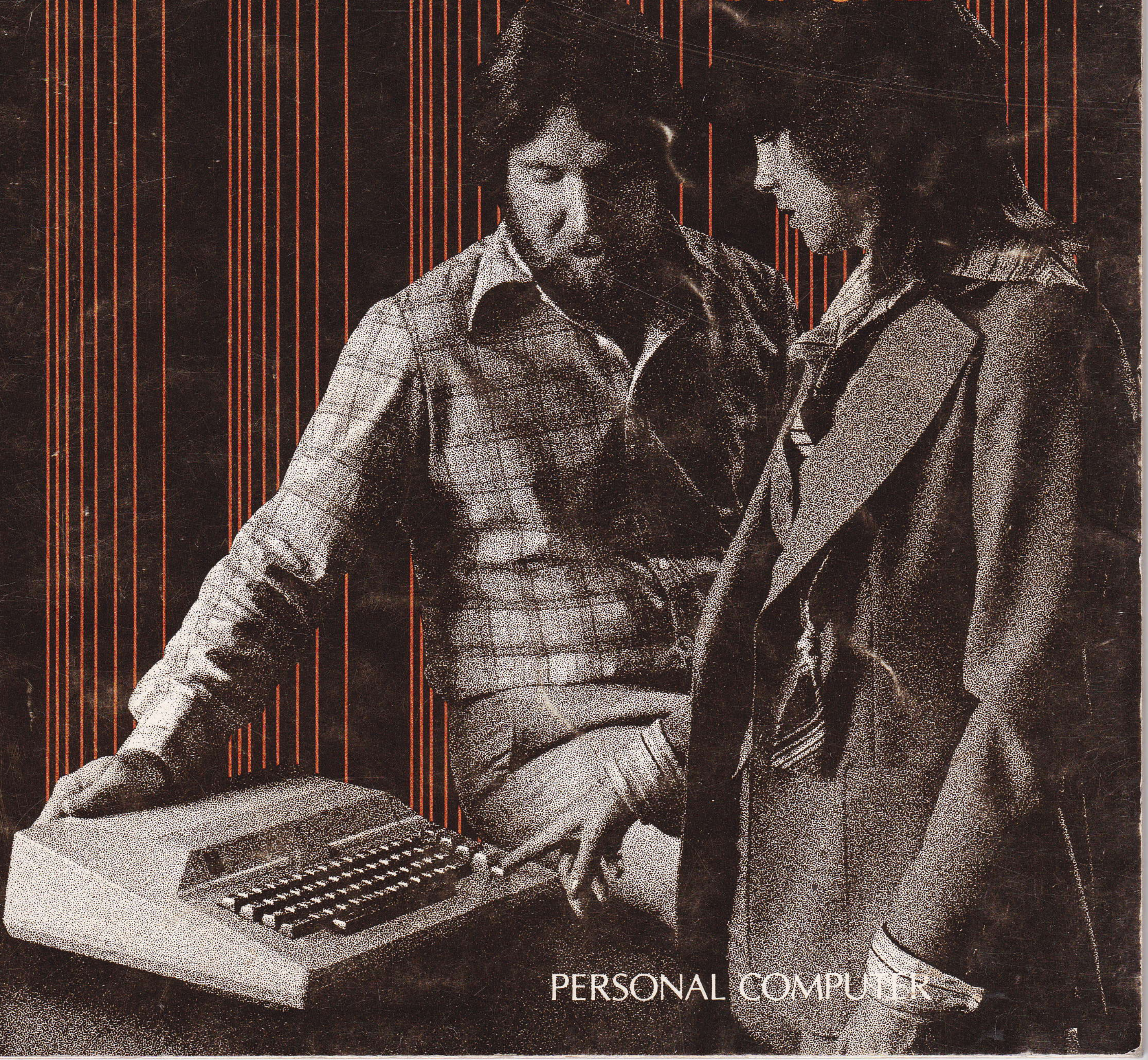
ATARI 800™

OPERATORS MANUAL



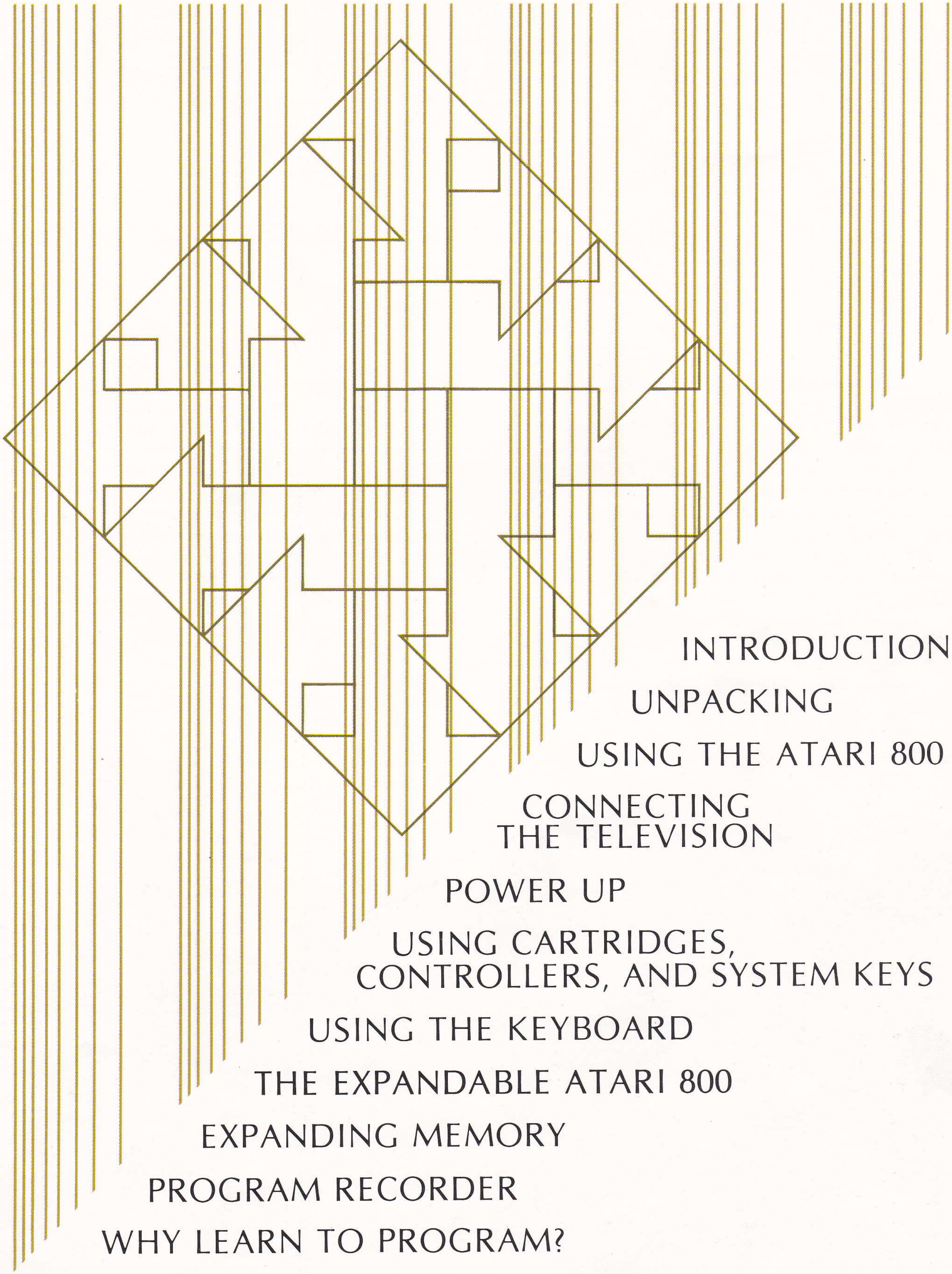
PERSONAL COMPUTER

Every effort has been made to ensure that this manual accurately documents the operation of the ATARI 800 computer. However, due to the ongoing improvement and update of the computer software, Atari, Inc. cannot guarantee the accuracy of printed material after the date of publication, nor can Atari accept responsibility for errors or omissions. Revised manuals and update sheets will be published as needed and may be purchased by writing to:

Atari Software Support Group
P.O. 9027
Sunnyvale, CA 94086

Printed in U.S.A.

TABLE OF CONTENTS



INTRODUCTION

ATARI 800™- the personal computer with something for everyone

Create a playfield for fast-paced games. Games provide hours of entertainment. You can purchase a wide variety of Game Cartridges. Each game is designed for one to four players. Just insert the cartridge and connect the controllers. The **ATARI 800™** will transport you to a fantasyland of skill and competetion. Some games simulate competitive sports such as Basketball, well-known board games such as Chess, or puzzles such as Hi-Q. Other cartridges are new inventions created especially for the **ATARI 800.** Games are adaptable to your own mood and style. Play with your family, with friends or by yourself. When players have a wide range of skills, use the computer to equalize competition by selecting a different level of skill for each player.

Deliver Interactive, audio-visual, Educational Materials. A variety of Interactive Educational Materials present more than twenty subjects through text, audio, diagrams and pictures. Each program carries on a dialogue with you through the **ATARI 800** screen, speaker and keyboard. These programs provide individualized instruction tailored to your pace and achievement level. You receive immediate feedback on your accuracy and understanding. Programs for kindergarten through college level will be available in a wide variety of subjects.

Provide a serious tool for home and business Information Mangement. Information Management packages aid you in the orderly storage and quick retrieval of numeric, graphic, pictorial and text information. The Financial Management programs help you balance your checkbook and keep track of income and expenses. The Nutritional package helps you plan healthful meals and diets which include all essential nutrients. The Investment Analysis programs aid in capital assets management by tracking stock portfolios. A Mailing List program sorts, edits and prints lists of names and addresses for you (an optional printer is needed).

Perform all the functions of a fully programmable General Purpose Computer. Your **ATARI 800** is actually a fully programmable General Purpose Computer. As you explore the capabilities of the **ATARI 800** you will increase your knowledge of computer science considerably. The Atari **BASIC** Language Cartridge expands the Operating System software to include a versatile, easy-to-learn, high-level programming language for both student and professional use. The Operating System software provides access to the central processing unit, memory, input/output (I/O) ports and file manipulation utilities. Even if you have no experience at all, the **ATARI 800** and an enquiring mind are all you need to learn to develop custom applications for your personal computer.

You have just purchased the basic system which includes:

- Computer Console
- TV Switch Box
- AC Power Adapter
- Program Recorder and Power Cord
- 4 Joystick Controllers
- 2 Cartridges
  Basketball Game
  Atari BASIC Language
- 2 Instruction Books
  Atari 800 Operators Manual
  Atari BASIC Programming Guide

UNPACKING

Your **ATARI 800** is a series of components which function together as a single system.

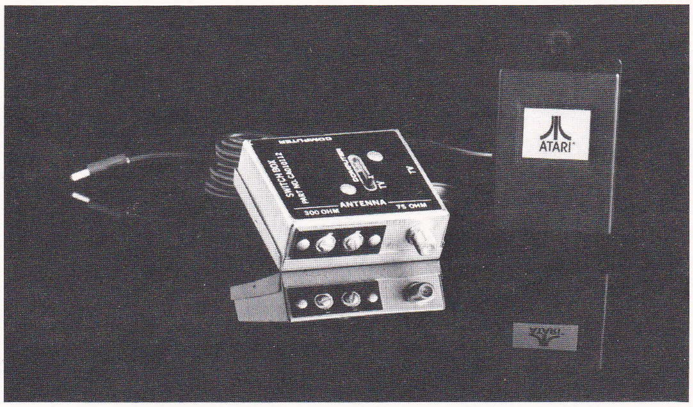
These components, pictured on the next page, are the computer hardware. They are made of silicon, metal, plastic, paper and ink. They are the tangible components of the **ATARI 800** Basic System. Equally important, but impossible to photograph is the **Atari 800** software. *Software* is the information and instructions encoded in the *hardware*. The software components of the **ATARI 800** Basic System are the Operating System software, the Basketball Game Program, Atari BASIC Language and, of course, the contents of the two books.

