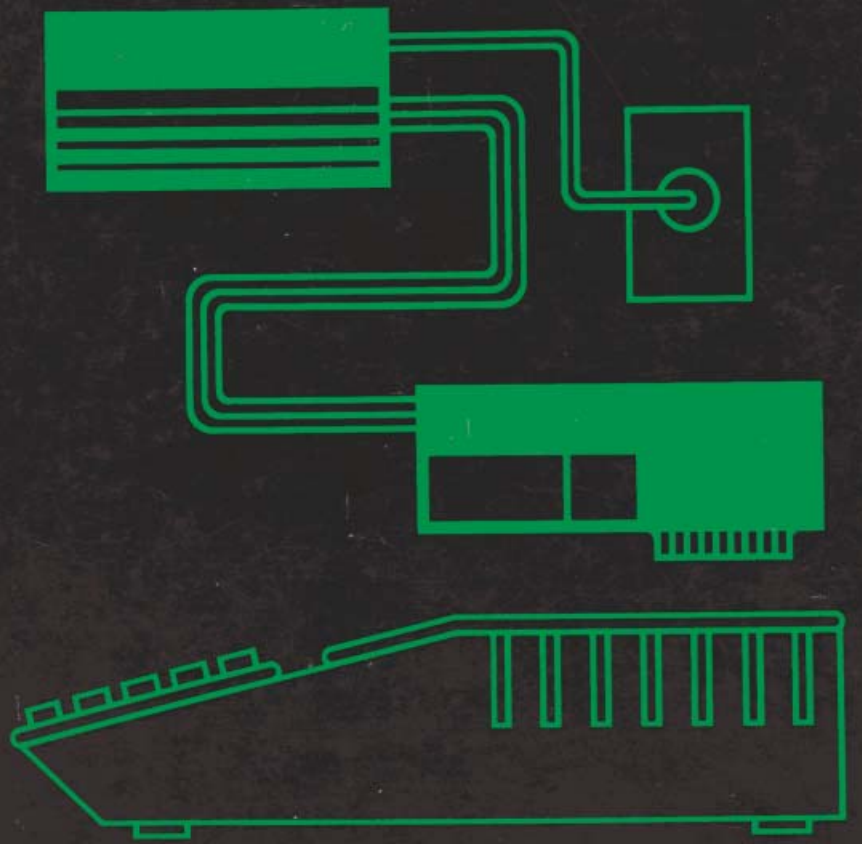
Hayes Microcomputer Products Inc.
5835 Peachtree Corners East, Norcross, Georgia 30092
98-101 161
Printed in U.S.A.

Hayes

MICROMODEM II™

OWNER'S MANUAL



EJABB
879.4981

MICROMODEM II OWNER'S MANUAL

A COMPLETE DATA COMMUNICATIONS SYSTEM FOR APPLE II AND BELL AND HOWELL COMPUTERS.

"This equipment is marketed pursuant to a waiver of FCC Rules Part 15 Subpart J. Operation of a computer with this equipment in a residential area may cause objectionable interference to radio and TV reception, because a computer with this equipment emits more radio frequency energy than the FCC Rules allow. If interference occurs, the user will be required to take all steps necessary to correct the interference."

Hayes Microcomputer Products Inc.

Second edition. Copyright © 1981 Hayes Microcomputer Products, Inc. All rights reserved. No part of this publication may be reproduced without prior permission of Hayes Microcomputer Products, Inc.

NOTICE

Please note the change in procedure below for using the SELF-TEST program with the Micromodem II.

When the message "UNPLUG THE MICROMODEM II FROM THE MICROCOUPLER" appears on the screen, disconnect the **modular telephone cable** from the Microcoupler.

DO NOT DISCONNECT THE BLUE RIBBON CABLE.

Disconnecting the ribbon cable may cause a properly functioning Micromodem II to receive a FAILURE reading from the SELF-TEST program.

The remainder of the SELF-TEST procedure is unchanged.

Hayes

Hayes Microcomputer Products Inc.
5835 Peachtree Corners East, Norcross, Georgia 300092

97-350 681

TABLE OF CONTENTS

Chapter 1: Introduction 1

Chapter 2: Installation 3
- Assumptions 3
- Parts 3
- Installing the Micromodem II Printed Circuit Board 5
- Installing the Microcoupler 5
- FCC Registration 5
- Notifying the Telephone Company 6
- Connecting the Microcoupler to the Telephone Line 6
- Precautions 6
  - Installing DOS 3.3 on the Micromodem II Diskette 7
  - Installing DOS 3.3 Using One Disk Drive 7
  - Installing DOS 3.3 Using Two Disk Drives 9

Chapter 3: Terminal Mode 11
- Entering Terminal Mode 11
- Dialing in Terminal Mode 12
- Hanging Up the Telephone in Terminal Mode 13
- Setting Baud Rate in Terminal Mode 14
- Stopping Unwanted Output in Terminal Mode 14
- Using a Printer in Terminal Mode 14
- Exiting Terminal Mode 15
- Example Session 1 (Terminal Mode) 16

Chapter 4: Remote Console Mode 19
- Entering Remote Console Mode 19
- Special Control Characters 20
- Pausing During Output in Remote Console Mode 20
- Resetting the Remote Apple II 20
- Going from Remote Console Mode to Terminal Mode 20
- Returning to Remote Console Mode from Terminal Mode 21
- Using a Printer in Remote Console Mode 21
- Hanging Up the Telephone in Remote Console Mode 21
- Cursor Movement in Remote Console Mode 22
- Example Session 2 (Remote Console Mode) 22

Chapter 5: Elementary Programming 25
- Dialing the Telephone 25
- Hanging Up the Telephone 27
- Answering the Telephone 27
- Transmitting and Receiving Data Via the Micromodem II 28

Chapter 6: Micromodem II Diskette Programs 29
SELF TEST 30
STORE & FORWARD 31
PICKUP 32
AUTO DIAL 32
DUMBO 33
TRANSFER 34
BASIC EXTRACTOR 35
ALARM 36
DIAL A HUMAN 36
ANSWER ON THE NTH RING MODIFIED 37
SOURCEON 38
DJ CONVERTER 38

Chapter 7: Micromodem II Memory Locations 39
DATA 39
STATUS/CRI 39
RI/CR2 40
OUTA 41
N, NØ, CHAR 41
ACIA 42
MODEM 42
LOCSE 42
FLAGS 42
CRDLY 44
Micromodem II Firmware 44
Advanced Programming Techniques 45
Selecting Baud Rate 45
Changing Character Formats 46
Sending a Break in BASIC 46
Turning off the Carrier Without Hanging Up 46
Entering Terminal Mode from a Program 48
Entering a Program from Terminal Mode 48
Automatic Log-on to a Time Share Computer 49
Micromodem II Memory Usage Chart 50

Chapter 8: Firmware Specifications 51
Entry Points 51
Default Initialization 51
Default Settings 51
Features of Input Routine (IIN) 52
Features of Output Routine (OUT) 53
Features of Dialing 54
Features of Terminal Mode 55
Software Controlled Options 56

Data Format 56
Baud Rate 56
Lower-to-Upper Case Translation 56
Options Selected Via Bits in Flag Byte 57
Additional Information 57

Chapter 9: Background Information 59
Modems 59
History 59
Transmission Speed 60
Communication Lines 60
Full-Duplex 60
Half-Duplex 61

Appendix A: Modifying and Using Datamover 63

Appendix B: ASCII Control Character Code Table 64

Appendix C: Return for Repair Procedure 65

Appendix D: Product Specifications 66

CHAPTER 1: INTRODUCTION

This Owner's Manual is intended for Micromodem II™ users. It provides detailed instructions for installation and use, descriptions of the functions of all programmable registers and controls, background information and suggestions. The manual does not contain technical information about the construction or design of the equipment.

Hayes Micromodem II is a complete data communications system for Apple II* and Bell and Howell computers. The Micromodem II allows a computer to communicate with another computer or be accessed by a remote terminal via telephone lines. The Micromodem II is completely compatible with the Bell 103-type modems. It can operate in full or half-duplex and has selectable transmission speeds of 110 and 300 bps.

The Micromodem II data communications system consists of three major parts: the Micromodem II printed circuit board, the Microcoupler™, and the Micromodem II diskette. The printed circuit board holds the Micromodem II, ROM (read-only memory) firmware and the serial interface. It plugs directly into the Apple II providing communications capability plus programmable auto dialing and auto answer capabilities.

The Micromodem II ROM firmware supports three operating modes: terminal mode, remote console mode and program control mode. The terminal mode simulates the operation of a dumb terminal allowing the computer to communicate with any other computer equipped with a Bell System 103-type compatible modem. In remote console mode, the Apple II computer can be accessed by a terminal/computer from a distance. With program control mode, the user can program the Micromodem II for more varied applications.

The Hayes Microcoupler is a device which connects the Micromodem II board to the telephone line. It waits for a dial tone, dials numbers, answers the telephone and hangs up when a transmission is over. Since it connects directly to the telephone line, the Microcoupler eliminates the losses and distortions associated with acoustic couplers. The Microcoupler is compatible with all North American standard telephone lines and is FCC approved for direct connection in the U.S. It works with standard dial or Touch-tone† telephone service.

NOTE: The Microcoupler will not work with PBX (private switchboard) systems that do not allow pulse dialing.

™Micromodem II and Microcoupler are trademarks of Hayes Microcomputer Products, Inc.
*Apple II is a registered trademark of Apple Computer, Inc.
†Registered trademark of the Bell System.

The third part of the Micromodem II system is the Micromodem II diskette. The diskette contains the following programs for use with the Micromodem II:

SELF TEST
STORE & FORWARD
PICKUP
AUTO DIAL
DUMBO
TRANSFER
BASIC EXTRACTOR
ALARM
DIAL A HUMAN
ANSWER ON NTH RING MODIFIED
SOURCEON
DJ CONVERTER

Descriptions and applications of these programs are discussed in Chapter 6 of this manual.

There is an optional Hayes DATACOMM software package available which allows Apple Pascal programmers to use the Micromodem II.

CHAPTER 2: INSTALLATION

ASSUMPTIONS

The following conditions are assumed in most examples and discussions in this manual. Exceptions are noted.

- The Micromodem II is installed in slot 3.
- Disk Operating System (DOS) is in effect.
- The Micromodem II is being used with an Apple II having auto-start capabilities.

If the Micromodem II is not installed in slot 3, modify the examples in this manual to reflect the slot in which the Micromodem II is actually located.

Users with machines not having DOS will need to make the following modifications in program statements given in this manual.

DOS		NON-DOS
PRINT D$; "PR#N"	=	PR#N
PRINT D$; "IN#N"	=	IN#N
PRINT D$; "PR#";SLOT	=	PR#SLOT
PRINT D$; "IN#";SLOT	=	IN#SLOT

NOTE: "N" is the number of the slot in which the Micromodem II is installed.

If the Micromodem II is connected to an Apple II without auto-start capabilities, always get to BASIC before proceeding with examples in this manual.

PARTS

The Micromodem II data communications system consists of five parts:

1. Micromodem II printed circuit board
2. Microcoupler
3. Micromodem II diskette
4. Ribbon cable to connect the circuit board to the Microcoupler.
5. Modular telephone cable to connect the Microcoupler to the telephone line.

Take each part out of the box and check to see that the system is complete and undamaged.

INSTALLING THE MICRO-MODEM II PRINTED CIRCUIT BOARD

BE SURE THE COMPUTER IS TURNED OFF BEFORE BEGINNING INSTALLATION.

Remove the cover of the computer. Along the back wall of the case you will find a row of eight printed circuit board connectors. These slots are numbered Ø-7 from left to right with the numbers printed on the computer's main circuit board behind each connector. Although the Micromodem II will work in any slot except slot Ø, slots 2-5 are recommended. Slots 6 and 7 are usually reserved for disk controllers.

To install the Micromodem II printed circuit board, turn off the computer, choose a slot, and insert the gold-plated fingers of the circuit board into the connector. Rock the board back and forth slightly until it is firmly seated in the connector.

INSTALLING THE MICRO-COUPLER

Connect the Micromodem II circuit board to the Microcoupler with the ribbon cable. On the front end of the circuit board is a connector with seven small pointed prongs. The connectors on the ribbon cable have seven corresponding holes. To ensure proper connection, the eighth prong on the board has been cut off and the corresponding hole on the cable connector has been filled. Plug one end of the cable into the circuit board connector so that the cable's connector is flush with the board. Thread the ribbon cable through one of the cable slots in the back of the case and replace the cover of the computer. Plug the remaining end of the ribbon cable into the corresponding connector in the Microcoupler in the same way as the cable was connected to the board.

FCC REGIS-TRATION

The Microcoupler portion of the Micromodem II system is registered with the Federal Communications Commission (FCC) which places three restrictions on its use:

- The Microcoupler cannot be connected to a party line or pay telephone.
- The telephone company must be notified that an FCC registered device is being installed.
- Hayes Microcomputer Products must make necessary repairs to the Microcoupler in order to maintain valid FCC registration. (See Appendix C for return for repair procedure.)

