as an electronic typewriter and want to erase what you've typed, just pull the computer reset slide switch toward you.
4. **Cartridge reset:** If you want to play a game cartridge, turn off your computer. Then, insert a game cartridge in the cartridge slot. Finally, pull the cartridge reset slide switch toward you.

The outside of the Memory Console holds more than the drives and the reset switches. It holds and protects the "brain" of your computer. Information from the input devices is sent here, processed, and then sent to the output devices.

Some of the main parts of the brain of a computer are illustrated in Figure 1-1-5.

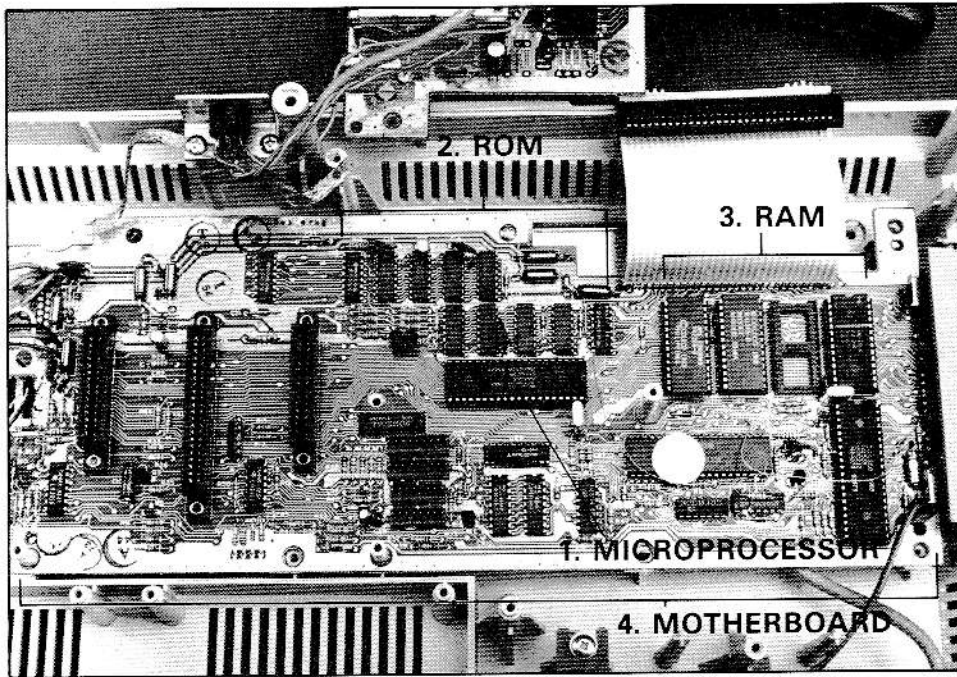


Figure 1-1-5

1. **Microprocessor:** The computer brain is a "microprocessor." This is a small "chip" which contains hundreds of electronic components, each of which is finer than a hair. To assure accuracy when a chip is made, it must be manufactured in a dust-free, sterile environment. The chip is then enclosed in a protective cover. The microprocessor is the part of a computer which actually processes information.
2. **ROM:** ROM is a computer term which means "read-only memory." ROM refers to a place in the computer's memory where instructions are permanently stored. Instructions which are essential to a computer are "burned" into ROM. (ADAM's word processing program, for example, is stored in ROM.) The computer can then "read" instructions from ROM whenever those instructions are needed. In a sense, ROM is like a book. You can read it, but you can't change it.
3. **RAM:** RAM is another computer term, and refers to a place in the computer's memory where you can store and change information and instructions. RAM means "random-access memory." Information which you type or which comes from digital data packs or game cartridges is stored in

RAM. Instructions which are stored in both ROM and RAM can be accessed directly, rather than sequentially, by a computer. RAM, in fact, could be more accurately referred to as "RWM" or "read-write memory" because information can be read from it or written to it.

Information gets into RAM from input devices and copies are sent to output devices. When the computer is turned off, all the information in RAM is lost. RAM indicates temporary memory, whereas ROM is permanent memory.

If you purchased the ADAM Family Computer System, your Memory Console has 80 K memory. This means it can hold 80,000-plus characters at one time. (The "K" stands for the word "thousand.")

On the other hand, if you purchased the Expansion Module #3, your Memory Console has 64 K. It can hold 64,000-plus characters. Additional information can be stored in the Video Game component.

4. **Motherboard:** The last item pictured is the "motherboard." All the elements of the computer's brain are assembled on this board. It contains printed circuits which connect these parts. The term "motherboard" comes from the early days of radio when people built their radio sets on their mothers' bread boards.

Printer

Your ADAM computer comes with a letter-quality printer. This means that letters printed with this printer will look like they have been typed on a typewriter. An example of words printed on a letter-quality printer is shown on the top in Figure 1-1-6.

Another type of printer is a "dot matrix" printer. With this printer, letters look like they have been printed by making a series of dots in the shape of letters. An example of words printed on a dot matrix printer is shown on the bottom in Figure 1-1-6.

These words have been printed
with a letter-quality printer.

These words have been printed
with a dot matrix printer.

Figure 1-1-6

Dot matrix printers are often used to print out form letters, long financial reports, and rough drafts because they can print much faster than letter-quality printers. Dot matrix printers are also used when fancy graphs or pie charts are printed. On the other hand, when people want material to look professionally typed, they select letter-quality printers.

Outside, the SmartWRITER printer looks a lot like a standard typewriter without keys. It even has many of the same parts as a typewriter.

The main printer parts are shown in the picture below.

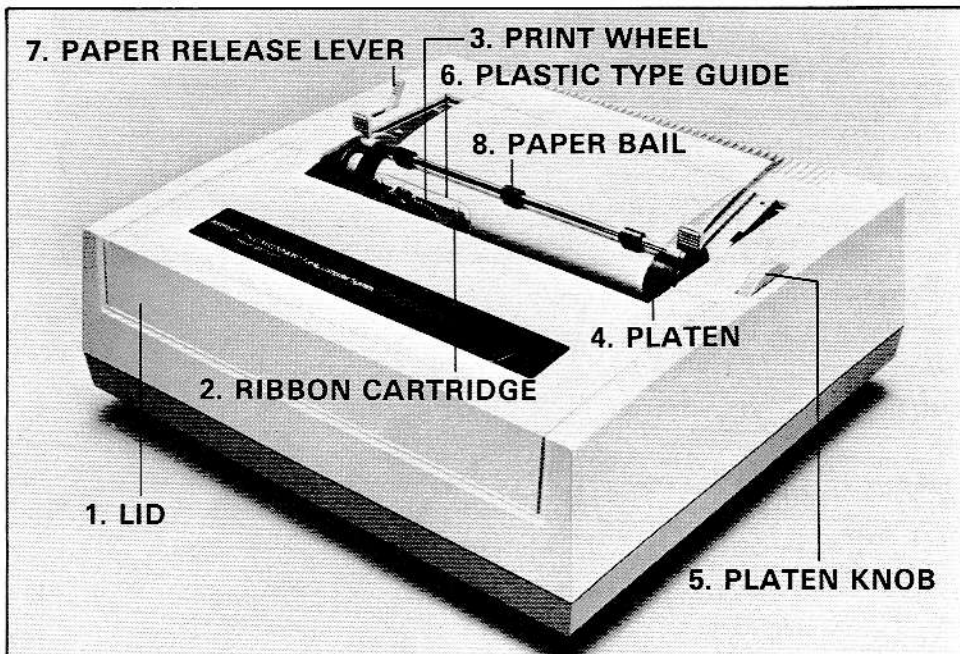


Figure 1-1-7

Like a typewriter, a printer has a lid (1) which protects the printing element and ribbon. The lid must be removed whenever you want to change the ribbon cartridge (2) or the print wheel (3). This helps keep fingers away from the print wheel when it is moving.

Whenever you put paper into the printer, you slide it behind and around the platen (4). You use the platen knob (5) to wind the paper around the platen till it reaches the plastic typeguide (6). If the paper is not straight, you can pull the paper release lever (7) forward and straighten the paper. Then, push the release lever forward. Make sure the paper stays behind the paper bail (8) when printing your letters, reports, etc.



2 Hooking It Up

Both the ADAM Family Computer System and Expansion Module #3 have been designed so that all the parts, or components, can be connected quickly and easily. However, your computer system should only be assembled by an adult.

FEDERAL COMMUNICATIONS COMMISSION REQUIRES THE FOLLOWING:

Do not simultaneously hook up more than one TV set to your computer. Do not use any more flat twin-lead from switch box to TV set than is supplied with the switch box. Do not connect ColecoVision or computer cable directly to any TV antenna or cable TV outlet. Avoid attaching loose wires to your TV antenna terminals when using your computer. Doing any of the above may cause interference to nearby TV sets and is against Federal Communication Commission regulations.

Hooking up the ADAM Family Computer System

The procedure for "hooking up" the ADAM Family Computer System begins here. For instructions on connecting Expansion Module #3 to the ColecoVision Video Game System, turn to page 20.

Complete the following seven steps to assemble your ADAM Family Computer System. The illustrations show you exactly how to do each step.

1. Make sure your TV is off.
2. Plug the coiled keyboard cord (the one not attached to a game controller) into the port on the back of the keyboard. Then, plug the other end of the keyboard cord into the left side of the Memory Console. (See Figure 1-2-2.)

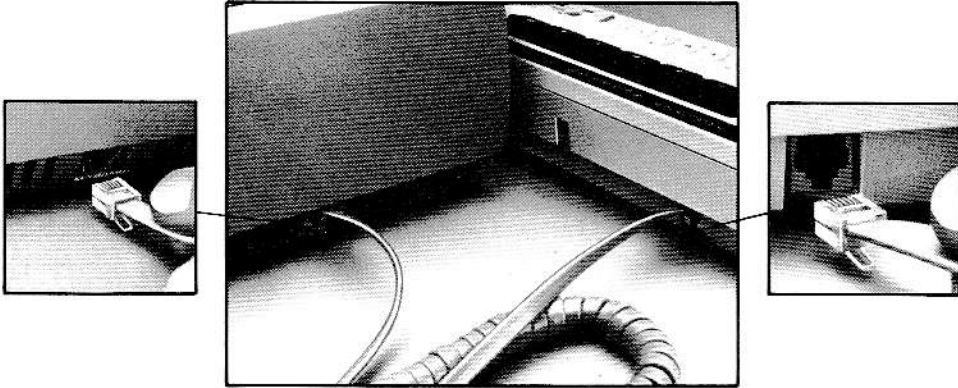


Figure 1-2-2

3. Plug the game controller cord into the right side of the Memory Console. (A second game controller can also be plugged into the Memory Console.) (See Figure 1-2-3.)

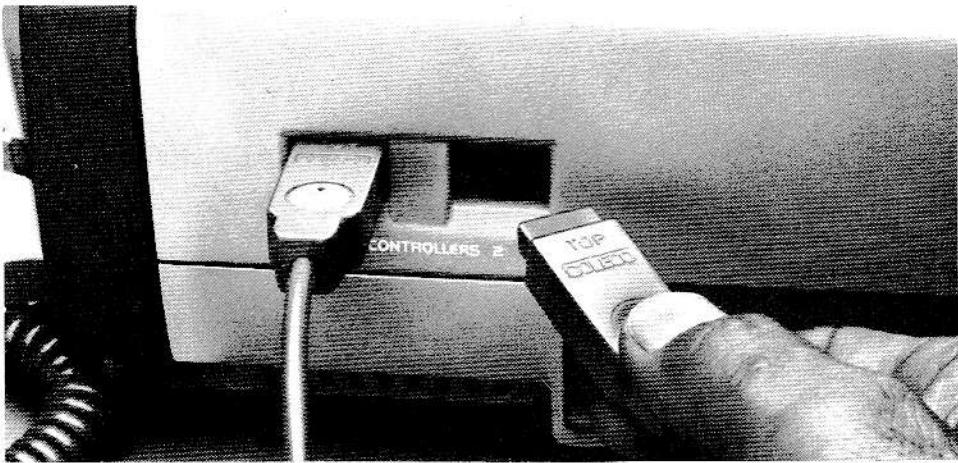


Figure 1-2-3

4. Attach the controller holder to the right side of the keyboard. (See Figure 1-2-4.) Then snap the game controller connected to the right side of the Memory Console into the controller holder. (See Figure 1-2-5.)

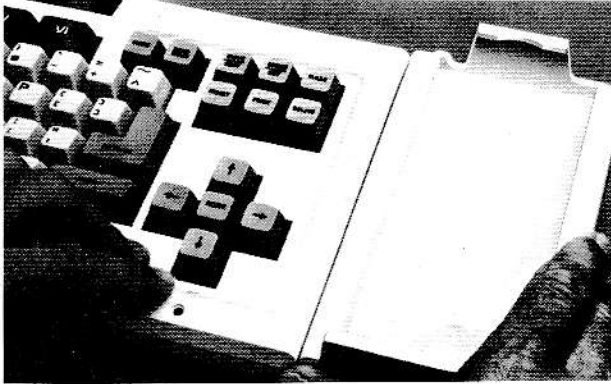


Figure 1-2-4



Figure 1-2-5

5. Plug the SmartWRITER printer cord into the left side of the Memory Console. (See Figure 1-2-6.)
6. As you may have noticed, an antenna switch box is provided with your computer. This switch box allows you to select either your ADAM computer or regular television viewing. Once this switch box is installed, it should not be removed.

Depending on the type of TV antenna, you will use one of four different methods to attach the switch box to your TV. Look at the pictures which follow to determine which method you will need to use. Attach the switch box to your antenna by completing the appropriate instructions.

If Flat "Twin-Lead" Antenna Wire Used

- a. Disconnect the flat twin-lead VHF antenna leads from your TV set.
- b. Connect the flat twin-lead from the switch box to the terminals on the back of the TV set marked "VHF."

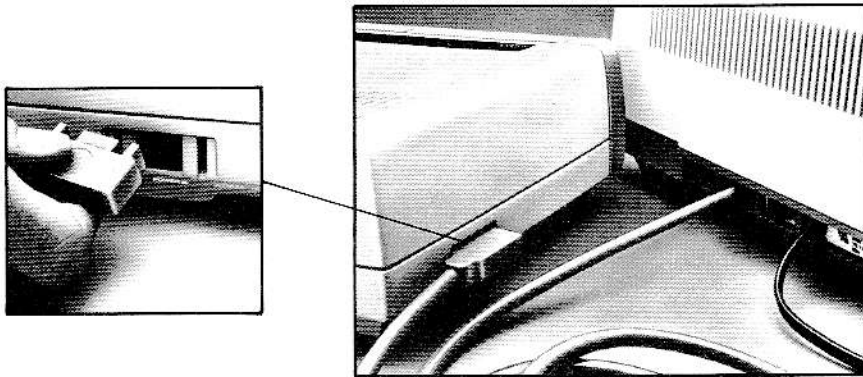
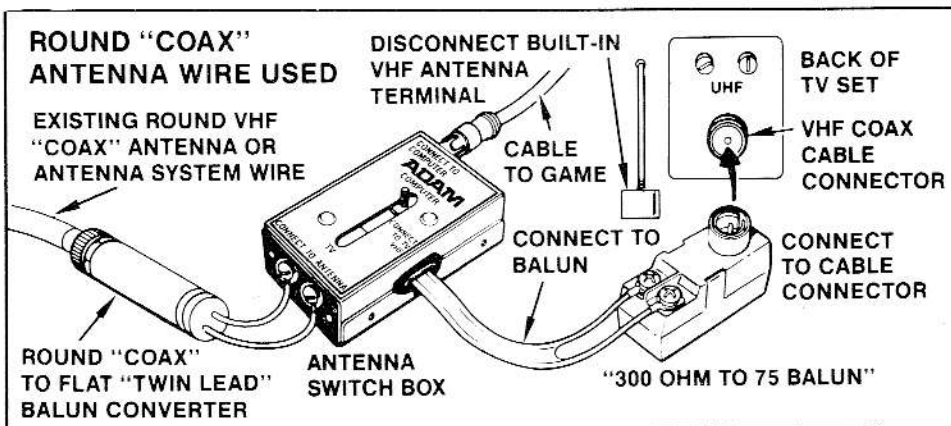


Figure 1-2-6

- c. Connect the flat twin-lead (300 OHM) TV antenna leads which you have just disconnected from the outdoor antenna, cable TV, or rabbit ears to the switch box marked "Antenna."

If Round "Coax" Antenna or Cable TV Used

- a. If you have cable TV or your TV has a round coaxial cable connector only, follow steps a through c as listed above. However, you may need to purchase a "75 OHM to 300 OHM Balun" matching transformer. This small part is available at a television service department or from your local electronics store and will permit proper connection to the switch box. (See Figure 1-2-7.)

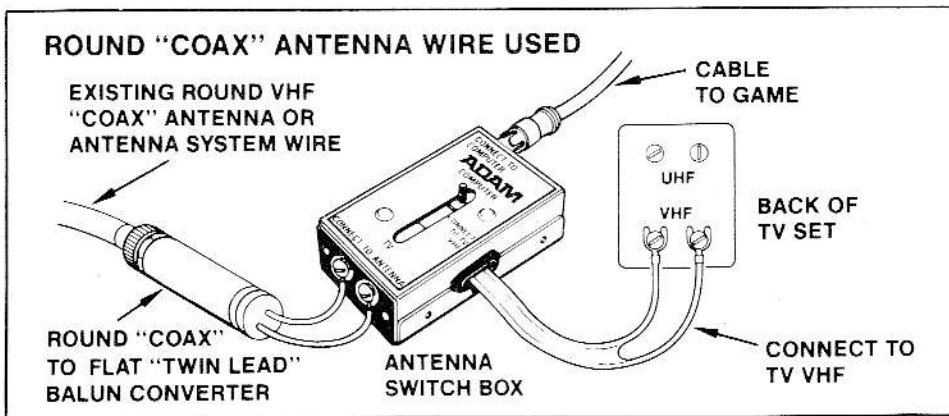


FOR ROUND "COAX" ANTENNA OR CABLE USE

Figure 1-2-7

If Outdoor Antenna Used

- a. Disconnect the internal VHF antenna plug from your TV set.
- b. Connect the flat twin-lead from the switch box to screw terminals on your "75 OHM to 300 OHM Balun." (NOTE: This balun is normally supplied with your TV set, or it can be purchased at your local television dealer, TV service department, or electronics store.)
- c. Plug the balun into the VHF antenna terminal on your TV.
- d. Connect the flat-lead (300 OHM) TV antenna leads from the outdoor antenna to switch box terminals marked "Antenna." (See Figure 1-2-8.)

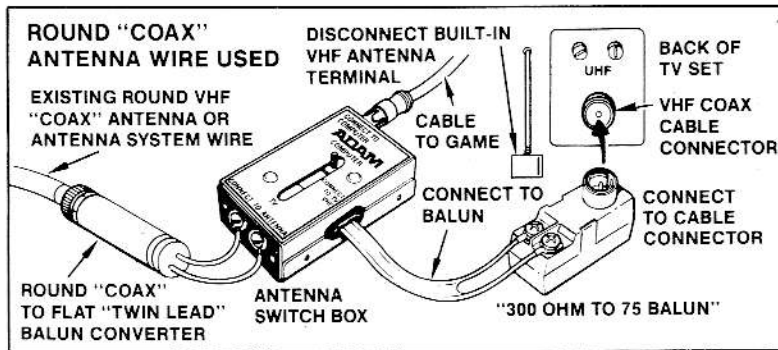


IF ROUND "COAX" ANTENNA OR CABLE TV USED FOR OUTDOOR ANTENNA

Figure 1-2-8

If Cable TV Used

- a. If you have cable TV, follow steps a and b as listed above to connect your "75 OHM to 300 OHM Balun" to the switch box.
- b. Connect existing round VHF coaxial antenna or antenna system wire through a coaxial balun to the switch box screw terminals marked "Antenna." (See Figure 1-2-9.)



FOR ROUND "COAX" ANTENNA OR CABLE USE

Figure 1-2-9

7. Plug the connecting cable to the antenna switch box where the box is labeled "connect to computer." Then, on the back of the Memory Console, plug the other end of the cable into the port labeled "TV" if a TV is used, or to the port labeled "monitor" if a video monitor is used.

NOTE: If you are connecting ADAM to a monitor, you will need a second cable (besides the cable that comes with ADAM) with either a 7-pin or 5-pin DIN connector at one end and a phono jack at the other end. Plug the DIN connector into the port marked "AUX VIDEO" on the back of the Memory Console. Plug the phono jack into the audio input port on the monitor. If you are creating your own DIN/phono jack cable, use the following information:

- Pin 1 to center of phono jack.
- Pin 2 to center of phono jack.

(See Figure 1-2-10.)

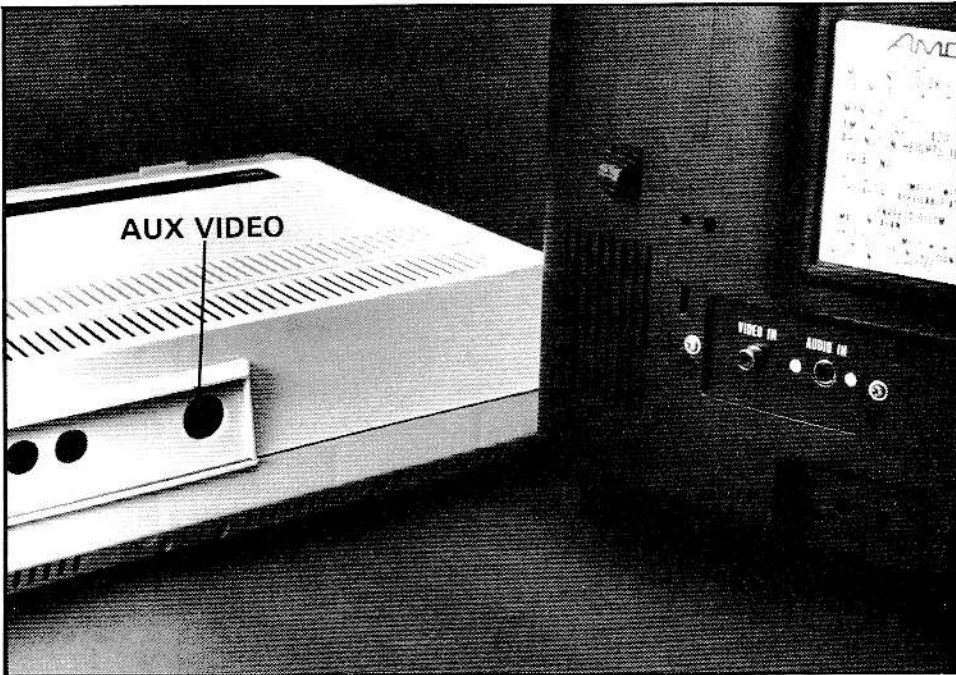


Figure 1-2-10

8. Plug the power cord from the back of the printer into a 110/120 volt electrical outlet. Or, use the 3-to-2 prong plug adapter, and plug the printer into a standard outlet as shown below. (See Figure 1-2-11.)

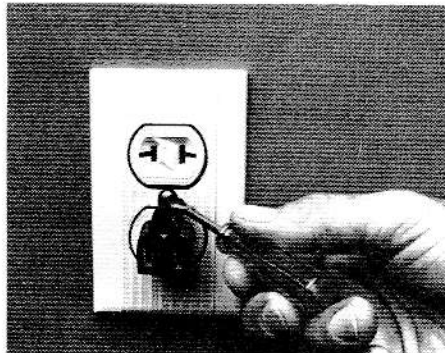


Figure 1-2-11

NOTE: Remember, once you have installed the switch box that came with your ADAM Family Computer System, it never has to be removed. Just move the switch back to "TV" for regular viewing.

IMPORTANT: DO NOT TURN ON THE COMPUTER SYSTEM YET. Turn to the next chapter for those directions.

Hooking Up Expansion Module #3 to Your ColecoVision Video Game System

Expansion Module #3 turns your ColecoVision Video Game System into an exciting computer that the whole family can use. Now you can play arcade-style games, use SmartWRITER Word Processing, balance the family budget, write your own programs, and much more.

Complete the following steps only if you have purchased Expansion Module #3 to add to your ColecoVision Video Game System. The corresponding illustrations show you exactly how to do each step.

1. Make sure the TV is off and game cartridges have been removed from the game system.
2. Make sure the ColecoVision ON/OFF switch is in the off position.
3. Unplug the game system power source from the outlet. Then unplug the power source from the game system, and put the power source away. (See Figure 1-2-12.) The power source is not used with Expansion Module #3.

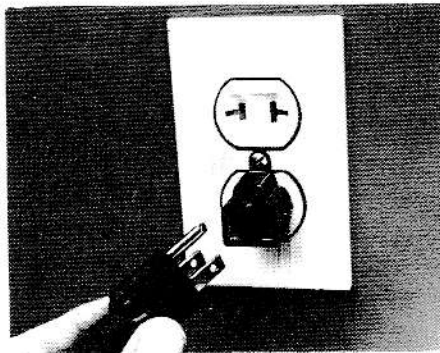


Figure 1-2-12

4. Place the ColecoVision Video Game System on the plastic tray which comes with Expansion Module #3. Make sure the plastic clips on the front and back of the tray snap into place as shown. (See Figure 1-2-13.)

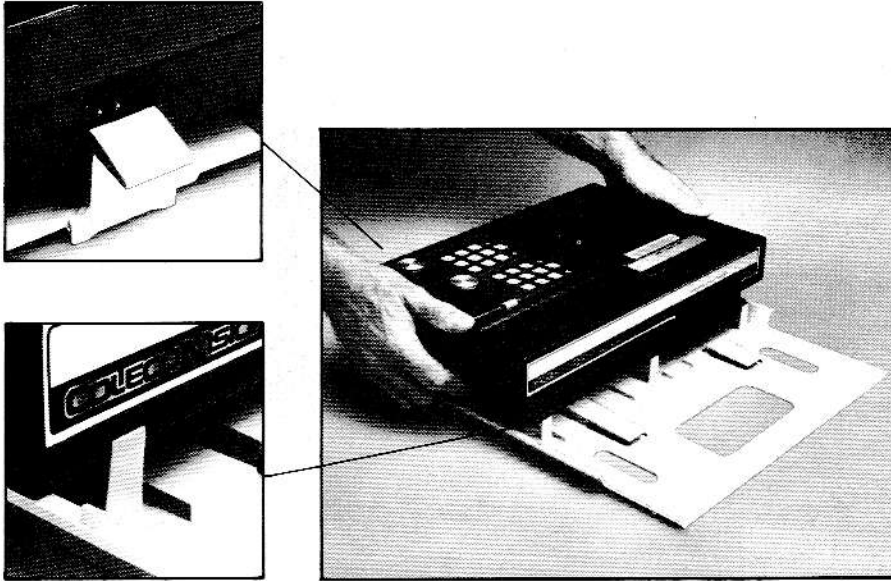


Figure 1-2-13

5. Lift the expansion module interface door on the front of the ColecoVision Video Game System. Make sure it snaps into position and remains open.

Next, slide the game system and console together. This is a safety measure that prevents you from accidentally using the power source with ADAM, and locks the game system and the console together. (See Figure 1-2-14.)

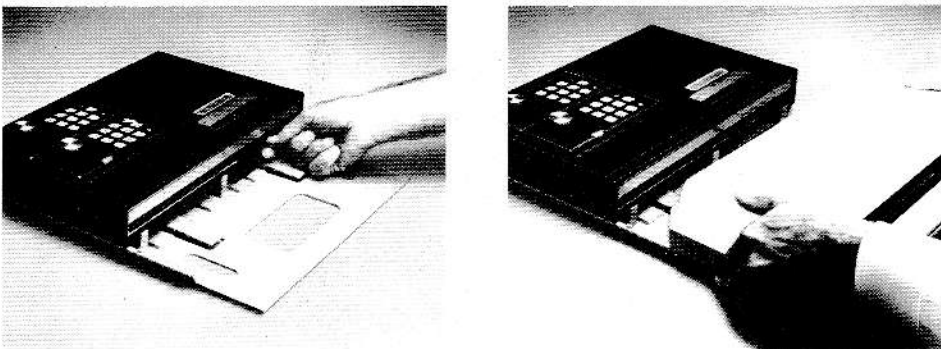


Figure 1-2-14

6. Plug the coiled keyboard cord into the port on the back of the keyboard. Then plug the other end of the keyboard cord into the front of the Memory Console. (See Figure 1-2-15.)

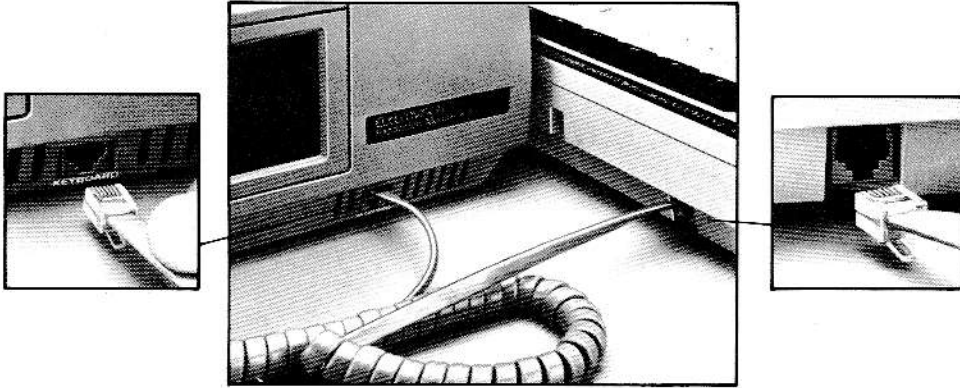


Figure 1-2-15

7. Attach the game controller holder to the keyboard. Then put one of the game controllers in it. (See Figure 1-2-16.)

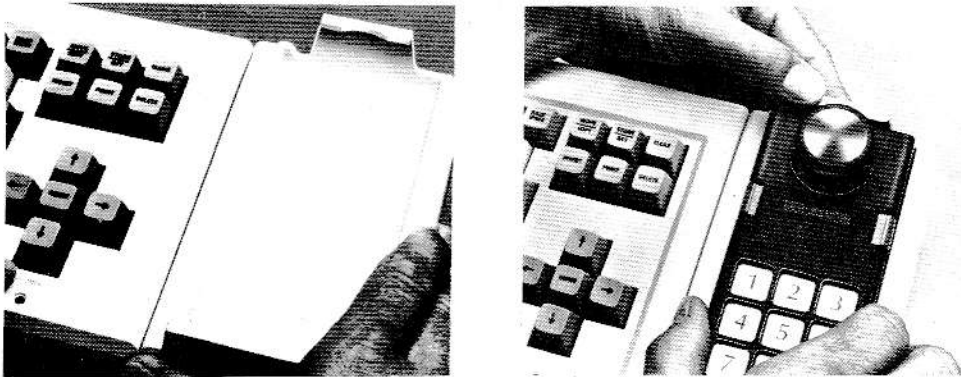


Figure 1-2-16

8. Plug the SmartWRITER printer cord into the left side of the memory console. (See Figure 1-2-17.)

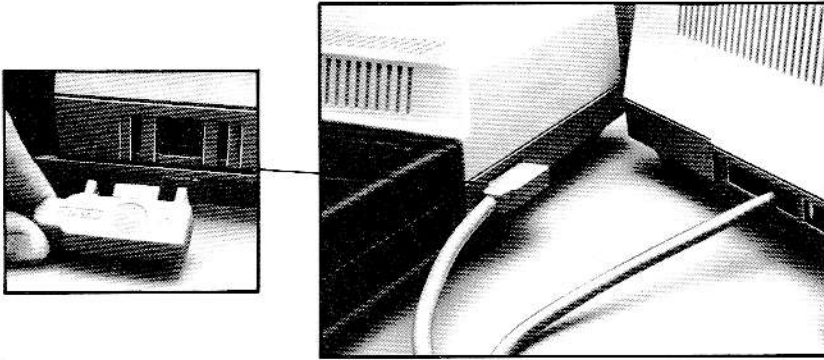


Figure 1-2-17

9. Make sure the video cable from the ColecoVision console is connected to the switch box and the switch box to a color TV. This step has probably been done. But if your console is not already connected to a TV, follow the instructions on pages 17 through 19 to complete this step.
10. Plug the power cord from the back of the printer into a 110/120 volt electrical outlet. Or use the 3-to-2 prong plug adapter, and plug the printer into a standard outlet. (See Figure 1-2-18.)

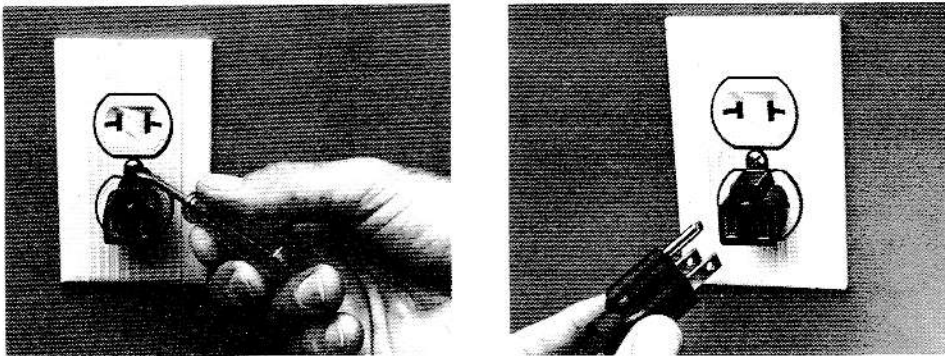


Figure 1-2-18

IMPORTANT: DO NOT TURN ON THE COMPUTER SYSTEM YET. Turn to the next chapter for those directions.



SmartWRITER Word Processing is built into my computer. As I type, words appear on the computer screen.

MARGIN/ TAB/ETC	SCREEN OPTIONS	SEARCH	HI-LITE	HI-LITE ERASE	SUPER/ SUBSCRIPT
--------------------	-------------------	--------	---------	------------------	---------------------

Using Your ADAM Computer

Once your computer is hooked up, you're ready to turn it on. You may need some help if you don't get the result you expect when you turn on the computer. In addition, if you haven't used a typewriter before, you may need help inserting the paper into the printer. You'll find all this information in this chapter, plus how to use:

- ADAM as a typewriter
- SmartWRITER Word Processing
- ready-to-run programs
- your own programs

Turning On Your Computer

Whether you have the ADAM Family Computer System or Expansion Module #3, you follow the same steps to turn on your system. Complete the steps listed below to turn on your computer.

1. Make sure the switch box which connects the console to the TV is set at "game." Or, make sure the switch box is set at "computer"—whichever is appropriate for your switch box.
2. Turn on the TV and set the sound at a moderate level. Select Channel 3 or 4—whichever one does not receive a TV station in your area. Also, select the corresponding channel on the back of the console.
3. Check the digital data pack drive to make sure no digital data pack is in it.
4. Locate the ON/OFF switch on the back of the SmartWRITER printer and flip it to the ON position.
5. Check the red POWER ON light located on the right side of the keyboard. This light should be ON whenever the system is on. If not, refer to the next page.

Troubleshooting

Sometimes you don't get the result you expect when you turn on your computer. Most problems are easy to solve. Refer to the information listed in Figure 1-3-2 to help solve any problem.

TROUBLESHOOTING CHART

Symptom	Cause	Remedy
Indicator Light not "On"	Computer not "On"	Make sure power switch is in "On" position
	Power cable not plugged in	Check power socket for loose or disconnected power cable.
	Power supply not plugged in	Check connection with wall outlet
	TV on wrong channel	Check other channel for picture (3 or 4)
	Incorrect hookup	Computer hooks up to VHF antenna terminals
	Video cable not plugged in	Check TV output cable connection
Random pattern on TV with game cartridge in place	Computer set for wrong channel	Set computer for same channel as TV (3 or 4)
	Cartridge not properly inserted	Reinsert cartridge after turning off power
Picture without color	Poorly tuned TV	Retune TV
Picture with poor color	Bad color adjustment on TV	Adjust color/hue/brightness controls on TV
Sound with excess background noise	TV volume up high	Adjust volume of TV
Picture OK, but no sound	TV volume too low	Adjust volume of TV
	Aux. output not properly connected	Connect sound jack to aux. input on amplifier and select aux. input

Figure 1-3-2

Using Your SmartWRITER Printer

The SmartWRITER printer that came with your computer is a letter-quality printer. This printer can be used with the computer either as an electric typewriter or as a printer.

The SmartWRITER printer uses any kind of paper up to 9 1/2" wide, including computer fanfold paper. If you use computer fanfold paper, a low-cost, paper-feed mechanism called a "tractor" is available. This helps keep fanfold paper from slipping in the printer.

A Pica 10 daisy wheel is included with your computer. You can use other daisy wheels to change the style of your typeface. Your SmartWRITER printer also uses a standard, interchangeable, high-quality, multistrike ribbon cartridge.

CAUTION: Be sure to remove the packing materials which are inside the printer. Remove the printer cover to do this. Then replace the cover securely before using the printer.

You insert a single sheet of paper into the SmartWRITER printer almost exactly as you would into a typewriter.

1. The paper is inserted from the back of the printer under the platen or roller. Make sure the paper is centered on the roller.
2. Turn the platen knob to move the paper forward until the top of the paper is just under the paper bar (sometimes called a paper bail).
3. If the paper is not straight, move the paper release lever forward and straighten the paper. Then release the lever to hold the paper firmly in place.

Clearing the Screen

You can clear the screen of anything you have typed by pulling the computer reset slide switch, which is on the top of the Memory Console, toward you. This "throws away" everything that is on the screen and lets you begin again.

If you're playing a game cartridge and want to begin again, pull the cartridge reset slide switch toward you.

Turning Your Computer Off

When you have finished using your computer, remember to complete the following steps:

- Make sure no digital data pack is in the drive on the Memory Console.
- Turn off the power switch on the back of the printer.
- If desired, move the switch on the switch box to the "TV" setting.
- If desired, turn off the TV or monitor.

Using ADAM as a Typewriter

When you first turn on your computer, it can be used as an electronic typewriter. As you can see by looking at your computer screen, typing with ADAM is similar to typing on a typewriter. There is a roller at the bottom of your screen where words appear as you type. At the top of your screen are left and right margin settings.

To practice using ADAM as an electronic typewriter, complete the steps listed below. (Your computer must be on to do this. If it is not on, complete the steps to turn on your computer first.)

1. Insert a piece of paper into your SmartWRITER printer the same way you would put paper into a typewriter. If you need help, turn to the previous page and follow the instructions provided there.
2. Type the text shown in Figure 1-3-3. As the text in the printer reaches the end of the line, be sure to listen for the "end-of-line" bell. Then press RETURN as you would on a typewriter. (Or, when you can't type another word, press RETURN to end the line; or press MARGIN RELEASE—Smart Key VI — and finish typing the word, then press RETURN.)

ADAM is the first home computer system that is easy to operate and understand. ADAM thinks like you, so you don't have to think like a computer.

Figure 1-3-3

3. Remove the paper you had inserted in your printer. Do this by turning the platen knob. Notice that what you typed now appears on your screen and on the paper.
4. Clear the screen by pulling forward on the computer reset slide switch which is located to the left of the cartridge slot on the Memory Console. (If you wish, you can now turn off the computer by pressing the switch on the back of the printer to OFF. Finally, turn off the TV.)

For more information on using ADAM as an electric typewriter, turn to Part II, "Electronic Typing Lessons."

Using SmartWRITER Word Processing

In addition to using ADAM as an electric typewriter, you can also use ADAM to do word processing. Using word processing is like using an electric typewriter—with one important difference.

When you use a typewriter, anything you type immediately appears on paper. If you make a mistake or want to change something, you can't. You have to type the page again. When you use word processing, anything you type appears on your computer screen. If you make a mistake or want to change something, you can. You correct your work on the computer screen. Then, when everything is perfect, you tell the computer to print a copy of your work.

When using word processing, you don't have to press RETURN at the end of each line. The computer automatically moves words that don't fit at the end of one line to the beginning of the next line. This "wraparound" feature makes typing easy and fast. The only time you press RETURN, when using word processing, is to end a paragraph or create a blank line.

SmartWRITER Word Processing is built into your ADAM Family Computer System. SmartWRITER Word Processing is simple to use and understand. You will find that it makes typing letters, homework assignments, and resumes easy and fun.

To practice using SmartWRITER Word Processing, complete the following steps:

1. To begin, make sure ADAM has been turned on and "Electronic Typewriter" appears at the bottom. Then, press the key labeled "ESCAPE/WP." Notice the change on your screen.
2. To practice using SmartWRITER Word Processing, type the following sentences. NOTE: As you type, you may hear a bell to indicate the end of a line. You do not need to press RETURN, just continue typing.

```
SmartWRITER Word Processing is built into my computer.  
As I type, words appear on the computer screen.
```

Figure 1-3-4

3. To end a paragraph, press RETURN. Then create a blank line by pressing RETURN again. You will see a RETURN symbol, a left-facing black triangle, display each time you press RETURN.
4. Type the following paragraph.

```
With word processing, I can correct my work before  
printing it. I can also store work and get it back  
later.
```

Figure 1-3-5

5. To end a paragraph, press RETURN.
6. Notice how the two sentences appear on your screen. If you would like to create a printed copy of your work, do the following:
 - a. Put paper in your printer (the same way you would put paper in a typewriter).
 - b. To begin the printing process, press the key labeled "PRINT." Notice the information that now appears on your screen.
 - c. Notice the key labeled "IV PRINT SCREEN" at the bottom of your screen. Locate the corresponding Smart Key (IV) at the top of the keyboard. Press this Smart Key (IV).
 - d. To print the information that now appears on your screen, press the key labeled "V."

As you can see, SmartWRITER Word Processing is easy to use. With SmartWRITER Word Processing you can delete mistakes, insert new material, copy information, move information from one place to another, and store your work on a digital data pack for editing or printing later.

To learn more about SmartWRITER Word Processing, look at Part III.

Using Ready-to-Run Programs

You can choose from a variety of ready-to-run programs when using ADAM. You may use any program or game stored on a digital data pack. Also, you can play ColecoVision game cartridges.

By adding Expansion Module #1 to ADAM, you can play video game cartridges made for the Atari 2600 Video Computer System, ActiVision, Imagic, Parker Brothers, and many more. Expansion Module #2 turns ADAM into a arcade-style racing machine. In addition, two exciting game controllers are available for ADAM—Roller Controller and Super Action Controllers.

NOTICE: When using any expansion ports for accessory equipment, shielded connecting cables are required. Contact the accessory manufacturer for further information.

CAUTION: Do not turn ADAM on or off while a digital data pack is in the drive, or permanent damage may be caused to the information stored on the data pack.

To use programs and games stored on digital data packs, do the following:

1. Insert the digital data pack into the digital data pack drive (on the Memory Console).
2. Pull the computer reset slide switch on the top of the Memory Console toward you. Then the name of the program will appear on your screen and the program will begin.

To begin playing ColecoVision game cartridges, you have to complete only three steps:

1. Turn off your computer at the switch behind the printer.

CAUTION: ColecoVision game cartridges should not be inserted when the computer is on or permanent damage may result.

2. Insert a ColecoVision cartridge in the game cartridge slot (on the Memory Console).
3. Turn on your ADAM Family Computer System. The name of the game will appear on your screen and the game will begin.

Writing Your Own Programs in SmartBASIC

There are lots of ready-to-run programs available for your ADAM Family Computer System. In addition, you can create your own programs using ADAM's SmartBASIC.

BASIC is a simple computer language designed especially for beginners. SmartBASIC is ADAM's special version of BASIC. It is just one of the languages that you can use to program ADAM.

When using SmartBASIC you will write a program in numbered lines, like the program shown at the top of Figure 1-3-6. Each line will contain a word that tells ADAM exactly what to do. When ADAM reads the lines, in numerical order, it will do what you want it to do. The example at the bottom of the illustration shows the result of running the BASIC program.

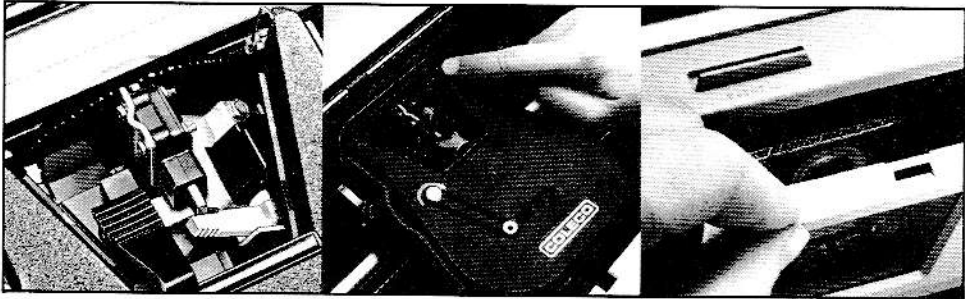
```
10 PRINT "MY NAME IS ADAM."  
20 PRINT "WHAT IS YOUR NAME?"
```

```
MY NAME IS ADAM  
WHAT IS YOUR NAME?
```

Figure 1-3-6

You don't have to learn SmartBASIC to use ADAM. Many people, though, enjoy writing their own programs. Just like sewing or carpentry, programming can be a creative and fun hobby.

