

Section 9 - QUICK REFERENCE GUIDE

COMMAND SUMMARY - The following list is a summary of commands. All commands issued from the local keyboard must be preceded by a 'Ctrl-A'. (The only exception to this rule is to hangup while dialing or waiting for a dialtone/carrier.)

LOCAL COMMANDS	FUNCTION
Ctrl-A Ctrl-F	Enter Terminal Mode in FULL duplex
Ctrl-A Ctrl-H	Enter Terminal Mode in HALF duplex
Ctrl-A Ctrl-I	Initialize Firmware to cold entry defaults
Ctrl-A Ctrl-P	Toggle Printer Buffer On/Off
Ctrl-A Ctrl-Q	Init dialing (pickup phone in Voice mode)
Ctrl-A Ctrl-S	Transmit a BREAK until another char is sent
Ctrl-A Ctrl-V	Enter voice mode (turns off modem & turns on handset)
Ctrl-A Ctrl-X	Exit Terminal Mode (carrier remains on)
Ctrl-A Ctrl-Z	Hangup phone

REMOTE COMMANDS	FUNCTION
Ctrl-N	Xmit linefeed & CR delay to remote
Ctrl-R	Exit terminal mode
Ctrl-S	Stops transmission (any key starts)
Ctrl-T	Enter terminal mode
Ctrl-Y	Enter Monitor
Ctrl-Z	Hangup phone

FIRMWARE-GENERATED MESSAGES - The Firmware outputs a message when it responds to a command or when a significant event has occurred (i.e. carrier found or lost). All messages begin with the prefix "MODEM:" to let you know where the message originated. What follows is a list of the various messages generated by the Firmware along with a brief description.

BEGIN TERM	Terminal mode entered
CARRIER?	Wait up to 45 seconds to receive valid carrier
CARR. OK	Valid carrier received
DIALING	Prompts user to enter phone number
DIALTONE?	Wait up to 20 seconds for continuous audio tone
END TERM	Exit terminal mode & return to BASIC
INIT	Firmware is re-initialized to coldstart defaults
NO CARR.	Carrier signal interrupted
OFF-LINE	Phone is hungup
ON-LINE	Phone is picked up
RING	Phone is ringing
TERM	Signifies non-voice mode after dialing
VOICE	Modem tones turned off for voice communication

TABLE 1 - MINI-SWITCH SETTINGS

Sw No.	ON	OFF
1	Pulse Dialing	Tone Dialing
2	110 Baud	300 Baud
3	not used	not used
4	Pickup on 9th ring	Pickup on 1st ring

7/28/82 - 1.0

FIRMWARE

FOR THE

APPLE[®]·CAT II[™]

THE
'MIRROR'

(201) 838-9027
 J. KOVACS
 MICRO-WARE DISTRIBUTING INC.
 1342 B ROUTE 23
 BUTLER, NJ 07405

729
RAK-WARE

41 Ralph Road

West Orange NJ 07052

COMMUNICATIONS FIRMWARE FOR THE APPLE-CAT

Firmware & Documentation by Bob Kovacs
Copyright (C) 1982 RAK-WARE

TABLE OF CONTENTS

Section	Title	Pg No.
1	Introduction	1
2	Firmware Installation	2
3	Dumb Terminal Mode	3
4	Remote Console Mode	5
5	Programmable Modem	6
6	Miscellaneous Notes	10
7	Apple-Cat vs. Comm card	11
8	Firmware Programming Details	13
9	Quick Reference Guide	15

NOTICE

The Firmware is sold on an "AS IS" basis. The purchaser assumes the entire risk of its operation. RAK-WARE will in no event be held liable for direct, indirect, or incidental damages including, but not limited to, any interruption of service, loss of business or anticipatory profits, or any other consequential damages.

RAK-WARE reserves the right to make changes and improvements to the Firmware and/or documentation at any time without prior notification.

The program (contained in the Firmware) and its documentation are copyrighted. You may not distribute copies of, or any part of, either this documentation or the program to others without specific approval from RAK-WARE.