NOTIFYING THE TELEPHONE COMPANY

The telephone company will need the following information before the Microcoupler is installed:

- The telephone number to which it is to be connected.
- The FCC registration number:
  BI 986H-62226-PC-E
- The ringer equivalence: Ø.4B
- The Microcoupler needs to be connected to one of the following modular jacks:
  USOC-RJ11W or USOC-RJ11C

If you plan to connect the Microcoupler to several different telephone lines, you may give the telephone company a list of the numbers to avoid having to notify them each time the Microcoupler is moved. The telephone company must also be notified when the Microcoupler is permanently removed from a telephone line. If the local telephone company has questions, refer them to Hayes Microcomputer Products, Norcross, Georgia.

CONNECTING THE MICROCOUPLER TO THE TELEPHONE LINE

After the telephone company has been notified, plug one end of the modular telephone cable into the Microcoupler and plug the remaining end of the telephone cable into a USOC-RJ11W or USOC-RJ11C telephone wall jack as you would a modular telephone. Detailed explanations of this procedure are usually found in the information pages in the front of the telephone book.

The Micromodem II data communications system is now completely installed and the SELF TEST program on the Micromodem II diskette can be utilized. The SELF TEST and its uses are described in Chapter 6.

PRECAUTIONS

Disconnect the Microcoupler if you experience trouble with the telephone line after installation. If the Microcoupler is responsible for the trouble on the line, do not use it until it has been repaired by Hayes Microcomputer Products. The telephone company is not responsible for disturbances caused by non-telephone company equipment.

The telephone company may change its equipment and is under no obligation to ensure that the new equipment will be compatible with the Hayes Microcoupler. However, it is unlikely that any changes in telephone equipment will make the Micromodem II unusable since the interface between the Microcoupler and the telephone line is functionally identical to the interface between the standard dial phone and the telephone line.

INSTALLING DOS 3.3 ON THE MICROMODEM II DISKETTE

DOS version 3.3 is a computer program necessary to operate the Micromodem II diskette with an Apple II computer. Since DOS is owned by Apple Computer, Inc., the Micromodem II diskette is shipped without the DOS 3.3 program. Follow the appropriate procedure below to transfer your copy of DOS 3.3 to the Micromodem II diskette.

INSTALLING DOS 3.3 USING ONE DISK DRIVE

The procedure below transfers the user's copy of DOS 3.3 to the Micromodem II diskette using one (1) disk drive.

1. Be sure the Apple II computer is turned off.
   Place the Apple diskette labelled "DOS 3.3-SYSTEM MASTER" in Drive 1.
   NOTE: The "DOS 3.3-SYSTEM MASTER" diskette should have come with the Disk II system. Be sure the diskette says DOS 3.3 on the label. Do not use a DOS version other than DOS 3.3 during this procedure.
2. Turn on the Apple II.
   When the disk drive stops, type: BRUN MASTER CREATE.
   Press the RETURN key.
3. Type the response HAYES MENU to the first question appearing on the screen.
   Press the RETURN key.
4. Remove the DOS 3.3-SYSTEM MASTER diskette from Drive 1.
5. Place the Micromodem II diskette in Drive 1.
6. Press the RETURN key.
   When the disk drive stops, the diskette will have DOS 3.3 on it.
   NOTE: If you have trouble with this procedure, begin again from Step 1. If you still have problems please contact the dealer in your area for assistance.
7. Press the ESC key.
8. If you have an Apple II Plus computer or an APPLESOFT ROM card in slot Ø, press the RETURN key to run the Micromodem II diskette. If you do **not** have one of the above, continue with steps 9-26.
9. Place the DOS 3.3-SYSTEM MASTER diskette in Drive 1.
   Press the RETURN key.
10. When the disk drive stops, type: BRUN FID.
    Press the RETURN key.
11. When the menu appears on the screen, choose item number 1.
    Press the RETURN key.
12. When the program asks for SOURCE SLOT?, type number of the slot in which your disk controller is located (usually it is in slot 6).
    Press the RETURN key.
    For the question DRIVE?, type the number 1.
    Press the RETURN key.

13. Answer the question: DESTINATION SLOT?, with the same number as you entered for SOURCE SLOT?
    Press the RETURN key.
14. For the question DRIVE?, answer 1.
    Press the RETURN key.
15. When the question FILENAME? appears, type: APPLESOFT.
    Press the RETURN key.
16. Press any key to begin.
    Follow the instructions that appear on the screen.
    NOTE: SOURCE DISK refers to the DOS 3.3-SYSTEM MASTER, DESTINATION DISK refers to the MICROMODEM II diskette.
    Swap diskettes when instructions appear on the screen.
17. Follow the instructions on the screen to return to the menu.
    Choose item 1.
    Press the RETURN key.
18. Answer FILENAME? with FPBASIC.
    Press the RETURN key.
19. Place the DOS 3.3-SYSTEM MASTER diskette in Drive 1.
    Press any key.
    Swap diskettes when instructions appear on the screen.
20. Follow the instructions on the screen to return to the menu.
    Choose item 9.
    Press the RETURN key.
21. Type LOAD APPLESOFT.
    Press the RETURN key.
22. *NOTE: The ^D in the next procedure is a Control-D. To achieve this hold down the key marked CTRL and press the D key. THIS WILL NOT APPEAR ON YOUR SCREEN.*

    Type: 310 PRINT "^DRUN HAYES MENU".
    Press the RETURN key.
23. When the prompt (]) appears, type: UNLOCK APPLESOFT.
    Press the RETURN key.
24. When the prompt (]) appears, type: SAVE APPLESOFT.
    Press the RETURN key.
25. When the prompt (]) appears, type: LOCK APPLESOFT.
    Press the RETURN key.
26. When the prompt (]) appears, type: PR#SLOT (SLOT is the number of the slot in which your disk controller is located.
    Press the RETURN key, to re-boot system and run the MICROMODEM II diskette.

INSTALLING DOS 3.3 USING TWO DISK DRIVES

The procedure below transfers the user's copy of DOS 3.3 to the Micromodem II diskettte using two (2) disk drives.

1. Be sure the Apple II computer is turned off.
   Place the Apple diskette labelled "DOS 3.3-SYSTEM MASTER" in Drive 1.
   NOTE: The "DOS 3.3-SYSTEM MASTER" diskettte should have come with the Disk II System. Be sure the diskette says DOS 3.3 on the label. Do not use a DOS version other than DOS 3.3 during this procedure.
2. Turn on the Apple II.
   When the disk drive stops, type: BRUN MASTER CREATE.
   Press the RETURN key.
3. Type the response HAYES MENU to the first question appearing on the screen.
   Press the RETURN key.
4. Remove the DOS 3.3-SYSTEM MASTER diskettte from Drive 1.
5. Place the Micromodem II diskette in Drive 1.
6. Press the RETURN key.
   When the disk drive stops, the diskette will have DOS 3.3 on it.
   NOTE: If you have trouble with this procedure, begin again from Step 1. If you still have problems please contact the dealer in your area for assistance.
7. Press the ESC key.
8. If you have an Apple II Plus computer or an APPLESOFT ROM card in slot Ø, press the RETURN key to run the Micromodem II diskette. If you do **not** have one of the above, continue with steps 9-26.
9. Place the DOS 3.3-SYSTEM MASTER diskette in Drive 1.
   Press the RETURN key.
10. When disk drive stops, type: BRUN FID.
    Press the RETURN key.
11. When the menu appears on the screen, choose item number 1.
    Press the RETURN key.
12. When the program asks for SOURCE SLOT?, type the number of the slot in which your disk controller is located (usually it is in slot 6).
    Press the RETURN key.
    For the question DRIVE?, type the number 1.
    Press the RETURN key.

13. Answer the question: DESTINATION SLOT?, with the same number as you entered for SOURCE SLOT?
    Press the RETURN key.
14. For the question DRIVE?, answer 2.
    Press the RETURN key.
15. When the question FILENAME? appears, type: APPLESOFT.
    Press the RETURN key.
16. Place the Micromodem II diskette in Drive 2.
    Press the RETURN key.
17. Follow the instructions on the screen to return to the menu.
    Choose item 1.
    Press the RETURN key.
18. Answer FILENAME? with FPBASIC.
    Press the RETURN key.
19. Press any key to begin. (You do not need to press RETURN.)
20. Follow the instructions on the screen to return to the menu.
    Choose item 9.
    Press the RETURN key.
21. Remove the DOS 3.3-SYSTEM MASTER disk from Drive 1.
    Take the Micromodem II diskette out of Drive 2.
    Place the Micromodem II diskette in Drive 1.
    Type: LOAD APPLESOFT, D1.
    Press the RETURN key.
22. *NOTE: The ^D in the next procedure is a Control-D. To achieve this hold down the key marked CTRL and press the D key. THIS WILL NOT APPEAR ON YOUR SCREEN.*

    Type: 310 PRINT "^DRUN HAYES MENU".
    Press the RETURN key.
23. When the prompt (]) appears, type: UNLOCK APPLESOFT.
    Press the RETURN key.
24. When the prompt (]) appears, type: SAVE APPLESOFT.
    Press the RETURN key.
25. When the prompt (]) appears, type: LOCK APPLESOFT.
    Press the RETURN key.
26. When the prompt (]) appears type PR#SLOT (SLOT is the number of the slot in which your disk controller is located).
    Press the RETURN key to re-boot system and run the MICROMODEM II diskette.

CHAPTER 3: TERMINAL MODE

The ROM on the Micromodem II has a completely self-contained terminal program which simulates the functions of a dumb terminal allowing the user to "converse" with a person with another computer/terminal at the other end of the telephone line. When this program is activated, it takes full control of the computer and the Micromodem II and puts them in terminal mode. As long as the terminal program is running, all other programs in the computer are inactive and will be undisturbed until terminal mode is exited.

ENTERING TERMINAL MODE

The terminal program is controlled by code sequences from the keyboard, therefore, the Micromodem II on board firmware must be given control of the keyboard before it can process any keyboard commands. The Apple II Monitor and BASIC interpreters provide a method for transferring keyboard control to any peripheral slot.

To transfer keyboard control to the Micromodem II firmware in slot 3, from Apple II BASIC, depress:

If a printer is being used in terminal mode a PR# statement indicating the printer slot should be in effect. A PR#3 statement in effect when entering terminal mode will cause characters to be doubled at the other end of the line.

To use the "attention" code of the Micromodem II, depress:

In the terminal program, all commands from the keyboard begin with CTRL-A. When the Micromodem II firmware recognizes CTRL-A it gives the following prompt.

DISPLAY:

MICROMODEM II:?

NOTE: Entering 2 CTRL-As will send 1 CTRL-A to the other end.

The following 8 command code sequences control the operations of the terminal program and the modem.

To start the terminal program in half-duplex mode, depress:

To start the terminal program in full-duplex mode, depress:

DISPLAY:

MICROMODEM II:BEGIN TERM

The Apple II and Micromodem II are now in terminal mode.

NOTE: Full-duplex must be used when calling another Apple II equipped with a Micromodem II and operating in remote console mode. When communicating with another Apple II also in terminal mode, both computers should operate half-duplex.

DIALING IN TERMINAL MODE

To instruct the Micromodem II to pick up the telephone and start dialing, depress:

DISPLAY:

MICROMODEM II:DIALING:

This command is accepted in terminal mode when the telephone is hung up. The Micromodem II then picks up the telephone to start dialing. The flashing cursor disappears from the screen for two seconds while the Micromodem II waits for a dial tone. When it reappears, start entering the digits of the telephone number. Each digit is dialed as it is entered and while it is being dialed, the cursor again disappears. It reappears when the Micromodem II is ready for another digit. Since the keyboard buffers one character, the next digit may be entered as soon as the previous one appears on the screen.

The Micromodem II accepts and dials digits Ø-9. An asterisk (*) instructs the Micromodem II to delay 2 seconds. This feature is useful when dialing through a PBX where it is necessary to wait for a second dial tone. When all the digits have been dialed, enter RETURN and the Micromodem II will begin listening for another modem's carrier. If the Micromodem II does not detect a carrier within 30 seconds, it will hang up the telephone.

DISPLAY:

MICROMODEM II:NO CARR.

MICROMODEM II:HUNG UP

To try again, repeat the procedure beginning with:

Do not pick up an extension telephone to listen to the Micromodem II dialing because the second telephone will prevent the dial pulses from being recognized by the telephone exchange. When the Micromodem II does detect a carrier, a connection is established.

DISPLAY:

MICROMODEM II:CONN.

When a connection is established with the computer at the other end of the line, data may be exchanged. Data entered is transmitted to the computer and data sent is displayed on the screen.

Anytime after the dialing is **completed**, you can pick up another extension and listen to the line. Listening is recommended if you have been unsuccessful dialing. It is possible that the line is busy; you have reached a wrong number; or the other computer is not online and a person has answered the telephone.

Once the connection is established and the modems have started their tones, hang up immediately as any extraneous noise will interfere with the modem signals, and the second telephone on the line will reduce the strength of the signal increasing the probability of errors.

