Franklin
ACE 500

User's Reference Manual

Franklin Computer

Computer

ACE 500 COMPUTER
USER'S REFERENCE MANUAL

FCC Statement

This equipment generates and uses radio frequency energy and if not installed and used properly, that is, in strict accordance with the manufacturer's instructions, may cause interference to radio and television reception. It has been type tested and found to comply with the limits for a Class B computing device in accordance with the specifications in Subpart J of Part 15 of FCC rules, which are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

Reorient the receiving antenna
Relocate the computer with respect to the receiver
Move the computer away from the receiver
Plug the computer into a different outlet so that computer and receiver are on different branch circuits.

If necessary, the user should consult the dealer or an experienced radio/television technician for additional suggestions. The user may find the following booklet prepared by the Federal Communications Commission helpful:

"How to Identify and Resolve Radio-TV Interference Problems".

This booklet is available from the U.S. Government Printing Office, Washington, DC 20402, Stock No. 004-000-00345-4.

WARNING:

This equipment has been certified to comply with the limits for a Class B computing device, pursuant to Subpart J of Part 15 of FCC Rules. Only peripherals (computer input/output devices, terminals, printers, etc.) certified to comply with the Class B limits may be attached to this computer. Operation with non-certified peripherals is likely to result in interference to radio and TV reception. In particular, components other than the Franklin Video Cable (Part Number VC 200) are not certified to comply with Class B interference limits and may result in interference.

FRANKLIN COMPUTER CORPORATION

SOFTWARE END USER LICENSE AND LIMITED WARRANTY

For information about revisions to software described in this manual, you should
periodically check with your authorized Franklin dealer.

**FRANKLIN COMPUTER
CORPORATION**

**HARDWARE LIMITED
WARRANTY**

Franklin Computer Corporation ("Franklin") warrants to the end user of this computer product that the hardware in this product will be free from defects in material and workmanship for a period of NINETY (90) DAYS from the date of original retail purchase, as evidenced by sales receipt.

On discovery of a defect, the end user must return this product (transportation charges prepaid) either to the authorized Franklin dealer from whom it was purchased, to any other authorized Franklin dealer, or directly to Franklin at the address given below. (Franklin will provide information concerning the location of authorized Franklin dealers; simply call the number given below or write to the address given below.) Each product returned must include the end user's name, address, and telephone number, as well as a brief description of the nature of the defect and a copy of the sales receipt bearing the appropriate Franklin serial number as proof of the date of the original retail purchase. Franklin will, at its option, repair or replace any product at no further charge to the end user on determination by Franklin, in its sole discretion, that the product was defective and that such defect arose within the duration of this limited warranty.

This warranty does not apply if, in the sole discretion of Franklin, the product has been damaged by accident, abuse, misuse, or misapplication, or as a result of service or modification by any party, including any dealer, other than Franklin, or if the Franklin serial number has been removed or defaced. FRANKLIN SHALL NOT BE RESPONSIBLE IN ANY WAY FOR ANY SERVICE OR MODIFICATION TO THIS PRODUCT BY ANY PARTY, INCLUDING ANY DEALER, OTHER THAN FRANKLIN.

This warranty applies only to hardware products manufactured by or for Franklin that can be identified by a proper Franklin serial number. Franklin software is warranted only pursuant to that separate writing packaged with such software by which all rights and duties in connection with such software are governed.

ALL IMPLIED WARRANTIES, INCLUDING ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, SHALL BE STRICTLY LIMITED IN DURATION TO THAT OF THE EXPRESS WARRANTY SET FORTH ABOVE, THAT IS, NINETY (90) DAYS FROM THE DATE OF ORIGINAL RETAIL PURCHASE.

THE WARRANTY AND REMEDY SET FORTH ABOVE ARE THE EXCLUSIVE REMEDY OF THE RETAIL BUYER AND END USER IN CONNECTION WITH THE MANUFACTURE, SALE, OR USE OF THIS PRODUCT

Franklin Computer Corporation
Attention: Service Department
Route 73 and Haddonfield Road
Pennsauken, New Jersey 08110
(609) 488-0666

Table Of Contents

Chapter 3 — The Keyboard

Chapter 4 — Franklin DOS 2 (Disk Operating System)

Chapter 5 — Franklin DOS 2 Commands And Programs

Chapter 6 — Introduction To Programming

Chapter 10 — Input And Output

List Of Illustrations

Chapter 1

So You've Bought A Computer

This chapter is a general introduction to the world of computers. It contains descriptions of some of the words and key concepts in use around computers with the intent of making them a little more understandable to the layman. The discussions here are non-technical but directed in practical ways to ways you'll actually use the computer.

You've taken the big step and bought a computer. It's a little scary because there's so much you don't know, but it's exciting because you know there's a lot you can do now—as soon as you learn how to use this little machine.

The first suggestion is to take it slow and easy. There's a lot to learn and it's easy when you do it step by step. Don't try to go too fast and become discouraged.

It is very important to try things. You really can't learn much just by reading or watching. Using a computer is a hands-on kind of thing. You can't break it and it won't bite (though you may be puzzled for a moment about some of the things it does), so just try everything you see and read about. This is the key to success. You don't have to type well to start, in fact a typing program on the computer might be a good way to become acquainted with computer operation.
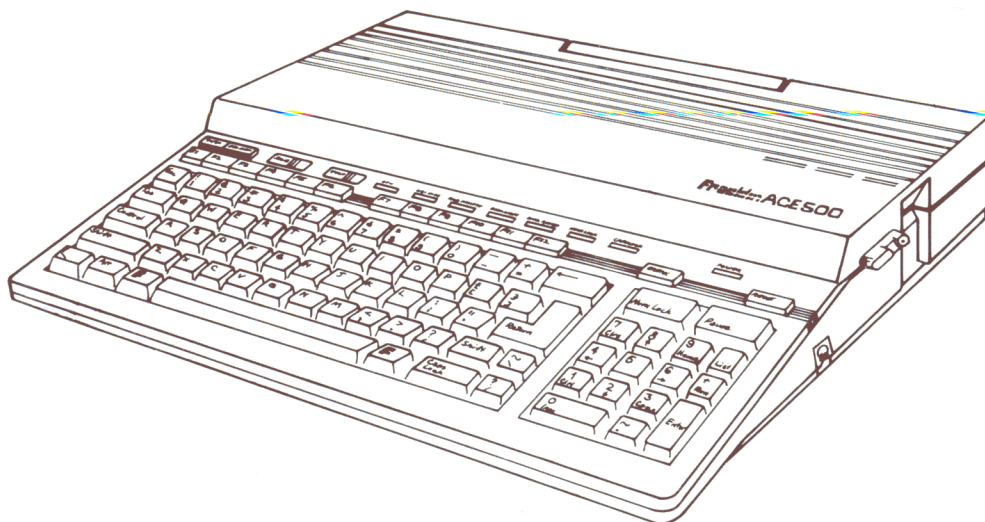
The second suggestion is to find a particular need, such as your budget or club news-letter, locate some software to fill the need, and work directly on that project until you're comfortable with the computer. The intent of this advice is to keep your work on the computer focused on one thing that can bring you immediate and visible suc-cess. If you start on something too large and too vague, like "learning to program", it is easier to become discouraged.

The third suggestion is to work together with someone or with a group, whether a course in school, a fellow student, a user's group, or a neighbor. Sometimes there's no substitute for experience. Working together is also more fun.

General Comments And Cautions

A computer isn't a fragile mystery machine. It needs reasonable care and handling, probably about the same as a stereo amplifier. The monitor used with it is like a small television set. Since they are both electrical devices, precautions during electri-cal storms, or power failures are wise and important but aren't special. Keep the computer reasonably clean and don't spill any liquids on it and you'll be fine.

The care of diskettes may be less familiar to first time users. Diskettes are used to store information, either programs to run on the computer or data used by programs. The information is stored in the magnetic coating on the surface of the diskette (like a tape recording). There are some obvious precautions:
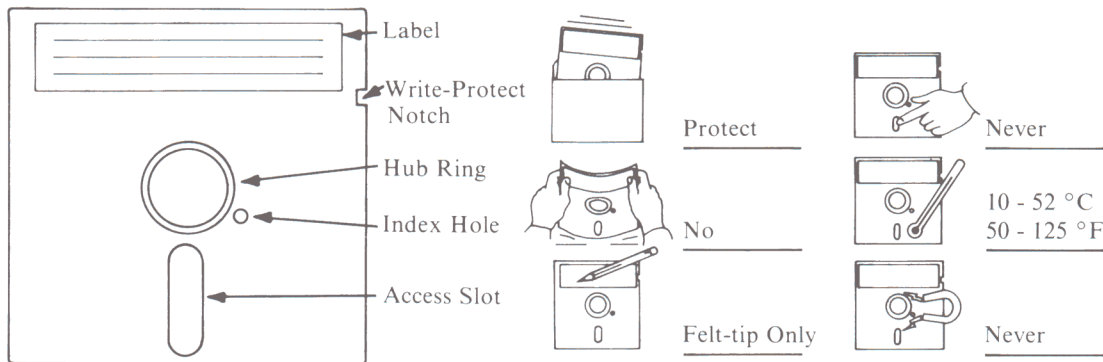
- Don't get the diskettes near magnets or magnetic fields.

- Keep the magnetic coating clean. Don't touch the diskette surface through the slots on the diskette. Always store the diskettes in their envelopes.

- Don't bend the diskettes any more than necessary or they won't turn freely in their covering. Protect them from extremes of heat or cold.

- Don't write on them with a pen or pencil as that may damage the coating on the diskette surface. Always use a felt-tip marker with a light touch.

Diskettes by themselves are fairly inexpensive. When they are full of data or programs that represent a lot of work or important information, they are extremely valuable. It makes good sense to make backup copies of important diskettes regularly and keep them in a safe place.

The notch on the side of the diskette is called the "Write-Protect" notch. When you cover it with an adhesive label, the computer can't write on that diskette or change its contents. Using the write-protect tab is an easy way to prevent accidents, such as overwriting or erasing an important program. The packet of diskettes you bought probably had a supply of write-protect tabs included with them. It's a good practice to put write-protect tabs on program diskettes you only want to read, and make frequent backup copies of the diskettes you use for recording your data.
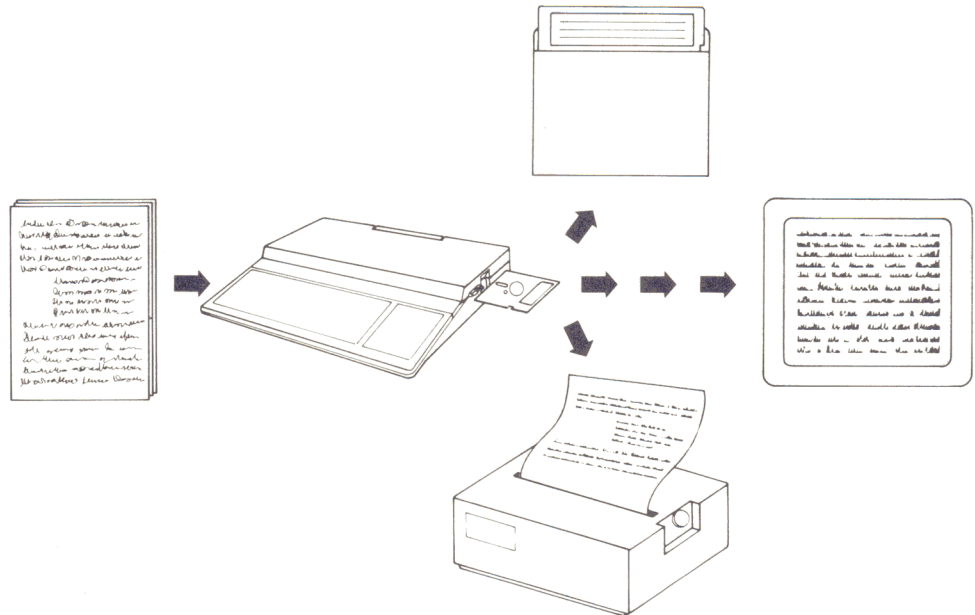
Computer Software

An important concept to understand is the difference between the computer (the physical machine) and the software (the programs) it runs. Software is a name for instructions that are interpreted and performed by the computer. The instructions may be to draw a picture of an attacking space armada, or they may be instructions to take numbers you enter in specific order and plot them on a graph.

Software includes management of what you see on the screen, including the questions the computer asks you, such as "What filename do you want?" or "Enter checkbook balance". The answers you give are entered and processed by the program in whatever way the program's designer intended.

The computer alone can be a device to write software, the process called programming, but most often it is used to operate software already written. This software is almost always brought to the computer on a diskette. The diskette stores the program and the computer reads it by means of the disk drive when you ask for the program by name. The information in the program moves from the permanent storage of the diskette to the internal memory of the computer, where it is stored until needed in the processing cycle.

Software Compatibility

Not all computers can use the same software. Sometimes the instructions in the software depend on particular features of one computer and can't be used on another that doesn't have those features. This general problem is called "software compatibility". If software is compatible, it can be used on the computer.

Fortunately, your Franklin ACE 500 computer is compatible with the Apple IIc computer, a popular and widely used computer, and vice versa. It is also compatible to a high degree with the Apple II and Apple IIe computers. Almost all software written for one computer can be used on the other. Software written for other computers, especially those outside the Franklin/Apple group, will almost never run on these computers.

Computer Memory

A computer has special places in its hardware for storing data, including program instructions or intermediate results in a mathematical calculation. This storage is called the computer's memory, but it is only temporary storage maintained by the electrical power of the computer, so if the power goes off, the memory is lost. This has important consequences for how you use your computer. The ACE 500 has 256 K bytes of internal memory (there is an optional configuration with 512 K memory). 256 K bytes of memory is roughly equivalent to 85 pages of typewritten text. A byte (pronounced "bite") is the unit used to measure memory and corresponds to the amount of memory needed to store one character, letter, numeric digit, etc.

On the other hand, memory storage on the diskette is permanent in the sense that it isn't dependent on the electrical power of the computer. It can be erased or rewritten by commands to the disk drive, but the information is recorded in the magnetic medium and lasts as long as the medium does. Each diskette has about 160 K bytes of storage on it. The exact amount of storage depends on a lot of factors, but 160 K bytes of storage is roughly equivalent to 53 pages of text.
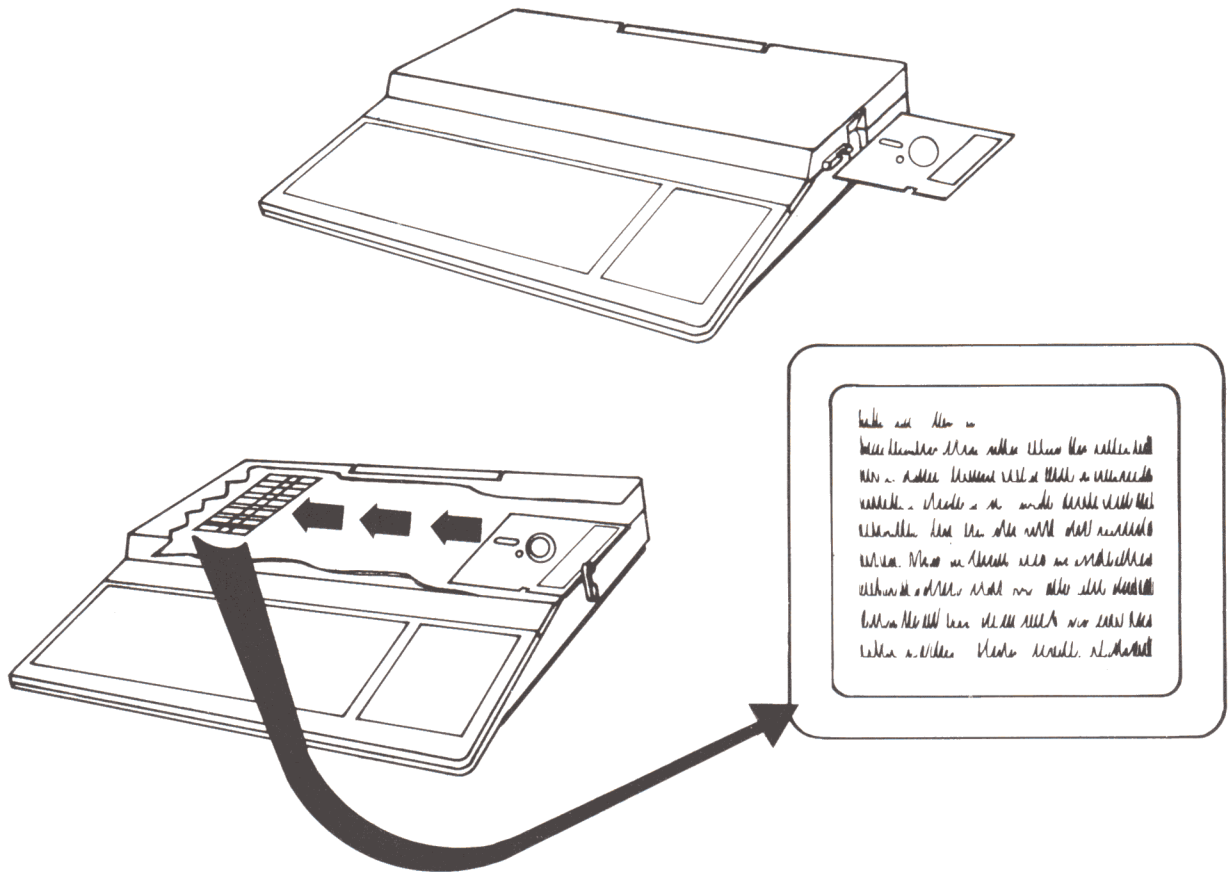
The differences between these two types of memory are important in practical ways as you operate your computer. When you write a program or enter new data for the computer in temporary memory, you must store it on the diskette before you turn off the computer or it will be lost from memory.

Using your ACE 500 Series Computer computer effectively doesn't depend on mastering all the material in this manual. Using a computer doesn't have to be an exhausting or intimidating experience. On the other hand, using this powerful tool can be very rewarding in creating solutions to problems, or beginning to understand the way these fascinating machines operate.

Each chapter contains a summary at its beginning, printed in italics like this, which describes the material in the chapter and why you might need it. If the summary suggests things you need to know, read on. The chapters are arranged with more technical material in the latter sections, so you can stop reading if you get stuck. The ACE 500 User's Manual describes features of the computer that you'll use all the time, and also features you may never need. The manual is a continuing source of reference information for you.

This manual contains simple instructions to help you get started and connect the peripheral devices to the computer. With a few simple instructions so you can use the Disk Operating System, you can use any software directly from a diskette.

This manual describes the way the ACE 500 Series Computer works. If you're interested in the computer itself, or writing programs that utilize its features, this manual will give you a good start. If you use your ACE 500 for application programs, such as word processing, accounting, or education, you'll probably learn a lot about those programs and less about the ACE 500 Computer itself, except how to prepare diskettes for the computer to use, copying files, etc. This manual is designed to let you find the material you want quickly, and then go on to something else.

Read, or scan, the manual so you'll know what things are here. If you need those things, now or later, you'll be able to find them quickly. Use the Table Of Contents and the Index to help you find things you need. Keywords and simple definitions of computer words are given in the Glossary.

Other Manuals And Sources Of Information

The *ACE 500 Technical Reference Manual* contains more detailed information about how the ACE 500 Series Computer works, including more detailed lists of Franklin DOS 2 and Franklin BASIC commands than occur in this manual, memory maps, and video display techniques. This reference manual is designed for programmers and experienced computer users, though it can be helpful to anyone interested in applying the power of the ACE 500 computer.

When you use other books, such as a BASIC tutorial, the list of BASIC commands found in Chapter 7 will help you customize your lessons to Franklin BASIC. Use this

manual as a reference to find out what applies to computers in general and to your ACE 500 in particular.

Since the Franklin ACE 500 Computer is designed to be extensively compatible with the Apple IIc and other Apple II series products, some Apple reference materials might be of interest to experienced computer users. You should use these materials carefully, and in conjunction with the ACE 500 Series Computer Manuals, so you can tell which parts of the material represent similarities and which are differences from the ACE 500 operating characteristics.

Computer user groups are gatherings of people of varied backgrounds with varied interests with the common aim of wanting to learn more about their computers. They are always a good source of practical information, whether answering questions or locating software to do a particular job, or evaluating peripheral equipment, or acquiring public domain programs. Your Franklin dealer can probably help you locate a user's group in your area.

Another popular source of communication among computer users is the electronic bulletin board. If you have a modem (a device that connects your computer to a telephone) you can call up a bulletin board, which is usually operated by an experienced computer user, and ask (or answer) questions about just about anything, or just follow someone else's discussions.

Chapter 2

Getting Started

This chapter is for all users. It contains a general description of the ACE 500 computer. It also describes a simple procedure for setting up the computer and getting started. There are pictures of the back panel and descriptions of the connections needed for attaching other devices to the computer.

The chapter also contains a simple description of starting the computer from the diskette and suggestions for using the disk drive.

Congratulations on your purchase of an ACE 500 personal computer. The ACE 500 Computer is an excellent choice because it is easy to operate and has all the power and capability you need for personal, business, home, and educational computing applications.

Features Of The ACE 500 Computer

Your Franklin ACE 500 Computer incorporates the advanced design features of other Franklin computers as well as extensive compatibility with one of the most widely available computers available today, the Apple IIc. It also includes a number
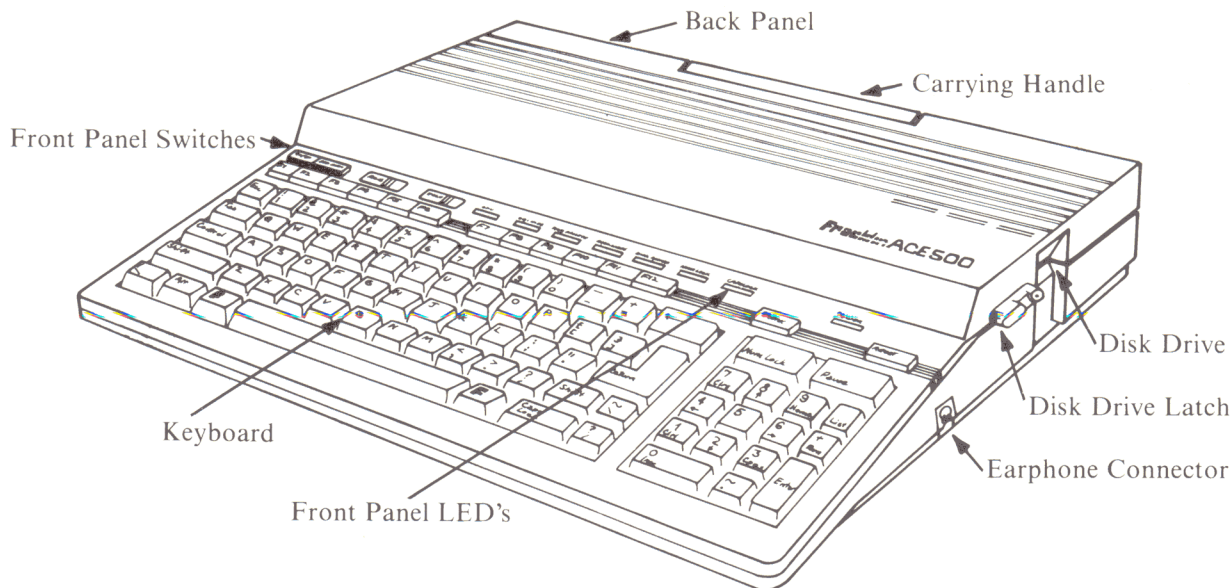
of additional features that make it an exceptional value in microcomputers. It is an excellent choice for ease of use, the variety of software available, and versatility for applications in the home, education, and small business. Although the ACE 500 Computer is simple enough to be a first computer for many people, it is still powerful enough to meet the advanced needs of experienced users.

The following list is a brief discussion of some of the features of the Franklin ACE 500 Computer. It is intended to help you become acquainted with your new computer. The features presented here are things that set the ACE 500 Computer apart from other computers and may be of particular interest for that reason.

- The ACE 500 Computer has many of the enhanced features of the Apple IIc computer, including color video and a mouse interface. It also includes the Double High Resolution Display feature of the Apple IIc which allows display of 560 by 192 pixels on the screen. The ACE 500 Computer has an expanded keyboard, including a numeric keypad and the open and closed F keys needed for many Apple software programs, and has, in addition, extra features such as 256 K of memory (optional 512 K), 12 programmable function keys, and switchable command keys on the numeric keypad.

- The ACE 500 Computer has advanced software, including Franklin's own Disk Operating System, Franklin DOS 2, and BASIC programming language, Franklin BASIC. This software incorporates features of both Apple's DOS 3.3 and ProDOS operating systems, and Applesoft BASIC, resulting in a high degree of compatibility for programs written for the Apple II, IIe, and IIc computers. In addition, Franklin's DOS 2 includes many improvements. There is also enhanced video software for 80 column video and graphics display and printer control software for four popular printers and their compatible printers. Several utility programs for diskette, memory, and file management are included on the Franklin DOS 2 diskette to help you use the ACE 500 Computer efficiently.

- The ACE 500 Computer uses the new 65SC02 microprocessor for its main processing unit and 256 K (optional configuration with 512 K) of memory. These components offer extra power for general computer operation and a number of additional benefits for advanced programmers.

- The ACE 500 Computer has sufficient built-in memory to run software requiring add-on memory cards. The ACE 500 computer RAM memory configuration emulates the RAM Works memory card.

- The ACE 500 Computer has a compact disk drive built into the computer housing for convenience and security. This special disk drive, together with Franklin's DOS 2 operating system, have the capability of using diskettes in 40 track format in addition to those in standard Apple 35 track format. This option allows you to increase your diskette storage capacity by 15% while still maintaining the ability to use standard Apple diskettes. The ACE 500 has the capability to accept a second disk drive with an external connector.

- The ACE 500 Computer has 12 programmable function keys that can make computer operation easier and more efficient. The keys are easily programmed to your own specifications by using a simple Franklin BASIC command. The switchable command keys on the numeric keypad make programming faster and easier. The integrated keyboard uses an advanced ergonomic design with many features and enhancements found only on more expensive computers.

- The ACE 500 Computer has built-in interface hardware and software for four of the most common printers, including graphics printing capability. The computer also has software controls for many printer functions and features included for easy user control of printing, including printer screen dump software.

- The ACE 500 Computer has built-in connectors for a mouse and/or a joystick so you can run programs that use these pointing and graphics input devices.

- The ACE 500 Computer has a built-in connector for a serial port, which could be used for any serial communications device, such as a modem to connect the computer to a telephone network. It also has built-in software for controlling and configuring the serial port operation.

- The ACE 500 Computer has a built-in connector for an RGB monitor.

- The ACE 500 Computer has a built-in connector for an external disk drive, in either the 5 1/4″ or 3 1/2″ diskette format (though not the 3 1/2″ Apple Unidisk drive which uses a different disk drive interface).

Getting Acquainted With The ACE 500 Computer

If you haven't done so already, take the computer out of its packing carton. Before you explore too far, read a little of the manual to get an idea of how things are supposed to work.

There are some important items packed with the computer, like the Warranty Card, the power cord and transformer, the video cable, this manual, and the Franklin

DOS 2 diskette. Take a minute to fill out the card and, for the moment, find a safe place for the diskette.

Unpack the computer. Save the packing materials so you can transport or store your computer safely if the need arises. Place the computer on your work surface carefully. There should be enough room to turn the computer and monitor to make the necessary connections.

Remove the cardboard head protector from the slot of the disk drive on the side of the computer. This protects the drive mechanism while the computer is being transported.
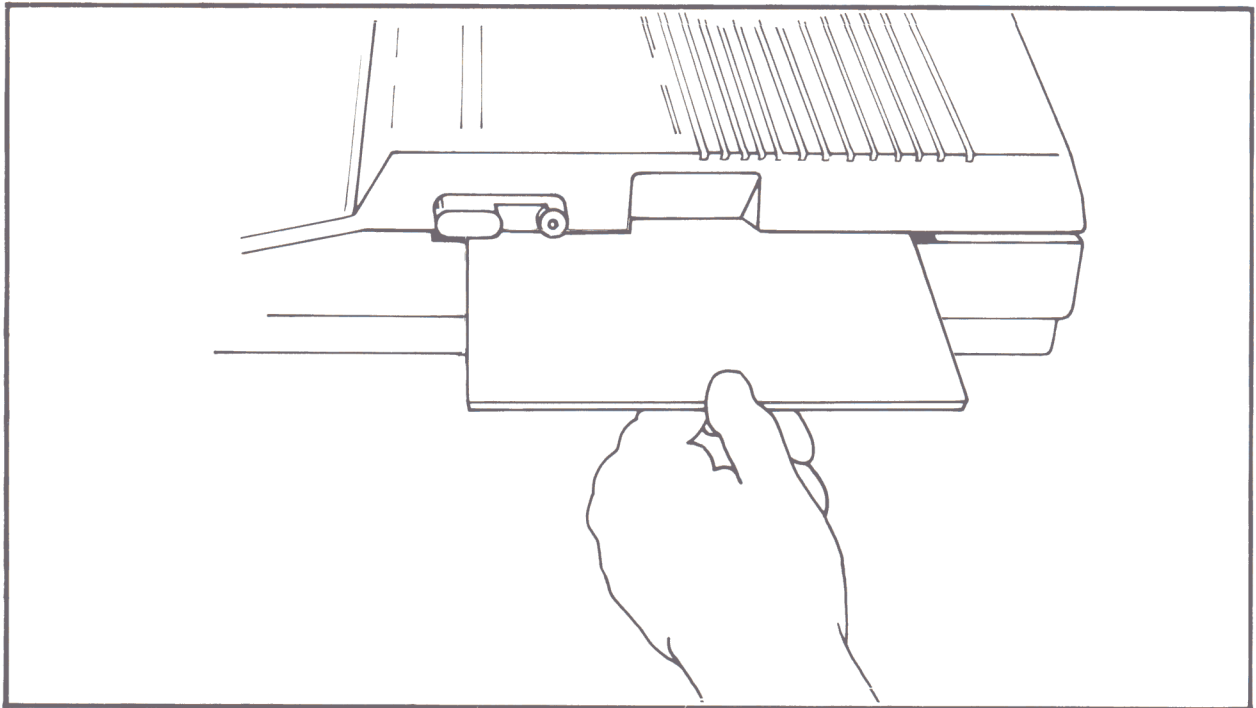
Take a minute and look at the computer and especially the front and back panels to orient yourself to some of the computer's features.

Front Panel Switches

There are a number of switches and special buttons above the keyboard that have important functions. If you don't understand what these functions are, don't be too concerned, the computer will operate well without any adjustments here.
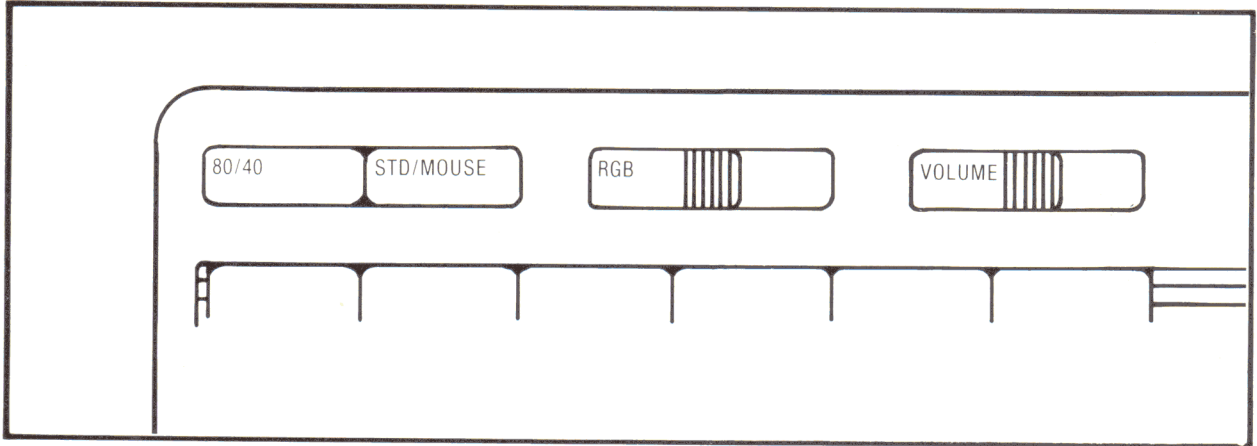
- The 80/40 push button is an indicator switch for certain application software to show your preference for a display screen with 40 columns or one with 80 columns. Not all software uses this indicator. The up position is for 80 columns and the down position is for 40 column display. (You can issue commands to change from 40 to 80 columns with the function keys F5 and F6, but you'll read more about that later.)

- The Std/Mouse push button selects whether the computer uses the normal set of characters (letters, numbers, etc.) or the mouse character set which has special symbols if the software you're using calls for it. The mouse character set is generally only used with special software and has to be activated internally by software as well as with this switch.

- The RGB slide switch has four positions which set the text color (white, green, amber, or blue) used on the black background. This switch functions only if an RGB monitor is connected. The actual colors depend on the match of the monitor's palette and the Apple/IBM palette selected with OPTION SWITCH #4. If there is a mismatch, the green, amber, and blue will be other colors.

- The slide switch marked "VOL" is an adjustment for the volume of the computer's speaker. The speaker makes a beep when there is an important event for you to notice, like the computer is ready to use, or you've entered an incorrect command. Application programs frequently use the speaker as accompaniment as you zap cosmic invaders.