PRODUCT/TRADEMARK ACKNOWLEDGMENTS

Applesoft & Apple II: Apple Computer Co.	Apple-Cat II: Novation, Inc.
Micromodem II: Hayes Microcomputer Products	Videoterm: Videx
Full-View 80: Bit3 Computer Corp.	Smarterm: Advanced Logic Systems

Section 1 - INTRODUCTION

The Firmware consists of a preprogrammed 2K byte read-only-memory (ROM) which physically resides on the Apple-Cat board. This Firmware contains a number of routines which instruct the modem without the need for additional software. In other words, it turns the Apple-Cat into a 'smart' board.

The Firmware has a command structure similar to Apple's Communication Card and the Hayes Micromodem II. In fact, many of the commands have identical syntax and function. In addition, several new features have been added to make the Apple-Cat even more versatile. There are, of course, hardware differences between the Apple-Cat and the other modem cards which preclude 100% interchangeability. Thus any existing applications program which reads or writes directly into control registers on those cards will probably not function correctly with the Apple-Cat without some modification. At the command level, however, there are no significant differences.

Here's a quick summary of features provided by the Firmware:

- * Three operational modes: Dumb Terminal, Remote Console & Programmable Modem.
- * Both PULSE and TONE dialing modes provided - switchable even while dialing.
- * Audible ring detect - your Apple actually 'rings' when receiving a call.
- * Ring-back option permits multi-use of a single phone line.
- * True dialtone detection prevents premature dialing.
- * Supports 80-column Cards and the '1-wire' shift key modification.
- * Built-in printer buffer minimizes lost data.
- * VOICE mode permits direct verbal communication using Apple-Cat handset.

Section 2 - INSTALLATION

Installation of the Firmware chip into the Apple-Cat board is relatively straightforward and shouldn't take more than 10 minutes to perform. It does require, however, that adequate precautions be taken to prevent damaging the chip due to either mishandling or static electricity. Do not hesitate to seek professional help if you are hesitant about installing the Firmware yourself. Read over the following set of instructions first before proceeding with installation.

Turnoff power to the Apple and remove the Apple-Cat board. Place the board on a flat surface with the electronic components facing up. Orient the board so that the slot contacts are facing you (the words 'NOVATION' should appear rightside up). Identify the empty Firmware socket which is located at the right end of the board. The word 'FIRMWARE' appears just under the socket. If your board already has something plugged into that socket, it must be removed before proceeding.

Since the Firmware can be damaged by static electricity, discharge yourself by momentarily touching a grounded metal surface. You can use your Apple's power supply case providing the Apple is plugged into a properly grounded 3-prong electrical outlet. Next remove the Firmware from its protective shipping wrapper and check to see that no pins are bent. Insert the Firmware into the Apple-Cat socket with a firm rocking motion until it is fully seated. The notch on the end of the Firmware should face to the left.

Inspect the Firmware closely to verify that all pins are properly inserted into the socket.

The Apple-Cat has a set of mini-switches located just below the phone line connection. These switches are used to set up several default conditions when the Firmware is initialized. For now, set all the switches down (OFF). If your phone line does not support tone dialing then set switch number 1 up (ON).

Plug the Apple-Cat card back into your Apple. Although slot number 2 is preferred, it can be plugged into any slot except slot number 0 (the left-most one).

Section 3 - DUMB TERMINAL MODE

The Dumb Terminal Mode turns the Apple into a simple teletype terminal for communication over a standard voice-grade telephone link. This mode is useful for many types of activities such as communication with a time-shared computer system, computer bulletin board or another Apple-Cat.