HANGING UP THE TELEPHONE IN TERMINAL MODE

To instruct the Micromodem II to hang up the telephone, depress:

DISPLAY:

MICROMODEM II:HUNG UP

If at any time during a call the carrier tone disappears for half a second or more, the Micromodem II will hang up the telephone.

DISPLAY:

MICROMODEM II:NO CARR.

MICROMODEM II:HUNG UP

This feature frees the line in the event the remote modem hangs up or the line goes down.

SETTING BAUD RATE IN TERMINAL MODE

To set the baud rate to 110 with 2 stop bits, depress:

To set the baud rate to 300 with one stop bit, depress:

A 300 baud rate is preferable in most cases because it is 3 times faster than 110 baud. However, if difficulty develops due to a poor telephone connection, the slower baud rate may be more effective because it is slightly less susceptible to errors. These commands do not produce a display on the screen.

STOPPING UNWANTED OUTPUT IN TERMINAL MODE

To simulate the effect of holding down the break key found on many terminals, depress:

On some time-sharing systems, this feature is used to stop unwanted output (somewhat like the CTRL-C command sequence for an Apple BASIC program.) This command does not produce a display on the screen. Depress any key to end the break.

USING A PRINTER IN TERMINAL MODE

If a printer is connected to the Apple II, the output in terminal mode may under certain conditions be sent to the printer as well as the display screen or to the printer alone. Follow normal procedures for selecting the printer for output before entering terminal mode.

If the printer is connected via a high-speed serial interface card the baud rate of the printer must be equal to or greater than the baud rate selected for the modem. If the baud rates are equal, it is essential that the character format also be the same.

The high-speed serial card defaults to a character format with 2 stop bits, and at 300 baud the Micromodem II defaults to 1 stop bit. The additional stop bit will make the printer about 10% slower than the modem and will cause about 1 character out of 10 to be lost.

NOTE: The Apple Serial Interface Card (part # 600-0029) will malfunction if any other interface card with more than 256 bytes of PROM is installed in the next higher slot. For example, if the Serial Interface Card is in slot 2 and the Micromodem II is in slot 3, the Serial Interface Card will not work. The Serial Interface Card should be installed on the left side of an empty slot or one containing a card with 256 or fewer bytes of PROM memory. Replacing the PROM on the Serial Interface Card with Apple's "P7-04" PROM will also alleviate the problem.

If the printer is connected via a parallel printer interface card, it will probably buffer an entire line at a time and then print the whole line after it receives a RETURN. Since the process is fairly fast, it is unlikely that any characters will be missed when entering the buffer. However, some characters may be missed when the RETURN is received and the line is printed. This loss of characters is a result of the time (a second or more) needed to print the whole line. Most time-sharing computers send several RUB OUT or NULL characters at the end of a line to allow printers to return their printheads to the left margin, but this delay is usually only a few tenths of a second at most.

When dialing another Apple II equipped with a Micromodem II, it is possible to select a delay after a RETURN of up to 2.55 seconds. This delay is adequate for most printers and printing terminals.

EXITING TERMINAL MODE

To instruct the Micromodem II to exit terminal mode, depress:

DISPLAY:

MICROMODEM II:END TERM

This command re-establishes communications between the keyboard and programs (such as BASIC) in the Apple II. Any program in effect before entering terminal mode will be in EXACTLY the same place after leaving terminal mode. For example, a BASIC program left waiting for data to be entered, will still be waiting for the data.

EXAMPLE SESSION 1 (TERMINAL MODE)

Assuming the Micromodem II is in slot 3 of an auto-start machine, the following command sequence demonstrates the procedure for calling another computer. In this example the Micromodem II is calling the XYZ Time Sharing Corporation's Computer System.

To connect the keyboard input of the Apple II to the Micromodem II, depress:

A BASIC prompt should appear on the screen.

DISPLAY:

\>

or

]

Most time-sharing machines operate in full duplex as does the XYZ Computer. To select full-duplex, depress:

DISPLAY:

MICROMODEM II:?

MICROMODEM II:BEGIN TERM

NOTE: If you do not know whether the system you are calling is full or half-duplex, assume half-duplex to begin with. If it is in full-duplex each character you send to the time-sharing machine will appear twice on your screen. You can enter the following command sequence and change the mode to full-duplex:
duplex:

DISPLAY:

MICROMODEM II:?

None of the commands are sent to the time-sharing machine.

The XYZ Computer is set up to run at 300 baud. To select 300 baud, depress:

DISPLAY:

MICROMODEM II:?

NOTE: Since the Micromodem II defaults to 300 baud, this step could have been omitted.

To tell the Micromodem II to prepare to dial a telephone number, depress:

DISPLAY:

MICROMODEM II:?

MICROMODEM II:DIALING:

At this point the Micromodem II is on the telephone line. It waits 2 seconds for a dial tone, and then displays a cursor after the word DIALING:. Enter the telephone number one digit at a time. The cursor disappears while the Micromodem II is dialing and reappears when the modem is ready to dial another digit. To make the display more readable, you may type parentheses and dashes at the appropriate places. These characters do not affect the Micromodem II.

To indicate that all the digits of the telephone number have been entered, depress:

DISPLAY:

MICROMODEM II:DIALING:1(404) 345-6789
MICROMODEM II:AWAIT CARR.

The Micromodem II will wait up to 30 seconds for a modem to answer the telephone and send its carrier. XYZ's computer answers the telephone, send its carrier and connection is established.

DISPLAY:

MICROMODEM II:CONN.
HELLO THIS IS THE XYZ CORP TIME-SHARING SYSTEM.

PLEASE SIGN ON:

Sign on and work with the time-sharing machine. To end communication with XYZ, enter the following command sequence and the Micromodem II will hang up the telephone.

DISPLAY:

MICROMODEM II:?

MICROMODEM II:HUNG UP

NOTE: Pressing the RESET *key on the Apple II would also have disconnected the Micromodem II from the Apple II keyboard and hung up the telephone.*

CHAPTER 4: REMOTE CONSOLE MODE

The Micromodem II in remote console mode allows the Apple II to be accessed from a remote location using a dumb terminal or another computer.

ENTERING REMOTE CONSOLE MODE

Assuming the Micromodem II is in slot 3 of an auto-start machine, the following procedure will prepare the remote Micromodem II to answer the telephone. Enter the following commands from the remote Apple II.

From Apple II BASIC, depress:

These commands cause input from the Apple II keyboard to be routed through the firmware on the Micromodem II card. When any program in the Apple requests input from the keyboard, the Micromodem II firmware checks to see if the telephone is ringing. The telephone rings.

DISPLAY: (Remote Apple II screen)

MICROMODEM II:RING

At the end of a ring, the Micromodem II answers the telephone.

DISPLAY: (Remote Apple II screen)

MICROMODEM II:AWAIT CARR.

The Micromodem II then turns on its carrier and waits up to 30 seconds for the modem on the other end of the line to respond with its carrier. When the carrier is detected, connection is established.

DISPLAY: (Remote Apple II screen)

MICROMODEM II:CONN.

After it connects, the Micromodem II sends a RETURN to the remote Apple II input. The RETURN will cause the Apple II to send its BASIC prompt (› or]).

The Apple II is now under remote control. Almost anything that can be done from the Apple's own keyboard can be done from the remote terminal. If a person were sitting next to the Apple II he could see everything that was occurring because all output to the modem also appears on the screen. The remote keyboard is also active and anything entered from it will be accepted as input to the Apple II.

SPECIAL CONTROL CHARACTERS

There are six special control characters which are accepted ONLY FROM A REMOTE TERMINAL. These control characters may be disabled by setting the code transparency bit (TRAN) in the FLAG WORD. (See Chapter 7)

PAUSING DURING OUTPUT IN REMOTE CONSOLE MODE

To instruct the Micromodem II to temporarily stop sending output, depress:

This command is useful when data is scrolling by too fast for the user to read or the buffer of the receiving computer or printer is filled. Pressing any key other than CTRL-S will re-start output. Sending a second CTRL-S will cause one more character to be sent.

RESETTING THE REMOTE APPLE II

To reset the remote computer without hanging up the modem, depress:

CTRL-Y has almost the same effect as the RESET button on the Apple II keyboard. However, CTRL-Y does not produce a hardware reset pulse (which would reset the Micromodem II and hang up the phone) nor does it return the Apple II input and output to the keyboard and display (which would disconnect the Micromodem II and the remote terminal).

GOING FROM REMOTE CONSOLE MODE TO TERMINAL MODE

To enable the remote Apple II keyboard and enter terminal mode, depress:

CTRL-T is useful when the user wants to communicate back and forth with a person sitting at the remote Apple II. All data and programs in the remote Apple remain undisturbed until CTRL-R is typed from the local terminal or the person at the Apple II types CTRL-A CTRL-X to exit terminal mode. CTRL-T in remote console mode will cause characters sent from the remote Apple II to be doubled at the local terminal if PR#3 is in effect. Before going to terminal mode use ONE of the following command sequences:

AT THE LOCAL TERMINAL:
PR#0
Switch to half-duplex
CTRL-T

OR

AT THE REMOTE APPLE II:
PR#0
CTRL-A CTRL-H

NOTE: In remote console mode, CTRL-R and CRTL-T commands are used when another person is on the other end of the line.

RETURNING TO REMOTE CONSOLE MODE FROM TERMINAL MODE

To return the remote Apple II to remote console mode, depress:

CTRL-R is used to reverse a CTRL-T command. CTRL-R allows the local user to take control of a remote Apple II computer and use all of its facilities.

USING A PRINTER IN REMOTE CONSOLE MODE

To prepare the Micromodem II to communicate with a printing terminal, depress:

CTRL-N does several things. First, it enables the insertion of a line feed after RETURNs. Without this, most printing terminals (and many CRT terminals) will print over and over on the same line. It also enables a short delay after the line feed to allow time for the physical movement of the print head. The delay is usually set to 30 msec. The delay length may be changed by setting memory location CRDLY. (See Chapter 7.) CTRL-N clears the Apple II screen and disables the remote display, effectively disabling the Apple II internal line-folding algorithm and enabling the user to use the full width of the terminal. CTRL-N also disables the remote keyboard except for CTRL-A sequences.

HANGING UP THE TELE-PHONE IN REMOTE CONSOLE MODE

If the remote Apple II is not in any looping operation or is waiting for input and the local terminal user hangs up the telephone, the Micromodem II will detect the loss of carrier and also hang up.

To instruct the Micromodem II to hang up the telephone, depress:

CTRL-Z instructs the Micromodem II to hang up the telephone. All programs and data in the Apple II will remain undisturbed. If you re-dial, the Micromodem II will answer the telephone, but you may not be in the same place as before CTRL-Z was entered.

CURSOR MOVEMENT IN REMOTE CONSOLE MODE

Several of the cursor movement commands from the Apple II keyboard may be used for editing from a remote terminal.

The Apple II keyboard left-arrow (ASCII BS, CTRL-H) character causes the Apple II to move its cursor left one position on the screen and discards the last character from its input buffer. This command is also valid from a remote terminal but the terminal screen may not display the backspace because the cursor movement is a function of the remote Apple II display. The Apple II firmware does echo the backspace, and many CRT terminals recognize this standard ASCII character as a backspace.

The Apple II keyboard right-arrow (ASCII NAK, CTRL-U) character causes the Apple II to move its cursor right one position and take the character under the cursor on the screen as its input. If the Apple II local display is enabled, it will pick up the next character from the screen.

The four escape sequences which cause cursor movement on the Apple II are not echoed by the Apple II firmware making it impossible for the local terminal to move its cursor in the same way.

EXAMPLE SESSION 2 (REMOTE CONSOLE)

To prepare the Apple II to function as a remote console, depress:

At the local terminal dial the Apple II telephone number and the Micromodem II will answer the phone.

DISPLAY:

〉

or

]

These prompts tell the user the Apple II is ready to run BASIC. When use of the Apple II as a remote console is completed, save the program or data and hang up.

To disconnect the Micromodem II and return the Apple II to BASIC, depress:

DISPLAY:

〉

or

]

CHAPTER 5: ELEMENTARY PROGRAMMING

The Micromodem II and its built-in firmware are designed to be used easily with BASIC programs in the Apple II. Most of the commonly needed functions can be performed with the usual BASIC INPUT, PRINT, IN# and PR# statements. Dialing and hanging up the telephone require the use of control characters which can be easily included in strings in a BASIC program.

The five operations that BASIC programs will need to include in order to use the Micromodem II are:

- Dial the telephone
- Hang up the telephone
- Monitor the telephone and answer it when it rings
- Transmit data via the Micromodem II
- Receive data via the Micromodem II

Most of the program fragments in this section contain identical statements for Integer BASIC and APPLESOFT. Differences are noted where applicable. The variable D$ contains the CTRL-D for DOS commands operating in a BASIC program.

NOTE: For more advanced functions such as non-standard data formats and code transparent operation, see Chapter 7.