To learn more about SmartBASIC and how you can write your own programs, look at Part IV and Part V.



DAISY WHEEL

**RIBBON
CARTRIDGE**

**DIGITAL
DATA PACK**

4

Maintaining the System

As you use your ADAM Family Computer System or Expansion Module #3, you will need to perform some common procedures such as changing the ribbon, changing the daisy wheel, and inserting and removing a digital data pack. Explanations of these procedures, and general system maintenance, are provided in this chapter.

Changing the Ribbon

Ribbons can be purchased wherever you bought your computer or at office supply stores. To change a ribbon, follow these steps:

1. Remove the printer cover by grasping it firmly and pulling up. (See Figure 1-4-2.)

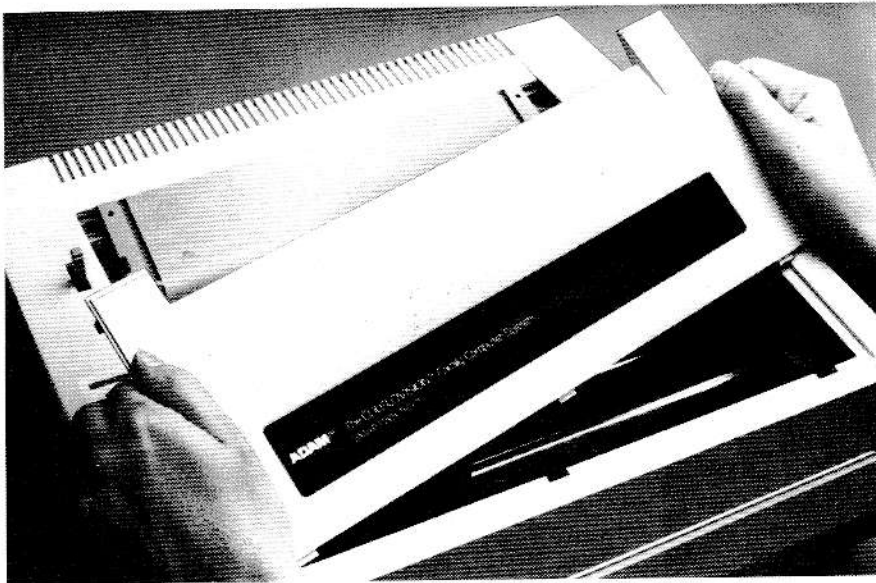


Figure 1-4-2

2. To remove the ribbon, press the notched latch. (See Figure 1-4-3.)

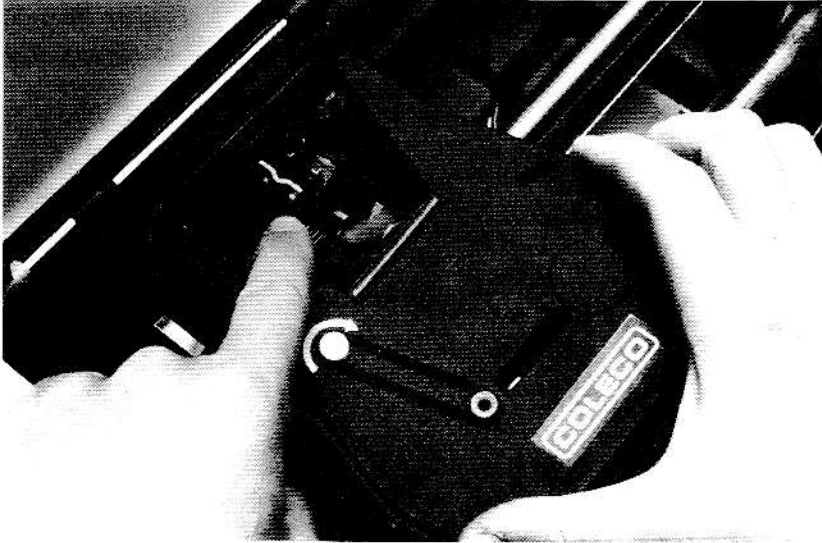


Figure 1-4-3

3. Pull the ribbon cartridge up and out.
4. Snap the new ribbon cartridge into the ribbon tray. Make sure the ribbon feeds around the right spring, between the daisy wheel and the plastic paper guide, and around the left spring.
5. Turn the ribbon advance clockwise until you hear a click. (See Figure 1-4-4.)

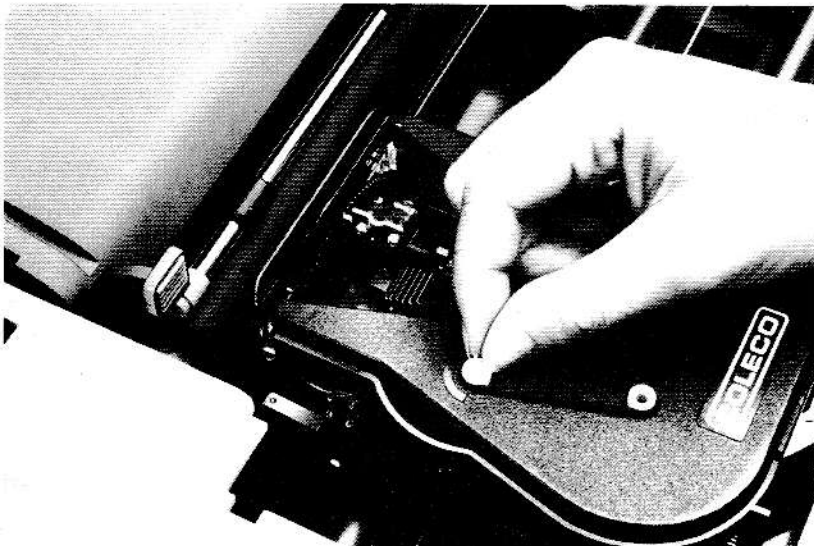


Figure 1-4-4

6. Replace the ribbon cover and press down firmly.

Changing the Daisy Wheel

Daisy wheels can be purchased where you bought your computer or at office supply stores. You can substitute other brands of daisy wheels as long as they are plastic (not metal), 10 pitch, and have 96 characters. Coleco will make multiple styles available.

WARNING: The power plug must be removed before a daisy wheel is changed.

1. Remove the cover from the printer. Then remove the ribbon cartridge. (For instructions, see the preceding text page.)
2. Press in the two latches located on either side of the print mechanism under the ribbon tray. (The latches are approximately midway from the roller to the rear of the tray.) (See Figure 1-4-5.)

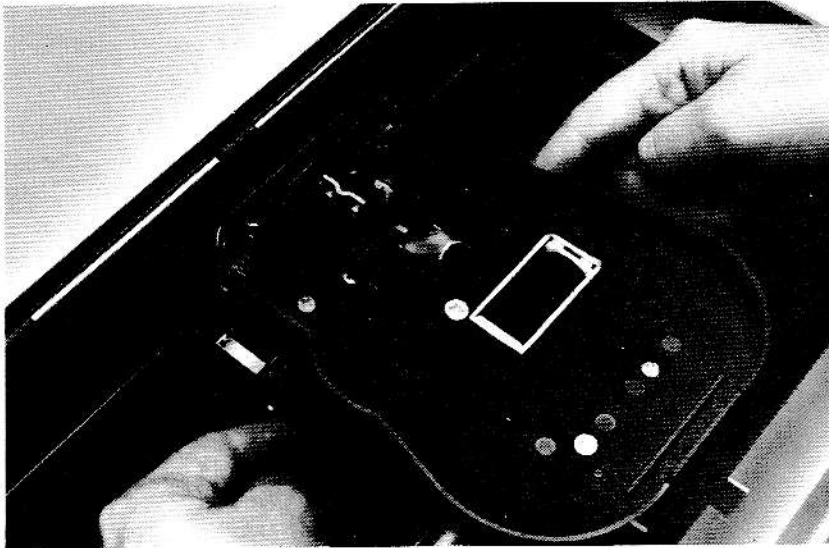


Figure 1-4-5

3. Tilt the print mechanism toward you. Then pull the daisy wheel up and off. (See Figure 1-4-6.)

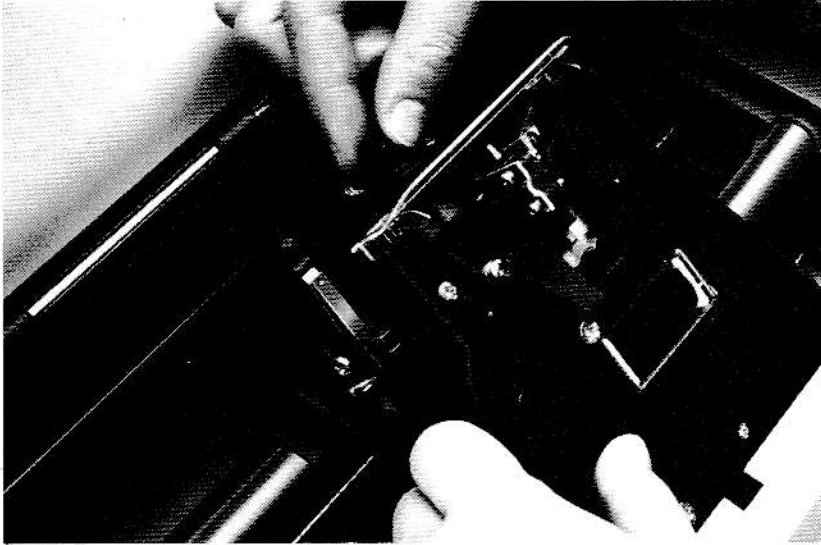


Figure 1-4-6

4. Place a new daisy wheel on the spindle, making sure the square hole in the daisy wheel lines up with its post. Push the print mechanism back down firmly. (See Figure 1-4-7.)

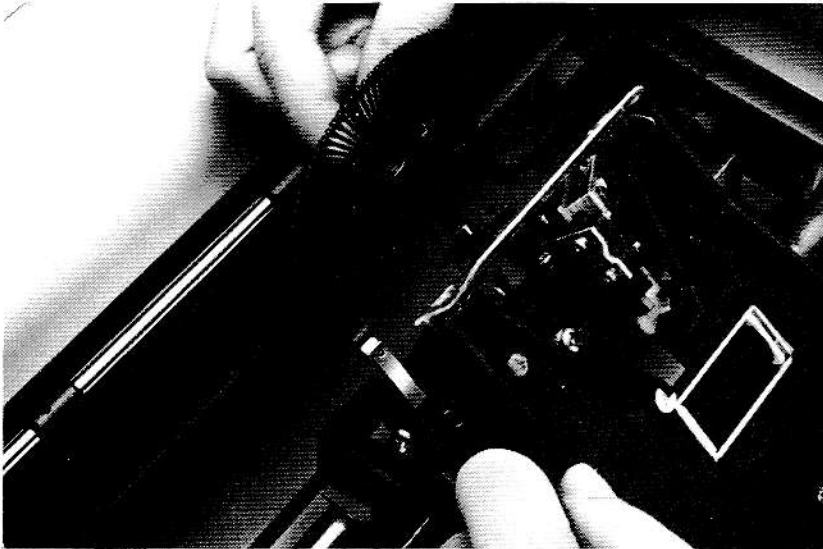


Figure 1-4-7

5. Replace the ribbon cartridge.
6. Finally, replace the cover.

Inserting and Removing Digital Data Packs

Digital data packs are specially designed for your computer. Although they may look like audio cassette tapes, digital data packs are very different. Do not attempt to use audio cassette tapes in place of digital data packs.

NOTE: Do not turn ADAM on or off while a digital data pack is in the drive.

To insert or remove a digital data pack, follow these steps:

1. To open the drive door, press the tape release bar (on top of the door). (See Figure 1-4-8.)

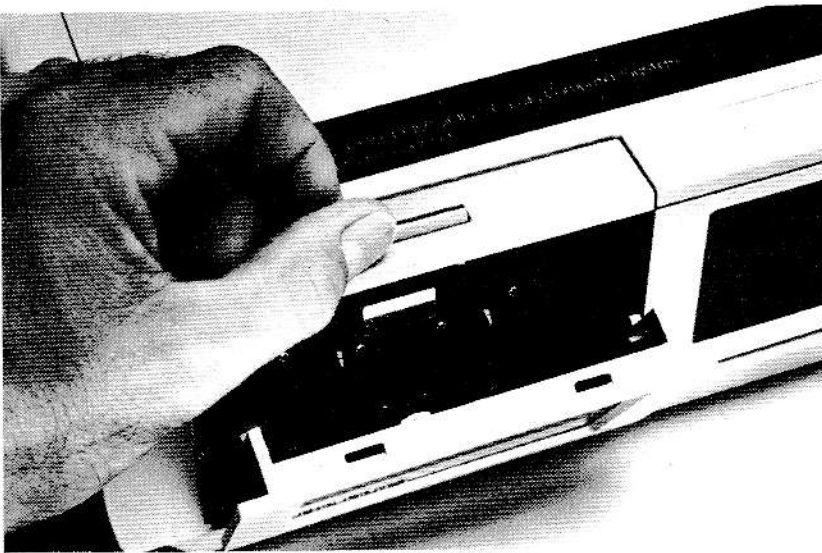


Figure 1-4-8

2. Insert or remove the digital data pack. **CAUTION:** To insert a data pack, make sure the label faces you and that the exposed tape is toward the bottom of the drive. (See Figure 1-4-9.) Damage may result to the tape if it is not inserted properly.



Figure 1-4-9

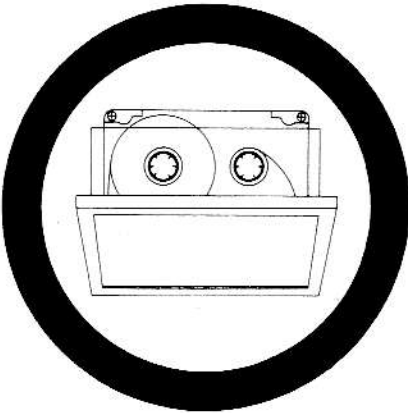
3. Close the drive door. (See Figure 1-4-10.) (NOTE: If the tape is inserted incorrectly, the door is difficult to close.)



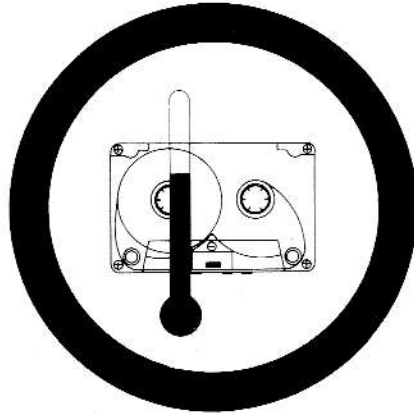
Figure 1-4-10

Hints on Taking Care of Your Computer

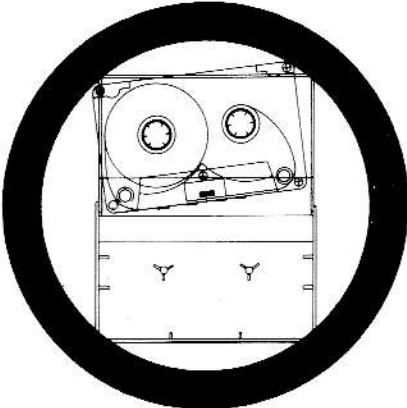
Using your ADAM Family Computer System is easy. The suggestions given on the next two pages will help you keep ADAM in good condition.



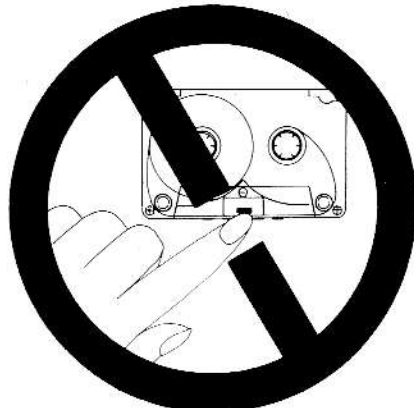
Protect



10°C-52°C



Insert Carefully



Never



No



No

Figure 1-4-11

Screen

- To clean your screen, use any glass or window cleaner.

Keyboard and Memory Console

- Keep food, liquids, and smoke away from the computer.
- Clean your keyboard and Memory Console with a dry cloth.

SmartWRITER Printer

- Keep fingers, clothing, hair, and jewelry away from the printer.
- Clean your printer with a dry cloth.

Digital Data Packs

- Do NOT turn ADAM on or off while a digital data pack is in the drive.
- Do NOT leave digital data packs in the drives.
- Keep digital data packs away from magnets, paper clips, motors, tools and other metal or magnetic items.
- Store and use digital data packs away from extreme heat or cold.
- Store digital data packs in a safe place, away from dust, temperature extremes, and electrical currents. Store digital data packs away from the printer and other computer parts.
- Keep food and liquids away from digital data packs.
- Do not touch the tape in any digital data packs.
- Insert and remove digital data packs carefully.

Part II

Electronic Typing

June 15, 1984

Doggie Train Center
223 Main St.
Any City, U.S.A.

Dear Manager,

We would appreciate it if you would add our new puppy's name to your dog obedience class which starts the first week of July. We have enclosed a check in the amount you requested to hold a place for our dog.

We hope your class will help us train this dog which is 6 months old and weighs almost 70 pounds. The dog has already attended another dog training school -- without much effect. Therefore, we hope your class will have much better results.

Sincerely,

The Adams Family

1

Using ADAM As A Typewriter

The letter shown on the opposite page is just one of the many kinds of letters you can type using ADAM as an electronic typewriter. Of course, you are not limited to typing letters. you can type phone lists, notes, memos, or anything else you want on ADAM.

When you use ADAM as an electronic typewriter, you use it just as you would any other typewriter. Its keyboard and printer work together just like a typewriter. When you press a letter or number on the keyboard, it's printed at once—even though the keyboard is not part of the printer.

You will, however, find it easier to use ADAM than a typewriter. First, what you type appears both on your TV screen and on paper. You'll find it much easier to read what you type on the screen than on paper. And it's in color, too. Second, you only need to touch the keys lightly on the ADAM keyboard to operate it. If you hold a key down, it will repeat. Finally, ADAM has Smart Key labels on the bottom of the screen which guide you through the steps for changing margins, tabs, and line spacing. The procedures for changing these settings are explained in this chapter.

For the two practices in this chapter, you can type either the letter shown on the opposite page or one of your own letters or notes. For the first practice, you should leave the margins, line spacing, and tabs at the standard settings. But before you type a second letter, you'll learn how to change those three settings. You can then type a letter with non-standard margins, line spacing, and tab settings.

Each practice begins with a brief introduction which explains what you'll be doing. Steps you must perform are numbered. The steps are briefly written, with additional information in parentheses. If you don't want the additional explanations, skip the information in parentheses.

Turn to the next page when you are ready to begin.

Lesson # 1—Typing a Letter Using the Standard Settings

In this first practice, you will use ADAM to type a complete letter using the standard settings for margins, line spacing, and tabs. Do each step which is listed below to type the letter at the beginning of the chapter or type a letter of your own.

1. Turn on the TV. Then press the power switch at the back of the Smart-WRITER printer to ON.
2. Insert paper into the printer. It should be centered on the roller. Roll the paper up by rotating the roller knob on the top right of the printer. Roll the paper up until it can be tucked under the paper bail.
3. Type your letter or whatever it is you want to type. Keep these points in mind:
 - ADAM is automatically set for a 60-character line with margins of 10 and 70, single spacing, and tab stops at every 5 characters.
 - Before typing the date, press the TAB key (on the left side of the keyboard) about three times so that the date will be indented. Then type the date and press RETURN. A RETURN symbol will display to remind you where you pressed RETURN.
 - After typing the date, press RETURN two or three times. RETURN not only ends lines, but it also creates blank lines. Then type the name and address and the salutation. Press RETURN where necessary.
 - When typing the body of the letter, listen for the end-of-line bell which sounds when you have typed about 55 characters on a line. (This will appear as 1-3/4 line on the screen.) Finish the word you are typing and press RETURN.
 - If you need to fit an extra few letters on a line, press Smart Key VI (at the top of your keyboard). This key is labeled as MARGIN RELEASE (on the bottom of your screen). To type past the margin, press this key.
 - Remember that you can't correct errors with a typewriter unless you use "white out" or a similar product.

(This lesson continued on the next page.)

4. When you have finished typing the letter, remove it from the printer by pulling the paper release lever toward you and pulling up on the paper. Be sure to move the lever back before inserting another sheet of paper.
5. Erase the letter from the screen by pulling the computer reset slide switch (which is on the Memory Console) toward you. This "clears" the letter from the screen so that you can type something else.

Congratulations! You have just typed your first letter. Now continue reading to find out how to change the standard settings.

Lesson # 2—Changing Margins

Before you type another letter (memo, note, or any other “document” of your choice), reset either the left or right margin by completing the steps below.

Make sure paper is in the printer. The print head should also be positioned at the beginning of a line. This happens when you use the computer reset switch or press RETURN.

1. Press Smart Key V (on the top of the keyboard). Notice that the current function of this key is MARGIN/TAB/ETC, as shown in the Smart Key label at the bottom of the screen. (The labels for these keys can change each time one of the Smart Keys is pressed. Because the top row of black keys can be used for many different tasks, they are called “Smart Keys.”)
2. Press Smart Key II, which is labeled HORIZ MARGIN on the screen. Notice that the Smart Key labels on the screen change again.
3. When the yellow prompt with HORIZ MARGIN appears, decide whether you want to set the left or right margin. Press Smart Key III LEFT to set the left margin, or press Smart Key IV RIGHT to set the right margin. The choice is yours! (The number shown on each label is the current margin setting.)
4. When the yellow prompt changes and reads HORIZ MARGIN USE ARROW KEYS, press the left and then the right arrow key (bottom right-hand side of the keyboard) to see the effect. The character position of the margin displays in the LEFT or RIGHT Smart Key label, at the bottom of the screen, depending on the margin you decided to reset. At the top of your screen, the dark bars mark the margin positions on the scale.
5. When you have a margin setting you like, press Smart Key VI (DONE). Watch the labels on the screen change again.

Lesson # 3—Changing Line Spacing

While most work is typed single-spaced, you can increase the spacing between lines. Spacing increases in half line segments. How many spaces can you put between lines? Complete this exercise to find out.

1. Press Smart Key V (MARGIN/TAB/ETC).
2. Press Smart Key V (LINE SPACING).
3. A yellow prompt displays with LINE SPACE #. The number identifies the line spacing in effect. To change the spaces between lines, press Smart Key IV (which is labeled UP) to increase the spaces between lines. Or press Smart Key V (labeled DOWN) to decrease the spacing. When you press either key, the line spacing number in the prompt changes. Try pressing both keys and see how wide and narrow the line spacing can be set.

Be sure to change the spacing so you can see the effect.

4. When finished, press Smart Key VI (DONE).

Lesson # 4—Changing Tabs and Typing a Second Letter

Tab stops are usually set at every 5 spaces. You can see the tab stops which are set on ADAM by looking at the margin scale across the top of the screen. The white dots, set every 5 characters, are tab stops.

Tab stops can be easily changed. Complete the steps below to change the tab stops on your screen.

1. Press Smart Key V (MARGIN/TAB/ETC).
2. Press Smart Key IV (TAB).
3. When the yellow prompt with TAB POINTER # USE ARROW KEYS appears, decide whether you want to clear or set a tab. In this case, you'll first clear a tab and then set one.
4. To clear a tab, press the right arrow key (at the bottom right-hand side of the keyboard) several times until the tab pointer in the margin scale at the top of the screen stops on an existing tab stop (marked by a white dot). Then press Smart Key IV (TAB CLEAR).
5. To set a tab, use the left or right arrow keys to position the tab pointer where the new tab stop should be. Refer to the margin scale at the top of the screen to do this. (You can set the tab wherever you want.) Then press Smart Key V (TAB SET) to set the tab.
6. You can repeat steps 4 and/or 5 if you wish to clear or set more tab stops. (If you want to clear all the tab stops, press Smart Key V (ALL CLEAR).)
7. When you have finished, press Smart Key VI (DONE).

Now that the margins, line spacing, and tab stops have been changed, type another letter or any other document using the new settings.

REMINDER: If you want to return to the standard settings, you can either reset each setting individually, pull the computer reset switch forward, or turn ADAM off and then turn it back on again. Whenever you turn ADAM on, the standard settings will be in effect regardless of what had been set before.

Reviewing What You Just Learned

In a short time, you learned how to use ADAM as an electronic typewriter. If you're familiar with a standard typewriter, you can tell how easy it is to use ADAM as a typewriter.

Here's a list of the key points you just covered in this chapter.

- To use ADAM as an electronic typewriter, just turn on the TV and the SmartWRITER printer.
- Use TAB and RETURN exactly as you would on a standard typewriter. TAB indents text on tab stops. RETURN is used to end a line or create a blank line. When you press RETURN on ADAM, you will see a RETURN symbol on the screen.
- ADAM comes with standard margin, tab, and line spacing settings. Margins are set at 10 and 70 (a 60-character line), tab stops are set at every 5 characters, and the line spacing is single.
- Margins, tabs, and line spacing are easy to change. Press the Smart Key (V) which is labeled MARGIN/TAB/ETC and the Smart Key labels on the bottom of the screen will guide you through the procedures.
- If you turn ADAM off and back on again or pull the computer reset switch, the standard margins, tabs, and line spacing will once again be in effect. ADAM does not remember any of the settings you made.
- If you want to type just a few more characters on any line, press Smart Key VI (MARGIN RELEASE).

While ADAM is certainly a wonderful typewriter, you may want to use it as an electronic typewriter only when you are typing envelopes or short documents, like notes, where mistakes don't matter. With SmartWRITER Word Processing, you can easily correct mistakes and print more than one copy of your work. Now that you've used ADAM as a typewriter, go to Part III and find out how easy it is to use SmartWRITER Word Processing.

Part III

SmartWRITER Word Processing

January 15, 1984

The Video Game Store
123 Main St.
Any City, U.S.A.

Dear Manager,

We would appreciate it if you would add our name to your mailing list. We are interested in receiving information about the latest games and educational programs available at your store. We're also interested in knowing when you will be having store-wide sales.

We would like to thank you for helping us select the ADAM Family Computer System. It's particularly in demand in the evening when everyone is usually home.

Sincerely,

The Adams Family

Typing Your First Letter

The letter shown on the opposite page is just one of the many kinds of letters you can type with SmartWRITER Word Processing. This letter was typed using the settings which are standard on your word processor. These settings include one-inch margins on all sides and single spacing between lines.

All that was done to type this letter was to turn on SmartWRITER, type the letter, correct any errors, and print it. Easy? Yes, very easy.

You, too, could type this letter doing only those four things. But you'd be missing some of the other wonderful features of SmartWRITER—such as changing the screen colors and saving your letter so you can print it again.

You can learn about these features and a few others by completing the ten exercises which begin on the following page. (If you're very impatient, all you need to do is complete parts 1, 7 and 9. But you'll be missing some of the real reasons for owning a word processor.)

Each exercise begins on a separate page. The purpose of the exercise is briefly explained. The numbered steps should be completed by you. Anything with an asterisk (*) is additional information.

In this exercise, you can type either the letter shown on the opposite page or one of your own letters. If you decide to type one of your own letters, then please type it so that it looks like the one shown—standard margins, single spacing, "tabbed in" first lines of paragraphs, and so forth.

Lesson #1—Entering WP and Choosing Your Color

To use SmartWRITER Word Processing, you must turn on your ADAM computer and be in the word processing mode. If you haven't done this, then complete the three steps which follow.

1. Turn on the TV or monitor and set the volume to a moderate level. (Remember, the switch box must be set to games or computer and the TV to channel 3 or 4.)
2. Turn on the computer by reaching around the back of the printer and pressing the power switch to ON. Notice the electronic typing screen appear on your TV or monitor. The computer tells you it's in "electronic typewriter" by displaying that message at the bottom left of the screen.
3. Press the key labeled ESCAPE/WP (located at the upper lefthand corner of the keyboard). This will put your computer into the word processing "mode." After doing this, notice that the "electronic typewriter" message has been replaced by Smart Key labels.

Before you begin typing, take a moment to look at the word processing screen on your TV or monitor. Compare it to the picture in Figure 3-1-2, so that you become acquainted with all the features available on your screen.

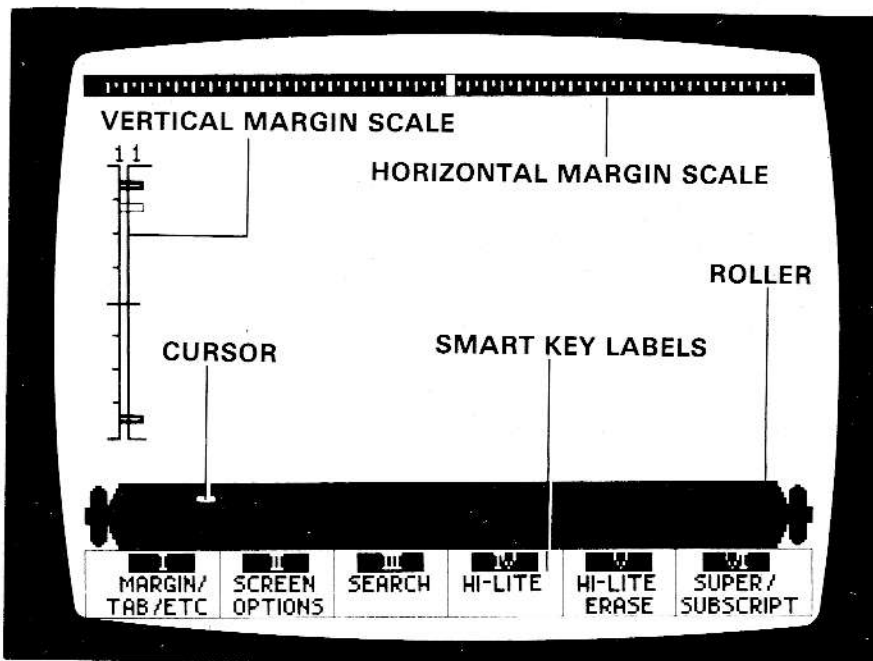


Figure 3-1-2

(This lesson continued on the next page.)

If you are interested in knowing more about the screen features, then read the following points marked with asterisks (*).

- * **Horizontal and vertical margin scales:** You can see where your left and right margins are set by looking at the red markers on the scale at the top of the screen. Tab stops are shown by white markers. The vertical margin scale shows where top and bottom margins are set.
- * **Smart Key labels:** These labels are at the bottom of the screen and marked with roman numerals I through VI. These correspond to the top row of keys, called Smart Keys, on the keyboard. The labels on the screen tell you what will happen when you press a Smart Key. These labels change to help guide you through procedures. Sometimes messages, which display in yellow in the label area, give you extra help.
- * **Roller:** Everything you type in the Standard Format is typed on this roller, which looks like the platen on a typewriter. If you select the Moving Window Format, which is explained in chapter 3, you will not type on a roller.
- * **Cursor:** This is a beam of light that looks like an underline. It shows where the first or next letter you type will appear on the screen.

You can also change the colors and sound level used on SmartWRITER. Want to see a different color? If so, complete the steps which follow.

1. Notice that Smart Key label II is SCREEN OPTIONS. Press Smart Key II at the top of the keyboard and see what happens to the labels.
2. The labels changed. The second label now reads COLOR SELECT. Press Smart Key II again and watch the labels.
3. Decide what background color you want on your screen and press the corresponding Smart Key (I-V). Check your screen.
4. Repeat steps 1-3 to see another color. Repeat those steps as many times as you wish. When you find the color you prefer, press Smart Key VI (DONE). Then continue to the next lesson.

Lesson # 2—Typing the Letter

Whenever you turn on SmartWRITER Word Processing, the following settings are in effect:

Left and right (horizontal) margins:	10 and 70
Top and bottom (vertical) margins:	1 inch each
Line spacing:	single

These settings were selected because they are considered standard for most personal, academic, and business work. You'll use these settings in this practice. In a later chapter, you will learn to change the margin and line spacing settings.

Follow the directions below to type the portion of the letter shown on the opposite page. If you wish, you may substitute one of your own letters.

1. Type the date, just as if you were using a typewriter. (If you're not used to a typewriter, then follow a-c below.)
 - a. To type the date, which is tabbed in, press the gray key labeled TAB (located on the left side of the keyboard) three times. Since several tab stops are set and shown on the top margin scale, you don't have to worry about setting tabs. As you press TAB, watch the cursor.
 - b. Then type the date using the white letter and number keys on the keyboard. The letters and numbers will appear on the top line of the roller. (If you make mistakes, ignore them for now.)
 - c. Finally, press RETURN three times—once to end the line and twice to make two blank lines. Notice that the date moves off the roller.
2. Type the name, address, and salutation (Dear Manager,) just as if you were using a typewriter. At the end of each line, press the RETURN key. Also, press RETURN each time you want a blank line.
3. Next, type the first paragraph of the letter—again using the typewriter keys. **But**, do not press RETURN at the end of each line. SmartWRITER Word Processing will take care of that for you.
4. When you have finished typing the first paragraph, press RETURN twice—once to end the paragraph and once to make a blank line. These are the only times you need to press the RETURN key when using SmartWRITER Word Processing.

Lesson # 2—Typing the Letter (cont.)

January 15, 1984

The Video Game Store
123 Main St.
Any City, U.S.A.

Dear Manager,

We would appreciate it if you would add our name to your mailing list. We are interested in receiving information about the latest games and educational programs available at your store. We're also interested in knowing when you will be having store-wide sales.

We would like to thank you for helping us select the ADAM Family Computer System. It's particularly in demand in the evening.

Figure 3-1-3

5. Now, type only the first three or four words of the second paragraph. (Type only this much, so that you can learn more of the differences between SmartWRITER Word Processing and a typewriter.)
6. See what happens to your letters when you press the gray BACKSPACE key three or four times.

(As you can see, pressing BACKSPACE is like using an eraser. It neatly erases letters—and numbers, too. You can use BACKSPACE whenever you are typing and see a mistake on the typing line. Just BACKSPACE and then retype.)
7. Next, retype the words (also called "text") that you just erased. Then continue to practice #3 to find out about some other important keys in SmartWRITER Word Processing.

Lesson # 3—Moving the Cursor

If you're accustomed to using a typewriter, you may wonder how to back-space without erasing letters or numbers. Look at the right side of your keyboard for the four keys with arrows on them and the key labeled HOME. These are called Cursor Control Keys because they let you move the cursor without erasing or "destroying" your text. (Remember, the cursor looks like a bright underline and tells you where the next character will be typed.)

Find out more about the Cursor Control Keys by completing these steps.

1. You should still see a couple of words from the second paragraph on the roller. You can move the cursor backwards over these letters without erasing them by pressing the key with the left arrow. Press this key a couple of times and watch the screen.
2. You can probably guess what's going to happen when you press the key with the right arrow on it. But go ahead, try pressing that key a couple of times too.
3. Before you reach for the up and down arrow keys, try something else first. Press the left arrow key again a couple of times. Then press the SPACEBAR—that's the long white bar at the bottom of the keyboard.

(If you're familiar with a typewriter, you may have been surprised that the SPACEBAR also erases text on SmartWRITER. Remember to use BACKSPACE and SPACEBAR only when you want to erase text—not to move the cursor.)

4. Go ahead. Try using the up and down arrow keys. Press each one a couple of times.
5. You can also use the HOME key to move the cursor. Try it now.
6. If you want to move the cursor rapidly press the HOME key and an arrow key at the same time and see what happens. In fact, try pressing HOME and left arrow, HOME and right arrow, HOME and down arrow, and HOME and up arrow.
7. Use the arrow keys and HOME to move the cursor around the roller. Also, use them to move the text up and down on the screen. Moving the text up and down on the screen is called "scrolling." Spend a couple of minutes now getting used to the effect of these keys. Then continue to the next lesson.

Lesson #4—Correcting Errors

When you use the Cursor Control Keys and “scroll” your text up and down the screen, you can read what you have typed. Scrolling lets you proofread your text. You can then see if there are any mistakes—sometimes called “typos” (which is short for typographical errors). Or, you can re-read your text and decide whether you want to change what you typed.

You’ve already learned one way to correct mistakes, by pressing BACK-SPACE and retyping text. There are two more ways to correct errors—adding or “inserting” text, and striking over text.

1. To follow these directions, you’ll find it easiest if you scroll to the end of the letter (which, by the way, is often referred to as a “document” in word processing). To do this, press HOME and either the up or down arrow key as many times as necessary to bring the last line of the letter onto the roller.
2. One kind of error is accidentally leaving out a word. Assume you wanted to add the word “Also” to the beginning of the second paragraph. Adding words on a word processor is called “inserting” text. Do steps a-d now.
 - a. First, use the arrow keys to move the cursor to the beginning of the sentence. Whenever you want to insert a word, move the cursor to where the insert is to begin.
 - b. Press the INSERT key (which is on the right side of the keyboard). Watch what happens to your text. (The text after the cursor position temporarily disappears from the screen.) This allows you to type as much text as you need. You should also see the message “TYPE TEXT” in the Smart Key label area.)
 - c. Type the word to be inserted—“Also” (without the quote marks)—and press the SPACEBAR once.
 - d. Press Smart Key VI (labeled DONE) and your text will reappear.
3. Next, you need to change “We” to “we” in that sentence. Do this by moving the cursor (with the arrow keys) so it’s under the “W.” Then type over the letter “W” with a lower case “w.” This procedure is called “strikeover” or “typeover.”

Continue to the next lesson to learn another way to correct your work.

Lesson #5—Correcting More Errors

You've now learned three different ways to correct mistakes: BACKSPACE, INSERT, and strikeover. There's one more method that you'll find very useful. It's called DELETE. It's usually used when you want to erase words from within a paragraph. BACKSPACE, on the other hand, is used to erase words you just typed.

In this lesson, delete the first two words—"Also we"—from the beginning of the second paragraph by completing the steps below.

1. Make sure the cursor is under the "e" in the word "we."
2. Press the DELETE key which is at the top right of the keyboard. (Notice that some of the Smart Key labels change and a message "HI-LITE TO DELETE" appears.)
3. "Highlight" or mark the text to be deleted as follows.
 - a. Notice that Smart Key label IV is HI-LITE. Press Smart Key IV. (When you do this, the label changes to HI-LITE OFF.)
 - b. Use the left arrow key to move the cursor back under each letter in the two words "Also we." A red underline will be drawn under these. (Any arrow keys can be used to draw this underline. Use whatever is appropriate for the situation.)
 - c. Press Smart Key IV again to turn the HI-LITE OFF.
4. Tell SmartWRITER to delete your text by pressing either Smart Key VI (which is labeled DELETE) or the DELETE key at the right of the keyboard. The choice is yours.

(NOTE: The group of keys at the top right of the keyboard are called Command Keys. They are MOVE/COPY, STORE/GET, CLEAR, INSERT, PRINT, and DELETE. Two other Command Keys are UNDO and ESCAPE/WP. The name on each Command Key describes the procedure it is used to perform. When you press certain Command Keys, such as DELETE, the Smart Key labels change. And you complete the procedure by following the messages, using the Smart Keys, or using the Command Keys.)

IMPORTANT: Do not type more text now or press other keys. You need to leave everything just as it is to complete the next lesson.

Lesson # 6—Using the “Oops” Commands

You may sometimes wish you hadn't done a procedure. For example, perhaps you deleted text and then decided you want it back. What can you do? Use one of the “oops” commands. UNDO is one of the two “oops” commands and will get back text you just deleted. Another “oops” command is ESCAPE. It lets you stop any command before finishing it. Complete the steps which follow to try out UNDO and ESCAPE.

1. Since you just deleted text in last lesson, assume you decided you want it back. As long as you have not pressed any key or typed anything, you can. How? Just by pressing UNDO (upper right-hand side of the keyboard). Press it now and watch the screen.

The text reappears. To remove the highlighting, press HI-LITE ERASE (V). Then move the cursor under all the text to be de-highlighted. Finally, press ERASE OFF (V).

(The text “Also we” should reappear. If it does not, then redo the last two lessons and press UNDO immediately after completing lesson 5.)

(UNDO can be used to get back text which you deleted, cleared, or replaced. You will learn more about CLEAR and REPLACE later.)

2. The second “oops” command is ESCAPE, which lets you “back out” of any procedure. Since you need to be in the middle of a procedure to do this, leave your cursor wherever it is, press INSERT, and type any word.

Assume you don't want the word you just typed and want to get rid of the INSERT procedure. All you do is press the key labeled ESCAPE/WP (upper left-hand side of the keyboard). Do this now and watch what happens.

(ESCAPE can be used to back out of such procedures as DELETE, INSERT, CLEAR, and MOVE. You can back out of any of these procedures as long as you have not completed the last step in the procedure.)

3. Now use any of the error correction features—BACKSPACE, INSERT, strikeover, or DELETE—to fix any mistakes in the letter. Remember to use the Cursor Control Keys to scroll your text to the roller so that you can make changes. And you can use UNDO or ESCAPE whenever necessary.

After you have corrected any errors, finish typing the letter. When finished, the basic style of your letter should match the letter at the beginning of the chapter. Then, go on to the next lesson—printing.

Lesson #7—Printing Your Letter

You can print letters, reports, or other documents anytime with SmartWRITER Word Processing.

- * You can print before you correct mistakes. (Some people prefer to proofread a printed copy rather than scrolling text on the screen.)
- * You can print after you correct mistakes.
- * Or, you can save text, recall or "get" it back and then print it.

Before you print, make sure you have a sheet of paper in the printer. If you don't, then insert one as follows. Refer to the diagram for assistance.

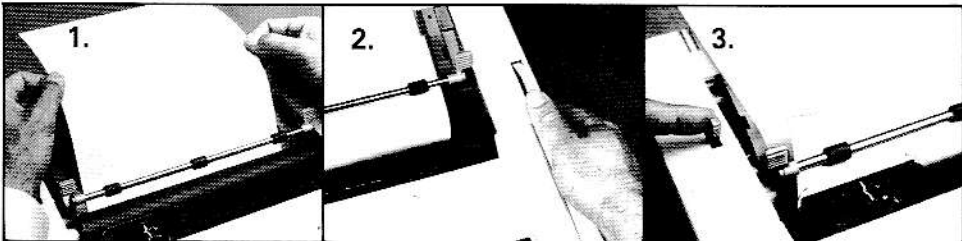


Figure 3-1-4

1. Insert paper from the back of the roller (or platen), making sure the paper is centered between the edges of the roller.
2. Use the platen knob to roll the paper forward until the top of the paper is under the paper bar (or paper bail).
3. Pull the paper bail up. (When the letter is being printed, you will need to make sure the paper goes under the bail and then pull it forward.)

(This lesson continued on the next page.)

With paper in the printer, you are ready to print your letter.

1. Press the key labeled PRINT (upper right-hand side of the keyboard). Watch what happens to the Smart Key labels when you do this.
2. Since you want to print your entire letter, look for the Smart Key label with PRINT WK-SPACE (workspace). Notice the other choices. Then press Smart Key V.

(The "workspace" includes everything you have typed—whether you can see it on the screen or not.)

3. Now that you're ready to print, look at the labels at the bottom of the screen. Several labels let you specify paper type and page numbering information. (These are introduced in chapter 4.) This time, you'll just print with the standard—which is no page numbering.

Notice that one Smart Key label is marked PRINT (V). Press Smart Key V and watch the printer!

Congratulations! You've just printed your first letter. Continue to the next lesson and find out how to save your letter for another exercise.

Lesson #8—Saving Your Letter

One of the best reasons for using a word processor is being able to save your work for later use. Saving or storing a letter allows you to get it back later, change it, and print it again.

In this lesson, you'll store the letter you just completed. You can then use the letter in other lessons.