The **ATARI 800** Basic System may be enhanced in several ways. Your Atari retailer has information on expanding your system with **RAM Memory Modules, a Printer, Floppy Disk Drives** and newly released Cartridge, Disk and Cassette Software.

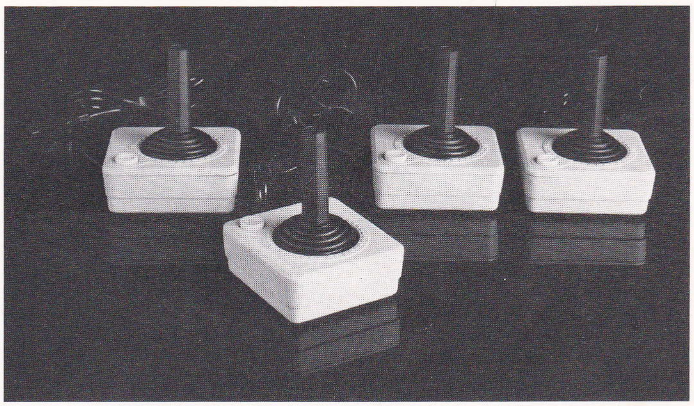
Save all your **ATARI 800 packaging materials.** The *accessory tray* and *polyethylene bag* will keep your **ATARI 800** components organized and dust free when in daily use. The *outer carton* and *foam end caps* will protect your **ATARI 800** during moving or long term storage.

cessing Unit (CPU), the Memory Bank containing the *Operating System Read Only Memory (ROM)* and 4k (4 thousand characters or "bytes") of user programmable *Random Access Memory (RAM),* and two *Expansion Sockets* for additional RAM Memory Modules. The Console also holds the *Keyboard, Cartridge Slots, Controller Jacks* and a *Serial I/O Port* for connecting to peripheral components.

The **TV Switch Box** allows you to change from regular TV reception to **ATARI 800 Computer** display by moving the sliding switch on the box. The **AC Power Adapter** plugs into a normal wall socket and converts it to the low voltage used by your **ATARI 800.**



The **Joystick Controllers** allow up to four people to control the computer display and participate in games simultaneoulsy.



The **Basketball Game Program** and **Atari BASIC Language** are contained in the **cartridges** included with your basic system. Additional cartridges may be purchased from your Atari retailer. Complete instructions accompany each cartridge.

The **ATARI Program Recorder** provides software storage in computer readable form. You may purchase **Preprogramed Cassettes** from your Atari retailer and you may use any blank, high-quality audio cassette tape to save programs you write yourself.

USING THE ATARI 800

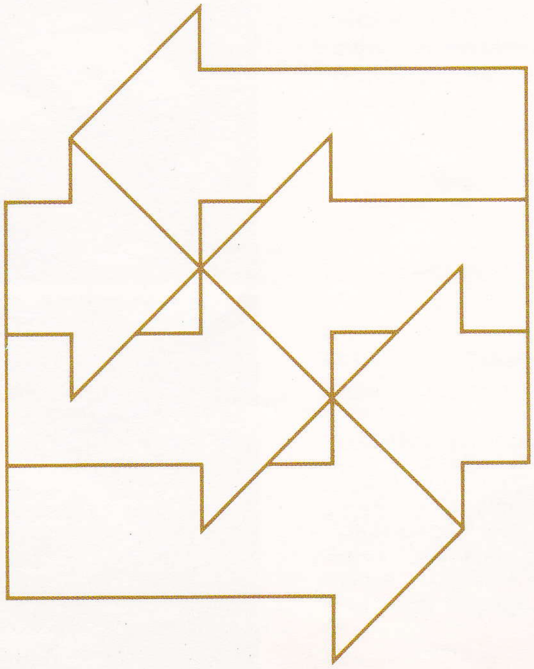
The **ATARI 800** Basic System is very much like an empty sheet of paper. It has the potential for an unlimited number of applications. But that potential remains dormant until you add the software.

ATARI 800 software consists of an integrated series of coded instructions called programs. The foundation programs are supplied in the **Operating System ROM Module.** They activate the *keyboard, controllers* and the *screen display*. These programs control the flow of all information within the computer.

To use the **ATARI 800,** you add a second level of software. This can be done by inserting a cartridge into a cartridge slot. The software transforms the **ATARI 800** into a special purpose machine for playing a game, presenting educational material, manipulating information or entering programs through the keyboard. The program recorder and the optional floppy disk drive provide additional methods for loading programs into the **ATARI 800.**

Atari Game Cartridges, designed by Atari's staff of professional programmers, contain programs that are permanently recorded in a ROM within the cartridge. They control the computer in machine language, the most intricate level on which to manipulate the computer. These Atari Game Cartridge programs produce full color, animated displays and complex electronic decision making. Each **ATARI 800** game is designed to be easy for you to learn but difficult to master. You select the computer's role...either as your opponent or your playfield...for many hours of innovative leisure.

Atari Application Cartridges have a more serious purpose. They are tools for increasing your speed and accuracy in handling words and numbers. Atari programmers identify and analyze problems of interest such as checkbook balancing or mailing list sorts and updates. They design a *generalized* solution to each problem, then program that solution in machine language and record it in the cartridge. When you insert the cartridge, the **ATARI 800** repeats this preprogrammed solution, substituting *your data* from the keyboard into its equations. Although all cartridges operate in the same general fashion, each cartridge causes the **ATARI 800** to respond in a different way. You will need to read the instruction sheet which accompanies each cartridge for specific details. Keep these instruction sheets and your personal notes on the operation of the **ATARI 800** together with this manual in a 3-ring binder.



CONNECTING THE TV SWITCH BOX

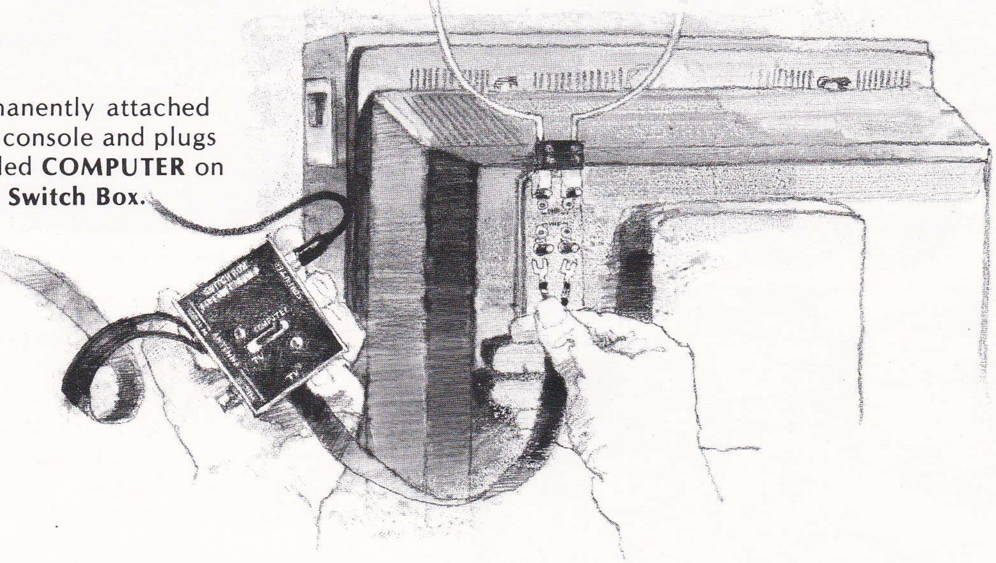
The **TV Switch Box** is designed to be permanently attached to your television set and will not interfere with normal television reception. It has an adhesive backing to hold it in place on the back of the television and may be used with any of Atari's home video products.

When this switch is in the **COMPUTER** position (as shown) your television receives signals from the **ATARI 800.** Tune your television to channel 2 or 3 whichever is *weaker* in your area. When the switch is in the **TV** position, signals come from your television antenna. You will need a screwdriver to install your **TV Switch Box.**

Disconnect your present antenna from the VHF terminals on the back of your television. Notice whether the antenna cable is the round, 75 OHM variety with a screw-on connector or the flat, twin-lead 300 OHM cable. Attach it to the matching 75 or 300 OHM connector labeled **ANTENNA** on the side of the **TV Switch Box.**

Now attach the short flat, twin-lead, 300 OHM cable labeled TV on the bottom of the **TV Switch Box** to the VHF screw terminals on the television.

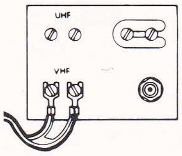
If your television antenna is of the 300 OHM variety, your **TV Switch Box** is now installed.



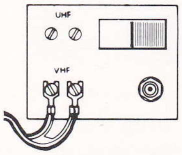
This cable is permanently attached to the **ATARI 800** console and plugs into the jack lableled **COMPUTER** on the side of the **TV Switch Box.**

TO EXTERNAL ANTENNA

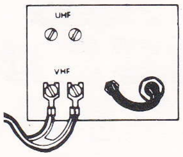
If your antenna is 75 OHMS, you must convert your television to accept the 300 OHM signal from the **TV Switch Box.** Check the antenna connections on the back of your television.



If it looks like this, loosen the screws holding the U-shaped slider and move it to the position marked **300 OHM or 300 .**



Like this? Push the switch to the **300 OHM (300)** position.

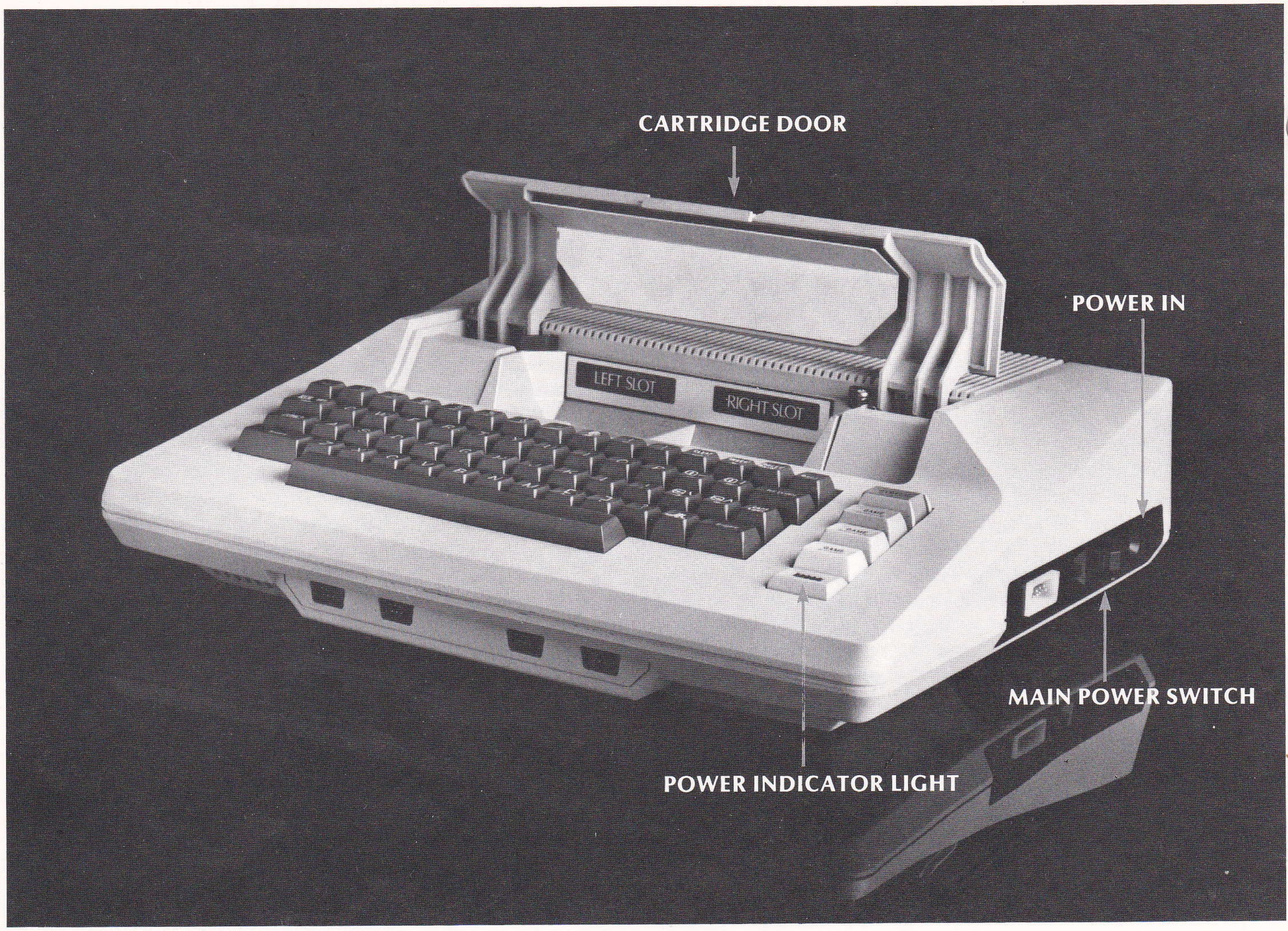


Like this? Screw the short round wire into the connector provided.