DIALING THE TELEPHONE

To dial the telephone, select the Micromodem II for output, send it a CTRL-Q followed by a telephone number in a string variable followed by a RETURN. Assuming the Micromodem II is in slot 3, the following fragment within a complete program will dial a long distance CBBS (Computer Bulletin Board System):

Integer BASIC

```
100 Q$ = " ": REM " "CONTAIN A CTRL-Q
200 PRINT D$; "PR#3":PRINT Q$;"1(404) 394-4220"
```

APPLESOFT

```
100 Q$ = CHR$(17)
200 PRINT D$; "PR#3":PRINT Q$;"1(404) 394-4220"
```

Statement 200 selects the Micromodem II for output (PR#3), then it sends a CTRL-Q followed by the telephone number. The RETURN is automatically supplied by BASIC as with any PRINT statement unless the statement ends with a semicolon (;).

When statement 200 is executed:

DISPLAY:

MICROMODEM II:DIALING:1(404)394-4220
MICROMODEM II:AWAITING CARR.

After the Micromodem II dials the telephone, it will wait up to 30 seconds for a carrier. The rest of the program will not be executed until the 30 seconds has elapsed or the Micromodem II has detected a carrier. If no carrier is detected:

DISPLAY:

MICROMODEM II:NO CARR.

MICROMODEM II:HUNG UP

If a carrier is detected:

DISPLAY:

MICROMODEM II:CONN.

Since it is possible for the dialing to be unsuccessful, the program should verify that a carrier was detected. The following fragment within a complete program will type "GOT IT" if a carrier is detected and "NOT HOME" if a carrier is not detected. Statements in this program fragment are the same for Integer BASIC and APPLESOFT.

```
400 IF PEEK(1656+3)>127 THEN 700
500 PRINT "NOT HOME"
600 END
700 PRINT "GOT IT"
800 REM: CONTINUE WITH PROGRAM
```

Memory location 1656+(slot number) contains the modem control word. The most significant bit of this word (which has a binary weight of 128) controls the telephone switch hook. For more detail on memory locations see Chapter 7.

An asterisk character (*) in a telephone number causes the Micromodem II to delay for 2 seconds. This delay is useful when the Micromodem is connected to a PBX requiring the caller to dial an extra number to get an outside line. The delay allows time for the outside dial tone.

If the last character of the telephone number is a CTRL-J (LINE FEED) character, the Micromodem II will omit its usual sequence of listening for a carrier tone and return immediately to the program. CTRL-J is effective when using the computer to dial the phone for some purpose other than establishing communications with another computer. With this feature, the Apple II can be programmed to operate as a repetitive dialer.

HANGING UP THE TELEPHONE

The BASIC program can instruct the Micromodem II to hang up the telephone by sending a CTRL-Z to the modem. The following fragment within a complete program will hang up the telephone.

Integer BASIC

```
1000 Z$ =" ": REM " " CONTAIN CTRL-Z
1100 PRINT D$;"PR#3":PRINT Z$
```

APPLESOFT

```
1000 Z$=CHR$(26)
1100 PRINT D$;"PR#3":PRINT Z$
```

As in the dialing fragment, the empty-looking quotes contain a non-printing control character. In this case it is a CTRL-Z. When statement 1100 is executed:

DISPLAY:

MICROMODEM II:HUNG UP

ANSWERING THE TELEPHONE

The following fragment within a complete program will wait for the telephone to ring, answer the telephone, wait for a carrier, and transmit a short message identifying itself. The statements in this program fragment are the same for Integer BASIC and APPLESOFT.

```
2000 PRINT D$;"IN#3":INPUT I$
2100 PRINT D$;"PR#3":PRINT "HELLO, THIS IS A SAMPLE
     PROGRAM"
2200 REM:ADDITIONAL PROGRAM STATEMENTS
```

There is no outward sign of activity when statement 2000 is executed until the telephone rings. When the telephone rings, the Micromodem II answers it.

DISPLAY:

MICROMODEM II:RING

MICROMODEM II:AWAIT CARR.

The Micromodem II will wait up to 30 seconds for a carrier and if it does not detect one:

DISPLAY:

MICROMODEM II:NO CARR.

MICROMODEM II:HUNG UP

If it does detect a carrier:

DISPLAY:

MICROMODEM II:CONN.

Only when a carrier is detected does the Micromodem II send a RETURN to the Apple II input. This will satisfy the BASIC INPUT statement at line 20000 and allow the execution of the BASIC program to continue at line 21000. The BASIC program will then transmit its message and stop at line 22000 or continue with the program.

TRANSMITTING AND RECEIVING DATA VIA THE MICROMODEM II

Once the Micromodem II has detected a carrier and established a connection, the BASIC program can transmit data by selecting the Micromodem II for output (PRINT D$;"PR#";SLOT) and send data using PRINT statements.

To receive data from a remote device via the Micromodem II, select the Micromodem II for input (PRINT D$;"IN#";SLOT) and perform an INPUT statement.

It is very important that IN#3 and PR#3 statements NOT be in effect at the same time when transmitting or receiving data except when the Apple II is in remote console mode.

CHAPTER 6: MICROMODEM II DISKETTE PROGRAMS

The Micromodem II data communications system includes a diskette containing various BASIC programs. The following is a list with brief descriptions of the programs contained on the diskette.

SELF TEST	Verifies the Micromodem II circuit board is functioning properly. The SELF TEST program does not test the Microcoupler.
STORE & FORWARD	Stores a message to be transmitted at a later time.
PICKUP	Picks up the telephone in answer mode.
AUTO DIAL	Automatically dials modem numbers from a menu.
DUMBO	Contains a terminal program written in BASIC.
TRANSFER	Transfers DOS text files from Apple II to Apple II.
BASIC EXTRACTOR	Extracts a BASIC program from another system.
ALARM	Functions as a computerized wake up call.
DIAL A HUMAN	Automatically dials numbers from a menu for voice communication.
ANSWER ON NTH RING MODIFIED	Answers the telephone after a preset number of rings.
SOURCEON	Dials up and logs on THE SOURCE.
DJ CONVERTER	Allows the Hayes Micromodem II to be used with the Apple Dow Jones Portfolio Evaluator.

SELF TEST

Application

The Micromodem II hardware has a built in self test capability. This feature is controlled by a bit in the MODEM control register. When this feature is selected, the modem transmitter and receiver both operate on the same frequency band so that the receiver can receive the data sent by the transmitter. The Micromodem II hardware is designed so that when it is disconnected from the Microcoupler, there is a calibrated "leak" from the output to the input. This leak simulates the losses which would be encountered on a typical telephone connection. Using this capability, a program can test virtually all of the hardware on the Micromodem II circuit board without being connected to a telephone line.

The SELF TEST program on the Micromodem II diskette is written in APPLESOFT BASIC, and will run in any Apple II configuration. The program utilizes a special entry point in the Micromodem II firmware which allows it to avoid a "false read" problem due to certain peculiarities of the 65Ø2 microprocessor.

Program Description

The program tests the Micromodem II hardware by setting it in each of the four possible combinations of mode and baud rate. It sends all 128 valid ASCII characters through the modem and verifies that all are received correctly.

The SELF TEST program assigns names to the various memory addresses used by the test routines. The test subroutine starts by clearing the ACIA register and setting it in normal mode. The number of characters sent and received is set to Ø.

A loop is formed in which the Receiver Register Full and Transmitter Register Empty bits are checked in the ACIA status register. When a character is present in the receiver, it is read and checked for accuracy. When the transmitter register is empty, the program sends the next character unless all 128 characters have already been sent.

The program turns off the modem carrier and resets the ACIA register. It waits until the carrier detect circuit responds with a built in delay of Ø.5 seconds.

The operator receives a report indicating the mode tested and the error count of the individual test. The program then turns on the modem in the next mode to be tested.

The number of errors in each test is added to the total errors and the total is reported upon completion of the SELF TEST.

STORE & FORWARD

Application

This program receives a short message and stores it in its memory for a period of time before forwarding it to another computer. STORE & FORWARD is useful if the Apple II is connected to a WATS line or if a message is to be held until late evening when the telephone rates are lowest.

Program Description

The STORE & FORWARD program selects the Micromodem II for output and verifies that the telephone is hung up. Then it ensures that anyone connecting to the Apple II cannot break out using CTRL-Y.

When the telephone rings, STORE & FORWARD answers and sends a short description of itself. Then it asks for and verifies a secret password. STORE & FORWARD requires a password because it has the capability of making costly long distance telephone calls. If the password is not correct after three tries, the Micromodem II hangs up the telephone.

Once the correct password has been entered, the program requests and inputs the telephone number to which the message is to be forwarded. After explaining its message size limitations, STORE & FORWARD asks how long to wait before forwarding the message and obtains a time. Then the program gives the caller a chance to verify all his input and start over if it is incorrect.

When it has correct input, the STORE & FORWARD program tells the caller good-by and hangs up the telephone. After waiting the specified length of time it dials the number supplied by the user. If it fails to detect a carrier, it hangs up, waits five minutes and tries again. The program does this up to 3 times before giving up and restarting itself.

After a connection is established, STORE & FORWARD waits a few seconds for the machine it has called to finish identifying itself before sending the message stored. Once the message has been sent the program hangs up and goes back to waiting for the telephone to ring.

NOTE: STORE & FORWARD defines the slot in which the Micromodem II is installed. If your Micromodem II is not in slot 3 you will need to change the line containing the slot number.

PICKUP

Application

PICKUP allows the Micromodem II to pick up the telephone and begin communicating without the telephone actually ringing. For example, the Micromodem II could pick up the telephone during a voice conversation without the persons involved having to hang up and dial again. To activate, depress:

RUN PICKUP

Program Description

The PICKUP program first defines the slot in which the Micromodem II is installed. Then it selects the Micromodem II for output and initializes it by sending the modem a RETURN. PICKUP selects standard character format: 8 data bits, no parity, 1 stop bit.

The program then takes the telephone off the hook, turns on the Micromodem II in answer mode and waits for a carrier. When a carrier is detected, the program puts the computer in terminal mode.

AUTO DIAL

Application

The AUTO DIAL program acts as a computerized telephone listing and automatic dialing service for modems and computers.

Program Description

The AUTO DIAL program clears the screen and displays the menu on the screen. The first statement tells how many statements follow in the menu. The remaining statements consist of two data strings: a name and a corresponding telephone number. The names and numbers in this menu are for other modems or computers.

AUTO DIAL then asks the user to select one statement by its line number. If a valid line is selected the Micromodem II dials the telephone. If it successfully reaches another modem, the AUTO DIAL program puts the Micromodem II in terminal mode. If unsuccessful, the program asks the user if he would like to try another number. If 0 is requested, the program assumes the user wants to dial manually and puts the Micromodem II directly into terminal mode. A negative line number exits the program.

The telephone numbers on the menu included in this program are frequently Computerized Bulletin Board Systems (CBBSs). They are usually operated by computer hobbyists, and several of them use Hayes Microcomputer Products, Inc., modems.

DUMBO

Application

The dumb terminal program included in the Micromodem II ROM was designed to accommodate most of the requirements for communicating with various systems. DUMBO contains subroutines that allow the user to write his own BASIC terminal program.

Program Description

After asking a few questions to determine options, DUMBO dials the telephone number the user supplies then enters a loop in which it directly interrogates the keyboard and the modem input port and transfers bytes between them, the screen, and the modem output. It checks for error conditions on the modem, and reports them to the user.

It does not check individual characters coming from either the keyboard or the modem with one exception. DUMBO detects CTRL-G (BELL) from the modem, and calls a special short beep routine. The beep the Apple II firmware generates is 100 msec long, which is 3 character times at 300 baud. This means that each time the remote computer sends a BELL, the program will detect several errors when it misses the two characters following the beep.

Most of the initialization is straightforward. Line 10 and 100 set up a simulated CHR$ function, which makes it easier to print the data obtained via PEEK's as characters rather than as numbers.

Line 150 installs a machine-language program that makes short beeps. It loads a smaller constant into the Y register then jumps into the beep-making routine in the Apple II monitor ROM. The machine code is:

```
300: LDY #$23
302: JMP $FBE4
```

It goes into the terminal loop from lines 300-570. Lines 300-340 are the status-checking loop. Lines 350-380 handle characters read from the keyboard; data is displayed on the screen on lines 410-420; input from the modem is handled by lines 440-480; and lines 490-570 handle errors.

During normal operation only the two least significant bits of the STATUS register should be on. Therefore, if line 310 finds a value greater than 3, it knows that there is some error condition. The expression X MOD 2 in line 320 effectively tests X for being odd. If it is odd, the least significant bit is a 1, and there is a character ready in the receiver data register. The program then goes to 500 to handle the character.

Line 330 reads the keyboard port; line 340 tests whether there is a character there; and line 350 releases the character if there is one. Line 360 and 370 transmit the character via the Micromodem II.

Line 410 is the simulated CHR$, and line 420 displays the character on the Apple II screen.

A character is read from the Micromodem II in line 440, and line 450 makes sure that the most significant bit is set. This makes sure that the characters will be recognized correctly regardless of the parity option selected. Line 460 looks for BELL characters. Most characters are displayed by the code at 470, but BELLs are handled specially by calling the custom short beep routine located at $300 = 768 dec.