1. With SmartWRITER Word Processing, text is stored on a digital data pack. Make sure a digital data pack is inserted in the digital data drive on your Memory Console. If a digital data pack is not in your computer's drive, then insert one now. (Refer to Part I for directions if you need assistance.) If you forget this step, the system will prompt you later to insert a data pack.
2. Press the STORE/GET key (located in the group of six keys to the right of the keyboard).
3. Now look at the Smart Key labels at the bottom of the screen. To store the contents of the entire workspace, press Smart Key V (STORE WK-SPACE).
4. Once again, look at the message and Smart Key labels. Press Smart Key III (DRIVE A) to store your text on the digital pack in Drive A. (You will always make this selection unless you add another digital data drive. In that situation, the digital data pack could be in Drive B.)
5. The message and Smart Key labels changed again. This time, you need to type a name for your letter. This must consist of 10 characters or less. For this example, type the name REQUEST. (It is not necessary to use the SHIFT key to type the name in capital letters. Your computer will make any letters appear as capitals in this situation.)
6. Now tell SmartWRITER you're ready by pressing Smart Key VI (STORE WK-SPACE). Then listen. You can hear your text being stored.

Lesson #9—Clearing Your Letter Off the Screen

Once you've stored your text, you can type another letter, resume, chart, or any other document.

But wait! Your old text is still on the screen. Before you can do anything else, you need to erase or clear your letter from the screen. You could use DELETE, but that would require highlighting the entire letter to erase it. A shorter way is to use the CLEAR procedure which is explained below.

1. First, press the CLEAR key at the right of the keyboard. Watch the Smart Key labels as you do this.
2. Next, tell SmartWRITER how much text you want to clear. In this case, clear the workspace by pressing Smart Key VI (CLEAR WK-SPACE).
3. Look at the message area once again. SmartWRITER asks if you're sure you want to get rid of the text. Since you do, press Smart Key VI (FINAL CLEAR). Watch what happens on your screen.

Clearing text is comparable to throwing it in the wastebasket. If you changed your mind as you were throwing it out, you could have pressed ESCAPE to get it back. Or, if you just completed the procedure and changed your mind, you could press UNDO. By having these features, SmartWRITER saves you from a lot of aggravation.

Now continue to the next page and complete the last lesson in this chapter.

Lesson # 10—Getting Your Letter Back on the Screen

Perhaps you're wondering if you'll ever see your letter again now that you've cleared it off the screen. You will because you stored it on a digital data pack.

Once you've stored text, you can bring it back any time. Get your letter back on the screen now by following the steps below.

1. Make sure the same digital data pack you stored your letter on is still in the drive. (Later, when you've stored letters and reports on several different digital data packs, you'll need to keep track of where everything is stored. You can learn about that in the next chapter.)
2. Press the STORE/GET key.
3. Look at the Smart Key labels and press Smart Key VI (GET).
4. Look at the labels again. Press Smart Key III (DRIVE A) to get the letter from the digital data pack in Drive A.
5. A File Directory of all the letters, charts, and other "files" on the digital data pack appears on the screen. Choose the file named REQUEST that you stored in lesson #8. Do this by using the arrow keys to move the File Directory pointer to the name of the file you want.
6. Press Smart Key VI (GET FILE). After a moment, the letter stored in the file appears on the screen.

(This lesson continued on the next page.)

At this point you can do one of the following with the letter that is on the screen. Read your options and then do whichever you prefer. (If you're feeling particularly adventurous, you can do combinations of these options. The choice is yours.)

- * Having retrieved your letter, you can throw it out again by clearing it off the screen. To do this, follow the steps in lesson #9. If you clear the letter from the screen, however, a copy will still remain on the digital data pack. (To delete the file from the data pack, you must follow the "delete file" procedure described in the next chapter.)
- * You can print your letter again by following the steps in lesson #7 and then clear it again by repeating the steps in lesson #9.
- * You can add to the letter or revise it, and then re-store it for use later. If you decide to re-store the letter, you can use the same file name—REQUEST. If you choose this option, the first version of your letter will be moved to the Back-up Directory.
- * Or, you can change the letter and re-store it as a separate version by using a different file name, such as REQUEST1 or VIDEO. If you choose this option, the first and second versions of your letter will be listed in the File Directory.

When you've finished, you can go directly to the next chapter. Or, you can turn off your system by completing the two steps listed below.

1. Turn off the TV.
2. Turn off the SmartWRITER system by pressing the power switch on the back of the printer to OFF.

Before going on to the next chapter, read the following page which contains a summary of the key points in lessons #1-10.

Reviewing What You Just Learned

In the last hour, you learned all the "basics" about SmartWRITER Word Processing. By using those procedures, you can type simple letters, reports, assignments, and other things on your ADAM computer.

When you type some of your work, you won't need all the procedures you learned about. After all, if you're a good typist, you just need to turn the system on, type the letter, print it, clear it, and turn the system off. And you're done!

On the other hand, if you need to make corrections or want to store the letter, you'll need to use additional SmartWRITER features.

Here's a list of the key points covered in this chapter.

- * To use SmartWRITER Word Processing, you must turn on your TV or monitor, your ADAM computer, and press ESCAPE/WP.
- * You type letters, reports, and other documents using the typewriter keys. However, you don't press RETURN at the end of each line of a paragraph. SmartWRITER wraps words automatically. Use RETURN only to end lines short of the right margin or create a blank line.
- * If you make mistakes, you can correct them by using one of these features: BACKSPACE, INSERT, DELETE, or typeover (strikeover).
- * To correct errors, the cursor must be at the error. You can move the cursor by using the Cursor Control Keys (arrow keys and HOME key). Unlike a typewriter, SPACEBAR and BACKSPACE will do more than move the cursor—they'll also erase text. So using the Cursor Control Keys is important.
- * You can print text by using the PRINT key. You can print text as many times as you wish.
- * If you want to use text again, you must STORE it. You can store it and then print it, or print it and then store it. Whenever you want to STORE text, you must give your letter or document a "file name" of 10 characters or less.
- * Whenever you're ready to start work on a new document, you must first CLEAR the old document off the screen.
- * You can get back on the screen any document you have stored. Use the STORE/GET key and tell the system which "file name" you want.

(This lesson continued on the next page.)

- * Whenever you have finished using the system, turn it off.

If you want to type some of your own work or take a break before going on to the next chapter, then do so. On the other hand, if you're ready to find out more about the "filing system," then continue to the next chapter. You don't have to complete the chapters in sequence, so you could work on the lessons in some other chapter.

Files Organized
by Document Type

Letters

- J.Smith DDP #3
- S.Pecos DDP #3
- A.Martwick DDP #12
- T.Zinn DDP #9

School Work

- Math.Nov 1 DDP #2
 - French.Nov5 DDP #2
 - Speech.Nov3 DDP #5
-

Files Organized
by Author

Linda

- Hist. Report DDP #5
- Eng. Report DDP #7
- Letter to John DDP #8
- Note to Maria DDP #8

Scott

- Resume DDP #1
 - Letter to
ABC Corp DDP #1
-

Getting Organized—File Maintenance

With a word processor, you can store or save everything you type. This allows you to edit your work or print it again and again. This is a big advantage over using a typewriter—where you have to type it again and again.

You'll soon find that you have a lot of stored documents—or "files" as they are called in word processing terminology. The situation becomes similar to recording lots of different songs on audio cassette tapes. Pretty soon it's hard to remember which song is on which tape.

Here's what you need to do if you and your family are storing lots of files.

- * Decide what documents should be stored on which digital data packs: You may want to organize your files so that only one kind of document is stored on a data pack. For example, one data pack could be labeled "LETTERS" and only letters stored there. Another could be labeled "Homework" and only homework stored there. Or you may want to give everyone in the family their own data packs. They would decide what to store on their data packs. They may even want to organize their own files by type.
- * Decide how to keep track of what's stored on the data packs: You can label your digital data packs with numbers or people's names. You'll find that this helps keep track of documents. You might even want to keep a written list of what's on what data pack—as in the examples shown on the opposite page.
- * Keep files up to date: You can delete files which are no longer needed by using the DELETE FILE procedure. This will free space on your digital data packs.
- * Make copies of very important files: Everyone who uses a word processor or computer knows you should make "back-ups" of important work that you will use again, such as a resume. While it's unusual to have a problem with a digital data pack or lose a digital data pack, it could happen.

In just two lessons, you'll learn how to copy and delete documents. You can rename a file by copying the document and then deleting the file stored under the old name. Instructions on storing and getting documents are in the previous chapter.

Lesson # 1—Copying a File

To complete this practice, you can use any digital data pack which has one or more documents already stored on it. This practice will show you how to copy a file that is on a data pack. This is sometimes called making a “back-up.”

1. Insert the digital data pack that contains the file to be copied into the digital data pack drive.
2. Move the file into your computer's memory as follows:
 - a. Press the STORE/GET key (upper right-hand side of the keyboard).
 - b. Press Smart Key VI (GET).
 - c. Press Smart Key III (DRIVE A). The File Directory appears on the screen. A File Directory pointer is pointing toward the name of one of the files listed on the File Directory.
 - d. Using the arrow keys, position the File Directory pointer so that it faces the name of the file you want to get.
 - e. Press Smart Key VI (GET FILE). In a moment, your file will appear on the screen.
3. Decide on which data pack you want to store the file. You can store it on the one that's currently in the drive or remove the pack from the drive and insert another data pack.
4. Now store the document as follows:
 - a. Press the STORE/GET key.
 - b. Press Smart Key V (STORE WK-SPACE).
 - c. Press Smart Key III (DRIVE A). A File Directory appears on the screen.
 - d. Type a file name for your file. Use 10 characters or less.
 - e. Press Smart Key VI (STORE WK-SPACE).
5. When the system has finished copying the file, it will position the cursor under the first character in the document. You can remove the document from your screen by using the CLEAR procedure.

You have just made a back-up of your document. You can use this same procedure to rename a file—that is, save a file under a new name. When you have done that, just delete the version of the file that is stored under the old name.

Lesson # 2—Deleting a File

Eventually, you will decide you want to do some “housekeeping” and get rid of some of the letters, reports, and other documents that are filed on your digital data packs. You can do this easily by deleting any files you no longer want. Complete the steps below to delete one version of the document you copied in the previous practice.

1. Make sure the digital data pack that contains the file to be deleted is inserted into the digital data pack drive.
2. Press the STORE/GET key.
3. When prompted, press Smart Key III (GET).
4. When prompted, press Smart Key III (DRIVE A). The File Directory appears on your screen.
5. Use the arrow keys to position the File Directory pointer in front of the name of the file to be deleted.
6. Press the DELETE key.
7. Then press Smart Key VI (FINAL DELETE).
8. To remove the File Directory from the screen, press ESCAPE/WP.

Remember, once you delete a document you have erased it and you cannot get it back. Make sure you only delete documents you no longer want.

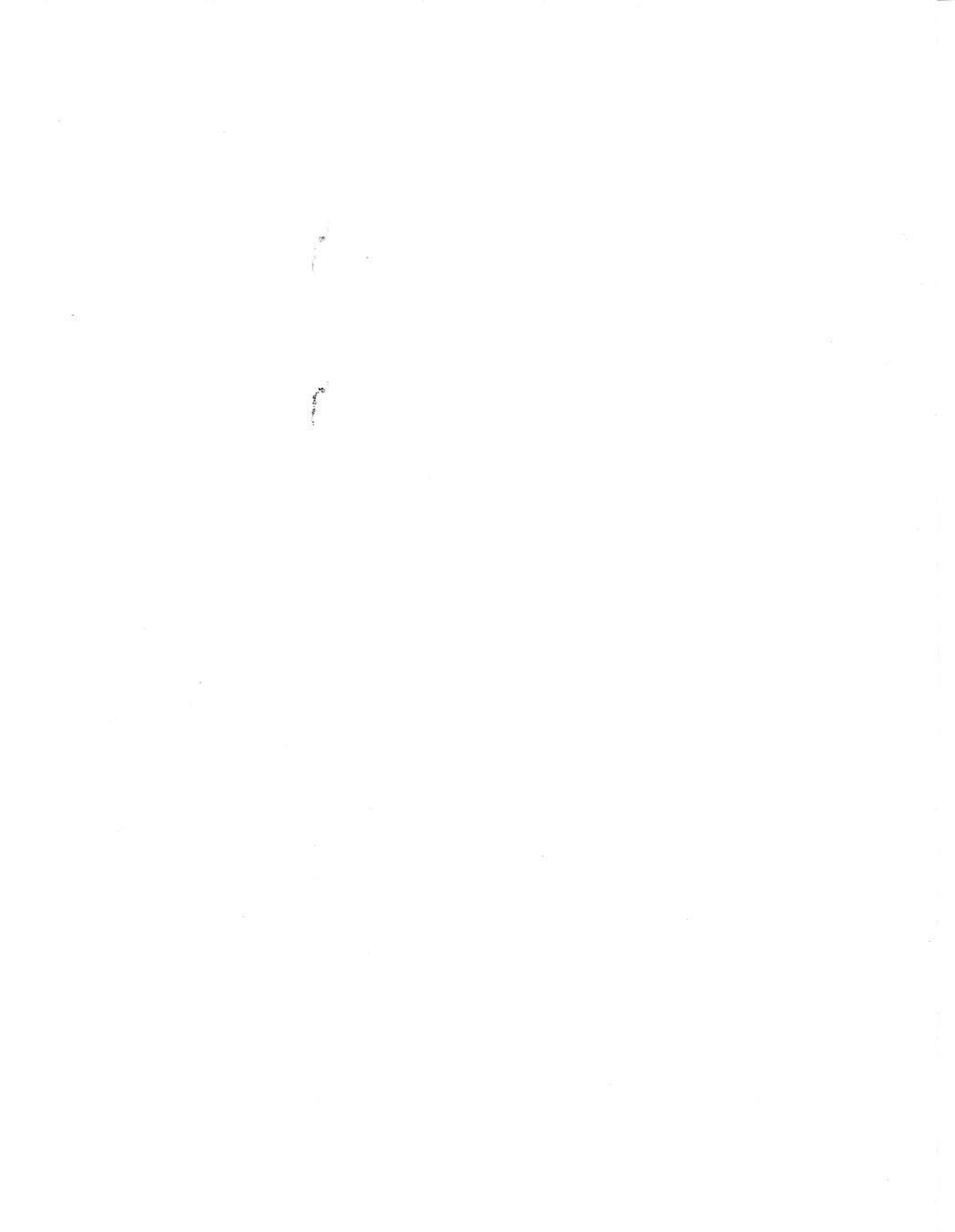
Reviewing What You Just Learned

File maintenance involves two kinds of tasks. First, you must decide where to store your documents and how to keep track of where they are. Second, you can use certain procedures to keep your files up to date.

Here is a summary of some of the key points involving files.

- * Documents can be stored using the STORE procedure, and recalled by using the GET procedure. (Documents which are stored are still displayed on the screen and must be cleared using the CLEAR procedure.)
- * Documents which have been stored on a digital data pack are listed on a File Directory.
- * The File Directory has a pointer that is moved with the arrow keys until it faces the name of the document you want to GET.
- * Each digital data pack has its own File Directory and Back-up Directory. When a file is stored using the same name a second time, the first version of the file is moved to the Back-up Directory. Each time a new version of the file is stored, using the same name, the previous version of the file is moved to the Back-up Directory. The old version of the file is automatically deleted from the Back-up Directory.
- * Documents which have been stored can be recalled using the GET procedure, deleted using the DELETE A FILE procedure, renamed using RENAME A FILE, or copied using COPY A FILE. Renaming is performed by copying a file first and then deleting the first version.
- * If a lot of documents are stored, you will need a way to keep track of them. You can label a data pack by number or with a name. You can also keep a written record of what document is stored on what digital data pack.
- * Make back-ups of very important documents.
- * Remove the File Directory from the screen by pressing the ESCAPE/ WP key.

Now that you've learned about the filing system, find out how to do some other things with your computer—such as typing a telephone list.



THE ADAMS FAMILY TELEPHONE DIRECTORY

312-491-0001	Allen, Jenny
777-5534	Berenson, Tom and Lydia
978-9966	Bergstrom, Alice and Jim
777-9111	Connant, John
505-998-0011	Derwinski, Paul
777-9944	Fernandez, Mark
978-1234	Fenner, Ollie and Alex
978-4321	Leary, Mollie and Jim
777-5678	Longworth, Julie
617-888-3215	Mason, Billie Jean
456-9223	Nathen, Les and Andy
777-9090	Robbins, Dr. Pauline
456-4545	Southtown Animal Hospital
879-0505	Travis, Amy and Dan

Typing a Telephone List

There's nothing more convenient than having a telephone list of the people you call most frequently at every telephone in the house. With Smart-WRITER Word Processing, you can type this list quickly, store it, and print it.

When numbers change, people move, or you want to add new information, you can simply "update" your list and reprint it. Convenient and easy!

To set up a list of telephone numbers, you change some of the standard settings that come with ADAM. Usually, you'll want to change the following ones:

- * Change the Standard Format to the Moving Window Format. This lets you see exactly what your document will look like.
- * Change the margins. You can make the directory wider or narrower, though this is optional.
- * Change the tabs. Usually you'll want to clear all the tabs and just set one or two—wherever different columns begin.
- * Change the line spacing. Your entries will be easier to read if they are typed in spacing-and-a-half or double spacing.

Whenever you want to type any chart or report with information in columns, you'll find that changing these four settings will make your job easier.

Complete the four lessons in this chapter to type your own phone list. You can type it with the numbers followed by the names (as shown on the opposite page) or with the names followed by the numbers. The approach shown here makes it easier to match the number with the name since only three or four spaces separate each. If names come before numbers, there can be a big gap between the name and number which makes it more difficult to read.

Lesson # 1—Changing to the Moving Window Format

The Moving Window Format looks very different from the Standard Format (which appears when SmartWRITER is turned on). Find out what the Moving Window Format looks like by completing these steps.

1. Press Smart Key II (SCREEN OPTIONS).
2. Press Smart Key VI (MOVING WINDOW). Notice the appearance of the screen. The roller or platen has been removed and the cursor is at the top of the screen.

You can try out this format before you type the phone list by typing a couple of lines of anything you want. When you're ready to continue, CLEAR the screen.

Lesson # 2—Changing Margins

You change the left and right margins on SmartWRITER exactly as you do on ADAM when it's used as a typewriter. The left and right margins are referred to as "horizontal" margins because they determine the number of characters across a page.

SmartWRITER is preset for a left margin of 10 and a right margin of 70. This allows 60 characters on a line. If you are using standard 8 - 1/2 inch wide paper, you will have a 1 inch left margin and a 1-1/2 inch right margin.

You can also set top and bottom margins which are called "vertical" margins if you want to use paper that is longer or shorter than 11 inch paper.

In this exercise, you'll make the right margin a little wider. The procedure is very similar for changing the other margin settings.

1. Press Smart Key I (MARGIN/TAB/ETC).
2. Press Smart Key II (HORIZ MARGIN).
3. Since you want to change the right margin, press Smart Key IV (RIGHT). The number below the word RIGHT is the position of the current right margin.
4. A yellow prompt displays which tells you to use the arrow keys. To make the right margin wider, press the right arrow key several times until the number below the word RIGHT is 75.
5. Press Smart Key VI (DONE).

Lesson # 3—Changing Tabs

Tab stops are set every 5 spaces on SmartWRITER. But when you type charts or phone lists, you usually have to change these settings. You'll find it's easiest to clear all the tab settings and then set only those you want.

The phone list shown at the beginning of the chapter was typed with two tab settings. One tab setting was at 14, so that phone numbers which did not have area codes could be tabbed in and lined up properly. The second tab setting was at 25, where the names began.

To calculate where tab settings are needed, remember that the left margin is still set at 10. You must decide how far in from 10 you want to tab for each column of information. In this case, an area code plus a dash takes up 4 spaces, so $4 + 10$ equals 14. The longest phone numbers take up 12 spaces, and then some space is wanted between columns. So if you add to the right margin (10) the number of spaces for a phone number (12) and some blank spaces (3), you may decide to set another tab at 25.

Complete the steps below to change tabs on your screen.

1. Press Smart Key I (MARGIN/TAB/ETC).
2. Press Smart Key IV (TAB).
3. Clear all the tab stops by pressing Smart Key V (ALL CLEAR).
4. Use the right arrow key to move the tab pointer to the first tab setting, which is 14. (Refer to the number in the yellow prompt).
5. Press Smart Key III (TAB SET).
6. Use the right arrow key to move the tab pointer to the second setting (25).
7. Press Smart Key III (TAB SET).
8. When finished, press Smart Key VI (DONE). Your tab settings display on the margin scale at the top of the screen.

Lesson #4—Changing Line Spacing

Most charts, phone lists, and other information typed in columns is easier to read if it is typed in spacing-and-a-half or double spacing. This helps your eye follow the information on one line, since there is more distance between lines.

Change the single spacing which is standard to 1-1/2 spacing for the phone list.

1. Press Smart Key I (MARGIN/TAB/ETC).
2. Press Smart Key V (LINE SPACING).
3. Press Smart Key IV (UP) once. Notice that the number in the prompt changes from 1 to 1-1/2.
4. Since you're finished, press Smart Key VI (DONE).

Lesson # 5—Typing, Storing, and Printing the Phone List

Now that your settings are correct, you can type the phone list. Read the hints listed below and then type the list.

- * If you have a phone number with an area code, type the complete phone number beginning at the left margin. Then press the TAB key on the left side of the keyboard once. Type the person's name—last name, comma, first name.
- * If an area code is not needed, press TAB once. Type the phone number. Press TAB again, and type the person's name.
- * When typing all upper-case letters, press the LOCK key (lower left corner of the keyboard). To return to lower-case letters, press LOCK again.
- * Press RETURN after you finish typing each person's name. It will appear as if your list is single-spaced, but it will print at 1-1/2.
- * Check the information on each line before you press RETURN. It's easier to correct column information before you go to the next line.

After you type the list, store it using the STORE procedure: press STORE/GET, press STORE WK-SPACE (V), insert a digital data pack in the drive, press DRIVE A (III), type file name, press STORE WK-SPACE (VI).

Then print the list. Use the PRINT procedure: insert paper in printer, press PRINT, press PRINT WK-SPACE (V), press PRINT (V). Repeat this procedure for each additional phone list you need.

When you have finished, you can CLEAR the screen: press CLEAR, press CLEAR WK-SPACE (VI), press FINAL CLEAR (VI).

You now have as many phone lists as you need, and you can "update" the one you typed at any time.

Reviewing What You Just Learned

Whenever you type lists, charts, or other information in columns, you'll probably want to use the Moving Window Format, change your tab stops, and change the line spacing. You may also want to change margins.

Of course, you can change tab stops, line spacing, and margins whenever you type anything if you want non-standard settings. Some people like to type text in the Moving Window Format, since each line on the screen equals one printed line. It does this by moving or scrolling the text sideways as it is typed. This gives the appearance of seeing your text through a moving window—thus the name Moving Window Format.

The best way to get familiar with any of these features is to try them. With some features, like the Moving Window Format, it's easier to try using it than to explain it. So if you wonder what something looks like, just try it.

If you're interested in using the automatic page numbering feature, look at the next chapter. If you are interested in typing formulas, try chapter 5.

PRINTING OPTIONS

* Amount to be printed

(text in screen some highlighted	text in screen	text in screen
Highlighted text	Text on the screen	Text in the workspace

* Type of paper to be used

(regular paper stacked)	(computer fan-fold paper)
Single sheets of paper	Computer fan-fold paper

* Whether pages are to be numbered

(2 sheets no page number	3 sheets numbered 1-3	3 sheets numbered 10-12
No page numbers	Pages numbered automatically beginning with 1 or any other number	

4

Using Your Printing Options

SmartWRITER offers you more than just the opportunity to print anything as many times as you wish. It provides you with some special printing options which are illustrated on the opposite page. You can:

- * Select different amounts of text to be printed—from just a word or two that you highlight, to the entire screen, to everything that's in the workspace. Most of the time, you'll print everything that's in the workspace (also called "memory"). This allows you to print everything you've just typed, even if it's scrolled off the screen.
- * Select the kind of paper to be used. You can use single sheets of any size paper. (Just remember to change the vertical margins on your computer if you use paper that is longer or shorter than 11 inches.) Or, you can use computer paper. This is often called fan-fold paper because it looks like a continuous scroll that is perforated every 11 inches. Fan-fold paper usually has holes on either side in case you are using a special feeder on the printer called a tractor. The edges with the holes can be easily separated from the rest of the paper because they are perforated.
- * Decide whether or not you want pages to be numbered automatically. Page numbers will not be printed unless you select this option. You can even have page numbers beginning with a number other than 1.

The printing options are easy to select because they are all presented after you press the PRINT key. To learn about these print options, complete the following lesson.

Lesson # 1—Selecting Different Printing Options

The practice provided below will guide you through all the printing options. To complete this lesson, you will need some text on the screen. Type anything you want, insert paper in the printer, and then complete the steps which follow.

1. Press the PRINT key.
2. Decide how much text you want to print. If you want to print everything on the screen or everything in the workspace, just press the appropriate Smart Key—either IV (PRINT SCREEN) or V (PRINT WK-SPACE).

If you want to print only highlighted text, you must escape from this procedure by pressing ESCAPE/WP. Highlight the text you want printed using HI-LITE and the arrow keys. Once the text is highlighted, you can begin this procedure again from step 1. You can then select PRINT HI-LITE (III). If you try to select PRINT HI-LITE (III) before highlighting text, the PRINT procedure automatically ends.

3. Decide what kind of paper you want to use. If you want to use single sheets, you do not press any key. If you want to use fan-fold, press Smart Key II (FAN-FOLD). Even if you don't intend to use fan-fold, see what happens when you press II. Notice how the labels change. Press II again to return to single sheets.
4. Decide whether you want pages numbered automatically. If you do not, you don't press Smart Key III or IV. However, assume you want pages numbered automatically. Press Smart Key IV (AUTO PAGE #?) to select this.

Furthermore, assume you want the first printed page to be number 4. Perhaps this is the beginning of a second chapter. Press Smart Key III (1ST PAGE IS) several times until a 4 appears in the label. If you overshoot, you must use ESCAPE/WP to end the procedure and then go through it again.

5. With your options selected, press Smart Key V (PRINT) and watch your document be printed.
6. If you have printed highlighted text, remove the highlighting by pressing V (HIGH-LITE ERASE), moving the cursor under highlighted text, and pressing V (ERASE OFF).

Repeat this exercise two or three times. Each time select different options and watch the results.

Reviewing What You Just Learned

There are three options available when you want to print:

- * The amount of text to be printed;
- * The type of paper to be used;
- * Whether pages are to be numbered automatically. If they are to be numbered automatically, then you need to select the first page number if it is to be greater than 1.

All the options can be chosen after you press the PRINT key, with the exception of printing highlighted text. If you want to print only some of the text on the screen, you must highlight it before pressing the PRINT key.

The print options are easy to use. You'll find the option to specify the first page number particularly helpful if you're writing a report with several chapters. You can store each chapter separately. When you print, just select the appropriate page number with which each chapter is to begin.

H₂O
e = mc²
this source.¹

¹ *The Lost Planet*, p.23.

Formula-Subscript
Equation-Superscript
Footnote reference-
Superscript
Footnote-Superscript

5

Typing Equations, Formulas, and Footnotes

Equations, formulas, and footnotes all require numbers to be typed above or below the typing line. Numbers (or letters) that are typed above the main typing line are called superscripts. Those typed below the typing line are called subscripts. Examples of text with subscripts and superscripts are shown on the opposite page.

On some computer systems, subscripts and superscripts are called print commands because you can't see their effect until they're printed. This is the case when using SmartWRITER. When you type text which is to be superscripted, it is enclosed on either side by characters that resemble upside-down "L's." Similarly, subscripts look as if they're enclosed between "L's."

When you print your text, however, subscripted and superscripted text prints half a line below or above the typing line. Because of this, you must change line spacing so that it's 1-1/2 or 2 rather than single-spaced.

Complete the steps listed below. Do them once with subscripted text, and once with superscripted text. Use the examples in the illustration or any of your own.

1. Change the line spacing to 1-1/2 as follows: press MARGIN/TAB/ETC (I), press LINE SPACING (V), press UP (IV) once, press DONE (VI).
2. Type text to the point where the subscript or superscript is to begin. For example, if typing the formula for water (H_2O), just type the "H."
3. Press Smart Key VI (SUPER/SUBSCRIPT).
4. When the yellow prompt displays, decide whether text is to be lowered or raised. To lower text, press Smart Key V (SUBSCRIPT). To raise text, press Smart Key VI (SUPERSCRIP). A special "L" symbol will display.
5. Type the text which is to be lowered or raised. This is often only one number or letter, such as the "2" in the first example.
6. Press Smart Key VI (DONE). Another special "L" symbol will display.
7. Type the rest of the text as usual.

MOVING TEXT

123 Main St.
Anywhere, U.S.A
Ms. Maggie Jones

Ms. Maggie Jones
123 Main St.
Anywhere, U.S.A

Before moving text

After moving text

COPYING TEXT

Lou, lou, skip to my lou
Skip to my lou, my darlin'

Lou, lou, skip to my lou
Lou, lou, skip to my lou
Lou, lou, skip to my lou
Skip to my lou, my darlin'

Before copying text

After copying text

SEARCHING FOR AND REPLACING TEXT

J. S. Sands Co.

J. S. Sands & Company

July 23, 1984

July 15, 1984

Jule Vernes

Jules Verne

Before replacing errors

After replacing errors
with correct text

6

Revising Text

Several procedures available with SmartWRITER are usually used to change or revise text you've already typed. These procedures are MOVE and COPY, and SEARCH and REPLACE. Although these terms are descriptive, here's a brief definition of each.

- * MOVE is used when you want to relocate text—that is, move it from its original location to a new location.
- * COPY allows you to duplicate text within a screen. The text remains in its original location and is copied to the new location.
- * SEARCH tells the system to go directly to specific text. This may save you from looking through several pages of text.
- * REPLACE allows you to automatically substitute one word or group of words for another word or group of words. You can instruct the system to replace each instance it finds, or only selected instances. REPLACE can save you time updating information (such as a new date) or correcting errors (such as an incorrectly typed name). Since the search/replace can be done automatically, you can be sure that each instance will be changed.

Since MOVE and COPY are very much alike, one exercise shows you how to do both. One exercise also explains how to use SEARCH and REPLACE. Although SEARCH can be done without REPLACE, REPLACE can be done only if SEARCH is done first.

You may find all these procedures useful when you're typing a new document, as in the MOVE and COPY examples shown on the opposite page. However, they are usually used when you want to revise text you've already typed.

Lesson # 1—Copying and Moving Text

If you want to copy or move text, you perform almost the same steps—only the result is different. Type three or four lines of text (or use text from the examples on the first page of the chapter). Then, complete the steps listed below to copy text.

When you have finished, complete the steps summarized in the last paragraph to move text.

1. Press the MOVE/COPY key.
2. Press Smart Key VI (COPY).
3. When prompted, highlight the text to be copied. Use the cursor keys to position the pointer under the first character to be highlighted and press Smart Key IV (HI-LITE FIRST). Use the cursor keys to position the pointer under the last character to be highlighted and press Smart Key V (HI-LITE LAST).
4. When prompted, move the cursor to the location where the text is to be copied.
5. Press Smart Key V (COPY) and then press Smart Key VI (DONE).

Scroll the text up and down with the cursor keys to see both instances of the text.

Now, move some text from one location to another as follows: press the MOVE/COPY key, press MOVE (V), position the cursor under the first character to be moved and press HI-LITE FIRST (IV); position the cursor under the last character to be moved and press HI-LITE LAST (V); position the cursor where the text is to be moved and then press MOVE (VI).

Lesson # 2—Searching For and Replacing Text

For this exercise, use any text of your choice. Make sure the same word or phrase is used two or more times in the text.

You could perform the first three steps to search for text, or all seven steps to both search for and replace text. Complete the entire exercise once. You can perform any of the variations suggested later.

IMPORTANT: Before using these procedures, use the cursor keys to position the first character of the first word on the roller. SEARCH and REPLACE begin with the text after the cursor. Text that is before the cursor is not affected by these procedures.

1. Press Smart Key III (SEARCH).
2. When prompted, type the old text you want to find.
3. Press Smart Key VI (START SEARCH). The system will search for the first occurrence or instance of the old text and bring that text to the platen. At this point, you could instruct the system to search for another occurrence, begin replace, or end the search procedure.
4. Press Smart Key V (REPLACE).
5. When prompted, type the new or replacement text.
6. At this point you have another choice—you can replace text selectively or everywhere. Decide which of the steps below to do.
 - Replace only this instance by pressing Smart Key V (REPLACE). Then search to the next instance by pressing Smart Key IV (SEARCH.) To replace the next instance, press IV (SEARCH) or skip by pressing III (SEARCH NEXT).
 - Replace all instances of old with new by pressing Smart Key VI (REPLACE ALL).

When you have finished, scroll the text up and down to examine the results. This completes the information on revision procedures.

Letters

- J. Smith DDP #3
- S. Pecos DDP #3
- A. Martwick DDP #12
- T. Zinn DDP #9

School Work

- Math. Nov 1 DDP #2
- French. Nov 5 DDP #2
- Science. Nov 7

Linda

- Hist. Report DDP #5
- Eng. Report DDP #7
- Letter to John DDP #8
- Note to Maria DDP #8

Scott

- Resume DDP #1

TELEPHONE DIRECT

- 1. Jenny
- son, Tom and Ly
- trou, Alice and
- nt, John
- nski, Paul

January 15, 1984

The Video Game Store
123 Main St.
Any City, U.S.A.

Dear Manager,

We would appreciate it if you would add our name to your mailing list. We are interested in receiving information about the latest games and educational programs available at your store. We're also interested in knowing when you will be having store-wide sales.

Products are also developed from comments made by insightful individuals. Not so long ago, a group of computer wizards put together what they thought was the best microcomputer available. Unfortunately, several other companies arrived at the completion of their systems at the same time. An observant friend suggested using an office electronic typewriter as the printer and keyboard, with this company supply computer

MS FR
-000V
7-557

- 8-9965
- 77-9111 Cor
- 998-0011 De
- 777-9944 Fi
- 978-1234 F
- 978-4321
- 777-5578
- 517-888-3215
- 456-9223
- 777-9090
- 456-4545

Expenses	1982	1983
Auto		
gas	552.00	623.00
insurance	350.00	362.00
maintenance	210.00	250.00
major repairs	525.00	450.00

- Mason, Bill
- Nathen, Les and Andy
- Robbins, Dr. Pauline
- Southtown Animal Hospital
- Travis, Amy and Dan

DALE L. OWENS

Business Address
1000 Main St.
Your City, State 00000
Job Responsibility

Ale
Jim

ive, and
eeded.

Applications—Typing Letters, Reports, Lists, and Resumes

ADAM can be used to type a wide variety of documents for personal use, business, or school. You will find listed below the standard settings which come into effect every time ADAM is turned on. These can be easily changed at any time. You can change margins and tab stops, line spacing, color and sound, page ends (text length), and format type (Standard or Moving Window) at any time—so that your settings affect only a portion of a document. The other settings (paper type and page numbering) affect the entire document.

ADAM comes preset with the following standards (the first six can be changed by pressing Smart Key I—MARGIN/TAB/ETC):

* Left and right (horizontal) margins	10 and 70, a 60 character line
* Top and bottom (vertical) margins	6 and 60, 1" top and 1" bottom margin
* Tab stops	Every 5 spaces, from 15 to 65
* Line spacing	Single
* Page end (text length)	Automatically after 54 lines
* Paper type	Letter size, 8-1/2" × 11"
* Color and sound (SCREEN OPTIONS)	Blue color, full sound
* Paper type (PRINT option)	Single sheets
* Page numbering (PRINT option)	No page numbers printed

The two sets of examples that follow—letters and reports, and lists and resumes—are illustrated, with explanations of how to achieve comparable results.

Lesson # 1—Letters and Reports

January 15, 1984

The Video Game Store
123 Main St.
Any City, U.S.A.

Dear Manager,

We would appreciate it if you would add our name to your mailing list. We are interested in receiving information about the latest games and educational programs available at your store. We're also interested in knowing when you will be having store-wide sales.

We would like to thank you for helping us select the ADAM Family Computer System. It's particularly in demand in the evening when everyone is usually home.

Sincerely,

The Adams Family

Figure 3-7-2

The top portion of a letter typed using the standard settings.

Products are also developed from comments made by insightful individuals. Not so long ago, a group of computer wizards put together what they thought was the best microcomputer available. Unfortunately, several other companies arrived at the completion of their systems at the same time.

Figure 3-7-3

Second chapter of a report, typed double-spaced, with margins of 15 and 65, automatic page numbering beginning with page number 8.

(This lesson continued on the next page.)

Many letters and reports can be typed using the standard settings. A typical letter typed with those settings is shown at the top of the opposite page.

To produce a standard letter or report, complete the following:

- * Turn the TV and SmartWRITER printer on, and press ESCAPE/WP.
- * Type the document. Use TAB to indent a line, and RETURN to end a line short or create a blank line.
- * Edit the document as you type, or when finished.
- * Store it, if appropriate. Print it. And clear it from the screen.

To type a letter or report which requires changing the system standards, complete the following:

- * Turn on the TV (or monitor) and ADAM and press ESCAPE/WP.
- * Change any of the MARGIN/TAB/ETC features (margins, tabs, line spacing) that you want to be special.
- * Type the document. Use TAB to indent a line, and RETURN to end a line short or create a blank line. As you type, change the MARGIN/TAB/ETC features again if necessary.
- * Edit the document as you type, or when finished.
- * Store it, if appropriate. Select any non-standard printing options (paper type, automatic page numbering) and print the document. When you have finished, clear it from the screen.

If you have stored a document on a digital data pack, you should consider how you will keep track of where it is stored. Even though each data pack has its own File Directory, you may have trouble finding the appropriate data pack if you have several of them—unless you have your own filing system.

You can, of course, recall or GET a stored file at any time. Then you can make additional edits or format changes. Once you have a document in memory (or workspace), you can print it any number of times.

Lesson # 2—Lists and Resumes

Expenses	1982	1983
Auto		
gas	552.00	623.00
insurance	350.00	362.00
maintenance	210.00	250.00
major repairs	525.00	450.00

Figure 3-7-4

List typed using the Moving Window Format, margins at 20 and 65, Tab stops for the second and third columns at 40 and 55, double-spaced, no page numbers.

RESUME		1
DALE L. OWENS		
Home Address		Business Address
111 Maple Ln.		1000 Main St.
Your City, State 00000		Your City, State 00000
Employment History		Job Responsibility
Photographer		Responsibility for producing
1981 - Current		4 major catalogs each year

Figure 3-7-5

Resume typed using Moving Window Format, single spacing, margins 15 and 70, Tab stop at 45, automatic page numbering beginning with page number 1.

Lists, resumes, charts, and other information which is typed in columns usually requires changing the system standards before you begin typing. These are the standards which are most frequently changed when typing column information.

- * Format—from Standard to Moving Window.
- * Margins—not always changed, but you should consider increasing left margin if document will be bound on the left side.

(This lesson continued on the next page.)

- * Tab stops—usually changed. Clear all the tab stops and then reset for your needs. If the first column is typed at the left margin, then tab stops are needed for each following column.
- * Line spacing—frequently, but not always changed. By increasing the spacing between lines, information in lists and charts is easier to read.
- * Page end—may be used to keep a chart or list on its own page. To do this, end the previous page with a mandatory PAGE END symbol and end the page on which the chart or list is typed with another mandatory PAGE END symbol.
- * Paper type—may be changed from single sheet to fanfold after pressing the PRINT key. Sometimes the paper length is changed so that longer charts can be put on one page. Be sure to change the PAPER TYPE on MARGIN/TAB/ETC if this is the case.
- * Automatic page numbering—may be selected if desired. This is selected after you press the PRINT key.

The settings used to type the documents on the opposite page are shown beneath the documents. These system changes are typical of those made when typing information in columns.

This concludes the SmartWRITER Word Processing lessons. Enjoy using SmartWRITER Word Processing on your ADAM computer.

Part IV

Introducing SmartBASIC



Getting Started (PRINT and GOTO)

How would you like to have a private servant? Wouldn't it be great to have someone who was always there and ready to help you?

This imaginary servant could organize your life. He or she could keep track of names, addresses, and phone numbers for you. This person could help you balance your checkbook or complete your homework. They could play games with you and even tutor you in a foreign language.

Now stop pretending. Believe it or not, you already have the servant we've been talking about. But it's not a person—it's your ADAM computer.

For your computer to do exactly what you want, you have to be able to talk to it in a language it will understand. Your ADAM computer doesn't understand English the way you and I do. Instead, it understands a special computer language called SmartBASIC.

How can you learn to speak SmartBASIC to your ADAM computer? Just read Part IV and Part V in this book.

As you might have noticed, both Part IV (Introducing SmartBASIC) and Part V (Doing More With SmartBASIC) are divided into six chapters. The first section of each chapter contains information for you to read and illustrations for you to look at. The second section contains practices for you to complete.

By reading each chapter in order, and then performing the practices, you can learn to talk SmartBASIC to your computer. This book won't teach you *everything* about your computer and SmartBASIC. It is, however, a good way to begin your computer education.

You don't have to have any computer experience to write SmartBASIC programs. In fact, once you have this book, you need only three other things before you begin.

First, you need an ADAM computer. Next, you need a television set or monitor that you will be using as a computer screen. Finally, you need the desire to learn. If you really want to learn how to talk to or "program" your computer—you can.

As you read earlier, your computer only understands the language called SmartBASIC. BASIC stands for *Beginner's All-purpose Symbolic Instruction Code*—or BASIC, for short.

Just like English, SmartBASIC has its own special words and rules. But don't worry, you'll find SmartBASIC easy to learn and understand.

Before you begin learning SmartBASIC, let's take a short guided tour of your computer keyboard. As you have probably noticed, most of the keys on your computer are like the keys on a typewriter. Normally, when you type, these keys will produce lower case letters on your screen. To produce upper case letters, hold down SHIFT and press the desired key. Look at the number keys (1, 2, 3, etc.) at the top of your keyboard. Normally, when you type, these keys will produce numbers on your screen. To produce the symbols that appear on the tops of these keys, hold down SHIFT and press the desired key.

Now, see if you can find the RETURN key on the right side of your computer. You might like to think of RETURN as your computer's "yes" key. Any time you want to tell your computer, "Yes, please do what I asked you to do," press RETURN.

Next, locate the SPACEBAR at the bottom of your computer. The SPACEBAR on your computer is just like the space bar on a typewriter. Everytime you want to type a blank space, press the SPACEBAR.

Finally, look at the cursor control keys on the right side of your keyboard. The four arrow keys can be used to move the cursor up, right, down, and left. In addition, when you are using SmartBASIC, press HOME to move your cursor to the upper left-hand corner of your screen.

In the practices at the end of this chapter, you will write your first program. Before you do that, however, you need to learn a little more about the SmartBASIC language.

As you read earlier, SmartBASIC uses special words. These words are known as "keywords" and look a lot like English words. For example, some of the keywords you will learn to use are: PRINT, RETURN, GOTO, RUN, LIST, LOAD, and SAVE.

You are going to be using keywords to write "programs" for your computer. A program is a specific set of instructions telling your computer exactly what you want it to do.

Just like any other language, SmartBASIC has some important rules. If you want your computer to understand what you're saying in your program, you have to follow these rules exactly. While these rules may be confusing at

first, don't worry. You will find everything you need to know in this book, and soon you'll be an expert!

You must follow two main rules when writing a computer program in SmartBASIC. First, instead of writing in sentences, you must write a program in numbered lines. Your computer will then read the lines you write in numerical order. Second, every line you write must begin with a keyword. You might like to think of these words as "action" words. When your computer sees one, it knows what "action" to perform.

Now that you know the two main rules of SmartBASIC, let's talk about the first three words you are going to be using in the lessons at the end of this chapter. These words are:

PRINT
GOTO
RUN

Illustration 4-1-2

Just like English, BASIC words have certain specific meanings. When your computer sees the word PRINT, it knows it must display what follows on your screen. When your computer sees the word GOTO, it knows to go to a different line. When your computer sees the word RUN, it knows to start using or "running" the program currently in its memory.

Now, before you write your first computer program, a quick word of warning. If you have experience using a typewriter, you might be used to using the number "0" and the letter "O" keys interchangeably. A typewriter might not know the difference between these keys but your computer does. It also knows the difference between the number "1" and the letter "l" keys. So be sure to use a number key when you want to type a number and a letter key when you want to type a letter.

On the next page, you will write your very first program. This program will only be two lines long. When you run the program, though, your computer will continue displaying your name until you tell it to stop.