A typical sequence of instructions to use this mode follows:

1. Turn-on and initialize Apple-Cat using an 'INW(slot)' from BASIC or '(slot)Ctrl-K' from the Monitor, where (slot) is the slot number in which the Apple-Cat is installed.
2. Enter Terminal Mode using a 'Ctrl-A Ctrl-F' or the 'Ctrl-A Ctrl-H' command sequence. (This means that you must depress and hold down the CTRL key while typing the letter which follows.) The command you should use depends on whether the system you are dialing into operates in FULL or HALF Duplex. If you are not sure then type 'Ctrl-A Ctrl-H' (you can always switch over to full duplex later if you want to). (Notice, by the way, that all Firmware commands issued from the keyboard consist of a 2 key command sequence where the first key is a Ctrl-A and the second key is a Firmware function code.)
3. You can now use the Apple-Cat to dial the phone. Either pulse or tone dialing may be selected as the default dialing mode by setting Switch # 1 located on the Apple-Cat board (see Table 1).
 - a) Begin the dialing operation with a 'Ctrl-A Ctrl-Q'. There will then be a short pause while the Apple-Cat picks-up the phone to go on-line and waits for a dialtone. The Firmware will display the messages 'ON-LINE' and 'DIALTONE?' to tell you what's happening. (If you are using the optional handset then you can listen-in on these activities.) Normally, it shouldn't take more than 3 to 4 seconds for dialtone detection. If no dialtone is found within 20 seconds then there is probably something wrong and the Firmware will hangup the phone (the message 'OFF-LINE' will also be displayed). If, for example, you forgot to plug in the phone line and then realize it after waiting a few seconds, you can abort dialing by typing 'Ctrl-Z'.
 - b) You can begin to enter the phone number to be dialed after the 'DIALING' message appears on the screen. You must wait until a digit is being dialed before another is entered. Remember that pulse-dialing is much slower than tone-dialing, so that there may be a noticeable delay between entries. Only the digits 0 to 9, the letters T and P and the symbols # (for 3 second pause) and + (for add'l dialtone) are recognized (also * and # for tone dialing). All other characters are ignored, although they will be displayed on the screen. You can add spaces, slashes or dashes to help break up the phone number so that it is easy to read and check for typing errors. The dialing operation is ended by hitting RETURN after the last digit. If you should make an error while dialing and want to abort, just hit 'Ctrl-Z' (the Ctrl-A isn't required here). This hangs up the phone and exits the dialing operation. You must go back to step 3a to re-dial.
 - c) The Firmware next turns off the handset and waits for the system you've called to answer the phone and return a carrier tone (the message 'CARRIER?' is displayed). Since you've originated the call, the Apple-Cat has automatically been configured to ORIGIN and will only communicate with a system configured to ANSWER. If the phone is not answered or a carrier is not found within 45 seconds then the Firmware hangs up the phone. You can exit from the 45 second wait at any time by typing 'Ctrl-Z'.

4. You get to this step only if a valid carrier tone has been received ('CARR. OK' is displayed on the screen). You are now ready to log-on to the system you've dialed into. Use the Apple just like a standard teletype.

If the carrier signal from the other system is interrupted or turned off for any reason then the Firmware will detect this condition and display the message 'NO CARR.'. If the carrier signal is not restored within 1 second then the phone is hungup. Avoid calling from phone lines which have the CALL-WAITING feature since this will interrupt communications if someone else is trying to reach you.

5. Full or Half duplex operation can be toggled at any time by entering either a 'Ctrl-A Ctrl-F' or 'Ctrl-A Ctrl-H' respectively. So if you are seeing double characters on the screen during input, then switch from Half to Full duplex. On the other hand, if the characters you type do not appear on the screen, then switch from Full to Half duplex.

6. Two separate methods can be used to toggle a printer on and off. If you have 'dumb' parallel printer card in Slot #1 (like Apple's card for Centronics-type printer interfaces) you can use a special built-in routine which buffers several lines of data. This will minimize the problem of lost data discussed below. Just type 'Ctrl-A Ctrl-P' to toggle the printer on and off. A short beep tone will sound each time you use this command.