The first thing the error routine looks for at line 490 is loss of carrier. This error can be singled out by a quick test based on the knowledge that its binary weight is 4, and that the program could only get here if X>3 (line 310). If the carrier has been lost, the X value will be between 4 and 7 (assuming that no other error bits are set). It is possible that other error bits may get set when the carrier is lost, but they will disappear when the error is cleared. The carrier will still be lost.

If the error is not a loss of carrier, the data register is read, which clears the error condition, and gives the user an error message before going back to the terminal loop.

When a loss of carrier is detected, the Micromodem II is selected for output and sent a CTRL-Z. The CTRL-Z hangs up the telephone.

TRANSFER

Application

TRANSFER is an APPLESOFT program designed to communicate with a copy of itself which has been loaded into another Apple II computer. It transfers text files from one Apple II to another.

Program Description

When TRANSFER starts, it puts the user into terminal mode so that he can communicate with the operator of the other Apple II and establish which files are to be sent.

When this decision is made, either one may type CTRL-A CTRL-X to exit terminal mode, followed by a RETURN. Both computers will then ask for a file name, and will open the appropriate disk file. TRANSFER asks whether the user wishes to send (S) or to receive (R) a file. As soon as both users have answered both questions the programs will start transferring the file one line at a

time, with the data being displayed on both screens as it is being sent. The two programs communicate back and forth to make sure that they stay in sync so that no data will be lost.

When the last line has been sent, the sending computer sends a CTRL-C to the receiving computer, which tells it that all the data has been sent. Both machines then close their files and go back into terminal mode.

BASIC EXTRACTOR

Application

The BASIC EXTRACTOR program is designed to extract a BASIC program from a dialup computer and save it in an Apple II text file. Once the BASIC program is in a text file, it can be manipulated with a user's original BASIC program to make simple changes to prepare it to run in APPLESOFT. It can then be submitted to APPLESOFT as an EXEC file. (See Apple D.O.S. Manual.)

The BASIC EXTRACTOR program is useful because most time-sharing systems are not set up to transmit BASIC programs to another computer, and the BASIC interpreters in the Apple II are not designed to accept programs from another computer.

Since the APPLESOFT input editor will allow the user to enter and edit programs which cannot be run in APPLESOFT, all the normal program editing facilities can be used to work on the foreign program while converting it to APPLESOFT. The BASIC EXTRACTOR program extracts and saves programs in chunks. The memory of the Apple II must be large enough to hold the parts of the programs that are extracted before being saved on disk.

Program Description

The BASIC program is extracted by issuing a series of LIST commands. It is recommended that the user have some knowledge of the line numbering of the program to be extracted. Many versions of BASIC have a RENUMBER command which can be used to determine the exact line numbering scheme.

BASIC EXTRACTOR first asks for the low line number, the high line number, and the line increment. It uses the information supplied to issue a series of LIST commands that will each ask for about 10 lines. The program has a buffer that will hold 300 lines in case the line numbering is slightly off.

BASIC EXTRACTOR assumes that the BASIC interpreter with which it is conversing puts out a recognizable sequence of characters when it has completed a LIST command to tell the program that the BASIC interpreter is finished. The lines LISTed are then saved. The program will ask the user for the word or message indicating completion of a command, i.e., OK, READY.

However, if the message is a single character such as], the APPLESOFT prompt, the changes below should be made in the BASIC EXTRACTOR program.

The J counter ensures that the program does not lose the rest of a line that has the single character prompt within the line itself.

The changes for APPLESOFT extraction are:

```
1400 INPUT "CHARACTER INDICATING END OF
     LISTING";OK$
2415 J=0
2480 IF J< 3 AND A$=OK$ THEN 2900
2485 J=J+1
2600 (delete line 2600)
```

ALARM

Application

ALARM operates as a computerized wake up call. It uses a Hayes Micromodem II and a Mountain Computer Apple Clock* to make a telephone call at a specified time and generate a distinctive sound when the telephone is answered by the party on the other end.

Program Description

The ALARM program reads the clock and displays the current time. It gets and checks the specified time and telephone number. ALARM then enters a loop in which the program reads the clock, updates the time of the display, and checks to see if the specified time has arrived.

When the time arrives, the program places the telephone call and generates a distinctive sound by switching the Micromodem II between originate and answer modes. When switching between modes, the program turns the Micromodem II transmitter off while keeping the telephone off hook. After 30 seconds, the program hangs up the telephone.

NOTE: It is necessary for the program to turn the modem transmitter off before changing the mode or it will not operate properly.

DIAL A HUMAN

Application

The DIAL A HUMAN program allows the Micromodem II to dial the telephone for voice communication instead of computer to computer communication. With the DIAL A HUMAN program, the Micromodem II finds and dials a telephone number when a person's name or number is entered. The program also allows the Micromodem II to re-dial the number when the connection is unsuccessful.

*Apple Clock is a trademark of Mountain Computer, Inc.

Program Description

DIAL A HUMAN can be used in two ways. The user can use the program to simply dial a number which the user enters at the time it is to be dialed. Or, the user can create his own "telephone book" by adding names and telephone numbers to the end of the DIAL A HUMAN program. When a name from the list is entered, DIAL A HUMAN will locate and dial the correct telephone number for voice communication.

Line 1000 contains the number of names and corresponding telephone numbers in the DATA statements that follow. Actual names and telephone numbers are listed in DATA statements beginning with 1010 as the following example illustrates:

RESET

```
1000 DATA    4
1010 DATA    DAVID, 765-4321
1020 DATA    HAYES MICROCOMPUTER, 1-(404)-449-8791
1030 DATA    INFORMATION,411
1040 DATA    OPERATOR,0
```

TYPE: UNLOCK DIAL A HUMAN
TYPE: SAVE DIAL A HUMAN
TYPE: RUN

Three commands are used with the DIAL A HUMAN program. When a "?" is entered, the program will display the menu of names and telephone numbers on file. The "/" symbol is used for re-dialing the last number dialed, and entering "END" will exit the program.

ANSWER ON NTH RING MODIFIED

Application

ANSWER ON NTH RING MODIFIED programs the Micromodem II to answer the telephone after a preset number of rings. The program allows a person time to answer the telephone first, or if no one answers, the computer will answer the telephone.

Program Description

ANSWER ON NTH RING MODIFIED waits for the telephone to ring. If it does not ring the specified number of rings, the ring count is reset and the telephone is not answered by the Micromodem II. The rest of ANSWER ON NTH RING MODIFIED is a variation of the PICKUP program described in this chapter. However, unlike PICKUP, this program does not wait indefinitely for a carrier. After a specified time if no carrier is detected, the program hangs up the telephone.

SOURCEON

Application

The SOURCEON program dials up and logs on THE SOURCE time-sharing system. Properly modified, the SOURCEON program could be used to log on to almost any computer.

Program Description

Line 800 needs the user's local access telephone number to Telenet. The variable PFX$ should contain the dial out number for an outside line in an office PBX system. Line 1000 will determine how many times the Micromodem II should keep trying. Line 1200 should contain the user's ID to THE SOURCE.

EXAMPLE:

```
1200 ID$="ID TCC1000 "+CHR$(2)+CHR$(16)+CHR$(23)
```

The CHR$ functions contain each control character for the user password.

Line 1300 contains the user Telenet signon.

EXAMPLE:

```
1300 TRM$="C 24520"
```

The program sends two RETURNs followed by a timeout loop for receiving characters. It checks for the correct prompt character before transmitting data to THE SOURCE. The Apple II is then put in terminal mode for the user to take control.

D J CONVERTER:

D J CONVERTER is a utility program that converts the Apple Dow Jones Portfolio Evaluator software (model #A2D0007, disk part #6000-2512-000) for use with the Hayes Micromodem II. With this conversion, the Micromodem II can be installed in slot 2 or 3.

NOTE: The Micromodem II must be installed in slot 2 for use with the Apple Dow Jones News & Quotes Reporter. It is recommended that the Apple Dow Jones Portfolio Evaluator be modified for slot 2 if both software packages are to be used.

DIRECTIONS FOR USE: LOAD the DJ CONVERTER program into the Apple II memory. Place a copy of the Dow Jones Portfolio Evaluator Disk in the disk drive and enter: RUN. When the program finishes running, the Dow Jones Portfolio Evaluator Disk will be converted for use with the Micromodem II.

CHAPTER 7: MICROMODEM II MEMORY LOCATIONS

The Micromodem II utilizes 11 memory locations in the Apple II. These locations can be set by the experienced programmer for specialized programming or changing defaults.

For convenience refer to the Micromodem II Memory Usage Chart on page 46 of this manual. The Memory Usage Chart summarizes all memory locations used by the Micromodem II which are described in this chapter. Both hexadecimal and decimal values are provided for each location.

DATA, STATUS/CR1, RI 1/CR2

Three of these locations DATA, STATUS/CR1 and RI/CR2 correspond to six hardware registers on the Micromodem II circuit board. Unlike read/write memory locations which have a single read/write cell, each of these Micromodem II memory locations consist of a pair of cells which are related to each other. One cell is read-only and the other is write-only. In some few cases it is possible to read back what has been written to a location but, usually, the data written and the data read back is different.

DATA

$C087+N0=-16249+16*N dec

This pair of memory cells is for modem data. Data which is written to this location is transmitted through the modem, and data which is received from the modem is read from the same location. In SELF TEST mode, data which is written to this location can be read back from it one character time later. Basic programs may read this part but should not write to it. For additional information, see the SELF TEST program in Chapter 6.

STATUS/CR1

$C086+N0=-16250+16*N dec

The second pair of cells reads the status and writes the controls of the Motorola 6850 ACIA chip. This chip performs all the parallel-to-serial conversion on output data and serial-to-parallel conversion on input data. The bits of the status register each have separate meanings and report on various conditions in the chip. The most important reports are Receiver Register Full and Transmitter Register Empty. Other bits report various errors which the ACIA chip is capable of detecting on received data.

The following codes describe the status bits used by the Micromodem II:

bit no.	7	6	5	4	3	2	1	0
name	X	PE	OVRN	FE	RESET	$\overline{CD}$	TRE	RRF
weight	128	64	32	16	8	4	2	1

X - Unused.
PE - Parity error detected.
OVRN - Receiver overrun error.
FE - Framing error.
RESET - Indicates that the Apple II has been RESET since the last time the Micromodem II was initialized.
$\overline{CD}$ - Not carrier detect. When set indicates that no carrier is present, or carrier has been momentarily lost since the last data character was read.
TRE - Transmitter Register Empty. Indicates readiness of ACIA transmitter to accept another character.
RRF - Receiver Register Full. Indicates presence of a valid data character in receiver register.

RI/CR2 The third location couples modem controls and ring detect. Codes output to this location control such functions as taking the telephone off the hook, turning on the modem transmitter, and setting mode and baud rate. A program can determine whether the telephone is ringing by reading this location.

The following information describes these cells individually:

CR2 **$C085+N0 = -16251+16*N dec (write-only)**
Modem control port. This write-only hardware register is updated from memory location "MODEM" (1656+N) each time a character is transmitted by the firmware.

bit no.	7	6	5	4	3	2	1	0
name	OH	X	X	ST	SET	MODE	TXE	BRS
weight	128	64	32	16	8	4	2	1

OH - Off hook. When this bit is set, the Micromodem II "picks up the telephone."
X - Unused.
ST - SELF TEST. When set causes the Micromodem II to enter SELF TEST mode. For details see the description of SELF TEST program in Chapter 6.
SET - Initialization flag. Used in conjunction with RESET status flag to control initialization of Micromodem II. When this bit is set to 0, the RESET status flag goes to 1. When the Micromodem II is called for input or output immediately after an IN# or PR# statement, the RESET flag is checked, and if it is set, the default values are applied. During operation this bit should always be set to a 1.
MODE - Selects originate or answer mode (1 = originate).
TXE - Transmitter enable. Turns on modem transmitter when set.
BRS - Bit rate select. Selects high (300 baud) rate when set.

RI **$C085+N0 = -16251+16*N dec (read-only)**
Ring indicator register. Only the most significant bit (bit 7) of this read-only register is used. When the telephone line is ringing bit 7 is 0. Between rings it is set to a 1.

OUTA **$C002+N00 = -16382+256*N**
Special output call location. A CALL or JSR to this location will cause the byte CHAR ($778 or 1912 dec) to be output to the modem.

CN,N0, CHAR Three locations, CN, N0 and CHAR, are actual memory used by the Micromodem II firmware for temporary storage of variables. These locations are located in an area of the Apple IIs memory that is set aside specifically for temporary variable storage by firmware on peripheral cards and may be shared by other peripherals in the Apple II. The Micromodem II firmware is designed to follow a standard method of sharing these memory locations so that it will not interfere with other peripherals or be interfered with by them.

CN, **$7F8=2040 dec**
Contains the hex value CN any time the Micromodem II has control of the computer.

N0 **$6F8=1784 dec**
Contains the hex value N0 any time the Micromodem II is in control of the computer.

CHAR **$778=1912 dec**
Each character sent or received is temporarily stored at this location. The SELF TEST program described in Chapter 6 also uses this location.