Lesson # 1—Writing a Program

Complete the following steps to write your first computer program. NOTE: If you make a mistake and want to start a line over, press RETURN and type the line again. In the next chapter, you will learn an easier way to correct mistakes.

1. To begin using SmartBASIC, complete the following steps:
 - a. Turn on your TV or monitor.
 - b. Move the channel selector on your TV to 3 or 4—whatever channel you have selected on the back of the Memory Console.
 - c. On the switch box (connected to your TV or monitor), move the switch to "COMPUTER" or "GAME."
 - d. Turn on your computer by flipping the switch at the back of the printer.
 - e. Insert the SmartBASIC data pack into the drive. (NOTE: To insert the data pack correctly, make sure the writing on the data pack faces you and that the writing is right-side up.)
 - f. Pull the computer reset switch toward you. When your screen turns black and you see a bracket ([]) at the bottom of your screen, SmartBASIC has been "loaded" and you are ready to begin.

2. To write the first line of your program, complete the following steps:
 - a. To number the first line of your program, type **10** (NOTE: Be sure to use the number "1" key and the number "0" key—not the letter keys.).
 - b. Notice that the number 10 appears at the bottom of the screen. To type a blank space, press SPACEBAR.
 - c. Type **PRINT** (NOTE: You do not have to hold down SHIFT to type capital or upper case letters. Just type each key (p-r-i-n-t). Your computer will automatically change lower case letters (in keywords) to capital letters.).
 - d. Press SPACEBAR.
 - e. To type quotation marks ("), hold down SHIFT and press the "quote" key (located directly to the left of RETURN).
 - f. Type your first name.
 - g. To type another set of quotation marks, repeat step e.
 - h. To tell the computer, "Yes, this is what I want the first line of my program to look like," press RETURN.

(This lesson continued on the next page.)

3. To write the second line of your program, complete the following steps:
 - a. To number the second line of your program, type **20**
 - b. Press SPACEBAR.
 - c. Type **GOTO**
 - d. Press SPACEBAR.
 - e. To tell the computer what line you want it to go to, type **10**
 - f. To tell the computer this is what you want the second line of your program to look like, press RETURN.

Lesson # 2—Running a Program

Now that you have written your first computer program, it's time to see what it does. To do that, follow the steps below to "run" your program twice.

1. To run your program, complete the following steps:
 - a. Type **RUN**
 - b. To indicate that you want to run the program, press RETURN.
2. To stop your program, hold down the CONTROL key (located to the left of the SHIFT key) and press the "C" key.
3. To run your program again, repeat step 1.
4. To stop your program, repeat step 2.
5. This is the last lesson in this chapter. If you want to leave SmartBASIC at this time, complete the following steps:
 - a. Remove the SmartBASIC data pack from the data pack drive.
 - b. Place the the SmartBASIC data pack in its plastic case.
 - c. Do *not* place the data pack on or near any part of your computer system.
 - d. If you want to use another digital data pack, insert the data pack into the digital data pack drive and pull the computer reset switch towards you.
 - e. If you want to use a game cartridge, complete the steps listed under "Using Ready-to-Run Programs" (Part I, Chapter 3).
 - f. If you want to turn off your computer, flip the switch on the back of the printer. Then, turn off your TV or monitor. (NOTE: If you want to watch your TV, move the switch on the switch box to "TV" and use your television as you normally would.)

Reviewing What You Just Learned

You did it! You just wrote your very first computer program. Now that you have some experience, you probably have lots of questions.

Why did you number the program lines 10 and 20? Why not 1 and 2? Why didn't you see quotation marks (around your name) when you ran your program? To find the answers to these and other questions, read the next chapter. Then complete the lessons at the end of the chapter to write another program.

```
10 PRINT "Pat"  
20 GOTO 10
```

Making Changes (INPUT and REM)

The two-line program you wrote in the last chapter looked like the one illustrated on the opposite page. (By the way, we're going to need an imaginary person to use for examples and illustrations. Let's call this fictional person Pat Jones. Figure 4-2-1 is your first introduction to Pat. In your program, your name should appear instead of Pat's—unless, of course, your name happens to be Pat.)

The first question many new computer users ask is, why are the program lines numbered 10, 20, 30, etc.? Why not number them 1, 2, 3, etc.? The answer is simple.

It would be nice if the first time every computer program was written it was perfect. But that rarely happens. Usually, the life of a computer program involves many steps.

First, every program starts with an idea. Someone thinks of a new way to use a computer. This concept may be something as simple as using the computer to draw a simple picture or "graphic." It might also be as complicated as sending a vehicle to outer space. Most of the time, the idea will be somewhere in between these two extremes.

Next, the idea is turned into a computer program. Someone who speaks the same language as the computer (SmartBASIC in the case of your ADAM computer) writes step-by-step instructions for the computer to follow. Each step in this set of instructions will be short—like the lines in your first program. When the computer completes the steps in order, it should perform its new job.

After the new program has been typed into the computer, it goes on a "trial run." Though most of the program may be correct at this point, there are usually some additions and corrections to be made. That's where the additional line numbers come into the picture.

If the first line of the program has been numbered 10, and the second line 20, nine new lines can be added between the original first two lines. Because the computer will always read the lines in numerical order, the program will continue to run successfully.

Because few programs are perfect in their original form, it is common practice to number program lines in 10's. That way, additional lines can easily be added later.

Now let's look at what happened when you ran your first program. Your program contained the following two lines:

```
10 PRINT "Pat"  
20 GOTO 10
```

Illustration 4-2-2

Yet, when you ran the program, your name was constantly displayed on your screen.

```
Pat  
Pat  
Pat  
Pat  
Pat  
Pat
```

(etc.)

Illustration 4-2-3

There are two big differences between these two illustrations. First, when you wrote your program, you put quotation marks around your name. But, when you ran the program, your name appeared without quotation marks. That's because quotation marks act as signals when used with the keyword PRINT.

The first set of quotation marks tells your computer to start displaying what follows. The second set of quotation marks tells your computer to stop displaying text. Because quotation marks act as signals, they aren't displayed when the program is run. It is important to remember, however, that quotation marks should always appear in pairs. You'll learn more about this in the lessons at the end of this chapter.

Now for the second difference. Why (when you only typed it once) did your name appear continuously on your screen? The answer is the keyword GOTO.

In line 10, your program told your computer to display your name. In line 20, your program told your computer to go back to line 10. In line 10, your program displayed your name again. In line 20, the computer went back to line

10 again. In line 10 it displayed your name, in line 20, it went back to line 10, etc., etc., etc.

These two lines are an example of what is known as an "endless loop." An endless loop is a series of program lines that are repeated again and again and again. On your computer, your program stopped only when you held down CONTROL and pressed "C." Then the message "BREAK IN 10" or "BREAK IN 20" appeared at the bottom of your screen. This is your computer's way of telling you that when you told it to stop running the program it was on either line 10 or line 20.

Now that you understand your first program, let's discuss the next program you are going to write. This program will be 12 lines long. In the program, you will use two new keywords:

```
INPUT
REM
```

Illustration 4-2-4

When writing your own programs, INPUT (along with PRINT) will be one of the keywords you use the most. When your computer sees the word INPUT in a program, it stops and waits for the person running the program to type something and press RETURN.

INPUT is always followed by a letter or a letter and a dollar sign (\$). The characters following the keyword INPUT are called "variables." If the operator will be typing a number when the program is run, INPUT is followed by a letter. This letter is sometimes called a "numeric variable." If the operator will be typing letter(s) or word(s) when the program is run, INPUT is followed by a letter and a dollar sign. This letter and dollar sign are sometimes called a "string variable."

The other new keyword you will be using is REM. REM stands for "REMinder" or "REMark." Text that appears after REM will not appear on the screen when the program is run. In fact, when running a program, your computer will completely ignore any line that begins with REM. What is the purpose of REM? Normally, you will use REM for two main reasons.

First, every program you write should begin with a title line. An example of a title line appears in Illustration 4-2-5.

```
10 REM SPANISH QUIZ
```

Illustration 4-2-5

Later you will learn how this title line will be important. Right now you should get into the habit of beginning every program with a title line.

The second way you might use REM is as a private notebook. If, when writing a program, you want to include information that you don't want the person using the program to see, store the information on a line following the keyword REM. For example, the first four lines of a program might look like this:

```
10 REM SPANISH QUIZ
20 REM CREATED 6/12/84
30 REM WRITTEN BY PAT JONES
40 PRINT "This is a Spanish quiz."
```

Illustration 4-2-6

But, when these four lines are run, the operator will only see this:

```
This is a Spanish quiz.
```

Illustration 4-2-7

Besides using REM and INPUT in your next program, you are going to be using the keyword PRINT in two new ways. Instead of telling you exactly what's going to happen, see if you can discover these new ways for yourself. Just follow the directions carefully and you will write a perfect program.

If you make a mistake, you can correct your program using three different methods. As you read in the last chapter, you can just retype incorrect lines.

Another method of correcting mistakes is to use arrow keys to position the cursor over a mistake. Then, beginning at the first incorrect character in the line, retype the line. When the line is correct, position the cursor at the end of the line and press RETURN.

The final method of correcting mistakes is used to erase or "delete" an entire line. When you want to erase a program line that already exists, type the line number and press RETURN. Your computer will erase the line you indicated from its memory (though the line will still appear on your screen).

Lesson #1—Writing a Program

Complete the following exercise to write your second computer program. If your first program is still in the computer's memory, don't worry. The previous program lines will be replaced as you enter your next program. NOTE: If you need help in loading SmartBASIC, consult lesson #1 at the end of the previous chapter.

1. To write the first line of your program, do the following:
 - a. To number the first line of your program, type **10** (NOTE: Be careful to use the number "1" key and not the letter "l" key. Also, be careful to use the number "0" key and not the letter "O" key.).
 - b. Type **REM**
 - c. Press SPACEBAR.
 - d. Type the program title, which is **DEMO**
 - e. To tell the computer, "Yes, this is what I want the first line of my program to look like," press RETURN.

2. To write the second line of your program, do the following:
 - a. To number the second line of your program, type **20**
 - b. Type **PRINT**
 - c. To type quotation marks ("), hold down SHIFT and press the "quote" key (located directly to the left of RETURN).
 - d. Type the following text: **Hello. I am your ADAM**
 - e. To type another set of quotation marks, repeat step c.
 - f. To tell the computer, "Yes, this is what I want the second line of my program to look like," press RETURN.

3. To write the third line of your program, do the following:
 - a. To number the third line type **30**
 - b. Type **PRINT**
 - c. Type quotation marks (Hold down SHIFT and press the "quote" key).
 - d. Type the following text: **Family Computer.**
 - e. Repeat step c.
 - f. To tell the computer, "Yes, this is what I want the third line of my program to look like," press RETURN.

(This lesson continued on the next page.)

4. To write the fourth line, do the following:
 - a. Type **40**
 - b. Type **PRINT**
 - c. Press RETURN (NOTE: This is the special use of PRINT that was mentioned earlier. You will learn more about PRINT, and what happens when it is used all by itself in a program line, in the next chapter.).

5. To write the fifth line:
 - a. Type **50**
 - b. Type **PRINT**
 - c. Type quotation marks ("").
 - d. Type the following text: **What is your name?**
 - e. Repeat step c.
 - f. To tell the computer this is the fifth line, press RETURN.

6. To write the sixth line:
 - a. Type **60**
 - b. Type **PRINT**
 - c. Press RETURN (NOTE: This, again, is the special use of PRINT. When you run the program you will see what happens to this and other lines that only contain the keyword PRINT.).

7. To write the seventh line:
 - a. Type **70**
 - b. Type **PRINT**
 - c. Type quotation marks ("").
 - d. Type the following text: **Type your name and press RETURN**
 - e. Repeat step c.
 - f. Press RETURN.

(This lesson continued on the next page.)

8. To write the eighth line:
 - a. Type **80**
 - b. Type **INPUT**
 - c. Press SPACEBAR.
 - d. Type the variable to follow INPUT, in this case, **n\$**
 - e. Press RETURN.

9. To write the ninth line:
 - a. Type **90**
 - b. Type **PRINT**
 - c. Press RETURN.

10. To write the tenth line:
 - a. Type **100**
 - b. Type **PRINT**
 - c. Press RETURN.

11. To write the eleventh line:
 - a. Type **110**
 - b. Type **PRINT**
 - c. Type quotation marks (").
 - d. Type the following text: **Hi**
 - e. To type a blank space, press SPACEBAR.
 - f. Type quotation marks (").
 - g. Type a semicolon (;).
 - h. Type **n\$**
 - i. Type a semicolon (;).
 - j. Type quotation marks (").
 - k. Type the following text: **. I hope we** (NOTE: Press SPACEBAR twice after typing the period (.) and before typing the rest of the text.)
 - l. Type quotation marks (").
 - m. Press RETURN.

(This lesson continued on the next page.)

12. To type the last line of the program:

- a. Type **120**
- b. Type **PRINT**
- c. Type quotation marks (").
- d. Type the following text: **can be friends.**
- e. Type quotation marks (").
- f. Press RETURN.

Lesson # 2—Running the Program

Complete the steps below to run the program you have just written.

1. Type **RUN**
2. To tell your computer that you want to run the program, press RETURN.
3. Follow the directions that appear on the screen.
4. When you reach the end of the program, repeat steps 1-3. This time, however, type the name of a friend instead of your name.
5. This is the last lesson in this chapter. If you want to stop using SmartBASIC at this time, consult lesson #2 at the end of the previous chapter.

Reviewing What You Just Learned

In this chapter you learned more about using the keyword PRINT to display information. For example, the first set of quotation marks (after PRINT) tells your computer to start displaying what follows. The second set of quotation marks (after PRINT) tells your computer to stop displaying text. Because quotation marks act as signals, they aren't displayed when the program is run. It is important to remember, however, that quotation marks should always appear in pairs.

You also learned about two new keywords: INPUT and REM. INPUT is always followed by a letter or a letter and a dollar sign (\$). The characters following the keyword INPUT are called "variables." REM stands for "REMinder" or "REMark." Text that appears after REM will not appear on the screen when the program is run.


```
10 REM DEMO
20 PRINT "Hello. I am your ADAM"
30 PRINT "Family Computer."
40 PRINT
50 PRINT "What is your name?"
60 PRINT
70 PRINT "Type your name and press RETURN."
80 INPUT n$
90 PRINT
100 PRINT
110 PRINT "Hi ";n$;". I hope we"
120 PRINT "can be friends."
```

3

Adding it Up (+, -, *, /)

The title of this chapter has two meanings. As you might have guessed, later in this chapter you will be learning how to use your computer to solve math problems. That's one of the meanings. But, before you go that far, you should understand exactly what happened in your last program and why. That's the other meaning: let's add up everything you've learned so far.

The program you wrote at the end of the last chapter looked like the one illustrated on the opposite page. When you ran the program, this is what appeared on your screen:

```
Hello. I am your ADAM
Family Computer.

What is your name?

Type your name and press RETURN.
Pat

Hi Pat. I hope we
can be friends.
```

Illustration 4-3-2

You already know two reasons why these illustrations are different. First, your computer ignores any lines beginning with the keyword REM. That's why line 10 didn't appear when the program was run. Second, a first set of quotation marks (after the keyword PRINT) tells your computer to start displaying text. A second set of quotation marks tells your computer to stop displaying text. That's why none of the quotation marks appeared when the program was run. Remember, your computer uses quotation marks as signals.

Some of the other differences might be a little more confusing. First, what happened to line 40? When you wrote the program, it contained only the keyword PRINT. But, when you ran the program, a blank line appeared between lines 30 and 50. As you have probably guessed, line 40 is an example of using PRINT to create a blank line.

If, when writing a program, you want to add a blank line to your screen, just use the keyword PRINT all by itself. In the illustrations on the two previous pages, you can see that this is what happened in lines 40, 60, 90, and 100.

Do you think you know what happened in line 80? As you read in Chapter 2, when your computer sees the keyword INPUT, it stops and waits for the person running the program to type information and press RETURN. As you might remember, when you were running your program and you reached line 80, your screen looked like this:

```
Hello. I am your ADAM
Family Computer.

What is your name?

Type your name and press RETURN.

?
```

Illustration 4-3-3

As you can see, when your computer encounters the keyword INPUT, a question mark appears on your screen. This is a signal that the computer is expecting you to type something.

Once you typed your name and pressed RETURN, your computer assigned your name to the characters n\$. In other words, every time your computer saw the characters n\$, it substituted your name. When you ran the program again and typed the name of a friend, your computer assigned the new name to the characters n\$. Then, when your computer saw n\$ again, it substituted your friend's name.

As you can see, the meaning assigned to the characters n\$ can be different every time the program is run. That's why the character(s) following the keyword INPUT (in our example, n\$) are called "variables." Their meaning may be different or "vary."

Speaking of variables, let's look at line 110. That's where this program's variable really goes to work. This line, as it appears in the program and when the program is run, is illustrated below.

```
[PROGRAM] 110 PRINT "Hi ";n$;" . I hope we"
[RUN]     Hi Pat. I hope we
```

Illustration 4-3-4

Obviously, the line begins by displaying the word "Hi" followed by a blank space. But what's that semicolon doing there?

Three punctuation marks—commas, semicolons, and colons—have special meanings on your computer. All three marks or "symbols" can be used two different ways. Inside quotation marks, commas, semicolons, and colons can be used in the same way that you would use them when typing or writing; as punctuation symbols. Outside quotation marks, these three symbols have different meanings.

Both commas (,) and semicolons (;) are used in PRINT statements. Outside quotation marks, a comma tells your computer to display what follows beginning at the middle of your screen. If information already displayed past the middle of the screen, a comma tells your computer to display what follows beginning at the left edge of the next line. Outside quotation marks, a semicolon (;) tells your computer to display what follows without adding any blank spaces.

A colon (:) can be used to join several statements together on one line. For example, in Illustration 4-3-5, two PRINT statements have been joined together on one line.

```
[PROGRAM]      20 PRINT "Hello.":PRINT "How are you?"
[RUN]          Hello. How are you?
```

Illustration 4-3-5

You will find that commas, semicolons, and colons are important in programs. They will save you time entering some programs. For example, if you want something to be displayed at the middle of a line, you don't have to press SPACEBAR repeatedly. Instead, all you have to do is type a comma and the desired information. In other programs, commas and semicolons will be so important that the program won't work successfully without them. For example, if you had left the semicolons out of line 100 in your last program, the information would have been separated onto three different lines when the program was run.

Now that you understand how semicolons were used in line 110, there are two other things you should notice about the program in Figure 4-3-1. First, there are no quotation marks around the variable that appears in this line. Whenever you want to display numbers, letters, words, or other characters that have been assigned to a variable in your own programs, do not surround the variable with quotation marks. In other words, if you wanted to change line 110 so that it only displayed the name typed in line 80 and nothing else, you would write this:

```
110 PRINT n$
```

Illustration 4-3-6

The final thing you should notice is the way spaces have been used in this line. If you had typed the line without a space after the word Hi, like this:

```
110 PRINT "Hi";n$". I hope we"
```

Illustration 4-3-7

When running the program, you would have seen this:

```
HiPat. I hope we
```

Illustration 4-3-8

As you can see from this example, if the correct number of blank spaces are not included within quotation marks, you will end up running some words together. When your computer displays the text assigned to a variable, it does not display any extra spaces before or after that text. If you want blank spaces to separate words or sentences, they have to be included within quotation marks.

Now, on to mathematics.

YOU USE:	FOR:	YOUR COMPUTER USES:
+	addition	+
-	subtraction	-
X	multiplication	*
÷	division	/

Illustration 4-3-9

As you can see, your computer uses two symbols that you should be familiar with and two that might be new to you. For addition, your computer uses the plus sign (+). For subtraction, your computer uses the minus sign (-). Instead of an X, your computer uses an asterisk (*) for multiplication. And, to divide two numbers, your computer uses a slash (/).

As you've probably noticed, your computer is a very logical machine. It likes to do things in a neat, orderly way—and it solves math problems in the same manner.

When your computer sees a math problem, the first thing it does is look for a set of parentheses, like this: (). If it finds parentheses, it performs the math problem or "operation" inside the parentheses before doing anything

else. For example, if you ask your computer to solve the problem $2*(3+4)$, it will tell you that the answer is 14; that is, $3+4=7$ and $2*7=14$.

If there is more than one set of parentheses in a problem, your computer will begin by solving the operation within the inner set of parentheses. Then it will solve the problem in the next largest set of parentheses, followed by the problem in the next outer set, and so on. For example, if you ask your computer to solve the problem $2*((3+4)-1)$, it will tell you that the answer is 12; that is, $3+4=7$, $7-1=6$, and $2*6=12$.

After your computer solves all the operations within parentheses, it performs all multiplication and division problems, working from left to right. Finally, it performs all addition and subtraction problems, again working from left to right.

The order in which your computer performs math operations is illustrated below.

- (1) OPERATIONS WITHIN PARENTHESES
- (2) MULTIPLICATION & DIVISION
- (3) ADDITION & SUBTRACTION

Illustration 4-3-10

Besides being very logical, you've probably noticed that your computer is very consistent. It will always perform mathematical operations in the order illustrated above. Though it may be difficult to get used to this order at first, you will soon find it easy to use and understand.

The important thing to remember is that your computer will always perform operations within parentheses first. Then, it will perform multiplication and division. Finally, it will add and subtract. So, when writing your own computer programs, you must be careful to use parentheses correctly. For example, if you ask your computer what $4*3+2$ equals, it will tell you that the answer is 14; that is, $4*3=12$ and $12+2=14$. However, if you change the problem to $4*(3+2)$, the computer will say that the answer is 20; that is, $3+2=5$ and $4*5=20$.

As you might have guessed, INPUT and PRINT statements often appear in programs with mathematical operations. For that reason, it is important to understand how INPUT, PRINT, and mathematics work together on your computer.

As you read in the last chapter, when the person running the computer program is going to be typing a number, the program contains the word

INPUT followed by a single letter. This letter is known as a variable. For example, in the program line illustrated below, the letter "a" is the variable.

```
30 INPUT a
```

Illustration 4-3-11

Once a computer operator types a number and presses RETURN, the variable is automatically replaced (with the number the operator typed) throughout the rest of the program. For example, let's say the operator typed the number 25 in the line illustrated above. Later in the program you might have a line like this:

```
60 PRINT a+3
```

Illustration 4-3-12

When the program is run, and it reaches line 60, this is what would appear on the screen:

```
28
```

Illustration 4-3-13

As shown, your computer would substitute the number 25 for the variable a. Then it would print the sum of $25 + 3$, or 28. If we changed line 60 to read:

```
60 PRINT "a+3"
```

Illustration 4-3-14

This is what you would see when the program was run:

```
a+3
```

Illustration 4-3-15

In other words, when a variable follows the keyword PRINT and is not surrounded by quotation marks, it will be treated as a number. On the other hand, when a variable follows the keyword PRINT and is surrounded by quotation marks, it will be treated as regular text.

The lessons at the end of this chapter will cover these topics in detail. But before you perform the exercises, you need to learn a new keyword—HOME. To understand what HOME does, you first have to understand how your computer uses your television or monitor.

As you know, your ADAM computer uses your television or monitor as a computer screen. No matter how big your television is, your computer divides it into 24 horizontal lines. When running a program, the bottom two lines of the screen will always remain blank. That leaves 22 lines available to display information. When you ran the program you wrote in chapter 1, you saw what happens when these 22 lines get full. Your computer just pushes information at the top of the screen out of view and moves new information onto the bottom. This process is known as "scrolling."

While scrolling didn't cause a problem in any program you have written so far, it might. To break your program into sections or "screenloads" that contain all the information you want the user to see at any given moment, you can use the keyword HOME. When your computer sees the keyword HOME, it erases the screen and moves the cursor to the upper left-hand corner of the screen. HOME can be used both within a program line (where it would be called a HOME "statement") and by itself (where it would be called a HOME "command").

Now, before you continue, let's look at some things that should help you complete the lessons at the end of this chapter. First, it's not unusual to make mistakes when typing a computer program. Though your computer consistently follows a strict set of rules, it knows that you sometimes make mistakes. To tell you that you have made an error, your computer uses special error messages.

```
20 PRINT "Hello
      ' " ' Expected ^
```

Illustration 4-3-16

When you press RETURN after typing a line with a mistake, an error message will appear. A caret (^) will point to the exact location of the mistake. For example, in the illustration above, the message indicates that line 20 should end with quotation marks.

If an error message appears on your screen, don't panic. Making mistakes is part of learning. An error message is just your computer's way of pointing out your mistakes. When you see an error message, just retype the line and continue entering the program.

Of course, sometimes you don't see a mistake until you use your program. For example, if you have misspelled a word in a PRINT statement, you may not realize it until you are running your program. To correct a line in this situation you would first have to use the LIST command. LIST is used to display program lines in numerical order.

For example, if after running a program, you want to see the program lines again, just type LIST and press RETURN. Your computer will display all the program lines again, in numerical order. The ways you can use LIST are illustrated below.

TO SEE:	TYPE:
All lines	LIST
One line (e.g., 20)	LIST 20
A group of lines (e.g., 20, 30, 40)	LIST 20-40 or LIST 20,40

Illustration 4-3-17

After you have used LIST to display program lines, you can use any method (explained at the end of chapter 2) to correct your program.

Lesson #1—Writing a Program

In this exercise you are going to copy a program that is printed below. You won't have step-by-step instructions to follow. Instead, you will have to remember the things you've learned in this and previous chapters. Entering your first program, all by yourself, can be a scary experience. You shouldn't have any problems, though, if you follow these simple rules:

- * Be careful to type the program exactly as it appears. Quotation marks and semicolons may be small, but they are very important in a computer program.
 - * Be patient. Typing a program into a computer takes time. If you try to rush the process, you'll probably end up making mistakes.
1. Type the program that appears below in Illustration 4-3-18.

```

10 REM DEMO
20 PRINT "Hello. I am your friendly"
30 PRINT "ADAM computer."
40 PRINT:PRINT
50 PRINT "I am very good at solving"
60 PRINT "math problems. To see"
70 PRINT "if I am as good as I say"
80 PRINT "I am, please type a number"
90 PRINT "and press RETURN."
100 INPUT a
110 PRINT:PRINT
120 PRINT "Thank you, now type another"
130 PRINT "number and press RETURN."
140 INPUT b
150 HOME
160 PRINT
170 PRINT "Thank you, now I can say"
180 PRINT a;" + ";b;" = ";a+b;"."
190 PRINT:PRINT
200 PRINT "And, did you know that"
210 PRINT a;" X ";b;" = ";a*b;"?"
220 PRINT
230 PRINT "See you later."

```

Illustration 4-3-18

Lesson # 2—Running the Program

Complete this exercise to run the program currently stored in your computer's memory.

1. Type **RUN**
2. Press RETURN.
3. Follow the directions that appear on your screen. (If the information on your screen is unclear, recheck your program and correct errors.)
4. When you reach the end of the program, repeat steps 1-3 to run the program again. This time, however, type different numbers. NOTE: In the next chapter you will get a chance to save a copy of the program you just wrote. So, turn off your computer only if you are not going on to chapter 4 at this time.

Reviewing What You Just Learned

The symbols used by your computer to perform mathematical operations are illustrated below.

YOU USE:	FOR:	YOUR COMPUTER USES:
+	addition	+
-	subtraction	-
×	multiplication	*
÷	division	/

Illustration 4-3-19

The order in which your computer performs math operations is illustrated below.

- (1) OPERATIONS WITHIN PARENTHESES
- (2) MULTIPLICATION & DIVISION
- (3) ADDITION & SUBTRACTION

Illustration 4-3-20

INPUT (followed by a variable) and PRINT are often used together in programs. When a variable follows the keyword PRINT and is not surrounded by quotation marks, it will be treated as a number. On the other hand, when a variable follows the keyword PRINT and is surrounded by quotation marks, it will be treated as regular text.

DIGITAL DATA PACK DRIVE



4

Storing Your Programs (SAVE and LOAD)

By reading the last three chapters you have learned a great deal. You have learned how to use statements in numbered lines to communicate with your ADAM computer. You have learned how to use INPUT along with variables to allow a user to enter his or her own information into a program. And, most important, you've learned that writing a computer program takes time.

The program you copied in the last chapter is, by most standards, short. Yet it took you some time to type. If, every time you wanted to use your computer you had to type a program, you would end up wasting a lot of time.

For that reason, computers always have some way of storing programs. Once a program has been stored, it can easily be recalled and used again. Different computer systems use different methods of storing programs. On your ADAM computer, you use a digital data pack to store programs.

Before you store a computer program, you need to understand some introductory computer concepts. First, an "electronic copy" of any program in your computer's memory can be recorded onto a digital data pack. This procedure is known as "saving" a program. After you save a program, it exists in two places. One copy remains in your computer. Another copy is stored on the digital data pack.

After you have stored a program, you can move a copy of the program back into your computer's memory at any time. This procedure is known as "loading" a program. After you load a program, it exists in two places. One copy remains on the digital data pack. Another copy is in your computer.

As outlined in the above procedure, you will use two new keywords: SAVE and LOAD. Because you will use these words by themselves (and not within numbered lines), SAVE and LOAD (like RUN) are known as "commands."

To tell your computer what program you want it to save or load, you use a program title (sometimes called a "file name"). This should be the title stored in the REM statement at the beginning of the program. That's why a title is important and should be included in every program you write. You may think that you will remember the name of every program you write.

But, once you begin to build a library of your own programs, it is easy to get confused.

Since the best way to learn about saving and loading programs is by doing it yourself, you will practice these procedures on the next page. Before you begin the lesson, make sure the blank digital data pack that came with your computer is available.

Lesson # 1—Saving a Program

Currently, you should have a program stored in your computer's memory. If you don't, type the program illustrated below to use in these lessons.

```
10 REM DEMO
20 PRINT "This is a program to see"
30 PRINT "how SAVE and LOAD work."
```

Illustration 4-4-2

Now, complete the steps below to store the program currently in your computer's memory.

1. Remove the SmartBASIC data pack from the drive.
2. Insert a blank data pack into the drive.
3. To save your program, complete the following steps:
 - a. Type **SAVE**
 - b. Type the program name, which in this case is **DEMO**
 - c. Press RETURN.
4. Listen to your digital data pack drive. You can hear it "whirring." When your cursor (|) reappears on the bottom of the screen, your program has been saved. If you want, remove the digital data pack from the drive and re-insert the SmartBASIC data pack.

Lesson #2—Clearing Your Computer's Memory

Now that you have saved this program, it exists in two places. One copy is on the digital data pack. Another copy remains in your computer's memory. You can use the NEW command to erase or "clear" your computer's memory. Follow the steps below to use NEW.

1. Type **NEW**
2. Press RETURN (NOTE: The program will still appear on the screen although it has been erased from your computer's memory. To make sure your program is no longer in your computer's memory, try "running" it.).

Lesson # 3—Loading a Program

Once a program has been saved, it can be "loaded" back into your computer at any time. Follow the steps below to load the program you saved earlier.

1. If you haven't already done so, remove the SmartBASIC data pack from the drive.
2. Insert the data pack containing the file you want to load into the drive. (NOTE: To see a list of the files stored on the digital data pack, type CATALOG and press RETURN.)
3. To load your program, complete the following steps:
 - a. Type **LOAD**
 - b. Type the program name, which in this case is **DEMO**
 - c. Press RETURN.
4. Listen to your digital data pack drive. You can hear it "whirring." When your cursor (I) reappears on the bottom of your screen, your program has been loaded.

Reviewing What You Just Learned

Two commands, SAVE and LOAD, allow you to store and recall programs.

After you save a program, it exists in two places. One copy remains in your computer. Another copy is stored on the digital data pack.

After you load a program, it exists in two places. One copy remains on the digital data pack. Another copy is in your computer.

COLOR	NUMBER
Black	0
Magenta	1
Dark Blue	2
Dark Red	3
Dark Green	4
Grey-1	5
Medium Green	6
Light Blue	7
Light Yellow	8
Medium Red	9
Grey-2	10
Light Red	11
Light Green	12
Light Yellow	13
Cyan (blue)	14
White	15

Color a Rainbow (GR, COLOR=, PLOT, VLIN, and HVLIN)

In the last four chapters, you have been using your ADAM computer in what is known as the text or "narration" mode. Anything you typed appeared in white letters on a black background.

Believe it or not, you have just begun to see what ADAM can do! There are two different ways you can draw pictures (or graphics) with your computer. The subject of this chapter is low resolution graphics. In Part 5 (Chapter 5) you will find a discussion of high resolution graphics.

To tell ADAM you want to change from the text mode to the low resolution graphics mode, you must use a GR statement. Then, using a COLOR= statement, you can choose one of the 16 colors listed on the opposite page. The color you select will be the color with which you can draw lines and shapes. For example, in the illustration below, line 10 indicates that the program will be in low resolution graphics. Line 20 indicates that the program will begin by drawing shapes in light blue.

```
10 GR
20 COLOR=7
```

Illustration 4-5-2

GR (like HOME) can be used as either a statement (within a program line) or as a command (by itself). Whenever your computer sees GR, it knows you want it to go into the low resolution graphics mode. (NOTE: If you want to leave either low resolution or high resolution graphics mode, use the keyword TEXT. Like GR, TEXT can be used both as a statement and a command. Whenever your computer sees TEXT, it knows you want it to go into the text or narration mode.)

Before you can learn to draw shapes in the low resolution graphics mode, you need to understand how your computer divides your screen. In low resolution graphics, your screen is divided into 40 columns (numbered 0-39, starting at the left edge of your screen) and 40 rows (numbered 0-39, starting at the top of your screen). Four lines are reserved for text at the bottom of the low-resolution graphics screen.

Once you have typed a GR statement and selected a color using a COLOR= statement, you are ready to begin "drawing" using three keywords: HLIN, VLIN, and PLOT.

HLIN stands for horizontal line. HLIN is used to draw horizontal lines in any row (0-39) in the low resolution graphics screen. HLIN must always be followed by: a space, the number of the first column in which the line is to appear, a comma (,), the number of the last column in which the line is to appear, a space, the word AT, a space, and the number of the row in which the line is to appear. In other words, the illustration below shows a light blue line that is to be drawn between columns 0 and 9, in row 10.

```
10 GR
20 COLOR=7
30 HLIN 0,9 AT 10
```

Illustration 4-5-3

VLIN stands for vertical line. VLIN is used to draw vertical lines in any column (0-39) in the low resolution graphics screen. VLIN must always be followed by: a space, the number of the first row in which the line is to appear, a comma (,), the number of the last row in which the line is to appear, a space, the word AT, a space, and the number of the column in which the line is to appear. In other words, the illustration below shows a light blue line that is to be drawn between rows 0 and 9, in column 10.

```
10 GR
20 COLOR=7
30 VLIN 0,9 AT 10
```

Illustration 4-5-4

Many times you will want to do more than draw lines in the low resolution graphics mode. You'll want to draw shapes and figures. The keyword PLOT allows you to "color" the intersection of columns and rows. Each time you use PLOT, a small block appears on your screen. By using several PLOT statements, you can create shapes and figures. PLOT should always be followed by a column number (0-39), a comma (,), and a row number (0-39). For example, if you ran the program in the illustration below, two light green squares would appear on your screen. One square would be in the upper left-hand corner of your screen (the intersection of column 0 and row 0). The other square would be in the lower right-hand corner of your screen (the intersection of column 39 and row 39).


```
10 GR
20 COLOR=12
30 PLOT 0,0
40 PLOT 39,39
```

Illustration 4-5-5

Now that you know how GR, HLINE, VLINE, and PLOT work together, perform the lessons at the end of this chapter to practice using the low resolution graphics mode.

Lesson # 1—Using HLIN

Complete the following steps to write and run a short program using HLIN.

1. Enter the program that appears below in Illustration 4-5-6. (NOTE: Row and column numbers do not appear on the screen.)

```
10 GR
20 COLOR=4
30 HLIN 0,39 AT 10
```

Illustration 4-5-6

2. Run the program that you typed in step 1. Notice the color of the line that appears on your screen. Also, notice where the line is drawn (between columns 0 and 39 in row 10).
3. To see your program listing again, complete the following steps:
 - a. Type **TEXT**
 - b. Press RETURN.
 - c. Type **LIST**
 - d. Press RETURN.
4. Add the following lines to your program:

```
40 PRINT "This is a demonstration of"
50 PRINT "HLIN."
```

Illustration 4-5-7

5. Run the program again.
6. Repeat step 3 to see your program listing.

Lesson #2—Using VLIN

Complete the following steps to write and run a short program using VLIN.

1. Type the two new lines shown below.

```
30 VLIN 0,39 AT 10  
50 PRINT "VLIN."
```

Illustration 4-5-8

2. To see your program lines in correct order, complete the following steps:
 - a. Type **LIST**
 - b. Press RETURN.
3. Run this new version of your program. Notice the color of the line that appears on your screen. Also, notice where the line is drawn (between rows 0 and 39 in column 10).
4. To see your program listing again, complete the following steps:
 - a. Type **TEXT**
 - b. Press RETURN.
 - c. Type **LIST**
 - d. Press RETURN.

Lesson #3—Using PLOT

Complete the following steps to write and run a short program using PLOT.

1. Type the two new lines shown below.

```
30 PLOT 10, 15  
50 PRINT "PLOT."
```

Illustration 4-5-9

2. To see your program lines in correct order, complete the following steps:
 - a. Type **LIST**
 - b. Press RETURN.
3. Run this new version of your program. Notice the color of the square that appears on your screen. Also, notice where the square is drawn (at the intersection of column 10 and row 15).
4. To see your program listing again, complete the following steps:
 - a. Type **TEXT**
 - b. Press RETURN.
 - c. Type **LIST**
 - d. Press RETURN.
5. If you want, add lines to this program to “plot in” other areas of your screen. You can also add **COLOR=** statements if you want to plot in different colors. For example, if the next line in your program was **COLOR=11**, any PLOT statement that followed this line would appear in a light red color when the program was run.

Reviewing What You Just Learned

To tell ADAM you want to change from the text mode to the low resolution graphics mode, you must use a GR statement. Then, using a COLOR= statement, you can choose one of 16 colors. The color you select will be the color with which you can draw lines and shapes. You can "draw" in low resolution graphics using HLIN, VLIN, and PLOT.

HLIN can be used to draw horizontal lines in any row (0-39). HLIN must always be followed by: a space, the number of the first column in which the line is to appear, a comma (,), the number of the last column in which the line is to appear, a space, the word AT, a space, and the number of the row in which the line is to appear.

VLIN can be used to draw vertical lines in any column (0-39). VLIN must always be followed by: a space, the number of the first row in which the line is to appear, a comma (,), the number of the last row in which the line is to appear, a space, the word AT, a space, and the number of the column in which the line is to appear.

PLOT allows you to "color" the intersection of columns and rows. Each time you use PLOT, a small block appears on your screen. PLOT should always be followed by a column number (0-39), a comma (,), and a row number (0-39).

```

10 REM MILEAGE
20 PRINT "This program determines how"
25 PRINT "many miles per gallon you"
30 PRINT "get while driving."
40 PRINT
50 PRINT "To begin, what was your"
55 PRINT "mileage reading the last time"
60 PRINT "you put gas in your car?"
70 PRINT
80 PRINT "Please type the correct number"
90 PRINT "and press RETURN. (NOTE: Do"
100 PRINT "not use commas.)"

```

1. Now, please type "you are in years" RETURN."

"in\$;"," DAM computer and I" INTRO T "Please

```

110 PRINT
120 INPUT a
130 PRINT:PR
140 PRINT "W
145 PRINT "t
150 PRINT
160 PRINT "P
170 PRINT "a
180 PRINT
190 INPUT b
200 HOME
210 PRINT "
220 PRINT "
230 PRINT
240 PRINT "
250 PRINT
260 PRINT
270 INPUT
280 HOME
290 PRINT:
300 PRINT
310 PRINT
320 PRINT
330 PRINT

```

```

10 REM MATH DEMO
20 PRINT "Please type your name and"
30 PRINT "press RETURN."
40 INPUT n$
50 PRINT:PRINT
60 PRINT "Hello ";n$;. This is a mat
70 PRINT "demonstration program."
80 PRINT
90 PRINT "If you will give me 2 number
95 PRINT "I will show you how good I"
100 PRINT "am at math."
110 PRINT
120 PRINT "To begin, type a number and"
130 PRINT "press RETURN. (Please do"
135 PRINT "not use commas.)"
140 INPUT a
150 PRINT
160 PRINT "Now type another number"
170 PRINT "and press RETURN."
180 INPUT b
190 HOME
200 PRINT
210 PRINT a;" + ";b;" = ";a+b
220 PRINT
230 PRINT a;" X ";b;" = ";a*b
240 PRINT
250 PRINT a;" - ";b;" = ";a-b
260 PRINT
270 PRINT a;" / ";b;" = ";a/b
280 PRINT:PRINT
290 PRINT "Well, what do you think "
300 PRINT n$;"? I hope I can help"
310 PRINT "you with math problems in"
320 PRINT "the future."
330 PRINT:PRINT
340 PRINT "See you later ";n$;."

```

type your name and" "TURN."

Now, please type" a friend and"

```

210 PRINT "Now, how m
220 PRINT "did you us
230 PRINT "Please type
240 PRINT "and press R
250 PRINT
260 PRINT
270 INPUT c
280 HOME
290 PRINT:PRINT
300 PRINT "Based on the i
310 PRINT "me you got --
320 PRINT
330 PRINT b-a/c;" MILES PE

```

graphics demon-" To begin, type" nd press RETURN."

```

PRINT
PRINT "ow old y
PRINT "and press
INPUT a
HOME
PRINT
PRINT "As you know
PRINT "I am your f
PRINT "can do a lo
PRINT "For example,
PRINT "organize you
PRINT "good at math and can eve

```

```

180 HOME
190 COLOR=3
200 HLINE 15, 20 AT 25
210 VLINE 15, 20 AT 35
220 VLINE 25, 35 AT 15
230 VLINE 25, 35 AT 20
240 PRINT "I hope you have enjoyed"
250 PRINT "this demonstration ";n$;."
260 PRINT "simple and fun. Before long"
270 PRINT "you will be showing ";f$
280 PRINT "how easy it is to use ADAM."
290 PRINT
300 PRINT "Good-bye ";n$;. See

```

or you. you" m very" 1"

6

Try These

If you have read the last five chapters, and performed the lessons at the end of each chapter, you are ready to begin writing your own computer programs. Writing a computer program requires both skill and creativity.

By performing the exercises in this book, you have acquired the basic skills necessary to write simple programs for your ADAM computer. The subject of this chapter is the other thing you will need to write a successful computer program—creativity.

First, if you don't want to write your own programs, don't worry. There are plenty of programs available in magazines and books (see the programs in this chapter and the programs in Part V, Chapter 6. You can also purchase ready-to-run software for your ADAM Family Computer System. In fact, if you never write a single computer program, you will still be able to put your computer to good use.

So why go to the trouble of writing your own programs? There are three main reasons.

First, as you have seen in the last five chapters, writing programs helps you understand how your computer works. The more programs you write, the better computer user you will become.

Second, many people find programming a pleasant and productive hobby. They enjoy the challenge of learning a computer language and creating programs.

Finally, if you write your own computer programs, you can design them to match your exact specifications. Just like people who sew their own clothes or build their own furniture, you can create computer programs to meet your individual needs.

If you want to write your own programs, where do you start? A good place to begin is by looking at the programs in this book. They should give you some ideas about the kinds of programs you might want to write.

Feel free to experiment with any program you copy from a book or magazine. The changes you make might not make any difference. Then again, your changes might make a program better. You never know until you try!

But let's say you've experimented with changing programs written by someone else, and you're now ready to write your very own program—from beginning to end. How do you start?

There are as many different ways to write a program as there are people writing programs. You will have to find the way that is right for you.

Some people like to start by writing the basic idea of a program on a piece of paper, along with some of the important program lines. Then they sit down at the computer and type in the program—using their notes as an outline. Other people like to write out each line of a program before they even sit down in front of a computer. Others never bother to write anything on paper. As soon as they get an idea for a program, they enter it into their computer.

Whatever method you use, remember that programming is a creative process, similar to playing a musical instrument. When you write a computer program you are combining your imagination with a versatile tool (your computer) to create a final product (the program).

Like any other creative work, a computer program reflects the personality of the person who made it. The computer programs you write will be different from the computer programs written by anyone else. There is no right or wrong way to write a computer program. If you write a program and it does what you want it to do, then you have written a successful program.

There are two ideas, however, that you might find useful when writing programs. First, there is nothing more frustrating than sitting in front of a computer and not knowing what to do. It is hard enough to remember where the different keys are located on your computer. If the program doesn't tell you exactly what to do, using the computer can be a very frustrating experience.