Front Panel LED Indicator Lights

The LED (Light Emitting Diode) indicator lights on the front panel of the computer are green and red and show:

| CPU | DBL HI RES | RGB PALETTE | DISK WRITE | DISK ACTIVITY | CAPS LOCK | NUM LOCK | POWER |
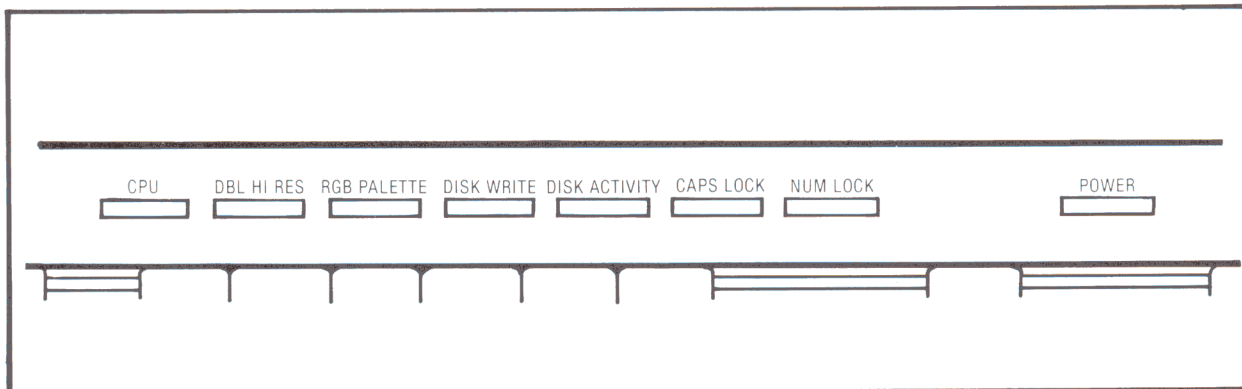
#1—The CPU LED registers any I/O activity of the computer CPU. The CPU is the Central Processing Unit and can be thought of as the computer's brain or calculator. The CPU's activity includes keyboard, disk drive, and printer activity.

#2—Double hires graphics is in use. This indicator is active any time the 560 x 192 pixel screen display is used.

#3—RGB Palette indicates whether the Apple or IBM palette of RGB colors is in use. The LED is On if the IBM color palette is selected and Off if the computer is using the Apple color palette.

#4—Diskette is being written to (green light)
#5—Disk drive is active. The disk drive lights show the activity of the disk drive. The red light is on whenever there is any disk drive activity, reading, writing, access, etc. The green light is on whenever the disk drive is writing to the diskette (which means that both lights are on).

#6—Caps Lock is On
#7—Num Lock is On. The Caps Lock and Num Lock LEDs show you the state of the keyboard selections made when you press those keys.

#8—Power is On

These lights show the state of activity of your computer and can be helpful to you.

Back Panel Connectors

Now look at the back panel and compare it to the illustrations to see what the connectors are used for.
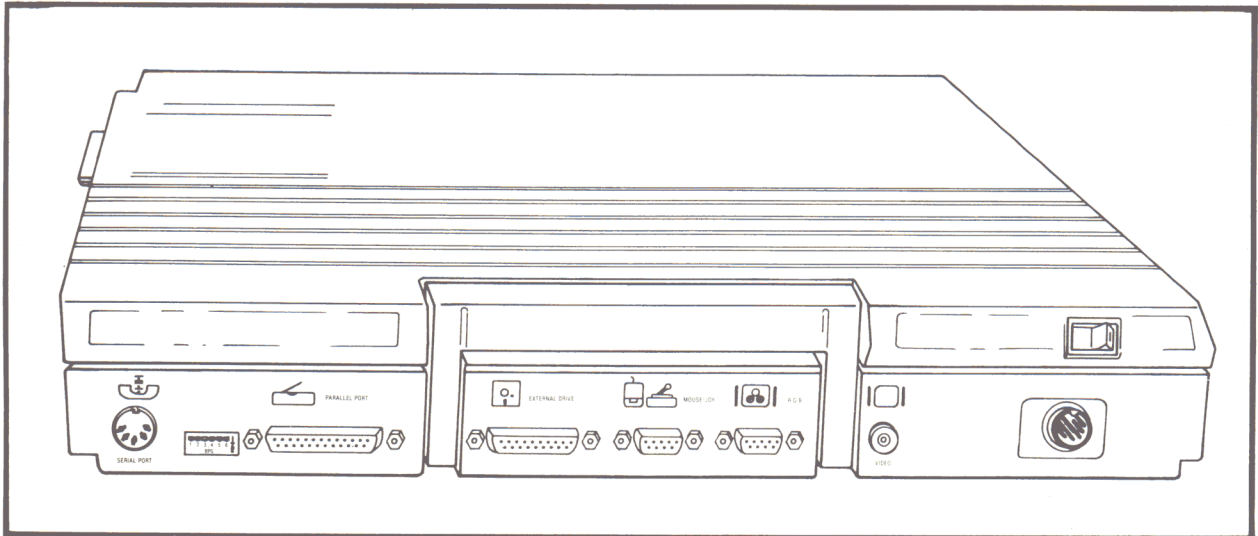
- The power switch and the power cord connector are built into the back of the ACE 500 Computer. The power supply is part of the power cord which plugs into the wall. The cord connects to the computer with a seven pin DIN plug which has a threaded sleeve and mating ring so you can make the connection secure.

- The Monitor connector is used to connect the video display device, a video monitor in either monochrome or composite color.

- The nine pin connectors are for an RGB color monitor, and either a joystick or a mouse. The symbols mark which connector is for which device.

- The nineteen pin connector is for an optional second disk drive. You can use either a 5 1/4″ or 3 1/2″ disk drive for the second drive. Compatible disk drives are available from a number of manufacturers (except the 3 1/2″ Apple Unidisk drive which uses a different disk drive interface) but be sure to specify that they must be Franklin ACE 500 or Apple IIc compatible.

- The 25 pin connector is for the parallel printer. The printer cable included with the ACE 500 computer has a standard DB25 plug on the computer end and a Centronics 36 pin connector on the printer end. This will work with any standard parallel printer. If you're printing graphics, refer to Chapter 10, *Input/Output*, for graphics printers supported by the special output software of the ACE 500 computer.

- The six OPTION SWITCHES are very important settings for several options for the computer's operation. You'll read more about them later, but just note that these are important switches and should be used carefully. Set or change them only when necessary.

- The five pin DIN plug is for a serial port, which can be used for a modem or other serial device.

- The 3.5 mm monaural jack for the headphone connector, located on the side of the computer near the disk drive.

Setting Up Your ACE 500 Computer

Connecting The Power Supply
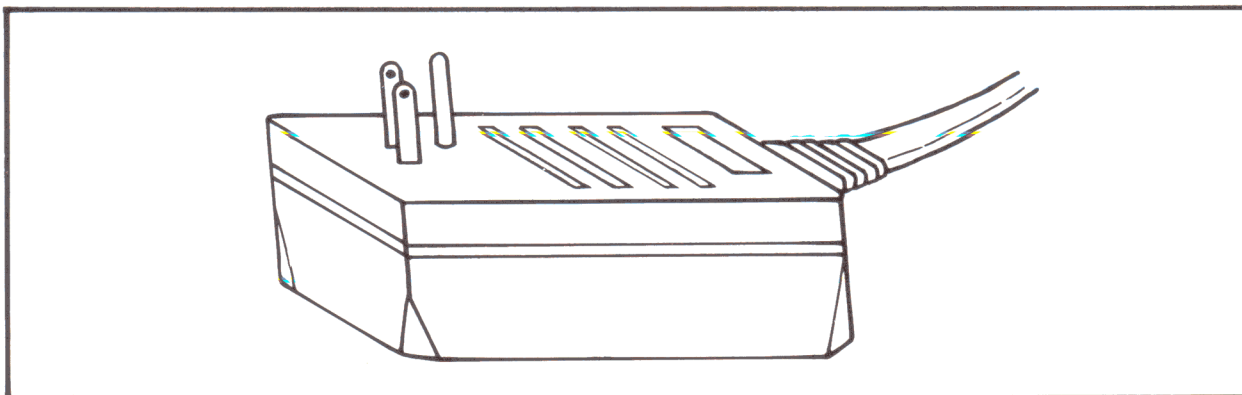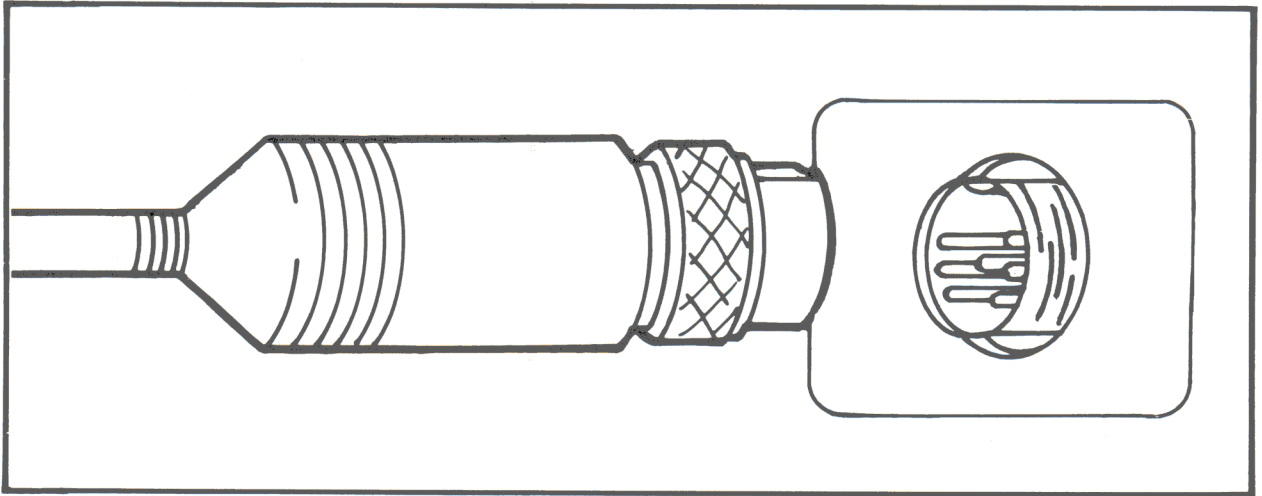
The first thing to do is connect the power cord. The external power supply transformer plugs into a grounded wall outlet. Now take the cord from the power supply and plug it into the back panel of the computer marked Power below the On/Off switch. The connector is the only one of its type so you won't get confused. The connector has a threaded sleeve and the mating ring so you can make the connection secure and not unplug the computer accidentally.

Don't turn on the computer just yet.

Although the computer uses relatively little electrical power (about the same as a small desk lamp), it is sensitive to changes in the power it receives. You won't get the best results from your computer if you use it in the same circuit with appliances (like air conditioners or electric motors) that make heavy or periodic demands on the electrical power.

Connecting The Video Monitor

Now connect the video monitor to the computer. If you're using a monochrome monitor, or a composite color monitor, take the cable provided with the ACE 500 Computer and plug it into the connector on the back panel. Connect the other end to the monitor itself.

Make sure the video monitor is turned off, then plug in the monitor's power cord.
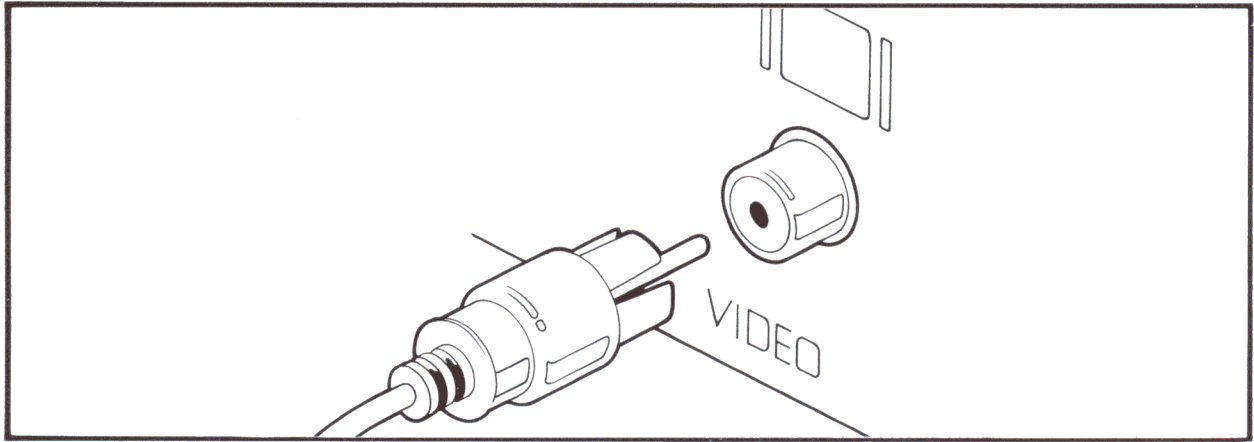
Connecting An RGB Monitor

The special signal needed for the RGB monitor is available from the DB9 connector marked with the screen and three dots (for RGB) on the back panel of the computer. Plug in the connecting cable, which should be supplied with the monitor, then set the RGB switches in the OPTION SWITCH group before you turn the RGB monitor on.

Choosing The RGB Monitor Configuration

RGB monitor manufacturers use different conventions for their RGB signal requirements. In general, there are two configurations for monitors, the Apple and the IBM. The main difference between the two types is that the Apple configuration uses a composite signal for the horizontal and vertical synchronization, and the IBM uses separate signals for them. The two types also use a different labelling system (called color palette) for colors. Some RGB monitors have a switch which permits selection of the Apple or IBM modes. If you have a choice, use the Apple mode.

Consult the monitor's manual to see which settings are required for the computer, and whether the Apple or the IBM configuration meets the needs of your RGB monitor. The descriptions below should help you choose the settings you need without too much trouble.

The OPTION SWITCH group has a set of 4 switches which select the RGB configuration used by the computer.

Option Switches

SWITCH #2 — This switch governs the polarity of the line that carries both the Composite Synchronization signal for Apple mode monitors and the Horizontal Synchronization signal for IBM mode monitors. The switch position Down is for Negative signals and the Up position is for Positive signals.

Check your RGB monitor manual to see which polarity you need, either for the composite or the horizontal sync signals for Apple or IBM type monitors, respectively.

SWITCH #3 — This switch governs the polarity of the line that carries the Vertical Sync Signal for IBM mode monitors. The Down position selects the Negative Vertical Sync signal. The Up position selects Positive Vertical Sync. Positive Vertical Sync is generally the more common setting.

SWITCH #4 — Apple/IBM Mode   This switch selects the configuration of the color matrix, or color palette, that the computer uses, either according to the Apple or the IBM formats. The Down position selects the IBM color palette, Up selects the Apple color matrix mode.

If your RGB monitor says "Apple compatible" or "IBM compatible", or has some other indication that it is designed to work with one of those computers, try the configuration suggested for that computer. The palette assignments for colors is somewhat arbitrary and may not matter until you use some software that produces red grass and orange sky. The four-position RGB slide switch on the front panel should give white, green, amber, and blue text in its four settings. If these colors aren't accurate, the wrong color palette has been selected.

SWITCH #5 — Monochrome Double High Resolution   This switch selects a single color (the text color selected with the RGB slide switch on the front panel) on a black background when the computer is used with a double high resolution screen. This is generally only used with specific software applications, such as Computer Assisted Design, where the four bit positions on the screen needed for RGB color are traded for maximum screen resolution.

The Down position of switch #5 selects monochrome display when the computer uses Double High Resolution display mode.

If you're using an RGB monitor, the Down position selects the monochrome selection and the Up position selects RGB. The monochrome screen in double high resolution mode offers increased visible resolution needed for some applications, such as

Computer Aided Design. The monochrome selection (the Down position) also offers better readability for double high resolution screens, especially those that use mixed text and graphics. You may need to try this switch in both settings for different software programs to see which screen presentation you prefer.

Consult the manual of your RGB monitor for the kinds of signals required for your monitor, or ask your dealer or the manufacturer for advice. The vertical sync is set incorrectly if the monitor screen rolls vertically, and the horizontal sync is wrong if the screen appears diagonally out of alignment. The appearance of these problems is familiar from adjusting your television set.

Turning On The Computer   Turn on the video monitor first, then turn on the computer using the switch on the back panel with nothing in the disk drive. The Power LED on the right side of the front panel of the computer should go on and the Disk Activity LED should go on as the computer tries to read the diskette. You should see a title line:

<p align="center">ACE 500 Version X.x</p>

on the screen and the disk drive should go on. Press <CTRL><RESET> to stop the disk drive, and you'll see a single character (a] , which is called a prompt) and a flashing block, called a cursor on the video screen. Turn off the computer and the monitor.
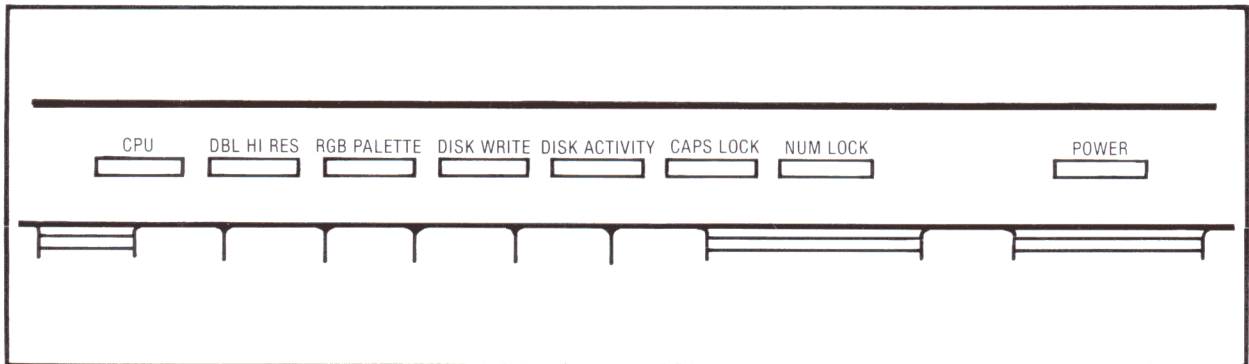
For now, that's all you need to do. The computer is getting power and is connected to the video display, and that was the purpose of this test. If the lights didn't go on, check the connections of the power cords and the power outlets. Check the brightness and contrast adjustments on the monitor. If it still doesn't work, check with your dealer.

**Turning On The Computer
With Disk Drives**

The ACE 500 Computer has one disk drive built into the computer case. This is the principal (or default) disk drive. If you have another disk drive connected externally to your computer, you can use it, but the instructions are a bit more complicated. For this procedure, use the internal disk drive. You'll learn more about disk drives and diskettes in later chapters.

To test your ACE 500 Computer computer and its disk drives, first make sure the computer is turned off. Then take the Franklin DOS 2 System Diskette, holding it carefully by the top edge, and insert it label side facing up into the disk drive.
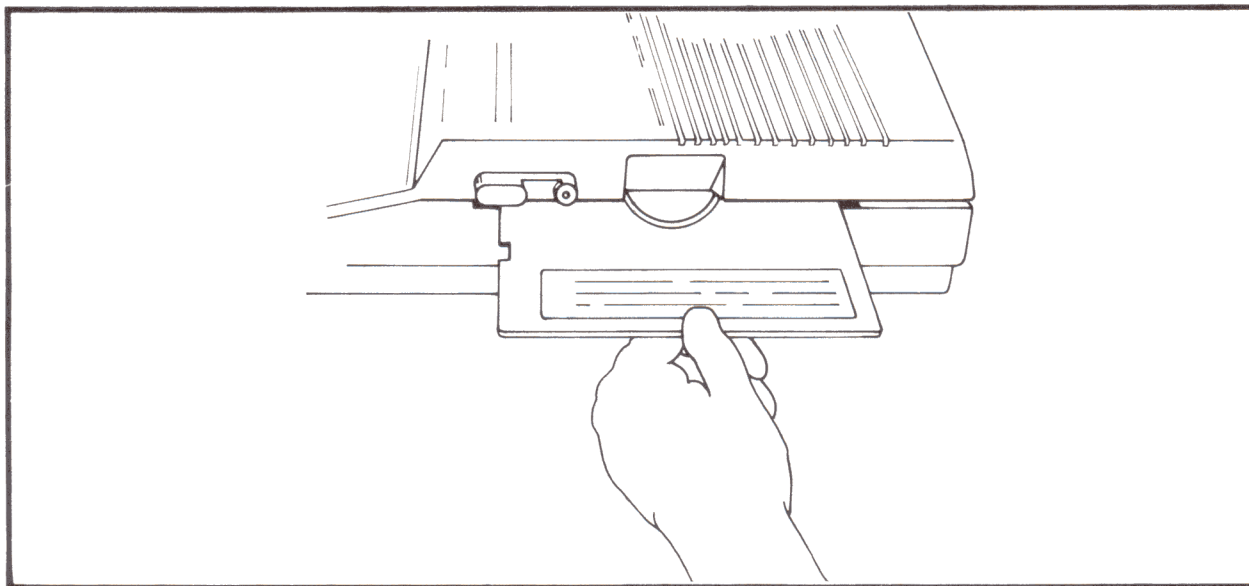
Inserting Diskette

Now turn on the video monitor and the computer. You should see the Power LED on the computer and the Disk Drive LED go on. Now close the disk drive latch by rotating it down across the opening. It is a good habit to keep the disk drive latch open whenever you turn the power on or off. This helps protect the diskette against power surges.

You'll hear a whirring noise as the drive reads the diskette. This message will appear on the screen:

ACE 500 System Diskette
Copyright 1986 by

FRANKLIN COMPUTER CORP

You can run any of these programs:

1 COPY   4 ONE DRIVE COPY
2 FUD   5 DIAGNOSTICS
3 VERIFY

You can run a program just by typing
   its number (1 thru 5).

You can find out what each program
   does by typing the 'H' key.

You can stop this program by typing
   the 'Q' key.

Which would you like?
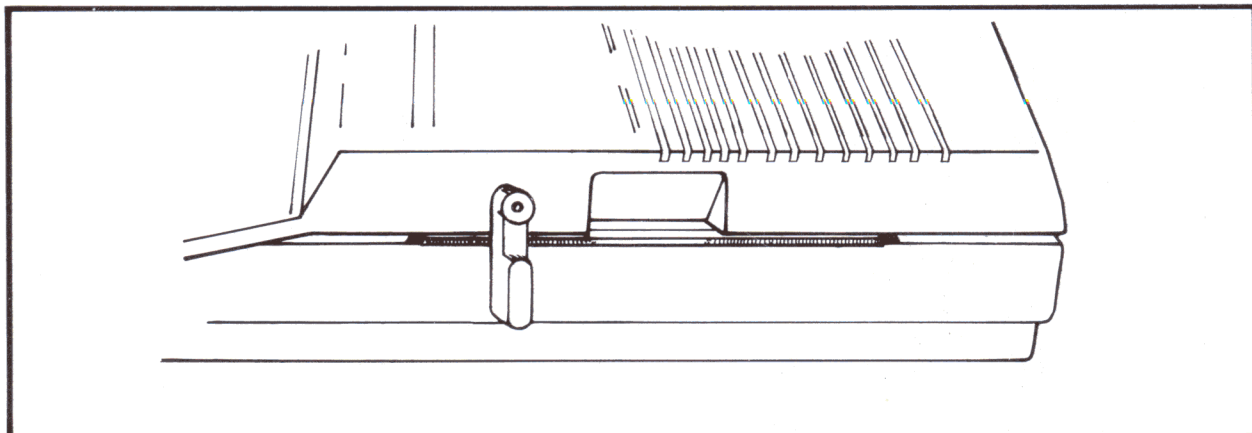
Disk Drive Latch — Closed

This is the HELLO program which runs automatically whenever you start up the Franklin DOS 2 diskette. It means that everything is working well.

Open the disk drive latch and turn off the computer for a while until you learn a little more about how it works and things to watch out for.

If all didn't work as it was supposed to, check the power connections for both the computer and the monitor, make sure you have the diskette inserted correctly and that the disk drive latch is closed. You may need to adjust the brightness and contrast of the video monitor.

Other ACE 500 Computer Features

Some of the other features of the ACE 500 Computer may be of interest to you but may involve a little more technical information than you are comfortable with. If you're interested about these things, read on, but realize that it's not necessary for you to understand all these details to use the computer effectively.

OPTION SWITCH Settings

The OPTION SWITCH group on the back panel has a number of settings, some of which involve concepts and procedures you'll encounter later, others which are fairly technical. The operation of these switches is described here for your reference. Don't set them without further information.
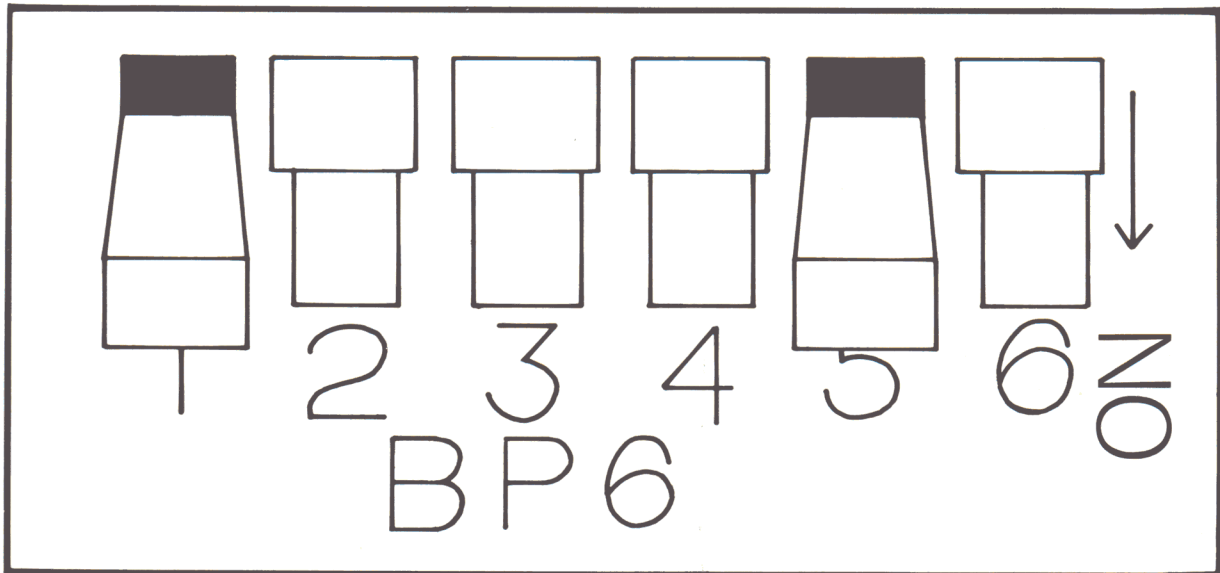
SWITCH #1 — Not Used.

SWITCH #2 — This switch governs the polarity of the line that carries both the Composite Synchronization signal for Apple mode monitors and the Horizontal Synchronization signal for IBM mode monitors. The switch position Down is for Negative signals and the Up position is for Positive signals.

Check your RGB monitor manual to see which polarity you need, either for the composite or the horizontal sync signals for Apple or IBM type monitors, respectively.

SWITCH #3 — This switch governs the polarity of the line that caries the Vertical Sync Signal for IBM mode monitors. The Down position selects the Negative Vertical Sync signal. The Up position selects Positive Vertical Sync. Positive Vertical Sync is generally the more common setting.

SWITCH #4 — Apple/IBM Mode   This switch selects the configuration of the color matrix, or color palette, that the computer uses, either according to the Apple or the IBM formats. The Down position selects the IBM color palette, Up selects the Apple color matrix mode.

If your RGB monitor says "Apple compatible" or "IBM compatible", or has some other indication that it is designed to work with one of those computers, try the configuration suggested for that computer. The palette assignments for colors is somewhat arbitrary and may not matter until you use some software that produces red grass and orange sky. The four-position RGB slide switch on the front panel should give white, green, amber, and blue text in its four settings. If these colors aren't accurate, the wrong color palette has been selected.

SWITCH #5 — Monochrome Double High Resolution   This switch selects a single color (the text color selected with the RGB slide switch on the front panel) on a black background when the computer is used with a double high resolution screen. This is generally only used with specific software applications, such as Computer Assisted Design, where the four bit positions on the screen needed for RGB color are traded for maximum screen resolution.

The Down position of switch #5 selects monochrome display when the computer uses Double High Resolution display mode.

If you're using an RGB monitor, the Down position selects the monochrome selec-

tion and the Up position selects RGB. The monochrome screen in double high resolution mode offers increased visible resolution needed for some applications, such as Computer Aided Design. The monochrome selection (the Down position) also offers better readability for double high resolution screens, especially those that use mixed text and graphics. You may need to try this switch in both settings for different software programs to see which screen presentation you prefer.

The slide switch on the front panel above the keyboard has four positions which select the text color used by the RGB monitor when operating in text mode. The colors are white, green, amber, and blue on the black background. Try the switch in different positions to see which you prefer. The factory setting has white text selected.

SWITCH #6 — Auto Line Feed for printer. Up is LF, Down is no LF. This switch determines whether the computer sends a line feed command to the printer at the end of each line or not. Some printers expect the computer to supply the command to move down a line, other printers supply the command themselves. Consult your printer manual to see what your printer requires for the line feed command. If you can't tell, set the line feed switch Down and try printing something. If the lines are double spaced, the printer is sending a line feed in addition to the one you're sending, so the proper setting is line feed Off.

Connecting Peripheral Devices

The DB25 connector on the back panel accepts a standard cable to connect to a parallel printer. Making the connection is simple, but you may need to consult the printer's manual for instructions on inserting paper, threading the printer ribbon, etc. Chapter 10, *Input/Output* of this manual discusses communication with the printer on a more technical level and may contain some material of interest to you.

PARALLEL PORT
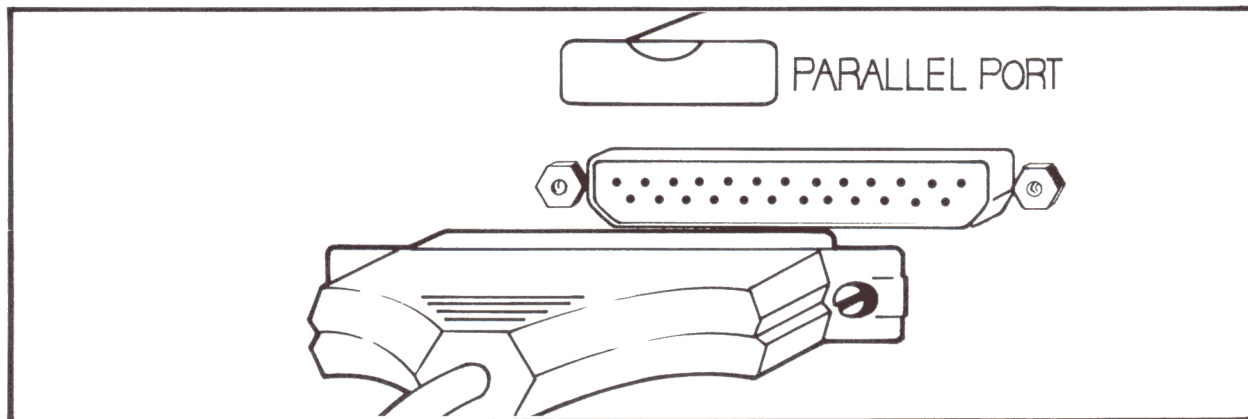
The joystick connector is a standard DB9 plug next to the RGB connector. Making the connection is simply a matter of plugging it in. If the software you're using accepts input from a joystick, you'll find that the joystick is active. There's no special command needed to turn it on.
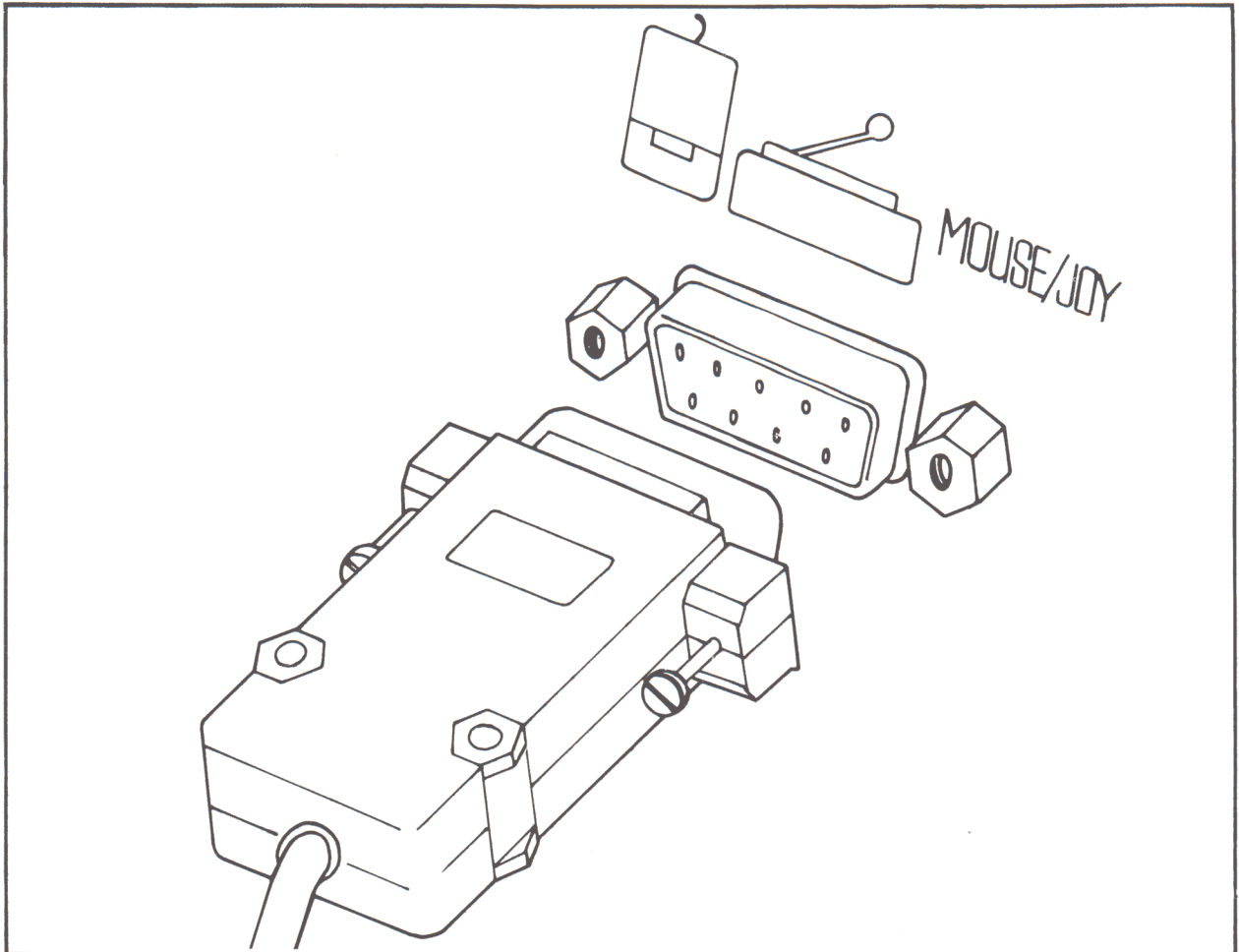
The Franklin ACE 500 Computer has the capability to expand to meet special needs of a number of peripheral devices. These devices are attached to the computer by means of the serial port connector on the back panel. A modem, a serial printer, or

other serial device, that is Franklin ACE 500 or Apple IIc compatible can be con-
nected to the serial port. A number of configuration selections have to be made for
serial communications operation. These are discussed in detail in Chapter 10, *Input/
Output*.