MODEM, ACIA, LOCSE, FLAGS, CRDLY The Micromodem II firmware uses five other memory locations that are reserved specifically for the slot in which the modem is installed. These locations are MODEM, ACIA, LOCSE, FLAGS and CRDLY. The Micromodem II firmware can determine which slot it is in and is therefore able to use these memory locations. In order for an individual program to use this memory to communicate with the Micromodem II firmware, the user must tell the program the slot location of the Micromodem II.

ACIA **$7F8=2040+N dec**
MODEM **$678=1656+N dec**
Two of the memory locations are RAM locations used to store the current contents of the CR1 and CR2 control registers. When modifying the settings of the control registers, it is preferable to write to the memory locations identified above (ACIA & MODEM) and let the Micromodem firmware write to the actual hardware registers. The actual hardware register is updated from these memory locations each time a byte is transmitted.

LOCSE **$6F8+N=1784+N dec**
This memory location contains a value which is exclusive-or'ed with all lower-case letters received through the modem. If this location contains a value of $20, then all lower-case letters will be translated to the corresponding upper-case letter. When the Micromodem II firmware is initialized for the first time (either for input or output), this memory location is initialized to $20. Writing a 0 to this location will disable the lower-to-upper case translation, and lower-case letters will be passed through.

FLAGS **$778+N=1912+N dec**
The fourth location contains flags which turn various firmware options on and off. Bits 0, 3, 5, 6 are used internally by the Micromodem II but the remaining flags are potentially useful to the programmer.

bit no.	7	6	5	4	3	2	1	0
name	DISPO	DLS	X	LFI	TERM	TRAN	KBDE	DLG
weight	128	64	32	16	8	4	2	1

Bit 7 (DISPO) Controls the display of modem output. When 0, this bit causes all output to Micromodem II to be displayed on the Apple II display screen. In terminal mode, bit 7 sets the duplex mode. A 0 indicates half-duplex (local echo) has been selected and a 1 sets the modem to full-duplex. Default is full-duplex.

Bit 6 (DLS) Used internally. Bit 6 indicates that the Micromodem II firmware is preparing to start dialing.

Bit 5 (X) Unused.

Bit 4 (LFI) Selects line feed insertion after RETURN. Normally, the Apple II firmware does not use line feed characters for any purpose, however most printing terminals and many CRTs require line feed characters to advance to the next line. Without line feeds, they print on the same line over and over. This option also enables an adjustable delay after the line feed character. This delay is needed on most printing terminals to allow sufficient time for printhead movement.

Bit 3 (TERM) Used internally. Indicates that the terminal program is running. If this bit is set, the terminal program will begin running the next time the Micromodem II is polled for input.

Bit 2 (TRAN) Selects code transparency. On input, the Micromodem II automatically discards three characters which could interfere with proper operation of the Apple II firmware. These are NULL (all zeros), RUBOUT (all ones) and LINE FEED. Selecting the code transparency option disables this feature. Normally the Micromodem II firmware responds to several control codes sent out by the program or received from the modem. In some cases this could interfere with the user's applications. This control bit allows you to turn these features off so that you can transmit and receive all 128 ASCII characters. CTRL-Q is recognized if the Micromodem II is hung up.

Bit 1 (KBDE) Enables the local keyboard. When this bit is set, the Micromodem II will accept inputs from the Apple II keyboard or from a remote keyboard connected via the telephone line. When the bit is reset, the Micromodem II will accept input only from the remote device except for CTRL-A sequences.

Bit 0 (DLG) Used internally. Flag indicates that dialing is in progress.

CRDLY

$5F8+N=1528+N dec
The remaining location holds the setting for the optional delay after a RETURN. The contents of this location specify the delay in increments of 100 msec.

MICRO-MODEM II FIRMWARE

The firmware on the Micromodem II resides in a single 2708 ROM chip. It occupies two discontinuous areas of memory space in the Apple II. Each peripheral slot in the Apple II has 256 bytes of memory space allocated to it for firmware. The address of this space is determined by the slot number. Programs which occupy this space must be written so that they will work regardless of the address they occupy, since the address varies depending on the slot in which the Micromodem II is installed.

The Apple II also allocates a single 2048-byte area which can be shared by all peripheral boards in the machine for their firmware. This area always has the same address, but since it must be shared with all the other peripherals which may be in the machine, it must have a bank switch. This switch turns the ROM in this area on when the Micromodem II is operating, and turns it off when the Micromodem II is inactive allowing other peripherals to use the memory space. When the Micromodem II is operating, the entire 1024 bytes of the onboard ROM are mapped into the lower half of this space.

There is one location in the 256-byte slot-dependent area that the user might need to access via a CALL statement. This is a special output data routine (located at $CN02=-16382+256*N dec) which outputs the byte stored in location CHAR ($778 = 1912 dec) through the modem. A special routine is needed in some applications which are operating in full-duplex, such as the SELF TEST program, because of an incompatibility between the 6502 microprocessor's indexed write timing and the 6850 ACIA chip. The BASIC POKE statement uses an indexed write, and a POKE to the DATA location will cause any data which the ACIA has received to be discarded.

ADVANCED PROGRAM-MING TECH-NIQUES

The following techniques provide extended capabilities for the experienced programmer by allowing him to change the Micromodem II firmware settings.

All of the examples in the advanced programming section make the following assumptions:

- The variable slot has previously been initialized to the slot number of the Micromodem II.
- The variable D$ has CTRL-D stored within it.

It is possible to change the baud rate of the modem and the number of data bits, stop bits and parity of the data sent and received by the modem. The two most common combinations, 300 baud, no parity and 1 stop bit; and 110 baud, no parity and 2 stop bits, are provided by the firmware. However, many more combinations are possible.

Before attempting to change these options, be sure the Micromodem II has initialized itself. The following line of BASIC will verify that the initialization has been completed:

```
100 PRINT D$"PR#"SLOT
```

SELECTING BAUD RATE

The baud rate is controlled by the least significant bit of the modem control byte. It is preferable to change the byte in the memory and allow the firmware to actually put the change in the hardware register. The following lines will change the baud rate without affecting any other modem functions:

To Select 300 Baud:

Integer BASIC
```
500 POKE 1656+SLOT,PEEK(1656+SLOT)/2*2+1
```

APPLESOFT
```
500 POKE 1656+SLOT,INT(PEEK(1656+SLOT)/2)*2+1
```

To Select 110 Baud:

Integer BASIC
```
500 POKE 1656+SLOT,PEEK(1656+SLOT)/2*2
```

APPLESOFT
```
500 POKE 1656+SLOT,INT(PEEK(1656+SLOT)/2)*2
```

CHANGING CHARACTER FORMATS

The format of characters sent and received by the modem is controlled by bits 2, 3, and 4 of the ACIA control byte. Again it is preferable to change the byte in memory and then let the firmware take care of the actual hardware. This byte normally contains a 1 in the least significant bit plus the appropriate bits in bits 2, 3, and 4 to select the appropriate format. The following line of code can be used to set a character format:

```
700 POKE 2040+SLOT,FSW
```

The value of FSW (FLAG STATUS WORD) can be selected from the following table:

Start Bit		Char. Length		Parity Bit		Stop Bits		Total Length		FSW Decimal	FSW Hex
1	+	7	+	EVEN	+	2	=	11	:	1	01
1	+	7	+	ODD	+	2	=	11	:	5	05
1	+	7	+	EVEN	+	1	=	10	:	9	09
1	+	7	+	ODD	+	1	=	10	:	13	0D
1	+	8	+	NONE	+	2	=	11	:	17	11
1	+	8	+	NONE	+	1	=	10	:	21	15
1	+	8	+	EVEN	+	1	=	11	:	25	19
1	+	8	+	ODD	+	1	=	11	:	29	1D

SENDING A BREAK IN BASIC

The following fragments in a complete program will send a break in BASIC.

```
800 POKE -16250*16+SLOT,PEEK(2040+SLOT)+96
900 POKE -16250*16+SLOT,PEEK(2040+SLOT)
```

Line 800 turns the break on and line 900 turns it off.

TURNING OFF THE CARRIER WITHOUT HANGING UP

It may be desirable to turn off the modem carrier without breaking the telephone connection. An example situation might be a game between two Apple II users in which the opponents could pickup their telephones and talk between rounds. With the modem transmitters running, they would only be able to hear the modem tones.

The following fragments within a complete program will allow the user to turn off the modem and restore voice communication, and later turn the transmitters back on and re-establish modem communication. Do not attempt to execute any INPUT statements from the Micromodem II while the carriers are turned off as this will cause the firmware to detect a loss of carrier and hang up the telephone.

To turn off the carrier:

```
1000 POKE -16251+16*SLOT,136
```

To re-establish modem communications:

```
1900 PRINT D$; "PR#0":PRINT D$ "IN#0:"INPUT "PRESS
     RETURN WHEN DONE WITH PHONE,"I$
2000 PRINT D$; "PR#"; SLOT
2100 IF PEEK(-16250+16*SLOT) MOD 8<4 THEN 2400
2200 X=PEEK(-16249+16*SLOT)
2300 GOTO 2100
2400 PRINT D$; "PR#";SLOT:PRINT "CONNECTION
     RE-ESTABLISHED"
```

Line 1000 turns off the modem carrier by writing directly to the modem control port containing the bits which keep the telephone off-hook and prevent the Micromodem II firmware from performing its initialization. Since the transmitter enable bit is off, the transmitter is turned off.

Line 1900 waits for the user to press RETURN when finished with the telephone. Line 2000 outputs a RETURN to the Micromodem II, which causes the firmware to copy its remembered status back into the modem control port. This turns the transmitter back on. Line 2100 reads the ACIA status port and checks the carrier detect bit to see if there is a carrier from the other end. Remember that you cannot do an INPUT statement again until there is a valid carrier or the firmware will hang up the telephone. Line 2200 unloads the ACIA data register. This operation is necessary to satisfy the ACIA chip. Line 2400 then sends a message to the interrupted program so that it will know that communication has been re-established. In a complete program, a message should be sent to tell the other computer to return to the interrupted program.

ENTERING TERMINAL MODE FROM A PROGRAM

It is often convenient to go back and forth between a BASIC program and terminal mode, especially when operating with two Apple II computers equipped with Micromodem IIs.

The terminal program in the Micromodem II firmware is entered from the input entry point if the TERM bit is set in the FLAG byte. For a program to activate the terminal mode, it needs to set that bit and then call for input from the Micromodem II. The following fragment within a complete program illustrates:

```
 5 D$=CHR$(4):REM D$=CTRL-D
10 PRINT D$; "PR#0"
20 PRINT D$; "IN#3"
40 POKE 1912+SLOT,10
50 INPUT I$ or END
```

NOTE: For full-duplex line 40 should read:

```
40 POKE 1912+SLOT,138
```

Location 1912+SLOT is the FLAG byte. The 10 is the sum of 8 + 2, where 8 is the binary weight of the TERM bit, and 2 is the binary weight of the KBDE bit. When going into terminal mode, be sure to set the KBDE bit or the Apple II keyboard will be disabled except for CTRL-A sequences. Since the DISPO bit (weight 128) was not set, this program will start the terminal mode in half-duplex (necessary to communicate with another Micromodem II in terminal mode). If full-duplex mode is needed, add the binary weight of the DISPO bit (128) to the constant which is POKE'd into the FLAG byte.

If the program is communicating with another Micromodem II-equipped Apple II, it can also put the other computer into terminal mode by sending it a CTRL-T. In order for this to work properly, the other Apple II must be waiting for input from its Micromodem II (be executing an INPUT or GET statement), and should not have the Micromodem II selected for output. The half/full-duplex status will depend on the setting of the DISPO bit in that computer.

ENTERING A PROGRAM FROM TERMINAL MODE

When exiting terminal mode after entering it with the above technique, you will return to the INPUT statement in the program (line 50 in the example). That INPUT statement will be waiting a RETURN. When it receives a RETURN, the program will continue executing at the next line.

The program can also return the other computer to remote console mode by sending a CTRL-R to the other Micromodem II. If the other computer entered terminal mode by setting the DISPO bit and executing an INPUT statement, then its program will be restarted if the CTRL-R is followed by a RETURN.

EXAMPLE:

```
50 INPUT I$:REM THE SAME STATEMENT 50 AS ABOVE
60 PRINT D$;"PR#";SLOT
70 PRINT R$
```

The string variable R$ is assumed to contain a CTRL-R. Since line 70 did not end with a semicolon(;) BASIC will supply a RETURN at the end of the line.

AUTOMATIC LOG-ON TO A TIME SHARE COMPUTER

When writing programs designed to automatically dial and log-on to a time share computer system, a timing problem may develop. All modems have a carrier detection delay time which can range from a few msec to several seconds. When the Micromodem II firmware has finished dialing, it waits for the other modem to send a carrier. When the Micromodem II detects the carrier, the Micromodem II turns on its own carrier and immediately returns to the BASIC program where the next line will execute. However, the other modem has not yet detected our carrier and a 2 to 4 second wait is necessary before trying to send data to the other modem.