You will find your programs easier to use if you are careful to include specific instructions. For example, if you want the person running your program to type their name and press RETURN, include a phrase like "Type your name and press RETURN." Or, if you would like your program to be a little more polite, you might say, "Please type your name and press RETURN."

This sounds like a very simple concept, but you would be surprised how important it is. Many people, when they begin writing programs, don't include instructions. They think that because they are only writing programs for themselves, they will always remember what information to type and/or what key(s) to press.

If you write a program, however, and then don't use it for a month or two, you would be surprised how much you can forget. Including directions will help you and other users get the most out of your computer programs.

The second thing you will want to consider is the way your program looks on the computer screen. Too much information on the screen can be difficult for the operator to read. Too little information and the operator might be bored. You will probably be your own toughest critic. But, after you are satisfied with a program, enlist the aid of your friends and family. Ask other people to run your program and listen to their comments. A different view might help you make a good program even better!

NOTE: Instead of lessons at the end of this chapter, there are four sample programs. If you want, enter one or more of these programs. And, if after you run the program you decide you want to keep it, be sure to use the SAVE procedure to store your work on a digital data pack. If you would like to print your work, consult the SmartBASIC Reference Guide in Part VI (PR#).

Program #1—INTRO

This is more of an information program. It contains three variables—the name of the operator (n\$), the age of the operator (a) and the name of a friend (f\$). Once it has been typed, the operator's name is used twice in the program, in lines 140 and 300. The friend's name is used once, in line 270. And the operator's age is used in line 220 to determine the approximate number of hours the operator has slept. If you wanted to do some research, you could change line 220 and 230. For example, you might find out how many miles the average person walks in a year and multiply the operator's age (a) by that number. After you have entered the program, use the LIST command if you want to proofread your work.

```
10 REM INTRO
20 PRINT "Please type your name and"
30 PRINT "press RETURN."
40 INPUT n$
50 HOME
55 PRINT
60 PRINT "Thank you. Now, please type"
70 PRINT "the name of a friend and"
75 PRINT "press RETURN."
80 INPUT f$
90 HOME
95 PRINT
100 PRINT "Thank you. Now, please type"
105 PRINT "how old you are in years"
110 PRINT "and press RETURN."
120 INPUT a
130 HOME
135 PRINT
140 PRINT "As you know ";n$;" ,"
150 PRINT "I am your ADAM computer and I"
160 PRINT "can do a lot of things for you."
170 PRINT
180 PRINT "For example, I can help you"
190 PRINT "organize your life. I am very"
195 PRINT "good at math and can even"
```

(This lesson continued on the next page.)

```
200 PRINT "tell you things about"
210 PRINT "yourself. (Did you know you"
220 PRINT "have slept approximately"
230 PRINT a*2922;" hours in your life?)"
240 PRINT
250 PRINT "You will find using ADAM"
260 PRINT "simple and fun. Before long"
270 PRINT "you will be showing ";f$
280 PRINT "how easy it is to use ADAM."
290 PRINT
300 PRINT "Good-bye ";n$;" See you later."
```

Illustration 4-6-2

Program #2—MATH DEMO

The program on the next page is similar to one that you may have entered earlier. It is a relatively short program, designed to show off your ADAM's math skills. You could, of course, make this program much more complicated. For example, after line 270, you could add a line that adds the operator's two numbers (a and b) together and then divides or multiplies by a different number.

When typing this program, lines 210, 230, 250, and 270 deserve some special attention. When typing these lines, be sure to include every semicolon (;) that appears in the program. As you learned earlier, semicolons tell your computer to display what follows on the same line. Also, be careful to include the proper number of blank spaces when typing these lines.

(This lesson continued on the next page.)

```
10 REM MATH DEMO
20 PRINT "Please type your name and"
30 PRINT "press RETURN."
40 INPUT n$
50 PRINT:PRINT
60 PRINT "Hello ";n$;". This is a math"
70 PRINT "demonstration program."
80 PRINT
90 PRINT "If you will give me 2 numbers,"
95 PRINT "I will show you how good I"
100 PRINT "am at math."
110 PRINT
120 PRINT "To begin, type a number and"
130 PRINT "press RETURN. (Please do"
135 PRINT "not use commas.)"
140 INPUT a
150 PRINT
160 PRINT "Now type another number"
170 PRINT "and press RETURN."
180 INPUT b
190 HOME
200 PRINT
210 PRINT a;" + ";b;" = ";a+b
220 PRINT
230 PRINT a;" * ";b;" = ";a*b
240 PRINT
250 PRINT a;" - ";b;" = ";a-b
260 PRINT
270 PRINT a;" / ";b;" = ";a/b
280 PRINT:PRINT
290 PRINT "Well, what do you think "
300 PRINT n$;"? I hope I can help"
310 PRINT "you with math problems in"
320 PRINT "the future."
330 PRINT:PRINT
340 PRINT "See you later ";n$;"."
```

Illustration 4-6-3

Program #3—MILEAGE

This program determines the number of "miles per gallon" you get when driving a car. The program asks the operator for three numbers—the mileage reading the last time gas was purchased, the mileage reading this time, and the number of gallons of gas used. Though this is a fairly simple, straightforward program, you could make some interesting changes. For example, at the end of the program you might add a line with your average miles per gallon figure. Then you could add a line that compares the current miles per gallon figure ($b - a/c$) to your average figure.

```
10 REM MILEAGE
20 PRINT "This program determines how"
25 PRINT "many miles per gallon you"
30 PRINT "get while driving."
40 PRINT
50 PRINT "To begin, what was your"
55 PRINT "mileage reading the last time"
60 PRINT "you put gas in your car?"
70 PRINT
80 PRINT "Please type the correct number"
90 PRINT "and press RETURN. (NOTE: Do"
100 PRINT "not use commas.)"
110 PRINT
120 INPUT a
130 PRINT:PRINT
140 PRINT "What is your mileage reading"
145 PRINT "this time?"
150 PRINT
160 PRINT "Please type the correct number"
170 PRINT "and press RETURN."
180 PRINT
190 INPUT b
200 HOME
210 PRINT "Now, how many gallons of gas"
220 PRINT "did you use?"
230 PRINT
240 PRINT "Please type the correct number"
250 PRINT "and press RETURN."
260 PRINT
270 INPUT c
280 HOME
290 PRINT:PRINT
300 PRINT "Based on the numbers you gave"
310 PRINT "me you got--"
320 PRINT
330 PRINT b-a/c;" MILES PER GALLON."
```

Program # 4—GRAPHICS

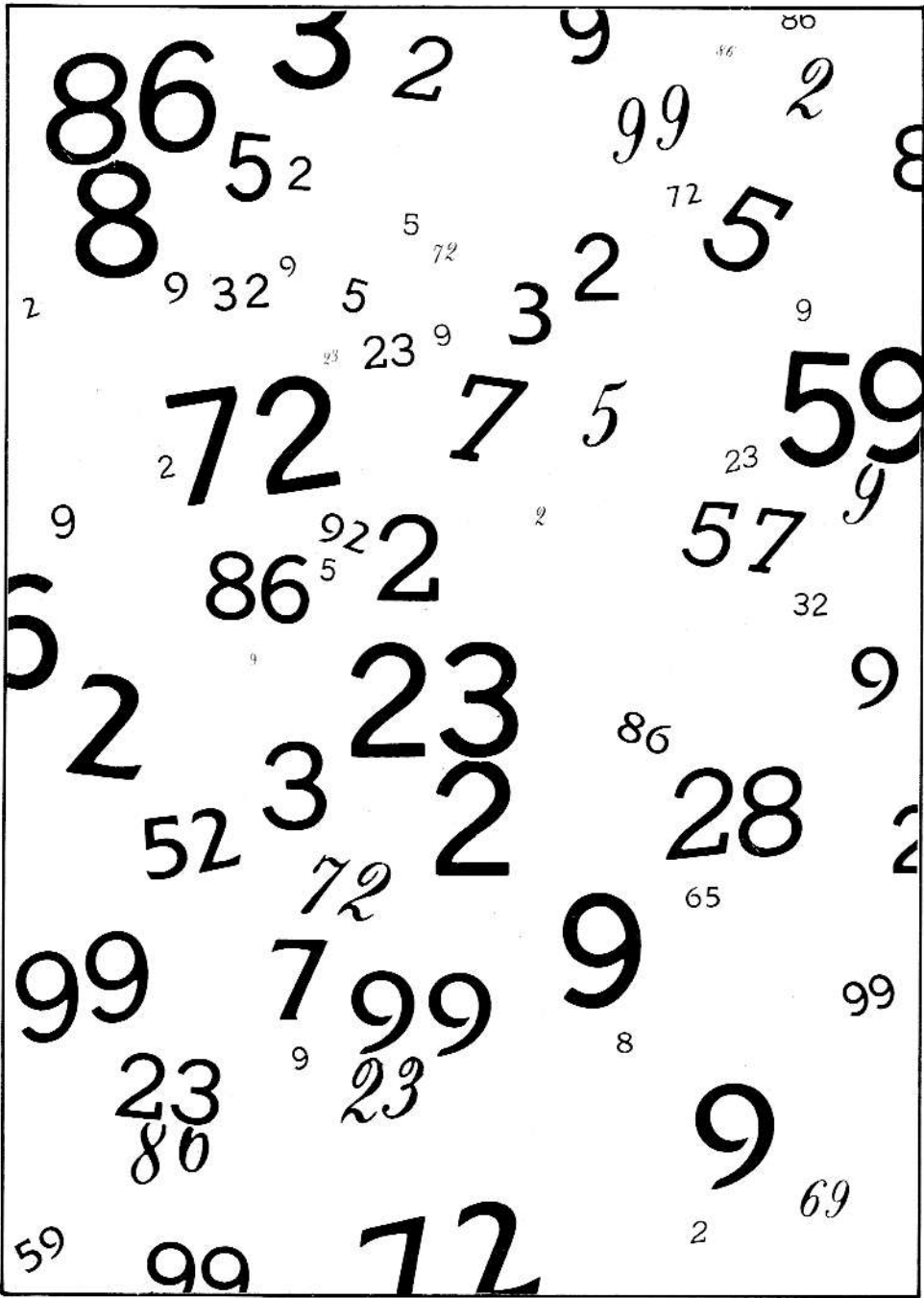
This is a simple graphics program, designed to show off your ADAM's graphics skills. Of course, you could change this program so it draws a variety of shapes in any of the 15 low-resolution graphics colors listed in Chapter 5. NOTE: The INPUT statement in line 160 is used to temporarily halt the display of information on your screen. When the user presses RETURN, the HOME statement (in line 170) causes the screen to clear and new information is displayed.

```
10 REM "GRAPHICS"
20 GR
30 PRINT "This is a graphics demon-"
40 PRINT "stration. To begin, type"
50 PRINT "your name and press RETURN."
60 INPUT n$
70 HOME
80 COLOR=3
90 HLIN 10, 15 AT 20
100 HLIN 10, 15 AT 30
110 VLIN 20, 30 AT 10
120 VLIN 20, 30 AT 15
130 PRINT "I have just drawn a"
140 PRINT "red square. To see a"
150 PRINT "green square, press RETURN."
160 INPUT a$
170 HOME
180 COLOR=6
190 HLIN 15, 20 AT 25
200 HLIN 15, 20 AT 35
210 VLIN 25, 35 AT 15
220 VLIN 25, 35 AT 20
230 PRINT "I hope you have enjoyed"
240 PRINT "this demonstration ";n$;"."
```

Illustration 4-6-5

Part V

Doing More with SmartBASIC



Pick a Number (RND and INT)

If you would like to write game programs for your ADAM computer, you will find that one of the most interesting aspects of SmartBASIC is the ability it gives you to select a random number. To see how this process works, you will need to understand two new keywords—RND (which stands for random) and INT (which stands for integer).

All the SmartBASIC keywords you learned in Part 4 were either statements (in numbered lines) or commands (written by themselves). RND and INT are both classified as functions. Functions are keywords that appear within statements in numbered lines. For example, functions often appear in PRINT statements.

The function RND, followed by a number in parentheses [(X)], allows you to select a random number. The problem, however, is that the number selected by RND will always have several decimal places.

To change the number selected by RND into a whole number (without decimal places) you can use the function INT. INT automatically changes a number containing decimal places to the next lowest whole number. So, for example, if you were to type the following line in a program:

```
20 PRINT INT (2.578)
```

Illustration 5-1-2

When the program was run, this is what you would see:

```
2
```

Illustration 5-1-3

To indicate the range in which you want to select a random number, use your computer's ability to perform multiplication and addition operations. Possible random number combinations are shown in Illustration 5-1-4.

<i>TYPE:</i>	<i>FOR POSSIBLE NUMBERS:</i>
INT(RND(X)*2)	0, 1
INT(RND(X)*2)+1	1, 2
INT(RND(X)*5)	0, 1, 2, 3, 4
INT(RND(X)*5)+1	1, 2, 3, 4, 5
INT(RND(X)*10)	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
INT(RND(X)*10)+1	1, 2, 3, 4, 5, 6, 7, 8, 9, 10

Illustration 5-1-4

Any number can be substituted for X in the Figure above. As you can see, the integer of RND(X) times a number (Y), gives Y potential random numbers between 0 and Y-1. For example, INT(RND(1)*20) would give you the possibility of 20 random numbers between 0 and 19. If you wanted 20 random numbers between 1 and 20, you would write: INT(RND(1)*20)+1. NOTE: Normally, ADAM will always generate the same sequence of random numbers. To create different sequences of random numbers, you can use a "negative seed." This is a process that asks the operator to type a number (in an INPUT statement) and then uses the number typed by the operator to determine a random number. For more information, see lesson #3 at the end of this chapter.

Previously, you have run programs that allowed the operator to assign a value to a variable by using INPUT statements. Normally, when using the random number feature in a program, you will want the computer to assign a value to a variable. You can accomplish this task with a LET statement.

LET statements are simple and easy to understand. LET a=3 means exactly what it appears to mean—give a the value of 3. So, if you were to write a program that said:

```
20 LET a=3
30 PRINT a
```

Illustration 5-1-5

When these two lines are run, you would see:

3

Illustration 5-1-6

LET statements can also be used to assign values to text or "string" variables. Just like when you type your name in a program and the computer assigns your name to a variable (for example, n\$), your computer can assign any word(s) to a string variable using a LET statement. For example, if you were to write a program that said:

```
20 LET a$="HELLO"  
30 PRINT a$
```

Illustration 5-1-7

When these two lines are run, you would see:

```
HELLO
```

Illustration 5-1-8

Selecting a random number to be assigned to a variable is also simple. LET a=INT(RND(1)*10) means: let a random value between 0 and 9 be assigned to the variable a. (NOTE: The use of LET is optional in SmartBASIC. In other words, LET c=25 and c=25 mean the same thing to ADAM. As you become more experienced in writing your own programs, you will probably stop typing the word LET. Most people, however, like to type LET when they are first learning to write programs.)

Lesson # 1—Using RND

Follow the steps below to write and run a short program using RND.

1. If a program is currently in your computer's memory, complete the following steps:
 - a. Type **NEW**
 - b. Press **RETURN**
2. Enter the following program on your computer:

```
10 TEXT:HOME  
12 PRINT:PRINT  
20 LET a=RND(1)  
30 PRINT a
```

Illustration 5-1-9

3. Run the program you entered in step 2. Notice the number that appears on your screen.
4. Write the number currently on your screen on a piece of paper.

Lesson # 2—Using INT

Follow the steps below to make changes to the program you just wrote.

1. Change line 20 of your program so that it matches the following line:

```
20 LET a=INT(RND(1))
```

Illustration 5-1-10

2. Run the program. Notice the number that appears on your screen.
3. Write the number currently on your screen on a piece of paper. Notice how this new number compares to the number you wrote down in the previous lesson.
4. Change line 20 so that it matches the following line:

```
20 LET a=INT(RND(1)*2)
```

Figure 5-1-11

5. Run the program again. Notice the different number that appears on your screen.
6. Write the number currently on your screen on a piece of paper. Notice how this new number compares to the number you wrote down in step 3.
7. Change line 20 so that it matches the following:

```
20 LET a=INT(RND(1)*10)
```

Illustration 5-1-12

8. Run the program again. Notice the different number that appears on your screen.
9. Write the number currently on your screen on a piece of paper. Notice how this new number compares to the numbers you wrote earlier.

Lesson #3—Using a “Negative Seed”

Normally, when you use RND and INT to select one or more random numbers, the same sequence of random numbers will always be generated. To create a different sequence of random numbers, you must use a “negative seed.” The term “negative seed” refers to any number selected by the operator (in an INPUT statement) and then used (with a negative sign) after the function RND. To practice using negative seeds, complete the steps below.

1. Run the program currently in your computer several times. Notice that the same number always appears on your screen.
2. Add the lines illustrated below to your program.

```
14 PRINT "Type a number and press RETURN."  
18 INPUT n
```

Illustration 5-1-13

3. Change line 20 of your program so that it matches the following:

```
20 LET a=INT(RND(-n)*10)
```

Illustration 5-1-14

4. Run the program again. When requested, type any number and press RETURN. Notice the number displayed by the program. Write this number down on a separate sheet of paper.
5. Repeat step 4 two more times. Each time, type a different number. Notice that the random number chosen by the program is different every time you run the program.

Reviewing What You Just Learned

In this chapter you learned how to use RND and INT to select a random value. Three possible random number selections are illustrated below. (NOTE: X indicates any number.)

TYPE:	FOR POSSIBLE NUMBERS:
INT(RND(X)*2)	0, 1,
INT(RND(X)*5)	0, 1, 2, 3, 4,
INT(RND(X)*10)	0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Illustration 5-1-15

To assign a value to a variable, use a LET statement. For example, LET d=42 means "let the variable d have a value of 42." String values can also have assigned variables. For example, LET e\$="YOU ARE A SMART PERSON.", assigns the words that appear between quotation marks to the variable e\$. So, if you were to type the following program:

```
10 LET e$="YOU ARE A SMART PERSON."
20 PRINT e$
```

Illustration 5-1-16

You would see this sentence when the program was run: YOU ARE A SMART PERSON.

```
10 REM SAMPLE
20 PRINT "Type a number and press RETURN."
30 INPUT a
40 IF a=3 THEN PRINT "You typed 3."
50 IF a<>3 THEN PRINT "You did not type 3."
```

Comparison Shopping (IF/THEN)

Is Brand X larger or smaller than Brand Y? Is Brand X the same price or more expensive than Brand Y? Is Brand Z equal to Brand X and Brand Y combined?

These are the kinds of questions you might ask yourself while shopping. You compare values and then take action based on your findings. Your ADAM computer can do the same thing. By using what is known as an IF/THEN statement, your computer can compare two things and then take action based on its findings.

In the program shown on the opposite page, the operator is asked to type a number. If the number 3 is typed, the program displays the message "You typed 3." If the operator types any number other than 3, the program prints the message "You did not type 3."

As you can see, an IF/THEN statement sets up a conditional situation. If the relationship tested for is true, then the program completes the instructions that appear after the word THEN. If you ran the program on the opposite page, and typed 3, this is what would appear on your screen:

```
Type a number and press RETURN.  
3  
You typed 3.
```

Illustration 5-2-2

If you ran the same program, and typed 4, this is what would appear on your screen:

```
Type a number and press RETURN.  
4  
You did not type 3.
```

Illustration 5-2-3

As you can see in illustrations 5-2-2 and 5-2-3, if the condition tested for in an IF/THEN statement is not true, the program goes to the next line.

When writing programs, you will find IF/THEN statements very useful. To create an IF/THEN statement, you must follow a certain procedure.

First, type the word IF. Next, type the first thing you want to compare. For example, you might want to compare the variable a (for a number) or a\$ (for a letter, letters, word, or words). Then, type one of the following signs:

<i>SIGN</i>	<i>MEANING</i>
=	equals
<	is less than
>	is greater than
<=	is less than or equal to
>=	is greater than or equal to
<>	is not equal to

Illustration 5-2-4

After typing the appropriate symbol, type the second variable, number, letter(s), or word(s) you want to use in your comparison. For example, if you wanted to compare the variable a to the number six, you would type 6. If you wanted to compare the variable a\$ to the word yes, you would type YES. (NOTE: If you use an IF/THEN statement to compare letter(s) or word(s), you must use string variables and quotation marks. For example, IF A\$="YES" THEN GOTO 70, checks to see if the word YES has been assigned to the variable A\$. If it has, then the program goes to line 70.)

After you have typed the second variable, number, letter(s), or word(s) you must type the word THEN. Next, you can type any statement that you would normally use in a program. For example, as you saw in Illustration 5-2-1, you can use the statement PRINT in an IF/THEN statement. You might also use GOTO in an IF/THEN statement.

```
200 PRINT "Again? (Type YES or NO and"  
210 PRINT "press RETURN.)"  
220 INPUT r$  
230 IF r$="YES" THEN GOTO 10  
240 PRINT "Ok. See you later."
```

Illustration 5-2-5

In the program lines above, the operator is asked to type either YES or NO. If the operator types YES, the program goes to line 10 (as specified in line 230). If the operator doesn't type YES, the program will skip the GOTO command and go on to line 240. Once at line 240, the program will print "OK. SEE YOU LATER."

Notice that line 240 could have been written:

```
240 IF r$="NO" THEN PRINT "Ok. See you Later."
```

Illustration 5-2-6

When writing your own programs, you will always have a choice. Either you can use an IF/THEN statement to test for one condition (as in Illustration 5-2-5), or you can use IF/THEN statements to test for all possible conditions (as in Illustration 5-2-6). For example, in Illustration 5-2-5, if the operator types MAYBE, I DO NOT KNOW, or HELP!!!, the program will still print "OK. SEE YOU LATER." If line 240 in this program is replaced with the one illustrated in 5-2-6, however, the program will only print "OK. SEE YOU LATER." if the operator types NO. Any response besides YES or NO and the program will not know what to do and will stop running.

The best way to learn about IF/THEN statements is by using them in programs. You will be doing that when you perform the lessons at the end of this chapter. Before you start the lessons, however, you should know about a new word—END.

As you have probably noticed, you don't need to tell your ADAM computer to stop running a program. When it reaches the last line of a program, it always stops. Sometimes, though, you will want a program to stop running before it reaches the last line. For those situations, you can use END.

When your computer encounters the word END, no matter where it is in a program, it comes to a halt. You could write a 50-line program, but if the second line contained the word END, that is as far as the computer would go. END, in other words, stops a program on any line.

Lesson # 1—Using IF/THEN Statements

Follow the steps below to type and run a program using IF/THEN statements.

1. If a program is currently in your computer's memory, complete the following steps:
 - a. Type **NEW**
 - b. Press RETURN.
2. Enter the following program on your computer.

```
10 REM GUESS
11 TEXT: HOME
12 PRINT "To begin this game, please"
14 PRINT "type a number and press"
16 INPUT "RETURN. ";a
18 HOME
20 LET N=INT(RND(-a)*5)+1
30 PRINT "I am thinking of a number"
40 PRINT "between 1-5."
50 PRINT
60 PRINT "You get 1 guess."
70 PRINT
80 PRINT "If you are right, you win."
90 PRINT "If you are wrong, I win."
100 PRINT:PRINT
110 PRINT "Type a number and press RETURN."
120 INPUT g
130 PRINT:PRINT
140 IF n=g THEN PRINT "Lucky guess. You win."
150 IF n<>g THEN PRINT "Ha, ha. I win. My number was
";n;"."
```

Illustration 5-2-7

3. Run the program several times.

(This lesson continued on the next page.)

4. Change line 140 to match the line in illustration 5-2-8.

```
140 PRINT "Lucky guess. You win."
```

Illustration 5-2-8

5. Run the program again and watch what happens.
6. Change line 140 back to:

```
140 IF n=g THEN PRINT "Lucky guess. You win."
```

Illustration 5-2-9

7. Run the program one more time. Notice how it has changed.
8. If you want, use the SAVE procedure to store a copy of this program before you continue.
9. If you do not want to save the program, erase it from the computer's memory by completing the following steps:
 - a. Type **NEW**
 - b. Press RETURN.

Lesson #2—Using END

Follow the steps below to type and run a program using END.

1. Enter the program that appears below.

```
10 REM COOKIES
12 TEXT:HOME
14 PRINT "To begin this game, type a"
16 INPUT "number and press RETURN.>";n
18 HOME
20 LET c=INT(RND(-n)*7)+1
30 PRINT "You are in a kitchen. There"
40 PRINT "are 7 cookie jars on a"
50 PRINT "counter in front of you --"
60 PRINT "but only 1 contains cookies."
70 PRINT
80 PRINT "You get 2 chances to find the"
90 PRINT "cookies."
100 PRINT
110 PRINT "When you are ready to start,"
120 PRINT "type your name and press"
130 INPUT "RETURN.>";n$
140 HOME
150 PRINT "Type the number (between 1-7)"
160 PRINT "of the jar you want to open"
170 PRINT "and press RETURN."
180 INPUT a
190 PRINT:PRINT
200 IF a=c THEN GOTO 340
210 PRINT "Try guessing ";
220 IF a<c THEN PRINT "HIGHER."
230 IF a>c THEN PRINT "LOWER."
240 PRINT
250 PRINT "Type another number and"
260 PRINT "press RETURN."
270 INPUT b
280 PRINT:PRINT
290 IF b=c THEN GOTO 340
300 PRINT "Too bad ";n$;" ". You did"
310 PRINT "not find the cookies. They"
320 PRINT "were in jar number ";c;" ."
330 END
340 PRINT "You found the cookies. Lucky"
350 PRINT "you. Save some for me, OK?"
```

Illustration 5-2-10

(This lesson continued on the next page.)

2. Run the program several times.
3. Delete line 330.
4. Run the program again and watch what happens.
5. Type line 330 again so it matches the line shown below:

330 END

Illustration 5-2-11

6. Run the program again and watch how it has changed.
7. If you want, use the SAVE procedure to store a copy of this program.

Reviewing What You Just Learned

In this chapter you learned about one of your computer's most powerful features: the ability to compare two things (letters, numbers, words, variables, etc.) and then take action based on that comparison. Your computer makes comparisons by using an IF/THEN statement.

The symbols used in IF/THEN statements are shown below:

<i>SIGN</i>	<i>MEANING</i>
=	equals
<	is less than
>	is greater than
<=	is less than or equal to
>=	is greater than or equal to
<>	is not equal to

Illustration 5-2-12


```
10 REM SAMPLE
15 TEXT:HOME
20 GOSUB 120
30 PRINT "This is line 30."
40 GOSUB 120
50 PRINT "This is line 50."
60 GOSUB 120
70 PRINT "This is line 70."
80 GOSUB 120
90 PRINT "This is line 90."
100 GOSUB 120
110 END
120 PRINT "HELLO"
130 RETURN
```

Take A Trip (GOSUB and RETURN)

Have you been noticing how much your computer is like a person? If you talk to it in a language it understands (SmartBASIC), and are polite (use SmartBASIC according to the rules—numbered lines, beginning with a statement, etc.), your computer will do just about anything you ask. It will display information (PRINT); it will select a random number (RND & INT). It will even make comparisons before it takes any action (IF/THEN).

There is another way your ADAM computer is like a person. If it has to do something (for example, display the word "HELLO") several times, it would much rather do the job once than go on doing the same thing over and over and over again.

For example, if you were to run the program on the opposite page, this is what you would see on your screen:

```
HELLO  
This is line 30.  
HELLO  
This is line 50.  
HELLO  
This is line 70.  
HELLO  
This is line 90.  
HELLO
```

Illustration 5-3-2

Lines 120 and 130 in this program are known as a subroutine. If you think of the computer program as "home" for your computer, then a subroutine is like a side trip. Every time your computer encounters the statement GOSUB, it leaves home and takes a little trip. It takes a trip to the line that appears after the statement GOSUB. (NOTE: The command GOSUB XX—where XX indicates a line number—is known as "calling a subroutine.")

While on this little trip, your computer behaves normally. It reads each line, in order, and performs the commands that appear in each program line until it sees the statement RETURN. RETURN is your computer's signal that it is time to end the trip and go home. In other words, when your computer sees

the statement RETURN (at the end of the subroutine), it automatically goes back to the line following the original GOSUB command.

Subroutines can be very handy when writing a program. In fact, you can combine GOSUB with an IF/THEN statement.

```
10 REM SAMPLE
15 TEXT:HOME
20 PRINT "Type a number and press RETURN."
30 INPUT a
40 IF a>5 THEN GOSUB 70
50 PRINT "This is line 50."
60 END
70 PRINT "This is line 70."
80 PRINT "This is line 80."
90 RETURN
```

Illustration 5-3-3

For example, if you were to run the program shown above and type a number larger than 5, this is what would appear on your screen:

```
Type a number and press RETURN.
6
This is line 70.
This is line 80.
This is line 50.
```

Illustration 5-3-4

However, if you were to run the program and type 5 or less, this is what would appear on your screen:

```
Type a number and press RETURN.
4
This is line 50.
```

Illustration 5-3-5

As you can see, to establish a subroutine you need two keywords: GOSUB and RETURN. In the exercises at the end of this chapter you will use subroutines in two different programs. Before you do that, though, there are two new special words you should learn: AND and OR.

In the last chapter, you learned how to use IF/THEN statements to test for certain conditions. Using AND and OR you can test for more than one condition.

For example, the statement `IF a=6 AND b=3 THEN PRINT "HELLO"` means, if the value assigned to the variable `a` is 6, and the value assigned to the variable `b` is 3, then (and only then) display the word HELLO. If `a` is equal to 5 and `b` is equal to 3, the word HELLO will not be displayed. Only when both conditions are true will the command following the word THEN be completed.

On the other hand, the statement `IF a=6 OR b=3 THEN PRINT "HELLO"` means, if `a` equals 6 and `b` equals 20, the word HELLO will be displayed. In addition, if `a` equals 0 and `b` equals 3, the word HELLO will still be displayed. In other words, when OR is present in an IF/THEN statement, at least one of the two conditions must be true. In comparison, when AND is present in an IF/THEN statement, both conditions must be true.

```
10 REM COMPUTERS
15 TEXT:HOME
20 PRINT "***COMPUTER PROFILE***"
30 PRINT:PRINT
40 PRINT "1) NO COMPUTER EXPERIENCE"
50 PRINT
60 PRINT "2) SOME COMPUTER EXPERIENCE"
70 PRINT
80 PRINT "3) COMPUTER EXPERT"
90 PRINT:PRINT
100 PRINT "Type the number (1-3) that"
110 PRINT "best describes your experience"
120 PRINT "and press RETURN."
130 INPUT a
140 HOME
150 IF a=1 THEN GOSUB 300
160 IF a=2 THEN GOSUB 340
170 IF a=3 THEN GOSUB 380
180 PRINT
190 PRINT "Would you like to run this"
200 PRINT "program again?"
210 PRINT
220 PRINT "Type Y (for yes) or N (for no)"
230 PRINT "and press RETURN."
240 INPUT r$
250 IF r$="N" OR r$="n" THEN GOTO 270
260 GOTO 15
270 PRINT "Good-bye. See you later."
280 END
300 PRINT "Your ADAM Family Computer"
310 PRINT "System is easy to use. You"
320 PRINT "will soon be a computer"
325 PRINT "expert."
330 RETURN
340 PRINT "You will appreciate ADAM's"
350 PRINT "letter-quality printer and"
360 PRINT "exciting graphics capability."
370 RETURN
380 PRINT "ADAM uses a specific version"
390 PRINT "of BASIC. It is known as"
395 PRINT "SmartBASIC."
400 RETURN
```

Illustration 5-3-6

Lesson # 1—Using Subroutines

Follow the steps below to type and run the program shown on the previous page. This program uses subroutines.

1. If a program is currently in your computer's memory, complete the following steps:
 - a. Type **NEW**
 - b. Press RETURN.
2. Enter the program in Illustration 5-3-6 on your computer.
3. Run the program several times. Each time, select a different number so a different subroutine is called. (For example, the first time you run the program, type 1 so subroutine 300 is called. The next time you run the program, type 2 to call subroutine 340. Finally, type 3 to call subroutine 380.
4. If you want, use the SAVE procedure to store a copy of this program before you continue.

(This lesson continued on the next page.)

```
10 REM "FLIP"
12 TEXT:HOME
14 PRINT "To begin this game, please"
16 PRINT "type a number and press"
18 INPUT "RETURN. ";n
19 HOME
20 LET c=INT(RND(-n)*2)
30 PRINT "I am going to flip a coin."
40 PRINT "Before I do, do you want"
50 PRINT "heads or tails?"
60 PRINT
70 PRINT "Type H (for HEADS) or"
80 PRINT "T (for TAILS) and"
90 PRINT "press RETURN."
100 INPUT f$
110 IF f$="H" OR "h" THEN PRINT "Ok. I have tails."
120 IF f$="T" OR "t" THEN PRINT "Ok. I have heads."
130 PRINT
140 PRINT "When you want me to flip the"
150 PRINT "coin, type your name and"
160 PRINT "press RETURN."
170 INPUT n$
180 HOME
190 PRINT:PRINT
200 IF c=0 THEN PRINT "HEADS"
210 IF c=1 THEN PRINT "TAILS"
220 PRINT:PRINT
230 IF c=0 AND f$="H" OR c=1 AND f$="T" THEN GOTO 270
240 IF c=0 AND f$="h" OR c=1 AND f$="t" THEN GOTO 270
250 IF c=0 AND f$="T" OR c=1 AND f$="H" THEN GOTO 290
260 IF c=0 AND f$="t" OR c=1 AND f$="h" THEN GOTO 290
265 END
270 PRINT "You win, ";n$;". "
280 END
290 PRINT "Oh good. I win."
```

Illustration 5-3-7

Lesson # 2—Using AND and OR

Follow the steps below to type and run a program using AND and OR.

1. To erase the program currently in your computer's memory, complete the following steps:
 - a. Type **NEW**
 - b. Press RETURN.
2. Enter the program in illustration 5-3-7 on your computer.
3. Run the program several times.
4. Delete lines 230 and 240. Run the program two more times. Watch what happens.
5. Retype lines 230 and 240 so that they match the lines illustrated below:

```
230 IF c=0 AND f$="H" OR c=1 AND f$="T" THEN GOTO 270
240 IF c=0 AND f$="h" OR c=1 AND f$="t" THEN GOTO 270
```

Illustration 5-3-8

6. Run the program again.
7. If you want, use the SAVE procedure to store a copy of this program before you continue.

Reviewing What You Just Learned

If you think of the computer program as “home” for your computer, then a subroutine is like a side trip. Every time your computer encounters the statement GOSUB, it leaves home and takes a little trip. It takes a trip to the line that appears after the statement GOSUB. (NOTE: The command GOSUB XX—where XX indicates a line number—is known as “calling a subroutine.”)

RETURN is your computer’s signal that it is time to end the trip and go home. In other words, when your computer sees a RETURN statement (at the end of the subroutine), it automatically goes back to the line following the original GOSUB command.

Using AND and OR (in IF/THEN statements), you can test for more than one condition. For example, the statement IF a=6 AND b=3 THEN PRINT “HELLO” means, if the value assigned to the variable a is 6, and the value assigned to the variable b is 3, then (and only then) display the word HELLO.

Lesson # 2—Using AND and OR

Follow the steps below to type and run a program using AND and OR.

1. To erase the program currently in your computer's memory, complete the following steps:
 - a. Type **NEW**
 - b. Press RETURN.
2. Enter the program in illustration 5-3-7 on your computer.
3. Run the program several times.
4. Delete lines 230 and 240. Run the program two more times. Watch what happens.
5. Retype lines 230 and 240 so that they match the lines illustrated below:

```
230 IF c=0 AND f$="H" OR c=1 AND f$="T" THEN GOTO 270
240 IF c=0 AND f$="h" OR c=1 AND f$="t" THEN GOTO 270
```

Illustration 5-3-8

6. Run the program again.
7. If you want, use the SAVE procedure to store a copy of this program before you continue.


```
10 PRINT 1
20 PRINT 2
30 PRINT 3
40 PRINT 4
50 PRINT 5
60 PRINT 6
70 PRINT 7
80 PRINT 8
90 PRINT 9
```

Around & Around (FOR/NEXT Loops)

You know how frustrating it is to do the same thing over and over and over again. Your computer feels the same way. If you want to display the numbers 1 through 9 in a program, for example, you can type the program as shown on the opposite page.

But there is a much easier way:

```
10 FOR a=1 TO 9
20 PRINT a
30 NEXT a
```

Illustration 5-4-2

These two programs would have the same result when run on your computer. Both programs would display the following information:

```
1
2
3
4
5
6
7
8
9
```

Illustration 5-4-3

Obviously, the program in Illustration 5-4-2 shows a much easier way to get the same job done. That is because of lines 10-30, which are known as a FOR/NEXT loop.

FOR/NEXT loops are a way for your computer and you to save time and energy. A FOR/NEXT loop always begins with a line that sets the lower and upper limits for a variable. For example, in Illustration 5-4-2, line 10 means that the variable *a* is going to have 9 different values with 1 being the lowest value and 9 being the highest value. When the FOR/NEXT loop begins, the lowest value is assigned to the variable.

That is why, when the program is run, it starts by displaying the number 1. When the statement NEXT a is encountered by the computer, it immediately goes back to the FOR a line (in this case, line 10) and assigns a new value to the variable (in this case, 2). Then, when the command PRINT a appears in line 20, the program displays the number 2. In line 30 it goes back to line 10 and changes the value of a to 3, prints 3 in line 20, goes back to line 10, and around, and around, and around, and around.

Your computer stops running a FOR/NEXT loop for only one reason. When the upper limit of the variable is reached, the computer will skip the NEXT statement and go on to the rest of the program. For example, if you were to write the following program:

```
10 FOR b=2 TO 6
20 PRINT b
30 NEXT b
40 PRINT "HELLO"
```

Illustration 5-4-4

This is what you would see when the program was run:

```
2
3
4
5
6
HELLO
```

Illustration 5-4-5

As you can see, you need three words to create a FOR/NEXT statement: FOR, TO, and NEXT. FOR and TO appear in the first line of the FOR/NEXT loop. NEXT appears in the last line of the loop.

Sometimes, when using a FOR/NEXT loop in a program, you won't want the variable to have every possible value between the lower and upper limits. For example, you might want only the odd numbers between 1 and 10.

```
10 FOR c=1 TO 10 STEP 2
20 PRINT c
30 NEXT c
```

Illustration 5-4-6

If you were to enter and run the program in Illustration 5-4-6, this is what you would see displayed on your screen:

1
3
5
7
9

Illustration 5-4-7

As you can see, the word STEP allows you to indicate how many numbers you want to skip in a FOR/NEXT loop. For example, if you wanted the variable d to have all the possible even values between 1 and 10, you could write:

```
10 FOR d=2 TO 10 STEP 2
20 PRINT d
30 NEXT d
```

Illustration 5-4-8

Then, when you ran the three lines shown above, this is what would appear on your screen:

2
4
6
8
10

Illustration 5-4-9

Lesson # 1—Using a FOR/NEXT Loop

Follow the steps below to type and run a short program using a FOR/NEXT loop.

1. If a program is currently in your computer's memory, complete the following steps:
 - a. Type **NEW**
 - b. Press RETURN.
2. Type the program in Illustration 5-4-10 on your computer.

```
10 TEXT:HOME
20 FOR a=1 TO 20
30 PRINT a
40 NEXT a
50 PRINT "This is Line 50."
```

Illustration 5-4-10

3. Run the program and watch what appears on your screen.
4. Change line 20 so it matches the line in Illustration 5-4-11.

```
20 FOR a=3 TO 12
```

Illustration 5-4-11

5. Run the program again. Notice how the numbers displayed on your screen differ from the numbers displayed when you ran the program earlier.

Lesson # 2—Using a FOR/NEXT loop with STEP

Follow the steps below to change the program currently stored on your computer by adding the word STEP.

1. Change line 20 so it matches the line shown below.

```
20 FOR a=1 TO 20 STEP 2
```

Illustration 5-4-12

2. Run the program and watch what appears on your screen.
3. Change line 20 so it matches the following line.

```
20 FOR a=1 TO 20 STEP 3
```

Illustration 5-4-13

4. Run the program again. Notice how the numbers displayed on your screen differ from the numbers displayed when you ran the program in step 2.
5. Change line 20 so it matches the following line.

```
20 FOR a=2 TO 20 STEP 3
```

Illustration 5-4-14

6. Run the program again. Notice how the numbers displayed on your screen differ from the numbers displayed when you ran the program earlier.

Reviewing What You Just Learned

In this chapter you learned about a way you can save time and energy by using FOR/NEXT loops. A FOR/NEXT loop always begins with a line that sets the lower and upper limits for a variable.

```
10 FOR d=1 TO 4
20 PRINT d
30 NEXT d
```

Illustration 5-4-15

For example, in the program above, line 10 means that the variable *d* is going to have four different values, with one being the lowest value and four being the highest value. When the FOR/NEXT loop begins, the lowest value is assigned to the variable.

Your computer stops running a FOR/NEXT loop for only one reason. When the upper limit of the variable is reached, the computer will skip the NEXT statement and go on to the rest of the program.

The word STEP allows you to indicate how many numbers you want to skip in a FOR/NEXT loop. For example, if you wanted the variable *e* to have all the possible even values between 1 and 20, you could write:

```
10 FOR e=2 TO 20 STEP 2
20 PRINT e
30 NEXT e
```

Illustration 5-4-16

COLOR	NUMBER
Black 1	0
Green	1
Violet	2
White 1	3
Black 2	4
Orange	5
Blue	6
White 2	7
Brown	8
Dark Blue	9
Grey	10
Pink	11
Dark Green	12
Yellow	13
Aqua	14
Magenta	15

More Rainbows (HGR, HGR2, HCOLOR=, and HPLOT)

Earlier (in Part IV, Chapter 5) you learned to use low resolution graphics on your ADAM computer. You can also use high resolution graphics in SmartBASIC.

To tell ADAM you want to change from the text mode to the high resolution graphics mode you must use either HGR or HGR2. HGR divides your screen into 256 columns (numbered 0-255, starting at the left edge of your screen) and 160 rows (numbered 0-159, starting at the top of your screen.) When you use an HGR statement in a program, four lines are reserved for text at the bottom of the high resolution graphics screen.

When you use HGR2 in a program, however, the entire screen is devoted to graphics display. After your computer sees HGR2 it divides your screen into 256 columns (numbered 0-255, starting at the left edge of your screen) and 192 rows (numbered 0-191, starting at the top of your screen).

When writing a program using high resolution graphics, you will first use HGR or HGR2 in a statement. Then, using an HCOLOR= statement, you can choose one of the 16 colors listed on the opposite page. The color you select will be the color with which you will be "drawing." For example, in the lines below, line 10 indicates that the program will be in high resolution graphics. Line 20 indicates that the program will begin by drawing in orange.

```
10 HGR
20 HCOLOR=5
```

Illustration 5-5-2

HGR and HGR2 can both be used as either a statement (within a program line) or as a command (by themselves). Whenever your computer sees HGR or HGR2, it knows you want it to go into the high resolution graphics mode. (NOTE: If you want to leave either low resolution or high resolution graphics mode, use the keyword TEXT. Like GR, HGR, and HGR2, TEXT can be used both as a statement and a command. Whenever your computer sees HGR or HGR2, it knows you want it to go into the text or narration mode.)

Once you have typed an HGR or HGR2 statement and have selected a color using an HCOLOR= statement, you are ready to begin "drawing" using HPLOT.

The keyword HPLOT allows you to "color" the intersection of columns and rows on the high resolution graphics screen. Each time you use HPLOT a small block appears on your screen.

HPLOT should always be followed by a column number (0-255), a comma (,), and a row number (0-159 or 191). For example, if you ran the program shown below, a small violet square (like a tiny dot) would appear on your screen at the intersection of column 50 and row 75.

```
10 HGR
20 HCOLOR=2
30 HPLOT 50,75
```

Illustration 5-5-3

By using TO, you can draw lines using HPLOT. For example, in the program below, line 30 tells your computer to draw a vertical, violet line between the intersection of column 50 and row 75 and the intersection of column 50 and row 100. Line 50 tells your computer to draw a horizontal, pink line between the intersection of column 51 and row 100 and the intersection of column 76 and row 100.

```
10 HGR
20 HCOLOR=2
30 HPLOT 50,75 TO 50,100
40 HCOLOR=11
50 HPLOT 51,100 TO 76,100
```

Illustration 5-5-4

Lesson #1—Using HGR

Complete the following steps to write and run a short program using HGR.