Other printer configurations can be toggled on and off only from outside the Terminal Mode using the usual 'PRN(slot)' command, where (slot) is the printer interface card slot number. You can temporarily exit the Terminal Mode using a 'Ctrl-A Ctrl-X', then issue a 'PRN(slot)' and return to the Terminal Mode using a 'Ctrl-A Ctrl-F' or 'Ctrl-A Ctrl-H' without disconnecting from the system with which you've established communications.

Your printer must be able to print at least as fast as the data is being received or else information will be lost. Some printers do not initiate printing until a RETURN is received, and it may take a second or more to actually print the line and reposition the print head for the next line. Again, information may be lost during this time. Most time-shared computer services, however, do have capability of accomodating hard-copy terminals which suffer from this very same problem. Usually the user has the ability to specify either a delay time or number of null characters subsequent to RETURN and before the next line begins. This gives the printer adequate time to print so that no leading characters are lost. Refer to your time-shared computer systems manual for the proper command syntax.

7. The Firmware lets you exit from the Dumb Terminal Mode and return to BASIC without disturbing the communications link (just type 'Ctrl-A Ctrl-X'). You can use this feature, for example, to run an up/download program after you've logged on to your favorite timeshared computer system to transfer data. Use 'Ctrl-A Ctrl-F' or 'Ctrl-A Ctrl-H' to return back to Dumb Terminal Mode.

8. After your session with the time-shared computer is completed and you've logged-off, type a 'Ctrl-A Ctrl-Z' to hang up the phone (an OFF-LINE message will be displayed). You can either exit the Terminal Mode with a 'Ctrl-A Ctrl-X' or dial up another system by going back to Step 3.

9. Even if you've exited the Dumb Terminal Mode, the Firmware is still "listening" for commands. You will have to type either 'INW0', '0Ctrl-K' or RESET to totally disconnect the Apple-Cat.

Section 4 - REMOTE CONSOLE MODE

This mode allows you to call your Apple from some remote location and make it perform just as if you were sitting in front of it. Although you actually may be miles from your Apple, your remote keyboard functions just like an Apple Keyboard and your remote display functions just like an Apple text screen (with the exception of pure cursor operations). This mode works either with or without DOS. With DOS active you can create, load, run and save files remotely. Here is a typical operational sequence:

1. Turn-on the Apple-Cat by typing 'IN#<slot>' and 'PR#<slot>', where <slot> is the number of the slot in which the Apple-Cat is installed. The order of these two commands isn't important. The Apple-Cat is now waiting for the phone to ring.
2. When a call is received the Apple will toggle the speaker like a phone bell and display 'RING' on the screen. As the phone rings, the Firmware starts to count the number of ring-noring cycles and will pickup the phone when the required number of cycles is achieved. An option is provided to answer the phone either on the 1st ring or the 9th ring. This is selected by Switch # 4 on the Apple-Cat board (see Table 1).

The Firmware employs a ring-back feature which is useful so that the phone line can be shared between an Apple operating in the Remote Console mode and a standard telephone answering machine. With the Firmware set to pickup on the 9th ring and the answering machine set for pickup on the 4th ring, then the answering machine will always pickup first under normal circumstances. With the ring-back feature, however, you can have the Apple-Cat answer the phone by calling and letting it ring once. Then hangup and call again. The Apple-Cat will pickup on the first ring if you callback within 40 seconds of your initial call.

3. Having picked up the phone, the Firmware next waits for a valid carrier to be received. The messages 'ON-LINE' and 'CARRIER?' are displayed to let you know what's happening. The caller must be in ORIGINATE mode since the Apple-Cat is in ANSWER mode. If a valid carrier is not received within 45 seconds of pickup then the Apple-Cat hangs up the phone ('OFF-LINE' is displayed). The Apple-Cat then patiently waits to answer the next call.

4. Depending on which BASIC is active, either an Integer Basic or Applesoft prompt is sent to the remote console to signify that you've been properly connected to your Apple. You can now use your Apple remotely.