The following fragment, when part of a complete program, provides this delay:

```
10 D$=CHR$(4) :REM D$=CTRL-D
30 PRINT D$;"PR#3"
40 PRINT CHR$(17);"394-4220"
50 FOR X=1 TO 3000
60 NEXT X :REM 4 SECOND DELAY
70 POKE 2043,9 :REM 7 BITS, EVEN PARITY
80 PRINT "PASS WORD"
90 PRINT D$;"PR#0"
100 PRINT D$;"IN#3" :REM ENABLE FOR INPUT
110 INPUT A$ :REM GET REPLY
```

The "PASS WORD" and the remainder of the program will depend on the user's application. "PR#3" and "IN#3" statements should never be in effect at the same time in this application.

MICROMODEM II MEMORY USAGE CHART

HARDWARE READ-ONLY REGISTERS

SLOT	1	2	3	4	5	6	7
RI	$C0C5 (-16235)	$C0A5 (-16219)	$C0B5 (-16203)	$C0C5 (-16187)	$C0D5 (-16171)	$C0E5 (-16155)	$C0F5 (-16139)
STATUS	$C096 (-16234)	$C0A6 (-16218)	$C0B6 (-16202)	$C0C6 (-16186)	$C0D6 (-16170)	$C0E6 (-16154)	$C0F6 (-16138)
DATA	$C097 (-16233)	$C0A7 (-16217)	$C0B7 (-16201)	$C0C7 (-16185)	$C0D7 (-16169)	$C0E7 (-16153)	$C0F7 (-16137)

HARDWARE WRITE-ONLY REGISTERS

SLOT	1	2	3	4	5	6	7
CR2	$C095 (-16235)	$C0A5 (-16219)	$C0B5 (-16203)	$C0C5 (-16187)	$C0D5 (-16171)	$C0E5 (-16155)	$C0F5 (-16139)
CR1	$C096 (-16234)	$C0A6 (-16218)	$C0B6 (-16202)	$C0C6 (-16186)	$C0D6 (-16170)	$C0E6 (-16154)	$C0F6 (-16138)
DATA	$C097 (-16233)	$C0A7 (-16217)	$C0B7 (-16201)	$C0C7 (-16185)	$C0D7 (-16169)	$C0E7 (-16153)	$C0F7 (-16137)

R/W MEMORY LOCATIONS USED BY FIRMWARE

SLOT	1	2	3	4	5	6	7
CRDLY	$5F9 (1529)	$5FA (1530)	$5FB (1531)	$5FC (1532)	$5FD (1533)	$5FE (1534)	$5FF (1535)
MODEM	$679 (1657)	$67A (1658)	$67B (1659)	$67C (1660)	$67D (1661)	$67E (1662)	$67F (1663)
LOCSE	$6F9 (1785)	$6FA (1786)	$6FB (1787)	$6FC (1788)	$6FD (1789)	$6FE (1790)	$6FF (1791)
FLAGS	779 (1913)	$77A (1914)	$77B (1915)	$77C (1916)	$77D (1917)	$77E (1918)	$77F (1919)
ACIA	$7F9 (2041)	$7FA (2042)	$7FB (2043)	$7FC (2044)	$7FD (2045)	$7FE (2046)	$7FF (2047)

FIRMWARE ENTRY POINTS

SLOT	1	2	3	4	5	6	7
ENT0	$C100 (-16128)	$C200 (-15872)	$C300 (-15616)	$C400 (-15360)	$C500 (-15104)	$C600 (-14848)	$C700 (-14592)
OUTA	$C102 (-16126)	$C202 (-15870)	$C302 (-15614)	$C402 (-15358)	$C502 (-15102)	$C602 (-14846)	$C702 (-14590)
OUT	$C105 (-16123)	$C205 (-15867)	$C305 (-15611)	$C405 (-15355)	$C505 (-15099)	$C605 (-14843)	$C705 (-14587)
IIN	$C107 (-16121)	$C207 (-15865)	$C307 (-15609)	$C407 (-15353)	$C507 (-15097)	$C607 (-14841)	$C707 (-14585)

SLOT INDEPENDENT MEMORY LOCATIONS

N0	$6F8 (1784)
CHAR	$778 (1912)
CN	$7FA (2040)

CHAPTER 8: FIRMWARE SPECIFICATIONS

ENTRY POINTS

Name	Address	Function
ENT0	CN00	Initial input or output call from the Apple II monitor. If the Micromodem II is reset, apply defaults. Determine by examination of the console switch locations whether a call is for input or output. Modify input or output console switch locations as necessary to direct further input and output to the correct routines. Perform input or output function as determined above.
OUTA	CN02	Special output call for full-duplex operation. Output a byte from the location CHAR in a manner which avoids the 6502 false read problem.
OUT	CN05	Normal output call. Output byte in A register subject to all output options, etc.
IIN	CN07	Normal input location. Obtain a byte from either the modem or keyboard in accordance with options set, and return it in the A register.

DEFAULT INITIALIZATION

Default initialization is performed on entry via ENT0 if the RESET signal is present from the modem. This signal is removed by operations performed by the initialization sequence and does not return unless a hardware reset occurs or the RESET bit in the modem control register is cleared.

DEFAULT SETTINGS

Lower-to-upper-case translation enabled. KBDE flag bit set, all other flag bits reset.
Data format: 8 data bits, no parity, 1 stop bit.
High baud rate (300 bps).
Telephone on-hook, modem turned off.
300 msec LINE FEED delay selected but not enabled.

FEATURES OF INPUT ROUTINE (IIN)

If the TERM is flag set, enters terminal mode.
If on-hook ignores input from the modem.
If on-hook and the telephone rings, answers it:

DISPLAY:

MICROMODEM II:RING

At end of the ring takes the telephone off-hook.

DISPLAY:

MICROMODEM II:AWAIT CARR.

Puts the modem in answer mode.
Turns on the carrier.
Waits up to 30 sec for other carrier.
If no carrier, abandons call.

If a carrier detected:

DISPLAY:

MICROMODEM II:CONN.

If TRAN flag not set, the following control characters are accepted from the modem:

Character	Response
CTRL-T	Sets TERM flag and enters terminal mode.
CTRL-N	Sets LFI flag to enable line feed insertion. Disables display (DISPO=1). Clears Apple II screen. Disables keyboard.
CTRL-Y	Jumps directly to Apple II monitor ($FF65).
LINE FEED($10)	Discards.
NULL($00)	Discards.
DELETE($7F)	Discards.

If TRAN flag not set and LOCSE = $20 then translates lower-case characters to equivalent upper-case.

If off-hook and no carrier is present, abandons call.

All CTRL-A sequences described in terminal mode except CTRL-A CTRL-Q (start dialing) are accepted from keyboard and processed regardless of status of KBDE (keyboard enable).

If KBDE flag is set, accepts all characters entered as valid input. If KBDE flag is reset, ignores all keyboard input except CTRL-A sequences.

When the call is abandoned:

Hangs up the telephone.
Turns off the modem.

DISPLAY:

MICROMODEM II:NO CARR.

MICROMODEM II:HUNG UP

FEATURES OF OUTPUT ROUTINE (OUT)

If on-hook and the character CTRL-Q is sent, initiates DIALING sequence.

If dialing is in progress, dials characters as sent (see DIALING).

If TRAN flag is not set, and CTRL-Z is sent, hangs up the telephone.

DISPLAY:

MICROMODEM II:HUNG UP

If DISPO flag is 0 (display selected), copies all modem output directly to the Apple II display. During actual transmission via the modem, the following characters are accepted from the modem input if TRAN is not set:

Character	Response
CTRL-S	Stops transmitting until any other character is received.
CTRL-Y	Jumps directly to Apple II monitor (at $FF65).

FEATURES OF DIALING

Initiated by output of CTRL-Q, if on hook, regardless of the status of TRAN flag.

Initiated by CTRL-Q in terminal mode.

Once DIALING is initiated, it is not exited until a LINE FEED RETURN, or CTRL-Z character is detected. On initiation, the following occurs:

DISPLAY:

MICROMODEM II:DIALING:

The phone is taken off-hook.

A two-second delay occurs to allow the telephone exchange to return a dial tone.

The timing of dial pulses is in accordance with Bell System specifications:

Each pulse = 61 msec on-hook

Interpulse delay = 39 msec

Interdigit delay = 600 msec minimum

Characters output while DIALING are handled as follows:

Character	Response
Digits 1-9	Dialed as 1 to 9 pulses
Digit 0	Dialed as 10 pulses
*	2-second delay (for second dial tone)
RETURN	Terminates dialing. Sets the modem to originate mode. DISPLAY: MICROMODEM II:AWAIT CARR. Waits up to 30 seconds for answering carrier. If a carrier is detected, enables the modem transmitter. If no carrier is detected, abandons call.
LINE FEED	Terminates DIALING; does not enable the modem or wait for a response.
CTRL-Z	Abandons call.
All others	Displays but otherwise ignores.

CTRL-Z received from the keyboard any time during the 30 second wait for an answering carrier causes the call to be abandoned. When the call is abandoned, the following message is displayed:

MICROMODEM II:NO CARR.

MICROMODEM II:HUNG UP

FEATURES OF TERMINAL MODE

Entered from input (IIN) when CTRL-A CTRL-F or CTRL-A CTRL-H sequences are entered from keyboard.

Entered from input (IIN) if TERM flag is set prior to calling IIN.

Exited (back to IIN) if CTRL-A CTRL-X is entered from the keyboard.

Exited (back to IIN) if CTRL-R received from the modem and TRAN flag is not set.

Entered if TRAN is not set and CTRL-T is received from modem.

DISPLAY: (on entry)

MICROMODEM II:BEGIN TERM

DISPLAY: (on exit)

MICROMODEM II:END TERM

The local display is switched through the console switch locations CSWL, CSWH and is compatible with Apple II serial and parallel printer interface cards.

The following control sequences are recognized from the keyboard.

Sequence	Response
CTRL-A CTRL-1	Sets modem speed to 110 baud, format 8 data bits, 2 stop bits.
CTRL-A CTRL-3	Sets modem speed to 300 baud, format 8 data bits, 1 stop bit.
CTRL-A CTRL-H	Sets half-duplex (enables display, DISPO=0)
CTRL-A CTRL-F	Sets full-duplex (disables display, DISPO=1)
CTRL-A CTRL-Z	Hangs up the telephone and turns off the modem.
CTRL-A CTRL-Q	Initiates dialing sequence.
CTRL-A CTRL-S	Transmits break until any other character is typed.
CTRL-A CTRL-X	Exits terminal mode.

SOFTWARE-CONTROLLED OPTIONS

There are many software-controlled options. This is a list of the important ones. For operational details of individual options, please see the appropriate section of this manual.

DATA FORMAT

Two standard data formats may be selected via CTRL-A sequences from the keyboard: 8 bits, no parity, 2 stop bits at 110 baud, and 8 bits no parity, 1 stop bit at 300 baud. Other data formats may be selected by storing appropriate values in the RAM location for the ACIA register.

BAUD RATE

300 baud is the default speed. 110 baud may be selected via a CTRL-A keyboard sequence. The rates may also be selected under software control by modifying the contents of the RAM location for the modem control register.

LOWER-TO-UPPER CASE TRANSLATION

Normally enables ($20 in LOCSE). May be disabled by writing 0 to LOCSE.

OPTIONS SELECTED VIA BITS IN FLAG BYTE

Several software-controlled options are selected by setting or clearing appropriate bits in the FLAG byte in RAM.

Bit	Option Controlled
DISPO	Local display of data transmitted via modem.
LFI	Insertion of LINE FEED after RETURN, also enables delay after RETURN. Duration of the delay is controlled by contents of location WAIT, which contains a delay factor in 10 msec units.
TERM	When set causes entry to terminal mode from IIN.
TRAN	Code transparency. When set disables recognition of all control characters from any source with the following exceptions: CTRL-A sequences from the keyboard CTRL-Q output if on-hook LINE FEED if LFI is selected
KBDE	Keyboard enable. If not set, only CTRL-A sequences are accepted from the local keyboard.

ADDITIONAL INFORMATION

Shared ROMs are shut off by access to CFFF when entered from any of the four valid entry points.

The flashing cursor is removed from the Apple II display when a character is received.

The flashing cursor is placed on the screen when awaiting a character in terminal mode.

Random number location (RNDH) is incremented while awaiting characters.

X and Y registers and status of interrupt enable are preserved through all entries.

ACIA control register (CR1) and MODEM control register (CR2) are refreshed from their RAM storage locations each time a character is transmitted.

The hex value CN (n=slot number) is maintained at location $7F8 during all operations.

CHAPTER 9: BACKGROUND INFORMATION

MODEMS

A modem is a data transmission device that allows terminal-computer and computer-computer communication over a telephone line. The primary reason for the development and popularity of modems is economic. Modems allow information to be moved, on demand, from one place to another at very low energy cost.

To communicate over a telephone line, the computer or terminal at each end of the line must be equipped with a modem. The modem at the sending end converts binary digital data (ones and zeros) from the terminal or computer into analog signals (suitable for transmission over a telephone line). The modem at the receiving end reverses the process. The word modem is a contraction for MODulator-DEModulator. A modulator is the device that changes digital data to analog signals and a demodulator changes analog signals back into digital data.