SERIAL PORT
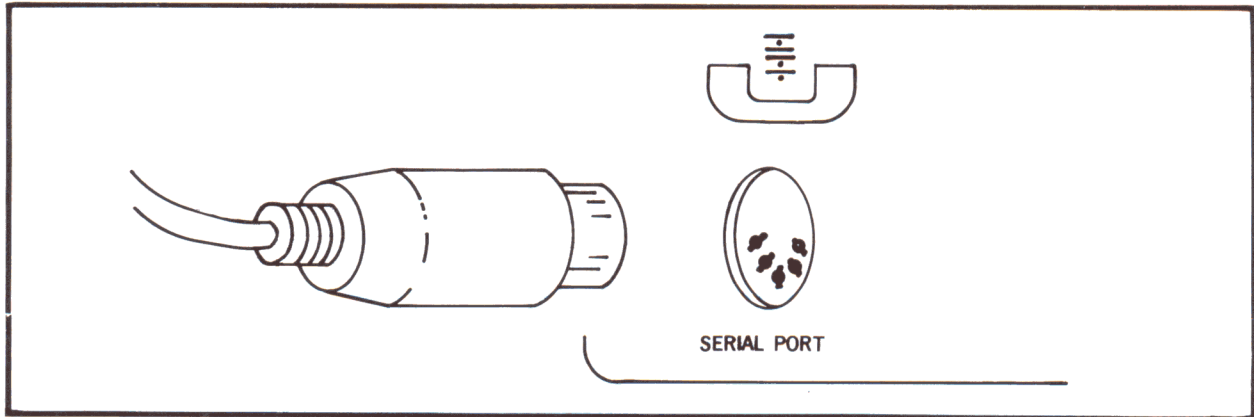
A second disk drive may be connected to the Franklin ACE 500 by means of the
nineteen pin connector on the back panel. The disk drive may be either 5 1/4″ or 3
1/2″ diskette format. Be sure to specify that the disk drive must be Franklin ACE
500 or Apple IIc compatible. The 3 1/2″ Apple Unidisk drive won't work because it
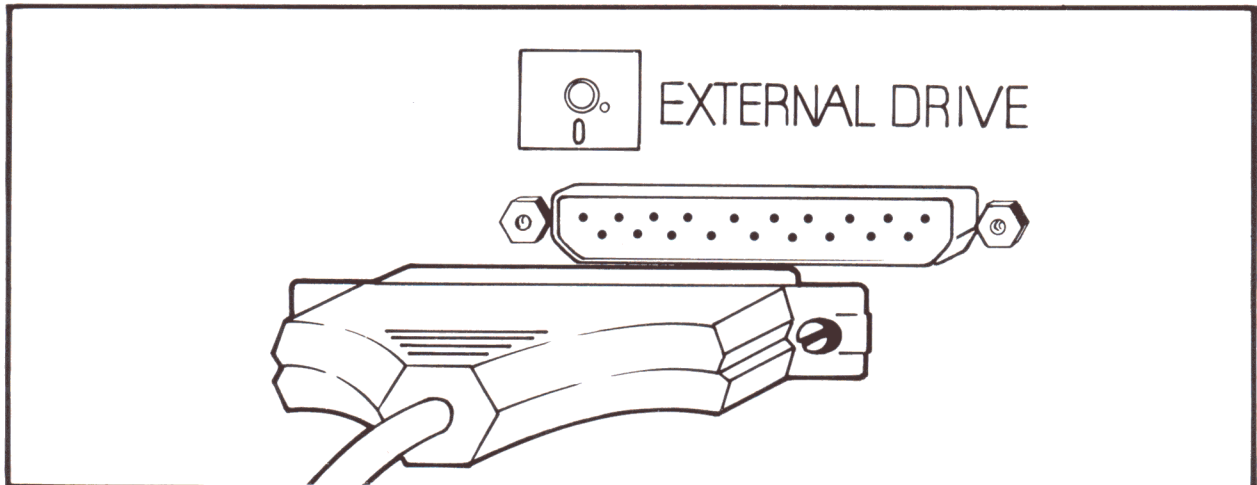
EXTERNAL DRIVE

uses a different disk drive interface. A disk drive connected externally is addressed with the operating system parameters slot 6, drive 2, or just d2. Refer to Chapters 4 and 5 on Franklin DOS 2 for operating system commands and parameters.

ACE 500 Computer Memory expansion to 512 K internal RAM can be installed as an option. This is a particularly useful feature for users of programs with large data requirements, such as Appleworks, or for users who want the convenience of using a RAM disk. The 256K memory in the ACE 500 is sufficient for many RAM disk applications, but some users need more memory. The FRAMDISK program included on the Franklin DOS 2 diskette provides the software for a RAM disk.

The extra RAM memory of the ACE 500 can also be used to emulate a number of the popular extra memory peripheral cards available, such as the RAM Works card. This permits special use by software routines, additional programming features for the video display, etc.

The case of the ACE 500 was not designed to be opened by the user. There are no user serviceable parts inside.

Chapter 3

The Keyboard

This chapter is for all users. It contains a description of the keyboard and the function of each key. It also describes special keyboard operations and characters used by computers, and the different operating modes of the keypad groups.

The ACE 500 Computer keyboard is one of the most important features of the computer because it provides your interface with the computer. The keyboard has a number of design features that benefit you while you use the computer.
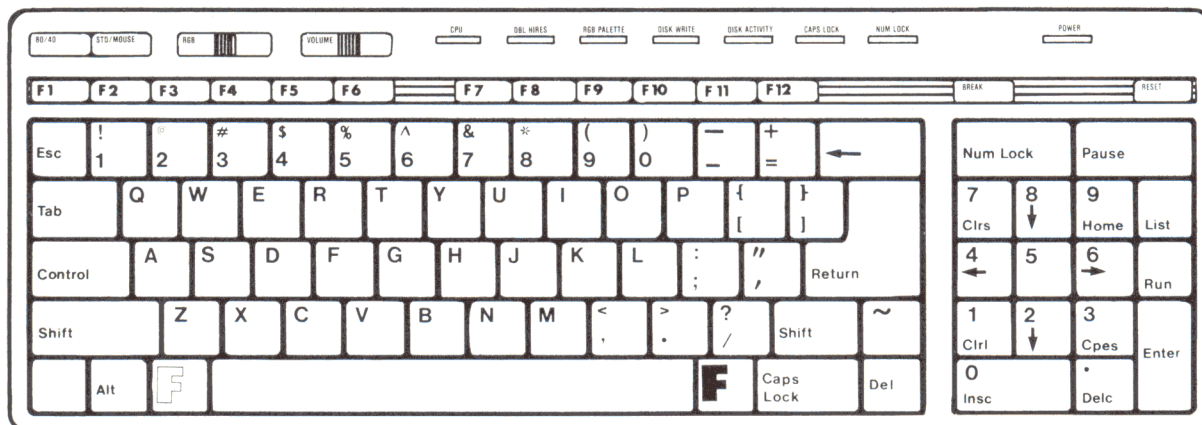
- The keytouch, shaped keytops, auto-repeat, and other built-in features make the keyboard comfortable and easy to use.

- The key arrangements follow ergonomic design standards to give maximum convenience for location of commonly used keys, like SHIFT and RETURN.

- The numeric keypad is designed for easy entry of numeric data, including the duplication of the RETURN function on the ENTER key.

- Twelve easily programmed function keys in convenient groupings allow custom application of many commands and functions.

Typewriter Functions Of The Keyboard

The keyboard on your ACE 500 Computer is very similar to the keyboard of a typewriter, but it has some keys usually only found on computers that you may not recognize.

This chapter will discuss each of the keyboard groups in turn, pointing out the similarities and differences between the computer keyboard and the typewriter keyboard.
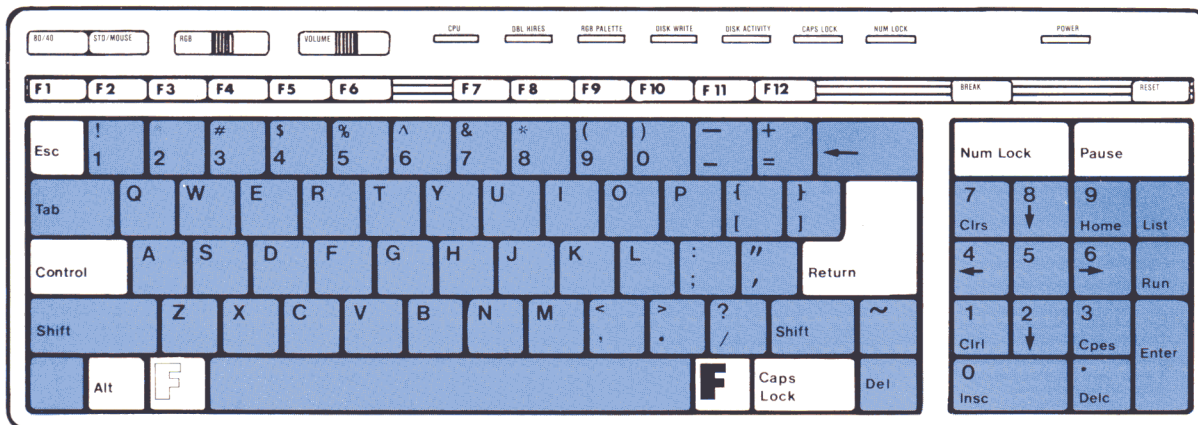
ACE 500 Computer General Keyboard

Your ACE 500 Computer has two main keyboard groups, the main keypad with the alphabetic and punctuation keys, and the numeric keypad on the right side. The design of these keyboard groups is as close as possible to the layout of a typewriter and a calculator so you can use your keyboard skills as you work with your computer.

The main keypad generates upper and lower case letters with the use of the SHIFT keys at the lower left and lower right. If a key has two symbols on it, the upper one is generated by SHIFT with the key and the lower symbol is produced by pressing the key alone. That's all just like using a typewriter.

Other keys usually found on a typewriter, like the BACKSPACE (shown as a left arrow), TAB, and SPACE BAR keys work in their usual way. The DEL key generates a special computer signal (a RUBOUT character, ASCII 127). Only users with technical applications are likely to use DEL.

Shift Keys

The illustration shows the different Shift keys available on the keyboard and their location.
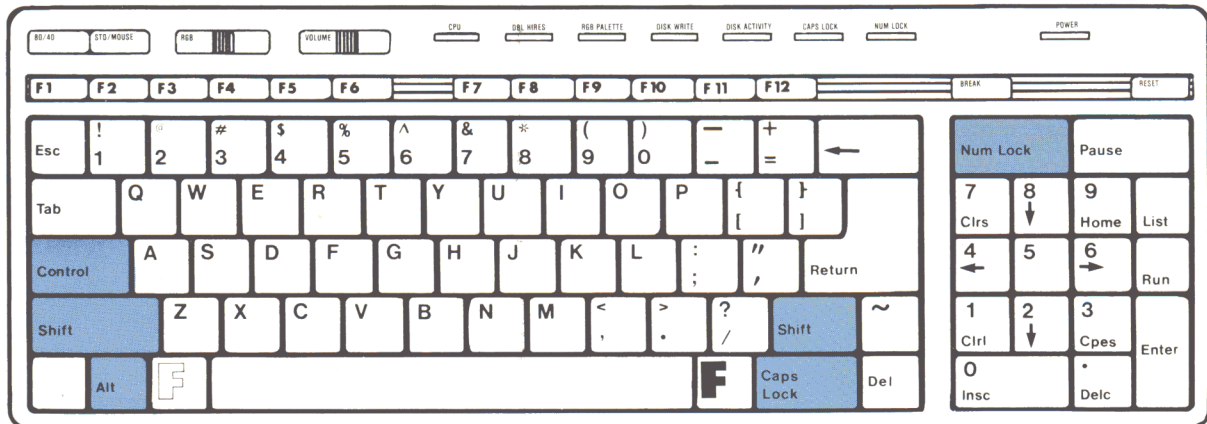
The familiar SHIFT key produces an uppercase letter when it is used with another key. There are other shift keys available on a computer that work in the same way.

The CAPS LOCK key is similar to the shift-lock key on a typewriter except it works only with the alphabetic characters on the computer. The indicator light above the keyboard marked CAPS LOCK indicates when the key is in upper case mode. You can turn it on and off by pressing the CAPS LOCK key successively.

The CONTROL key, usually abbreviated CTRL, located at the lower left of the keyboard, is another shift key. If you hold down the CONTROL key while you press another key, you generate a special control code. These control codes are often used for special commands to the computer but aren't printed or displayed on the screen. You'll use control codes often as you operate the computer. They are written as CTRL A, which means that you hold down the CTRL key while you press A. Sometimes a frequently used control code is given its own key. For instance, the RETURN key produces CTRL M.

The NUM LOCK key controls the action of the keys in the numeric keypad, whether they generate the numbers and symbols marked on their keytops or the other characters, either cursor movement arrows or editing commands. You'll learn more about these functions later in this chapter. For now, the object is to understand the operation of the NUM LOCK key in shifting from one mode of operation to

another. The LED indicator light above the keyboard marked NUM LOCK shows when the numeric keypad is in "number mode".

The ALT key is also a shift key, but it generates special codes that are used only in special technical applications that are discussed in the *ACE 500 Technical Reference Manual.* It is also used for some special commands, as you'll see later.

Command Keys

The illustration shows the various command keys available on the keyboard and their location.
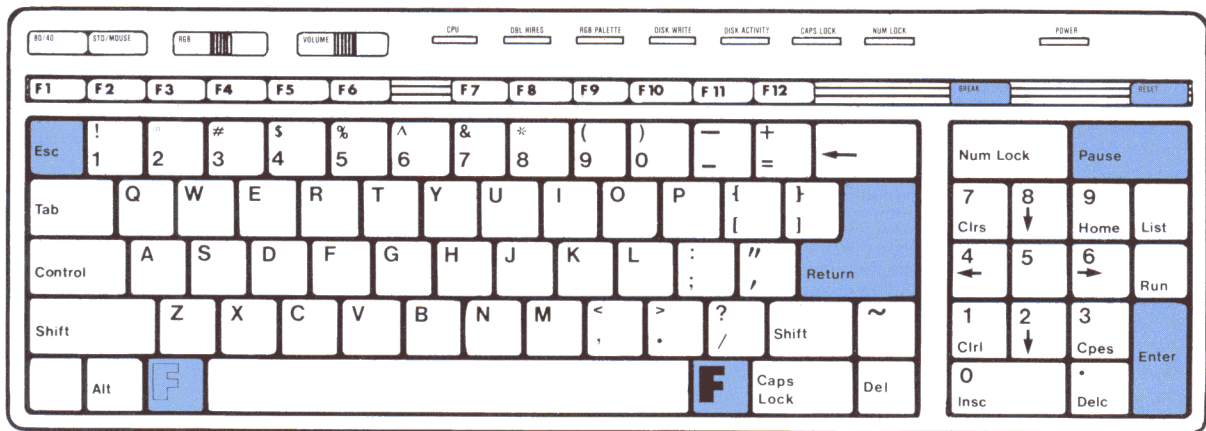
The RETURN (for Carriage Return) key is a large key in easy reach for its frequent use. There is one RETURN key at the right side of the main keypad and another one (labelled ENTER) at the right side of the numeric keypad for your convenience. For many operations, the RETURN key is just a carriage return. It can serve an additional function on a computer, and that is as a command key. When you enter a command on the computer, you conclude it by pressing RETURN. In this sense RETURN is a way of saying "This is my command, execute it." RETURN is one of the most powerful and most frequently used keys on the keyboard.

The ENTER key on the numeric keypad is the same as the RETURN key. It performs the same function and is repeated in that location as a convenience to you when you're entering a lot of numeric data.

The ESC (for Escape) key is often used for commands, either as an introducing character for a command sequence, or as a response to a question by the computer. An

example might be ESC A, in which you press ESC and then you press A. Sometimes ESC means special things to some software applications, and it's best to use it carefully.

The PAUSE key sends a signal to the computer to stop the display. For instance, if the computer puts a large piece of text on the screen and it moves off faster than you can read it, press the PAUSE key to stop the display. Press it again to resume the display. PAUSE sends a signal (CTRL S, also called XOFF, which in computer language is "transmit off") to the computer to stop sending data to the screen, then sends another signal (CTRL Q, also called XON, for "transmit on") to resume sending data. This pair of signals is called the XON/XOFF transmission protocol.

The BREAK key sends a signal to the computer to stop the execution of a BASIC program you're running. This is a fairly drastic step and may cause some data to be lost. BREAK is the same as CTRL C, which is sometimes used as a special computer command.

The RESET key represents a very drastic step. When you reset the computer you clear the memory and therefore almost always lose the program you're working on. On the other hand, sometimes the computer seems to get stuck in some process and you can't regain control. Resetting the computer is the way to stop the computer's activity and start it again. Because the reset function is so powerful you have to press CTRL RESET to activate it. This two-key procedure prevents accidental pressing of the RESET key. Reset is also called a "warm start".

The "open F" and "closed F" keys are special keys that don't actually do much themselves but are used together with other keys in special commands. They are sometimes used in application programs and correspond to the "open Apple" and "closed Apple" keys on Apple computers.

Function Keys

The illustration shows the location of the 12 function keys on the keyboard.

Function keys get their name because they generate a function, such as a command, or a series of characters, instead of just a single character or control code like other keys. They are also called "programmable" keys because they can be programmed to generate any command or definition you want. Specific instructions on controlling the function keys and programming their contents is contained in Chapter 8, *Function Keys*. The definitions the keys have when you boot the Franklin DOS 2 diskette are given below.

Default Function Key Definitions

The Franklin DOS 2 diskette contains a set of definitions for the function keys in the start up file that is loaded each time you boot the diskette. The definitions are:

Key	Definition	Description
F1	CATALOG,D1<RETURN>	Catalog of drive 1
F2	CATALOG,D2<RETURN>	Catalog of drive 2
F3	PR#3 <RETURN>	Turn on 80 columns
F4	PR#1 <RETURN>	Turn on printer output
F5	<ESC> 4	Switch from 80 to 40 columns
F6	<ESC> 8	Switch from 40 to 80 columns
F7	COPY	Copy a diskette file
F8	TYPE	List a diskette file
F9	LOCK	Lock a diskette file
F10	UNLOCK	Unlock a diskette file
F11	DELETE	Delete a diskette file
F12	RENAME	Rename a diskette file

Note that <RETURN> and <ESC< in the definitions are single keys, not the spelled out versions of the commands. The definitions are shown here in uppercase letters, but Franklin DOS 2 could process them just as well as lowercase letters. Note also that the commands that require <RETURN> for execution have it included in

their definition. For other commands that require additional input, like COPY, you have to supply the additional information and <RETURN>. For instance, the command line

COPY filename <RETURN>

would be typed:

F7 filename <RETURN>.

The operation of the Franklin DOS 2 commands themselves are given in Chapters 4 and 5 on Franklin DOS 2 and Chapter 9 on the video display. As far as the function keys are concerned, all you have to remember is that the key produces the sequence of characters shown in the definitions whenever it is pressed.

Numeric Keypad Operation

The illustration shows the numeric keypad. The functions and operation of its keys are described below.
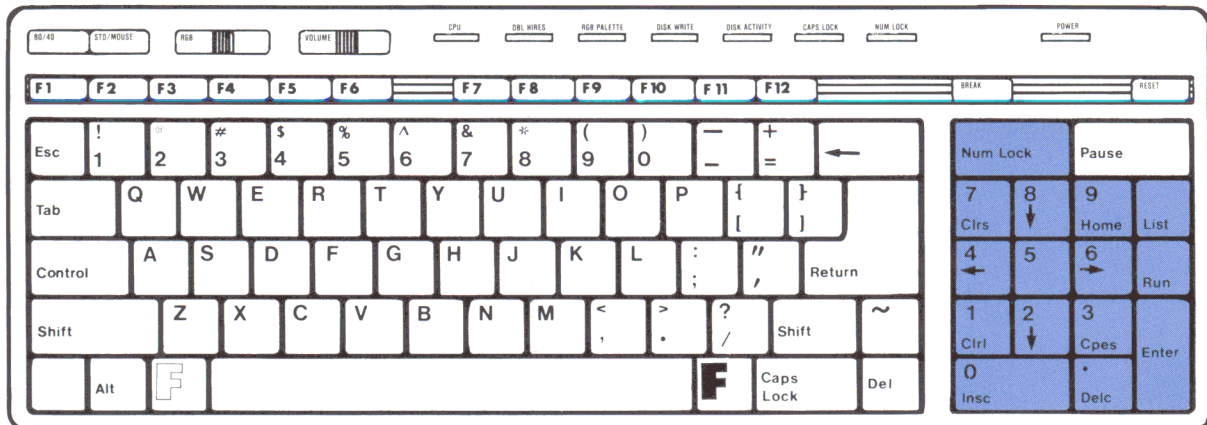
The numeric keypad operates in two modes, the numeric mode in which the keys generate numbers and arithmetic symbols, and an editing mode in which the keys generate the cursor movements and the various BASIC editing commands shown on their keytops. The NUM LOCK key controls the choice between these two modes, with the NUM LOCK LED showing the numeric mode operation.

Numeric Mode Commands

When the numeric keypad is in numeric mode (NUM LOCK on), the keys generate numerals 0 through 9, ., +, -, and ENTER (which is the same as RETURN).

Numeric Keypad Editing Commands

When the numeric keypad is in editing mode (NUM LOCK off), the keys generate the commands listed below. Some of the editing commands are part of the enhanced video software. They are available only when that software is activated. You may need to try each application program you use to see if the enhanced video software is activated by it, and therefore if the editing commands on the numeric keypad are operational. If you are using your own programs, remember to turn on the enhanced video software before using these commands.

The arrow keys, RUN, LIST, and ENTER do not require the enhanced video software and should be available for general use any time the editing pad is activated.

RUN issues the BASIC command to RUN a program.

LIST issues the BASIC command to LIST a program.

ENTER is the same as the RETURN key.

The arrow keys move the cursor on the screen.

The other editing keys listed below are functional only when the enhanced video software is active.

CLRS—Clear The Screen
This command clears the entire screen and moves the cursor to the home position, column one of line one.

HOME—Home The Cursor
This command moves the cursor to the home position, which is the first column of the first line.

CLRL—Clear The Line
This command clears the entire line designated by the cursor position. The cursor position itself is unaffected.

INSC—Insert A Character
This command inserts a space character at the current cursor position and pushes other characters on the line one position to the right to make room for it. Only char-

acters to the right of the cursor column are affected. If there is a character in the last column of the line, that character is lost.

DELC—Delete A Character
This command deletes the character at the current cursor position and moves all the columns on the right side one space from the right to close up the line. A space character appears in the last column of the line.

CPES—Copy Entire Screen Line
This command is useful in editing BASIC programs and has a very specialized operation. This command sends all the characters on the cursor line to the current program for processing or execution. This is equivalent to moving the cursor to the end of the line and then hitting RETURN. The CPES command works regardless of the cursor position on the line.

CPES is designed for editing BASIC statements. Suppose you had typed a line of a program that had a mistake in it. First, you would LIST the line, then use the escape mode cursor movement keys to locate the section of the line that needed correction, then make the changes with the INSC and DELC keys. When the changes are complete, just press CPES. The corrected line is sent to BASIC with a RETURN without any additional cursor movement required.

If you use the CPES command on a full line (all 40 or all 80 characters are used), you may notice that the last character on the line is cleared to a space character after CPES executes. Don't worry, this is only the display screen's processing of the command. The correct and complete line was sent to the running program.

Chapter 4

Franklin DOS 2 (Disk Operating System)

This chapter is for all users. It describes the operation of DOS, the disk operating system software. It describes starting the computer with diskettes, manipulating the programs and data on diskettes, and controlling the flow of information to and from the disk drives.

The latter parts of the chapter give examples of creating your own boot diskettes and using some basic utility programs on the Franklin DOS 2 diskette.

The Franklin Disk Operating System, known as Franklin DOS 2, allows your ACE 500 Computer to use the thousands of application programs available on diskette. The Disk Operating System is software that handles interactions with diskettes and disk drives. It's used whenever you use the disk drive.

The disk operating system, Franklin DOS 2, and Franklin BASIC (Franklin BASIC programming language) are sometimes thought of as a single program. They are not, but they work closely together and you probably don't need to be concerned about the differences. Whenever you turn on the computer with the Franklin DOS 2 diskette in the disk drive, Franklin DOS 2 software goes from the diskette to the computer's memory where it's available for your use until you turn the computer off, or load an application program.

You'll do well to start your computer with the Franklin DOS 2 diskette or an exact duplicate each time you turn on the power (unless you're using another operating system, such as Pascal, exclusively).

Buying Franklin DOS 2 Application Programs

Your ACE 500 Computer will run virtually any Apple DOS 3.3, Apple ProDOS, Franklin DOS, or Floating Point BASIC programs, with three major exceptions:

1. Programs written in Integer BASIC. If a listing of the Integer BASIC program is available, you may be able to alter the program so it can be entered as a Floating Point BASIC program.

2. Older programs for Apple computers that produce lowercase characters unconventionally (sometimes referred to as the "shift mode" option).

3. Programs that interact with the operating system in non-standard ways.

Many application program diskettes come with an operating system on the diskette, so you may not be obliged to boot with the Franklin DOS 2 diskette first. The best way to find out is to try it and see. Be aware, however, that the special features of Franklin DOS 2 will not be available to you if you rely on the operating system provided on the application program diskette. Even if you do boot first with the Franklin DOS 2 diskette, you may find that the application program's operating system overrides Franklin DOS 2 by overwriting it in memory (this generally doesn't happen with Franklin BASIC).

If you're not sure a program will run on your ACE 500 Computer, check with your dealer or perhaps your local user's group, or try it out before you buy it.

Disk Drives For Your Computer

Disk drives are vital parts of a computer system because they allow easy movement of programs or data into and out of the computer. When you run an application program, the program is read from the diskette by the computer on your command, and loaded into the computer's internal memory for operation. When you write a pro-

gram, or a letter, or make a mailing list, that data is taken from the computer's internal memory and stored on a diskette so it will be available to you when you want it. Without this storage facility, the computer alone is a limited tool.
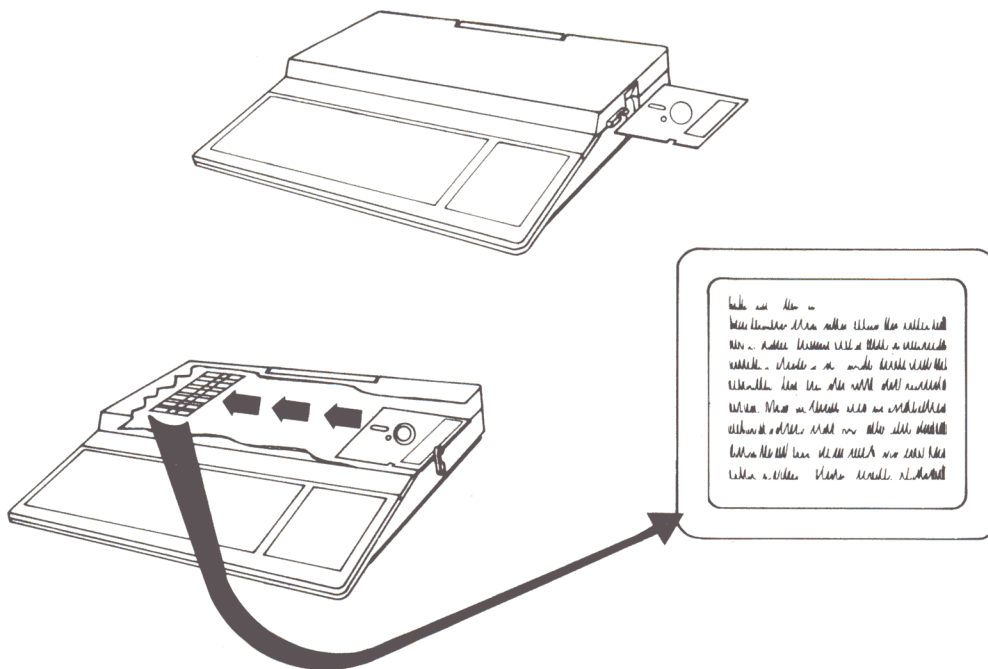
Your ACE 500 Computer has one disk drive built in and it may have an additional drive connected through the connector on the back panel. If you have a second disk drive, you'll need to be aware of the conventions Franklin DOS 2 uses to designate each drive and the ways to use them both.

Diskettes For Your Computer

You'll need a supply of diskettes to use your ACE 500 Computer effectively. You'll use them to make copies of your operating system diskettes, as well as for data storage and programs.

When you purchase blank diskettes, buy 16-sector or soft-sectored 5 1/4 inch single sided, single density diskettes. These are the most common diskettes available.

The diskettes you fill with data or programs are relatively inexpensive compared to the price of the computer, but they represent a lot of time and effort, so you need to take good care of them to protect this investment. Be careful with your diskettes.

Here are some suggestions for protecting your diskettes.
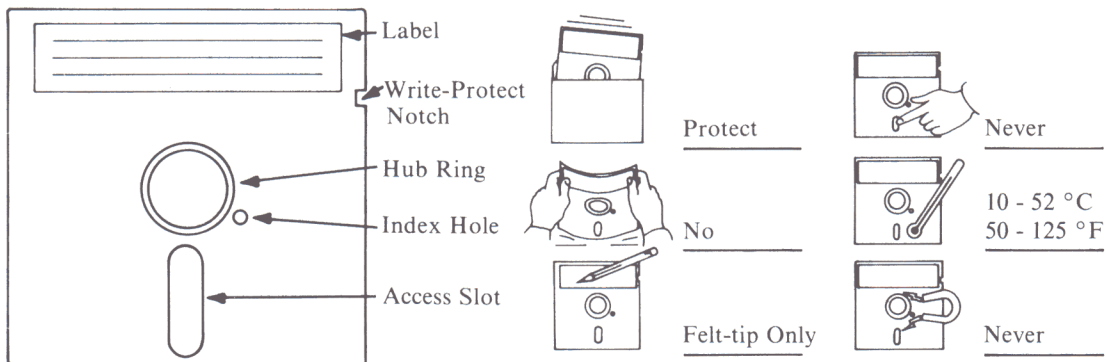
- Always make backup copies of important diskettes, especially diskettes that hold your data, operating system, or programs.

- Handle the diskettes carefully. Always keep them in their protective envelopes. Store them flat and protected; keep and use them at normal room temperature.

- Diskettes are made of a plastic medium with a magnetic coating enclosed in an envelope. The diskette is sensitive to dirt, to smoke, and to the oil on your fingers, so don't touch the exposed areas of the diskette. Keep diskettes away from any magnetic and electrical disturbances, such as electric motors, magnets, video monitors, etc.

The ACE 500 Franklin DOS 2 System Diskette

The ACE 500 System diskette contains Franklin DOS 2 as well as a number of useful programs for copying diskettes, for handling files you use, and other general computer tasks.

Chapter 2, *Getting Started*, of this manual showed you how to boot the ACE 500 Computer using the Franklin DOS 2 diskette. The following procedure reviews that sequence to remind you how to start the computer.