Federal Communications Commission Regulations Require:

- Avoid using any longer twin-lead wire from the TV/Game Switch Box to your television that is supplied with the TV/Game Switch Box.
- Avoid connecting the twin-lead wire from the TV/Game Switch Box to any television antenna or cable-TV outlet.
- Avoid attaching loose wires to your antenna terminals when using your **Atari 800 Personal Computer** .

Any of the above may cause interference to nearby television sets and is against Federal Communications Commission (FCC) regulations.



POWER UP

Once the **TV Switch Box** is connected, plug the **AC Power Adapter** into any 115 volt outlet (ordinary house current). Plug the end of the AC Power Adapter cable into the jack labeled **POWER IN** on the side panel of the Computer Console.

Push the *main power switch,* labeled **POWER ON/OFF** in the side panel, to **ON.** In order to prevent damage to the computer and cartridge, a secondary switch automatically turns **ATARI 800** off whenever the cartridge door is lifted. The *power indicator light* will only go on when the main power switch is **ON** *and* the cartridge door is closed. If the television is also on, tuned to channel 2 or 3 and properly connected, you will see a display from the **ATARI 800** on the screen. Whenever the cartridge door is opened the power light will go out and the screen display will go blank. The **ATARI 800** will automatically restart when the cartridge door is closed again.

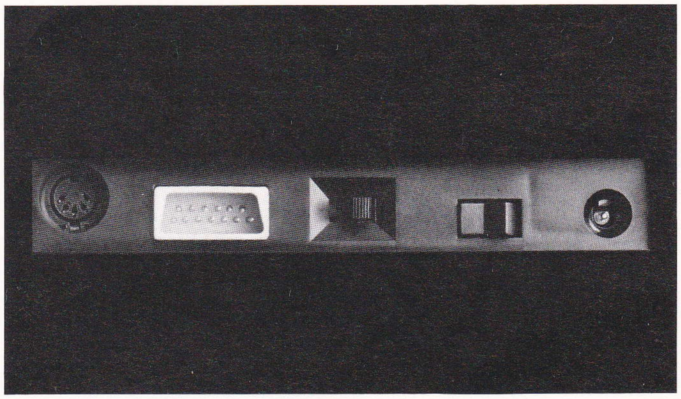
Although the **ATARI 800** is engineered for durability, do not allow liquids or tiny objects to get inside the Computer Console. Also, keep the brightness and contrast controls on your television set at moderate or low levels. A long period of exposure to a static display at excessively high contrast or brightness can leave a permanent shadow on your television screen.

SIDE PANEL

A *video monitor display screen* or a *video tape recorder* may be used in place of, or in addition to, an ordinary television for output. See the **ATARI Reference Manual** for DIN jack specifications.

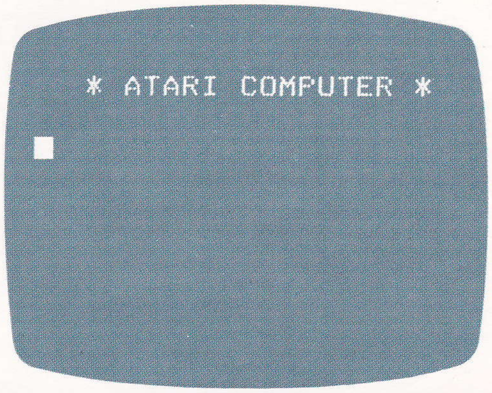
The **ATARI 800 Program Recorder** plugs into the Serial I/O jack labeled **PERIPHERAL.** Additional components may be attached to the **ATARI 800** in daisy chain series from this jack. **Section 8** explains the use of the **Program Recorder.** Refer to the operating manuals that come with each of the components for further information.

Set the switch labeled **2-CHAN-3** to channel 2 or 3, whichever has a *weaker* signal in your viewing area. Be sure your television is tuned to the same channel.



SCREEN DISPLAY

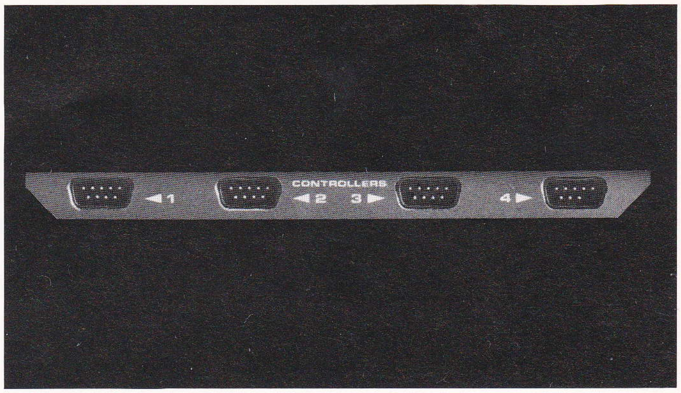
This *display* will appear on power up when cartridge slots are *empty*. Each cartridge displays its own *initial screen* which is explained in the cartridge instructions.

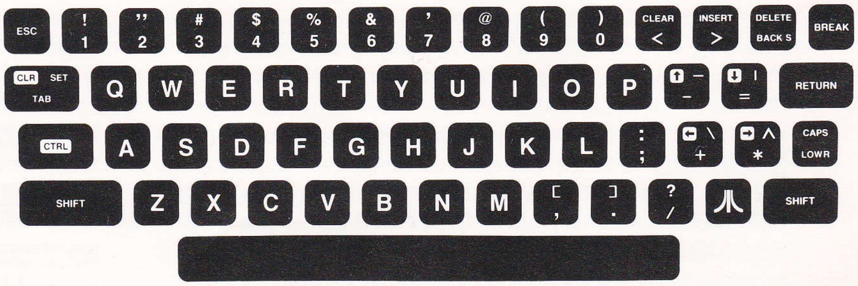


```
* ATARI COMPUTER *
█
```

CONTROLLER JACKS

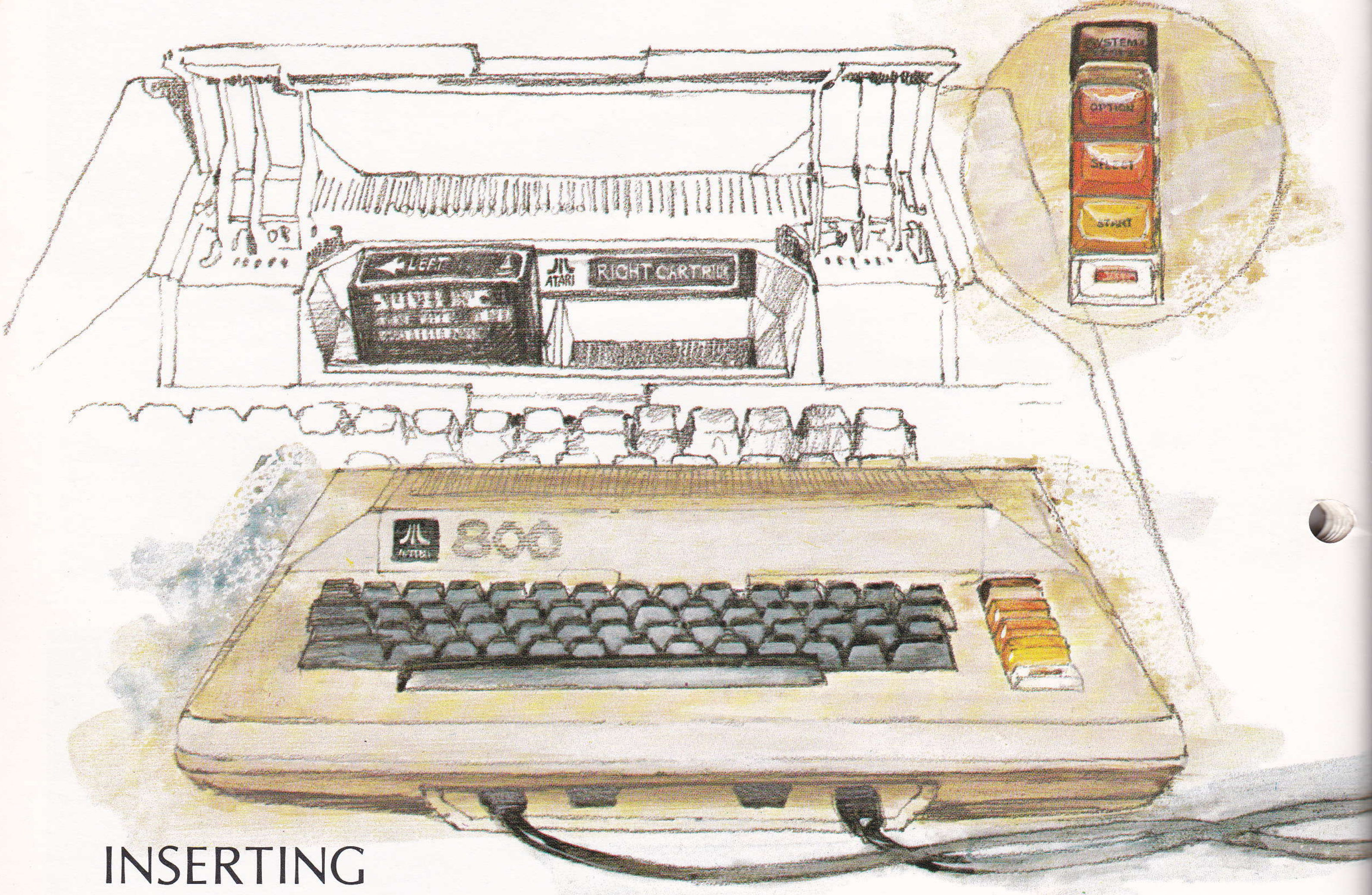
Controller Jacks accommodate one to four *controllers* for games and other applications. These jacks may also be used for future components such as a light pen or musical keyboard.





KEYBOARD

The **ATARI 800 Keyboard** is especially designed for flexibility. It has alphabetic, numeric, graphic and control functions which are detailed in **Section 7.** Each key has the capacity to be redefined by instructions from an individual cartridge. This provides you, the **ATARI 800** user, with the most efficient set of symbols for each **ATARI 800** Computer application.