The Micromodem II is designed to be completely compatible with the communication frequencies and modulation techniques of the Bell System (Western Electric) Model 103 low speed modem. The Bell System 103 modem and its various equivalents are the most widely used type of modem in North America. It is used by virtually all time sharing systems as their standard mode of access. This popularity is due to the simplicity of the 103's FSK (frequency shift keying) modulation technique, the reasonable cost of the circuitry required to implement it and the number of Model 103 compatible modems already installed with which to communicate.

HISTORY

Modems were first developed by the telephone company which claimed that modems were a part of the telephone system and therefore a part of its protected monopoly. The claim was not disputed until the development of the acoustic coupler, a modem that is not directly wired to the telephone line. The acoustic coupler transmits and receives data by "listening" and "speaking" through the handset of a regular telephone. The telephone company claimed the acoustic coupler was an illegal device and use of one could result in termination of telephone service. As a result of this claim, the Carterphone Company, a manufacturer of an acoustically-coupled device for use with two-way mobile radios, was forced out of business and filed suit against the telephone company. The Supreme Court decided in favor of Carterphone and ruled the acoustic couplers were legal devices. More recent decisions have broadened the regulations for interconnect, giving the FCC power to license devices for use with the telephone network and limiting the telephone company's protected monopoly to the running of the network which connects telephones together.

As smaller, less expensive microcomputers have become more widespread, the Bell 103-type modem (including the Micromodem II) has become increasingly popular.

TRANSMISSION SPEED

Transmission speed is the speed at which data is transmitted over a communications line. This speed is expressed in bits per second. It takes several bits to determine a character, and the number of bits in a character depends on the transmission code and transmission techniques used. Each bit transmitted requires a specific amount of time on the line, and the bit rate or bits per second (bps) is the reciprocal of this amount of time. Normal data transmission is handled by 7 or 8 bits to produce a character plus added control bits which normally expand the number of bits per character by 2 or 3.

A modem that transmits data at 110 bits per second or 10 characters per second (cps) is referred to as having a 110 baud rate. Likewise, a 300 baud rate denotes that data is being transmitted at a speed of up to 300 bps or 30 cps.

By far, the largest number of modems in use are the 0-300 bps variety. These units are used for interactive terminal-computer and computer-computer communications, data acquisition, time sharing and data logging, financial transactions and information utility applications. Almost all 0-300 bps modems in the U.S., including acoustic couplers, are compatible with the Bell System standard 103-type modems.

COMMUNICATION LINES

The communication line is the connection between computers and terminals. This line is usually classified according to the direction in which it moves data: simplex, full-duplex and half-duplex. A simplex line is one which carries data from one point to another in only one direction. Mass media devices such as TVs and radios are examples of simplex communications.

FULL-DUPLEX

A full-duplex line provides two way communication by using two communications lines, one going in each direction. The prime example is two-way telephone conversation which allows communications from both directions at the same time. One important feature of a full-duplex modem is echo-plex. When a character is typed, it travels to the distant computer and is echoed back before it appears on the user's screen. The advantage of this echo-plex procedure is that a character garbled on the telephone line will appear garbled on the screen and a character lost in transmission will not appear at all. Most time-sharing and data access systems feature echo-plex.

HALF-DUPLEX

A half-duplex line allows two way communication with only one communication line. In most half-duplex systems, the terminal must wait for the computer on the other end to finish before it can transmit data. In other words, you cannot receive data while transmitting data. An example of half-duplex communications is the CB radio which only allows one person to communicate at a time.

The 103-type modem was designed to take advantage of full-duplex communications lines. Until recently, the Bell 103-type modem was the only full-duplex modem. Most faster modems for larger computers are half-duplex.

APPENDIX A: MODIFYING AND USING DATAMOVER

USING DATAMOVER WITH THE MICRO-MODEM II

It is necessary to install a patch to your copy of Datamover for use with the Micromodem II.

First LOAD the Datamover. It resides at location $8Ø0 through $A4F.

From Monitor (prompt '*') do:
879: AE F8 Ø7 A9 ØA 9D B8 Ø6 2Ø ØC FD 4C

From Integer BASIC (prompt ')') do:

```
1Ø FOR L=2169 TO 2182
2Ø READ A: POKE L,A
3Ø NEXT L
4Ø DATA 174, 248, 7, 169, 1Ø, 157, 184
5Ø DATA 6, 32, 12, 253, 76
6Ø END
```

Save the patched program on disk for later use.

The Datamover program and its operation are documented in the manual, *Communications Interface Card, Addendum to the Installation and Operating Manual* published by Apple Computer, Inc. When using the Micromodem II with this program, instead of putting the telephone in the acoustic coupler, type CTRL-A CTRL-Q, the telephone number and RETURN. The Micromodem II will establish the telephone connection.

APPENDIX B: ASCII CONTROL CHARACTER CODE TABLE

CODE	HEX	DEC	CODE	HEX	DEC	CODE	HEX	DEC	CODE	HEX	DEC
NUL	00	0	SP	20	32	@	40	64	`	60	96
CTRL A	01	1	!	21	33	A	41	65	a	61	97
CTRL B	02	2	"	22	34	B	42	66	b	62	98
CTRL C	03	3	#	23	35	C	43	67	c	63	99
CTRL D	04	4	$	24	36	D	44	68	d	64	100
CTRL E	05	5	%	25	37	E	45	69	e	65	101
CTRL F	06	6	&	26	38	F	46	70	f	66	102
CTRL G	07	7	'	27	39	G	47	71	g	67	103
CTRL H	08	8	(	28	40	H	48	72	h	68	104
CTRL I	09	9	)	29	41	I	49	73	i	69	105
CTRL J	0A	10	*	2A	42	J	4A	74	j	6A	106
CTRL K	0B	11	+	2B	43	K	4B	75	k	6B	107
CTRL L	0C	12	,	2C	44	L	4C	76	l	6C	108
CTRL M	0D	13	–	2D	45	M	4D	77	m	6D	109
CTRL N	0E	14	.	2E	46	N	4E	78	n	6E	110
CTRL O	0F	15	/	2F	47	O	4F	79	o	6F	111
CTRL P	10	16	0	30	48	P	50	80	p	70	112
CTRL Q	11	17	1	31	49	Q	51	81	q	71	113
CTRL R	12	18	2	32	50	R	52	82	r	72	114
CTRL S	13	19	3	33	51	S	53	83	s	73	115
CTRL T	14	20	4	34	52	T	54	84	t	74	116
CTRL U	15	21	5	35	53	U	55	85	u	75	117
CTRL V	16	22	6	36	54	V	56	86	v	76	118
CTRL W	17	23	7	37	55	W	57	87	w	77	119
CTRL X	18	24	8	38	56	X	58	88	x	78	120
CTRL Y	19	25	9	39	57	Y	59	89	y	79	121
CTRL Z	1A	26	:	3A	58	Z	5A	90	z	7A	122
ESC	1B	27	;	3B	59	[	5B	91	{	7B	123
FS	1C	28	<	3C	60	\	5C	92	l	7C	124
GS	1D	29	=	3D	61	]	5D	93	}	7D	125
RS	1E	30	>	3E	62	-	5E	94	~	7E	126
US	1F	31	?	3F	63	—	5F	95	DEL	7F	127

APPENDIX C: RETURN FOR REPAIR PROCEDURES

When returning a unit for repair, it must be accompanied by proof of date of purchase. Units returned without proof of date of purchase or out of warranty units will be repaired or replaced (at Hayes' option) and the customer will be charged for parts and labor.

1. Call Hayes Customer Support for a return authorization number (RA number).

2. If possible, pack the Micromodem in its original box.

3. If the original box is not available, pack the Micromodem in a sturdy corrugated box and cushion it with NON-STATIC material such as newsprint. Do not use highly static materials like plastic wrap or styrofoam based packaging materials (peanuts or beads), as they may further damage the modem in transit. NOTE: Ask your dealer for a Hayes warranty/repair corrugated box to mail the unit to Hayes for repair.

4. When returning a Micromodem to the Hayes facility for repair, always include the following information:

   NAME
   ADDRESS
   CITY, STATE, ZIP
   TELEPHONE NUMBER
   RETURN AUTHORIZATION NUMBER
   PROBLEM DESCRIPTION

   A short description of the problem(s) experienced is adequate.

5. All Micromodems returned to Hayes for repair should be shipped UPS or U.S. Postal Service prepaid.

   HAYES MICROCOMPUTER PRODUCTS WILL NOT ACCEPT MODEMS THAT HAVE BEEN SHIPPED C.O.D. It is recommended that modems be insured when shipped.

6. Mail package to:

   Hayes Microcomputer Products, Inc.
   Attention: Warranty/Repair
   5835 Peachtree Corners East
   Norcross, Georgia 30092
   RA Number

   Please include RA number on mailing label.

7. HAYES MICROCOMPUTER PRODUCTS WILL NOT ACCEPT MODEMS SHIPPED FROM OUTSIDE THE UNITED STATES.

APPENDIX D: PRODUCT SPECIFICATIONS

DATA FORMAT	Serial, binary, asynchronous 7 or 8 data bits, 1-or 2-stop bits, odd, even or no parity.
LOWER CASE CHARACTERS	Can be optionally converted to upper case, or can be passed through unmodified
FIRMWARE	1Ø24 byte read only memory (ROM)
POWER CONSUMPTION	1.5 W. Typical
CARD SIZE	7″ × 3″ including connector fingers
MICRO-COUPLER SIZE	5½″ × 3¼″ × 1⅜″
MODEM COMPAT-IBILITY	Bell System 1Ø3-compatible originate or answer mode, dial pulse dialing and auto answer — 5Ø dBm receive sensitivity — 1Ø dBm transmit level 11Ø or 3ØØ baud data rates
FCC REGISTRA-TION	FCC Registration No. BI986H-622Z6-PC-E Ringer Equivalence Ø.4B Connects with modular jacks RJ11W or RJ11C
SUPPLIED WITH	Modem interface card, firmware in ROM, Microcoupler, diskette, connector cables, complete owner's manual.

TWO-YEAR LIMITED WARRANTY

Implied warranties limited to duration of express warranty (some states do not allow limitations on how long an implied warranty lasts, so the above limitation may not apply to you).

Hayes Microcomputer Products, Inc., warrants to the original owner that each of its products, and all components therein contained will be free from defects in materials and or workmanship for two years from the date of purchase.

In the event of malfunction or other indication of failure attributable directly to faulty workmanship and or material, then, upon return of the product with proof of date-of-purchase to **Hayes Microcomputer Products, Inc.,** 5835 Peachtree Corners East, Norcross, Georgia 30092 (postage paid), **Hayes Microcomputer Products, Inc.**, will, at its option, repair or replace said products or components, to whatever extent it shall deem necessary to restore said product to proper operating condition. During the first two years after the date of purchase all labor and materials will be provided without charge. There shall be no warranty for either parts or labor after the expiration of two years from the date of purchase.

Units must be returned postage prepaid and insured. Units returned without proof of date-of-purchase, or out-of-warranty units returned will be repaired or replaced (at the option of **Hayes Microcomputer Products, Inc.**) and customer will be charged for parts and labor.

Products will be returned to customer after repair or replacement has been completed by carrier and method chosen by **Hayes Microcomputer Products, Inc.,** to any destination within the United States of America. Should the customer desire some other specific form of conveyance, or be located beyond the USA borders, then the customer must bear the cost of return shipment.

The responsibility for the failure of any **Hayes Microcomputer Products, Inc.** computer product, or component thereof, which at the discretion of **Hayes Microcomputer Products, Inc.**, shall have resulted from accident, abuse, or misapplication of the product, shall be assumed by the customer, and **Hayes Microcomputer Products, Inc.** shall assume no liability as a consequence of such events under the terms of this warranty.

While every effort on the part of **Hayes Microcomputer Products, Inc.** has been made to provide clear and accurate technical information on the application of its products, **Hayes Microcomputer Products, Inc.** assumes no liability in any events which may arise from the use of said technical information.

This warranty gives you specific legal rights and you may also have other rights which vary from state to state.

This warranty is in lieu of all other express warranties which now or hereafter might otherwise arise with respect to this product. Any and all implied warranties, including the warranties of merchantability and fitness for particular use, shall have no greater duration than the duration period for the express written warranty applicable to this product as shown above, and shall terminate automatically at the expiration of such duration period. No action shall be brought for breach of any implied or express warranty after one year subsequent to the expiration of the duration period of the express written warranty.

Incidental and consequential damages caused by malfunction, defect or otherwise and with respect to breach of any express or implied warranty, are not the responsibility of **Hayes Microcomputer Products, Inc.**, and to the extent permitted by law, are hereby excluded both for property and, to the extent not unconscionable, for personal injury damage.

Please note that laws vary from state to state and that some of the provisions of this warranty may not be appropriate to the laws of your jurisdiction.

Hayes Microcomputer Products Inc.
5835 Peachtree Corners East. Norcross, Georgia 30092
(404) 449-8791

FOR YOUR RECORDS

Fill out the following information for your records. It is very helpful when discussing repair questions with the factory.

PRODUCT

105731

SERIAL NO.

PLACE OF PURCHASE

9/19/81

DATE OF PURCHASE