1. Insert the Franklin DOS 2 diskette with the label up into the horizontal slot of the disk drive. Push it all the way in but don't force it after it stops.
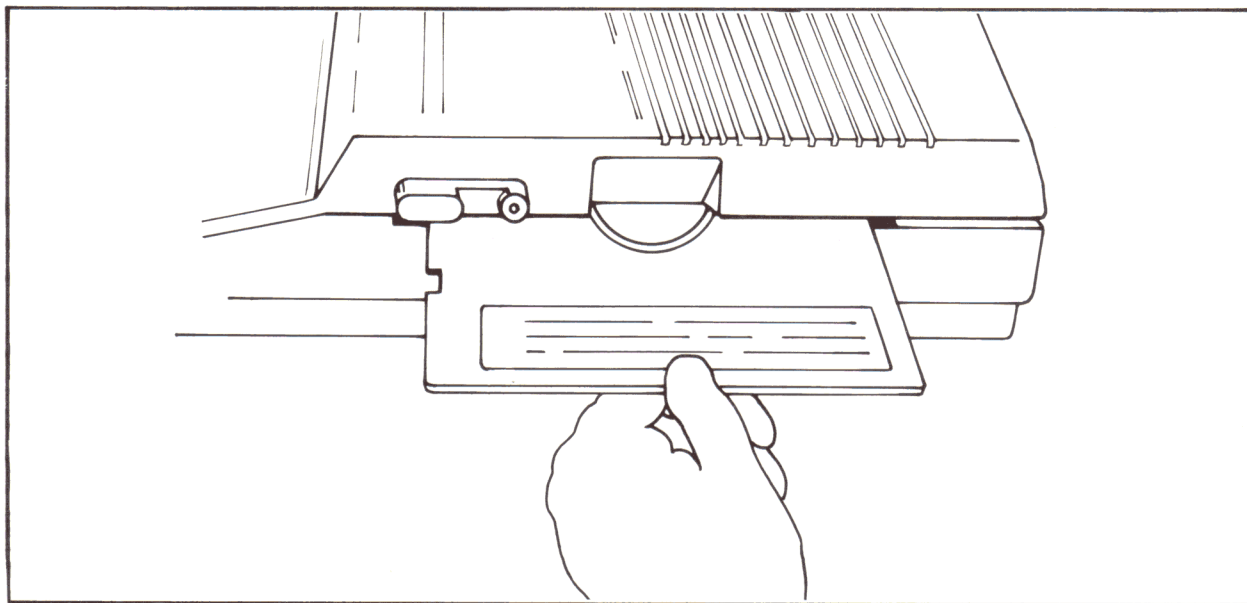
2. Turn on the power to the monitor. Then turn on the power to the computer with the power switch on the back panel.

3. Turn the latch on the disk drive down across the disk drive slot. This puts the reading mechanism down onto the surface of the diskette (think of putting the stylus of a record player down onto the surface of the record). It's a good idea always to open the disk drive latch whenever you turn the power on or off as a protection against power surges that could harm your diskettes.

The computer now starts to read the diskette. There may be a little noise as the disk drive adjusts itself and then reads Franklin DOS 2 into its internal memory. You'll see a message appear on the screen, something like this:

ACE 500 System Diskette
Copyright 1986 by

FRANKLIN COMPUTER CORP

You can run any of these programs:

1 COPY   4 ONE DRIVE COPY
2 FUD   5 DIAGNOSTICS
3 VERIFY

You can run a program just by typing
    its number (1 thru 5).

You can find out what each program
    does by typing the 'H' key.

You can stop this program by typing
    the 'Q' key.

Which would you like?

This message is the HELLO program that runs automatically each time you boot the Franklin DOS 2 diskette. It tells you (among other things) that everything is ready to go.

Other Franklin DOS 2 diskettes may have slightly different contents or screen displays as Franklin updates the software contents of the ACE 500 system software. The "H" selection is a kind of help message which describes the operation of each of the programs listed in the menu. If you don't want to use any of those programs, press the "Q" key to quit the menu and go into BASIC.

Making Backup Copies Of Your Franklin DOS 2 Diskette

Before you use your Franklin DOS 2 diskette (or any new diskette) you should make a backup copy of it. Use the copies for everyday use and keep the original diskette in a safe place. This is a good habit to establish, and one you'll appreciate if you ever "crash" an important diskette.

The Franklin utility programs, COPY and ONE DRIVE COPY, make it easy to make backup copies of diskettes. Before you proceed, get a few blank diskettes to make your copies.

One Disk Drive Systems

Making a copy of a diskette is a common procedure, one you'll do many times with your computer. The software takes care of all the details, but the strategy is simple. The disk drive reads the contents of the source diskette into the computer's memory, then you swap the source diskette in the disk drive for the destination diskette, and the computer writes the contents of its memory onto the destination diskette. This process is repeated until the entire diskette is copied. The software tells you on the screen what to do at what point, so you won't get lost. You just have to be careful not to mix up the source and the destination diskettes.

You can choose to run ONE DRIVE COPY program from the HELLO program (selection 4), or exit HELLO and get the Franklin DOS 2 prompt. When you see the Franklin DOS 2 prompt] and the cursor, type:

<div align="center">

RUN ONE DRIVE COPY <RETURN>

</div>

Either procedure loads the program you need to make the copy of the diskette. The screen displays:

<div align="center">

Copy Program Vx.x
Franklin Computer Corp.

This program copies any DOS disk.
Which drive is the original disk in?
Press 1, 2, or ESC to exit.

</div>

ONE DRIVE COPY asks you to identify the drive which holds the original diskette, and next, which drive has the duplicated diskette. Because you only have one drive, you have to respond with drive 1 for both the source and the destination diskettes. Press 1 to show drive 1 for the original diskette. The program then asks:

> Which drive is the blank disk in?
> Press 1, 2, or ESC to exit.

Press 1 to show drive 1 for the destination diskette. This signals the program to prompt you on the screen when it's time to swap the source and destination diskettes back and forth in drive 1. The next screen shows:

> Please insert the source disk and
> press RETURN to continue or ESC
> to quit.

The program reads part of the source diskette into memory, then tells you to insert the destination diskette, then writes the contents of memory to the destination diskette, and finally asks you to put the source diskette back in so it can start the process again. The process is repeated until the entire diskette has been copied. The closing message is:

> The disk has been copied. Press
> any key to continue.

When you've made a few copies, put the original Franklin DOS 2 diskette in a safe place and proceed with a duplicate Franklin DOS 2 diskette. When you're finished with copying, press <ESC> to leave the copy program. You're now back in Franklin DOS 2, and should have the Franklin DOS 2 prompt,] , and flashing cursor.

Two Disk Drive Systems

If you have an external disk drive in addition to the internal disk drive, follow this procedure:

Use the HELLO program to help you make backup copies. Just select option 1 from the HELLO Menu, by pressing 1. You could also exit the HELLO program, get the Franklin DOS 2 prompt,], and type:

> RUN COPY <RETURN>

Either procedure loads the copy program for two disk drives into the computer's memory. The screen displays:

<div align="center">

Fast Copy Vx.x

Franklin Computer Corp.

Insert the disk to be copied into drive one, and a blank disk into drive two.
Press:
    1 for a 35 track copy
    2 for a 40 track copy
      or ESC to return to Franklin DOS 2

</div>

Insert the diskettes in the disk drives as instructed. Since your Franklin DOS 2 diskette is in 35 track format (you'll learn more about that later), press 1. There is no need to press RETURN to start the copying. When the copy is finished, you'll see the message:

<div align="center">

The copy has been written and verified.

</div>

Make two copies of your Franklin DOS 2 diskette. The COPY program asks if you want to make another copy, and all you have to do is replace the first copy with another blank diskette and run COPY again. When you're finished, remove the original Franklin DOS 2 diskette from drive 1 and store it in a safe place.

Replace the original Franklin DOS 2 diskette in drive 1 with one of the duplicates and press <ESC> to exit from the COPY program. The exit message says:

<div align="center">

Place a disk with a copy of DOS or CP/M on it into drive one, then press any key to re-boot.

</div>

This reboots the operating system and gives you a chance to test your duplicate Franklin DOS 2 diskette. Press any key to proceed.

Catalog Of A Diskette

Franklin DOS 2 has a command that will give you a list of files and programs contained on any diskette in the Franklin DOS 2 format. This catalog gives you a lot of

useful information about your files and their status on the diskette. You can learn a lot about Franklin DOS 2 by looking at the catalog of the Franklin DOS 2 diskette.

You can use the CATALOG command by typing CATALOG and then <RETURN>, or you can use an Franklin DOS 2 abbreviation. In Franklin DOS 2, typing / and then <RETURN> is the same as the CATALOG command.

Try the CATALOG command. Type:

<div align="center">

/ <RETURN>

</div>

You'll see a listing something like this:

> Franklin DOS 2 X.XX, Vol 0, 35T, Free secs: nnn
>
>   *D   1 ACE 500 Vn.n ----BOOT FILES
>   *F 21 HELLO
>   *D   1 ----------PROGRAM FILES
>   *B 20 FUD
>   *B 24 COPY
>   *B 17 ONE DRIVE COPY
>   *B 22 VERIFY
>   *B   8 FRAMDISK
>   *B   5 PRAMDISK
>   *F   8 AWSTARTUP11
>   *F   7 AWSTARTUP12
>   *F   7 AWSTARTUP13
>   *B   6 DIAGNOSTICS
>   *B   7 BOOT13
>   *B   7 RAMTEST
>   *B   3 CHAIN
>   *B   3 CHAIN2
>   *D   1 ----------DATA FILES
>   *B   7 RAM.DIAG
>   *B   8 DISK.DIAG
>   *B   4 GRAPHICS. DIAG
>   *B 34 GRAPHICS.DIAG.DATA
>   *B   8 KEYBOARD.DIAG
>   *B   4 PADDLES.DIAG

If the catalog is too long for the display screen, you can stop it by pressing PAUSE, the space bar, or by pressing the <CTRL> and <S> keys simultaneously. To see

the rest of the display, press any other key. If you press <ESC> or <CTRL> <C> or the <BREAK> key, the display will stop but you won't be able to restart it.

35 And 40 Track Diskette Formats

The legend 35T in the top line of the catalog refers to the 35 track single sided format of the diskette. Franklin DOS 2 will run several other diskette formats automatically, but most disk drives are not mechanically capable of using them. The ACE 500 Computer disk drives are capable of using either 35 track or 40 track diskette formats. Most disk drives on Apple and some older Franklin systems can use only 35 track formats.

Franklin DOS 2 analyzes diskettes to determine which format is used on them. CATALOG is a quick way to find out which format a diskette has. This is important, especially when you make copies of diskettes. Some copy programs, like ONE DRIVE COPY, only copy diskettes of the same format so the copies automatically have the same format as the original. Other programs, like COPY, the Franklin DOS 2 COPY command and the FUD utility program (which you'll learn about shortly) do permit copying from dissimilar formats.

The copies of the Franklin DOS 2 diskette you made are in the same 35 track format as the original. You remember that COPY asked you for the format of the diskette before it made the copy. Many copy programs select the format automatically, others ask you to select it specifically.

Franklin DOS 2 divides each track on the diskette into 16 sectors, each containing the equivalent of 256 characters (about as many characters as this paragraph contains). The legend "Free secs" in the top line refers to the number of unused sectors on the diskette, and is a measure of the space remaining for additional storage on that diskette.

It's good advice to stick to one format for your diskettes. If you have diskettes written in different formats Franklin DOS 2 will recognize them, but you might not always remember which is which. The commands to move files from diskette to diskette are very specific about whether they're going from similar to dissimilar formats. There are a few things to consider when you choose a diskette format.

35 track diskettes are much more common than 40 track diskettes. On the other hand, 40 track diskettes have about 15% more storage on each diskette. If you have little interaction with other users and diskette formats, it makes sense to choose the 40 track format because of the added storage. If you encounter 35 track diskettes frequently, at user's groups or software exchanges, you'll avoid confusion if your own

diskettes are 35 track too.

Diskette File Names

Each of the entries in the catalog list is a "file". To the computer, anything with a unique name is a file. This can be used loosely to mean a program (which is a set of commands), or lists of data the computer uses in diagnostic tests, or a letter or list of addresses that corresponds more closely to the ordinary definition of a file you might use in an office setting.

The catalog of the Franklin DOS 2 diskette has been divided with lines of dashes into categories of files so you can identify them more easily. The first group of files are BOOT FILES. These are files essential for getting the computer running. You'll use them to create diskettes that allow you to boot from a cold start (more about that in a moment). The next group of files are PROGRAM FILES. These are utility programs that help you by doing various file management tasks, like making backup copies of diskettes and so on. The third group of files are DATA FILES, which are just raw data that can't be used except by the diagnostic programs for testing procedures.

The asterisk at the beginning of each line means that the file is locked to protect it against accidental changes. You'll learn more about locking, and unlocking, files shortly.

The letter next to it indicates the type of file:

F = Floating Point BASIC programs
B = Binary files (machine language programs)
T = Text files
S = Assembly language source files
R = Relocatable object files
O = Binary object files
D = Binary data files

You don't have to worry about understanding all these terms. Most of them are used only in technical situations where their meaning is clear.

The next number is the size of the file, given in the numbers of sectors it takes up on the diskette.

The last entry is the name of the file. Franklin DOS 2 file names have from one to thirty characters. Spaces may be used within a name. The first character must be a

letter, but then any other characters (including non-printing control characters) may be used except the comma.

Slots, Drives, and Volumes

Franklin DOS 2 uses three more numbers to identify diskettes and locations for information. The three numbers are slot number, drive number, and volume number.

The slot number identifies the slot on the computer's main electronics board for the device you're using, and is the way the computer locates or addresses the device, whether a disk drive, printer, or other peripheral. For the disk drives, the slot is number 6. If you used a RAM disk drive (which would normally use a plug-in RAM memory card), you could use any unoccupied slot for it, but usually either slot 3 or 4. A slot is both a physical connector on the main electronics board and an address in memory which can be used for access to another device.

The drive number identifies the number of the disk drive in the particular slot. Each slot can accomodate two disk drives, number 1 and number 2. A slot can accomodate only one RAM disk drive.

The volume number designates a particular diskette if it has been labelled with a volume number, or it identifies a portion of the Franklin DOS 2 part of a hard disk. The volume number is seldom used except on hard disks. The volume number may be from 0 to 254. If you don't use a volume number, Franklin DOS 2 won't care and will use the diskette in the drive you specified. If you do use a volume number, it must match the volume number on the diskette. To avoid potential confusion, most people leave out the volume number specifications.

Slots, drives, and volumes are identified with S, D, and V followed by the appropriate numbers (without a space). All three numbers are set off by commas after the file name. Thus ,S6,D2,V254 is a complete diskette identification to Franklin DOS 2. The slot, drive, and volume designations may be given in any order, e.g. ,D2,V22,S6.

According to the way Franklin DOS 2 sees things in the ACE 500 Computers, the internal and external disk drives connected to the computer are controlled through slot 6. Therefore, you would designate the internal disk drive as: S6,D1; and the external drive as S6,D2.

If you don't use slot or drive numbers in a command, Franklin DOS 2 uses the last slot and drive numbers you did specify (called the default values). Of course, if you only have two disk drives, they are always in slot 6, so that is always your default slot value and you don't have to specify it at all. (There are other times when you do

need to tell Franklin DOS 2 about slot numbers you're using, for instance, that the printer is in slot 1, but that's another chapter.)

If you booted your computer with the Franklin DOS 2 diskette in drive 1, your default values are S6, D1. If you always use that disk drive for your diskettes, you don't have to specify any slot or drive numbers. If you want to use a diskette in drive 2, all you have to do is put ,D2 (slot 6 is still the default) after the command or file name you want to use on drive 2. Then drive 2 becomes your default drive. If you don't use volume numbers, V0 is always your default value.

Here's another example. Suppose you have a file called DATA ONE on a diskette with volume number 1 and a file called DATA TWO on a diskette with volume number 2. When the first diskette is in drive 1 and the second diskette is in drive 2, the full Franklin DOS 2 names of these two files would be:

DATA ONE, S6, D1, V1 and

DATA TWO, S6, D2, V2

If you switched the diskettes from drive to drive, their identifications would be:

DATA TWO, S6, D1, V2 and

DATA ONE, S6, D2, V1

Suppose you wanted to LOCK (or some other Franklin DOS 2 command) one of the files. Just put the diskette with the file on it in the disk drive, then specify:

LOCK DATA ONE, S6, D2, V1

In actual practice, you'll probably only use drive numbers frequently because slot numbers will almost always be 6, and volume numbers are seldom used.

Starting Franklin DOS 2   Now that you know how to use different kinds of diskettes, there are a few things you should know about disk operating systems and how they work with the computer.

If you try to turn on the computer without a diskette in the disk drive, the drive spins without result. If you press <CTRL><RESET> the processing cycle stops and you see a prompt and flashing cursor on the screen. At this point the computer is operat-

ing with software it has in its permanent memory (called ROM, for Read Only Memory). This software includes Franklin BASIC so you could write some programs if you wanted to, but you couldn't store them on diskette because Franklin DOS 2, the disk operating system, isn't available. Franklin DOS 2 controls all the interaction with the diskettes and disk drives.

Some application programs equipped with their own disk operating systems (or customized versions of Franklin DOS 2) will replace all or part of Franklin DOS 2 with their own commands. You seldom know anything about it until you try to use an Franklin DOS 2 command and find it doesn't work. If you want to restore Franklin DOS 2 to the computer's memory, there's an easy way to do it, called a reboot, or cold start, command. The reboot command is the same as turning the computer off and then on again.

First, make sure all operations of the previous program are finished and the disk drive lights are off. Then place the Franklin DOS 2 diskette in drive 1 and close the latch.

Now hold down the <CTRL> <OPEN F> keys while you press and release <RESET>. This command makes the computer read the diskette in drive 1.

This is a very powerful command because it overrides all other computer processes or programs. It replaces any programs in memory with the contents of the new diskette. This feature makes it very useful for changing programs or changing diskettes, but it is a command to use carefully.

The reboot command is a convenient way to start an application program or to change from one diskette or operating system to another.

Making Bootable Diskettes

When you prepare diskettes for your own use, there are two different types you can make. Each uses a different configuration of the operating system, whether Franklin DOS 2 or some other DOS, and each has its own purpose. As you use your computer your work patterns will lead you to select these different types according to your needs. If what follows seems somewhat technical, don't be discouraged. Try to recognize the practical use of each kind of diskette, even if you don't completely understand how or why the software works like this.

The first type of diskette is the "absolute minimum" diskette which you would use for data storage. This diskette provides a format suitable for receiving data, but it has no operating system so it can't be used as a boot diskette. The Format selection in the

FUD utility program on your Franklin DOS 2 diskette prepares these "absolute minimum" diskettes.

Generally, you'll use formatted diskettes for text files or as storage diskettes because they have the largest amount of available space on them. You'll use them only with some other program or diskette that makes Franklin DOS 2 commands available to you.

The second kind of diskette is the "boot", or cold start, diskette, which has a copy of Franklin DOS 2 on it. This allows you to use Franklin DOS 2 commands whenever you need them, but more importantly, the <CTRL><OPEN F><RESET> command will work with this diskette. If you have an application program in the computer and want to change to a new program, a new diskette, or to Franklin DOS 2, the <CTRL><OPEN F><RESET> command with a boot diskette is a quick and easy way to do it. This procedure won't work on a diskette that is formatted but doesn't have Franklin DOS 2 on it. You can make boot diskettes by using the INIT command discussed later.

A less frequently used term, the "warm start" command, refers to the <CTRL><C> or BREAK commands. The warm start command breaks and restarts the program processing activity but leaves the DOS and operating software intact, so it's an easy way to stop a program without losing everything.

Boot diskettes are useful for application programs that you use frequently because they are easy to boot and because Franklin DOS 2 provides many useful file management commands.

The INIT Command

The INIT command (short for "initialize") prepares blank diskettes for use as boot diskettes. The initialization process consists of three separate operations, but it is an automatic process that's easy to do.

First, the command formats the diskette. This means that it clears all the storage space on the diskette, prepares a catalog space, and readies the diskette for 35 or 40 track data storage.

Next, it writes Franklin DOS 2 onto the diskette. Last, it specifies a filename that is to be run automatically when the diskette is booted.

Suppose you wanted to initialize a blank diskette so that a file named HELLO ran each time you turned on the computer and booted Franklin DOS 2. This HELLO

file could contain a label for the diskette, perhaps a command to display the catalog of the diskette, or some other useful things. The Franklin DOS 2 diskette that came with the computer, for instance, runs the HELLO program automatically. The procedure to create a boot diskette is simple.

1. Take a blank diskette, or one you're ready to erase. Initializing a diskette destroys everything on it, even locked files. Make sure you've removed the write-protect tab and insert the diskette in drive 1.

2. Type this command:

INIT HELLO <RETURN>

Franklin DOS 2 takes care of each step of the process and finishes the initialization in about 15 seconds. INIT chose the 35 track format because that's the format of the Franklin DOS 2 diskette. If you wanted to use another diskette format, such as 40 track, you would use this command:

INIT HELLO, T40 <RETURN>

If you wanted to initialize a diskette in drive 2, you would use this command:

INIT HELLO, D2 <RETURN>

or, you could use both command parameters together:

INIT HELLO, D2, T40 <RETURN>

The next step in creating a bootable diskette is to transfer the file HELLO onto the diskette. Right now Franklin DOS 2 on the new diskette knows it's supposed to run the file HELLO as part of the boot process, but it can't run a file that isn't on the diskette.

The next chapter describes Franklin DOS 2 commands and utility programs on the Franklin DOS 2 diskette. It tells you how to copy individual files from one diskette to another. But at this point you have an opportunity to do a little customizing of your diskettes, because INIT will take a program currently in the computer's memory and put it onto the diskette as the HELLO program. What that means is that if you've written an Franklin BASIC program for this purpose, it's automatically put on the diskette. Similarly, if there's some other program or programs in memory that

you don't want, they get put on the diskette too. Just be a little careful and be aware of what's in memory.

If you're familiar with BASIC programming, you can clear the computer's memory by using the command NEW, or by rebooting the computer. Clearing the memory and transferring the desired program individually is a more certain procedure than using INIT. You'll want to use INIT's automatic load feature with the LOAD and SAVE commands of BASIC to create your own boot files. Refer to Chapter 6, *Introduction To Programming*, of this manual for some suggestions on programming and boot files.

The Franklin DOS 2 COPY Command

The Franklin DOS 2 COPY command, like the COPY program on the Franklin DOS 2 diskette, works only in a two drive system. If you don't have two disk drives, you must use the COPY selection of the FUD utility program to transfer the files. That procedure is outlined in the next chapter. It's very similar to the COPY command procedure that follows.

To copy HELLO from the Franklin DOS 2 diskette in drive 1 to the diskette you've just initialized in drive 2, type the following command:

<div align="center">COPY HELLO, D2 <RETURN></div>

If the destination diskette already contains a file called HELLO, Franklin DOS 2 deletes the file by writing the new one over it. At this point you have a new diskette with Franklin DOS 2 and the HELLO file on it.

To verify that all these procedures went as planned, use a catalog command for the new diskette. Type:

<div align="center">CATALOG, D2 <RETURN></div>

You should see the HELLO file listed as the only file in the catalog. Now move the new diskette to drive 1 and try rebooting with the <CTRL><OPEN F> <RESET> command.

Franklin DOS 2 Commands And Programs

This chapter contains information about Franklin DOS 2 commands and the utility programs included on the Franklin DOS 2 Diskette. There is also information on DOS 2 Error Messages.

This chapter presents some of the most commonly used commands and Franklin utility programs associated with Franklin DOS 2. A detailed listing of all the Franklin DOS 2 commands is available in the *ACE 500 Technical Reference Manual*. The utility programs discussed here are not part of Franklin DOS 2, the Franklin Disk Operating System, but they are contained on the Franklin DOS 2 diskette and will be used closely with Franklin DOS 2 in managing your diskettes and files.

Franklin DOS 2 Commands

RUN And BRUN Commands

Franklin DOS 2 offers a feature that simplifies using most programs. Most operating systems oblige you to specify the type of program you want to run, that is, you have to know whether the program is written in BASIC or binary language, or in some other kind of file format. Each file format then requires a specific RUN command for its type of files.

Traditionally, the RUN command is reserved for type F or BASIC files, and the BRUN command for type B or binary files. Franklin DOS 2 takes all that uncertainty out of the RUN command by substituting a single abbreviated RUN command, which is just — (a hyphen) and the filename. The Franklin DOS 2 — command may be used instead of any RUN, BRUN, or EXEC command. For example, you can run the HELLO program by typing:

— HELLO <RETURN>

You'll see the same messages that appeared on the screen when you booted the Franklin DOS 2 diskette.

This command has eliminated all the technical vocabulary and jargon associated with trying to figure out which of the RUN commands to use with each file. This is especially helpful to people who don't really want or need to know what the technical distinctions between the different kinds of files are.

CATALOG Command

The Franklin DOS 2 CATALOG command is one you know already. It lists the contents of the diskette you specify, along with the space available on the diskette and other useful information.

Franklin DOS 2 has an abbreviation for the CATALOG command. You can just type / instead of CATALOG. Try the abbreviation to get a catalog of the diskette in drive 2 by typing:

/,D2 <RETURN>

LOCK And UNLOCK Commands

The LOCK command protects files so you can't accidentally add to, delete, rename, or change them in any way. All the files on the Franklin DOS 2 diskette are locked. The asterisk in the catalog listing showed the locked status of the files.

The UNLOCK command removes this protection from a file so that you can make changes to it.

You can try out these commands with a copy of the Franklin DOS 2 diskette that doesn't have a write-protect tab on it. Type:

UNLOCK HELLO <RETURN>

Now type:

/ <RETURN>

You'll see that HELLO no longer has an asterisk in front of it. You can lock the HELLO file again by typing:

LOCK HELLO <RETURN>

REN And RENAME Commands

The RENAME command lets you change the name of a file. Franklin DOS 2 lets you abbreviate RENAME to REN. For example, to change the name of HELLO to BEGINNING PROGRAM, you would type:

REN HELLO, BEGINNING PROGRAM <RETURN>

Now type:

/ <RETURN>

to see the changed catalog.

Change the file back to its original name by typing:

<div align="center">REN BEGINNING PROGRAM, HELLO <RETURN></div>

If you try to rename a locked file, you'll get a LOCKED FILE error message. If you really want to rename the file, you just have to unlock it first.

(If you remember the INIT procedure to initialize a diskette, you were asked to name a file to run whenever the diskette was booted. The file you named was HELLO. If HELLO isn't on the diskette under the name HELLO, the boot procedure won't work, so it's important to change the file name back to HELLO.)

DEL And DELETE Commands

The DELETE command lets you remove unwanted files from your diskettes permanently. Franklin DOS 2 lets you abbreviate DELETE to DEL. Since Franklin DOS 2 contains no provision for restoring files, be careful when you use DELETE. All the files on your Franklin DOS 2 diskette are needed, so you don't have a practice file to delete, but the command is simple enough you don't need a test. The form of the command is:

<div align="center">DEL filename <RETURN></div>

If you try to delete a locked file, you'll get a FILE LOCKED error message. If you really want to delete the file, you just have to unlock it first.

INIT Command

INIT is the initializing command in Franklin DOS 2 that you already used in preparing bootable diskettes. If you want to know the explicit procedures for using INIT, refer to the preceding chapter of this manual or the *ACE 500 Technical Reference Manual.*

COPY Command

COPY is the Franklin DOS 2 command you used to copy individual files when you prepared bootable diskettes. The COPY command is similar to the Copy selection in FUD, though the COPY command requires a two disk drive system while the FUD utility program can use a double or a single drive procedure. Neither of these copy commands should be confused with the COPY and ONE DRIVE COPY programs which are individual utility programs used to copy entire diskettes.

Franklin DOS 2 Error Messages

When the DOS 2 operating system encounters an error in a command line or cannot execute a DOS 2 command, it responds with an error message. The error message tries to explain what the problem is so you can make the correction and go on with your program. Sometimes the error messages aren't entirely clear, so these brief explanations may help.

CAN'T COPY TO THE SAME DRIVE

This error indicates that the COPY command can't be used to copy a file to the drive it's already using.

DISK FULL

This error may result either from all the space on the diskette being taken by files, or by all the space in the catalog being full.

DIRECT COMMAND ONLY

This error occurs when the COPY command is used in a running BASIC program, a prohibited usage. COPY must be used directly from DOS 2 and for immediate execution.

END OF DATA

This error indicates that an attempt was made to read past the end of data in a file.

FILE LOCKED

This error indicates that a command, such as RENAME or DELETE, cannot be executed because the file specified has been locked.

FILE NOT FOUND

This error indicates that the filename specified cannot be located, either because the name is incorrect or the drive is incorrectly specified.

FILE TYPE MISMATCH

This error occurs when a file type specified doesn't match the type of file required by the command, such as trying to use a binary LOAD command with an ASCII (not binary) file.

I/O ERROR

This error indicates that there is some difficulty in input or output.

NO BUFFERS AVAILABLE

This error indicates that all the allocated file buffers are filled. A file must be closed or a buffer emptied to make a buffer available.

NOT DIRECT COMMAND

This error indicates that the command entered is not one which can be executed directly but must be executed by a running BASIC program.

PROGRAM TOO LARGE

This error indicates that the program specified in the BLOAD command is too large for the available memory. It can also mean that the program specified in a BSAVE command is protected and cannot be saved.

RANGE ERROR

This error signals that a numeric parameter is outside its permitted range of values.

SYNTAX ERROR

This error is a miscellaneous error for an incorrect command entry, or some other command DOS 2 cannot interpret.

VOLUME MISMATCH

This error indicates that the volume number found on the diskette does not match the volume number specified.

WRITE PROTECTED

This error indicates that some write command was attempted to a diskette which had its write-protect notch covered.

Franklin DOS 2 Utility Programs

Franklin DOS 2 programs are different from Franklin DOS 2 commands. Programs are separate files; Franklin DOS 2 commands are always part of Franklin DOS 2. Franklin DOS 2 programs are listed separately on the catalog of the diskette, the commands are not. Franklin DOS 2 programs have to be executed with the Franklin DOS 2 RUN, or —, command, while Franklin DOS 2 commands need a particular kind of command line, as with REN for renaming files. It doesn't really make a lot of difference to you whether an Franklin DOS 2 function is a command or program, except that programs have to be specifically copied onto diskettes, while commands are included automatically with Franklin DOS 2.

There are a number of utility programs on the Franklin DOS 2 diskette. You've already seen two of them, COPY and ONE DRIVE COPY.

The Franklin DOS 2 COPY Program

COPY will copy any Franklin DOS 2 (or other DOS, CP/M, or Pascal) diskette on a two disk drive computer. It works with any diskette format, but the source and destination diskette formats must be the same. All you have to do is put the original in one drive, a blank diskette in the other drive, and hit the proper key. COPY formats the destination diskette, copies the entire contents of the source diskette onto it, and then verifies that the copy is a good one.

The Franklin DOS 2 ONE DRIVE COPY Program

The ONE DRIVE COPY program is slower than COPY, but it works on computers with only one disk drive. It works with any diskette format, but the source and destination diskette formats must be the same. It also performs three functions, formatting, copying, and verifying at the same time.

Other copy programs can copy single files, like the Franklin DOS 2 COPY command or FUD's Copy. Some programs, like FUD, can also work with unlike diskette formats, so you can copy a file (or a whole diskette) from 35 track to 40 track format diskettes.

FUD — A General File Management Program

The letters FUD stand for Franklin Utility for Diskettes. FUD lets you remove unused files, make copies of files, determine if your files are still readable, and perform other mundane but essential functions.

To run the FUD program, boot a copy of an Franklin DOS 2 diskette and type:

— FUD <RETURN>

After a few seconds, you'll see the FUD opening screen:

FUD version x.x
by
Franklin Computer Corp.

MAIN MENU
S — Show files on diskette
C — Copy files
D — Delete files
L — Lock files
U — Unlock files
V — Verify files
F — Format diskette
Q — Quit

Please press letter for function desired:

This is FUD's main menu, the list of options open to you. The screen line where the words MAIN MENU now appear always lets you know what you're doing. As you use each of the options, the information displayed on this line changes so that it corresponds to the function you're performing.

To choose one of FUD's options, just type the letter to the left of the hyphen for the option that you want. You don't have to press <RETURN> to begin the procedure.

Diskette Management Commands

Selection F (Format A Diskette)

Every diskette must be initialized or formatted before you can use it. The COPY program you used to make your backup diskettes formatted the diskettes automatically as part of the copying procedure. INIT also formatted the diskette as part of its job. Each of those functions did formatting as part of a larger task with other functions that you may not want or need. This formatting command performs only the formatting function.

Formatting prepares a diskette so you can store data files on it. This option gives you the maximum storage space for a diskette because it adds no other files, such as Franklin DOS 2 boot files. The computer can use formatted diskettes but cannot boot from them.

The F formatting option allows you to choose the diskette format you want, either 35 or 40 track.

Select F from the MAIN MENU by pressing <F>. The screen displays the Format title and asks for the disk drive you want to use for the formatting operation:

<div align="center">

FORMAT
Which drive has diskette to format?
enter 1, 2, or RETURN:
</div>

The Default selection is shown under the blinking cursor and will be chosen if you press <RETURN>. Be careful because the default drive is probably the drive you used to run the FUD program and probably still contains your Franklin DOS 2 diskette. You have an opportunity later to stop the procedure, or you can remove your Franklin DOS 2 diskette now, since the FUD program has been loaded into memory.

After you respond with the disk drive number, FUD asks for the diskette format you want:

<div align="center">

Which disk format would you like?
1) 35 track, single sided
2) 40 track, single sided
3) 40 track, double sided
enter 1, 2, or 3:
</div>

Press the number corresponding to the diskette format you want, either 1 or 2. Option 3 doesn't apply to ACE 500 disk drives.
FUD then asks for a diskette volume number:

What volume number would you like on the diskette? enter 0 — 254 :

The volume number is an identification number placed on each diskette when it's formatted. Although the number can range from 1 to 254, most people just use 0 to avoid confusion because Franklin DOS 2 accepts 0 to mean any volume number. The Default choice, which is shown under the blinking cursor, is 0 and can be chosen by pressing <RETURN>.

This message now appears on the screen:

Please place the diskette to be formatted into slot 6, drive #x then press RETURN to continue. Press CTRL-C if you wish to quit:

If you've changed your mind about formatting this diskette, you can hit <BREAK> or <CTRL> C and you'll see FUD's MAIN MENU again. Otherwise, be sure that you put the correct diskette into the drive specified. Formatting a diskette destroys all the information on it. It's a good idea to see a catalog of each diskette before you start the formatting procedure.

When you're sure that everything is ready, press <RETURN>. The formatting operation takes about 15 seconds. You see this message on the screen below the previous message:

Formatting a diskette, relax a minute

You get this message when the format operation is finished:

The diskette has been formatted.
Press RETURN to format another diskette or CTRL-C to exit.