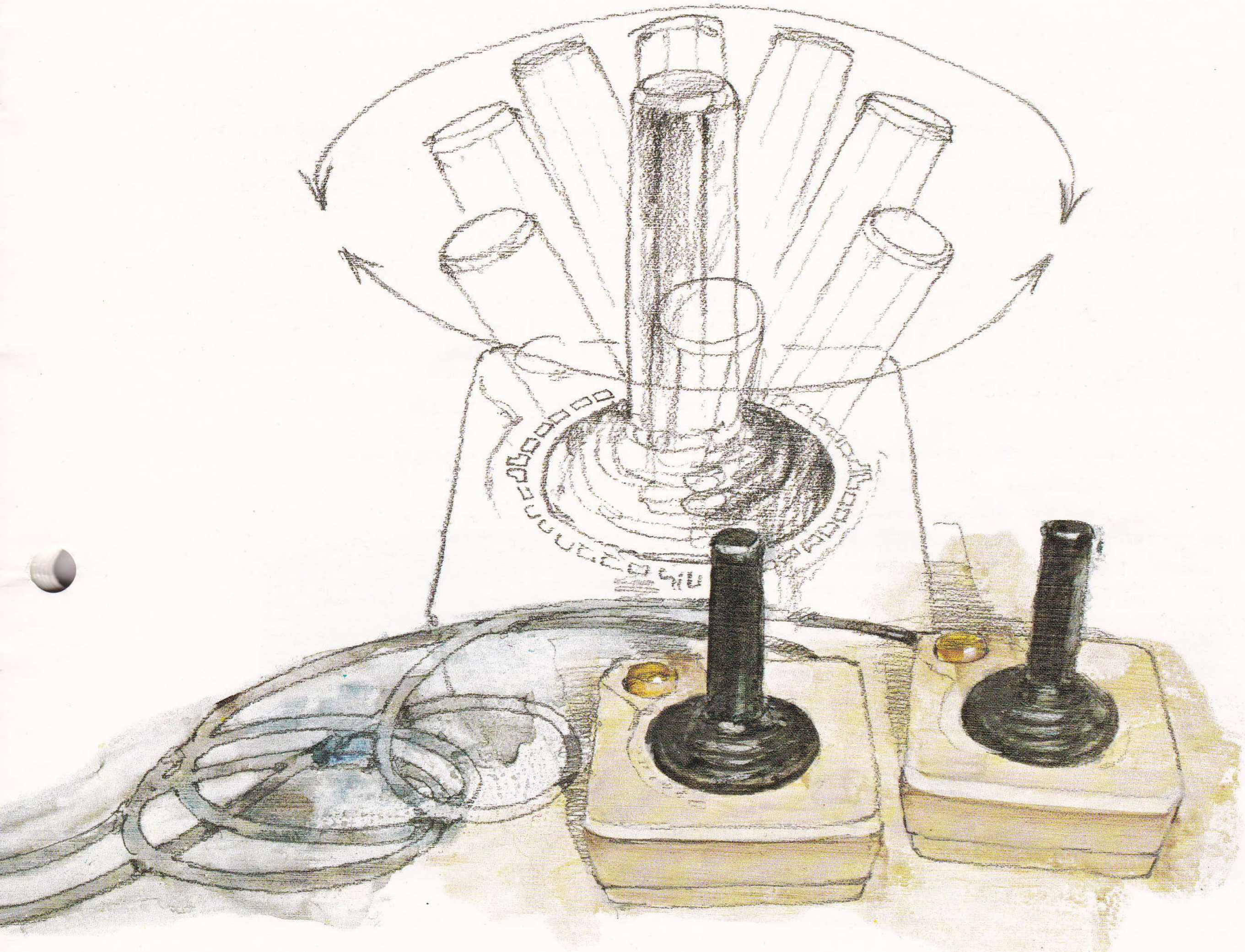
INSERTING CARTRIDGES

To insert a cartridge, pull the lever marked **PULL OPEN.** The cartridge door will spring open. Hold the cartridge with the label toward you and the opening down. Push it firmly, straight down, into the slot. It will snap gently into place. When it is all the way in the cartridge slot the door will just close over the cartridge without touching it. Push GAME START to begin running a cartridge. Single cartridges should always be inserted in the left slot and will be labeled **LEFT CARTRIDGE** on the end opposite the opening. Double cartridge sets will be labeled **LEFT CARTRIDGE** and **RIGHT CARTRIDGE.** Insert them both in the correct cartridge slots.

THE SYSTEM KEYS

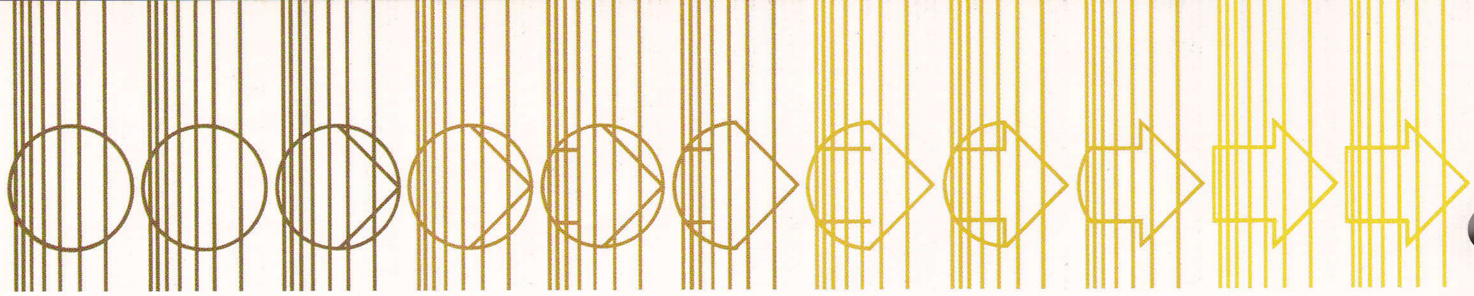
The four keys to the right of the keyboard allow you to select different starting positions within a cartridge. Each starting position is the beginning of a game or application stored within a single cartridge.

Push SYSTEM RESET to stop the computer and restart from the beginning of the cartridge. Push GAME SELECT to see the initial screen at the beginning of the next game or application. Push GAME OPTION to choose among the variations possible within a game or application. After you have made your choices with the GAME SELECT and GAME OPTION keys, push GAME START to begin the action. More complete instructions are provided with each cartridge.



USING THE JOYSTICK CONTROLLERS

Many cartridges use the Joystick Controllers to move images on the display screen. All four joysticks are identical and can plug into any of the controller jacks shown above. Each joystick has one button and eight possible stick positions. Hold the joystick with the button in the upper left-hand corner and push the top of the stick in the direction of the arrow, as shown in the diagram. Consult the cartridge instructions to determine whether joysticks should be used and, if so, what each position means.



USING THE ATARI 800 KEYBOARD

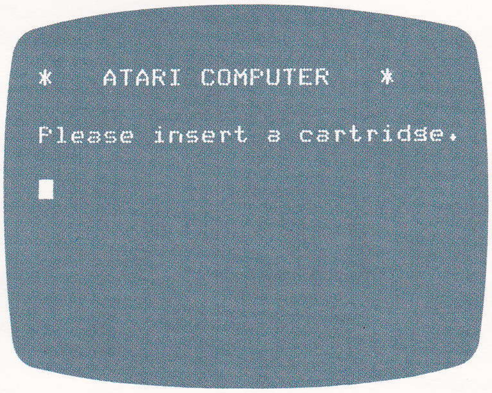
The **ATARI 800** Keyboard puts a versatile set of *printing characters* and *display control functions* at your fingertips. Many special-purpose application cartridges and some games will require you to type your responses or commands to the **ATARI 800** Computer using these keys. Most keystrokes produce a visable change on the **i** display screen. However, there are a few keys which are only used in combination with others. To investigate the effects of each key, power up your **ATARI 800** without a cartridge in either of the cartridge slots. You will see the display pictured below.

Notice the square below the **P** in Please. This square is called the CURSOR. A cursor is a mark which indicates where the next character you type will appear on the screen. The ability to move the cursor to any position on the screen and change the characters being displayed is one of **ATARI 800's** most useful features.

A glance at the keytops on your **ATARI 800** tells you that its keyboard closely resembles an ordinary typewriter.

Play with the keyboard for a few minutes. There is no way you can damage your **ATARI 800** by typing, and you will quickly become familiar with the various key functions. Each key will repeat its function rapidly if you depress it for longer than one second.

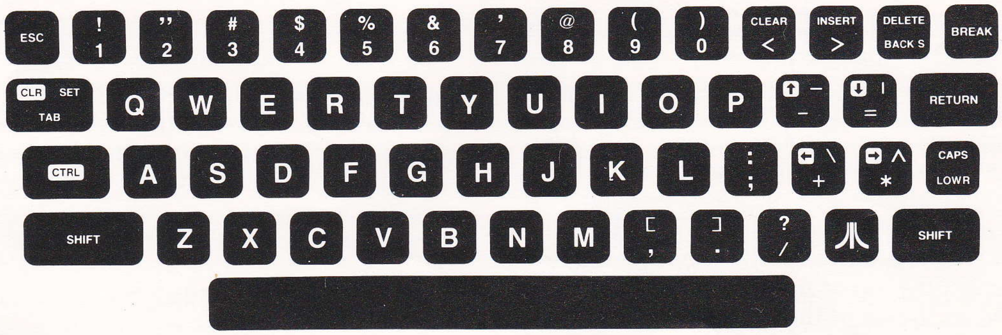
Type lower case letters, numbers, some punctuation marks, and math symbols, just as you would on a typewriter, by pushing single keys.



```
*   ATARI COMPUTER   *

Please insert a cartridge.

█
```



Pressing either of the **SHIFT** keys and holding it down while pressing another key will produce the upper case letters or the character shown on the upper half of the keytop on the actual keyboard. The diagram shows the characters that will be produced by the **SHIFT** and key combination.

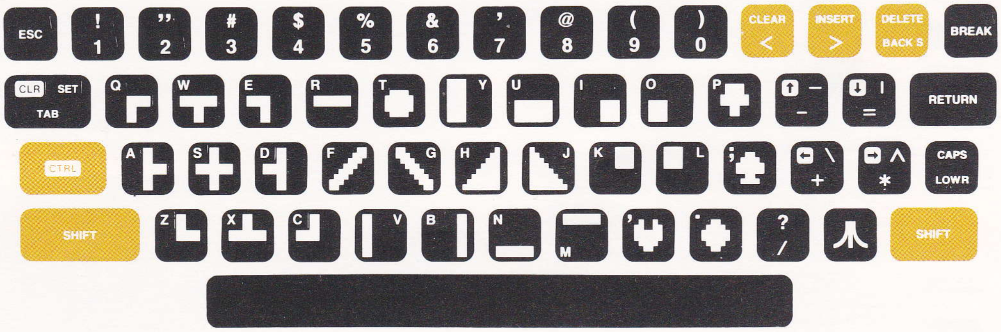
The RETURN key has three functions. First, it moves the printing mechanism to the left margin and down one line of the screen. Notice that your **ATARI 800** will do this for you automatically after forty characters even if you don't push RETURN. Forty is the greatest number of characters which will fit on a single *physical line* across the screen. However, your computer allows you to combine three lines of the display (120 characters) into a single entity called a *logical line*. Logical lines will be important when programming in BASIC language. Second, RETURN marks the end of a logical line for the computer. At times it will be convenient to push RETURN at the end of each physical line, making it coincide with each logical line. At other times the longer logical line will provide greater flexibility. Third, RETURN activates the computer. The specific action taken depends on what software is controlling the computer at the time RETURN is pushed.

SET-CLR-TAB operates much like the TAB key on a regular typewriter. SHIFT and SET-CLR-TAB clear the TAB stop under the cursor. CTRL and SET-CLR-TAB set a TAB stop at the cursor position. SET-CLR-TAB by itself spaces the cursor over to the next TAB stop. This key operates on logical lines so you can set tabs at any position up to the 120th.

The yellow keys shown on the diagram below are the display control functions. They move the cursor on the screen and modify the display. After reading the description of how each key works, stop and try it on your **ATARI 800**.

SHIFT CLEAR erases all characters on the screen and leaves the cursor unchanged.

CTRL CLEAR moves the cursor to the home position at the upper left corner of the screen.



The control key CTRL functions as a second type of shift. When it is depressed in conjunction with another key a character from a completely new set of characters appears on the screen. These "graphic" characters can be used to produce interesting pictures, designs, and graphs either without a cartridge or with the ATARI BASIC cartridge. Nothing will be displayed if you press the CTRL and SHIFT keys simultaneously.