1. Enter the program that appears in Illustration 5-5-5.

```
10 HGR
20 HCOLOR=4
30 HPLOT 10, 25 TO 100, 25
```

Illustration 5-5-5

2. Run the program that you typed in step 1. Notice the color of the line that appears on your screen. Also, notice where the line is drawn (between columns 10 and 100 in row 25).
3. To see your program listing again, complete the following steps:
 - a. Type **TEXT**
 - b. Press RETURN.
 - c. Type **LIST**
 - d. Press RETURN.
4. Add the following lines to your program:

```
40 PRINT "This is a demonstration of"
50 PRINT "HPLOT."
```

Illustration 5-5-6

5. Run the program again.
6. Repeat step 3 to see your program listing.
7. To continue, perform the next lesson.

Lesson # 2—Using HGR2

Complete the following steps to see the difference between HGR and HGR2.

1. Type the new line shown below.

```
10 HGR2
```

Illustration 5-5-7

2. To see your program lines in correct order, complete the following steps:
 - a. Type **LIST**
 - b. Press RETURN.
3. Run this new version of your program. Notice that the text no longer appears on your screen.
4. To see your program listing again, complete the following steps:
 - a. Type **TEXT** (NOTE: The word TEXT will not be visible on your screen.)
 - b. Press RETURN.
 - c. Type **LIST**
 - d. Press RETURN.

Lesson #3—Doing more with HPLOT

Complete the following steps to write and run a short program using PLOT.

1. Type the three new lines shown below.

```
10 HGR
60 HCOLOR=14
70 HPLOT 100, 26 TO 100, 101
```

Illustration 5-5-8

2. To see your program lines in correct order, complete the following steps:
 - a. Type **LIST**
 - b. Press RETURN.
3. Run this new version of your program. Notice the color of the two lines that appear on your screen. Also, notice where the lines are drawn.
4. To see your program listing again, complete the following steps:
 - a. Type **TEXT**
 - b. Press RETURN.
 - c. Type **LIST**
 - d. Press RETURN.
5. If you want, add lines to this program to "plot in" other areas of your screen. You can also add HCOLOR= statements if you want to plot in different colors. For example, the next line in your program could be HCOLOR=1. Then any HPLOT statement that followed this line would appear in green when your program was run.

Reviewing What You Just Learned

To change from the text mode to the high resolution graphics mode, use HGR or HGR2. HGR divides your screen into 256 columns and 160 rows. When you use an HGR statement in a program, four lines are reserved for text at the bottom of the high resolution graphics screen.

When you use HGR2 in a program, however, the entire screen is devoted to graphics display. After your computer sees HGR2, it divides your screen into 256 columns and 192 rows.

When writing a program using high resolution graphics, you will first use HGR or HGR2 in a statement. Then, using an HCOLOR= statement, you can choose one of 16 colors.

Once you have typed an HGR or HGR2 statement and selected a color using an HCOLOR= statement, you are ready to begin "drawing" using HPLOT. The keyword HPLOT allows you to "color" the intersection of columns and rows on the high resolution graphics screen. Each time you use HPLOT a small block appears on your screen.

When your screen is in the low or high resolution graphics mode, you must use the keyword TEXT (as a statement or command) to return to the text or narration mode. In other words, if you have just finished running a graphics program and want to see a line-by-line listing of your program, you must first type TEXT and press RETURN. Then you can use the LIST command to see individual lines in the program.


```

1  TEXT:HOME
2  INPUT "Type a number and press RETURN.;"
3  HOME
4  FOR P=1 TO 19
5  LET L=INT(RND(-n)*19)+1
6  GOSUB L*10
7  NEXT P
8  END
9  END

10 PRINT " ADJ";
15 RETURN
20 PRINT " NOUN"
25 RETURN
30 PRINT " ADJ";
35 RETURN
40 PRINT " NOUN"
45 RETURN
50 PRINT " ADJ";
55 RETURN
60 PRINT " NOUN"
65 RETURN
70 PRINT " ADJ";
75 RETURN
80 PRINT " NOUN"
85 RETURN
90 PRINT " ADJ"
95 RETURN
100 PRINT " NOUN"
105 RETURN
110 PRINT " PHRASE"
115 RETURN
120 PRINT " PHRASE"
125 RETURN
130 PRINT " PHRASE"
135 RETURN
140 PRINT " PHRASE"
145 RETURN
150 PRINT " PHRASE"
155 RETURN
160 PRINT " PHRASE"
165 RETURN
170 PRINT " PHRASE"
175 RETURN
180 PRINT " PHRASE"
185 RETURN
190 PRINT " PHRASE"
195 RETURN

590 RETURN
700 PRINT "MEAT-LESS"
710 GOSUB 900
720 PRINT "Omelettes (p."
730 PRINT "Vegetarian Piz"
780 PRINT
790 RETURN
900 PRINT
910 PRINT "May I suggest"
920 PRINT

10 REM COOKBOOK
15 TEXT:HOME
20 PRINT "Hello. I am your new"
30 PRINT "cooking assistant. Tell"
40 PRINT "me the main ingredient of"
50 PRINT "your meal and I will give"
60 PRINT "you some recipe suggestions."
70 PRINT:PRINT
80 PRINT "*****MAIN INGREDIENT*****"
90 PRINT
100 PRINT " 1) GROUND BEEF"
110 PRINT " 2) CHICKEN"
120 PRINT " 3) LAMB CHOPS"
130 PRINT " 4) MEAT-LESS"

CONV
ME
This program con
'1) FEET TO METE
'2) METERS TO FI
PRINT
"Type the number
"process you wan
"and press RETUR
T a
a=1 THEN GOSUB 250
a=2 THEN GOSUB 330
INT:PRINT
INT "Would you like
INT "program again?
INT "Type Y (for YE
INT "and press RETU
INPUT d$
HOME
(This program continu
"fer."

Type the number of feet
want to convert to mete
and press RETURN."
PRINT
" FEET = ";b/3.28;" MET

Type the number of meter(s)
you want to convert to feet"
and press RETURN."
INT
" METERS = ";c*3.28;" FEET"

her"
JRN."

0 P$
80 PRINT "number and
40 PRINT "number and
50 INPUT b
260 IF b=t THEN GOTO 410
270 IF b=p THEN GOTO 440
280 PRINT
290 PRINT "Try one more time. Type a"
300 PRINT "number and press RETURN."
310 INPUT c
320 IF c=t THEN GOTO 410
330 IF c=p THEN GOTO 440
PRINT:PRINT
bad "in";, you did"
treasure."

next page.)

(This program continued on next page.)

PRINT:PRINT
PRINT "Please type a number"
PRINT "(between 1 - 4) and"
PRINT "press RETURN."
PRINT
INPUT c
HOME
c=1 THEN
c=2 THEN
c=3 THEN
c=4 THEN

10 REM TREASURE
12 PRINT "To begin this game, type a"
14 INPUT "number and press RETURN.;"
15 TEXT:HOME
16 LET t=INT(RND(-n)*10)+1
17 LET p=INT(RND(-n)*10)+1
20 PRINT "You are on a tropical island."
30 PRINT "In front of you are 10 large"
40 PRINT "rocks. Under one rock is a"
50 PRINT "chest full of treasure."
60 PRINT
70 PRINT
80 PRINT "You get 3 chances to find th"
90 PRINT "treasure. But watch out, a"
100 PRINT "pirate might get it first."
110 PRINT
120 PRINT
130 PRINT "To begin, type your name"
140 PRINT "and press RETURN."
150 INPUT n$
160 HOME
170 PRINT "Type the number (1-10) of"
175 PRINT "rock you want to move and"
180 PRINT "press RETURN."
190 INPUT a
200 IF a=t THEN GOTO 410
210 IF a=p THEN GOTO 440

(This program continued on next
"p. 32)"

TURN
PRINT "LAMB CHOPS"

```

6

More Programs to Try

You have learned a lot about SmartBASIC. If you have read Parts 4 and 5, you are ready to begin writing your own programs!

You can use subroutines, IF/THEN statements, and random numbers in your own programs. You can use low resolution or high resolution graphics. You can write game programs and home management programs. The only limit is your imagination!

If you are hesitant to try writing your own programs, enter one of the programs from this book. Then change it. Add or delete lines. See if you can make it a better program. Then, using what you have learned, write a similar program. If you need assistance, consult the SmartBASIC Reference Guide in Part 6.

Remember, there are no right or wrong answers in programming. If a computer program does what you want, then you have written a successful program.

NOTE: Instead of lessons at the end of this chapter, there are four sample programs. If you want, enter one or more of these programs. And, if after you run the program you decide you want to keep it, be sure to use the SAVE procedure to store your work on a digital data pack. If you would like to print your work, consult the SmartBASIC Reference Guide in Part 6 (PR#).

Lesson # 1—POEM

Believe it or not, this program is designed to write poetry. All you have to do when you type the program into your computer is replace the notation ADJ with an adjective (for example, TALL, GRASSY, COOL, etc.), the notation NOUN with a noun (for example, FOREST, FLOWER, TREE, etc.), and the notation PHRASE with a phrase (for example, DRIFTING QUIETLY, BENDING IN THE WIND, etc.).

You will, of course, want to have a theme for your poem. So, you will want to select adjectives, nouns, and phrases on the same subject. For example, in the paragraph above, all the sample illustrations might come from a poem about a forest. Choose your own words on any subject that interests you and you will be surprised at the results. Because of the random value established for I (in line 6), every time you run this program you will get a different poem. Sometimes the poetry is horrible. But other times it is remarkably good. Nothing to rival Shakespeare, perhaps—but still quite effective poetry with unusual images and phrases.

NOTE: The spacing on this program is very important. Be sure you press the SPACEBAR after you type a quotation mark and before you type an adjective, noun, or phrase. This space insures that the words in your poem will have blank spaces between them. Also, be sure to type semicolons (;) where indicated. As you know, each semicolon tells your computer to display the next PRINT statement on the same line.

(This lesson continued on the next page.)

```
1  REM POEM
2  TEXT:HOME
3  INPUT "Type a number and press RETURN.";n
4  HOME
5  FOR P=1 TO 19
6  LET L=INT(RND(-n)*19)+1
7  GOSUB L*10
8  NEXT P
9  END

10 PRINT " ADJ";
15 RETURN
20 PRINT " NOUN"
25 RETURN
30 PRINT " ADJ";
35 RETURN
40 PRINT " NOUN"
45 RETURN
50 PRINT " ADJ";
55 RETURN
60 PRINT " NOUN"
65 RETURN
70 PRINT " ADJ";
75 RETURN
80 PRINT " NOUN"
85 RETURN
90 PRINT " ADJ";
95 RETURN
100 PRINT " NOUN"
105 RETURN
110 PRINT " PHRASE"
115 RETURN
120 PRINT " PHRASE"
125 RETURN
130 PRINT " PHRASE"
135 RETURN
140 PRINT " PHRASE"
145 RETURN
150 PRINT " PHRASE"
155 RETURN
160 PRINT " PHRASE"
165 RETURN
170 PRINT " PHRASE"
175 RETURN
180 PRINT "...";
185 RETURN
190 PRINT
195 RETURN
```

Lesson # 2—TREASURE

This is a fairly advanced random number game. In lines 20 and 30, the program selects two random values between 1 and 10. These values are then assigned to the variables t and p. Later in the program, the operator gets three chances to guess the value assigned to t. If they guess the value assigned to t, the program skips to line 410 and says that they have found the treasure. If, while trying to guess the value assigned to t, the operator happens to guess the value assigned to p, the program skips to line 440 and says that the pirate has come back and taken his treasure. If, after three tries, the operator does not guess the value assigned to t, the program says "Too bad" and reveals the value of t (lines 350-400).

```
10 REM TREASURE
12 PRINT "To begin this game, type a"
14 INPUT "number and press RETURN. ";n
15 TEXT:HOME
20 LET t=INT(RND(-n)*10)+1
30 LET p=INT(RND(-n)*10)+1
40 PRINT "You are on a tropical island."
50 PRINT "In front of you are 10 large"
60 PRINT "rocks. Under one rock is a"
70 PRINT "chest full of treasure."
80 PRINT
90 PRINT "You get 3 chances to find the"
100 PRINT "treasure. But watch out, a"
110 PRINT "pirate might get it first."
120 PRINT
130 PRINT "To begin, type your name"
140 PRINT "and press RETURN."
150 INPUT n$
160 HOME
170 PRINT "Type the number (1-10) of the"
175 PRINT "rock you want to move and"
180 PRINT "press RETURN."
190 INPUT a
200 IF a=t THEN GOTO 410
210 IF a=p THEN GOTO 440
```

(This lesson continued on the next page.)

```
220 PRINT
230 PRINT "Try again. Type another"
240 PRINT "number and press RETURN."
250 INPUT b
260 IF b=t THEN GOTO 410
270 IF b=p THEN GOTO 440
280 PRINT
290 PRINT "Try one more time. Type a"
300 PRINT "number and press RETURN."
310 INPUT c
320 IF c=t THEN GOTO 410
330 IF c=p THEN GOTO 440
340 PRINT:PRINT
350 PRINT "Too bad ";n$;", you did"
360 PRINT "not find the treasure."
370 PRINT
380 PRINT
390 PRINT "It was under rock number ";t;". "
400 END
410 PRINT "Congratulations ";n$;". "
420 PRINT "How does it feel to be rich?"
430 END
440 PRINT "Look out. Here comes the"
445 PRINT "pirate. He is taking the"
450 PRINT "treasure. It was hidden"
460 PRINT "under rock number ";t;". "
```

Illustration 5-6-3

Lesson # 3—F/M CONV

This program helps you change feet to meters and meters to feet. It begins by asking the operator what process they want to use, 1 or 2. Then, depending on the operator's answer, the program jumps to one of two sub-routines. After converting measurements, the program gives the operator the option of repeating the conversion process.

Though you may not need this specific program, you can probably adapt it for a useful purpose. For example, if you travel to a foreign country, you can use this same basic program structure to convert between American currency and any other monetary form. If you follow a regular exercise program, you might change this program so it converts exercise time to calories burned and vice versa.

```
10 REM F/M CONV
15 TEXT:HOME
20 PRINT "This program converts"
30 PRINT
40 PRINT "1) FEET TO METERS"
50 PRINT "2) METERS TO FEET"
60 PRINT:PRINT
70 PRINT "Type the number (1 OR 2) of"
80 PRINT "process you want to use"
90 PRINT "and press RETURN."
100 INPUT a
110 HOME
120 IF a=1 THEN GOSUB 250
130 IF a=2 THEN GOSUB 330
140 PRINT:PRINT
150 PRINT "Would you like to run this"
160 PRINT "program again?"
170 PRINT
180 PRINT "Type Y (for YES) or N (for NO)"
190 PRINT "and press RETURN."
200 INPUT d$
210 HOME
```

(This lesson continued on the next page.)

```
220 IF d$="Y" OR d$="y" THEN GOTO 20
230 PRINT "Ok, see you later."
240 END
250 PRINT
260 PRINT "Type the number of feet you"
270 PRINT "want to convert to meter(s)"
280 PRINT "and press RETURN."
290 INPUT b
300 PRINT:PRINT
310 PRINT b;" FEET = ";b/3.28;" METER(S)"
320 RETURN
330 PRINT
340 PRINT "Type the number of meter(s)"
345 PRINT "you want to convert to feet"
350 PRINT "and press RETURN."
360 INPUT c
370 PRINT:PRINT
380 PRINT c;" METERS = ";c*3.28;" FEET"
390 RETURN
```

Illustration 5-6-4

Lesson #4—COOKBOOK

Are you tired of eating the same thing every night for dinner? Meet your new cooking assistant...this simple computer program.

If you take a moment and look through this program, you will see that it starts by offering the operator a choice. They can select 1, 2, 3, or 4, to indicate the main ingredient of the meal they are planning. Then, depending on the number they select, the information in a subroutine is displayed on the screen. Each subroutine lists a possible recipe and the appropriate page number in a cookbook. Then, the operator has a choice (in line 270). If they type Y, the program goes back to line 70. If they type N, to indicate that they don't want to use the program again, the program says "Ok. See you later." and ends (lines 350 and 360).

You, of course, could easily change this program to meet your individual needs. All you have to do is list the choices you want to offer after line 100. Then, like lines 200 through 240, make sure the appropriate subroutine is called. Finally, create subroutines for each choice, listing recipes and page numbers.

```
10 REM COOKBOOK
15 TEXT:HOME
20 PRINT "Hello. I am your new"
30 PRINT "cooking assistant. Tell"
40 PRINT "me the main ingredient of"
50 PRINT "your meal and I will give"
60 PRINT "you some recipe suggestions."
70 PRINT:PRINT
80 PRINT "*****MAIN INGREDIENT*****"
90 PRINT
100 PRINT " 1) GROUND BEEF"
110 PRINT " 2) CHICKEN"
120 PRINT " 3) LAMB CHOPS"
130 PRINT " 4) MEAT-LESS"
140 PRINT:PRINT
150 PRINT "Please type a number"
155 PRINT "(between 1 - 4) and"
160 PRINT "press RETURN."
170 PRINT
180 INPUT c
190 HOME
200 IF c=1 THEN GOSUB 400
210 IF c=2 THEN GOSUB 500
230 IF c=3 THEN GOSUB 600
240 IF c=4 THEN GOSUB 700
250 IF c>4 THEN GOTO 150
```

(This lesson continued on the next page.)

```
260 PRINT
270 PRINT "Would you like to make"
280 PRINT "another selection?"
290 PRINT
300 PRINT "Type Y(es) or N(o) and"
310 PRINT "press RETURN."
320 INPUT r$
330 HOME
340 IF r$="Y" OR r$="y" THEN GOTO 70
350 PRINT "Ok. See you later."
360 END
400 PRINT "GROUND BEEF"
410 GOSUB 900
420 PRINT "Meat Loaf (p. 26)"
430 PRINT "Tacos (p. 73)"
440 PRINT "Spaghetti (p. 68)"
480 PRINT
490 RETURN
500 PRINT "CHICKEN"
510 GOSUB 900
520 PRINT "Chicken Kiev (p. 39)"
530 PRINT "Chicken Chow Mein (p. 46)"
540 PRINT "Fried Chicken (p. 32)"
580 PRINT
590 RETURN
600 PRINT "LAMB CHOPS"
610 GOSUB 900
620 PRINT "Stuffed Chops (p. 56)"
680 PRINT
690 RETURN
700 PRINT "MEAT-LESS"
710 GOSUB 900
720 PRINT "Omelettes (p. 83)"
730 PRINT "Vegetarian Pizza (p. 112)"
780 PRINT
790 RETURN
900 PRINT
910 PRINT "May I suggest:"
920 PRINT
930 RETURN
```

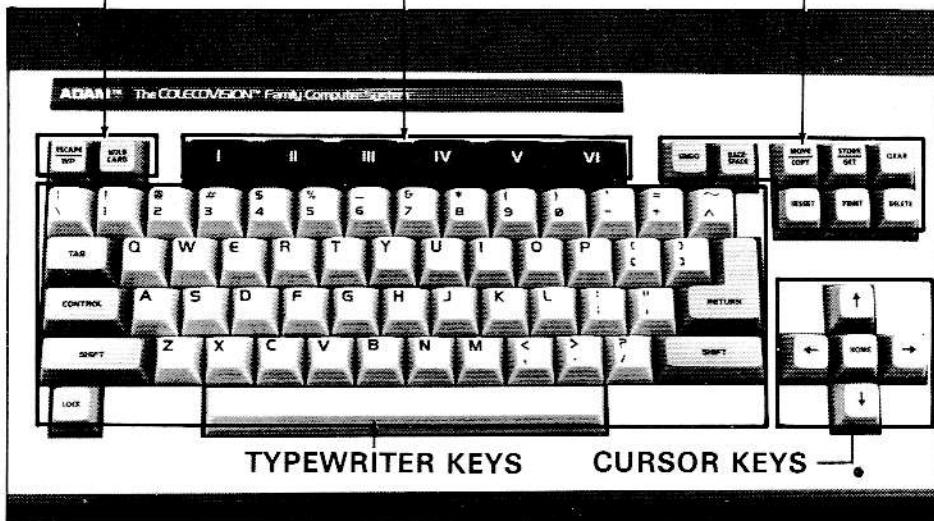
Illustration 5-6-5

Part VI
Appendix

COMMAND KEYS

SMART KEYS

COMMAND KEYS



1

SmartWRITER REFERENCE GUIDE

Almost everyone who uses SmartWRITER will, at some time, want more information on how to perform certain functions or procedures. Each function which can be performed on SmartWRITER Word Processing is listed alphabetically in this chapter, beginning on the next page.

All the functions are performed by using Cursor Control Keys, Command Keys, or Smart Keys. These keys are shown in the picture on the opposite page. Their major functions can be described as follows.

- *Cursor Control Keys:* These are the four arrow keys and the key labeled HOME on the right side of your keyboard. You use Cursor Control Keys to move the cursor around the screen, to scroll text on the screen, or to indicate which file is to be printed or recalled.
- *Command Keys:* You can perform several SmartWRITER functions by using Command Keys. The function of each Command Key is described by its label. Two Command Keys (ESCAPE and UNDO) let you change your mind and restore text. Six other Command Keys (DELETE, INSERT, MOVE/COPY, CLEAR, PRINT, and STORE/GET) are located to the right of the typing keys. When you press one of these six keys, messages appear in the Smart Key Label area on the screen to help guide you through the steps. Two other Command Keys (WILD CARD and CONTROL) are reserved for future use.
- *Smart Keys:* Smart Keys are the six keys (I-VI) at the top of your computer keyboard. Smart Key labels appear on the bottom of your computer screen. These labels change when you press Smart Keys to guide you through a procedure.

Arrow keys

The four arrow keys are located to the right of the typing keys. You use these keys the way you use the SPACEBAR and BACKSPACE keys on a typewriter. Arrow keys move the cursor in the direction of the arrow: up (↑), down (↓), left (←), or right (→).

In the Standard Format, you press the up arrow key to position the cursor at the top of the roller. Then, each time you press the up arrow, the bottom line of text on the screen moves down to the roller.

You press the down arrow key to position the cursor at the bottom of the roller. Then, each time you press the down arrow, the next line of text that has been temporarily stored below the roller is moved up to the roller.

In the Moving Window Format, up arrows and down arrows are also used to move text onto your screen.

Clear (CLEAR)

Sometimes you'll want to erase or "clear" large amounts of typed text. You can do this by pressing the CLEAR key. When you press this key, it's like throwing paper into the wastebasket. (If you change your mind, you can reach into the "wastebasket" and get your words back by pressing UNDO immediately.)

1. Press CLEAR.
 2. When the Smart Key labels change, to indicate how much text you want to clear, press CLEAR SCREEN (V) or CLEAR WK SPACE (VI).
 3. When the Smart Key labels change, press FINAL CLEAR (VI) or CLEAR.
- * After you print or save text and want to start typing something new, use CLEAR to quickly erase your old text.

Color

You can change the background color of your screen using the Smart Key labeled SCREEN OPTIONS (II).

1. Press SCREEN OPTIONS (II).
2. Press COLOR SELECT (II).
3. To change the background color, select one of the following: WHITE (I), GREEN (II), BLACK (III), GRAY (IV), or BLUE (V).
4. Press DONE (VI).

Copy (MOVE/COPY)

COPY allows you to duplicate any quantity of text. The text remains in its original location and is copied to the new location.

1. Press the MOVE/COPY key.
2. Press the Smart Key COPY (VI).
3. Highlight the text to be copied. Use the arrow keys to position the cursor under the first character to be copied and press HI-LITE FIRST (IV). Then position the cursor under the last character to be copied and press HI-LITE LAST (V).
4. Position the cursor where you want the copied text to appear.
5. Press COPY (V).
6. Press DONE (VI). Surrounding text will automatically be readjusted.

Copy a File

Sometimes, you may want to create a copy of a file that you have already typed. This is often referred to as making a "back-up." This assures you of having a copy in case a digital data pack is damaged or your original file is accidentally erased.

1. Insert the digital data pack that contains the file to be copied into the digital data pack drive.
2. Follow the "get" procedure to move the file into your computer's memory.
3. If you want to copy the file onto the same data pack with a different name, complete the store procedure. If you want to copy the file to a different data pack, put the new data pack into the digital data pack drive. Then use the store procedure to save your file under the same or a different name.

Control (CONTROL)

This key, like WILD CARD, is reserved for future use. If you press CONTROL, you will not see any change in your text.

Delete (DELETE)

You can use DELETE to erase any amount of text from your workspace—one letter, several words, a line, a paragraph or more. To indicate exactly what text you want to delete, highlight the information you want to erase.

1. Press DELETE.
2. Highlight the text to be deleted, if it has not yet been highlighted. (Use the Cursor Control Keys, press HI-LITE (IV), use arrow keys, and press HI-LITE OFF (IV).) As you highlight text, a red underline is drawn under text to mark it for deletion.

If you underline too much text, press HI-LITE ERASE (V), use the arrow keys, and then press ERASE OFF (V).

3. Press FINAL DELETE (VI). Highlighted text will be erased from your screen. Text following the deleted material will automatically be moved up so no "holes" exist in your work.
- * This function can also be done by first highlighting text you want to delete, and then pressing the DELETE key twice—rather than using the Smart Keys.
 - * To erase a screenful of text or more, use the CLEAR key.

Delete a File

Sometimes you may want to erase a file that is stored on a digital data pack. When this is the situation, use the procedure below called "delete a file." When you delete a file, it is erased from the digital data pack and can no longer be recalled using the "get" procedure.

1. Insert the digital data pack that contains the file to be deleted into the digital data pack drive.
2. Press STORE/GET to display a File Directory.
3. Press GET (VI).
4. Press DRIVE A (III) to get text from the data pack in Drive A.
5. Using arrow keys, position the File Directory pointer so that it faces the name of the file to be deleted.
6. Press DELETE twice, or press DELETE and FINAL DELETE (VI).

Escape (ESCAPE/WP)

What happens if you're in the middle of setting a new right margin and you decide you've made a mistake? Or, you start to insert some text and realize you're adding it in the wrong paragraph? When you use SmartWRITER, you can "escape" from these situations quickly. How? Just press the ESCAPE/WP key—one of the two "oops" commands available on your ADAM Family Computer.

1. Whenever you are in the middle of a procedure and want to "back out" of it, press ESCAPE/WP. The procedure is cancelled, and text returns to the way it was before you began the procedure.
- * If you highlight text during a procedure, and then press ESCAPE/WP, you will back out of the procedure but the highlighting will remain on your screen. You can erase the highlighting by pressing HI-LITE (IV), using the Cursor Control Keys to remove the red underlining, and then pressing HI-LITE OFF (IV).

Get (STORE/GET)

Once text has been stored, you can bring it back to the screen. This is useful if you want to edit your text or review it without printing it.

1. Make sure the digital data pack containing the file you want is in the digital data pack drive.
2. Press the STORE/GET key.
3. Press GET (IV).
4. Press DRIVE A (III) to get text from the data pack in Drive A.
5. Using arrow keys, position the File Directory pointer on the name of the file you want to get.
6. Press GET FILE (VI).

HI-LITE [HI-LITE OFF] (IV)

To indicate what part of your document you want to copy, move, delete, or print, you must highlight text. Highlighting is the way you tell SmartWRITER Word Processing what text you want to affect during a command. For example, if you want to delete a paragraph, you must highlight the paragraph to be deleted. Then your computer knows what part of your text you want to delete.

Highlighted text is underlined in red on your screen. While text is being highlighted, the HI-LITE OFF label is present. When you want to stop highlighting, press HI-LITE OFF (IV). The HI-LITE label will return.

1. Use Cursor Control Keys to position the cursor under the first character to be highlighted.
 2. Press HI-LITE (IV).
 3. Use arrow keys to move the cursor under each character to be highlighted.
 4. Press HI-LITE OFF (IV).
- * HOME plus an arrow key can be pressed simultaneously to highlight larger quantities of text. HOME plus left or right arrow highlights from the cursor position to the end of the line. HOME plus up or down arrow, or HOME alone, can highlight several lines of text at one time.
 - * If you underline too much text, press HI-LITE ERASE (V), use the arrow keys, and then press ERASE OFF (V).

HI-LITE ERASE [ERASE OFF] (V)

This Smart Key is used to change highlighted text back to unhighlighted text. While highlighted text is being changed back to regular text, the ERASE OFF label is present. When you want to stop the procedure, press ERASE OFF (V). The HI-LITE ERASE label will return.

1. Use Cursor Control Keys to position the cursor under the first character in text to have highlighting removed.
 2. Press HI-LITE ERASE (V).
 3. Use arrow keys to move the cursor under each character to be unhighlighted.
 4. Press ERASE OFF (V).
- * To highlight large amounts of text, see the note under HI- LITE.
 - * HOME and arrow keys can be used together to erase highlighting from large areas of text.

Home (HOME)

HOME is in the center of the arrow keys. When the Standard Format is in effect, HOME moves the top line of text on the screen down to the roller and positions the cursor in the top left position on the screen. If the Moving Window Format is in effect, HOME moves the cursor to the top left position on the screen.

Home + Arrow Keys

If HOME and an arrow key are pressed at the same time, the cursor will move rapidly in the direction of the arrow. If the Standard Format is in effect, HOME plus the left or right arrow moves the cursor to the edge of the roller in the direction of the arrow. HOME plus the up arrow moves the cursor to the top of the roller and at the same time moves the top line of text on the screen down to the roller. HOME plus the down arrow moves the screenful of text that is below the roller onto the screen. If the Moving Window Format is in effect, HOME plus an arrow always moves the cursor to the edge of the screen in the direction of the arrow.

Insert (INSERT)

You can add new text to something you have already typed by using INSERT. You must also use the INSERT key when you want to split a page by adding a "Page End" marker, or add superscripted or subscripted material.

1. Put the cursor where you want to begin inserting text.
 2. Press INSERT. The text to the right of the cursor is temporarily moved off the screen.
 3. Type the text you want inserted. This can be any amount.
 4. When finished, press DONE (VI). The text that moved away when INSERT was pressed reappears immediately after the text you just typed.
- * You must use INSERT to add a Page End marker to existing text.
 - * You must use INSERT to add superscripted or subscripted material to existing text.

Line Spacing

When you use SmartWRITER Word Processing, you can change the spacing between lines. For example, you can use single (1) spacing, one-and-a-half (1.5) spacing, double (2) spacing, two-and-a-half (2.5) spacing, triple (3) spacing, and so forth.

Line spacing changes are not evident on the screen. That is, you will not see text displaying in double or triple spacing. You will see this effect only when you print your text.

The system is preset for single spacing throughout a document.

1. Press MARGIN/TAB/ETC (I).
2. Press LINE SPACING (V).
3. To change the spacing between lines, press UP (IV) to increase the spaces between lines, or press DOWN (V) to decrease the spaces.
4. Press DONE (VI).

Margins—Left and Right

SmartWRITER Word Processing comes with standard left (10) and right (70) margin settings. These settings can easily be changed using MARGIN/TAB/ETC (I). Changes in left and/or right margin settings can be seen in the horizontal margin scale at the top of the page.

1. Press MARGIN/TAB/ETC (I).
2. Press HORIZ MARGIN (II).
3. To indicate whether you want to change the left or right margin, press LEFT # (III) or RIGHT # (IV).
4. To indicate where you want to set your margin, use left and right arrow keys to position the indicator at the top of your screen. Notice that the number of the new setting appears in the Smart Key Label.
5. Press DONE (VI), or, if you now want to set top and bottom margins, press TO: VERT MARGIN (V).

Margins—Top and Bottom

SmartWRITER Word Processing comes with standard top (6 lines or 1") and bottom (6 lines or 1") margin settings. These settings can be easily changed using MARGIN/TAB/ETC (I). Changes in the top and bottom margin settings can be seen in the horizontal margin scale on the left side of the screen.

1. Press MARGIN/TAB/ETC (I).
2. Press VERT MARGIN (III).
3. To indicate whether you want to change the top or bottom margin, press TOP # (III) or BOTTOM # (IV).
4. To indicate where you want to set your margin, use up and down arrow keys to position the indicator at the left side of your screen. Notice that the number of the new setting appears in the Smart Key Label.
5. Press DONE (VI), or, if you now want to set right and left margins, press TO: HORIZ MARGIN (V).

Move (MOVE/COPY)

MOVE allows you to remove any quantity of text from its original location and move it to a new location.

1. Press the MOVE/COPY key.
 2. Press the Smart Key MOVE (V).
 3. Highlight the text to be moved as follows. Highlight the first character of the text to be moved by positioning the cursor under the first character and pressing HI-LITE FIRST (IV). Then position the cursor under the last character to be moved and press HI-LITE LAST (V).
 4. Move the cursor to the location where the text is to be moved.
 5. Press MOVE (II) to move the text. Surrounding text automatically re-adjusts.
- * When moving long passages of text, please note that you can move only the highlighted text on the current screen. If text was highlighted and then pushed off the screen, it will remain highlighted, but must be brought back onto the screen before it can be moved. In Moving Window Format, "current screen" includes the text to the left and right of the text visible on the TV screen.

Moving Window Format

When you first begin using SmartWRITER Word Processing, it is in the Standard Format. In the Standard Format, you type on a roller at the bottom of your screen. Then, as you continue to type, your text moves up the screen.

By using SCREEN OPTIONS (II) you can select the Moving Window Format. In the Moving Window Format, text you type appears at the top of your screen and not in the roller. As you continue to type, your text moves down the screen.

In the Moving Window Format, your screen is like a window that moves over your text. You see part of what have typed as you screen moves in four-character increments across the text.

Text you type in the Moving Window Format can be as wide as 80 characters. It will appear on your screen exactly as it will when it is printed (with the exception of line spacing, which always displays as single-spaced). For this reason, Moving Window Format is usually used when you want to type columns or charts.

1. Press SCREEN OPTIONS (II).
2. Press MOVING WINDOW (VI).

* When you insert, delete, and make other changes in the Moving Window Format, your text will automatically be re-adjusted.

Page End

SmartWRITER Word Processing automatically fills each page according to the top and bottom margins you set. If you did not change the standard setting, then each page is 60 lines long. You can create shorter pages, however, by using the END PAGE feature.

You can also use this feature to split any pages you have already typed. When you do this, you must follow the second procedure.

ENDING A SHORT PAGE AS YOU TYPE

1. Position the cursor where you want the page to end.
2. Press MARGIN/TAB/ETC (I).
3. Press END PAGE (VI). A special Page End marker (the letter "E" inside a circle) now appears on your screen.

SPLITTING AN EXISTING PAGE

1. Position the cursor where you want the page to end.
 2. Press INSERT.
 3. Press END PAGE (IV).
 4. Press DONE (VI). Text which follows the "end page" marker is now on a new page.
- * NOTE: If the first procedure is not functioning, you can use the second procedure to achieve the same results.

Paper Size

The most commonly used paper is 11" long. SmartWRITER Word Processing is preset to this "letter" size paper. By using MARGIN/TAB/ETC, you can also use 14" long paper. This size paper is sometimes called "legal" size.

1. Press MARGIN/TAB/ETC (I).
2. Press TYPE OF PAPER (I).
3. The paper size which SmartWRITER is currently set for is listed in the message area above Smart Keys I and II. To change the paper size setting, press LETTER 11 (III) or LEGAL 14 (IV).
4. Press DONE (VI), or, if you now want to set top and bottom margins, press TO: VERT MARGIN (V).

Paper Type

You can insert single sheets of paper into your printer to print text or you can print your work on fanfold paper. Some people call fanfold paper "computer paper." It is a continuous length of paper that has special holes on each side and is divided into perforated sheets.

You can use fanfold paper with or without a "tractor feeder"—a device that holds the computer paper firmly in place.

Indicate what type of paper you want to use during the printing procedure.

1. Press PRINT.
 2. Specify the amount of text you want to print by pressing PRINT HI-LITE (III), PRINT SCREEN (IV), or PRINT WK-SPACE (V).
 3. The type of paper SmartWRITER is expecting you to use appears on your screen in the message area. To change from single sheet to fanfold, or from fanfold to single sheet, press Smart Key II.
- * Refer to the PRINT procedure for the remainder of the print steps.

Print (PRINT)

After you make corrections and changes to your work, you are ready to print. Text must be on the screen in order for it to be printed. (Thus, if you have stored a document, you must use the GET procedure to bring it to the screen. Then you can print your text. The text on the digital data pack is not changed and remains there. You can clear the copy off the screen when you have finished printing it.)

Before you begin the print procedure, make sure paper is inserted in your printer. To get the correct margin at the top of your paper, put the top edge of your paper just under the paper bar or paper bail.

1. Press PRINT.
2. Press the appropriate Smart Key to specify the amount of text you want to print—only highlighted text on the screen (III), all the text that is on the screen (IV), or all the text in the workspace (V).

If you want to print highlighted text (III), be sure to highlight text before pressing PRINT. If you have not done this, then either press ESCAPE/WP or Smart Key III and restart the procedure.

3. Decide whether you want to change the paper type. The type of paper SmartWRITER expects you to use appears on the screen in the yellow message area. To change from single sheet to fanfold or from fanfold to single sheet, press Smart Key III. If you do not want to change paper type, then skip this step.
 4. Decide if you want to print your text with page numbers automatically provided by the system. If so, press AUTO PAGE #? (IV). Then decide whether you want to number the first page with a number other than 1. If so, press 1ST PAGE # IS 1 (III) as many times as necessary to cause the appropriate first page number to appear in the Smart Key label. If you do not want automatic page numbering, skip this step.
 5. When you are ready to print, press PRINT (VI). Your text begins printing.
- * To stop the printer, press STOP PRINT (V). To restart it, press PRINT (V) again.
 - * If you don't want to change the paper type or use automatic page numbering, only complete steps 1, 2, and 5.

Rename a File

Sometimes, you may want to change the name of a file that you have stored. With SmartWRITER Word Processing, this is easy.

1. Use the GET procedure to bring a file back to your screen.
2. Use the STORE procedure to store the file, only this time give the file a different name.
3. If you want to erase the file under its original name, use the DELETE A FILE procedure.

Replace

You can use SmartWRITER to automatically substitute one word (or group of words) for another word (or group of words) throughout your workspace. This procedure can save you time in updating information and correcting spelling errors.

For example, you may type a report that contains the name "Smith" in several places. Later, you discover that the correct spelling of the name is "Smithe." You could use this procedure to automatically correct your text and replace every "Smith" with "Smithe." When you replace every occurrence of a word, it is referred to as "replace all." However, sometimes you may want to replace some occurrences of a word. Perhaps the word "Tuesday" should only occasionally be replaced with "Wednesday" in a report. In this case, you would need to look at each occurrence of the word "Tuesday" and decide whether or not to replace it with "Wednesday." This procedure is referred to simply as "replace."

The two procedures are performed identically except for step 7. NOTE: SEARCH searches from the position of the cursor on. Therefore, to search through the entire document, the first line of the document must be on the roller before you begin this procedure.

1. Press SEARCH (III).
2. Type the old text you want to find.
3. Press START SEARCH (VI).
4. Press REPLACE (V).
5. Type the new text that you want to replace the old text.
6. Then decide whether you want to "replace all" or simply "replace" some occurrences, and complete the appropriate step.
 - To replace all instances of old text with new text, press REPLACE ALL (VI).
 - To replace this instance of old text with new text, press REPLACE (V). Then, to locate the next instance of the old text, press SEARCH NEXT (IV). Each time the cursor locates old text, you must decide whether to replace it (press REPLACE) or leave it (press SEARCH NEXT).
7. To end the procedure early, press DONE (VI); otherwise the procedure ends automatically.

Reset

There are two reset switches on ADAM and Expansion Module #3. Here is how they are used.

- * On ADAM: When the system is in typewriter mode and you want to erase text, pull the Computer Reset slide switch (which is to the left of the cartridge port) toward you. If the system is in typewriter mode and you want to play a game cartridge, pull the Cartridge Reset slide switch (which is to the right of the cartridge port) toward you.
- * On Expansion Module #3: When the system is in typewriter mode and you want to erase text, pull the Computer Reset slide switch (which is located on top of the unit near the front) toward you. If you want to go from typewriter mode to playing games, pull the Cartridge Reset slide switch (which is on the ColecoVision console) toward you.

Screen Options

The Smart Key labeled SCREEN OPTIONS (II) gives you many choices. You can change the color you see or the sound you hear in SmartWRITER Word Processing. You can also move between the Standard Format and the Moving Window Format using this key. See the information listed under "color," "sound," or "Moving Window Format."

Search

Using SEARCH (III) you can find text within a workspace. You may want to change the text you are looking for, or you may want to check some information. You can also use this procedure to move to a specific section of text in a workspace.

NOTE: SEARCH begins looking for text from the position of the cursor. Therefore, to search through an entire document, the first line of the document must be on the roller before you begin this procedure.

1. Press SEARCH (III).
2. Type the text you want to find.
3. Press START SEARCH (VI).
4. If necessary, to find the next instance of text, press SEARCH NEXT (IV).
5. If necessary, continue repeating Step 4.
6. To end the procedure early, press DONE (VI); otherwise the procedure ends automatically.

Sound

SmartWRITER Word Processing communicates with you by using a variety of sounds. For example, when you have almost filled up the maximum area available for your workspace, you will hear a special sound.

Using SCREEN OPTIONS (II), you may select one of three different sound levels: no sound, partial sound, or full sound. When SmartWRITER Word Processing begins, the sound feature is set at "full."

1. Press SCREEN OPTIONS (II).
2. To select a different sound setting, press one of the following keys: NO SOUND (III), PARTIAL SOUND (IV), or FULL SOUND (V).

Standard Format

When you first begin using SmartWRITER Word Processing, it is in the Standard Format. In the Standard Format, you type on a roller at the bottom of your screen. One line of text fills up your roller. Then, as you continue to type, your text moves up the screen.

By using SCREEN OPTIONS (II) you can select the Moving Window Format. Later, you can use SCREEN OPTIONS (II) again to return to the Standard Format after using the Moving Window Format.

1. Press SCREEN OPTIONS (II).
2. Press STANDARD FORMAT (VI).

Store (STORE/GET)

You can store any quantity of text—from a word to an entire workspace. To store text, a digital data pack must be inserted into the computer. You must also type a name (up to 10 characters) for your text. This name is called a “file name.”

1. Make sure a digital data pack is inserted in a digital data pack drive.
2. Press the STORE/GET key.
3. Decide how much text you want to store and press the appropriate Smart Key: STORE HI-LITE (III), STORE SCREEN (IV), or STORE WK-SPACE (V).

To store highlighted text (III), be sure to highlight text before pressing STORE/GET. If you did not do this, then press ESCAPE/WP or Smart Key III and begin the procedure again.

4. To indicate where you want to store your work, press DRIVE A (III) or another Smart Key.
5. Type a file name for your text (up to 10 characters).
6. Press STORE (VI).

Subscript

Subscripted characters appear below the regular line of type. Before you subscript characters, be sure to set your line spacing at 1 1/2 or greater. Be sure to type text up to the character to be subscripted before doing step 1.

1. Press SUPER/SUBSCRIPT (VI).
 2. Press SUBSCRIPT (V). A special "Begin Subscript" marker (an L-shaped character facing to the right) appears on your screen.
 3. Type the text to be subscripted.
 4. Press DONE (VI). A special "End Subscript" marker (an L-shaped character facing to the left) appears on your screen.
- * To erase subscript markers, use DELETE, BACKSPACE, or CLEAR.
 - * To add subscripted text, use INSERT.

Superscript

Superscripted characters appear above the regular line of type. Before you superscript characters, be sure to set your line spacing at 1 1/2 or greater. Be sure to type characters up to the character to be superscripted before doing step 1.

1. Press SUPER/SUBSCRIPT (VI).
 2. Press SUPERScript (VI). A special "Begin Superscript" marker (an upside-down L-shaped character facing to the right) appears on your screen.
 3. Type the text to be superscripted.
 4. Press DONE (VI). A special "End Superscript" marker (an upside-down L-shaped character facing to the left) appears on your screen.
- * To erase superscript markers, use DELETE, BACKSPACE, or CLEAR.
 - * To add superscripted text, use INSERT.

Tabs

Tabs (sometimes called “tab stops” or “tab settings”) can be used to indent the first line of a paragraph or each item in a list. Tabs are also used when typing name and address lists, budgets, and charts—especially when the Moving Window Format is in use.

To begin setting tabs, press the MARGIN/TAB/ETC Smart Key (I). You are then prompted through the procedure by yellow messages and labels which change in the Smart Key Label area. During this procedure, you have the opportunity to set one tab at a time, clear one tab at a time, or clear all tabs. You will use the following steps to clear and set tabs.