The diskette is now ready to store programs or data. You can press <RETURN> to format another diskette or <CTRL> C to return to FUD's MAIN MENU.

File Management Commands

FUD has six file management commands:

> S (show the files on a diskette)
> C (copy files)
> D (delete files)
> L (lock files)
> U (unlock files)
> V (verify files)

Each option can operate either on single files or on groups of files. Whenever a file management function requires a file name, you can enter either a single file name so the command operates on that file alone, or you can enter what's known as an ambiguous file name which can stand for a group of individual files.

A file name is ambiguous if it has either of the characters ? or = in it. The ? character means "any character is acceptable." For instance, the ambiguous file name A?E refers to all file names three characters long beginning with A and ending with E. It could refer to ACE, APE, and AXE, but not ICE or APPLE.

The = character means "any group of characters is OK." For example, a file name of A= means any file name starting with the letter A. The hypothethical file name JOHN'S=STUFF could refer to files like JOHN'S OLD STUFF or JOHN'S VALUABLE STUFF or even JOHN'SSTUFF, in which = refers to no characters at all.

Any number of ?s and =s can be used in a file name, as long as you can keep straight what the name you've created really refers to. It's possible to create an ambiguous file name that includes more file names than you intended. For example, by using an overly ambiguous file name when deleting files, you could eliminate files you really wanted to save.

FUD tries to protect you from such calamities. Whenever you use an ambiguous file name, FUD poses the question: "Ask before processing each file?" before it begins to execute the command. If you answer Y, a question mark will appear to the right of each file name included in the ambiguous file name you've specified. Three options are open to you: press Y and the command will be performed for that file, press N and the command will not be performed on that file, or press <BREAK> or <CTRL> C to end the command.

FUD also usually tries to perform all operations to the diskette in the last drive that you specified. The last drive you used is called the default drive. If you want the

command to use a diskette or file in another drive, simply type ,D1 or ,D2 after the file name to select the correct drive for that file.

Selection S (Show The Files On The Diskette)

This option is like the CATALOG command. It lets you see what files are on the diskette, and it's convenient to have this command in FUD because you can use it with the other file management commands. The other advantage of the Show Files command is that you can use ambiguous file names to restrict the list to a specific group of files so you can handle groups of files as units.

When you select S, the screen clears and then displays this message:

> Show diskette catalog
>
> which file(s)?
> RETURN for all files
> CTRL-C to abort
> ESC for help
> Defaults: Slot 6, drive 1, volume 0
> :

Start by pressing <RETURN>. Disk drive 1 should start up, since you haven't specified anything otherwise. The names of the files on the diskette in drive 1 appear on the screen. The display is very similar to the one you saw when you typed / <RETURN> or CATALOG <RETURN>. To see all the files on drive 2, you could type:

> =,D2 <RETURN>

If you type an ambiguous file name, FUD will display all files on the diskette that match that description. If the file name you type is not on the diskette, or if no files match the ambiguous file name description, you'll get the message:

> *no files found*

The <ESC> selection on the menu shows a help message that reminds you of the rules for file names, identification parameters, and ambiguous file name symbols.

If FUD must display more files than will fit on the screen at one time, the flashing cursor appears at the bottom of the list that is displayed as a signal that there is more

to come. You can display the rest by pressing any key, or press <CTRL> C to stop the display.

After all the files are displayed, the total of all their file sizes is shown, along with the number of unused sectors (256 characters each) remaining on the diskette. The question

<div align="center">

all done?

</div>

appears. If you answer N, you're asked which files to display. If you answer Y, the main menu of FUD appears on the screen.

The other five selections in FUD work much the same way.

Selection D (Delete Files)

The D option in FUD performs the same function as the DELETE command described in the Franklin DOS 2 section, except that with D, you can specify ambiguous file names.

The D option will not delete locked files. If you try to do so, you'll get an error message.

WARNING: Be particularly careful when using this command with ambiguous file names. It's easy to wipe out entire diskettes this way. FUD will give you a chance to decide whether you want the files displayed individually so you can decide in each case if you really want to delete that file. You could also use the Show Files command with your selected ambiguous filename to see which files are matched to the description given.

Selection L (Lock Files)
Selection U (Unlock Files)

The L and U options perform the same functions on files as the LOCK and UNLOCK options described in the Franklin DOS 2 section. The FUD versions have the advantage of using ambiguous file names.

Selection V (Verify Files)

The V option checks whatever files you select to make sure that they're still readable.

If the FUD program can read all your files without any errors occurring, you can be fairly confident that all the data is still in good condition.

Selection C (Copy Files)

The C option lets you copy a file or group of files from one diskette to another. You're first asked which files you want to copy. If you enter a single file name, only that file is copied. If you specify an ambiguous file name, all files that match the ambiguous description are copied. You're then asked which drive you want to copy the file or files to. If you want to change the drive number from the one that the screen displays, simply enter the appropriate number (e.g. ,D2) before you press <RETURN>.

You can also change the normal slot number or volume number at this point. Should that ever be necessary, just enter the information after the file name. For example, ,S4,D2,V33 would specify a copy of the file to slot 4, drive 2, volume 33.

If your source drive, the one holding the diskette you're copying from, is different from the destination drive, the drive holding the diskette you're copying to, then the copying process goes on without any diskette changes.

If you only have one disk drive, you must specify that drive as both the source and the destination drive. FUD guides you through the diskette exchange procedure, but you have to make sure you don't accidentally place the wrong diskette in the drive. You'll definitely ruin the copy of the file you're making, and you may ruin the entire diskette.

FUD recognizes all the Franklin DOS 2 formats automatically, so you can copy from a diskette of one format to another diskette of another format. In fact, option C and the Franklin DOS 2 COPY command are the most convenient way to do this. Because FUD allows the use of ambiguous file names, you can copy the entire contents of a diskette by using = as the file name to be copied.

If a file with the same name as the file to be copied already exists on the destination diskette, the message:

<p align="center">already exists, replace it?</p>

appears. If you answer N, no copy is made. If you answer Y, the copy is made and the new file writes over the old file, deleting it permanently, even if it's locked. Be careful.

BOOT13 Program

This utility program allows you to boot the older format 13-sector DOS 3.2 diskettes. If you have some of these older format diskettes, you can run BOOT13 by typing:

<div align="center">— BOOT13 <RETURN></div>

Then insert the 13-sector diskette in drive 1 and respond to the screen prompt with <RETURN>. The diskette will boot.

CHAIN And CHAIN2 Programs

The CHAIN programs are used by programmers to chain programs together into a single program. The programs must be in Floating Point BASIC language.

VERIFY Program

The VERIFY Program reads an entire diskette to determine the integrity of the data stored on it. If you are having difficulty using a particular diskette, use VERIFY to determine whether or not the diskette is faulty.

With the Franklin DOS 2 diskette in drive 1, type:

<div align="center">— VERIFY <RETURN></div>

and follow the instructions. VERIFY will not affect the data stored on the diskette.

DIAGNOSTICS Programs

The diagnostics programs are used for analyzing problems you may have with your computer. There are separate diagnostic programs entitled RAM, DISK, GRAPHICS, KEYBOARD, and PADDLES. Each of these individual diagnostic tests is entered through the DIAGNOSTICS program. Type:

<div align="center">— DIAGNOSTICS <RETURN></div>

Then select the test you want from the menu displayed.

RAM Disk Programs

The ACE 500 has 256K of RAM memory built in, or 512 K of memory in the optional configuration. This memory size is sufficient to use part of the RAM memory as a temporary disk drive where programs or files can be stored for very quick access. This temporary disk drive is called a RAM disk. It is created and operated by the FRAMDISK program on the Franklin DOS 2 diskette.

The memory also has the correct configuration to emulate several popular extra-memory peripheral cards, such as RAM Works, which have special programming features and are used by some application software. The applications of these memory features are not discussed in this section. Refer to the *ACE 500 Technical Reference Manual.*

This section describes the FRAMDISK program and a number of other utility programs associated with operating a RAM disk. There are six programs in the RAM disk group: RAMTEST, FRAMDISK, PRAMDISK, and the AWSTARTUP11, AWSTARTUP12, and AWSTARTUP13 programs. Each of these will be considered in turn. Some of this more detailed information is intended for advanced applications and may not be needed for using the RAM disk.

The FRAMDISK Program

FRAMDISK is a program to create a RAM disk (a high speed disk drive using the extra memory available in the ACE 500 computer) to use with the DOS 2 operating system.

With some important exceptions, the RAM disk you create with the FRAMDISK program behaves just like a regular disk drive which you access by using normal operating system commands like CATALOG, etc. with the designated slot number.

The most important difference between RAM disks and regular disk drives is that RAM disk storage is temporary. Data or programs stored on the RAM disk are lost whenever the power is turned off or when the operating system is changed. Information stored on the RAM disk must be saved to a permanent storage medium before it is lost by one of those events. Since many commercial application programs use their own operating systems, it is necessary to take care of the contents of the RAM disk before turning off the power or changing an operating system.

You can create a RAM disk simply by running the FRAMDISK program:

— FRAMDISK <RETURN>

The screen then displays:

DOS 5.X RAM disk driver VX.X
Installed successfully - now using
expansion RAM for a RAM disk in slot 3.

When you run the FRAMDISK program you can include some additional parameters on the DOS 2 command line, such as:

— FRAMDISK {3 {file1 {file2} <RETURN>

This command line creates a RAM disk by running the FRAMDISK program and installs it in slot 3. It also interprets the parameters as commands to copy file1 and file2 onto the RAM disk from the diskette that contained FRAMDISK.

The command line may be used with or without parameters. If you don't specify a slot number, the command uses slot 3 (called the default value). Obviously, you can't use a slot number which already has an assigned peripheral device, or a slot number greater than 7. If you don't list file names on the command line, none are loaded at that time. If you do use parameters, you must use the curly brackets as separators. No spaces are allowed in the parameter string except where they occur in filenames. Here are some other examples:

— FRAMDISK — installs the RAM disk in slot 3 and loads no files.

— FRAMDISK {ED} — installs the RAM disk in slot 3 and loads a file called ED.

— FRAMDISK {4} — installs the RAM disk in slot 4 and loads no files.

— FRAMDISK {5{BUG {ASM} — installs the RAM disk in slot 5 and loads files called BUG and ASM.

The FRAMDISK program automatically calculates the amount of memory available for the RAM disk (which is the amount of RAM installed in the computer less 64 K, since bank 0 is not used for the RAM disk). A CATALOG of the RAM disk will show the number of free sectors which you can multiply by 256 to obtain the number of bytes of RAM available on the RAM disk. The RAM disk catalog allows only 56 filenames; any more filenames will produce a "disk full" error message even if there is memory space available for the files.

If you have the RAM disk set up in slot 3, any of the normal DOS 2 commands can use it like a normal disk drive by specifying the correct slot number with the file name. For example, these commands would use the RAM disk:

CATALOG, S3
— FUD, S3

When you run the FRAMDISK program, the RAM disk is cleared (initialized). If the RAM disk is somehow disconnected but not initialized, for instance by loading an application program which destroyed DOS 2, it is sometimes possible to call or logically select the RAM disk without initializing or losing the data stored on it. This is done by running the FRAMDISK program but specifying a slot number of zero. This will connect the RAM disk to the operating system but not initialize the memory banks in slot 3. Then try to access slot 3 (or the slot of the previous RAM

disk), for instance with the CATALOG command, to see if you can get access to the data. Obviously, the success of this operation depends on what kind of event broke the connection with the RAM disk.

It is a prudent practice to use the RAM disk for storing programs that you have stored elsewhere on diskettes rather than using it to store data. The advantage of the RAM disk is its speed of access and loading. Its disadvantage is the temporary nature of its storage. Both of these situations recommend the RAM disk for storage of frequently used programs rather than for storage of data.

The RAMTEST Program

RAMTEST is a diagnostic program to test the RAM memory in the ACE 500 computer. The program automatically detects the amount and configuration of the memory and tests all of it until stopped by your command. When you run RAMTEST the screen displays all the possible banks of RAM memory (these are hexadecimal numbers corresponding to 64 K banks of memory for the 256 K ACE 500 computer) with the tested status of each bank.

The status indicators are OK, meaning there is no error; BAD, meaning an error has been detected in that RAM bank; or "—" which indicates that that bank is not available in the current memory configuration. Intermittent or "Soft" errors are usually shown as BAD, while more persistent or "Hard" errors will be shown as the absence of that bank of memory.

The screen display also shows the total number of errors found in all the RAM banks. The test shows specific information about the last memory error detected, but this information usually has to be deciphered by an engineer. You can make the test pause and hold the error display by pressing the space bar. The test resumes when you press the space bar again.

The test runs continuously until you press the ESC key to stop the RAMTEST program.

RAM Disk Utilities For Apple Programs

The next four utility programs in the RAM disk group work ONLY under the Apple ProDOS operating system and cannot be used under DOS 2. These programs can be found on the Extended Memory Utility which is on the flip side of the Franklin DOS 2 diskette.

The PRAMDISK Program

This program creates a RAM disk under the ProDOS operating system. PRAMDISK automatically installs the RAM disk as slot 1, drive 1, and gives it a volume designation of /ACERAM. It does not use parameters in its command line and

won't accept other slot, drive, or volume designations. PRAMDISK must have at least 128 K of RAM in the computer to run. It uses the amount of RAM available for the RAM disk less 64 K (bank 0 of the memory is used for other purposes).

The volume directory of the RAM disk created can contain a maximum of 51 files, any or all of which can be used as subdirectory files if desired. Files may be copied to or from the RAM disk by means of the ProDOS FILER program or some other copy program, or by a start up program like the AWSTARTUP program described in the following section.

The AWSTARTUP Programs

The following AWSTARTUP program information applies only to an ACE 500 equipped with the 512 K RAM option. If an attempt to run Appleworks installed with AWSTARTUP is made on a 256 K RAM computer the following error message will be seen: Sorry, you need the 512 K RAM option to run AWSTARTUP. Installation failed!

There are three versions of the AWSTARTUP program on the Franklin DOS 2 diskette, AWSTARTUP11, AWSTARTUP12, and AWSTARTUP13, which correspond to Versions 1.1, 1.2, and 1.3 of the Appleworks program. Use the utility program which corresponds to the version of Appleworks you have. The following procedure will treat them as the same program.

AWSTARTUP is a BASIC program which runs under ProDOS that automatically runs the PRAMDISK program to create a RAM disk, and then loads and runs all the Appleworks files from the RAM disk. This technique allows you to run the Appleworks programs from the RAM disk and have a disk drive available for your data diskette. Using the RAM disk also significantly increases the execution speed of the programs.

In order to use the AWSTARTUP program, you must first make a start up diskette. You can make a start up diskette by making a copy of the Appleworks start up diskette, then add the following files to the new diskette: BASIC.SYSTEM, PRAM-DISK, and AWSTARTUP. The BASIC.SYSTEM file is taken from the Apple ProDOS Utility program Version 1.1.1. or later. Transfer the PRAMDISK and AWSTARTUP programs from the Extended Memory Utility diskette. Then rename the AWSTARTUP program AW and rename the APLWORKS.SYSTEM file AW on your start up diskette. This completes the process. You can use this diskette as a start up diskette. It will create a RAM disk automatically and load the Appleworks programs on it.

To run the Appleworks programs, insert this start up diskette in the disk drive and boot the system, and you'll see the Appleworks opening screen. The diskette will prompt you when to swap the start up diskette and the Appleworks diskette. When you are finished you can store both diskettes, since the Appleworks files needed from them are stored on the RAM disk.

There is an important thing to notice about this technique. Any changes you make in the Appleworks configuration (for printer settings, for instance) are only made in the RAM disk version, and thus are only temporary. If you want to make these changes permanently, you must make the changes in the diskette version of the file, or copy the modified printer configuration file SEG.PR from the RAM disk to the Appleworks diskette.

Chapter 6

Introduction To Programming

This chapter gives an overview of what programming is and goes through some simple applications for your own programs. This introduction is a very limited tutorial for some elementary BASIC programs, including a Hello program for programming the function keys and for a boot diskette.

This chapter is a discussion about BASIC programming. Specific listings and definitions of BASIC commands are given in the following chapter.

Writing a program simply means giving the computer some instructions to do something. You can think of a Franklin DOS 2 command like CATALOG as a program, although Franklin DOS 2 does the work and all you do is type CATALOG to run it. As you need to solve more complicated and personalized problems, your need for more complicated commands increases. Running CATALOG, deleting old files, renaming some files, etc. could be thought of as a program (that is, a list of instructions) that you're doing one step at a time. Programming is really a fairly common experience.

Some people think that using a computer requires the ability to write programs for it. They've heard that programming is a difficult job, and so they're hesitant to go near a computer. In fact, that's not true at all. You've seen that there are many things you can do with your ACE 500 Computer by using a few simple commands. Thousands and thousands of application programs are already written for you. They use the same techniques of menus and questions to help you along. None of that involves programming at all.

On the other hand, programming can be fun. You don't have to become an expert programmer or devote a lot of time to it if you don't want to, but you might find that it's rewarding to create your own solution to a problem, or generate something of your own.

The aim of this chapter is to remove a little of the fear of programming and to suggest a few simple things to do with Franklin BASIC and Franklin DOS 2. This chapter cannot substitute for a course in programming or an instructional book on the subject.

For those who are familiar with computers and have some experience in programming, this chapter will point out a few things you need to know about Franklin BASIC. More details, including the command listings for Franklin DOS 2 and Franklin BASIC, are available in the next chapter.

The BASIC Programming Language

The first thing to recognize is that BASIC programming language is just that, a language. The language takes one word and translates it into another word or group of words. It takes familiar English vocabulary, uses it in particular ways with a kind of grammar that specifies correct and incorrect forms for statements, and turns it into computer language.

For instance, a command like PRINT is interpreted by BASIC as a series of operations, something like "Search for the thing to be printed. Take the thing to be printed; store it in this part of memory. Take it from memory at the right time; send it to the printer at the designated output location in such and such a format." The advantage of BASIC as a programming language is that it allows you to ignore all those intermediate steps and simply write PRINT, a command you understand as plain English.

Franklin BASIC

Franklin's BASIC is designed to be compatible with most common BASIC programs, especially Applesoft BASIC programs. Franklin BASIC differs from Applesoft BASIC in that it supports and displays lowercase letters, which Applesoft does not. Franklin BASIC doesn't have the commands found in Applesoft BASIC that control cassette tape drive storage. Franklin BASIC has the FKEY command, which is not available in Applesoft BASIC, to program the function keys on your ACE 500 computer. In all other ways, Franklin BASIC and Applesoft BASIC are the same, so any programs or materials using one language will work with the other.

Unfortunately, not all BASIC languages are so compatible. There are many dialects of BASIC, some that are linked to other kinds of hardware, some that are from different manufacturers. In many of these BASIC languages some things are the same, like LOAD and SAVE commands. Some things are only slightly different, like using "?" to abbreviate PRINT in Franklin BASIC. Other things, like "clear screen" commands, vary greatly depending on the BASIC and on the computer hardware.

When you see a BASIC program, or want to purchase programs or books on BASIC, you should be aware of the different dialects of BASIC. Use the same compatibility criteria that you use for application software.

Using The Franklin BASIC Programming Language

To program successfully, you have to realize that the computer and Franklin BASIC are very literal in interpreting your statements. The computer doesn't know that Bill's name doesn't really have three L's in it, or that you meant to put a comma where you typed a period. The computer takes things exactly as it finds them, so be careful and follow Franklin BASIC grammar and syntax carefully.

The other key concept in programming is the idea of sequence. Commands must be given in order, and the computer worries a lot about finishing one thing before going on to another, so it wants all commands in a program to be numbered in order. You'll learn how to do that shortly.

The only other thing to programming is learning the commands, and they're mostly simple words you've seen before. Putting the commands in proper order to solve a problem is the creative task for you.

BASIC And DOS 2 Commands

There may be some confusion about BASIC, a programming language, and DOS 2, a Disk Operating System. They both have a lot of commands and sometimes it's hard to tell whether a command is part of one or the other. If you're just using the computer it usually doesn't matter. If you're writing a program, it does matter, and you have to label the DOS 2 commands. The BASIC commands and the DOS 2 commands are listed in their respective chapters of this manual.

The other thing to keep in mind is that it is DOS 2, the Disk Operating System, that is loaded from the diskette when it is booted. And it is DOS 2 that controls input and output from the diskette. Franklin BASIC is always resident in the computer's memory, even if you don't boot from the diskette.

Creating A Simple Program

Try writing a simple program. Just follow the instructions exactly as they're given below, without inserting any punctuation or typing errors, and you'll do fine.

Make sure you've booted a backup copy of your Franklin DOS 2 diskette (without a write-protect tab, so you can write to it) in the disk drive of your ACE 500 Computer.

When you see the prompt and cursor on the screen, type:

New <RETURN>

The NEW command clears any programs out of memory so you are ready to type in a new program. Now type:

```
10 Print "This Disk Belongs To"
20 Print "Franklin User" <RETURN>
```

You can substitute your name within the quotation marks in line 20. Notice that each line in BASIC must end with a <RETURN>. Then continue typing:

```
30 Print "Its Contents Are" <RETURN>
40 Print "<CTRL><D>"; "Catalog" <RETURN>
```

Make sure you hold down the <CTRL> while you press the <D> key. You shouldn't see anything between the quotation marks because <CTRL><D> is a non-printing character. By typing the <CTRL><D> in line 40, you tell Franklin BASIC that the rest of the line is an Franklin DOS 2 command, which is CATALOG in this case. All Franklin DOS 2 commands must be preceded by <CTRL><D> or its ASCII equivalent when used in a BASIC program.

That's your program. When you see the prompt and cursor, type LIST to see what you've typed appear on the screen. Retype any line that contains an error, then type:

```
— <RETURN>
```

to RUN the program. It's fun to see how these simple commands produce a customized diskette title, or label if you want to print it.

To store this program on the Franklin DOS 2 diskette (using the name A BASIC PROGRAM), type:

```
Save A Basic Program <RETURN>
```

You'll see the disk drive work for a few seconds. Then when the prompt and cursor reappear, type:

```
List <RETURN>
```

Your program will reappear, so you know it's still in memory. Also, Franklin DOS 2 has saved a copy of it and stored it on the Franklin DOS 2 diskette. Now type:

Save Another Basic Program   <RETURN>

This command saves the current contents of memory, which is the identical program, on the diskette. It doesn't matter to Franklin DOS 2 that there are two identical programs on the diskette as long as their names are different.

When the prompt and cursor reappear, type:

/ <RETURN>

The CATALOG command shows the diskette has both A BASIC PROGRAM and ANOTHER BASIC PROGRAM stored on it. Now type:

New <RETURN>
List <RETURN>

The NEW command has removed the program from memory, so the LIST command had no program to list, and nothing appeared. To put the program back in memory again, you have to read it from the diskette. Type:

Load A Basic Program   <RETURN>
List <RETURN>

And your program is back in memory. If you only wanted to run the program, you could have used an Franklin DOS 2 command (—A BASIC PROGRAM <RETURN>) to run it directly from the diskette. If you want to change the program, you have to load it into memory with the LOAD command.

This exercise has given you some idea about how a program is constructed. First, you've seen the difference between having a program in memory, where it's temporary, and having it stored on the diskette. You've learned how to use the Franklin BASIC commands NEW, LOAD, SAVE, and LIST. You have a little familiarity with using PRINT statements and using Franklin DOS 2 commands in programs.

You've also seen how the structure of the program plays an important part in its operation. The line numbers you put on the left side of the line give the correct sequence of operation of the program commands. If you wanted to add a line, you could type it out of sequence with a line number that showed its correct order and

Franklin BASIC would place it in the proper place when you gave a LIST command. A technique useful to programmers is having a command in one line refer to an earlier line, thus creating a loop of commands that process data through the same operation at the computer's speed.

If you made any typing mistakes, you've also seen how fussy Franklin BASIC is that the grammar of your statements is correct. The SYNTAX ERROR reply is very familiar to programmers.

This example showed you how Franklin BASIC and Franklin DOS 2 are related and interconnected, but still separate parts of the computer's operations. Using an Franklin DOS 2 command is permitted in Franklin BASIC, but it must be identified as such.

Some Programming Suggestions

You may think that this little program is really too simple for any practical use, but it can be modified into a handy tool. You can see how to change the actual display— just change the PRINT lines, either in number or in content, to say whatever you want.

Next, when you create bootable diskettes, use this little program as your HELLO program. All you have to do is name it HELLO (deleting the other HELLO program first) and put it on the diskette when you initialize it. Franklin DOS 2 will run it each time the diskette is booted.

A further refinement is to use this little program to run another program, such as an application program on a particular diskette, or a function key definition program. In this case, you would use the <CTRL><D> sequence to identify an Franklin DOS 2 command, but instead of CATALOG, the command could be something like "—A Function Key Definition Program". This way you could load your function key definitions for each diskette automatically each time Franklin DOS 2 boots the diskette. Refer to Chapter 8, *Function Keys*, for some suggestions on such a program.

This introduction to programming has taken a program which was simple in design and execution and shown you how effective it can be in the way you operate your computer. Of course there is much more to Franklin BASIC and to writing programs than this, but this exercise should show you how easy and rewarding programming can be.

Chapter 7

Franklin BASIC

This chapter presents specific information about Franklin BASIC for users who are somewhat familiar with programming techniques. This material is not intended to substitute for instructions on how to write programs or technical information about software procedures.

This chapter also describes features of Franklin BASIC and the ACE 500 computer which are unique, such as the BASIC editing keys on the keyboard and special Franklin BASIC commands and abbreviations.

The commands in this chapter show you how Franklin BASIC works. Together with a tutorial on BASIC programming, they will greatly expand your use and command of the Franklin ACE 500 computer. Each command is given with a simple definition of its purpose and an example of usage. If particular syntax or parameters are involved, these are discussed briefly. The intention is to give you an idea of the contents of Franklin BASIC commands as they are similar to or are different from other general BASIC programming commands. The chapter concludes with a listing of BASIC reserved words and error messages.

What Is Franklin BASIC?

Franklin BASIC is a programming language designed for use with the Franklin ACE computers and Franklin DOS 2. It is written to be compatible with Apple's Applesoft BASIC language. Programs that are written in Applesoft BASIC will work on Franklin computers and vice versa. The instructions and commands are very similar, though Franklin BASIC has some additional features (such as lowercase letters) that make using it a little easier.

Compatibility Of BASIC Languages

Although Franklin and Applesoft BASIC languages are very similar, other BASIC languages differ somewhat. These other dialects of BASIC are sometimes linked to hardware differences between machines of different manufacturers, others are simple differences in proprietary design. Many of these language dialects share some things, like LOAD and SAVE commands, but differ on others. Some are slightly different, like Franklin BASIC's abbreviation of the PRINT command as "?". Other commands, like color type and screen addressing commands, are generally much more hardware dependent.

When you see a BASIC program, or want to purchase programs or books on

BASIC, you should be aware of these different dialects of BASIC. Use the same compatibility criteria that you use for application software.

General Remarks On Writing BASIC Programs

The commands that follow are all part of Franklin BASIC. The commands in the preceding chapters were part of Franklin DOS 2. Both types of commands can be used in your programs if you designate the Franklin DOS 2 commands with the <CTRL><D> character in the program line.

Franklin BASIC can use upper or lowercase alphabetic characters, while Applesoft BASIC can use only uppercase characters.

Each program line must have a <RETURN> at the end to indicate that it is a complete instruction statement.

Mathematics In BASIC Programs

BASIC programs recognize normal mathematical symbols and perform mathematical operations directly without program statements. Each statement must be concluded with a <RETURN> to indicate it is complete. Statements must use the Print command to display their result immediately.

+ designates an addition operation
– designates a subtraction operation
* designates a multiplication operation
′ designates a division operation
⌢ expresses an exponent

Large and small numbers are written in exponential notation, either n.nnnnE +11 or m.mmmmmmE-09. Numbers are rounded at nine digits.

BASIC statements Sin, Cos, and Tan represent trigonometric functions for Sine, Cosine, and Tangent relations with the argument given in radians. Atn represents the arc tangent function.

Log represents the natural logarithm function. Exp (x) is the exponential of x. Abs (x) is the absolute value of x, Int (x) is the integer value of x, and Sgn (x) is the sign of x. Sqr (x) is the square root of x.

Mathematical operations are performed in this order of priority: negative numbers, exponentials, multiplications and divisions in sequence from left to right in the statement, then additions and subtractions in sequence from left to right. If there is any potential for confusion in a statement, use parentheses to delimit mathematical

groups. BASIC can interpret nested sets of parentheses accurately.

Some examples:
    Print 15 / (2 + 3) returns 3
    Print 5 + (1 + 2 ^3) * (8 – 7) / 9 returns 6
    Print Sin ^2(A) + Cos ^2(A) returns 1

Special ACE 500 Computer Programming Features

The Franklin ACE 500 Computer has some special features that make programming a little bit easier for you. In addition to the user features in Franklin BASIC and Franklin DOS 2, there are a number of special keys on the keyboard which are useful for BASIC program editing, specifically those special keys marked on the numeric keypad.

These editing functions for BASIC programs are part of the enhanced video software, which is discussed in detail in Chapter 9, *The Display*. The editing functions of the special keys are listed below.

One Key BASIC Editing Commands

The special editing keys are located on the numeric pad.

The keys are active as editing keys when the NUM LOCK key is off, and when the enhanced video software is activated. The enhanced video software can be activated by turning On 80 column mode video display, which can be done with function key F6 or the command PR #3 <RETURN>. These commands are marked on keytops in the cursor keypad where they are convenient for editing. In a way, these keys are like special function keys in that they have special definitions but they are different because they are functional only when the enhanced video software is active.

RUN—Run A Program — This command is a one key command identical to the Franklin DOS 2—command or the Franklin BASIC RUN command.

LIST—List A Program — This command is a one key command to display the program listing.

ENTER—Return — This command is the same as the RETURN key.

CLRS—Clear The Screen — This command clears the entire screen and moves the cursor to the home position, column one of line one.

HOME—Home The Cursor — This command moves the cursor to the home position, which is the first column of the first line.

CLRL—Clear The Line — This command clears the entire line designated by the cursor position. The cursor position itself is unaffected.

INSC—Insert A Character — This command inserts a space character at the current cursor position and pushes other characters on the line one position to the right to make room for it. Only characters to the right of the cursor column are affected. If there is a character in the last column of the line, that character is lost.

DELC—Delete A Character — This command deletes the character at the current cursor position and moves all the columns on the right side one space from the right to close up the line. A space character appears in the last column of the line.

CPES—Copy Entire Screen Line — This command is useful in editing BASIC programs and has a very specialized operation. This command sends all the characters on the cursor line to the current program for processing or execution. This is equivalent to moving the cursor to the end of the line and then hitting <RETURN>. The CPES command works regardless of the cursor position on the line.

CPES is designed for editing BASIC statements. Suppose you had typed a line of a program that had a mistake in it. First, you would LIST the line, then use the escape mode cursor movement keys to locate the section of the line that needed correction, then make the changes with the INSC and DELC keys. When the changes are complete, just press CPES. The corrected line is sent to BASIC with a <RETURN> without any additional cursor movement required.

If you use the CPES command on a full line (all 40 or all 80 characters are used), you may notice that the last character on the line is cleared to a space character after CPES executes. Don't worry, this is only the display screen's processing of the command. The correct and complete line was sent to the running program.

Alphabetic Listing Of BASIC Commands

This listing gives the command statement and any arguments or parameters that can be used with it. If more than one form is valid, examples of each form are given. The command statement is defined and some simple usage of the command given.

Asc (a) ASCII function—This produces the decimal number representing the chart location of the ASCII character entered. ASC (A) = 65

Call–nnnnn—Executes the subroutine which is located at the decimal address specified. Call –nnnnn calls the routine at nnnnn.

Catalog,Sn,Dn—Franklin DOS 2 command to list the files on a diskette. Catalog can be abbreviated as / .

**Chr$(nnn) Character function**—This produces the character representing the decimal ASCII number entered. Chr$ (65) = A

Clear—This command clears all string and array variable values to zero.

Color = n—This command specifies the color for low resolution graphics, with n = 1 to 15. Color is initialized by the GR statement.

Cont—The continue statement resumes program execution after a BREAK command, Stop, or End.

Data—The data statement specifies a list of entries (numbers, characters, or strings) for use by Read statements.

Del mm, nn—The delete statement removes lines from a program. The beginning and ending line numbers must be shown.

Delete filename, Dn—The Franklin DOS 2 command deletes files from a diskette in drive Dn. It can be abbreviated to Del.

Dim Name (nn)—A dimension statement allocates memory space for storage of an array. Dim Label (100) allocates space for the array Label containing 101 elements. Dim List$ (20) allocates space for 21 strings in the array List. If no Dim statement is given, an array is assigned memory space for 11 elements.

End—This statement ends a program.

Fkey n, "defn"—This command enters a definition "defn" for the function key n (1 to 12) on your Franklin computer.

For—The For statement is used with a Next statement to form a loop. For A = 1 to 10 ... Next A performs the instructions in the loop 10 times, then goes to the next instruction after A.

Gosub nnn—This command causes the program execution to branch and go to the subroutine at line number nnn. If the subroutine ends with a Return statement, the program returns to the line following the original Gosub instruction.

Goto nnn—This statement causes the program to jump to the line number specified by nnn.

Gr—This statement turns on low resolution graphics, clears the screen, and sets Color = 0.

Hcolor = n—This statement specifies the color used in high resolution graphics, with n = 1 to 7.

Hgr—This statement turns on high resolution graphics, clears the screen, sets Hcolor = 0, and displays page 1.

Hgr2—This statement turns on double high resolution graphics and uses page 2.

Hlin mm,nn at rr—This statement draws a horizontal line on the high resolution graphics screen from column mm to column nn in row rr.

Home—This command clears the screen and homes the cursor; in graphics mode it homes the text area of the screen only.