The cursor control functions, CTRL , CTRL CTRL and CTRL move the cursor on the screen in the direction shown by the arrows on the keytops. When you move the cursor over a letter, that letter is shown in "inverse video" on the screen. When you move the cursor away from it using the cursor controls the letter is unchanged. If you put the cursoh over a character and then push another printing key, the new character will replace the one previously shown.

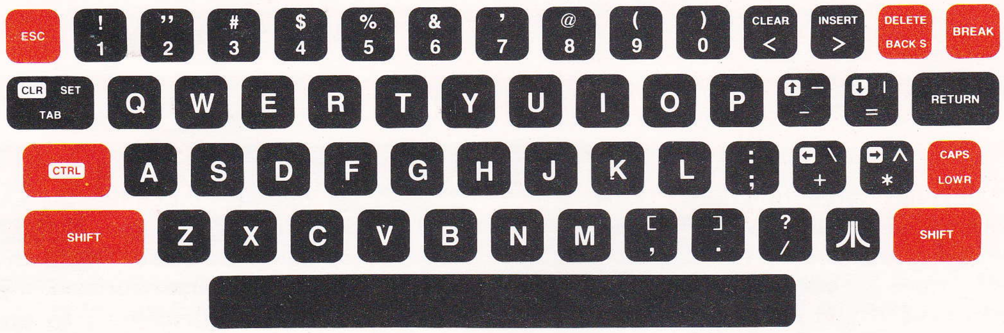
The line insert function SHIFT INSERT creates a space for a new line by moving the logical line containing the cursor and all lines below it down one line. Important: Any information on the bottom line of the screen will be lost.

The character insert function CTRL INSERT makes a space for a new character by moving the character under the cursor to the right. The rest of the line also shifts to the right. The cursor remains on the space which is now available for a new character.

DELETE BACK S erases each character as the cursor moves back one space at a time.

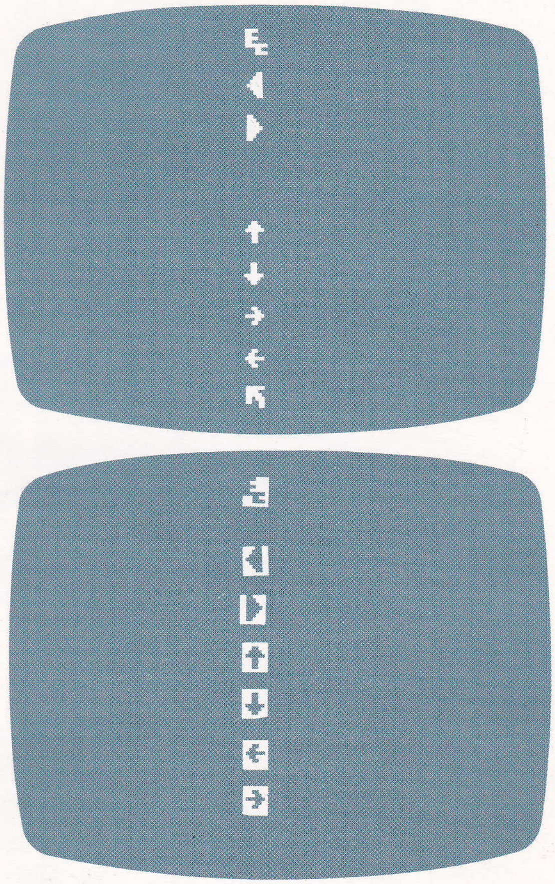
The line delete function SHIFT DELETE BACK S removes one whole logical line. If there are lines below the deleted one they will all move up one line leaving a blank line at the bottom of the screen.

The character delete fucntion CTRL DELETE BACK S erases the character under the cursor by moving all the characters to the right of the cursor one space to the left.



The keys shown in red in the diagram above have functions not found on an ordinary typewriter. The ESC key disables the cursor control movements and prints a graphic character instead. For example, press ESC, then hold down CTRL while typing DELETE BACK S. Instead of the text moving to the left you will see graphics. This function will prove very useful when you begin to program in BASIC. The other characters produced in this manner are:

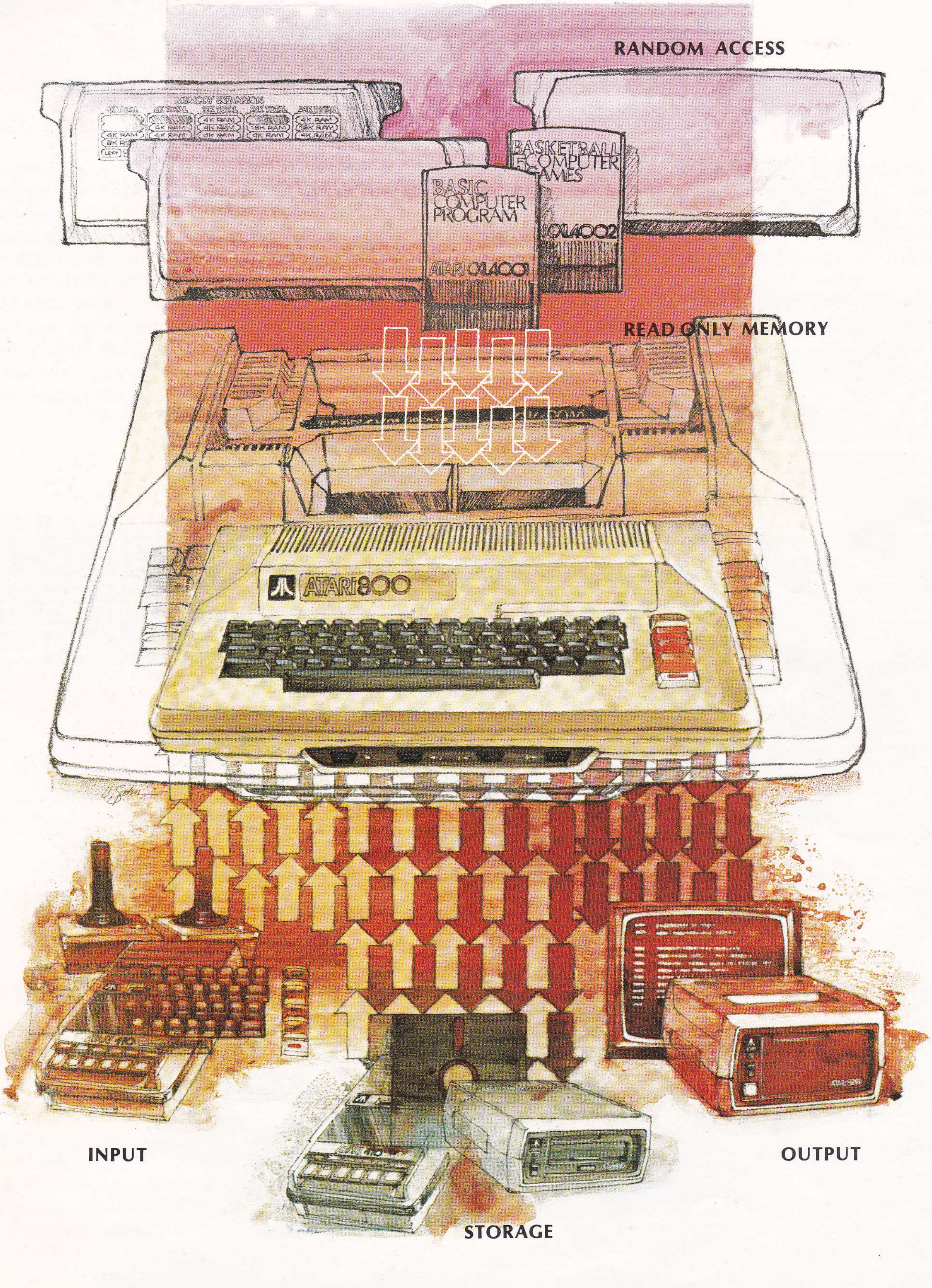
FIRST PUSH	PUSH SIMULTANEOUSLY	
	ESC	ESC
	ESC	DELETE BACK S
	ESC	SET-CLR-TAB
ESC	CRTL	↑ − _
ESC	CTRL	↓ | =
ESC	CTRL	→ ^ *
ESC	CTRL	← \ +
ESC	CTRL	CLEAR
	ESC	RETURN
ESC	CTRL	DELETE BACK S
ESC	CTRL	INSERT
ESC	SHIFT	DELETE BACK S
ESC	SHIFT	INSERT
ESC	CTRL	SET-CLR-TAB
ESC	SHIFT	SET-CLR-TAB



The CAPS/LOWR key allows you to change the usage of the SHIFT and CTRL keys. Hold down SHIFT and then press CAPS/LOWR. This will lock all the alphabet keys into upper case. However, those keys which show two or three characters on the keytops remain unchanged. Note that this is not the way the shift lock works on an ordinary typewriter. To remove this upper case lock push the CAPS/LOWR key again. Now hold down the key and press CAPS/LOWR. This will lock ALL keys into their CTRL characters. You will find this feature useful when creating pictures with the graphic characters.

The 1 key switches characters into inverse video. Press it again to go back to normal display.

BREAK interrupts the computer while it is busy following instructions. Refer to the cartridge instruction sheet for its exact function.