TO CLEAR TABS

1. Press MARGIN/TAB/ETC (I).
 2. Press TAB (IV).
 3. Use the left and right arrow keys to position the cursor at an existing tab stop as shown on the margin scale at the top of the screen.
 4. Press TAB CLEAR (IV).
 5. Repeat steps 3 and 4 to clear additional tabs.
 6. When finished, press DONE (VI).
- * To clear all tab stops, complete the first two steps, and then press ALL CLEAR (V), followed by DONE (VI).

TO SET TABS

1. Press MARGIN/TAB/ETC (I).
2. Press TAB (IV).
3. Use the left and right arrow keys to position the cursor where the new tab stop should be. Refer to the top margin scale as you do this.
4. Press TAB SET (III) to set a tab stop.
5. Repeat steps 3 and 4 to set additional tabs.
6. When finished, press DONE (VI).

Undo (UNDO)

UNDO is sometimes referred to as one of the "oops" Command Keys. It will allow you to change your mind immediately after completing any of the four functions—DELETE, CLEAR, REPLACE, and REPLACE ALL. (Notice that all these functions destroy or change text.)

All you do is press the UNDO key and the effects of the procedure are erased. For example, if you deleted a paragraph, all you have to do to get it back is press UNDO. This must be done *immediately* after having *completed* the procedure. (If you are in the middle of a procedure and change your mind, don't use UNDO. Instead, use ESCAPE to "back out" of the procedure.)

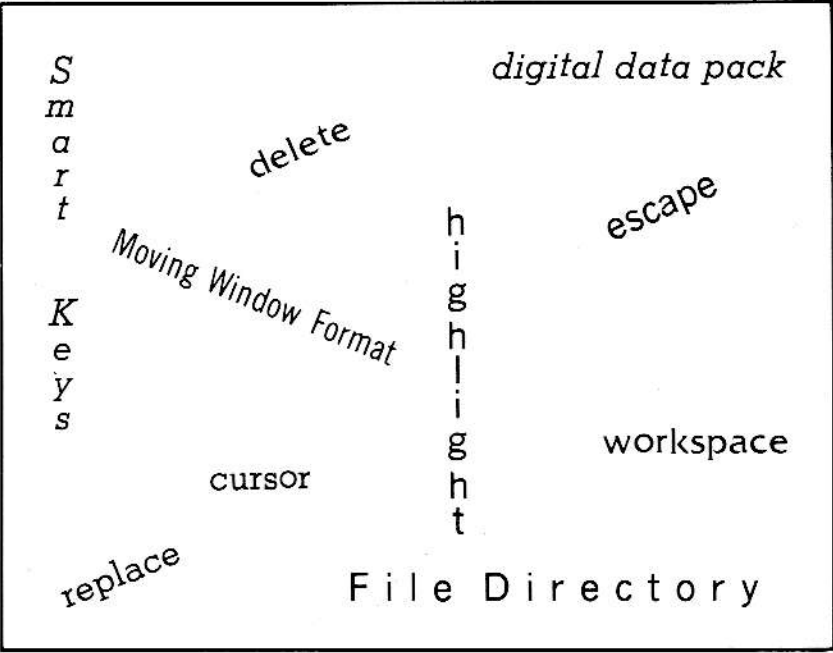
1. Immediately after having completed a "destructive" procedure, press UNDO. The effect of the procedure will be "undone."
- * If you have highlighted text during a procedure and then press UNDO, your screen will return to normal and text will remain highlighted. You can erase the highlighting by pressing HI-LITE ERASE (V), using the Cursor Control Keys to remove the red underlining, and then pressing ERASE OFF (V).

Wild Card (WILD CARD)

This key is reserved for future use. If you press WILD CARD, you will not see any effect on your text.

WP (ESCAPE/WP)

Whenever you turn your computer on and want to use SmartWRITER rather than the electronic typewriter, press the key labeled ESCAPE/WP. To get out of the word processing mode, you can use a reset switch.



Glossary of SmartWRITER Terms

Each word processing system uses some special words to describe its functions. Some of these words have become fairly standard word processing terms, and others are unique to each system.

You may find the list of terms below helpful to know when you are using SmartWRITER word processing.

<i>TERMS</i>	<i>EXPLANATIONS</i>
Arrow Keys	There are four Arrow Keys located to the right of the typing keys. These keys are used to move the cursor in the direction of the arrow: up, down, left, or right.
clear	Whenever you want to erase a screenful or more of text, you can use the CLEAR key. You can clear either a screenful at a time or the entire workspace.
Command Keys	Currently, eight of the ten Command Keys are used to do or begin SmartWRITER procedures. The label on each Command Key identifies what pressing that key will do—delete, insert, print, clear, etc.
copy	This procedure allows you to duplicate any amount of text which you mark or highlight. Text remains in its original location and is copied to the new location. Use the MOVE/COPY key to begin this procedure.
cursor	The small, bright “underline” character that appears on your screen is called a cursor. It shows you where the next letter or number will be typed on the screen. Sometimes cursors have other shapes, such as the triangle shape in the File Directory.

- Cursor Control Keys** There are a total of five Cursor Control Keys available on ADAM—four Arrow Keys and HOME. These keys are located to the right of the typing keys and are used to move the cursor or scroll text.
- delete** This procedure is used to erase any amount of text from your workspace. To indicate exactly what text you want to delete, you highlight (or mark) the text to be deleted. Use the DELETE key to begin this procedure.
- digital data pack** The letters, reports, and other documents that you type and want to save for future use are stored on a digital data pack. A digital data pack looks like an audio cassette tape, but they are not interchangeable. A data pack operates at very high speed.
- digital data pack drive** In order for ADAM to store your letters and reports, a digital data pack must be inserted into a digital data pack drive. The "drive" is comparable to the turntable on a record player, in that the drive holds the data pack so that it can be used. One drive comes with the system; a second drive can be purchased and added onto the system.
- document** Anything you type on a word processor—a letter, report, chart, etc. — is often referred to as a "document."
- edit** This word processing term is often used to describe one of the best features of a system—the capability to correct mistakes or change text for other reasons. With a word processor, you do not retype the entire text—only the words which need changes.
- error message** When certain procedures are not performed correctly, the system displays an error message in the Smart Key label area on the screen. This message helps identify what you did incorrectly, so that you may complete the procedure.

- escape** Some word processors let you "back out" of the middle of a procedure and return text to the way it was before you started the procedure. On SmartWRITER, this feature is called escape and is performed by pressing the ESCAPE/WP key.
- fanfold paper** Sometimes fanfold paper is called "computer paper." It is a continuous length of paper that has special holes on each side and is divided into perforated sheets. It is very useful if you do a lot of printing because you don't have to hand-feed each sheet of paper.
- file** The letter, report, or other document you store on a digital data pack is referred to as a "file." Once stored, a file can be recalled to the screen by using the GET procedure.
- File Directory** The names of all the files (letters, documents, etc.) that are stored on a digital data pack are recorded on a File Directory. A File Directory is a part of each digital data pack. By using the STORE/GET key you can see the names of any files stored on a digital data pack.
- File Name** In order to store a file (letter, report, or other document), you must type a name for it. You do this after pressing the STORE/GET key when the system prompts you to type a File Name. To recall a document, you must press the STORE/GET key and point to the File Name you want.
- format** This word processing term is used to describe the appearance (or layout) of a printed page. The format of a printed page is determined by the top and bottom margins, left and right margins, and the line spacing.
- get** The GET procedure is used to recall text to the screen that has previously been stored on a digital data pack. Use the STORE/GET key to begin this procedure and to display the names of the files stored on that data pack.

highlight (or HI-LITE)	To highlight text is to mark it with a red underline. This indicates the part of your text you want to copy, move, delete, or print. To highlight text, use the Smart Key labeled HI-LITE. After highlighting text, you must turn the highlight off.
home	HOME is one of the five Cursor Control Keys which are located to the right of the typing keys. In Standard Format, home is the top left position on the roller. In Moving Window Format, home is the top left position on the screen.
horizontal margins (or HORIZ MARGIN)	Left and right margins are referred to collectively as "horizontal margins." SmartWRITER comes with standard settings of left margin 10 and right margin 70. These settings can be changed by using the Smart Key labeled MARGIN/TAB/ETC.
insert	Insert is a word processing term meaning "to add." By using the INSERT key, you can add new text to the middle of something you have already typed.
message	Whenever you press certain Command Keys or Smart Keys, there is a change in the information in the Smart Key label area of the screen. Sometimes a message displays in yellow in part of the Smart Key label area. The message helps guide you through the procedure.
move	The move procedure is used to relocate text in your document. Use the MOVE/COPY key to begin this procedure and then highlight the text to be moved.
Moving Window Format	This format is an alternative to the Standard Format. When you use the Moving Window Format, text will print exactly as it appears on the screen—except for line spacing. For this reason, Moving Window Format is usually used when you want to type columns, charts, and certain reports.

- procedure** Tasks which a word processor can be used for are called "procedures." Some procedures are insert, delete, move, copy, search, and replace.
- replace** With SmartWRITER, you can search for specified text and replace it with some other text. You can have the system do this automatically for you throughout the document—a procedure called "replace all" (or global replace on some other systems). Or you can search and replace selectively—a procedure simply called "replace."
- reset** There are two reset switches on an ADAM computer. The Computer Reset is to the left of the cartridge port; the Cartridge Reset is to the right of the cartridge port. Computer Reset is used to erase text from the typewriter mode, and to reset a game that's on a digital data pack. Cartridge Reset is used to get the system ready to play a game that is on a cartridge.
- roller** The roller is the black area on the Standard Format that looks like the platen on a typewriter. Text is typed on the roller in the Standard Format.
- scroll** As you type text in a word processor, the text "moves" or scrolls up the screen. When it reaches the top of the screen, it scrolls out of view. It can be scrolled back at any time. This movement of text up and down the screen is called "scrolling." To scroll text, use the Cursor Control Keys.
- search** The search procedure is used to find any text you specify anywhere within your document. Use the Smart Key labeled SEARCH to begin this procedure.

- Smart Key labels** Whenever the ADAM Family Computer is in the word processing mode, you can see Smart Key labels at the bottom of the screen. These labels are numbered I through VI and tell you what function will occur when you press the corresponding Smart Key (I-VI) at the top of the keyboard. These labels change when certain Smart Keys or Command Keys are pressed. Sometimes a message will display in the Smart Key label area to help guide you through a procedure.
- Smart Keys** Smart Keys are the six keys labeled I through VI at the top of the keyboard. Their labels appear at the bottom of the screen and change whenever you press certain keys.
- Standard Format** The Standard Format is preset on your system. In this format, you type on a roller at the bottom of your screen. As you continue to type, text scrolls up and, eventually, off the screen. Text can be scrolled back onto the roller at any time.
- store** The store procedure is used to save any amount of text on a digital data pack. Once stored, text can be recalled. Text is stored under a file name. Use the STORE/GET key to begin a store procedure.
- subscript/superscript** These procedures are used when you want to type formulas, equations, or any text below or above the typing line. Subscripted text appears below the typing line, and superscripted text above it.
- vertical margins** The margins that are at the top and bottom of the page are referred to as vertical margins. Top and bottom margins are automatically set at one inch (six typing lines). The relative positions of top and bottom margins appear on the margin scale on the left side of your screen.

workspace

When the term workspace (WK-SPACE) is used on a Smart Key label, it refers to the entire document you have typed. This document (letter, report, etc.) can be as short as a few words, or several pages long. While you are working on that document, it is kept in the computer's workspace or memory. Text in the workspace can be cleared, printed, or stored. Once the computer is turned off, everything in its workspace is erased.

undo

If you have just deleted, cleared, or replaced text, you can undo the procedure and return your text to the way it was before beginning the procedure. How? Just press the UNDO key.

```
10 REM MILEAGE
20 PRINT "This program determines how"
25 PRINT "many miles per gallon you"
30 PRINT "get while driving."
40 PRINT
50 PRINT "To begin, what was your"
55 PRINT "mileage reading the last time"
60 PRINT "you put gas in your car?"
70 PRINT
80 PRINT "Please type the correct number"
90 PRINT "and press RETURN. (NOTE: Do"
100 PRINT "not use commas.)"
```

1. Now, please type
you are in years"
RETURN."

```
w ";n$";"  
ADAM computer and I"  
INTRO  
T "Ple"
```

```
110 PRINT  
120 INPUT a  
130 PRINT:PRI  
140 PRINT "W  
145 PRINT "t  
150 PRINT  
160 PRINT "P  
170 PRINT "e  
180 PRINT  
190 INPUT b
```

```
10 REM MATH DEMO
20 PRINT "Please type your name and"
30 PRINT "press RETURN."
40 INPUT n$
50 PRINT:PRINT
60 PRINT "Hello ";n$;". This is a math"
70 PRINT "demonstration program."
80 PRINT
90 PRINT "If you will give me 2 numbers,"
95 PRINT "I will show you how good I"
PRINT "am at math."
```

se your name and"
RN."

Now, please type"
a friend and"
."

```
200 HOME  
210 PRI  
220 PRI  
230 PRI  
240 PR  
250 PR  
260 PR  
270 I  
280 H  
290 F  
300  
310  
320  
330
```

```
10 PRINT " ADJ";  
15 RETURN  
20 PRINT " NOUN"  
25 RETURN  
30 PRINT " ADJ";  
35 RETURN  
40 PRINT " NOUN"  
45 RETURN  
50 PRINT " ADJ";  
55 RETURN  
60 PRINT " NOUN"  
65 RETURN  
70 PRINT " ADJ";  
75 RETURN  
80 PRINT " NOUN"  
85 RETURN  
90 PRINT " ADJ"  
95 RETURN  
100 PRINT " NOUN"  
105 RETURN  
110 PRINT " PHRASE"  
115 RETURN  
120 PRINT " PHRASE"  
125 RETURN  
130 PRINT " PHRASE"  
135 RETURN  
140 PRINT " PHRASE"  
145 RETURN  
150 PRINT " PHRASE"  
155 RETURN  
160 PRINT " PHRASE"  
165 RETURN  
170 PRINT " PHRASE"  
175 RETURN  
180 PRINT "...";  
185 RETURN  
190 PRINT  
195 RETURN
```

n, type a number and
RETURN. (Please do"
commas.)"

```
210 PRINT "Now, how m  
220 PRINT "did you us  
230 PRINT  
240 PRINT "Please type  
250 PRINT "and press R  
260 PRINT  
270 INPUT c  
280 HOME
```

```
PRINT  
PRINT  
INPUT a  
TIME  
INT  
INT "As  
INT "I a  
INT "can  
INT  
INT "For  
INT "orga  
T "good
```

```
10 REM COOKBOOK  
15 TEXT:HOME  
20 PRINT  
30 PRINT "Hello. I am your new"  
40 PRINT "cooking assistant. Tell"  
50 PRINT "me the main ingredient of"  
60 PRINT "your meal and I will give"  
70 PRINT "you some recipe suggestions."  
80 PRINT:PRINT  
90 PRINT "*****MAIN INGREDIENT*****"  
100 PRINT "  
110 PRINT "  
120 PRINT "  
130 PRINT "  
1) GROUND BEEF"  
2) CHICKEN"  
3) LAMB CHOPS"  
4) MEAT-LESS"
```

(This program continued on next page.)
future.
T
you later ";n\$";"
of a \$
HOME
COLOR=6
HLIN 15, 20 AT 25
VLIN 15, 20 AT 25
VLIN 25, 20 AT 35
VLIN 25, 35 AT 15
PRINT "I hope you 20
PRINT "this demonstration
using ADAM"
Before long"
wing ";f\$
to use ADAM."

```
200  
210  
220  
230  
240  
250  
260  
270  
280  
290  
300 PRINT "Good-bye ";n$;". See
```

you"
n very"
,"

SmartBASIC Reference Guide

In Part 4 and Part 5 you were introduced to SmartBASIC and how you can use it to write your own computer programs. If you have completed the lessons in Parts 4 and 5, you know how easy and fun writing your programs can be! If you want to learn more about writing your own programs, you will find this chapter helpful.

The information in this chapter is divided into four areas. The first section contains information about numeric and string variables (e.g., a, b\$). The second section covers operators (e.g., +, <). The use of punctuation symbols (e.g., :, ;) is discussed in the third section. Commands, statements, and functions (e.g., RUN, PRINT, ABS) are explained in the final section of this chapter.

VARIABLES

In SmartBASIC there are two different types of variables. If a variable represents a number, it is called a "numeric variable." If a variable represents letter(s) and/or word(s), it is called a "string variable."

A numeric variable must be represented by a single letter. For example, the statement INPUT a could be used in a program that asks the user to type a number.

A string variable must be represented by a single letter followed by a dollar sign. For example, the statement INPUT b\$ could be used in a program that asks the user to type letter(s) and/or word(s).

OPERATORS

SmartBASIC uses two mathematical symbols that you will recognize to perform addition and subtraction operations. The symbols ADAM uses to perform multiplication and division might be new to you.

<i>Operation</i>	<i>Operator</i>
Addition	+
Subtraction	-
Multiplication	*
Division	/
Raising to a power (exponentiation)	^

Illustration 6-3-2

As explained in Part 4, SmartBASIC performs mathematical operations in a very specific order. First, operation(s) within parentheses are performed. Then, working from left to right, multiplication and division operations are completed. Finally, addition and subtraction operations are completed—again working from left to right.

In connection with IF/THEN and IF/GOTO statements, SmartBASIC uses six other operators.

<i>Meaning</i>	<i>Operator</i>
Equal to	=
Less than	<
Greater than	>
Less than or equal to	<=
Greater than or equal to	>=
Not equal to	<>

Illustration 6-3-3

As listed on the previous page, these operators can be used to analyze values. For example, the statement IF a<=6 GOTO 700 means—if the value of a is less than or equal to six then go to line 700. If the value of a is greater than six, the next line will be executed in order. Three logical operators (AND, OR, and NOT) can be used to combine IF/THEN and IF/GOTO formulas. For example, IF a<=6 AND b>=2 GOTO 700 means—if the value of a is less than or equal to six and the value of b is greater than or equal to two then go to line 700.

PUNCTUATION SYMBOLS

Inside quotation marks, in a PRINT statement, punctuation symbols are used in the same way as when you write. Three punctuation symbols (comma, semicolon, and colon), however, have special uses in SmartBASIC.

In PRINT statements, the comma (,) can be used to indicate that the information that follows should be displayed beginning at the left edge of your screen or in the middle of your screen—whatever location is next on the current or next line. For example, if you were to write the following line in a program:

```
10 PRINT 1, 2, 3
```

Illustration 6-3-4

This is what you would see when the program was run:

```
1           2
3
```

Illustration 6-3-5

The semicolon (;) can also be used in PRINT statements. It tells your computer to display what follows without adding a blank space. For example, if you were to write the following line in a program:

```
10 PRINT "AD";"AM"
```

Illustration 6-3-6

This is what you would see when the program was run:

```
ADAM
```

Illustration 6-3-7

A colon (:) can be used to combine more than one statement on a line. For example, if you want to create two blank lines when your program runs, you could write two lines that just contain PRINT. Using a colon, however, you can combine these two lines as follows—PRINT:PRINT.

COMMANDS, STATEMENTS, AND FUNCTIONS

The rest of this chapter explains the commands, statements, and functions in SmartBASIC. The entries in this section are arranged in alphabetical order. Of course, you don't need to learn all about SmartBASIC to use your ADAM Family Computer System. However, you may find the explanations and sample programs in this section helpful. NOTE: Advanced commands, statements, and functions are indicated by the notation (**advanced**).

ABS

ABS is a function that returns the absolute value of a number. The absolute value of a positive or negative number is the same number without the plus or minus sign. In other words, the absolute value of -3 is 3. The absolute value of $+56.78$ is 56.78. NOTE: When using ABS in a program, be sure to include the number which follows it within parentheses [()].

Sample Program

```
10 PRINT ABS(-273)
20 PRINT ABS(+367.8)
30 PRINT ABS(-4.567)
```

Run

```
273
367.8
4.567
```

APPEND (**advanced**)

APPEND is a command used to add data to an existing sequential file. APPEND should be followed by a filename, a comma (,), the letter D, and a drive number (1 or 2). NOTE: A sequential file should be closed (using the CLOSE command) before APPEND is used.

ASC

ASC is a function that returns the ASCII number for any character. Every letter, number, or symbol that you can type on your computer has a corresponding ASCII number. Since the letters of the alphabet correspond to a progressive series of numbers (A=65, B=66, C=67 . . . X=88, Y=89, Z=90) the ASC function can be used to alphabetize information.

Sample Program

```
10 PRINT ASC("T")
20 PRINT ASC("U")
30 PRINT ASC("V")
```

Run

```
84
85
86
```

ATN (**advanced**)

ATN is a function that returns the measurement (in radians) of an angle. ATN is the opposite of TAN and must always be followed by a number within parentheses. For example, ATN(X) returns the measurement of an angle whose tangent is X.

BLOAD (**advanced**)

BLOAD is a command used to load images from binary files into your computer's memory. BLOAD should be followed by a filename, a comma (,), the memory location where the file will be loaded, a comma (,), and a drive number. If an address or drive are not specified, your computer will use default settings. NOTE: Information to be retrieved with BLOAD must first be stored using BSAVE.

BRUN (**advanced**)

BRUN is a command used to load binary files and automatically move to a starting address. BRUN should be followed by a filename, a comma (,), the memory location where the file will be loaded, a comma (,), and a drive number. If an address or drive are not specified, your computer will use default settings.

BSAVE (advanced**)**

BSAVE is a command used to store binary information. BSAVE should be followed by a filename, a comma (,), the starting address of the information to be saved, a comma, the length of the information to be saved, a comma (,), and a drive number. NOTE: Both the address and length values must be included with BSAVE and may appear in decimal or hexadecimal numbers. The use of a drive number is optional.

CALL (advanced**)**

CALL is a statement used to execute a machine-language routine stored at a specific location within the computer's memory. CALL must be followed by a number between 0 and 65535 to indicate the location of the machine language routine.

CATALOG

CATALOG is a command that allows you to display a list of programs stored on a digital data pack. To use the CATALOG command, you must follow three steps.

1. Insert a digital data pack into the digital data pack drive.
2. Type CATALOG.
3. Press RETURN.

CHR\$

CHR\$ is a function that takes an ASCII number and converts it to the appropriate number, letter, or symbol.

Sample Program

```
10 PRINT CHR$(72)
20 PRINT CHR$(69)
30 PRINT CHR$(76)
40 PRINT CHR$(76)
50 PRINT CHR$(79)
```

Run

```
H
E
L
L
O
```

CLEAR

CLEAR is a statement that erases all values assigned to numeric and string variables.

Sample Program

```
10 LET n=42
20 LET s$="THIS IS A TEST"
30 CLEAR
40 PRINT "n = ";n
50 PRINT "s$ = ";s$
```

Run

```
n = 0
s$ =
```

CLOSE

CLOSE is a command used to close a file that has been previously opened. To use the CLOSE command, complete the following steps. (NOTE: In the lines below, PROGRAM stands for the name of your program.)

1. Type the following line to begin your program:

```
10 d$=CHR$(4)
```

2. Type the following line as the next line of your program:

```
20 PRINT d$;"OPEN PROGRAM"
```

3. Type the information to be added to or deleted from the file.
4. Type the following line to end your program and close your file:

```
100 PRINT d$;"CLOSE PROGRAM"
```

CLRERR

CLRERR is a statement used to cancel the effect of ONERR GOTO.

Sample Program

```
10 ONERR GOTO 50
20 INPUT "Type your name and press RETURN.>";n$
30 CLRERR
40 INPUT "Type your age and press RETURN.>";a
50 PRINT "You made a mistake. Try again."
60 GOTO 20
```

Run

Because of the ONERR GOTO statement in line 10, if the operator types a number in line 20, the program will skip to line 50 to indicate there has been a mistake. No other error message will be displayed. However, if the operator types a word in line 40, an error message will be displayed because of the CLRERR statement in line 30.

COLOR=

COLOR= is a statement that sets the color of the lines to be drawn in the low resolution graphics mode. (To put ADAM into the low resolution graphics mode, the statement GR must come before COLOR=.) The 16 available colors and their corresponding numbers are listed below.

COLOR	NUMBER
Black	0
Magenta	1
Dark Blue	2
Dark Red	3
Dark Green	4
Grey-1	5
Medium Green	6
Light Blue	7
Light Yellow	8
Medium Red	9
Grey-2	10
Light Red	11
Light Green	12
Light Yellow	13
Cyan (blue)	14
White	15

Sample Program

```

10 GR
20 COLOR=6
30 VLIN 5,25 AT 20

```

Run

This program draws a green vertical line between rows 5 and 25 in column 20.

COS (advanced**)**

COS is a function that returns the cosine of an angle measured in radians. COS is used only with right triangles and must always be followed by a number within parentheses.

CONT

CONT is a command used to continue a program that has been stopped by a STOP or END statement. CONT will also restart a program that you have stopped by holding down CONTROL and pressing the C key. There is an important difference between CONT and RUN. CONT starts a program from where it stopped. RUN, on the other hand, starts a program from the beginning. To use the CONT command, complete the following steps:

1. Type CONT.
2. Press RETURN.

DATA

DATA is a statement used to list information that will be used by a READ statement. Any program that contains a DATA statement must also contain a READ statement. Information listed in a DATA statement should be separated by commas. String information that contains commas, spaces, or colons should be surrounded by quotation marks. NOTE: To avoid getting an "OUT OF DATA" error message, be sure to use a "marker" at the end of each DATA statement. For example, in the program below, the word ZERO indicates the end of the DATA statement. The program, in line 30, tests to see if N\$ is equal to ZERO. If it is, lines 40 and 50 are skipped and the program ends.

Sample Program

```
10 DATA one, two, three, four, five, zero
20 READ n$
30 IF n$="zero" THEN GOTO 60
40 PRINT n$
50 GOTO 20
60 PRINT "This has been a demonstration."
```

Run

```
one
two
three
four
five
This has been a demonstration.
```

DEF FN

DEF FN is a statement that gives you the ability to define a calculation as a function. Then, instead of writing out the same formula again and again in a program, you can just type the function name. If you have a long calculation or formula that is used repeatedly throughout a program, the DEF FN can save you time and space.

When using the DEF FN statement, you must first name and define the function as illustrated in line 30 in the Sample Program. (You will find it helpful if the function name contains variable letters that will help you remember what the function does.) Then, later, when you want to use the function you have defined, just type the function name—which is FN followed by the variable letters you selected. You don't have to type the formula or calculations again. Once you have defined a function ADAM does all the work!

Sample Program

```
10 PRINT "The function FNI(Y) changes inches"
20 PRINT "to yards."
30 LET I=72
40 DEF FNI(Y)=I/36
45 PRINT:PRINT
50 PRINT "For example, 72 inches = ";FNI(Y);" yards."
```

Run

```
The function FN(Y) changes inches
to yards.
```

```
For example, 72 inches = 2 yards.
```

DEL

DEL is a command used to delete individual program lines or groups of program lines. To delete one or more program lines:

1. Make sure the program containing the lines to be deleted is displayed on your screen.
2. Type DEL.
3. Press SPACEBAR.
4. To indicate the line(s) to be deleted:
 - To delete a single line, type the line number (for example, 25).
 - To delete a group of lines, type the first line number, a comma (,), and the last line number (for example, 25,40).
5. Press RETURN.

DELETE

DELETE is a command used to delete or “erase” programs stored on a digital data pack. To delete a program:

1. Make sure the digital data pack containing the program to be deleted is inserted in the digital data pack drive.
2. Type DELETE.
3. Press SPACEBAR.
4. Type the name of the program to be deleted.
5. Press RETURN.

DIM

DIM is a statement used to establish the number of elements in an array. For this reason, DIM is helpful in creating charts. In SmartBASIC you can have a one-, two-, or three-dimensional array. In addition, the elements in an array can be numbers or strings.

Sample Program—#1

```

10 PRINT "This is an example of a one-"
20 PRINT "dimensional numeric array."
30 PRINT:PRINT
40 DIM a(16)
50 FOR n=1 TO 16
60 LET a(n)=n
70 PRINT a(n),
80 NEXT n

```

Run—#1

This is an example of a one-dimensional numeric array.

1	2
3	4
5	6
7	8
9	10
11	12
13	14
15	16

Sample Program—#2

```

10 PRINT "This is an example of a one-"
20 PRINT "dimensional string array."
30 PRINT:PRINT
40 DIM a$(6)
50 LET a$(1)="Washington"
60 LET a$(2)="Jefferson"
70 LET a$(3)="Adams"
80 LET a$(4)="Lincoln"
90 LET a$(5)="Roosevelt"
100 LET a$(6)="Truman"
110 FOR b = 1 TO 6
120 PRINT a$(b),
130 NEXT b

```


Run—#2

```
"This is an example of a one-  
dimensional string array."
```

```
Washington      Jefferson  
Adams           Lincoln  
Roosevelt      Truman
```

DRAW (**advanced**)

DRAW is a statement used in high resolution graphics to draw a shape (that has previously been loaded into your computer's memory). DRAW must be followed by a number to indicate what shape you want to draw. Then, to indicate where the shape should be drawn, type: AT, the column location, a comma (,) and the row location.

END

END is a statement used to end program execution. Unlike STOP, when a program encounters an END statement it does not display a "BREAK IN" message.

Sample Program

```
10 PRINT "This is a demonstration"  
20 END  
30 PRINT "of the END statement."
```

Run

```
This is a demonstration
```

EXP (**advanced**)

EXP is a function that raises numbers to the indicated power. EXP should always be followed by a number that appears within parentheses, for example EXP(X).

FOR/NEXT (STEP)

FOR and NEXT statements are used to create what are known as "FOR/NEXT" loops. FOR/NEXT loops control how many times certain lines of a program are repeated. As illustrated below in Sample Program #2, STEP may be used in the first line of a FOR/NEXT loop. The number following STEP indicates how much the value of the variable in the FOR/NEXT loop should be increased. (NOTE: If STEP is not present in a FOR/NEXT loop, STEP 1 is assumed.)

Sample Program—#1

```
10 FOR a=1 TO 6
20 PRINT "NOW a = ";a
30 NEXT a
```

Run—#1

```
NOW a = 1
NOW a = 2
NOW a = 3
NOW a = 4
NOW a = 5
NOW a = 6
```

Sample Program—#2

```
10 FOR a=1 TO 6 STEP 2
20 PRINT "NOW a = ";a
30 NEXT a
```

Run—#2

```
NOW a = 1
NOW a = 3
NOW a = 5
```

FRE (**advanced**)

FRE is a function that returns the number of unused bytes available in your computer's memory. FRE should always be followed by a variable within parentheses (e.g., FRE(X)). Then, when the program is run, the variable will be replaced by a number indicating the number of bytes available.

GET

GET is a statement used to transfer a single character from the user to the program. GET causes a program to pause until one key is pressed. Like INPUT, GET allows the user to assign a value to a variable. Unlike INPUT, GET does not require the user to press RETURN. As soon as one key is pressed, the program continues.

Sample Program

```
10 PRINT "To begin this program, type: b"  
20 GET a$  
30 IF a$<>"b" THEN GOTO 20  
40 PRINT "Thank you. This has been a demonstration."
```

Run

```
To begin this program, type: b  
b  
Thank you. This has been a demonstration.
```

GOSUB

GOSUB is a statement used to call a subroutine. GOSUB must be followed by a line number indicating the first line in the subroutine. The last line in the subroutine must be a RETURN statement. When RETURN is encountered, the program will go back to the line following the GOSUB statement.

Sample Program

```
10 PRINT "This is line 10."  
20 GOSUB 50  
30 PRINT "This is line 30."  
40 END  
50 PRINT "This is a subroutine."  
60 RETURN
```

Run

```
This is line 10.  
This is a subroutine.  
This is line 30.
```

GOTO

GOTO is a statement used to direct your computer to a specific line within a program. When ADAM reaches a GOTO statement in a program, it doesn't automatically advance to the next line in the program. Instead, it moves to the line number indicated after the GOTO statement.

Sample Program

```
10 PRINT "This is line 10."  
20 GOTO 40  
30 PRINT "This is line 30."  
40 PRINT "This is line 40."
```

Run

```
This is line 10.  
This is line 40.
```

GR

GR is a statement or command used to send your computer into the low resolution graphics mode. In the low resolution graphics mode, your screen is divided into 40 columns (0-39) and 40 rows (0-39). On the bottom of the screen are four lines reserved for text. NOTE: To return to the regular or "narrative" screen after using low resolution graphics, you must use the keyword TEXT as a command or statement.

Sample Program

```
10 GR  
20 COLOR=7  
30 VLIN 5,30 AT 10
```

Run

This program draws a light blue vertical line between rows 5 and 30 in column 10.

HCOLOR=

HCOLOR= is a statement that sets the color of points to be plotted in the high resolution graphics mode. (To put ADAM into the high resolution graphics mode, the statement HGR or HGR2 must come before HCOLOR=.) The 16 available colors and their corresponding numbers are listed below.

COLOR	NUMBER
Black 1	0
Green	1
Violet	2
White 1	3
Black 2	4
Orange	5
Blue	6
White 2	7
Brown	8
Dark Blue	9
Grey	10
Pink	11
Dark Green	12
Yellow	13
Aqua	14
Magenta	15

Sample Program

```
10 HGR
20 HCOLOR=14
30 HPLLOT 0, 159 TO 19, 159
```

Run

This program draws an aqua line at the bottom of the screen.

HGR

HGR is a statement or command used to send your computer into the high resolution graphics mode. In the high resolution graphics mode, your screen is divided into a grid measuring 256 columns (0-255) by 160 rows (0-159). (Points on this grid are then plotted using the statement HPLOT.) On the bottom of the screen are four lines reserved for text. NOTE: To return to the regular or "narrative" screen after using high resolution graphics, you must use the keyword TEXT as a command or statement.

Sample Program

```
10 HGR
20 HCOLOR=2
30 HPLOT 0, 159
```

Run

This program plots a violet point at the bottom left-hand corner of the screen. Notice that four lines remain at the bottom of the screen.

HGR2

HGR2 is a statement or command used to send your computer into the high resolution graphics mode. The difference between HGR and HGR2 is that HGR2 does not leave four lines reserved for text at the bottom of the screen. Therefore, the grid established with the statement HGR2 is 256 columns (0-255) by 192 rows (0-191). NOTE: To return to the regular or "narrative" screen after using high resolution graphics, you must use the keyword TEXT as a command or statement.

Sample Program

```
10 HGR2
20 HCOLOR=1
30 HPLOT 0, 191
```

Run

This program plots a green point at the bottom left-hand corner of the screen. Notice that you can plot points at the bottom of the screen using the HGR2 statement.

HIMEM: (**advanced**)

HIMEM: is a statement used to change the highest location in your computer's memory. HIMEM: is pre-set to allow you the maximum amount of space available and it is suggested that you reset HIMEM: only when working in assembly language. NOTE: HIMEM:, when used, should be followed by a number indicating the appropriate memory location.

HLIN

HLIN is a statement used to draw horizontal lines between columns in the low resolution graphics mode.

Sample Program

```
10 GR
20 COLOR=2
30 HLIN 0,19 AT 15
```

Run

This program draws a dark blue line between columns 0 and 19 in row 15.

HOME

HOME is a statement or command that erases the screen and sends the cursor into the upper left-hand corner. If you're using low resolution graphics and use a HOME statement, the text at the bottom of the screen is cleared and the cursor returns to the beginning of line 21.

Sample Program

```
10 FOR a=1 TO 100
20 PRINT a;
30 NEXT a
40 HOME
```

Run

This program will display the numbers 1-100. After displaying the number 100, it will erase the screen and return the cursor to the upper left-hand corner.

H PLOT

H PLOT is a statement used to plot points and draw lines in the high resolution graphics mode. When using H PLOT you must first indicate the column number (0 to 255) of the point you want to plot. Then you must indicate the row number (0 to 159 if you have used HGR, 0 to 191 if you have used HGR2). To draw lines instead of plotting individual points, you can indicate more than one set of points to be plotted.

Sample Program

```
10 HGR
20 HCOLOR=5
30 H PLOT 0, 0 TO 0, 159
```

Run

This program draws an orange line at the left edge of the screen.

H TAB

H TAB is a statement that moves the cursor in a horizontal line—without changing or moving existing text. H TAB must be followed by a column number (0-39).

Sample Program

```
10 H TAB 19
20 H TAB 0
30 H TAB 19
40 H TAB 0
```

Run

This program will move the cursor to the middle of the screen and back to the left margin twice.

IF/GOTO

IF/GOTO is a statement used to analyze a situation and then direct the computer to a different line in the program based on that analysis. (NOTE: See the beginning of this chapter for a list of operators that can be used in IF/GOTO statements.)

Sample Program—#1

```
10 PRINT "If you want to see lines 80-100,"
20 PRINT "type YES. Then, press RETURN."
30 PRINT
40 PRINT "If you want to skip to line 110,"
50 PRINT "type NO. Then, press RETURN."
60 INPUT p$
70 IF p$="NO" GOTO 110
80 PRINT "This is line 80."
90 PRINT "This is line 90."
100 PRINT "This is line 100."
110 PRINT "This is line 110."
```

Run—#1

If you want to see lines 80-100,
Type YES. Then, press RETURN.

If you want to skip to line 110,
Type NO. Then, press RETURN.
YES
This is line 80.
This is line 90.
This is line 100.
This is line 110.

Run—#2

If you want to see lines 80-100,
Type YES. Then, press RETURN.
If you want to skip to line 110,
Type NO. Then, press RETURN.
NO
This is line 110.

IF/THEN

IF/THEN is a statement used to analyze a situation and then decide on a course of action based on that analysis. For example, depending on the operator's response to an INPUT statement, a program might display different types of information. (NOTE: See the beginning of this chapter for a list of operators that can be used in IF/THEN statements.)

Sample Program—#1

```
10 PRINT "To begin, please type your age"  
20 PRINT "(in years) and press RETURN."  
30 INPUT a  
40 PRINT:PRINT  
50 IF a>=50 THEN PRINT "Ah...the wisdom of age."  
60 IF a<50 THEN PRINT "Ah...the energy of youth."
```

Run—#1

```
To begin, please type your age  
(in years) and press RETURN.  
50
```

```
Ah...the wisdom of age.
```

Run—#2

```
To begin, please type your age  
(in years) and press RETURN.  
49
```

```
Ah...the energy of youth.
```

INIT (**advanced**)

INIT is a command used to delete every program stored on a digital data pack. To use INIT, place the data pack to be erased in a drive. Then type INIT, press SPACEBAR and type the desired volume name. (Optionally, to indicate a drive number, you can then type a comma (,) the letter D, and the drive number (1 or 2).) Finally, press RETURN. NOTE: You do not need to use this command on blank digital data packs.

INPUT

INPUT is a statement that allows the user to assign values to variables. When the the computer sees the word INPUT, it waits for the user to type a response and press RETURN. (NOTE: In SmartBASIC, INPUT can be used to display information as illustrated in *Sample Program—#2*.)

Sample Program—#1

```
10 PRINT "Type your name and press RETURN."  
20 INPUT n$  
30 PRINT  
40 PRINT "Thank you ";n$;"."
```

Run—#1

```
Type your name and press RETURN.  
Becky  
  
Thank you Becky.
```

Sample Program—#2

```
10 INPUT "Type your name and press RETURN. ";n$  
20 PRINT  
30 PRINT "Thank you ";n$;"."
```

Run—#2

```
Type your name and press RETURN. Becky  
  
Thank you Becky.
```

INT

INT is a function used to round numbers to their integer (or whole) value. The result of the INT function used with any number is always less than or equal to the number. In other words, if INT is used with a positive number, the result is the positive number without any decimal places. If INT is used with a negative number, the result is the next negative number (without decimal places) less than the original negative number.

Sample Program

```
10 PRINT "The integer value of 37.653 IS  
";INT(37.653);"."  
20 PRINT "The integer value of -4.56 IS ";INT(-  
4.56);"."
```

Run

```
The integer value of 37.653 IS 37.  
The integer value of -4.56 IS -5.
```

INVERSE

INVERSE is a statement or command used to change the appearance of your screen. Normally, your screen displays white letters on a black background. If you use INVERSE, your screen will display black letters on a white background. NOTE: To return to regular text, use the keyword NORMAL either in command or as a statement.

Sample Program

```
10 PRINT "This shows white letters on a black"  
20 PRINT "background."  
30 INVERSE  
40 PRINT "This shows black letters on a white"  
50 PRINT "background."  
60 NORMAL  
70 PRINT "Now everything is back to normal."
```

Run

```
This shows white letters on a black  
background.  
This shows black letters on a white  
background.  
Now everything is back to normal.
```

LEFT\$

LEFT\$ is a function that returns the specified number of characters in a string, beginning with the first character on the left.

Sample Program

```
10 LET a$="Hello. Good-bye. I'm back again."  
20 PRINT a$  
30 PRINT:PRINT  
40 PRINT LEFT$(a$,6)
```

Run

```
Hello. Good-bye. I'm back again.
```

```
Hello.
```

LEN

LEN is a function that counts the number of characters (including spaces and other symbols) in a string.

Sample Program

```
10 LET a$="Hello. My name is ADAM."  
20 PRINT a$  
30 PRINT:PRINT  
40 PRINT "In the sentence above there"  
50 PRINT "are ";LEN(a$);" characters."
```

Run

```
Hello. My name is ADAM.
```

```
In the sentence above there  
are 24 characters.
```

LET

LET is a statement that allows you to assign values to numeric or string variables. The use of LET in SmartBASIC is optional. So, for example, instead of saying LET a=25, you can say a=25.

Sample Program

```
10 LET a=25
20 LET b$="HELLO"
30 PRINT a
40 PRINT b$
```

Run

```
25
HELLO
```

LIST

LIST is a command used to display the lines of a program on the screen. LIST can be used to display a single line, a group of lines, or all the lines in a program. To temporarily stop a listing program, you can hold down the CONTROL key and press S. Then hold down CONTROL and press S again to continue listing the program. To use the LIST command, follow the steps below.

1. Type LIST.
2. ● To list a single line, type the line number (for example, 25).
● To list a group of lines, type the first line number, a dash (-) or comma (,) and the last line number (for example, 25-40 or 25,40).
3. Press RETURN.

LOAD

LOAD is a command that allows you to recall a program currently stored on a digital data pack. To use the LOAD command, you must follow five steps.

1. Remove the SmartBASIC data pack from the drive.
2. Insert the digital data pack that contains the program you want to load into the digital data pack drive.
3. Type LOAD.
4. Press SPACEBAR.
5. Type the name of the program you want to load.
6. Press RETURN.
7. Remove the data pack from the drive.
8. Re-insert the SmartBASIC data pack in the drive.

LOCK

LOCK is a command that allows you to protect files (that are stored on a digital data pack) from being accidentally deleted or changed. To use the LOCK command, you must follow six steps.

1. If the file to be locked is not already stored on a digital data pack, complete the SAVE procedure.
2. Make sure the data pack containing the file to be locked is in the digital data pack drive.
3. Type LOCK.
4. Press SPACEBAR.
5. Type the name of the program to be locked.
6. Press RETURN.

LOG (**advanced**)

LOG is a function that finds the natural logarithm of any number larger than zero (0). LOG must always be followed by a number in parentheses, for example LOG (X).

LOMEM: (**advanced**)

LOMEM: is a statement used to change the lowest location in your computer's memory. LOMEM: is pre-set to allow you the maximum amount of space available and it is suggested that you reset LOMEM: only when working in assembly language. NOTE: LOMEM:, when used, should be followed by a number indicating the appropriate memory location.

MID\$

MID\$ is a function that returns the specified number of characters in a string. The first number after the variable indicates the position of the first character to be returned. The second number after the variable indicates the number of characters to be returned.

Sample Program

```
10 LET a$="Hello. Good-bye. I'm back again."  
20 PRINT MID$(a$,20,8)
```

Run

```
I'm back
```

MON (**advanced**)

MON is a command used to monitor information as it moves between a digital data pack and your computer's memory. MON must be followed by a space and then one or more of the following letters (separated by commas): C (monitors commands to the data pack), I (displays input from the data pack), O (displays output from the data pack), or L (displays each line as it is loaded from a data pack). For further information, see **NOMON**.

NEW

NEW is a command that deletes the current program from your computer's memory. To use the NEW command you must complete two steps.