5. You may find that your remote console (e.g. CRT terminal or teletype) is displaying successive lines on top of one another. This is because it is not issuing a line feed after each RETURN is received from the Apple. There are two options that can be used to correct this problem. Most CRT terminals can be configured to generate an automatic line-feed after each RETURN. This may be a simple matter of setting an appropriate switch. Otherwise type a 'Ctrl-N' which tells the Firmware to start sending linefeeds.

6. Another potential problem occurs with hardcopy terminals where the leading characters on a line are lost. This happens when the terminal is busy the end of the last line and repositioning the print head for the start of the next line. Type a 'Ctrl-N' to tell the Firmware that it should wait a short time after a RETURN before sending the next line (this command also causes linefeeds to be sent after each RETURN). The wait time defaults to 30 milliseconds (equivalent to one character at 300 Baud and about three characters at 110 Baud). You can easily increase this time to a maximum of 1.27 seconds by POKEing the number of 10 milli-second delays into location 1528+SLOT. This value should not exceed 127.

7. A 'Ctrl-S' command is used to temporarily halt output from the Apple-Cat to your remote console. Typing another character will start it again.

8. You can type a 'Ctrl-Y' to stop certain operations. This command acts sort of like RESET but also puts you into the Monitor. You can get back into BASIC by typing 'Ctrl-C' (or '300G' if DOS is 'up').

9. Typing a 'Ctrl-Z' from the remote console causes the Apple-Cat to hangup the phone. You can either issue this command at the end of your session or just hang up the phone on your end. If you just hangup the phone then the Firmware will detect a carrier loss and hangup too.

Section 5 - PROGRAMMABLE MODEM

This mode allows you to incorporate the various features of the Apple-Cat modem into your own application program. The basic operations generally required for communications are:

	BASIC Statement
a) Answer the phone if it rings	IN#<slot>
b) Receive characters from a remote system	INPUT A\$
c) Dial a number (and wait for carrier)	PRINT CHR\$(17);PHONE\$
d) Send characters to remote system	PRINT B\$
e) Hang up the phone	PRINT CHR\$(26)

Items a and b are input functions whereas items c,d and e are output functions. Input and output functions are handled through the 'IN#<slot>' and 'PR#<slot>' commands respectively. Incoming characters are read via the INPUT statement and outgoing characters or Firmware commands are output via the PRINT statement.

So after turning on the Firmware with a 'PR#' command you can command the Apple-Cat to pickup the phone and dial a number. This is done with a 'PRINT CHR\$(17);"123-4567"' statement. You should determine whether or not a valid connection has been established before attempting to send data. This can be accomplished by testing the value of location 1656+SLOT. If the value is more than 127 then a valid carrier tone is being received. Otherwise communication has not been established either because the phone was not answered or the line was busy. If a valid carrier tone is not present after dialing then the Firmware hangs up the phone for you.

Sending data over the phone line is as simple as printing it. All characters output through a PRINT statement will be sent over the phone line. If you wish to display messages on the screen without sending them over the phone line then issue a 'PR#0' first. Later you can reconnect the Apple-Cat with a 'PR#<slot>'. After you have completed your call a 'PRINT CHR\$(26)' will hang up the phone.

Use the 'IN#' command to have the Apple-Cat monitor the phone line and pick it up if it rings. Check for a valid carrier by testing location 1656+SLOT as described above. If a valid carrier is present then data can be received through either the INPUT or GET statements.

It is good practice to include a 'POKE 2040+SLOT,0' statement at the beginning of the program. This insures that the Firmware will be initialized to COLD startup conditions when the first 'PR#' or 'IN#' is executed. Subsequent commands to turn on the Firmware input or output functions will result in a WARM startup. (See 'Firmware Programming Details' for further details concerning initialization.)

Issuing 'PR#' and 'IN#' commands from within BASIC programs must be done properly in order for them to work correctly. This means that if DOS is 'up' then the command must appear in a PRINT statement and have been preceded by a RETURN and Control-D. (See your DOS Manual for further details.)