Hplot—This statement draws points or lines on the high resolution graphics screen. There are two types of arguments for this statement: Hplot mm,nn draws a point at column mm, row nn. Hplot jj,kk to mm,nn draws a line from point jj,kk to point mm,nn.

Htab nn—The horizontal tab command moves the cursor to column nn on the screen. Note that nn = 1 to 255, and that these column numbers can be displayed as more than one 40 column line: in 40 column mode, the command Htab 50 positions the cursor on line 2 column 10.

If ... Then—These statements form a conditional loop. The If statement is evaluated; the line is executed if the statement is true and ignored if false. For example If A <= 4 Then Goto 100. The Goto statement would be executed as long as variable A was less than 4.

**Input "screen query"; var1, var2$**—The input statement allows the program to interact with the user and obtain immediate input for variables by displaying a question on the screen and accepting keyboard data in particular format in reply. For instance, the statement Input "Enter cost, a comma, and name of item"; cost, name$

assigns the values entered by the user to the variables "cost" and the string variable "name$".

Int (nnn.nnn)—The integer function returns the largest integer less than or equal to the number entered. Int (6.24) is 6.

Inverse—The inverse statement reverses the video display of the screen so characters are shown as dark on a light background.

**Left$ ("var$", nn)**—This statement returns the specified number of characters from the left side of the string variable specified. Left$ ("Mississippi",4) returns Miss.

Len ("var$")—This statement returns the length (character count) of the string variable. Len ("BASIC Program") is 13.

List—This function shows the program lines contained in memory on the screen. Specific lines can be listed with line numbers as arguments. List shows the entire program. List nnn shows only line nnn. List -nnn shows the beginning of the program up to line nnn. List nnn- shows line nnn to the end of the program. List mmm-nnn (or List mmm,nnn) shows lines mmm to nnn.

Load filename,Dn—The Load command reads the specified file from the specified disk drive and loads it into memory.

**Mid$ ("var$",m,n)**—This statement takes a substring from the middle of a string variable starting at character m of the string and going to character n. Mid$ ("My gal Sal",4,8) returns gal S.

New—The New command clears the contents of memory and sets all variables to zero.

Next—The Next command is used in the For...Next loop. Refer to For for an explanation of the function.

Normal—The Normal command resets the Inverse function to normal video display, that is, light characters on a dark background.

Notrace—The Notrace command turns off the Trace function. Refer to Trace for an explanation of the function.

Onerr Goto nnn—This statement is used to cause the program to execute the Goto line nnn statement instead of halting program execution when an error message is encountered.

Peek (-nnnnn)—This command returns the contents of the memory location specified by the decimal number nnnnn.

Plot mm,nn—This command plots a location on the low resolution graphics screen at column mm and row nn.

Poke (mmmmm,nn)—This command stores the value nn in binary form in the decimal address location mmmmm.

Print—This statement displays something on the screen. Different arguments and format modify the display. Print alone produces a carriage return and line feed. Print nnn displays nnn. Print "xxx" displays exactly the contents within the quotation marks. Print var displays the current value of the variable. Print a;b;c displays abc with no spaces between the characters. Print a,b,c displays a b and c with tab spaces between them.

Pr#n—A DOS 2 command that specifies a destination for data or commands, where n is one of the computer's slots, 0 to 7. For instance, the printer is usually slot 1, so data to the printer would be directed by the command Pr#1.

Read—The Read statement in a program identifies the elements or variables as they appeared in a previous Data statement. For instance Read var1, var2$, var3$ interprets the elements it finds in the Data statement as the specified variables, with the last two being string variables.

Rem—The Remark statement is not processed by the program so it can be used for clarifying explanations in the body of the program.

Return—The Return statement is used at the end of subroutines as a signal for program execution to return to the statement immediately following the Gosub just executed.

**Right$ ("var$", nn)**—The Right$ string function returns the number of characters specified by nn starting from the right end of the string. Right$ ("Washington, DC",2) returns DC.

Rnd (nn)—This statement generates a random real number between zero and one for any nn > 0. For nn = 0 it returns the most recently generated random number.

Run—This command executes the program in memory. Run nnn begins execution of the program at line number nnn.

Run filename,Dn— A DOS 2 command to execute a program. If the program is stored on the diskette, it is first loaded into memory. Franklin DOS 2 allows Run to be abbreviated as — .

Save filename,Dn—A command that stores the program currently in memory on the diskette under the filename specified.

Str$ (aaaa)—The string function returns the string that represents the contents of the argument given. For instance, Str$ (62.5) returns 62.5.

Tab—This function must be used in a Print statement. The Tab function moves the cursor on the screen to the specified Tab fields.

Text—This statement resets the display screen to text mode from graphics mode.

Trace—The Trace function displays the line number of each statement to be displayed as it is executed as a device to aid program debugging. Refer also to the Notrace function for the resetting command.

Val (aaa)—This function interprets the string in the argument as a numeric value. If no number occurs, or a letter occurs first, the function returns zero.

Vlin jj,kk at mm—This command draws a vertical line on the low resolution graphics screen from row jj to row kk in column mm.

Vtab (nn)—This command moves the cursor to the specified vertical row on the display screen.

Reserved Words In BASIC The words used for BASIC programming statements and commands cannot be used for other things, such as filenames or variables, because BASIC tries to interpret them as instructions.

The following list contains words and symbols reserved for the use of BASIC. These

words, or words that use the first two or three identical letters, must not be used except in BASIC program statements for BASIC commands.

&	Hgr2	Pop	Val
	Himem:	Pos	Vlin
Abs	Hlin	Print	Vtab
And	Home	Pr#	
Asc	Hplot		Wait
At	Htab	Read	
Atn		Recall	Xplot
	If	Rem	Xdraw
Call	In#	Restore	
Chr$	Input	Resume	
Clear	Int	Return	
Color=	Inverse	Right$	
Cont		Rnd	
Cos	Left$	Rot=	
	Len	Run	
Data	Let		
Def	List	Save	
Del	Load	Scale=	
Dim	Log	Scrn(	
Draw	Lomem:	Sgn	
		Sin	
End	Mid$	Spc(	
Exp		Speed=	
	New	Sqr	
Fkey	Next	Step	
Flash	Normal	Stop	
Fn	Not	Store	
For	Notrace	Str$	
Fre			
	On	Tab(	
Get	Onerr	Tan	
Gosub	Or	Text	
Goto		Then	
Gr	Pdl	To	
	Peek	Trace	
Hcolor=	Plot		
Hgr	Poke	Usr	

**Franklin BASIC Error
Messages**

The following list shows the error messages which BASIC uses in response to prob-
lems encountered while interpreting BASIC statements. The error messages are brief
and sometimes a little too vague to direct you to the precise fault, but the descriptions
that accompany them should help.

?Bad Subscript Error
This results from an array element incorrectly entered or an array incorrectly
dimensioned.

?Can't Continue Error
This informs you that the program cannot continue for some reason, perhaps because
a line has been deleted or added.

?Division By Zero Error
Mathematical statement contains an illegal division by zero.

?Formula Too Complex Error
This generally results from interlocking two If...Then statements.

?Illegal Direct Error
This error arises when a statement such as Input, Def FN, Get, or Data is used for
immediate execution.

?Illegal Quantity Error
This error results from arithmetic statements which cannot be evaluated. These
include string, Poke/Peek, or graphics commands with arguments outside the valid
range; square roots or logs of negative numbers; or negative array subscripts.

?Next Without For Error
This error results from a mismatch in the For...Next statement, either from a missing
element or variable.

?Out Of Data Error
This error results from a mismatch in Read and Data statements, either from insuffi-
cient Data elements or too few data elements.

?Out Of Memory Error
This error indicates that the program is too large for the available memory. It may
also result from too many levels of loops or expressions. This error is also used to
indicate whether Lomem or Himem statements are invalid.

?Overflow Error

This error indicates that result of the numerical calculation performed is larger than can be expressed by the computer.

?Redim'd Array Error

This error occurs when a second dimension statement for a single array is given. This usually means that the computer assigned the default array dimension before the Dim statement was entered. This can be corrected by moving the Dim statement earlier in the program.

?Return Without Gosub Error

This error is given when a Return statement occurs without a corresponding Gosub statement.

?String Too Long Error

This error indicates that a string variable has exceeded 255 characters, either from adding two strings or not clearing a string variable before it is used again.

?Syntax Error

This most common error indicates that there is something incorrect about the statement or parameter entry so it cannot be interpreted by BASIC. Check punctuation, parentheses, spelling errors, etc.

?Type Mismatch Error

This error usually occurs when a numeric variable is matched to a string variable.

?Undef'd Statement Error

This error indicates that the line number in a Goto, Then, or Gosub statement doesn't exist.

?Undef'd Function Error

This error indicates that a function called in the program has never been defined in the program.

Chapter 8

Function Keys

This chapter is for all users. It describes the operation of the twelve function keys on the keyboard. The current definitions of the keys are given and there are instructions for changing the definitions to new ones.

The latter part of the chapter gives instructions for programming the function keys by means of a BASIC start up file along with some general comments on programming the keys.

ACE 500 Computer Function Keys

ACE 500 Computers have twelve special keys that are sometimes called Function Keys, Programmable Keys, or User Defined Keys. They are labelled F1 to F12 and are located at the top of the keyboard.
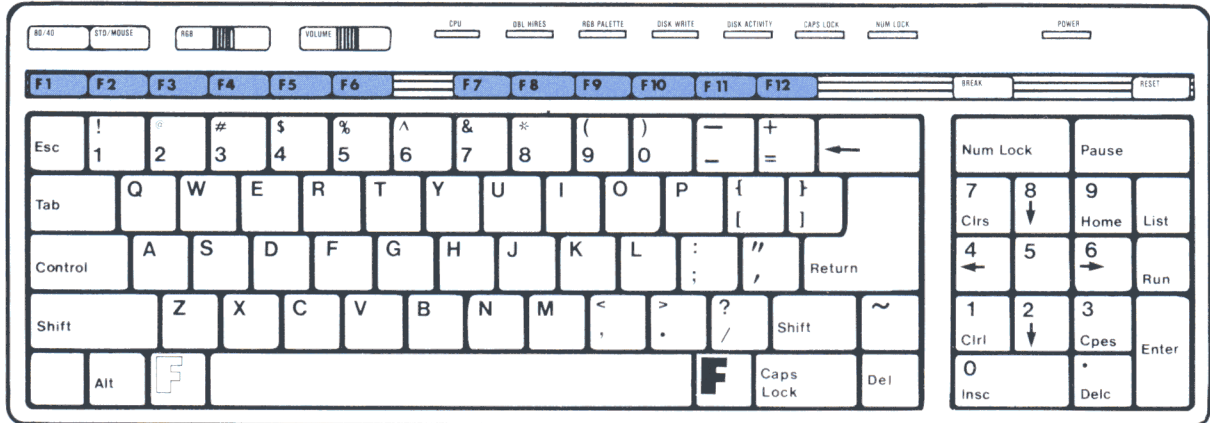
These keys are called programmable because you can give them any definition you want, though the word "programmable" implies a little more effort than is required to enter your definitions. They are called function keys because they can perform a "function", such as a command, or enter several characters at once instead of a single character or control code like the other keys on the keyboard.

Using The Function Keys

Using a function key is like using any other key except you produce the key definition instead of the key character. The function keys have definitions assigned to them that are entered each time you boot the Franklin DOS 2 diskette. To see how the function keys work, boot the Franklin DOS 2 diskette if you haven't done so already, and press F1. You'll see

<p align="center">CATALOG, D1</p>

appear on the cursor line, and then you'll see the catalog of the diskette displayed. You may have guessed that the function key definition included <RETURN> since the CATALOG command was executed automatically, even though <RETURN> doesn't appear on the screen.

That's all there is to using a function key. Just press the key and it works as if you had typed in the characters that make up its definition. The list of the assigned definitions follows and instructions for entering your own definitions are given later in this chapter.

There is a caution about using the function keys that you should know about. The function key definitions are stored on the Franklin DOS 2 diskette and read into memory each time you boot the diskette. They will always work correctly when Franklin DOS 2, Franklin BASIC, or the MONITOR program are running. However, if you run another program, perhaps an application program which uses its own operating system, the function key definitions may be destroyed or unavailable because of that program's use of memory. In these cases the function keys generate strange (or "garbage") characters when you press them. The function keys also generate strange characters when you use them together with the CTRL and ALT keys.

If you start the computer without booting the Franklin DOS 2 diskette, or don't have the Function Key definitions loaded for some reason, pressing the function keys simply produces a space, just as if you had pressed the space bar of the keyboard.

Default Function Key Definitions

The Franklin DOS 2 diskette contains a set of definitions for the function keys in the start up file that is loaded each time you boot the diskette. The definitions are:

Key	Definition	Description
F1	CATALOG,D1<RETURN>	Catalog of drive 1
F2	CATALOG,D2<RETURN>	Catalog of drive 2
F3	PR#3 <RETURN>	Turn on 80 columns

F4	PR#1 <RETURN>	Turn on printer output
F5	<ESC> 4	Switch from 80 to 40 columns
F6	<ESC> 8	Switch from 40 to 80 columns
F7	COPY	Copy a diskette file
F8	TYPE	List a diskette file
F9	LOCK	Lock a diskette file
F10	UNLOCK	Unlock a diskette file
F11	DELETE	Delete a diskette file
F12	RENAME	Rename a diskette file

Note that <RETURN> and <ESC> in the definitions are single keys, not the spelled out versions of the commands. The definitions are shown here in uppercase letters, but Franklin DOS 2 could process them just as well as lowercase letters. Note also that the commands that require <RETURN> for execution have it included in their definition. For other commands that require additional input, like COPY, you have to supply the additional information and <RETURN>. For instance, the command line

COPY filename, D2 <RETURN>

would be typed:

F7 filename, D2 <RETURN>

The operation of the Franklin DOS 2 commands themselves are given in the Franklin DOS 2 chapter and the video display chapter. As far as the function keys are concerned, all you have to remember is that the key produces the sequence of characters shown in the definitions whenever it is pressed.

Programming The Function Keys

It's easy to put your own definitions on the function keys. There is a special Franklin BASIC command, FKEY, to do it for you. You can use this command any time you see the] prompt. The command has the function key number and definition in this form:

FKEY number, "definition"

For instance, to program key F7 to enter a command to clear programs from memory, you would type:

```
FKEY 7, "NEW" CHR$(13) <RETURN>
```

In this definition NEW is the BASIC command and CHR$(13) is the BASIC representation for CTRL M, which is RETURN. The second <RETURN> is for the execution of the FKEY command.

You might want to use one of the function keys for a name or number you used frequently, such as:

```
FKEY 8,
"Donaudampffahrtschiffsleutenantsassistenten"
```

which would save you a lot of typing.

A new definition for a key automatically replaces the previous definition. The total number of characters for all function key definitions is 233, but be aware that Franklin BASIC doesn't check this number for you. If you exceed 233 characters in your definitions you'll destroy the contents and/or the numbering of all the definitions.

There are a few more remarks needed for users who know a little about programming. The FKEY command is unique to Franklin BASIC and is used specifically for the Franklin ACE 500 computers. The arguments for the FKEY command are any valid BASIC numeric expression in the range 1 to 12, and any valid BASIC string expression for the definition. For instance,

```
FKEY 6, "Make my day!"
```

is equivalent to:

```
A$="Make ":B$="my ":C$="day!"
X = 3*2
FKEY X, A$B$C$
```

Using A Start Up File

When you initialize a new diskette the INIT command asks you to name a start up file that will run automatically each time that diskette is booted. One of the things most frequently put into a start up file is definitions of the function keys.

Suppose you were creating boot diskettes for several different application programs, one for your word processor, one for your children's math tutor, and one for your kitchen recipe planner. Each program has commands you use frequently and it

would be very convenient to have those commands readily available on the function keys. By making a separate start up file for each boot diskette with individual function key definitions for each application program, you personalize your computer operations and make them more efficient and convenient.

Here are a few suggestions before you get started:

- Use the application program for a while before you choose the commands to put on the keys. Then you'll want to think about which commands are most important, which function key locations are most convenient for you, etc. Some of the default definitions on the Franklin DOS 2 diskette may still be useful to you.

- Write out a list of the commands you want and make a careful count of the total number of characters. Make adjustments as needed while your list is still in the planning stage.

- Keep careful notes or labels for your diskettes so your different boot files and definitions are clear. It's always possible to LIST the start up file to find out what's in it and what definitions have been made, but that's not always convenient.

When you're ready to make your start up file, named HELLO in this example, just include an FKEY definition for each function key you want to define. Refer to the Chapter 6, *Introduction To Programming*, for some suggestions. For instance, this procedure will create a sample start up file.

```
NEW <RETURN>
10 FKEY 1, "Definition 1"
20 FKEY 2, "Definition 2"
   .        .         .
   .        .         .
   .        .         .
120 FKEY 12,"Definition 12"
END <RETURN>
SAVE HELLO <RETURN>
```

This program, which could include other things you might want, will define the function keys as shown each time you boot the diskette.

Chapter 9

The Display

This first part of this chapter is for general use in selecting and connecting a video monitor. It contains additional information about connecting RGB monitors and their requirements in interacting with the ACE 500 Computer.

The latter parts of this chapter describe the graphics capability of the computer, including the use of low resolution graphics, high resolution graphics, and double high resolution graphics screens. Listings of special 80 column mode video display software and its use is described for advanced programmers.

Choosing A Video Display Device

Your ACE 500 Computer can be connected to a number of different display devices. The type of display device you choose depends on the use you want from your computer. There are three types of display monitors: monochrome, composite color, and RGB color.

- Monochrome display monitors are fairly inexpensive display devices that are designed to produce very sharp letters and graphics. They are excellent for 80 column display, such as you might use with word processing or spreadsheet programs. On the other hand, they don't display color. Color is often used with games or educational software, so this may be a disadvantage. Monochrome monitors are usually available with either amber or green screens.

- Composite color display monitors are somewhat more expensive. The composite monitors produce color in a simple way and use the same signal from the computer as the monochrome monitor. Their resolution is not very good (about like a television set), but they are less expensive than RGB monitors. They generally use a 40 column display, though newer models can also produce 80 column text.

  The composite color monitor uses the signal (a standard NTSC composite color signal) from the normal video output on the back panel of the ACE 500 Computer.

If you use a composite color monitor with software that utilizes a double high resolution screen, or on with mixed text and graphics, the color will be displayed best with Switch #5 of the OPTION SWITCH group in the Down position.

• RGB (for Red - Green - Blue) monitors control the generation of each color signal independently and produce very sharp color images. They require special signals from the computer and are generally more expensive than the other monitors.

The signal needed for RGB monitors is provided from one of the DB9 connectors on the back panel of the ACE 500 computer.

The type of display you choose should balance factors like the applications you'll use on your computer, your personal preference, your budget, etc. Discuss the matter with your Franklin dealer and look at the different display monitors available.

Connecting The Monochrome Or Composite Color Monitor

Connecting the monochrome or composite color display monitor is very simple. You've probably done it already in the Getting Started chapter, but here's a brief review.

Plug the monitor cable provided into the RCA jack connector marked with the icon for a monitor screen on the back panel of the ACE 500 Computer. This connector provides signals for either monochrome monitors or composite color monitors.

Then plug the other end of the monitor cable into the display monitor. The display monitor has controls for controlling the contrast and brightness of the display. Adjust
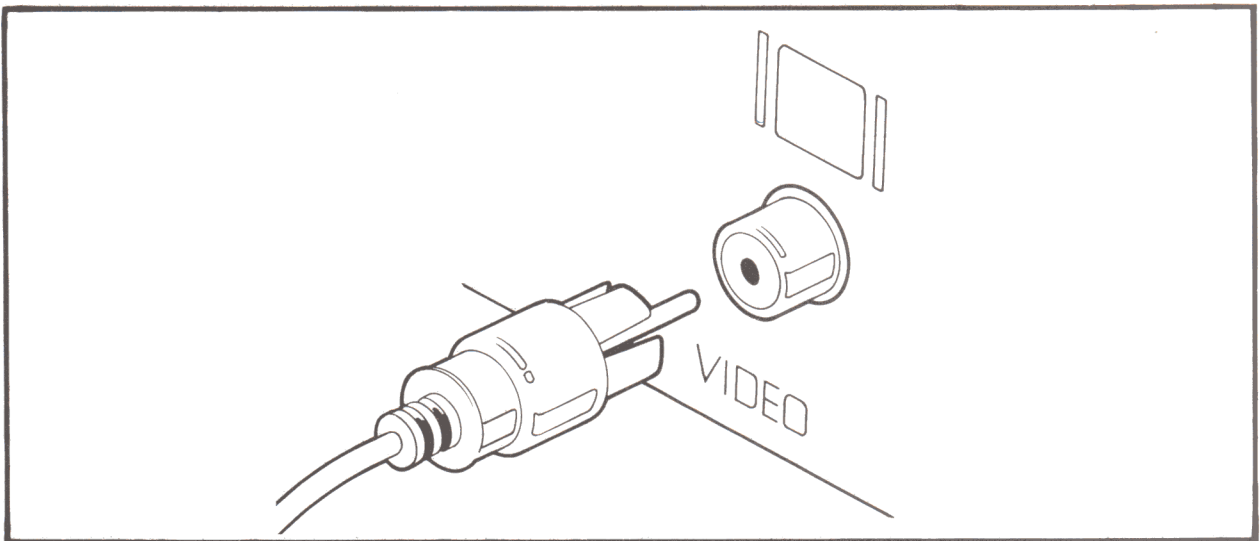
these to your preferences and to the room lighting. If you leave the computer on for long periods of time without using it, you should turn down the brightness of the display. The unchanging display very gradually burns into the phosphor, making a permanent mark. When you're working with the computer, the display keeps changing so burning in an image is not a problem.

Connecting An RGB Monitor

The special signal needed for the RGB monitor is available from the DB9 connector marked with the screen and three dots (for RGB) on the back panel of the computer. Plug in the connecting cable, then set the RGB switches in the OPTION SWITCH group before you turn the RGB monitor on.

Choosing The RGB Monitor Configuration

RGB monitor manufacturers use different conventions for their RGB signal requirements. In general, there are two configurations for monitors, the Apple and the IBM. The main difference between the two types is that the Apple configuration uses a composite signal for the horizontal and vertical synchronization, and the IBM uses separate signals for them. The two types also use a different labelling system (called color palette) for colors. Some RGB monitors have a switch which permits selection of the Apple or IBM modes. If you have a choice, use the Apple mode.

Consult the monitor's manual to see which settings are required for the computer, and whether the Apple or the IBM configuration meets the needs of your RGB monitor. The descriptions below should help you choose the settings you need without too much trouble.

The OPTION SWITCH group has a set of 4 switches which select the RGB configuration used by the computer.

SWITCH #2 — This switch governs the polarity of the line that carries both the Composite Synchronization signal for Apple mode monitors and the Horizontal Synchronization signal for IBM mode monitors. The switch position Down is for Negative signals and the Up position is for Positive signals.

Check your RGB monitor manual to see which polarity you need, either for the composite or the horizontal sync signals for Apple or IBM type monitors, respectively.

SWITCH #3 — This switch governs the polarity of the line that carries the Vertical Sync Signal for IBM mode monitors. The Down position selects the Negative Vertical Sync signal. The Up position selects Positive Vertical Sync. Positive Vertical Sync is generally the more common setting.

SWITCH #4 — Apple/IBM Mode   This switch selects the configuration of the color matrix, or color palette, that the computer uses, either according to the Apple or the IBM formats. The Down position selects the IBM color palette, Up selects the Apple color matrix mode.

If your RGB monitor says "Apple compatible" or "IBM compatible", or has some other indication that it is designed to work with one of those computers, try the configuration suggested for that computer. The palette assignments for colors is somewhat arbitrary and may not matter until you use some software that produces red grass and orange sky. The four-position RGB slide switch on the front panel should give white, green, amber, and blue text in its four settings. If these colors aren't accurate, the wrong color palette has been selected.

SWITCH #5 — Monochrome Double High Resolution   This switch selects a single color (the text color selected with the RGB slide switch on the front panel) on a black background when the computer is used with a double high resolution screen. This is generally only used with specific software applications, such as Computer Assisted Design, where the four bit positions on the screen needed for RGB color are traded for maximum screen resolution.

The Down position of switch #5 selects monochrome display when the computer uses Double High Resolution display mode.

If you're using an RGB monitor, the Down position selects the monochrome selection and the Up position selects RGB. The monochrome screen in double high resolution mode offers increased visible resolution needed for some applications, such as Computer Aided Design. The monochrome selection (the Down position) also offers better readability for double high resolution screens, especially those that use mixed text and graphics. You may need to try this switch in both settings for different software programs to see which screen presentation you prefer.

The slide switch on the front panel above the keyboard has four positions which select the text color used by the RGB monitor when operating in text mode. The colors are white, green, amber, and blue on the black background. Try the switch in different positions to see which you prefer. The factory setting has white text selected.

Consult the manual of your RGB monitor for the kinds of signals required for your monitor, or ask your dealer or the manufacturer for advice. The vertical sync is set incorrectly if the monitor screen rolls vertically, and the horizontal sync is wrong if the screen appears diagonally out of alignment. The appearance of these problems is familiar from adjusting your television set.

Standard Display Features   Your Franklin ACE 500 Computer has a number of standard features that enhance the display capability of the computer. Both 40 and 80 column display widths are standard. The 40 column display can show 40 characters on each line; the 80 column

display shows up to 80. Both display widths use 24 lines on the screen. These two choices are available by using keys F5 and F6. The 80/40 switch above the keyboard is a marker for your display preference for software which uses both types of display. Not all software examines the setting of this switch to determine your preference however.

The 80 character line is very convenient for word processing because it displays the length of the line you type on a typewriter, usually with a little to spare. That means the lines and paragraphs you type on the computer appear on the screen just about the same way they will appear on paper when you print them.

On the other hand, many educational and entertainment programs use a 40 column display. There are some good reasons for this. First, these programs are written so they can be used with the poorer resolution of a television set as the display screen. Second, the attractive graphics and animated cartoons employ a graphics capability of the computer that is directly related to the 40 column display. You'll learn more about that shortly.

Computer programs choose the correct display automatically. You don't have to worry about choosing the 40 or 80 column displays until you want to write your own programs.

Additional Display Features

There are a number of additional features built into the display capability of your ACE 500 Computer. Almost all of them are fairly technical and are generally used only by programmers. You may want to read further to see what possibilities exist with your computer, but you don't have to understand this information to use the computer effectively. The software you use, whether an application program or a video game, controls the display directly; you're just an observer.

If you're interested in programming and creating graphic displays, this information will be of interest to you. You can obtain additional information from the *ACE 500 Technical Reference Manual*.

Character Sets

The monitor screen displays uppercase letters, lowercase letters, numbers, punctuation marks, and a few uncommon symbols. These characters, and 32 more (called control codes) that aren't usually shown on the screen, make up the 128 characters of the ASCII character set. ASCII (pronounced "ASK ee") is the acronym for American Standard Code for Information Interchange. You can learn more about the ASCII character set by referring to the appendix of this manual that lists the ASCII characters, and the *ACE 500 Technical Reference Manual*.

Generating a character with a computer involves an ASCII code. When you press a key on the keyboard you generate a signal called an ASCII code. It is a number between 0 and 127 that is a unique computer signal. Different keys generate different ASCII codes, some generated by the key or some only with the <SHIFT> or <CTRL> keys in addition to the key itself. This signal is then processed by the computer, stored, etc. as instructed. When the computer sends the signal to the display for you to see, it checks the signal in the character set to see what pattern of dots that signal represents. For example, pressing a <SHIFT><A> produces an ASCII code 065. This 065 is moved through the computer, perhaps stored on the diskette, and at some point it is sent to the screen. The computer looks in the character set and says "ASCII 065 means display this dot pattern, uppercase A."

This concept is important because the ACE 500 Computer has a second character set available to you. This alternate character set has graphics characters, such as the icons used with a mouse and mouse-controlled software. Refer to the *ACE 500 Technical Reference Manual* for details on this other character set. This manual simply explains its presence in case you run across it accidentally. When you use this other character set, the ASCII codes generated by the keyboard are used to produce the dot patterns of the alternate graphics (Mouse) character set.

The Mouse Character Set

To use the Mouse character set, you have to set the front panel switch to the Mouse set, and the software you're using must have controls to turn it on. It is not sufficient
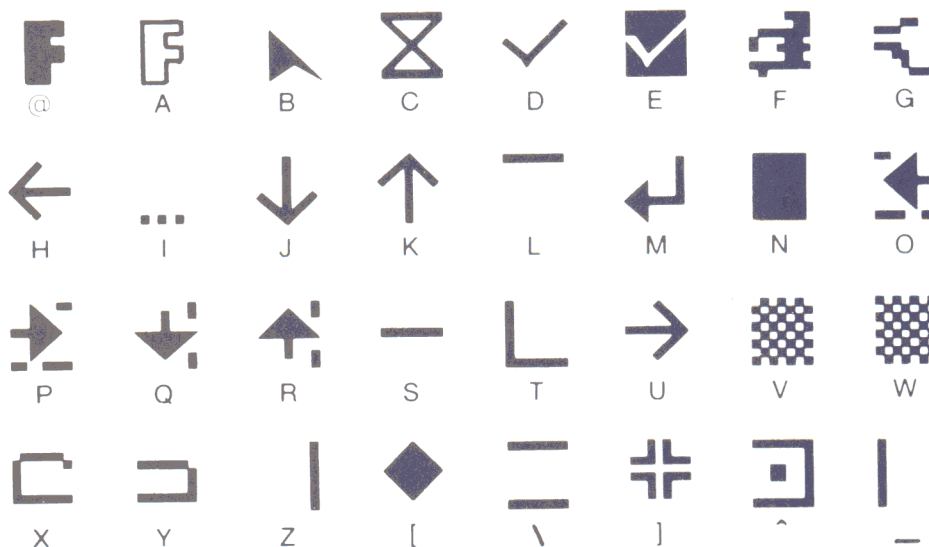
to press the switch labelled Mouse/Std unless the software also calls for it. The switch should be On for Mouse programs and most newer software programs. If the program calls for a specific character set and the switch is in the wrong setting, you'll see unusual characters or displays, which should be cleared by changing the position of the switch.

The following graphics characters are generated in the Mouse character set corresponding to the ASCII code generated by the character shown at the lower left corner of each diagram.

The software using the alternate character set takes care of the access and control of the characters in the Mouse character set.

Additional Display Techniques

Any character or graphic can be displayed in inverse video so it stands out as a dark character on a light background. Graphics characters and inverse video can be combined to enhance any screen display. In text mode the ACE 500 Computer has the capability to display regular characters, inverse video characters, or flashing characters. The commands for these display features are available in the BASIC programming language. INVERSE, FLASH, and NORMAL are display commands described in Chapter 7, and in the *ACE 500 Technical Reference Manual*.

Generally Franklin DOS 2 and Franklin DOS 2 programs use 40 column mode for video display. The 80 column mode has other capabilities, and there are other graphic display modes, such as lores, hires, and double hires display modes that are discussed briefly below.

Mixed Text Displays

The ACE 500 Computer has two display modes that mix text and graphics display modes. These mixed modes are available in lores, hires, and double hires graphics display modes and display four lines of text at the bottom of a graphics screen.

Low Resolution Graphics

Low resolution graphics (usually called lo-res, or lores) uses a screen composed of a grid of blocks 40 wide and 24 high. The grid has no spacing between blocks, so contiguous blocks merge to form a single large block.

The blocks can use any of 16 different colors. The lores colors and the numbers to select them are:

0 black	4 dark green	8 brown	12 green
1 magenta	5 gray1	9 orange	13 yellow
2 dark blue	6 medium blue	10 gray2	14 aqua
3 purple	7 light blue	11 pink	15 white

High Resolution Graphics

High resolution graphics (usually called hi-res, or hires) uses a screen composed of a grid of points 280 wide by 192 high. Contiguous dots of the same color merge to form a larger image.

These points can use any of 8 selectable colors. The hires colors and the numbers to select them are:

0 black1	2 violet	4 black2	6 blue
1 green	3 white1	5 orange	7 white2

The graphics screen has two pages of memory allocated to it. This means that two graphics images can be stored at the same time. You may have noticed this when you switched from 40 to 80 column text modes and observed that the text for each mode was preserved even while the other mode was operating. That storage demonstrated the two-page storage system used by the display. That same two-page storage system is used for graphics images. One graphics image can be stored in page one and another in page two. Other memory management techniques allow you to use some of the screen for graphics and some for text. Refer to the *ACE 500 Technical Reference Manual* for further details.

Double High Resolution Graphics

A further display mode is available which allows you to display 560 x 192 dots on the screen. This is called "double" high resolution because it effectively doubles the horizontal resolution of the screen by interleaving the memory contents of page one and page two for alternate points on the screen.

Double hires graphics uses a different color generation technique which can produce one color for each four screen dots, so the color resolution is 140 color groups per horizontal line. There are 16 colors available, the same ones as available with lores graphics color display, listed below:

0 black	4 dark green	8 brown	12 green
1 magenta	5 gray1	9 orange	13 yellow
2 dark blue	6 medium blue	10 gray2	14 aqua
3 purple	7 light blue	11 pink	15 white

Using lores, hires, and double hires graphics requires some programming skill and experience. Refer to Chapter 7 on BASIC or the *ACE 500 Technical Reference Manual* for further details about these graphics display techniques and methods for using them.

Some software programs that use the double high resolution screen, particularly those that mix text and graphics, have a more legible appearance on the screen if Switch #5 of the OPTION SWITCH group is in the Up position. This switch selects the monochrome double high resolution display and eliminates the poor readability caused by the adjacent pixels of displayed characters being of different colors. You may want to try this switch in both positions for different pieces of software to determine the best double high resolution screen presentation.