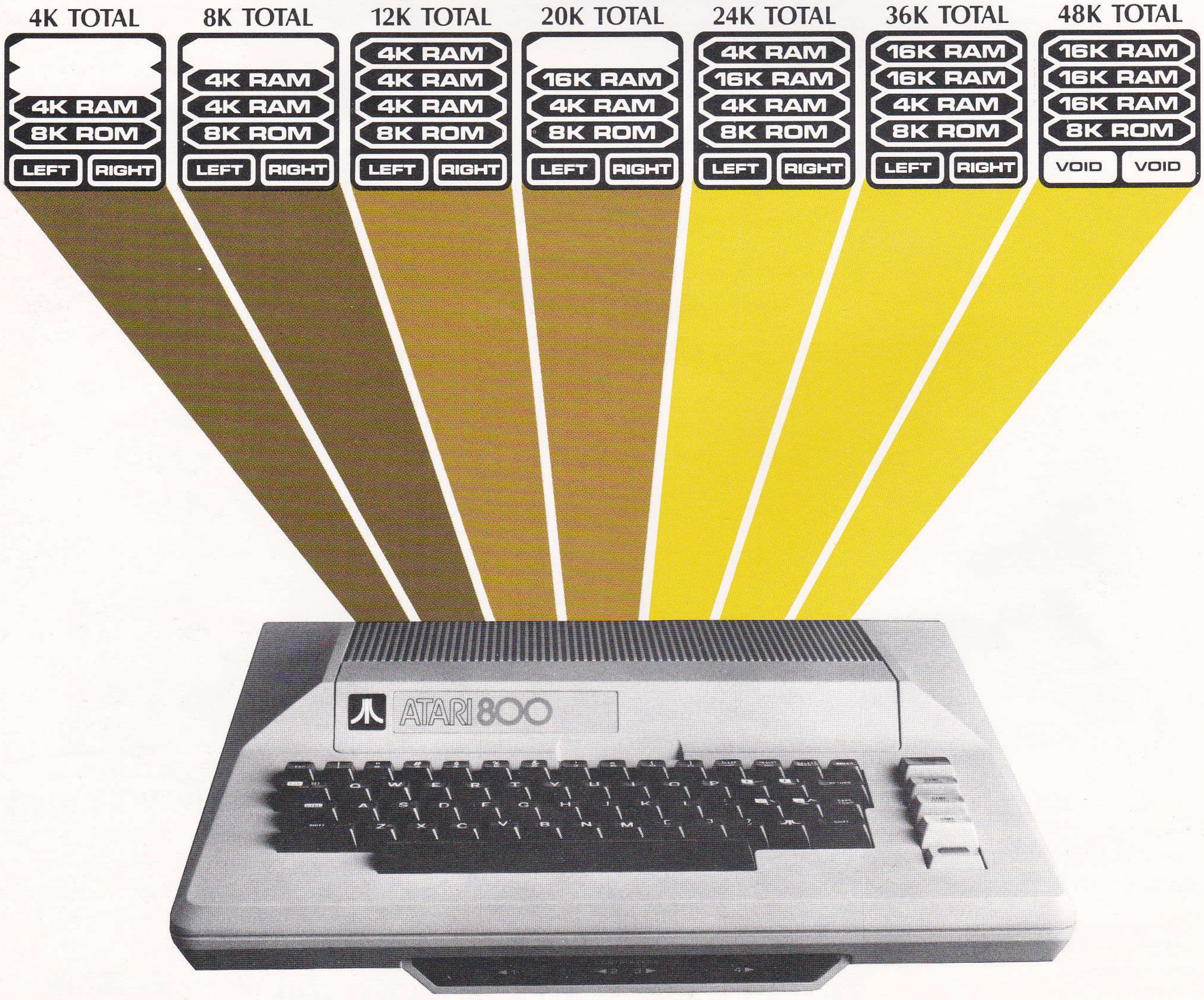


The EXPANDABLE ATARI 800

EXPANDING MEMORY

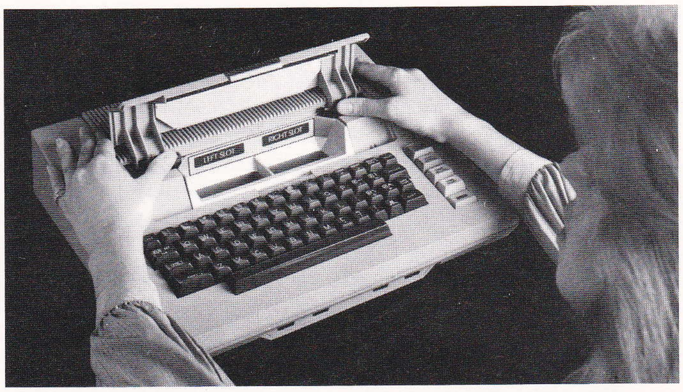
Your **ATARI 800 Basic System** comes with one **4k RAM Memory Module** installed. This module contains approximately 4,000 storage cells for your computer to use. You may want to expand the capacity of your computer system to allow you to use and write longer programs. Expansion is accomplished by inserting additional **RAM Memory Modules** into one or both of the empty sockets in the *Memory Bank*. These modules may be purchased in 4K and 16K versions from your Atari dealer.

Memory Modules must be inserted in one of the seven arrangements pictured below. Note that option 7 - 48k of memory - disables the cartridge slots and should only be used by experienced computer operators.

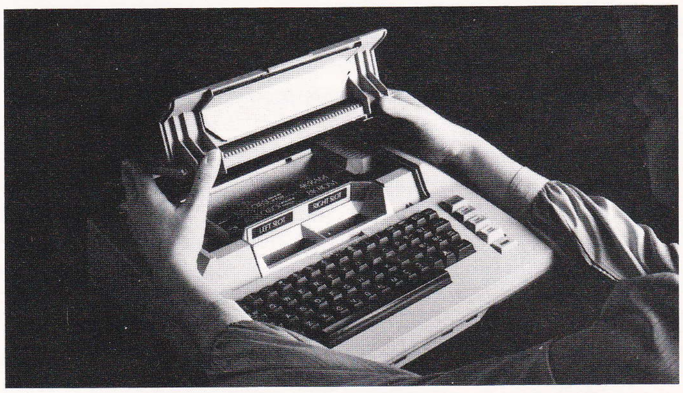


4K TOTAL	8K TOTAL	12K TOTAL	20K TOTAL	24K TOTAL	36K TOTAL	48K TOTAL
		4K RAM		4K RAM	16K RAM	16K RAM
	4K RAM	4K RAM	16K RAM	16K RAM	16K RAM	16K RAM
4K RAM	4K RAM	4K RAM	4K RAM	4K RAM	4K RAM	16K RAM
8K ROM	8K ROM	8K ROM	8K ROM	8K ROM	8K ROM	8K ROM
LEFT RIGHT	LEFT RIGHT	LEFT RIGHT	LEFT RIGHT	LEFT RIGHT	LEFT RIGHT	VOID VOID

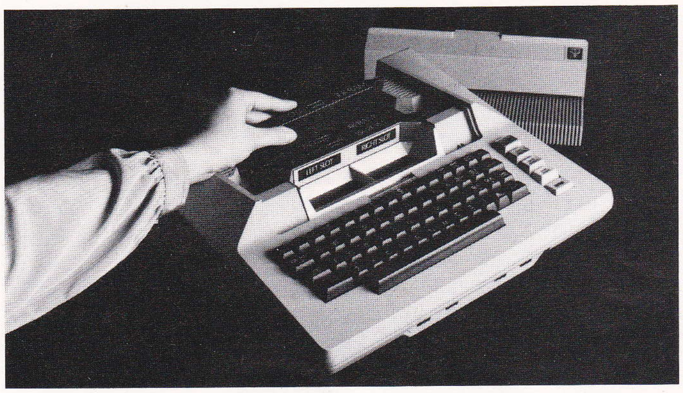
The *Memory Bank* is located under the ribbed top cover of the **ATARI 800** Console. You may install modules yourself by opening the cover and arranging the modules in one of several configurations.



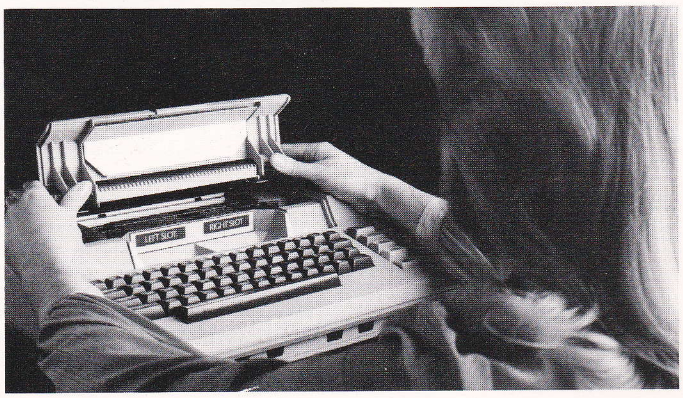
First, turn the power off by the **main power switch** and open the cartridge door. Next rotate the *two black clamps* outward as far as they will go (see illustration). Lift up the front edge of the cover slightly and slide the entire hinged door and ribbed top cover toward you.



Inside the Memory Bank you will see four module sockets, two of which are already occupied with the **Operating System 8k ROM** and one **4k RAM Memory Module**. The Operating System **8k ROM** must remain in the front socket. The other three sockets are available for expanding your computer's RAM Memory.



When inserting modules, press firmly on both sides and push straight down.



To reassemble the **ATARI 800,** slip the back of the ribbed cover into place first. The two metal tabs on the underside of the cover fit into two slots at the back edge of the console. Seat the front edge of the cover and replace the black plastic clamps by rotating them forward. Close the cartridge slot door and power up again.

THE PROGRAM RECORDER

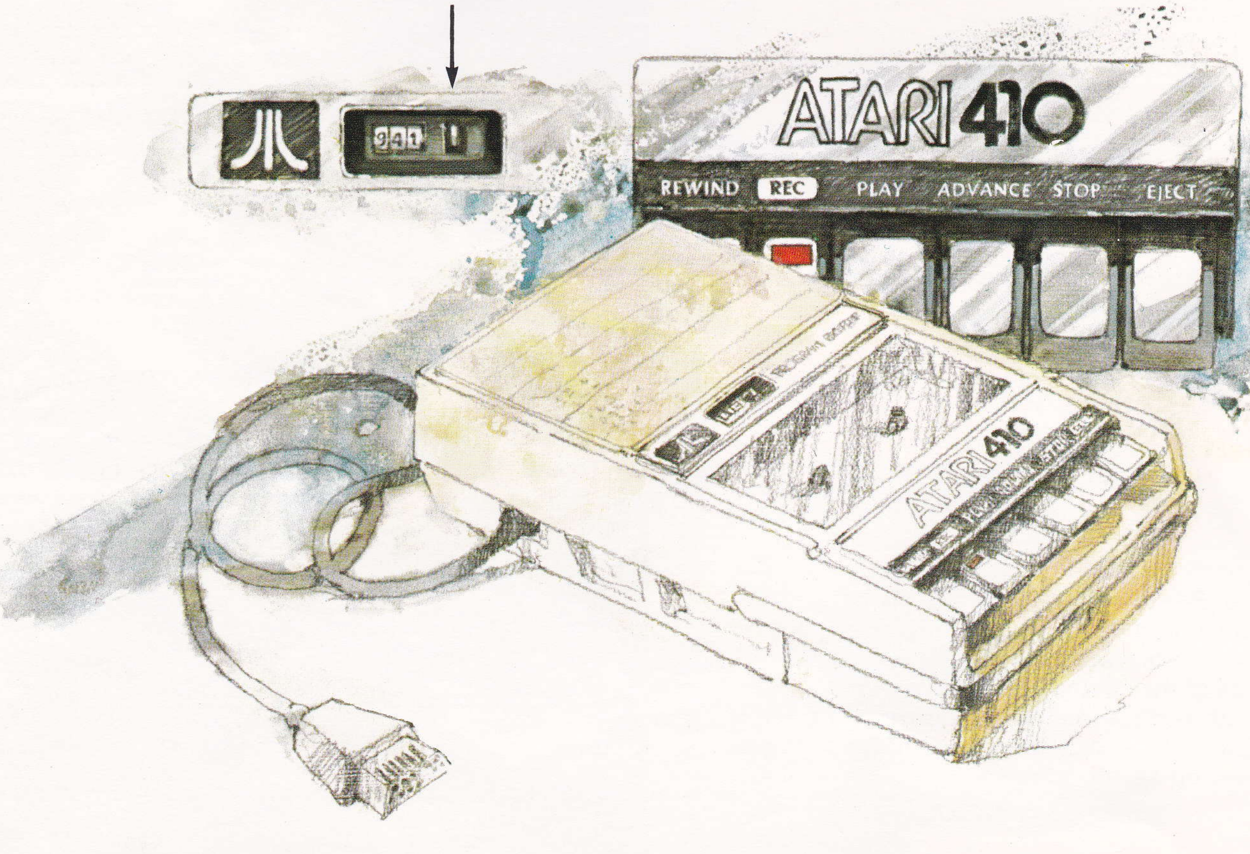
The **Atari Program Recorder** is used with a cartridge to hold blocks of software too large to be maintained in cartridge form. Programs, recorded on magnetic tape are copied by the computer from the tape into RAM memory. Once in memory, these programs can be run or modified by the user according to instructions given through the keyboard. Working in BASIC language, the user can also type his own programs into memory from the keyboard, then store them on tape for later use or modification.

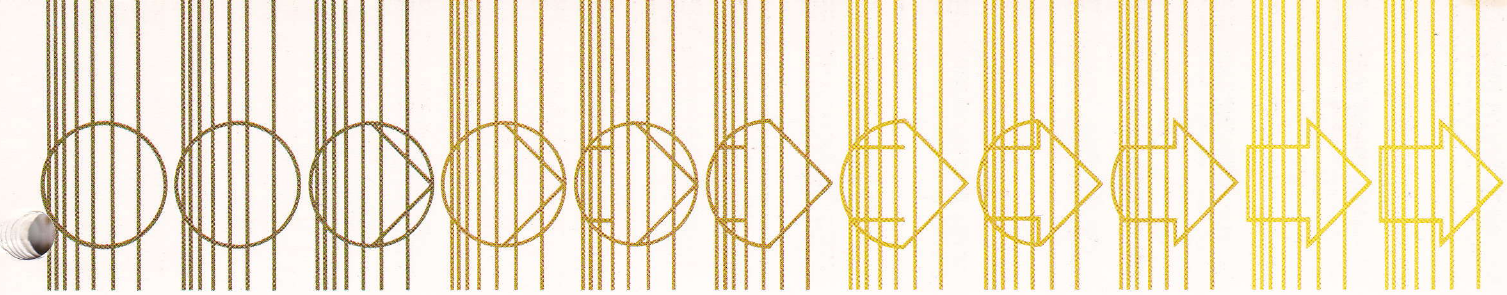
To set up your **Program Recorder, plug your Recorder Power Cord** into the jack labeled AC and into an ordinary wall socket. The other cord, permanently attached to your recorder plugs into the jack labeled PERIPHERAL in the side panel of the **ATARI 800** Console.