1. Type NEW.
2. Press RETURN.

NOMON (**advanced**)

NOMON is a command used to cancel the MON command. NOMON should be followed by a space and then one or more of the following letters (separated by commas): C, I, O, or L. For more information, see **MON**.

NORMAL

NORMAL is a statement or command used to cancel the effect of the INVERSE statement.

Sample Program

```
10 PRINT "This shows white letters on a black"
20 PRINT "background."
30 INVERSE
40 PRINT "This shows black letters on a white"
50 PRINT "background."
60 NORMAL
70 PRINT "Now everything is back to normal."
```

Run

```
This shows white letters on a black
background.
This shows black letters on a white
background.
Now everything is back to normal.
```

NOTRACE

NOTRACE is a statement or command used to cancel the effect of the TRACE statement.

Sample Program

```
10 TRACE
20 PRINT "Because of TRACE, you can see"
30 PRINT "each line number as it is"
40 PRINT "executed."
50 NOTRACE
60 PRINT "Now, because of NOTRACE, you can't"
70 PRINT "see line numbers."
```

Run

```
Because of TRACE, you can see
each line number as it is
executed.
Now, because of NOTRACE, you can't
see line numbers.
```

ONERR GOTO

ONERR GOTO is a statement used to override the usual way your computer handles errors. ONERR GOTO must be followed by a line number. Then, when an error occurs, the program will skip to the line indicated instead of displaying an error message. To cancel the effect of ONERR GOTO, use the statement CLRERR.

Sample Program

```
10 ONERR GOTO 50
20 INPUT "Type your name and press RETURN. ";n$
30 CLRERR
40 INPUT "Type your age and press RETURN. ";a
50 PRINT "You made a mistake. Try again."
60 RESUME
```

Run

Because of the ONERR GOTO statement in line 10, if the operator types a number in line 20, the program will skip to line 50 to indicate that there has been a mistake. Then, because of the RESUME statement in line 60, the program will return to line 20 and no other error message will be displayed. However, if the operator types a word in line 40, an error message will be displayed because of the CLRERR statement in line 30.

ON GOSUB

ON GOSUB is a statement used to evaluate variables or expressions. Depending on the value of the variable (or result of the expression) included in the ON GOSUB statement, your computer knows what subroutine to call.

Sample Program

```
10 INPUT "Type 1 or 2 and press RETURN. ";n
20 ON n GOSUB 30, 40
25 END
30 PRINT "This is line 30 and you typed 1.":RETURN
40 PRINT "This is line 40 and you typed 2.":RETURN
```

Run

```
Type 1 or 2 and press RETURN. 1
This is line 30 and you typed 1.
```

ON GOTO

ON GOTO, like ON GOSUB, is a statement used to evaluate variables or expressions. Depending on the value of the variable (or result of the expression) included in the ON GOTO statement, your computer goes to the indicated line.

Sample Program

```
10 INPUT "Type 1 or 2 and press RETURN. ";n
20 ON n GOTO 30, 40
30 PRINT "This is line 30 and you typed 1."
35 END
40 PRINT "This is line 40 and you typed 2."
```

Run

```
Type 1 or 2 and press RETURN.1
This is line 30 and you typed 1.
```

OPEN

OPEN is a command used to gain access to a file. To use the OPEN command, complete the following steps. (NOTE: In the lines below, PROGRAM stands for the name of your program.)

1. Type the following line to begin your program:

```
10 d$=CHR$(4)
```

2. Type the following line as the next line of your program:

```
20 PRINT d$;"OPEN PROGRAM"
```

3. Type the information to be added to or deleted from the file.

4. Type the following line to end your program and close your file:

```
100 PRINT d$;"CLOSE PROGRAM"
```

PDL (**advanced**)

PDL is a function used to control the position of the cursor in game play and when other graphics are displayed on your screen. Normally, there are two paddles (or "game controllers") attached to your computer. You may attach as many as four paddles (PDL(0), PDL(1), PDL(2), and PDL(3)). Paddle positions are numbered left to right (0-255) and top to bottom (0-255).

PEEK (**advanced**)

PEEK is a statement used to examine information stored in a specific memory location. PEEK should always be followed by a space and then an address (0-65535) indicating the memory location to be read.

PLOT

PLOT is a statement, used in low resolution graphics, to color in the intersection of columns (0-39) and rows (0-39).

Sample Program

```
10 GR
20 COLOR=4
30 PLOT 5,7
```

Run

This program colors the intersection of column 5 and row 7 dark green.

POKE (advanced**)**

POKE is a statement used to change the contents of a memory location in RAM. POKE should always be followed by a space, the address of the memory location to be changed (-32767-65535), a comma (,) and the value (0-255) to be stored. NOTE: When you first load SmartBASIC, you can not use addresses higher than 54160.

POP

POP is a statement that is used in a subroutine before the RETURN statement. When your computer encounters a POP statement, it doesn't return to the line after the GOSUB statement that called the subroutine. Instead, it "pops" back to the line following the previous GOSUB statement.

Sample Program

```
10 PRINT "This is line 10."  
20 GOSUB 30  
24 PRINT "This is line 24."  
27 END  
30 PRINT "This is line 30."  
40 GOSUB 60  
45 PRINT "This is line 45."  
50 RETURN  
60 PRINT "This is line 60."  
70 POP  
80 RETURN
```

Run

```
This is line 10.  
This is line 30.  
This is line 60.  
This is line 24.
```

POS

POS is a function that reads the current column (0-39) position of the cursor. POS, when it is used in a program, must always be followed by an open parenthesis [()], the number zero (0), and a closed parenthesis [)]. Then, when the program is run, the zero will be replaced with the correct column number.

Sample Program

```
10 PRINT "The cursor position is ";POS(0)  
20 PRINT "Now the cursor position is ";POS(0)
```

Run

```
The cursor position is 24  
Now the cursor position is 28
```

POSITION (advanced**)**

POSITION is a command used to move the file pointer forward in a sequential file. POSITION must be followed by a space, the name of the file, a comma (,), the letter R, and a number indicating how far (in records) you want the file pointer to move. NOTE: To use POSITION, you must first open a file and then later close the file.

PR #

PR# is a command used to transfer SmartBASIC information to your printer. PR#0 indicates your television screen or monitor and PR#1 indicates your television screen. To print a copy of a program, complete the following steps:

1. Type the following commands:

```
PR#1:LIST:PR#0
```

2. Press RETURN.

NOTE: Any other devices added to your computer will be numbered between 2 and 254.

PRINT

PRINT is a statement used to display text. PRINT can also be used to display variable values and create blank lines. In SmartBASIC, the statement INPUT can also be used to display text when the user will be typing a response.

Sample Program

```
10 PRINT "This is a PRINT demonstration."  
20 INPUT "Type your name and press RETURN. ";n$  
30 PRINT  
40 PRINT "Thank you ";n$". "  
50 PRINT  
60 PRINT "I appreciate your help."
```

Run

```
This is a PRINT demonstration  
Type your name and press RETURN. Susan
```

```
Thank you Susan.
```

```
I appreciate your help.
```

READ

READ is a statement used to assign information from DATA statements to variables. READ must be followed by a string (n\$) or numeric (a) variable. For more information, see **DATA**.

RECOVER

RECOVER is a command used to recover back-up copies of SmartBASIC files. (When you use CATALOG to see files stored on a digital data pack, the current version of a file is indicated by an "A", and the back-up version of a file is indicated by an "a.") To access a back-up file using RECOVER, complete the following steps:

1. Remove the SmartBASIC data pack from the drive.
2. Insert the data pack containing the file to be recovered in the drive.
3. To see a list of the files stored on the digital data pack:
 - a. Type CATALOG.
 - b. Press RETURN.
4. To recover a back-up file:
 - a. Type RECOVER.
 - b. Press SPACEBAR.
 - c. Type the file name.
 - d. Press RETURN.
5. Load the file using the LOAD procedure.
6. Remove the data pack from the drive.
7. Insert the SmartBASIC digital data pack in the drive.

REM

REM is a statement used to store information within a program that will not be seen or used when the program is run. REM stands for REM(ark) and any information following REM will be ignored by ADAM.

Sample Program

```
10 PRINT "Hello"  
20 REM THIS IS A SECRET MESSAGE  
30 PRINT "Good-bye"
```

Run

```
Hello  
Good-bye
```

RENAME

RENAME is a command used to change the name of a file that has been stored on a digital data pack. To use the RENAME command, complete the following steps:

1. Remove the SmartBASIC data pack from the drive.
2. Insert the digital data pack that contains the program you want to rename into the digital data pack drive.
3. Type RENAME.
4. Press SPACEBAR.
5. Type the current name of the file you want to rename.
6. Type , (comma).
7. Type the new name of the file.
8. Press RETURN.
9. Remove the digital data pack from the drive.
10. Insert the SmartBASIC data pack in the drive.

RESTORE

RESTORE is a statement used with READ and DATA. RESTORE resets the DATA pointer so that a DATA statement can be used again. This process also avoids the OUT OF DATA error message.

Sample Program

```
10 DATA 1, 2, 3, 4
20 READ A, B, C, D
30 PRINT A, B, C, D,
40 RESTORE
50 READ E, F, G, H
60 PRINT E, F, G, H
```

Run

```
1          2
3          4
1          2
3          4
```

RESUME

RESUME is a statement used with ONERR GOTO. Placed at the end of the error routine, it returns the program to the line following the error.

Sample Program

```
10 ONERR GOTO 50
20 INPUT "Type a number and press RETURN. "; n
30 PRINT n; " X 6 = "; 6*n
40 END
50 PRINT "You made a mistake."
60 PRINT "I will select 50 for you."
70 LET n=50
80 RESUME
```

Run

Because of the ONERR GOTO statement in line 10, if the operator types letter(s) or word(s) in line 20, the program will skip to line 50 to indicate that there has been a mistake. Then the program selects a number (50) for the operator and assigns this value to the variable n. Then, because of the RESUME statement in line 60, the program will return to line 30.

RETURN

RETURN is a statement that should always be used at the end of a subroutine. It instructs your computer to go back to the statement after the GOSUB statement that called the subroutine. RETURN must never be used without a corresponding GOSUB statement.

Sample Program

```
10 PRINT "This is line 10."  
20 GOSUB 50  
30 PRINT "This is line 30."  
40 END  
50 PRINT "This is a subroutine."  
60 RETURN
```

Run

```
This is line 10.  
This is a subroutine.  
This is line 30.
```

RIGHT\$

RIGHT\$ is a function that returns the specified number of characters in a string, beginning with the first character on the right.

Sample Program

```
10 LET a$="Hello. Good-bye. I'm back again."  
20 PRINT a$  
30 PRINT:PRINT  
40 PRINT LEFT$(a$,15)
```

Run

```
Hello. Good-bye. I'm back again.  
  
I'm back again.
```

RND

RND is a function that returns a random number between 1 and 0 (zero). RND should always be followed by a number or numeric expression within parentheses, e.g., RND(2). Different sequences of random numbers may be generated by using a "negative seed" instead of this number (for more information see Part 5, Chapter 1). The result of RND(X) can be operated on by using addition, subtraction, multiplication, and division. As illustrated below, INT can then be used to find the integer of this number.

Sample Program

```
10 PRINT "This is a number between 1-5 ";INT(RND(1)*5)
20 PRINT "This is a number between 1-10:
";INT(RND(1)*10)
30 PRINT "This is a number between 1-25:
";INT(RND)(1)*25)
40 PRINT "This is a number between 1-100:
";INT(RND)(1)*100)
```

Run

```
This is a number between 1-5: 3
This is a number between 1-10: 4
This is a number between 1-25: 2
This is a number between 1-100: 70
```

ROT= (**advanced**)

ROT= is a statement that establishes the spatial relationship of a shape drawn in high resolution graphics. ROT= must be followed by a number between 1 and 63, with each number representing 5.625 degrees of rotation.

RUN

RUN is a command that executes the program currently stored in the computer's memory. To use the RUN command, you must follow two steps.

1. Type RUN.
2. Press RETURN.

SAVE

SAVE is a command that allows you to store the program currently in your computer's memory. To use the SAVE command, complete the following steps:

1. Remove the SmartBASIC data pack from the drive.
2. Insert a digital data pack in the digital data pack drive. (NOTE: Be sure this is a data pack that has enough blank space to store your program.)
3. Type SAVE.
4. Press SPACEBAR.
5. Type the name you want to give to your program. (This name can be up to 10 characters long.)
6. Press RETURN.
7. If desired, remove the digital data pack from the drive.
8. If desired, insert the SmartBASIC data pack in the drive.

SCALE= (**advanced**)

SCALE= is a statement used to set the relative size of shapes drawn in high resolution graphics. SCALE= must be followed by a number between 1 and 255. SCALE=1 indicates an exact reproduction of the shape; SCALE=2 indicates a shape drawn twice the size of the one specified; SCALE=3 indicates a shape drawn three times the size of the one specified, etc.

SCRN

SCRN is a function used to analyze colors on the graphics screen.

Sample Program

```
10 GR
20 COLOR=14
30 PLOT 2,5
40 IF SCRN(2,5)=14 THEN PRINT "There is
a blue spot on your screen."
50 IF SCRN(2,5)<>14 THEN PRINT "Whoops!
I made a mistake!"
```

Run

(A blue spot will appear in column 2, row 5.)

There is a blue spot on your screen.

SGN

SGN is a function that indicates whether the value of a number is positive, zero, or negative.

Sample Program

```
10 PRINT SGN(3)
20 PRINT SGN(0)
30 PRINT SGN(-3)
```

Run

```
1
0
-1
```

SIN (**advanced**)

SIN is a function that returns the sine of an angle measured in radians. SIN is used only with right triangles and must always be followed by a number within parentheses.

SPC

SPC is a function used to insert spaces between displayed text. SPC must be followed by a number between 0 and 39 and is particularly useful when creating right-justified text.

Sample Program

```
10 PRINT SPC(8);"KING"  
20 PRINT SPC(7);"LLOYD"  
30 PRINT SPC(1);"NAVRATILOVA"
```

Run

```
      KING  
     LLOYD  
NAVRATILOVA
```

SPEED=

SPEED= is a statement that allows you to control how fast characters are sent to your screen or printer. When SPEED= is used it must be followed by a number between 0 (the slowest) and 255 (the fastest). Normally your computer runs at 255—the highest possible speed.

Sample Program

```
10 SPEED=0  
20 PRINT "Slow"  
30 SPEED=100  
40 PRINT "Faster"  
50 SPEED=200  
60 PRINT "Even faster"  
70 SPEED=255  
80 PRINT "The fastest"
```

Run

```
Slow  
Faster  
Even faster  
The fastest
```


SQR

SQR is a function that calculates the square root of any positive number.

Sample Program

```
10 PRINT "The square root of 9 is ";
20 PRINT SQR(9)
```

Run

```
The square root of 9 is 3
```

STOP

STOP is a statement used to end the running of a program. When a program encounters STOP it displays a BREAK IN X message where X is the line containing the STOP statement. After the program stops, you have two choices. You can either use the CONT command to start the program at the line following the STOP statement, or you can use RUN to start the program at the beginning.

Sample Program

```
10 PRINT "This is the first line."
20 PRINT "This is the second line."
30 STOP
40 PRINT "This is the fourth line."
50 PRINT "This is the fifth line."
```

Run

```
This is the first line.
This is the second line.
```

```
BREAK IN 30
```

STR\$

STR\$ is a function that changes numbers into strings. The advantage of using this function is that you can then use functions (MID\$, LEFT\$, etc.) to manipulate strings.

Sample Program

```
10 LET n=378
20 PRINT STR$(n)
30 PRINT
40 PRINT LEFT$(STR$(n), 2)
```

Run

```
378
37
```

TAB

TAB is a function that indicates in what column (0-39) information should be displayed. TAB, in other words, must always be used with PRINT. NOTE: To display information at the column indicated, you must follow the TAB function with a semicolon.

Sample Program

```
10 PRINT TAB(5);"Using the";
20 PRINT TAB(20);"TAB function"
30 PRINT TAB(5);"you can";
40 PRINT TAB(20);"create charts."
```

Run

```
Using the      TAB function
you can       create charts.
```

TAN (**advanced**)

TAN is a function that returns the tangent of an angle measured in radians. TAN must always be followed by a number within parentheses.

TEXT

TEXT is a statement or command used to return to a narrative or regular screen after the use of low or high resolution graphics. When your computer encounters a TEXT statement, it automatically clears the screen and positions the cursor in the upper left-hand corner.

Sample Program

```
10 GR
20 COLOR=7
30 VLIN 5,15 AT 10
40 PRINT "This is a graphics screen."
50 PRINT "To return to a text or narrative"
60 PRINT "screen, type: t"
70 GET a$
80 IF a$<>"t" THEN GOTO 70
90 TEXT
```

Run

This program draws a light blue vertical line between rows 5 and 15 in column 10. It then displays the text in lines 40-60. Then, if the operator presses the T key, the screen returns to the text mode. If the operator does not press the T key, an orange line continues to be displayed.

TRACE

TRACE is a statement or command used to debug programs. When TRACE is included in a program, the computer displays the number of each line as it is executed.

Sample Program

```
10 TRACE
20 PRINT "This line is traced."
30 PRINT "This line is traced too."
40 NOTRACE
50 PRINT "This line is not traced."
```

Run

```
This line is traced.
This line is traced too.
This line is not traced.
```

UNLOCK

UNLOCK is a command that allows you to remove the LOCK command from files stored on a digital data pack. To use the UNLOCK command, you must follow five steps.

1. Make sure the data pack containing the file to be unlocked is in the digital data pack drive.
2. Type UNLOCK.
3. Press SPACEBAR.
4. Type the name of the program to be unlocked.
5. Press RETURN.

USR (**advanced**)

USR is a function used to call a machine language routine that has been stored using POKE.

VAL

VAL is a function that takes numbers that have been written as a string and changes them back to numbers.

Sample Program

```
10 LET n$="100, 98, 85, 97"  
20 PRINT VAL (n$)
```

Run

```
100, 98, 85, 97
```

VLIN

VLIN is a statement used to draw vertical lines in the low resolution graphics mode. As illustrated below, VLIN must be followed by three numbers. The first number indicates the row where the line begins. The second number indicates the row where the line ends. The third and final number indicates the column in which the line is drawn.

Sample Program

```
10 GR
20 COLOR=12
30 VLIN 5,20 AT 10
```

Run

This program draws a vertical line between rows 5 and 20 in column 10.

VTAB

VTAB is a statement that indicates on what line (0-39) text should be displayed.

Sample Program

```
10 PRINT VTAB(2);"This is line 2."
20 PRINT VTAB(4);"This is line 4."
```

Run

```
This is line 2.
This is line 4.
```

WAIT (advanced**)**

WAIT is a statement or command used to halt a program until a specified bit pattern appears at a designated port location.

WRITE

WRITE is a command that must be used before any PRINT statements can be written to a file. To use the WRITE command within a program, complete the following steps. (NOTE: In the lines below, PROGRAM stands for the name of your program.)

1. Type the following line to begin your program:

```
10 d$=CHR$(4)
```

2. Type the following line as the next line of your program:

```
20 PRINT d$;"OPEN PROGRAM"
```

3. Type the following line to indicate that you want to write information into a file:

```
30 PRINT d$;"WRITE PROGRAM"
```

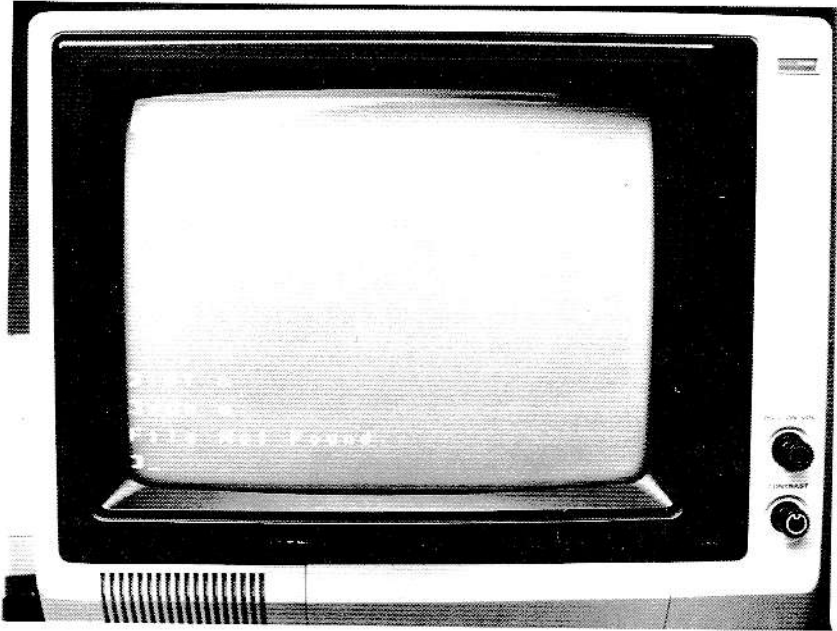
4. Type the PRINT statements to be written into your program.

5. Type the following line to end your program:

```
100 PRINT d$;"CLOSE PROGRAM"
```

XDRAW (**advanced**)

XDRAW is a statement used like DRAW (in high resolution graphics) with one exception. The color used with XDRAW is the complement of the color already in effect at the specified point(s). XDRAW must be followed by a number to indicate what shape you want to draw. Then, to indicate where the shape should be drawn, type: AT, the column location, a comma (,) and the row location.



SmartBASIC Error Messages

This chapter contains a list of the error messages you might see when using SmartBASIC. Also, under the heading of EXPLANATIONS, you will find a description of the cause of each error.

ERROR MESSAGES	EXPLANATIONS
BAD SUBSCRIPT	You have tried to exceed the dimensions established for an array.
BREAK	This message will always be followed by a line number to indicate what line number your computer was on when it stopped running your program.
CAN'T CONTINUE	The CONT command will not work for any one of the following reasons: you have encountered an error, a program line has been changed, or the program was never run.
CONTROL BUFFER OVERFLOW	The limit of your computer's input buffer has been exceeded.
DISK FULL	There is no space available on the digital data pack.
DIVIDE BY ZERO	Division by zero is not allowed.
END OF DATA	You have read past the end of a text file.
FILE LOCKED	The file cannot be changed. To change the file you must first use the UNLOCK command. (NOTE: To see which files are locked, use the CATALOG command. Locked files are indicated by an asterisk (*) in front of the filename.
FILE NOT FOUND	Your computer can not locate the filename you have indicated.

FILE TYPE MISMATCH	The file stored on your digital data pack does not correspond to the command indicated.
FORMULA TOO COMPLEX	The expression being evaluated is too complex for your computer. To use the same expression and avoid this problem, break the calculations into two lines.
ILLEGAL FUNCTION ASSIGNMENT	You have tried to use a word reserved for your computer's use in a function name.
ILLEGAL MODE	INPUT, GET, DATA, and DEF FN can only be used within program statements.
ILLEGAL QUANTITY	This message may appear in the following situations: you have used a negative array subscript, you have used LOG with zero or a negative number, you have used SQR with a negative number, or you have used TAB, SPC, ON/GOTO, POKE, PEEK, WAIT, LEFT\$, RIGHT\$, or MID\$ with inappropriate information.
I/O ERROR	This message indicates an input/output error. Check to make sure your digital data pack is correctly positioned within the drive. If you are using a blank digital data pack for the first time, type INIT and press RETURN. This procedure prepares or "initializes" the data pack.
LANGUAGE NOT AVAILABLE	The SmartBASIC digital data pack is not in the drive, or you have attempted to write a program in a language other than SmartBASIC.
NEXT WITHOUT FOR	There is no corresponding FOR statement for the indicated NEXT statement.
NO BUFFERS AVAILABLE	Too many text files have been left open.
NOT DIRECT COMMAND	The command you just typed should appear within a program.

OUT OF DATA	Your program has run out of data to read. In other words, your program has tried to execute a READ statement but all the information stored in the corresponding DATA statement has already been read.
OUT OF MEMORY	This message may appear in the following situations: there is no more RAM available for your program, too many variables have been used, there are more than 24 nested GOSUB levels, there are more than 10 nested FOR loops, there are more than 36 nested levels of parentheses (), LOMEM: has been set too high or too low, or HIMEM: has been set too high.
OVERFLOW	The answer to a calculation is larger than your computer can handle.
PROGRAM TOO LARGE	There is not enough space available for your program.
RANGE ERROR	A command parameter is too large for your computer to identify.
REDIM'D ARRAY	Your program contains two dimension statements for the same array.
RETURN WITHOUT GOSUB	There is no corresponding GOSUB statement for the indicated RETURN statement.
STACK OVERFLOW	Too many FOR/NEXT and GOSUB statements have been used.
STRING TOO LONG	A string contains more than 255 characters.
SYNTAX ERROR	A command, statement, or function is not recognized by your computer. The cause may be a missing (or extra) parenthesis and/or comma. A misspelled filename or keyword may also cause this error message.
TYPE MISMATCH	A string has been used in place of a number or vice versa.
UNDEF'D FUNCTION	A function must be defined before it can be used.

ASCII

escape

file name

hardware

m
i
c
r
o
c
o
m
p
u
t
e
r

byte

letter-quality

database

s
y
n
t
a
x

printer

word wrap

Glossary of Computer Terms

Listed below are definitions for some of the most commonly used words in the computer field. You do not need to know these words to use your ADAM computer. However, you may find knowing these definitions helpful.

TERMS	EXPLANATIONS
application	A certain job, such as organizing household records, that can be solved by using a computer.
ASCII	A character code used for representing information. ASCII is pronounced "ask-ee" and stands for A(merican) S(tandard) C(ode) for I(nformation) I(nterchange).
BASIC	A widely used programming language developed at Dartmouth College. The word BASIC stands for B(eginners') A(ll-purpose) S(ymbolic) I(nstruction) C(ode).
bit	The smallest unit of data recognized by a computer. A bit can have only two values—0 or 1. Bit is short for b(inary dig)it.
board	A board (made of fiberglass, press paper, or plastic resin) within a computer that contains chips and electronic circuits.
bug	A mistake in a program. See debugging .
bus	A group of electrical connections that carry signals within a computer system.
byte	The number of bits used by a computer to represent a single character. Usually, one byte is equal to eight bits.
character	A single letter (A-Z), number (0-9), or symbol (\$, :, *).

chip	A very small wafer made of silicon.
COBOL	A programming language used for business applications. The word COBOL stands for CO(mmon) B(usiness) O(riented) L(anguage).
CP/M	A popular operating system for microcomputers. CP/M stands for C(ontrol) P(rogram) for M(icrocomputers).
CPU	The C(entral) P(rocessing) U(nit) of a computer that controls the transfer of information and performs calculation.
cursor	The small, bright "underline" character or square that appears on your screen. The cursor on a computer is like the point of a pencil—wherever the cursor is positioned is where your computer will be "writing" next. Cursors can be different shapes (rectangle, square, parenthesis, etc.), depending upon the system. Cursors are sometimes referred to as prompts.
daisy-wheel	A print head that is shaped like a wheel, with each character at the end of a spoke. A printer that uses a daisy-wheel creates printed material that looks like typed material.
data base	A collection of organized information.
data processing	An application when a computer only works with numbers.
debugging	Correcting the mistakes in a program.
digital data pack	The programs and other information that you type on your ADAM Family Computer System can be saved for future use on a digital data pack. A digital data pack looks like an audio cassette tape, but they are not interchangeable. A digital data pack operates at a very high speed.

digital data pack drive	In order for ADAM to store programs and other information, a digital data pack must be inserted into a digital data pack drive. The "drive" is comparable to the turntable on a record player, in that the drive holds the data pack so that it can be used. One drive comes with the system; a second drive can be purchased and added onto the system.
disk	A flat, circular plate with a magnetic coating used to store information for some computers. The size and capacity of computer disks can vary. Disks are sometimes called "diskettes."
disk drive	A device used to read information from or write information onto a disk.
document	Anything typed on a word processor (for example, a letter, report, or chart).
DOS	An operating system that stores information on disks. DOS stands for D(isk) O(perating) S(ystem) and is pronounced "dass."
dot matrix printer	A printer that forms characters by using many small dots. Dot matrix printers are usually less expensive than letter-quality printers.
file	A group of information, with a specific name, that is treated as a single item. A computer stores information as a "file" so it can be recalled later.
file name	The characters assigned by a user to identify a file.
FORTRAN	A high level programming language generally used for scientific work. The word FORTRAN stands for FOR(mula) TRAN(slator).
graphics	Lines and figures used to draw charts and other illustrations on a computer.
hard copy	Computer output printed on paper.
hardware	The physical equipment in a computer system.

input	Information entered into a computer's memory.
K	A way to measure the size of a computer's memory. One K equals 1,024 bytes.
keyboard	The keys of a computer used to communicate information and instructions. Most computer keyboards contain standard typewriter keys and special function keys. Some computers (like ADAM) also have a "number pad" to make typing numbers easier.
letter-quality printer	A printer (like the one that came with your ADAM Family Computer System) that produces printed copies like those created on a typewriter.
listing	A printed copy of a computer program.
mainframe	A very large computer that can store a great deal of information.
memory	The part of a computer where numbers, words, and other information is stored.
microcomputer	A computer that is physically small (like ADAM) and can usually fit on a desk or table.
modem	A device used to transmit information between computers over telephone lines. The word modem stands for mo(dulator) dem(odulator).
monitor	A television or other device used to see the information being processed by a computer.
motherboard	The main board, inside a computer, into which all other boards are inserted.
number crunching	Performing complicated mathematical calculations.
operating system	A group of programs that controls the overall operation of a computer.
output	Information created by the computer. This information might be displayed on a screen, printed on a printer, or stored on a tape or disk.

PASCAL	A high level programming language. PASCAL is named after the French mathematician Blaise Pascal.
password	A unique series of letters, numbers, or characters assigned to a user to limit and/or identify access to a computer or computer system.
peripherals	Additional hardware (for example, game controllers or a modem) that can be attached to a computer.
printout	Any hard copy produced by a computer.
program	A sequence of instructions that tell a computer exactly what to do.
prompt	A message, displayed on the computer screen, directing the user to perform a specific action.
RAM	Memory space within a computer where you can store and recall information. RAM stands for R(andom) A(ccess) M(emory).
ROM	Memory space within a computer where you can only recall information. ROM stands for R(ead) O(nly) M(emory).
spreadsheet	A program that turns a computer's memory into a large balance sheet. Numbers can be entered in the program and "what if" situations explored by using the computer's ability to quickly recalculate numbers.
software	The programs that make a computer perform certain jobs.
syntax	Rules that apply to the way a computer program should be written.
syntax error	A mistake made in the use of a computer language.
system	A combination of computer hardware and software that performs one or more specific functions.

telecommunications	Communication between computers, using telephone wires and/or cables.
tractor feed	A piece of equipment attached to the printer, designed to help move paper through the printer.
user	The person using a computer, sometimes known as the "operator."
variable	Character(s) in a program that can be assigned value(s). For example, the variable A might be assigned the value of 257 and the variable N\$ might be assigned the value of "NAME."
word processor	A computer program that allows the user to type, change, and then print text.
word wrap	A common feature in word processing programs that automatically moves words that are too long to fit at the end of one line down to the beginning of the next line. Word wrap eliminates the need to press RETURN at the end of each line. Instead, the user only presses RETURN to end a short line or create a blank line.

6 Index

—A—

ABS, 256
ADAM computer,
 care of, 40-42
 setting up the, 13-24
 troubleshooting, 25-26
 turning off the, 27-28
 turning on the, 25
 use as a typewriter, 28, 45-51
AND, 189
APPEND, 256
Arrow keys, 222, 245
ASC, 257
ATN, 257

—B—

Backspacing, 59, 60
BLOAD, 257
BRUN, 257
BSAVE, 258

—C—

CALL, 258
Cartridge reset, 8
CATALOG, 258
CHR\$, 258
CLEAR, 67, 222, 245, 259
CLEAR WK-SPACE, 67
CLOSE, 259
CLRERR, 260
COLOR =, 145, 261
Color, changing background, 223
Color select codes, 144, 200

Command keys, 221, 245
Commands, 256-303
Computer reset, 8, 238, 249
CONT, 262
CONTROL, 224
COPY, 223, 245
Copying a file, 74, 224
Copying text, 94
Correcting text, 61-62
COS, 261
Cursor, 245
 moving the, 60

—D—

Daisy wheel changing, 37-38
DATA 262
DEF FN, 263
DEL, 264
DELETE, 62, 225, 246, 264
Delete a file, 225
Digital data pack drives, 8, 246
 inserting and removing, 39-40
DIM, 265
Document defined, 246
DONE, 57
DRAW, 266

—E—

Edit defined, 246
END, 178-179, 266
Equations, typing of, 91
ERASE OFF, 63
Error messages, 246, 305-308
ESCAPE, 62, 226, 247
ESCAPE/WP, 56, 63, 225, 243

EXP, 266
Expansion module #3, setting up the,
20-24

-F-

Fan-fold paper, 247
File defined, 247
File directory defined, 247
File maintenance, 73-77
File name, 247
File renaming, 236
Files
 copying of, 74
 deleting of, 75
FINAL CLEAR, 67
Footnotes, typing of, 91
FOR/NEXT loops, 193-199, 267
Format defined, 247
Formulas, typing of, 91
FRE, 267
Functions described, 256-303

-G-

Game cartridge slot, 8
GET, 226, 245, 268
GET FILE, 68
Glossary,
 computer terms, 311-316
 error messages, 305-308
 SmartWRITER terms, 245-251
GOSUB, 183-190, 268
GOTO, 107, 269
GR, 145, 269
Graphics,
 high resolution, 201-207
 low resolution, 145-151

-H-

HCOLOR = , 201-202, 206, 270
HGR, 201-202, 206, 271
 using, 203
HGR2, 201-202, 206, 271
 using, 204
HI-LITE, 62, 227, 248
HI-LITE ERASE, 63, 228
HIMEM, 272
HLIN, 146, 148, 151
HPLOT, 202, 205-206

HOME, 60, 131, 228, 248, 272
Home and arrow keys, 228
Horizontal line (HLIN), 146, 148, 151
Horizontal margin, 248
HPLOT, 273
HTAB, 273

-I-

IF/GOTO, 274
 mathematical symbols, 254
IF/THEN, 173-180, 275
 symbols used in, 174, 180, 254
INIT, 275
INPUT, 115, 276
 defined, 4
INSERT, 61, 229, 248
INT, 165-171, 277
Integers (INT), 165-171, 277
INVERSE, 277
I/O devices, 5
I/O ports, 5

-K-

Keyboard components, 5-7

-L-

LEFT\$, 278
LEN, 278
LET, 166-167, 279
Letters, typing of, 98-99
Line spacing, changing of, 49, 83, 229
LIST, 131-132, 279
Lists, types of, 100-101
LOAD, 137, 141, 280
LOCK, 280
LOG, 280
LOMEM, 281

-M-

Maintaining the system, 35-42
Margin setting, 48, 230
Mathematical operations, 254
Mathematical symbols, 128, 135
Memory console, 7-10
Microprocessor, 9

MID\$, 281
 MON, 281
 Monitor, 7
 clearing the screen, 27
 Motherboard, 10
 MOVE/COPY, 62, 223, 231, 248
 Moving text, 94
 Moving window format, 80, 232, 248

-N-

Negative speed random numbers, 166, 170
 NEW, 140, 281
 NOMON, 281
 NORMAL, 282
 NOTRACE, 282
 Numeric variable, 253

-O-

ON GOSUB, 284
 ON GOTO, 284
 ONERR GOTO, 283
 "Oops" commands, 63
 OPEN, 285
 Operators, 254
 OR, 189
 Output defined, 5

-P-

Page end, 233
 Paper size and type, 234
 PDL, 285
 PEEK, 285
 PLOT, 146, 150, 151, 286
 POKE, 286
 POP, 287
 POS, 287
 POSITION, 288
 PR#, 288
 PRINT, 107, 235, 289
 PRINT WK-SPACE, 65
 Printer, *see* SmartWRITER Printer
 Programs,
 COOKBOOK, 216-217
 F/M CONV, 214-215
 GRAPHICS, 161
 INTRO, 156-157
 MATH DEMO, 158-159
 MILEAGE, 160

POEM, 210-211
 ready to run, 30-31
 TREASURE, 212-213
 Punctuation,
 marks, 127
 symbols, 255

-R-

RAM, 9
 Random numbers (RND), 165-171
 negative speed, 166, 170
 READ, 289
 RECOVER, 290
 REM, 115-116, 290
 RENAME, 291
 Renaming a file, 236
 Replacing text, 95, 237, 248
 Reports, typing of, 98-99
 Reset, 238, 249
 RESTORE, 292
 RESUME, 292
 Resumés, typing of, 100-101
 RETURN, 58, 183-190, 293
 Revising text, 93-95
 Ribbon changing, 35-37
 RIGHTS\$, 293
 RND, 165-171, 294
 negative speed, 166, 170
 ROM, 9
 ROT =, 294
 RUN, 107, 294

-S-

SAVE, 137, 139, 295
 Saving files, 66
 SCALE =, 295
 SCREEN OPTIONS, 57, 238
 SCRIN, 296
 Scroll, 248
 SEARCH, 239, 248
 Searching for and replacing text, 95
 Setting up ADAM computer, 13-24
 SGN, 296
 SIN, 296
 Smart key message, 248
 Smart keys, 221, 250
 SmartBASIC, 31-32
 colors and, 144-151
 error messages, 246, 305-308
 GOTO, 107

INPUT, 115
introduction to, 105-107
mathematics and, 128-129, 135
PRINT, 107
punctuation marks, 127
reference guide, 253-303
REM, 115-116, 290
storing programs, 137-143
SmartWRITER printer, 10-11
changing tabs, 82
changing margins on, 81
changing daisy wheel for, 37-38
changing the ribbon for, 35-37
options, 87-90
reference guide, 221-243
using the, 26-27
SmartWRITER terms glossary, 245-251
SmartWRITER word processing, 28-30
Sound, 239
SPACEBAR, 60, 61
SPC, 297
SPEED = , 297
SQR, 298
Statements, 256-303
STOP, 298
STORE/GET, 226, 240
Standard format, 240, 250
Store, 250
STORE WK-SPACE, 66-70
STORE/GET, 62, 66, 68, 70, 250
Stored files, recalling of, 68-69
STR\$, 299
String variable, 253
Subscripts, 91, 241, 250
Superscripts, 91, 241, 250

-T-

TAB, 58, 299
Tabs, 242
Tabulation, changing of, 50
TAN, 299
Telephone list, typing a, 79-85
TEXT, 145, 300
TRACE, 300
Typing lessons, 46-51

-U-

UNDO, 62, 63, 67, 243, 251
UNLOCK, 301

-V-

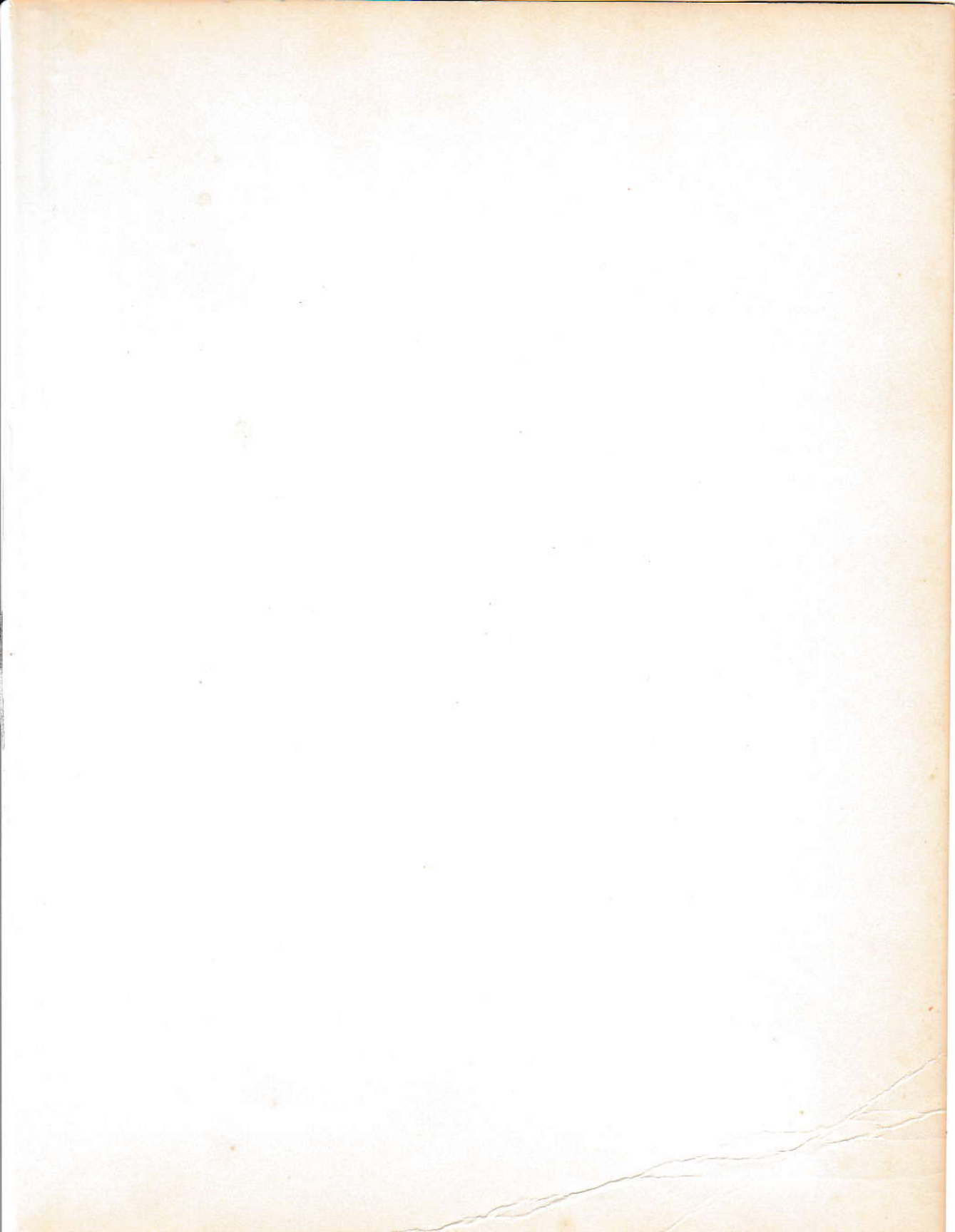
VAL, 301
Variables, 129-130, 253
numeric, 253
string, 253
Vertical line (VLIN), 146, 149, 151
Vertical margins, 250
VLIN, 146, 149, 151, 302
VTAB, 302

-W-

WAIT, 302
WILD CARD, 243
Word processing, 28-30, 55-71
clearing the screen, 67
correcting errors, 61-62
deleting text, 62
file maintenance, 73-77
inserting text, 61
moving the cursor, 60
printing, 64-65
printing options, 87-89
procedures, 249
recalling stored files, 68-69
revising text, 93-95
saving text, 66
screen features, 57
standard printer settings, 97
superscripts and subscripts, 91, 241, 250
typing a telephone list, 79-85
typing equations, formulas, and footnotes, 91
typing letters, reports, lists, and resumes, 97-101
Work space, 251
WRITE, 303

-X-

XDRAW, 303



ADAM USER'S GUIDE

A.J. Bell and E.Q. Bell

Here it is— a step-by-step tutorial designed for beginning users of the ADAM! This fun and easy-to-use guide takes you "by the hand" and walks you through the entire ADAM Family Computer System— from hooking it up to programming it! Covers every feature of the ADAM you need to know, including SmartWRITER for word processing. . SmartBASIC for programming. . plus a comprehensive reference section that includes:

- A SmartWRITER Reference Guide— alphabetically lists each SmartWriter command, describes what each command does, and explains how to use it!
- A SmartBASIC Reference Guide— alphabetically lists the most frequently used SmartBASIC key words and describes their functions!
- A Glossary Of Computer Terms— including 60 of the most frequently used computer buzzwords!
- Plus— a glossary of SmartWRITER terms, SmartBASIC error messages, and more!

CONTENTS

What You've Got/Hooking It Up/Using Your ADAM Computer/Maintaining The Computer/Using ADAM As A Typewriter/Typing Your First Letter/Getting Organized— File Maintenance/Typing A Telephone List/Using Your Printing Options/Typing Equations, Formulas, And Footnotes/Revising Text/Applications—Typing Letters, Reports, Lists And Resumes/Getting Started/Making Changes/Adding It Up/Storing Your Program/Color A Rainbow/Try These/Pick A Number/Comparison Shopping/Take A Trip/Around And Around/More Rainbows/More Programs To Try



ISBN 0-89303-300-6