Enhanced Video Software

The Franklin ACE 500 Computer has some additional software built into the video section of its controls. This software is generally associated with the 80 column and keyboard control of the computer, but it is of general use and interest for advanced use of the computer and its capability. This software includes several general enhancements and keyboard control commands as well as some new commands that operate in either 40 or 80 column mode.

The special features of the enhanced video software are available only if you turn them on with the 80 column command, PR #3 <RETURN>, which you can also do by pressing the F5 function key. The screen will clear and characters you type will appear in 80 column mode.

The enhanced video software is always available in 80 column mode. However, it is also available in 40 column mode if you've switched back and forth from 80 to 40 column modes. The best way to tell if the video software is active is to look at the cursor. The enhanced video software always uses a solid, unblinking block cursor in either 40 or 80 column mode. If you see a blinking dot pattern cursor, the enhanced video software is not active and the commands described here aren't available.

Enhanced Video Software Commands

There are three different command groups available in the enhanced video software, control codes, escape sequences, and one key commands. All these commands can be given from the keyboard but only the control codes can be given from a program, and then only if the enhanced video software has been activated. A program can execute these control codes by means of the appropriate PRINT statements after a PR#3, or equivalent command, has been given.

Control Code Commands
Control codes are executed by holding down the CTRL key while you press the other key. The Control code commands are given, with a brief explanation, below. If an invalid command is given, it is ignored.

CTRL G   Bell
This command sounds the computer's bell.

CTRL H   Cursor left
This command moves the cursor one position to the left. If the cursor is at the first column of a line, it will wrap around to the end of the previous line. If the cursor is in the first column of the first line, the command is ignored.

CTRL J   Cursor down
This command moves the cursor down one line on the screen in the same column. If the cursor is on the last line of the screen, the whole screen will scroll up one line, with the top line of the screen disappearing off the top and a new bottom line appearing.

CTRL K   Clear to end of screen
This command clears the screen starting at the current cursor position and continuing to the end of the screen. The cursor remains in the same position.

CTRL L   Clear entire screen and home cursor
This command clears the entire screen and positions the cursor in the first column of the first line.

CTRL M   Carriage return
This command is the same as a carriage return. This is equivalent to pressing <RETURN>.

CTRL N   Reverse video off
This command turns off reverse video display mode.

CTRL O   Reverse video on
This command turns on reverse video display mode, which means that characters are displayed as dark characters on a light background.

CTRL Q   Switch to 40 columns, keep enhanced video software
This command switches the video to 40 column mode. This command does not deactivate the enhanced video software. If you had data on the screen in the previous 80 column mode operation, the data would be displayed but truncated to the first 40 columns.

CTRL R   Switch to 80 columns, keep enhanced video software
This command switches the video to 80 column mode. This command doesn't deactivate the enhanced video software. If you had data on the screen from the previous 40 column mode, it will appear in the first 40 columns of the 80 column display. Columns 41 through 80 will be blank.

CTRL U(kybd)   Cursor right (from keyboard only)
This command moves the cursor one position to the right when it is executed from the keyboard.

CTRL U(prgm)   Switch to 40 columns, exit enhanced video software
This command switches the video display to 40 column mode and deactivates the enhanced video software when the command is executed from a program.

CTRL V   Scroll down
This command scrolls the screen down one line. The new top line will be cleared and the bottom line disappears.

CTRL W   Scroll up
This command scrolls the screen up one line. The new bottom line will be cleared and the top line disappears.

CTRL X   Disable Mouse Text
This command disables the display of the mouse text characters and uses the normal ASCII character set.

CTRL Y   Home cursor
This command moves the cursor to the home position, line one, column one.

CTRL Z   Clear line
This command clears the entire line marked by the cursor. The cursor position is unchanged.

CTRL   Cursor right
This command moves the cursor one position to the right. If the cursor is in the last column of the line, it is moved to the first column of the next line. If the cursor is in the last column of the last line, the screen scrolls up one line and the cursor moves to the first column of the new last line.

CTRL [Enable Mouse Text
This command enables the mouse text character set.

CTRL]   Clear to end of line
This command clears the cursor line starting from the cursor column and continuing to the end of the line. Data in columns in front of the cursor are not affected.

CTRL   Position cursor to X,Y

This command positions the cursor to any location on the screen. The command interprets the next two characters typed (or PRINTed from a program) after this command as the column number and the line number of the new cursor location. The numbers are determined by subtracting 32 from the ASCII value of the characters. Remember that lines and columns start their numbering at 0, not at 1. Therefore the command CTRL !! (the ! is ASCII 33) positions the cursor in the second column of the second line.

Escape Sequence Commands

Escape sequences are special commands. You execute an escape sequence by pressing the ESC key, releasing it, then pressing the next key in the sequence. In this way, ESC is not used like a SHIFT or CTRL key because it is not held down while the second key is pressed. An escape sequence is also different in that it is not executed except when entered from the keyboard.

Escape sequences are used frequently by the enhanced video software for multiple commands in "escape mode". Many control code cursor movement commands have limited utility because they result in one movement for each command and make it difficult to move long distances. Escape mode commands allow simpler commands that don't have this restriction. Escape mode is marked by a special cursor, a solid block with a plus sign in it. Whenever you see the escape mode cursor it means that the enhanced video software is expecting the second character of an escape sequence.

The escape sequences used by the enhanced video software are described below. If an invalid character or sequence is entered, it is ignored. Those sequences which use escape mode are marked.

ESC @   Clear screen and home cursor

This sequence clears the entire screen and moves the cursor to the first column of the first line.

ESC A   Cursor right

This sequence moves the cursor one position to the right. If the cursor is in the last column of the line, it moves to the first column of the next line. If it is in the last column of the last line, the screen scrolls up one line and the cursor moves to the first column of the new bottom line.

ESC B   Cursor left

This sequence moves the cursor one position to the left. If the cursor is in the first column of a line, it moves to the last column of the preceding line. If the cursor is in the first column of the first line on the screen, the command is ignored.

ESC C   Cursor down

This sequence moves the cursor down one line staying in the same column. If the cursor is in the last line, this command causes the screen to scroll up one line, with a cleared bottom line appearing in the last position and the top line disappearing from the top position.

ESC D   Cursor up

This sequence moves the cursor up one line staying in the same column. If the cursor is in the top line of the screen, the command is ignored.

ESC E   Clear to end of line

This sequence clears the line, starting from the cursor position and continuing to the end of the line. Data on the line in front of the cursor is unaffected.

ESC F   Clear to end of screen

This sequence clears the screen, starting at the current cursor position and continuing to the end of the screen. The cursor position is unaffected.

ESC I   Cursor up, stay in escape mode

This sequence moves the cursor up one position, like the ESC D command, except this command stays in Escape mode.

ESC J   Cursor left, stay in escape mode

This sequence moves the cursor left one position, like the ESC B command, except this command stays in Escape mode.

ESC K   Cursor right, stay in escape mode

This sequence moves the cursor right one position, like the ESC A command, except this command stays in Escape mode.

ESC M   Cursor down, stay in escape mode

This sequence moves the cursor down one position, like the ESC C command, except this command stays in Escape mode.

ESC 4   Switch to 40 columns, keep enhanced video software

This sequence switches the video display to 40 column mode. The enhanced video software remains active in 40 column operation. If there was 80 columns of data previously displayed, the first 40 columns would be displayed and the last 40 columns would be truncated.

ESC 8   Switch to 80 columns, keep enhanced video software

This sequence switches the video display to 80 column mode. The enhanced video software remains active. If the previous display mode was 40 columns, the data displayed will appear in the first 40 columns of the new 80 column display, with columns 41 to 80 cleared.

ESC CURSOR RIGHT	**Cursor right, stay in Escape mode**
ESC CURSOR LEFT	**Cursor left, stay in Escape mode**
ESC CURSOR UP	**Cursor up, stay in Escape mode**
ESC CURSOR DOWN	**Cursor down, stay in Escape mode**

The cursor arrow keys in the numeric keypad are active in escape mode. When the escape mode cursor (solid block with the plus sign) is present, you can move the cursor by using the arrow keys on the cursor keypad.

ESC CTRL Q   Switch to 40 columns, exit enhanced video software

This sequence switches the video display to 40 column mode and deactivates the enhanced video software.

One Key BASIC Editing Commands

These BASIC editing commands are included with this group because they are part of the enhanced video software. They were described earlier in Chapter 7, *Franklin BASIC*. They are available when the enhanced video software is activated. They also require that the NUM LOCK key be off. These commands are marked on keytops in the cursor keypad where they are convenient for editing. In a way, these keys are like special function keys in that they have special definitions but they are different because they are functional only when the enhanced video software is active. Related keys, RUN, LIST, and ENTER, are adjacent and are also convenient for editing, but their functions are not part of the enhanced video software.

CLRS — Clear The Screen

This command clears the entire screen and moves the cursor to the home position, column one of line one.

HOME — Home The Cursor

This command moves the cursor to the "home" position, which is the first column of the first line.

CLRL — Clear The Line

This command clears the entire line designated by the cursor position. The cursor position itself is unaffected.

INSC — Insert A Character

This command inserts a space character at the current cursor position and pushes other characters on the line one position to the right to make room for it. Only characters to the right of the cursor column are affected. If there is a character in the last column of the line, that character is lost.

DELC — Delete A Character

This command deletes the character at the current cursor position and moves all the columns on the right side one space from the right to close up the line. A space character appears in the last column of the line.

CPES — Copy Entire Screen Line

This command is useful in editing BASIC programs and has a very specialized operation. This command sends all the characters on the cursor line to the current program for processing or execution. This is equivalent to moving the cursor to the end of the line and then hitting <RETURN>. The CPES command works regardless of the cursor position on the line.

CPES is designed for editing BASIC statements. Suppose you had typed a line of a program that had a mistake in it. First, you would LIST the line, then use the escape mode cursor movement keys to locate the section of the line that needed correction, then make the changes with the INSC and DELC keys. When the changes are complete, just press CPES. The corrected line is sent to BASIC with a <RETURN> without any additional cursor movement required.

If you use the CPES command on a full line (all 40 or all 80 characters are used), you may notice that the last character on the line is cleared to a space character after CPES executes. Don't worry, this is only the display screen's processing of the command. The correct and complete line was sent to the running program.

Enhanced Video Software Command Summary

Control Codes

CTRL G	Sound bell
CTRL H	Cursor left
CTRL J	Cursor down

CTRL K	Clear to end of screen
CTRL L	Clear the screen
CTRL M	Carriage return
CTRL N	Turn off reverse video
CTRL O	Turn on reverse video
CTRL Q	Switch to 40 column mode
CTRL R	Switch to 80 column mode
CTRL U(kybd)	Cursor right
CTRL U(prgm)	Switch to 40 column mode and deactivate enhanced video software
CTRL V	Scroll down
CTRL W	Scroll up
CTRL X	Disable mouse text
CTRL Y	Home the cursor
CTRL Z	Clear the line
CTRL	Cursor right
CTRL [	Enable mouse text
CTRL]	Clear to end of line
CTRL	Position the cursor

Escape Sequences

ESC @	Clear the screen
ESC A	Cursor right
ESC B	Cursor left
ESC C	Cursor down
ESC D	Cursor up
ESC E	Clear to end of line
ESC F	Clear to end of screen
ESC I	Cursor up (stay in Escape mode)
ESC J	Cursor left (stay in Escape mode)
ESC K	Cursor right (stay in Escape mode)
ESC M	Cursor down (stay in Escape mode)
ESC 4	Switch to 40 column mode
ESC 8	Switch to 80 column mode
ESC CURSOR RIGHT	Cursor right (stay in Escape mode)
ESC CURSOR LEFT	Cursor left (stay in Escape mode)
ESC CURSOR UP	Cursor up (stay in Escape mode)
ESC CURSOR DOWN	Cursor down (stay in Escape mode)

ESC CTRL Q   Switch to 40 column mode and deactivate enhanced
                              video software.

One Key BASIC Editing Commands

CLRS — Clear The Screen
HOME — Home The Cursor
CLRL — Clear The Line
INSC — Insert A Character
DELC — Delete A Character
CPES — Copy Entire Screen Line

Chapter 10

Input and Output

This chapter contains information of interest to all users to help set up printers and modems. Much of the information presented here is general because specific material depends on the particular printer and modem and is contained in the manuals of those devices.

The latter parts of this chapter contain more complicated commands for controlling the graphics printing capability of the computer under program control.

Input and output refer to communication between the computer and a peripheral device. The term "peripheral" describes any device outside the computer itself which sends or receives information to or from the computer. The peripheral device and the computer are usually connected by cables of different kinds. A disk drive is a common computer peripheral. In fact, a connection can be made between two computers, so that each becomes a peripheral of the other. More commonly, peripheral devices mean printers and modems. Printers give you permanent and readable output. Modems convert computer signals into audible tones for transmission over telephone lines.

Parallel And Serial Communication

Each character (letter, number, etc.) handled by a computer is made up of eight bits of data. Communication, or the input and output of data, involves moving those eight bits of data for each character from one device to another along the cable that connects them.

There are two techniques for sending information along a cable: parallel and serial. In parallel transmission, the eight bits of data are sent simultaneously on eight parallel wires. This is fast and efficient, but it works well only over short distances (say 10 to 20 feet). Serial transmission sends the eight bits of data per character one after the other along a single line. This is somewhat slower (though still very fast), but much more dependable over long distances.

Most simple printers use a parallel connection because it is easy to connect and its communication procedures are well established. In the office or home environment, it is the preferred method of connecting a peripheral device, such as a printer, to a computer. More complicated printers, longer distances to the printer, or problems with radio frequency interference sometimes require a serial interface connection.

Your Franklin ACE 500 Computer has both a parallel interface and a serial interface built in. The parallel interface is usually connected to the printer and the serial interface is usually connected to a modem, if one is used.

Making The Parallel Connection

Connecting the computer to the printer is a simple task. Plug the printer cable into the outlet on the rear panel of the computer. The cable is a standard parallel cable connecting a DB-25 connector to a Centronics 36 pin connector on the printer.
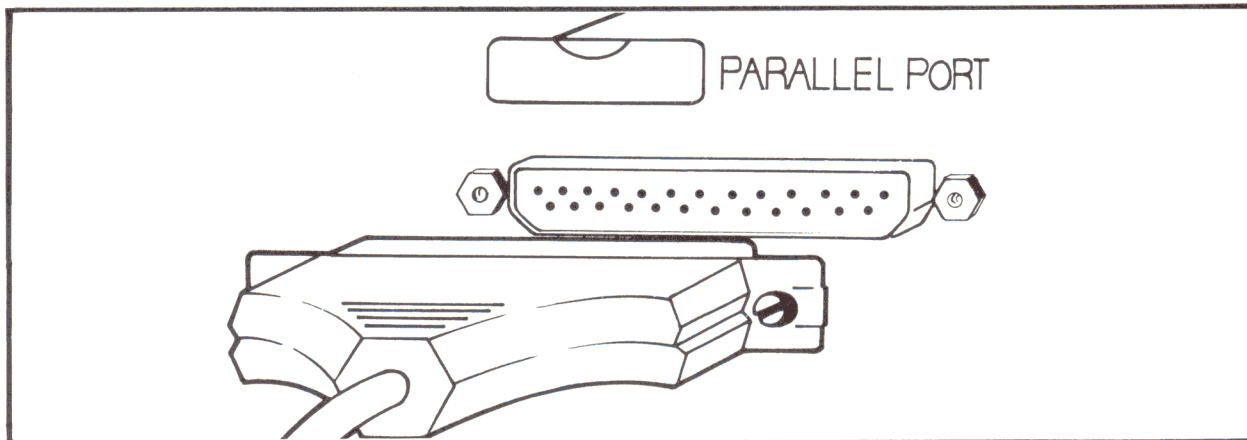
Different printers have different features and options. Your ACE 500 computer has been designed to utilize the features of four popular printers, the Epson printer, the Okidata printer, the C. Itoh printer, and the Star Micronics Gemini printer. Many printers by other manufacturers are compatible with these printers. The computer has built-in software commands and controls to make using these printers easy for you.

Now consult the printer manual and make the connections needed there. The cable should plug into a connector on the back of the printer. You then have to plug in the printer's power cord, insert the paper, check the ribbon, etc.

Now try a little test to see if the printer's connection with the computer is working properly. Turn on the printer and boot the ACE 500 Computer with the Franklin DOS 2 diskette. Exit the HELLO program, and when you see the Franklin DOS 2 prompt], type PR#1 <RETURN>. (You could also press F4 which has this command as its definition.) This turns on the computer's connection to the device in slot #1, which is the printer. Your printer should make some sort of response to this connection, like resetting the print head or adjusting the paper feed. Then type a sentence

on the computer and press <RETURN>. The printer should print the sentence you typed.

If the printer responded as indicated, you're all set. If it didn't, check all your connections. Consult your printer manual for suggestions. If it still doesn't work, check with your dealer.

There is one additional setting you need to make before you use your printer. OPTION SWITCH #6 on the back panel controls whether the computer sends a line feed command to the printer at the end of each line or whether the printer generates the line feed command itself. The OPTION SWITCH #6 setting is Up for sending an automatic line feed, and Down for no line feed command. Consult the printer manual to see which setting your printer requires.
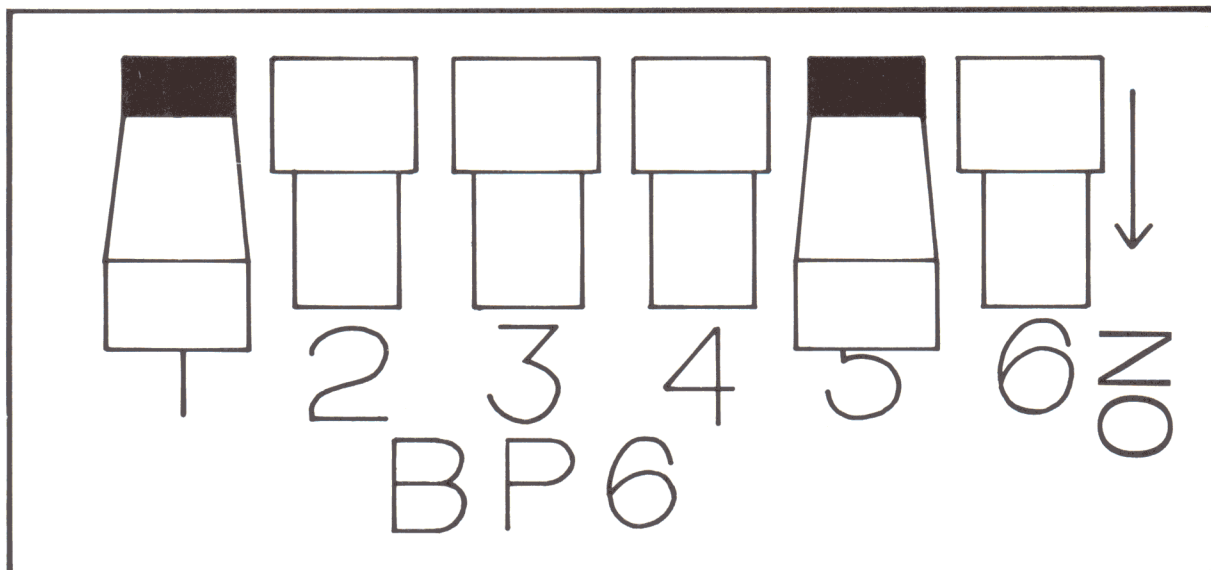
If you can't locate the information, you can try a simple experiment when you've got your printer running to see whether the printer needs the line feed command or whether it generates its own line feed commands. Set the OPTION SWITCH #6 to the Up position so the computer sends a line feed at the end of each line. Then print out a few lines of text. If the text appears on the printer double-spaced, you can conclude that the printer sent line feed commands in addition to those of the computer and you should turn off the line feed switch, which is OPTION SWITCH #6 Down.

Making The Serial Connection

Conventions for using a serial interface are not well standardized. Signals to control the transfer of data are handled in different ways by different devices and manufacturers. The serial connector is a DIN 5-pin connector. You can use any serial device

SERIAL PORT
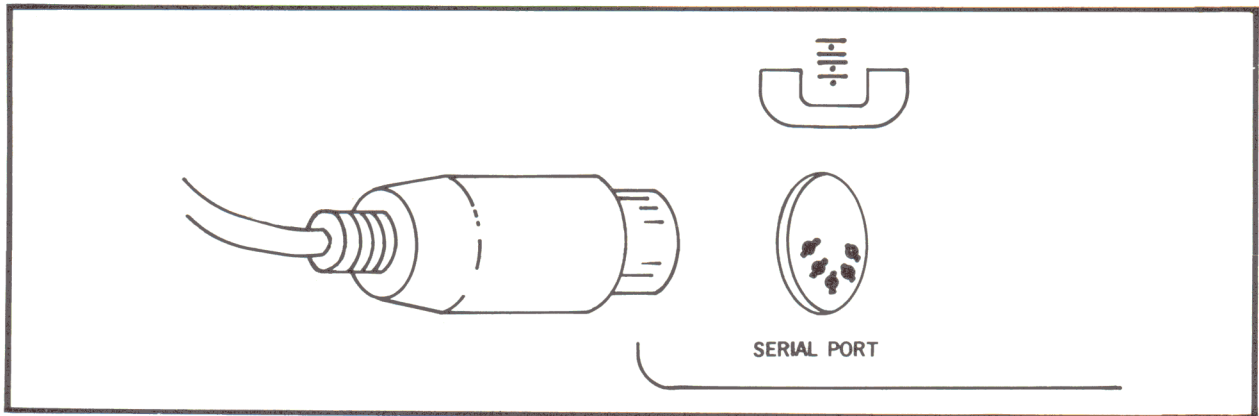
with that interface, but be sure and check that the connection is Franklin ACE 500 or Apple IIc compatible. You can make the connection by plugging it in and making the serial device ready for operation. Make sure the signals on each line in the cable are the same for each device. The pin assignments for the connectors are given in the Appendix of this manual.

The physical connection is only part of the story of the serial interface. The two devices, the computer and the serial device, need to agree on the form of the communication between them, often called the "handshaking" or protocol. There are a number of settings that need to be made for the serial interface protocol. In most cases the actual setting is less important than the fact that the computer and the serial device use the same settings, so consult the manual of the serial device to see what is required.

The Franklin ACE 500 computer has serial interface hardware and firmware built in to accomodate a large variety of settings and communications arrangements. Your first objective is to get the interface working correctly, then you can use some of the more advanced features.

Configuring The Serial Interface

The serial interface connection involves four settings for the protocol: baud rate (the speed of transmission), parity (a verification procedure), character length (the number

of transmitted bits used for actual data), and stop bits (the number of bits used to mark space at the end of the data). Check the manual of the other device to see what is needed for these settings, then use the following procedure to set those values for the computer.

The settings you make will remain until the power is turned Off or the computer is Reset. You can boot the computer or use a number of different programs without changing these settings. You should note however that some application programs with their own operating systems use a Reset command as part of their boot process. These settings could also be made part of a special startup file on a boot disk for your serial port software.

The serial interface configuration is set with these settings each time you turn the computer On:

Baud rate	300 baud
Character length	8 bits
Stop bits	1
Parity	None

If any of those settings are correct for your protocol, you won't have to use the commands to change them.

The first step in setting the protocol is to activate the serial interface. You do this like you did for the parallel interface, by typing PR#2 <RETURN>. This directs all of the computer's output to the device connected to slot #2, which is the serial interface. From this point on all the commands you type on the keyboard will go to the serial device. When you're finished, type PR#0 <RETURN> (slot #0 is unused) to return control to the computer, or PR#6 <RETURN> (slot #6 is the disk drive), to boot the diskette currently in the disk drive. You could also use the PRINT statement in BASIC programs to activate the serial device with PR#2 <RETURN>.

When the serial device is active, use the following commands to set the communications protocol. These commands will be discussed in more detail later, but for now, use them to set up the protocol.

Baud Rate
The baud rate is the speed of communication, given in bits per second. The computer can use a number of rates, so check the manual of the serial device for the rate needed.

<CTRL A> nn B   Set Baud Rate (nn = 1 to 15)

This command sets the baud rate of the serial interface. Enter nn as the number of the baud rate selected.

nn	Baud Rate
1	50 baud
2	75 baud
3	110 baud
4	135 baud
5	150 baud
6	300 baud
7	600 baud
8	1200 baud
9	1800 baud
10	2400 baud
11	3600 baud
12	4800 baud
13	7200 baud
14	9600 baud
15	19200 baud

Select the baud rate you want, say 1200 baud (n = 8), and type <CTRL A>8B<RETURN>. That will set the baud rate.

Character Length
The character length is the number of bits of data the make up a single character. The normal ASCII character set has 128 characters (designated 0 to 127) and can be expressed as 7 bit binary numbers (2 to the 7th power). In some situations graphics characters are sent as characters 128 to 255 and 8 bit binary numbers (2 to the 8th power) are needed. Select the character length you need, or leave the setting as currently configured.

Stop Bits
Stop bits are used to mark the end of a character and separate the data characters. The important thing about stop bits is that the computer and the serial device agree on the number of stop bits being used. Check with the other device to see how many stop bits are needed, then set the computer accordingly.

The commands to select character length and stop bits are combined in the Data Format command:

<CTRL A> n D   Set Data Format (n = 0 to 7)

This command sets the number of bits in the character and the stop bits sent for serial port transmissions. Enter n for the combination of bits selected.

0 = 8 data, 1 stop	4 = 8 data, 2 stop
1 = 7 data, 1 stop	5 = 7 data, 2 stop
2 = 6 data, 1 stop	6 = 6 data, 2 stop
3 = 5 data, 1 stop	7 = 5 data, 2 stop

Parity

Parity is a procedure that tests the accuracy of transmission of each character. It can be used or not used. If it is used, there are a number of different types available, such as Odd or Even, etc. The important thing about the parity selection is that the computer and the serial device agree on what parity to expect. Check with the other device to see what parity is needed, then set the computer to provide it.

<CTRL A> n P   Set Parity (n = 0 to 7)

This command sets the parity selection used for the serial interface protocol. Enter for n the character that corresponds to the parity selection you want.

n	Parity Selection
0	None
1	Odd parity
2	None
3	Even parity
4	None
5	Mark parity
6	None
7	Space parity

When you have finished these protocol settings you can test the serial connection by using the same simple test procedures suggested above for the parallel printer except substituting the Franklin DOS 2 command PR#2 because the serial connection is to slot #2 of the computer.

If the device is a serial printer, try printing a few characters or sentences. If the device is a modem, you won't be able to see much because the power-up setting has Screen Echo set to Off. The command to turn On Screen Echo is <CTRL A> I. With Screen Echo On, you will see every character on the screen that is transmitted to the serial port.

Input/Output Commands For Slot 1 And Slot 2

The information that follows is for more experienced users and relies on some familiarity with computer terminology and software design. If you're interested in how Franklin DOS 2 and the ACE 500 Computer works, you may find this information helpful, otherwise, you can ignore it. You don't have to understand these commands or procedures to operate your computer successfully.

For Franklin DOS 2, the parallel (printer) interface appears as slot number 1 and the serial (modem) interface appears as slot number 2. The device in either slot is activated with the Franklin DOS 2 command PR#n. This directs all output to that slot (and also the screen, depending on the setting of screen echo) until <CTRL><RESET> is pressed or until another PR# is given with another slot number. Entering PR#0 is recommended since slot 0 is unoccupied and thus will not create a conflicting command. Note that by default, output will not go to the screen for slot 1 or for slot 2 (screen echo is off for the both interfaces in the power-up configuration). You can select transmission of data to the screen for either slot by using software control commands as described in the next sections.

The Franklin DOS 2 command PR#2 directs output to slot 2 (the modem port). The additional Franklin DOS 2 command IN#2 causes all input to be taken from the modem serial port rather than the keyboard. <CTRL><RESET> or an IN# command to another slot number will deactivate modem input. Note that a new IN# command (such as IN#0) must come through the modem port to be effective.

The commands to operate slots 1 and 2 are similar but different. They are shown in separate groups below.

Input And Output Commands For Slot 1

Most of the time you will probably use the printing commands of the application programs you're using. But for your own programs and special needs the ACE 500 Computer has a number of built-in software commands and controls for printers. These commands and descriptions of how to use them follow. Understanding these commands is not necessary to use your computer or printer effectively, though they make some processes easier.

When you send data from the computer to the printer you can change some of the interface options for the appearance of the printout with commands from the keyboard. These commands are preceded by a "command character" which informs the computer that a command follows. In the power-up setting, this command character is <CTRL I>, the TAB character, (ASCII 9). In BASIC programs you would use CHR$(9). The individual commands are letters preceded by this command character.

Because the command character is the TAB character, which you may need to use in your text, you may want to change the command character to something else. The command for changing the command character is the last command in the following list. Note that <CTRL I> in the list stands for the command character and could be changed to any other ASCII character you have selected.

Slot 1 Printer Control Command Summary

Command	Function
<CTRL I> A	Epson screen dump
<CTRL I> B	ignored
<CTRL I> C	C. Itoh screen dump
<CTRL I> D	ignored
<CTRL I> E	Okidata screen dump
<CTRL I> F	Select inverse image dump
<CTRL I> G	Select normal image dump
<CTRL I> H	Select hires graphics dump
<CTRL I> I	Turn on screen echo
<CTRL I> J	Turn off screen echo
<CTRL I> K	Turn off auto linefeed
<CTRL I> L	Turn on auto linefeed
<CTRL I> M	Select primary page dump
<CTRL I> nnnN	ignored
<CTRL I> O	Select non-rotated dump
<CTRL I> P	ignored
<CTRL I> Q	Select rotated dump
<CTRL I> R	Reset all settings to defaults
<CTRL I> S	ignored
<CTRL I> T	Select text dump
<CTRL I> U	Allow character codes 0-255
<CTRL I> V	Restrict character codes to 0-127

\<CTRL I\> W	Select secondary page dump
\<CTRL I\> X	ignored
\<CTRL I\> Y	Gemini screen dump
\<CTRL I\> Z	Enter transparent mode
\<CTRL I\> nnn	ignored
\<CTRL I\> \<ctrl chr\>	Change command designator character

These individual commands are explained in more detail below. In general, these commands allow you to send the contents of the screen to the printer. This is called a screen "dump". There are commands that allow you to modify the characteristics of the printed screen, such as rotating it, or showing it in reversed video. Commands to change these characteristics are given before the screen dump command.

Slot 1 Printer Software Default Settings

The following settings are the default values for the printer screen dump characteristics. These are the settings for the computer from the factory and are used after each power On or each \<CTRL I\> R Reset command.

Output Characteristic	Default Setting
Screen Echo	Off
Screen Dump	Hires graphics
Memory Page	Primary page
Video Display	Normal video
Rotation	Not rotated
Auto Linefeed	Same as OPTION SWITCH #6
Code Range	0 to 127 only

Slot 1 Printer Software Commands

\<CTRL I\> A   Epson screen dump
This command prints a screen of data in the format needed for an Epson printer. Set the parameters for the printing characteristics before you use this command. After it has started, the screen dump can be stopped by pressing ESC. If you want to print out a hires graphics screen you must have the Graftrax option installed on your Epson printer.

\<CTRL I\> C   C. Itoh screen dump
This command prints a screen of data in the format needed for a C. Itoh printer. Set

the parameters for the printing characteristics before you use this command. After it has started, the screen dump can be stopped by pressing ESC.

<CTRL I> E   Okidata screen dump

This command prints a screen of data in the format needed for an Okidata printer. Set the parameters for the printing characteristics before you use this command. After it has started, the screen dump can be stopped by pressing ESC. If you want to print out a hires graphics screen you must have the Oki-Plot option installed on your printer.

<CTRL I> F   Select inverse image dump
<CTRL I> G   Select normal image dump

The screen dump can be printed either as normal printing (the light dots on the screen will be printed as black dots by the printer) or as inverse printing (the light dots are printed as the non-printed dots by the printer). These commands inform the printer software which printing image is desired. This selected printing mode remains in effect until another command is given to change it.

<CTRL I> H   Select hires graphics dump

This command informs the printer software that the next screen dump is for a hires graphics screen. The software will assume all subsequent commands apply to hires graphics screens until it receives a command for a text screen dump (CTRL I T). The hires graphics screen dump is the default setting.

<CTRL I> I   Turn on screen echo
<CTRL I> J   Turn off screen echo

These commands turn the screen echo on and off. Screen echo means that each character that is printed is also displayed on the screen. Normally screen echo is on so you can follow the progress of the data being printed. Printing is not affected if you turn off the screen echo.

<CTRL I> K   Turn off auto linefeed
<CTRL I> L   Turn on auto linefeed

These commands duplicate and override the setting of OPTION SWITCH #6 on the back panel of the computer. This switch controls whether the computer sends a line feed command at the end of every line, or not. Refer to the earlier section of this chapter on connecting the printer for further discussion of the line feed commands and determining the requirements of your printer.

<CTRL I> M   Select primary page dump
<CTRL I> W   Select secondary page dump
The ACE 500 Computer computer has video memory allocated for two separate
"pages" of display for each display mode. That means there are two pages of text
screens and two pages of graphics screen data stored in memory at the same time.
These commands select either the primary (page 1) or the secondary page (page 2) as
the page to be sent to the printer. This assignment remains in place until another page
assignment command is given. The memory is unaffected by the printing operation.

<CTRL I> O   Select non-rotated dump
<CTRL I> Q   Select rotated dump
The data on the screen can be printed exactly as it appears on screen or it can be
rotated 90 degrees so the contents of the display screen appears on the long axis of
the printed page. These commands allow you to set the rotation of the screen dumps.
The rotation assignment remains in place until another rotation assignment is given.

<CTRL I> R   Reset all settings to defaults
This command resets all slot 1 interface settings to the factory default settings.

<CTRL I> T   Select text dump
This command informs the printer software that subsequent screen dumps are text
data. This is the complementary command to the Select hires screen dump command
(CTRL I H). This assignment remains in place until a hires graphics dump command
is given.