The **ATARI Program Recorder** is similar to an ordinary audio cassette recorder. Press **EJECT** to open the cassette door and disengage the cassette. Insert a prerecorded cassette and close the door. To insure that the tape is at the beginning, press **REWIND** and wait until the tape stops automatically. Press the **tape counter reset button** until the counter shows 000. Check your Cassette Program Instructon Sheet to determine where on the tape your program begins. If you need to, advance the tape to the correct starting number by pressing **ADVANCE** and then **STOP**. Then, press **PLAY**.

Your Cartridge Instruction Sheet will tell you what to type on the keyboard to have the computer begin to read the tape. After the program is completely loaded into the computer, the tape will stop automatically. Press **STOP** on the Program Recorder to turn off the motor, finally, type a command or press **START**, whichever is specified on your instruction sheet, to begin using the program.

TAPE COUNTER RESET BUTTON





WHY LEARN TO PROGRAM?

You may find that there are no preprogrammed cartridges or cassettes which make the computer do exactly what you want. Or, you may decide that learning to program is an important part of your career or educational plan.

Even if you have no practical use for computer programming you will find that learning to write programs in BASIC for your **ATARI 800** is an exciting and valuable experience. Programming sharpens your skill in thinking—in analyzing problems and devising step-by-step solutions. It deepens your understanding of computers in general, and no one can deny that computers are becoming a major force in modern society. A knowledge of programming makes you a more informed consumer and citizen who no longer accepts "it was the computer's fault" as an excuse for bad management.

But perhaps the most important reason to learn to program is that it is fun. Start by instructing the computer to draw pictures and to print verbal messages on the display screen. Soon you will be choosing more and more complex tasks for your **ATARI 800** and will be enjoying the challenge of designing programs which allow the computer to do your bidding.

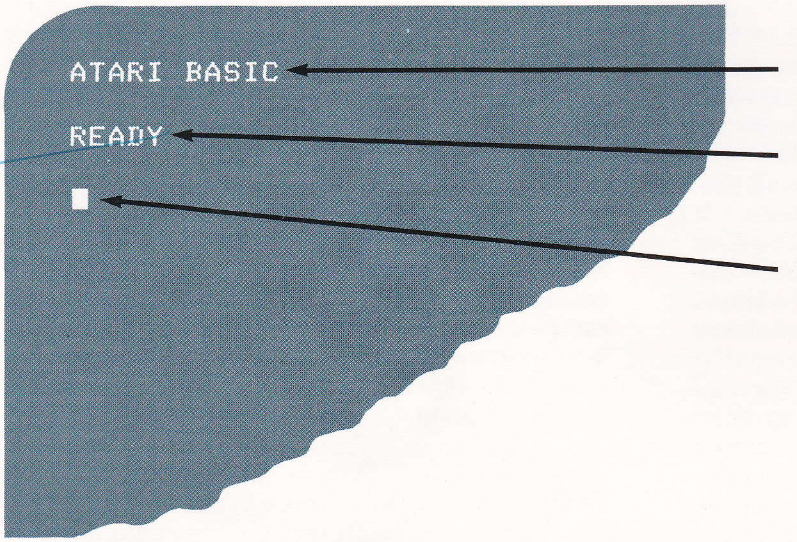
Good computer programs are usually created in three stages - *design, coding* and *debugging.* During design you choose a task for the computer and analyze it into component parts. During coding you translate these parts from their English or mathematical form into a computer language, in this case Atari BASIC. You type your coded program into the **ATARI 800** Computer. As each line is typed Atari BASIC will check it and report any mistakes in coding. After you have corrected these mistakes you can try to *RUN* your program. That is, you direct the computer to follow the set of instructions you have given it. Often you find you have made other errors. Sometimes the **ATARI 800** may succeed in running your program, yet the result is not exactly what you intended. You have made an error in design and need to go back and plan your program more carefully. At other times the computer will tell you that it can't follow your instructions as given because they contain logical or grammatical errors. You have made a second stage error in coding or typing. At the third stage—debugging— you find and correct all of your remaining errors. You continue to run and debug your program until the **ATARI 800,** under the control of your program, produces the results you desire.

Note that no computer actually "solves problems" or "answers questions." Using your design, the computer performs the instructions you have given it. It mechanically produces your "solution" to a problem or repeats your "answers" whenever you run the program.

BASIC (Beginners All-Purpose Symbolic Instruction Code) was invented so that people could learn to write programs quickly and easily. To you, the user, the BASIC language is a set of rules which tell you how to give the computer the instructions it needs to do your bidding. To the computer, BASIC is the same set of rules written in machine language. They allow it to translate your BASIC instructions into action. We call this machine language program the *BASIC Language Interpreter.* It is contained in the **Atari BASIC Language Cartridge.** The program which you write is called the *BASIC Source Program.* You enter your source program into the **ATARI 800** RAM memory by typing it on the keyboard with the **BASIC Cartridge** in the left cartridge slot. The **ATARI 800** uses the Operating System Programs (in the Operating System ROM), the BASIC Language Interpreter (in the cartridge) and your BASIC source program (typed on the keyboard or loaded from cassette or disk and stored in RAM memory) to enable it to follow your instructions.

WRITING BASIC SOURCE PROGRAMS

As an introduction to programming in Atari BASIC you will explore the use of eight BASIC codes, or commands: RUN, BREAK , RETURN , PRINT, INPUT, LIST, IF...THEN and GO TO. Start out by inserting the BASIC cartridge and Powering up. (See Sections 6 and 5.) Your initial screen display will be:



```
ATARI BASIC

READY

█
```

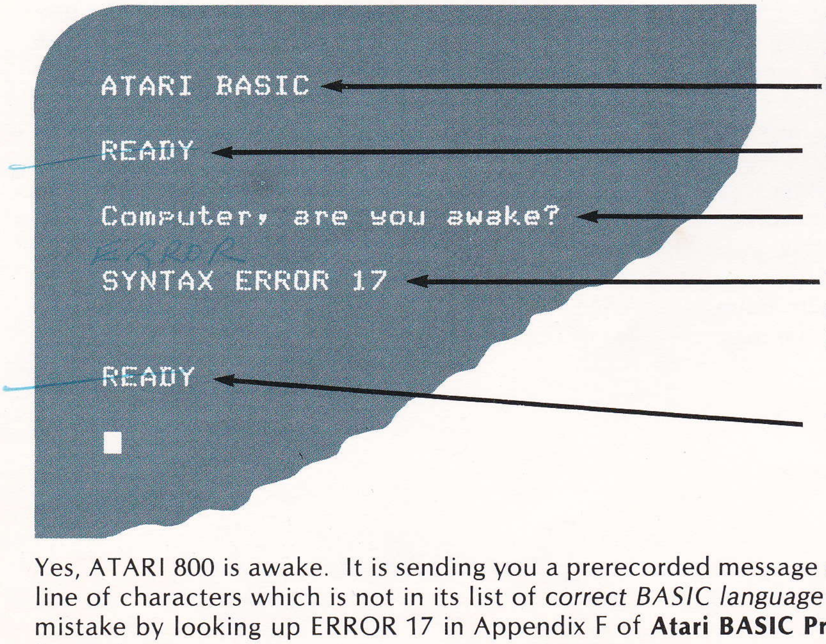
This line is called the **header.** It identifies the software currently running on the computer.

This is the *BASIC prompt.* It is a message from BASIC to you telling you that the computer is waiting for a command.

This mark is the **cursor.** It tells you where the next character will be printed.

Now type an English sentence. You might try, `Computer, are you awake?`

Then push the RETURN key. Your new display will look like this:



```
ATARI BASIC

READY

Computer, are you awake?
ERROR
SYNTAX ERROR 17

READY

█
```

This header will disappear off the top of the screen as new lines are added at the bottom.

BASIC prompt

You typed this line.

This is an ERROR MESSAGE from BASIC telling you your source program cannot be interpreted.

The BASIC prompt again.

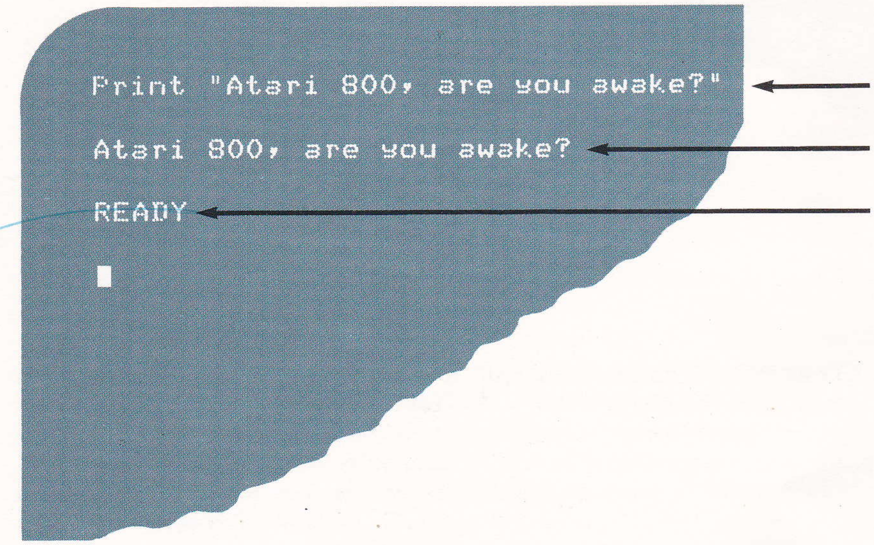
Yes, ATARI 800 is awake. It is sending you a prerecorded message saying that you have typed in, or entered, a line of characters which is not in its list of *correct BASIC language codes*. You may find out more about your mistake by looking up ERROR 17 in Appendix F of **Atari BASIC Programming Guide** or the **ATARI Reference Manual**.

Before you do that, try entering the line again with these changes. Type the line exactly as shown, including the word 'Print' and the quotation marks:

```
Print "Atari 800, are you awake?"
```

Be meticulous when you type. The computer cannot guess what you meant to include nor can it ignore extra characters. The computer will echo each character on the screen as you type. The line above is correct BASIC code.

Push RETURN and the new screen will be:



```
Print "Atari 800, are you awake?"

Atari 800, are you awake?

READY

█
```

Echo of your typing.

Here the computer followed your instruction from the line above.

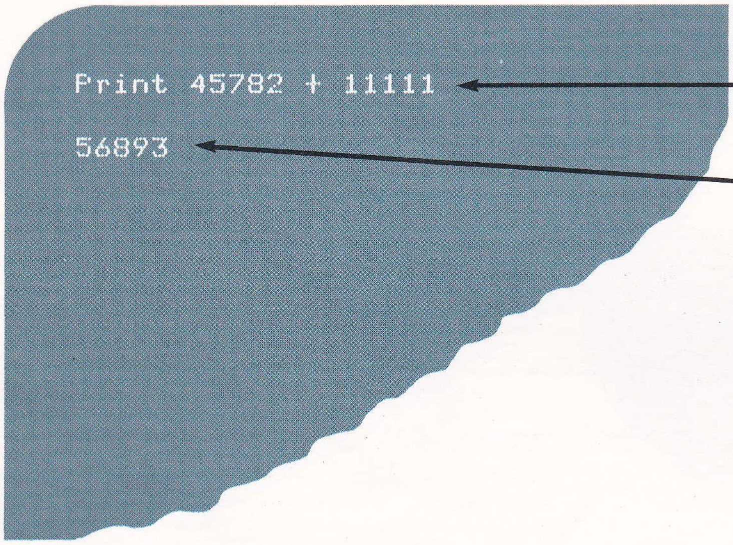
BASIC prompt

If your screen display dosen't look like the one above, check the echo of your typing to make sure you have copied every character exactly. Be sure to start your line with the cursor at the left margin. When you have found your mistake push RETURN and try again.