<CTRL I> U   Allow character codes 0-255
<CTRL I> V   Restrict character codes to 0-127
The data transmitted to the printer is normally sent as character codes in the range of
0 to 127, corresponding to ordinary text characters. If codes between 128 and 255
are received in normal printer operation, the printer software subtracts 128 from each
character code to make it correspond to normal range characters. In some situations,
such as printing graphics characters, you may need to use character codes 128 to 255
and the printing characters they represent. These commands allow you to control the
transmission of these character codes.

<CTRL I> Y   Gemini screen dump
This command prints a screen of data in the format needed for a Star Micronics
Gemini printer. Set the parameters for the printing characteristics before you use this
command. After it has started, the screen dump can be stopped by pressing ESC.

<CTRL I> Z   Enter transparent mode
Transparent mode is an operating environment of the computer in which characters
are not analyzed or interpreted but treated transparently and passed through to the
printer. This is useful because control codes, commands, command designator char-
acters (like the TAB character), don't complicate the printing of a document.

For example, if you use the command designator character <CTRL I> (TAB, the
default value) and try to print a document which has TAB characters in it, the printer
software won't know how to interpret the TAB characters it finds, whether to inter-
pret them as printing commands or to interpret them as TABs in the text. One solu-
tion to this problem is to change the command designator character to something not
used in the document. If the document has a lot of control characters in it, such as a
program, this solution may not be feasible and the only remaining option may be to
put the printer software into transparent mode so it treats all characters as pure data.

There is a problem with using transparent mode to print. Because transparent mode
doesn't recognize commands, there isn't a way to issue a command to exit from
transparent mode. Your only choices are to reset the computer or issue a PR #1 com-
mand, both of which reinitialize the printer software and exit transparent mode. Be
careful when you use transparent mode and be sure you execute all the commands
needed to print your document correctly before you enter it.

<CTRL I> <ctrl chr> Change command designator character
This command allows you to change the command designator character to some
control character other than <CTRL I>. Just enter the new command designator
control character after the <CTRL I> . For instance, <CTRL I> <CTRL K>
would make the new command designator character <CTRL K>, which you would
have to use to enter commands.

**Input And Output
Commands For Slot 2**

The Franklin ACE 500 computer has a number of hardware and software features
built in to help you manage the operation of the serial interface. These features do
three things: allow quick and easy configuration of the serial interface protocol, set up
printing and communications protocols for application programs, and provide a sim-
ple terminal mode for other communication tasks. You may not need all of these fea-
tures right now, but it's nice to have them available to you. Look over this material
to see the things presented here, then refer to it as you need it.

The serial port software is activated by the PR#2 <RETURN> command from the
keyboard, or the PRINT PR#2 command from BASIC programs. This sends all
computer output, and PRINT statements, to the serial port. You can activate the

serial port software for input by the IN#2 <RETURN> command, or the corresponding PRINT command. This means all input requests must be received from the serial port. Commands to the serial port must be part of the output stream, so the PR#2 command must precede them.

The serial port software can be deactivated by switching to another port, such as PR#0 (an unused slot), or PR#6 (the disk drive slot) which will boot the diskette. Commands to reboot deactivate the software but do not change the serial port configuration. Commands to Reset, or Power On, deactivate the software and return the port configuration to the factory set state.

The factory set configuration for the serial interface is:

Selection	Factory Setting
Screen Echo	Off
Auto Linefeed	Off
Character Set	0 - 127 only
Baud Rate	300 baud
Character Length	8 bits
Parity	None
Stop Bits	1

Slot 2 Serial Interface Command Summary

This list summarizes the serial port software commands:

Command	Function
<CTRL A> A	Clear Screen (terminal mode only)
<CTRL A> nn B	Set Baud Rate (n = 1 to 15)
<CTRL A> C	ignored
<CTRL A> n D	Set Data Format (n = 0 to 7)
<CTRL A> E	Printer Echo On (terminal mode only)
<CTRL A> F	Printer Echo Off (terminal mode only)
<CTRL A> G	ignored
<CTRL A> H	Invoke Parameter Menu (terminal mode only)
<CTRL A> I	Screen Echo On

<CTRL A> J	Screen Echo Off
<CTRL A> K	Auto Linefeed Off
<CTRL A> L	Auto Linefeed On
<CTRL A> M	40 Column Screen (terminal mode only)
<CTRL A> nnN	ignored
<CTRL A> O	80 Column Screen (terminal mode only)
<CTRL A> nP	Set Parity (n = 0 to 7)
<CTRL A> Q	Quit Terminal Mode (terminal mode only)
<CTRL A> R	Reset To Default Settings
<CTRL A> S	ignored
<CTRL A> T	Enter Terminal Mode
<CTRL A> U	Allow Character Codes 0 - 255
<CTRL A> V	Restrict Character Codes To 1 - 127
<CTRL A> W	ignored
<CTRL A> X	ignored
<CTRL A> Y	ignored
<CTRL A> Z	Enter Transparent Mode
<CTRL A> nn <CR>	ignored
<CTRL A> <ctrl chr>	Change Command Designator Character

Slot 2 Serial Interface Software Commands

Once the Serial Interface Software is activated, you can enter commands to use its features. These commands all use a command introducer, which is <CTRL A> in the power-up configuration. This character is used before all serial interface software commands. When the computer sees the <CTRL A> character and the next character is a valid command character, this command is not sent to the serial port but immediately processed by the computer. If the command is not a valid one, both the <CTRL A> and the following characters are sent to the serial port.

If you want to send a <CTRL A> character to the port as part of your data or something, you have two options: you can change the command introducer character to something besides <CTRL A>, or you can use Transparent Mode transmissions which transmit all characters without any processing. Both of these options are discussed in detail below.

<CTRL A> A   Clear Screen (terminal mode only)
This command clears the screen. It operates only in Terminal mode (refer to the T command).

<CTRL A> nn B   Set Baud Rate (nn = 1 to 15)
This command sets the baud rate of the serial interface. Enter nn as the number of the baud rate selected.

nn	Baud Rate
1	50 baud
2	75 baud
3	110 baud
4	135 baud
5	150 baud
6	300 baud
7	600 baud
8	1200 baud
9	1800 baud
10	2400 baud
11	3600 baud
12	4800 baud
13	7200 baud
14	9600 baud
15	19200 baud

<CTRL A> n D   Set Data Format (n = 0 to 7)
This command sets the number of bits in the character and the stop bits sent for serial port transmissions. Enter n for the combination of bits selected.

0 = 8 data, 1 stop	4 = 8 data, 2 stop
1 = 7 data, 1 stop	5 = 7 data, 2 stop
2 = 6 data, 1 stop	6 = 6 data, 2 stop
3 = 5 data, 1 stop	7 = 5 data, 2 stop

<CTRL A> E   Printer Echo On (terminal mode only)
<CTRL A> F   Printer Echo Off (terminal mode only)
Printer echo causes all transmitted and received characters from the serial port to be sent also to the parallel printer. This command works only if you're in Terminal mode (refer to the T command). These commands turn printer echo on and off. The default setting is printer echo off.

<CTRL A> H   Invoke Parameter Menu (terminal mode only)
This command displays a menu on the screen that helps you change the communica-

tions parameters of the serial port. This command only operates when you are in Terminal mode (refer to the T command). When the Help Parameter Menu is on the screen, only the following keys are active:

Key	Key Effect
down arrow	select the next parameter to change
up arrow	select the previous parameter to change
SPACE	change the parameter to its next value
RETURN	keep the parameters as selected on Menu and exit Help Parameter Menu
ESC	don't change parameters previously selected and exit Help Parameter Menu

When you have the Help Parameter Menu on the screen, the arrow keys move from one parameter to the next, then the SPACE bar moves through all the possible settings of that parameter. If you want to keep the settings you've chosen, exit the Menu with RETURN. If you don't want the new settings but want the original settings, exit the Menu with ESC.

<CTRL A> I   Screen Echo On
<CTRL A> J   Screen Echo Off
These commands control screen echo, which determines whether characters sent to the serial port are displayed on the screen or not. Screen echo off doesn't display characters on the screen, and is the power-up setting of the computer.

<CTRL A> K   Auto Linefeed Off
<CTRL A> L   Auto Linefeed On
These commands control the Auto Linefeed function for the serial port. Auto Linefeed On means that a linefeed character is automatically sent to the serial port (printer, modem, etc.) after every carriage return character.

<CTRL A> M   40 Column Screen (terminal mode only)
<CTRL A> O   80 Column Screen (terminal mode only)
These commands switch the screen display from 80 to 40 column mode and back again. They operate only in Terminal mode (refer to the T command).

<CTRL A> nP   Set Parity (n = 0 to 7)
This command sets the parity selection used for the serial interface protocol. Enter for n the character that corresponds to the parity selection you want.

n	Parity Selection
0	None
1	Odd parity
2	None
3	Even parity
4	None
5	Mark parity
6	None
7	Space parity

<CTRL A> Q   Quit Terminal Mode (terminal mode only)
This command exits Terminal mode. The command operates only if you're in Terminal mode (refer to the T command).

<CTRL A> R   Reset To Default Settings
This command resets all the slot 2 interface settings to the power-up default settings.

<CTRL A> T   Enter Terminal Mode
This command enters Terminal mode of operation for the serial port. It is especially useful for bi-directional communication with the serial port. With Screen Echo On, both input and output characters are displayed on the screen. Other serial interface software commands can be used in this mode with the <CTRL A> command introducer. These commands are processed locally by the computer and not transmitted to the serial port.

<CTRL A> U   Allow Character Codes 0 - 255
<CTRL A> V   Restrict Character Codes To 1 - 127
These commands permit or restrict the serial port transmissions to ASCII character codes in the ranges indicated. Using the full character set is necessary for some kinds of graphics transmissions where entire range of codes is needed. When transmission is restricted, characters which are received in the range of 128 - 255 will have 128 subtracted before they are transmitted.

<CTRL A> Z   Enter Transparent Mode
This command enters transparent mode of operation for data being sent to the serial port. This means that no characters will be processed, including control characters and the CTRL A command introducer, but will be transmitted "transparently" to the serial device as pure data. This is useful for transmitting data or programs that have a

lot of control characters which might otherwise change the serial port settings unintentionally. The disadvantage to Transparent Mode is that you cannot give it a command to change anything, including the Transparent Mode setting. You have to reintialize the serial port software with the PR#2 command or use <CTRL><RESET> to exit Transparent Mode and regain control of the computer. Also remember to send all the needed configuration commands before you send the Transparent Mode command.

<CTRL A> <ctrl chr>   Change Command Designator Character

This command allows you to change the command designator character from <CTRL A> to some other control character of your choice. Enter for <ctrl chr> the control character you choose for the new command designator. Remember that all your commands must now use this designator. For instance, the command <CTRL A> <CTRL B> would change the command designator to <CTRL B> and all commands would have to begin <CTRL B>.

Glossary

Acoustic coupler — A device into which you place the handset of a telephone to allow transfer of information between computers. See Modem.

ALT key — A shift key generally used only in technical applications that produces special ASCII codes with other keys.

Ambiguous file name — A file name that refers to more than one file. See Wildcard characters.

Application program — A program such as a word processor or spreadsheet that allows you to apply your computer's power to a particular task.

ASCII — American Standard Code for Information Interchange. A set of 128 numerical codes for printed characters and control characters.

ASCII file — A text file in which data is recorded in ASCII code form, that is, character by character.

Backup — A copy of a diskette.

BASIC — Beginner's All-purpose Symbolic Instruction Code. A computer language.

Baud rate — The speed of data transmission between a computer and a peripheral, given in bits per second.

Binary file — A file, usually on diskette, that holds data in binary form. See ASCII file.

Boot — To insert a diskette and start a computer so that a disk operating system is loaded into memory from the diskette.

Buffer — A quantity of memory where data temporarily waits. A buffer is used as a holding point until the preceding process is completed.

Bug — An error in a program that prevents it from running as planned.

CAPS LOCK key — A special kind of shift key that shifts all alphabetic characters to uppercase when the LED is on. Other uppercase symbols and punctuation marks are unaffected.

Caret — A typographic character; an arrow head pointing up. An ASCII character, also called up-arrow, used with the corresponding letter to symbolize a control code. For example, <CTRL> N is also written ^N. See Control code.

Catalog — A list of files on a diskette. or The Franklin DOS 2 command to display the list of files on the diskette.

Central Processing Unit — The CPU stores and retrieves data in memory and executes logical and arithmetic operations on it. The simple operations performed by the CPU are combined with each other to make higher level operations. The CPU includes the microprocessor and supporting circuits.

Character — A symbol on the keyboard, including the space and control keys like <TAB> and <RETURN>. See ASCII, Control codes.

Character Set — The collection of characters corresponding to the ASCII codes 0 to 127. Generally this is a standard set of alphabetic characters, but it may also be any group of graphic or other characters.

Cold boot — Starting a computer, usually from a power off state, so that all memory is erased and the operating system is read in from scratch.

Command — An instruction from the user to the operating system or application program.

Compatibility — Similarity between one computer and another that allows the same program or diskette to be used on both of them.

Control codes — The ASCII codes generated by certain keys when used with <CTRL>. For convenience, some control codes are produced by special keys, such as <ESC> and <RETURN>. See ASCII and Caret.

CPU — See Central Processing Unit.

Cursor — A mark on the screen which shows where the next character will appear, or which designates the next function that will be performed. Usually an underline or

solid rectangle blinking on and off. The appearance of a different type of cursor is often an indication that different operating software is active.

Data — A very general term for information that is available in a machine readable form so it can be processed by a computer.

Default, Default value — The choice among many options the computer will use if you don't specify otherwise.

Destination — The diskette or other device to which data will be sent, especially the diskette, or drive containing the diskette, on which a copy is being made. See Source.

Device — A very general term for something attached to a computer to perform a specific task. Modems, printers, and disk drives are devices.

Direct-connect modem — A modem which is wired directly into the telephone system. An acoustic modem produces sounds that are received by the telephone microphone and converted into electrical signals by the telephone. A direct connect modem generates the signals itself and connects directly into the telephone network. See Modem.

Disk, diskette — A magnetic recording medium used to store data, especially in personal computers. A diskette consists of a flexible disk which spins inside a protective envelope.

Disk drive — The mechanism which spins the diskette and records or plays back the data stored on the diskette.

Display — The characters and graphics which appear on the computer's monitor. See Monitor.

DOS — Disk Operating System. Several different proprietary operating systems use the generic term DOS. Franklin DOS 2, used on Franklin computers, is compatible with DOS 3.3, used on Apple computers of the II/IIe family, and ProDOS, used with the Apple IIc computer. See Operating system.

80/40 Column switch — A push button switch above the keyboard to aid software selection between 80 and 40 column display modes.

Enter — To type data into the computer.

Error message — A report from a computer program (operating system or programming language) that something has failed to go as planned.

Escape key, escape sequence — A command that is initiated with the <ESC> key, followed by one or more other keys.

File — An organized and named collection of data in memory or on a diskette, either individual information items or a program.

File name — The name which identifies a file to the operating system for storage and retrieval.

File type — A classification of files according to purpose. Each operating system has different conventions for naming these classifications, but the names are important to identify for the computer what handling and interpretation is required of each file.

Floating Point BASIC — A version of the BASIC language in which decimal points are permissible in numbers. Used with Franklin DOS 2. Similar to other popular versions of BASIC.

Floppy diskette — A reference to the flexibility of the diskette, though you should still be careful about handling the diskette and not bending it. See diskette.

Format — To prepare a diskette for use by recording an organizational scheme upon it.

Function key — A key which can produce a function, command, or group of characters, instead of just a single character. This function is a stored definition entered by the keystroke.

Graphics mode — A feature of ACE 500 computers which allows pictures to be drawn on the screen.

Handshaking — Control signals transmitted between two devices to coordinate receiving and transmitting data, especially in serial interface communications. The signals may be on data lines or on separate communication lines depending on the handshaking controls chosen.

Hardware — The electronic circuits and mechanical components that make up a computer.

Hard disk — A data storage system using one or more rigid magnetic disks as media for recording data. The storage capacity of a hard disk is very much larger than a floppy disk.

Hex — An abbreviation for Hexadecimal. Hexadecimal refers to a base 16 number system, used for convenience in programming. The numbers 10 through 15 are represented by the letters A through F, thus 1E is 30, FF is 255, etc.

Initialize — To format a diskette. More generally, to make a device or a program ready for use by clearing its contents.

Interface — The logic circuits which transmit data between the computer and the peripheral device, including keyboard and disk drive as well as external devices like printers and modems.

Inverse (Reverse) video — A type of video display in which characters appear as dark objects on a light background.

I/O (Input/Output) — A very general term for the communication between the computer and external peripheral devices.

Joystick — A game controller.

K — An abbreviation for kilo, meaning one thousand. In computer usage it means 2 to the tenth power, or 1024, and needs further designation of units, such as Kilobytes of memory.

LED — Light Emitting Diode. A small light used as an indicator of activity. The disk drive lights, the Power light, the Caps Lock key light, etc. above the keyboard are all LED's.

Load — To bring a program or data from a diskette or other location into computer memory.

Machine — Computer.

Main electronics board — The large board on which are mounted the memory circuits, the CPU and other processing circuits of a computer.

Media — The physical material used by a storage device to record data. Usually

magnetic tape or magnetic coatings on a diskette.

Memory — Data storage circuits that are directly available to the CPU. See CPU, RAM, ROM.

Menu — A popular method of control used in programs. A list of options is presented on the screen for the user to view and make a choice.

Microprocessor — The principal component of the CPU. It performs mathematical and logical operations on data stored in memory. See Central Processing Unit, Operating System.

Modem — Modulator/Demodulator. Converts data from a computer representation to another kind of signal, either electrical impulses or audible tones, for reception and transmission through telephones. See Acoustic coupler, Direct-connect modem.

Monitor — An output device for a computer, smaller than a television set, but designed for the display of characters.

Motherboard — The colloquial name for the main electronics board. Derived from its ability to accept and support other circuit boards.

Mouse — A hand-held serial communications device you slide around the desk surface to control movement of the cursor and other computer operations.

NUM LOCK key — A key that controls the operation of the numeric keypad either to produce the numbers on its keytops or the editing functions marked there. The LED indicates number mode.

Operating system — A program which performs often-used tasks such as disk drive and keyboard access so that those commonly used instructions will not have to be written into every program. Operating systems are specific to microprocessors or families of them. Franklin DOS 2 runs only on the 65SC02, or 6502 microprocessor.

Paddles — A game controller.

Parallel interface — A communications circuit used to transmit data over eight lines simultaneously.

Parameter — Data value transmitted as part of a command to a device or program

to make it perform a specific function or in a specific way.

Parity bit — A signal added to the data in serial communications. Used as a check on the accuracy of transmission.

Pascal — A high-level computer language.

Peripheral device — Something you attach to the computer which adds to its capability or performance, such as a printer, a modem, or an external disk drive.

Pixel — A tiny point on the display screen.

Port — A memory location through which the CPU communicates with peripheral devices or a connection from the computer to some external device. Thus a printer has a physical port where it attaches to the computer and a memory port where the computer sends it data to be printed.

Power supply — The ACE 500's power transformer that converts the power from a normal wall outlet to power the computer can use.

Program — A set of instructions which directs the operation of the computer. Also, to write these instructions.

Programmable keys — These are keys which can receive, store, and generate a definition you program onto the key. See also Function keys.

PROM — Programmed Read Only Memory. Memory circuits which hold data with power off and cannot be changed after the original storage of data. Differs from ROM in that a PROM is empty and data is stored by a relatively inexpensive device when needed.

Prompt — A character, word, phrase, or statement that tells the user to supply input for a program.

RAM — Random Access Memory. Memory circuits in a computer which hold data only while power is on, but which can be rapidly changed, read, and erased. Random access really means direct access. Any memory cell can be read or changed without reference to any other.

Read — The computer operation of securing previously recorded data from a diskette.

Reset — A command that interrupts whatever process the computer is involved in. The results of the interruption depend on the software instructions for reset currently being used. Also a key on the keyboard.

RF modulator — A device that turns computer display output into a simulated TV broadcast, so it can be displayed on a TV set.

ROM — Read Only Memory. Memory circuits which can be accessed directly, but which hold data even with power off and cannot be changed after data is originally stored.

RS-232 — An engineering standard for the transmission of data. Used with the serial interface of almost all personal computers.

Run — To execute a program in a computer.

Save — To store data or a program in a diskette file.

Scroll — A process on the display screen that moves the lines of text up or down to make room for more text.

Sector — A subdivision of a track on a diskette. See Track.

Serial interface — A data transmission circuit using a single line for data.

Slot — An elongated socket on the main electronics board which holds a peripheral board, or the address in memory used to access the peripheral device connected to that position on the board.

Software handshaking — Status information exchanged between computer and peripheral as control codes in the data stream.

Source — The diskette, or the disk drive holding it, from which data is being read. Especially, the diskette from which a copy is being made.

Start up diskette — A diskette prepared with a Disk Operating System on it so it can be used to boot, or start up, the computer.

Stop bits — The signal used in serial communications to indicate that an entire character has been transmitted.

Terminal — A combination of keyboard and display screen which communicates with a computer (usually CPU and memory functions) in another location.

Toggle — A command to the computer which is always given in the same format, but which has alternate effects. Equivalent to a push button light switch, which is always pushed, but turns the light on or off depending on the previous state of the light.

Track — One of the concentric circles that divides diskettes for storage purposes. The ACE 500 disk drives can use 35 or 40 track diskette formats.

Transparent mode — A processing mode of the computer where all characters, even command characters, are treated as data and displayed or printed. Setting transparent mode eliminates almost all normal command control of the computer, since all commands except reinitializations like PR#n, are treated as data only.

Volume — A seldom used designator for individual diskettes recognized by Franklin DOS 2.

Utility — A program or series of programs that sort through files, reorganize the contents of a diskette, and perform a number of other housekeeping or maintenance functions.

Warm boot — A program restart command, CTRL C or BREAK, which breaks the program processing but leaves the operating system intact.

Wildcard character — A character used in a file name to represent another character or string of characters.

Write — To record data onto a storage medium such as a diskette. Also, to send data to a peripheral.

Write-protect tab — A piece of adhesive paper which covers a notch on the side of a diskette envelope. A disk drive will not write or make any change to a diskette with this notch covered. Note, this applies to 5 1/4″ diskettes only. Other size diskettes use other write-protect methods.

Appendix

ASCII Character Set   The charts show the decimal number for the chart location of each character and the character displayed on the screen for that ASCII code value.

000 N_L	001 S_H	002 S_X	003 E_X	004 E_T	005 E_Q	006 A_K	007 B_L	008 B_S	009 H_T	010 L_F	011 V_T	012 F_F	013 C_R	014 S_O	015 S_I
016 D$_E$	017 D$_1$	018 D$_2$	019 D$_3$	020 D$_4$	021 N$_K$	022 S$_N$	023 E$_B$	024 C$_N$	025 E$_M$	026 S$_B$	027 E$_C$	028 F$_S$	029 G$_S$	030 R$_S$	031 U$_S$
032	033 !	034 "	035 #	036 $	037 %	038 &	039 '	040 (	041)	042 *	043 +	044 ,	045 —	046 .	047 /
048 0	049 1	050 2	051 3	052 4	053 5	054 6	055 7	056 8	057 9	058 :	059 ;	060 <	061 =	062 >	063 ?
064 @	065 A	066 B	067 C	068 D	069 E	070 F	071 G	072 H	073 I	074 J	075 K	076 L	077 M	078 N	079 O
080 P	081 Q	082 R	083 S	084 T	085 U	086 V	087 W	088 X	089 Y	090 Z	091 [	092 \	093]	094 ^	095 —
096 `	097 a	098 b	099 c	100 d	101 e	102 f	103 g	104 h	105 i	106 j	107 k	108 l	109 m	110 n	111 o
112 p	113 q	114 r	115 s	116 t	117 u	118 v	119 w	120 x	121 y	122 z	123 {	124		125 }	126 ~	127 ▉

Mouse Character Set

@	A	B	C	D	E	F	G

H	I	J	K	L	M	N	O

P	Q	R	S	T	U	V	W

X	Y	Z	[	\	]	^	_

ACE 500 Computer Pin Assignments

The following diagrams give the pin assignments for the connectors on the ACE 500 Computer.

$\overline{\text{PSTRB}}$	1	○		
			○	14 NC
PRTR∅	2	○		
			○	15 NC
PRTR1	3	○		
			○	16 NC
PRTR2	4	○		
			○	17 NC
PRTR3	5	○		
			○	18 GND
PRTR4	6	○		
			○	19 GND
PRTR5	7	○		
			○	20 GND
PRTR6	8	○		
			○	21 GND
PRTR7	9	○		
			○	22 GND
NC	10	○		
			○	23 GND
PBUSY	11	○		
			○	24 GND
NC	12	○		
			○	25 GND
NC	13	○		
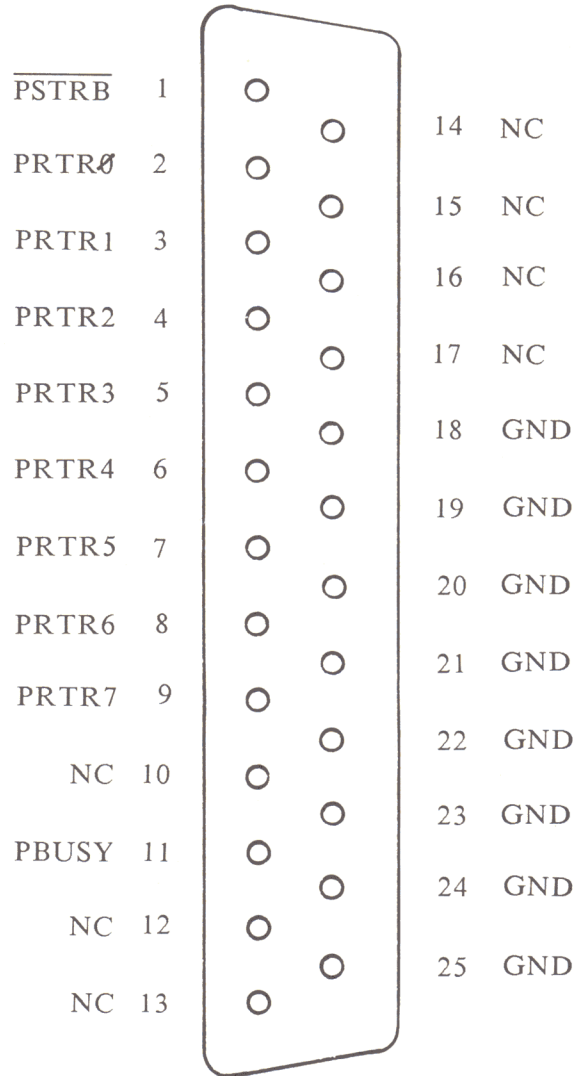
Parallel Printer Connector
DB-25 Female Connector

GND	1		
		11	PHASE0
GND	2		
		12	PHASE1
GND	3		
		13	PHASE2
GND	4		
		14	PHASE3
-12V	5		
		15	$\overline{\text{WRTGATE}}$
+5V	6		
		16	NC
+12V	7		
		17	$\overline{\text{DRVENB}}$
+12V	8		
		18	RDDATA
NC	9		
		19	WRTDATA
WRTPROT	10		
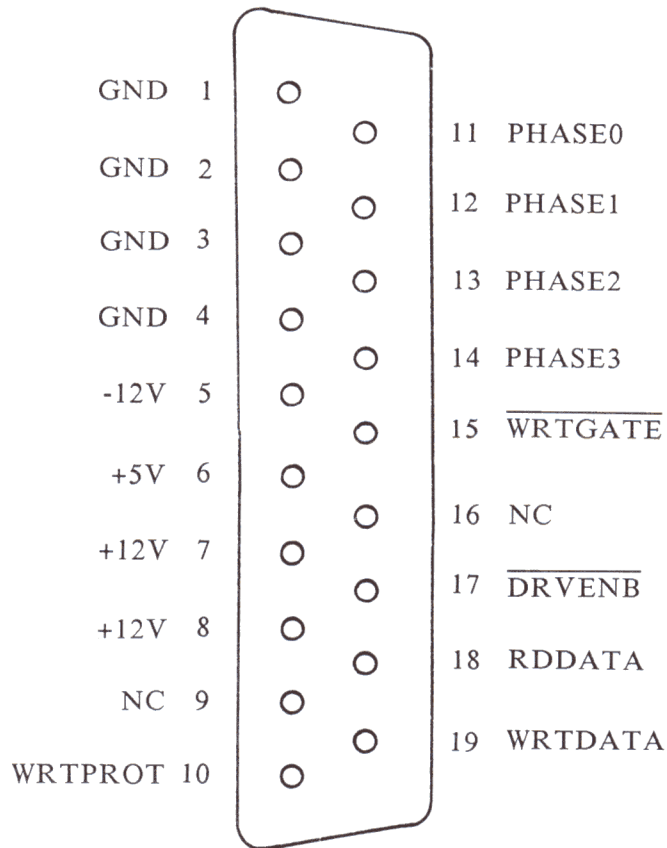
External Disk Drive Connector
DB-19 Female Connector

DSR2B 5　　　　　　　　1 DTR2B

RD2B 4　　　　　　　　2 TD2B

3 GND
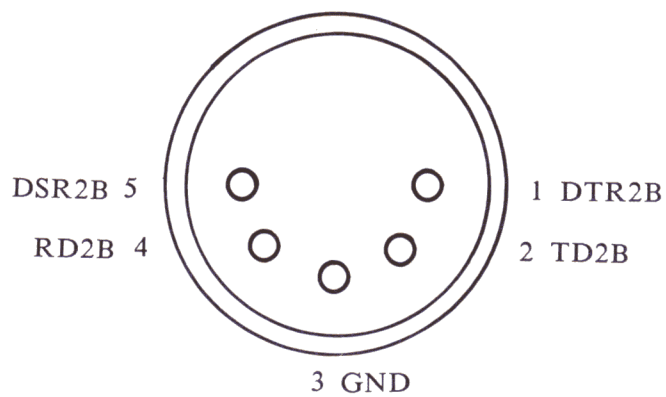
Serial Port
DIN 5 Connector

GND 1

GND 2

RED 3

GREEN 4

BLUE 5

6 INTENSITY

7 NC

8 COMP/HORIZ SYNC

9 VERT SYNC

RGB Port
DB-9 Female Connector

7 +5 Sense

+ 5V 6   1 GND Sense

+ 12V 5   2 GND

- 12V 4   3 GND
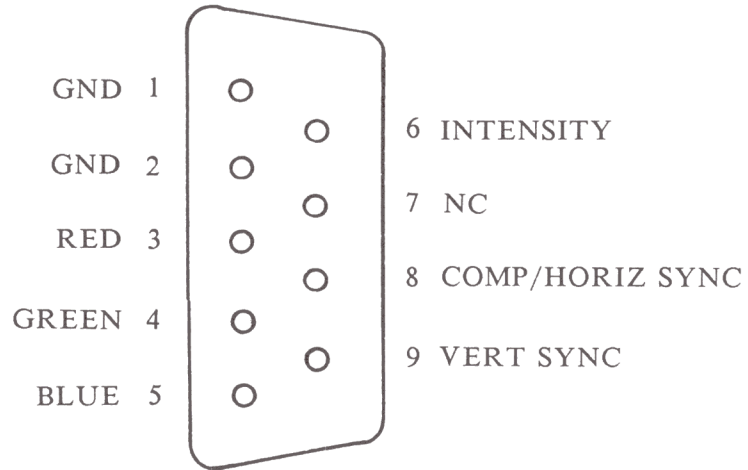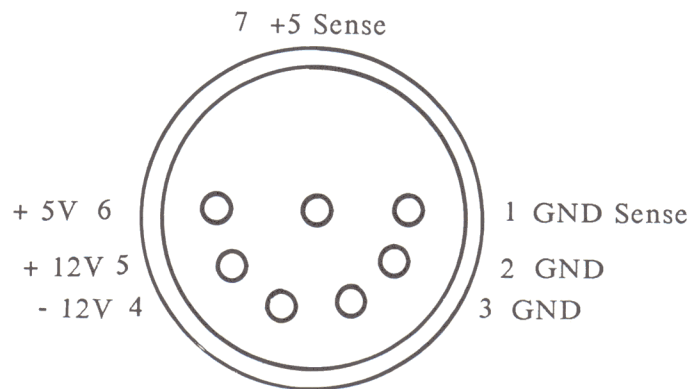
Power Connector
DIN 7 Connector

GameSW1 1 ○

+ 5V 2 ○

GND 3 ○

XDir 4 ○

PDL0/Xmove 5 ○

○ 6 NC

○ 7 GameSW0

○ 8 PDL1/YDir

○ 9 Ymove
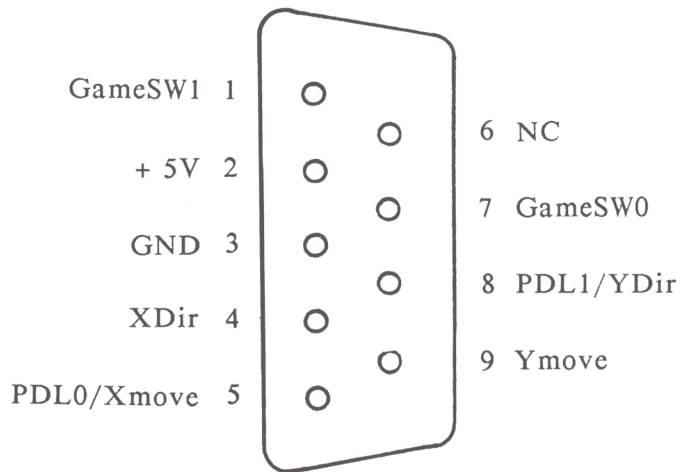
Mouse/Joystick Port
DB-9 Female Connector

Index

Franklin

ACE 500 User's Reference Manual

Franklin
Computer