Play around with the Print command for a few minutes. You may put any characters you like inside the quotation marks including all the graphic and control characters described in Section 7. Experiment with the cursor controls and the logical lines of up to 120 characters. What happens if you forget to put in the quotation marks? Try it and see.

NOTE: Occasionally, pushing RETURN may fail to produce the READY prompt. If this happens, try pushing RETURN . If this doesn't work, power down and up again to restart BASIC.

You can also use your **ATARI 800** as a calculator. Use the Print command to make your **ATARI 800** evaluate arithmetic expressions. After the BASIC prompt type:



```
Print 45782 + 11111

56893
```

Don't forget Print!

Press **RETURN**.

Here's your answer!

You can enter any expression and the **ATARI 800** will do the arithmetic. Remember that BASIC language does not tell you the correct expression to use to solve number problems. You must figure out how to calculate interest payments, sales commissions, expected time of arrival, or whatever you want to know. BASIC does have a large repertoire of arithmetic, algebraic, and trigonometric functions available. But you, the programmer, must still tell the computer how to combine them to produce meaningful results. You will find a complete discussion of BASIC language arithmetic in Section 3 of Chapter 2 and Appendix L—of **Atari BASIC Programming Guide.**

The **ATARI 800** evaluates expressions in the same way you were taught to do it in elementary school. Everything inside parentheses ( ) will be done first, then exponentiation ^ . Exponentiation means multiplying a number times itself several times. For example, 5 ^ 3 means 5 times 5 times 5 and is equal to 125. You will often see this expression written 5^3. It is read "five raised to the third power". Type the exponentiation symbol on the computer by pressing **SHIFT** and **↑**. All multiplication * and division / is done next. Addition + and subtraction − are performed last. If you have forgotten that the order in which calculations are done is important, try a few problems. You will find that (4+6)/5=2 while 4+6/5=5.2 and 4/5+6=6.8.

The **ATARI 800** will print any whole number value in the range −999999999 to 999999999 as a whole number. It will express fractions and numbers outside this range in scientific notation as a decimal number between one and ten as a power of ten. For example, if the value of your expression is 1234567891 the **ATARI 800** will write it as 1.23456789E+9, which means move the decimal point nine places to the right. You read this number as 1.23456789 times ten to the ninth power (1,000,000,000).

Spend some time familiarizing yourself with the way the **ATARI 800** handles numbers. Notice again that your computer does not provide the correct method for solving a problem involving numbers. It only does the arithmetic on the numbers you give it.

You have been using the Atari BASIC Language in the *immediate mode*. This means that the **ATARI 800** follows your command as soon as you give it and then forgets about it. In *program mode,* you may give the computer about two hundred lines of instructions. Each line is stored until the RUN command is entered. Then the computer follows one instruction after another until the whole task is completed.

Although most BASIC commands can be given in immediate mode, only a few of them are useful to the beginner. RUN is almost always used in immediate mode to start at the beginning of a program. It is typed as a three letter sequence in either upper or lower case. No action occurs until you press RETURN . RETURN signals the computer to read what you have typed and to execute the entire series of instructions. Pressing the BREAK key stops the program from running and displays the BASIC prompt. BREAK is the only BASIC command which executes without being followed by RETURN . BREAK and RETURN are the only single key commands in BASIC. All others are typed as a series of letters followed by RETURN .

If you wish to enter a program into the **ATARI 800** in *program mode,* start each line at the left margin of the screen with a line number. You may number your lines with any whole numbers between one and 64000. You may enter program lines in any order. However, the computer will rearrange them and execute them starting with the smallest line number and proceeding to the next higher number until all lines have been followed. Most BASIC language programmers number their lines by tens so that they have nine line numbers available in case they wish to add new lines between the original ones.

To get started, power down and up once again. This clears any previous programs out of RAM memory. Then type this:

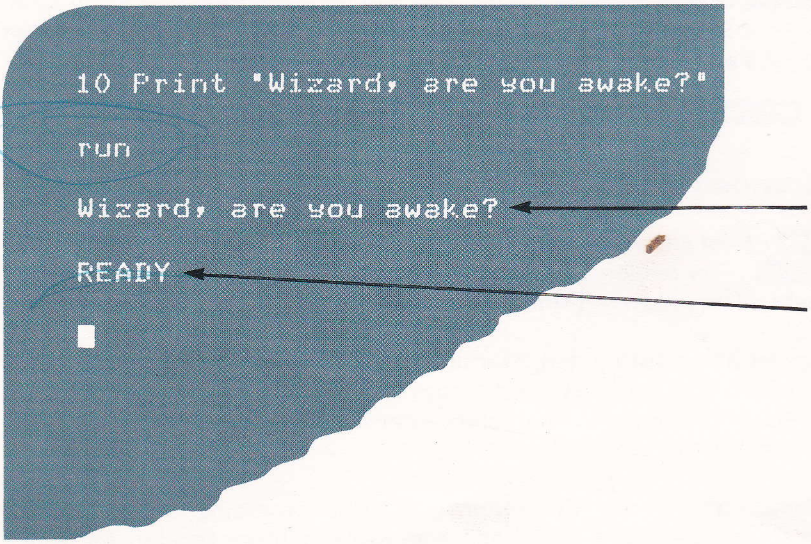
```
10 Print "Wizard, are you awake?"
```

Note that the number zero (0) is different from the letter "o" and one (1) is different from the letter "l".

If you make a mistake use the display control keys to correct your line, then press RETURN. Nothing special will happen on the display. However, the computer has stored your one line program away in its RAM memory and has made a note to itself telling which memory cells hold your line. Now type:

run RETURN

The **ATARI 800** will display:



```
10 Print "Wizard, are you awake?"

run

Wizard, are you awake?

READY

█
```

When you typed run, the computer found your stored instruction and followed it.

After completing its task, the computer signals you with the BASIC prompt.

Congratulations! You have just entered and run your first BASIC program.

To edit any line of BASIC code, use the display control keys described in Section 7. Position the cursor over the characters you wish to change and make the corrections within a logical line. When the line is correct, press RETURN. The **ATARI 800** will substitute the corrected line in place of the original one in RAM. To remove a whole line, type its line number and RETURN. For practice, replace the word "Wizard" in line 10 with your own name. Be sure to make your changes in the numbered line 10, not the line displayed after "run."

To see the program that is currently stored in RAM, type:

```
List RETURN
```

`List` is the command that makes the computer display your BASIC source program. Any time you wish to see the instructions your computer is following, push:

BREAK and then type: `List` RETURN

Note that you may use BREAK to interrupt the computer even if it is busy running a program or calculating. All other commands must be given only after RETURN is displayed. Commands may be typed in any combination of upper and lower case letters but they will always appear in a listing in upper case.

Now add to your program by entering these lines:

```
10 PRINT "Wizard, are you awake?" RETURN

20 Input A$  RETURN

30 If A$ = "Yes" then print "Your programming

career has just begun.": Goto 60  RETURN

40 If A$ = "No" then print "Technology may

pass you by...": Goto 10  RETURN

60 Print "This program has ended."  RETURN
```

Press RETURN only where indicated and don't leave out the colon (:) near the end of each line. BASIC language ignores blank spaces unless they are inside quotation marks so you can leave them out when you need to save space. However, put them in if you can so your listings will be easier for people to read.

Run this program by typing `run` and RETURN, then answer the question "Are you awake"? several times. Type your answer after the ? and then push RETURN. Try responding Yes: yes; no; and maybe. What happens when your answer is neither "Yes" or "No"? Right. The program prints the message in line 60 and then ends.

To understand what the computer is doing, get a new listing and, examine it line by line. Line 10 has the command PRINT "something." This something which you put between the quotation marks is called a string constant. String means a string of letters, numbers, and/or graphic characters—constant because it will remain the same every time you run this program.

Line 20 has the command INPUT something. This means: "Computer, wait until something is typed on the keyboard. Show whatever is typed on the screen and store it in a space in RAM memory labeled A$. When RETURN is pressed go on to the instruction with the next higher line number." A$ (read A string) is called a string variable—a string of letters, numbers, and/or graphic characters which will be different every time you run this program. Each string variable within a single program must have a different label. Many labels are available in BASIC but, for now, limit yourself to A$ through Z$.

Line 30 has IF...THEN PRINT "something": GOTO 60. When the condition following IF is met, i.e. when the computer finds the letters Y-e-s stored in the place in memory labeled A$, then the rest of the line is followed. It prints "Your programming career has just begun." The computer encounters the colon (:) next. This tells it that the logical line isn't over yet, another command is coming. The GOTO command says "skip down to line 60 and ignore the lines in between."

When the condition is not met, i.e., when the computer finds Y-E-S or any other characters in A$ then the rest of the line is ignored and the computer goes on to the next numbered line. In this case line 40.

Line 40 works the same way as line 30. If Y-E-S has been entered at line 20, then its condition isn't met either and line 40 doesn't print anything. It just goes on to the next numbered line which is 60. (Notice that there is no line 50.) Line 60 prints its message and since there are no more lines, the program ends and READY is displayed. If N-o has been entered at line 20 then all of the line 40 is executed. The message in line 40 is printed and program control jumps back to line 10. This happens to begin the program over again.

Suppose you, the all-powerful programmer, wish to insist that the user, the person who runs or uses this program, must answer "Yes" or "No" before the program ends. You will want the computer to print a message whenever something other than "Yes" or "No" has been entered. After this message the question should be repeated on the display screen. Now try to figure out what line to add to make the computer do this. Experiment with line 50 before you go on.

There are many ways this line may be written. Here is a line 50 which will do the trick:

```
50  Print "You must answer 'Yes' or 'No' to

so on.": Goto 10
```

Enter this line, list and run this revised program several times. To anticipate what the display will look like, pretend you are the computer and follow the program. Remember that computers are machines which blindly stick to their programs no matter how nonsensical the results may be. This is why the design stage of programming is so important. It is up to you, the programmer, to plan each display the user will see, to anticipate what the user will type in response to that display, and to tell the computer what to do with every response.

You can use the format shown in this first sample program to code an unlimited number of dialogue type programs. You will need to change the string constants and occasionally rearrange to GOTO's. Here is another example:

```
10 PRINT "This is the OCEANS PROGRAM."

20 PRINT "Hello, what is your name?"

30 INPUT N$

40 PRINT " O.K., "; N$ ;

50 PRINT ". Name an ocean that begins with P."

60 INPUT P$

70 IF P$="Pacific" THEN PRINT "Perfect." : GOTO 90

80 PRINT "That isn't the one we are looking for.

Try again." : GOTO 50

90 PRINT "Now name one that begins with A."

100 INPUT A$

110 IF A$="Atlantic" THEN PRINT "You're too

good,"; N$ ;". Now give someone else a turn.": GOTO 10

120 PRINT "Try again.  Remember, spelling counts!": GOTO 90
```

Don't be discouraged if you are confused at this point. Many people are inexperienced in the kind of logical thinking required to write programs. You may need to allow yourself plenty of time to become comfortable with the step-by-step nature of the computer, and the sometimes infuriating attention you must pay to detail. But remember that *inexperience* does not mean *inability*. There are a tremendous number of new and sophisticated concepts presented in this book. You will want to read it many times and keep it handy so that you can look up details when you forget them.

You have just begun to discover what you can make your **ATARI 800** do with BASIC language. Your next step is to read and study **Learn Atari BASIC.** If you are a "natural" at programming you will absorb its contents quickly and easily. If not, you should allow yourself several weeks or perhaps months to master the material presented. There is no hurry since at every stage of learning you will be creating your own programs to use and share on your **ATARI 800** Computer.

ATARI®

A Warner Communications Company