Here is an example Applesoft program which automatically dials another computer system and sends it a short message:

```

100 SL=2                (APPLE-CAT IS IN SLOT #2)
110 D%=CHR$(4)         (CONTROL-D FOR DOS COMMANDS)
120 MD=1656+SL        (FIRMWARE "MODEM" REGISTER)
130 Q%=CHR$(17)       (CTRL-Q TO START DIALING)
140 Z%=CHR$(26)       (CTRL-Z TO HANGUP PHONE)
150 P#="123-4567"     (COMPUTER SYSTEM PHONE NO)
160 M1$="THIS IS THE APPLE-CAT CALLING"
170 M2$="TESTING..1..2..3..TESTING"
180 M3$="BYE-BYE"
190 POKE 2040+SL,0
200 PR#SL

210 PRINT Q%;
220 PRINT P#;
230 IF PEEK(MD)<128 THEN 900
240 PRINT
250 PRINT M1$
260 PRINT M2$
270 PRINT M3$
280 GOSUB 600:END

600 PRINT Z%          (HANGUP PHONE)
610 IN#0:PR#0        (RESTORE NORMAL I/O FUNCTIONS)
620 RETURN

900 GOSUB 600
910 PRINT"SYSTEM NOT AVAILABLE"
920 END

```

The following segment waits for the phone to ring and accepts a short message:

```

300 POKE 2040+SL,0    (INITIALIZE FIRMWARE)
305 IN#SL:PR#SL      (TURN ON FIRMWARE)
310 INPUT "WAITING FOR CALL";I$
320 IF PEEK(MD)<128 THEN 310    (LOOP IF HUNGUP)
330 PRINT"OK TO SEND A 3 LINE MESSAGE"
340 PRINT
350 FOR I=1 TO 3:PRINT"LINE #";I
360 INPUT M$(I):NEXT I
370 GOSUB 500
380 INPUT"IS THAT CORRECT (Y/N)";I$
390 IF I$<>"Y" THEN 350
400 PRINT"THANKS FOR CALLING"
410 PRINT Z%        (HANG UP)
420 HOME:FLASH
430 PRINT"MESSAGE RECEIVED":NORMAL
440 GOSUB 600:END

500 PRINT:PRINT"YOUR MESSAGE IS:"
510 FOR I=1 TO 3:PRINT M$(I)
520 PRINT:RETURN

```

EXAMPLE PROGRAM

The following Applesoft program illustrates various techniques for programming the Apple-Cat through the Firmware.

The purpose of the program is to automatically dialup and logon to the Telephone Software Connection. This software company distributes its programs right over the phone. Several programs are available at no charge (other than the telephone call) while others are paid for via charge card.

It is assumed that the Apple-Cat is located in Slot #2. If it is not then change Line #35. After typing in the program then save it on diskette with the name "CALL TSC:AC". Make sure that DOS is 'UP' and that there is an initialized diskette in your drive before using CALL TSC:AC.

(Program used with permission from TSC.)

```

0 REM CALL TSC:AC
1 REM CALL TSC (C) 1980 BY ED MAGNIN -ALL RIGHTS RESERVED-
2 REM DISTRIBUTED BY: TELEPHONE SOFTWARE CONNECTION (TM) INC.
  P.O. BOX 6548, TORRANCE, CA 90504
  (213)516-9430:VOICE, (213)516-9432:MODEM
4 REM 10/14/81
5 REM MODIFIED 7/26/82 FOR THE APPLE-CAT W/FIRMWARE BY BOB KOVACS
10 D%=CHR$(4) : N#=" TELEPHONE SOFTWARE CONNECTION (TM) INC."
  P2$="(213) 516-9432 24 HOUR MODEM" : DIM A(40)
15 TEXT : HOME : INVERSE : PRINT N# TAB(21-LEN(P2$))/2)
  P2$ TAB(40)" " : NORMAL : POKE 34,3
20 REM -----
25 REM INSERT CORRECT INFO:
30 REM APPLE-CAT SLOT NO.:
35 SLOT=2
40 REM YOUR FIRST NAME:
45 F$="JOHN"
50 REM YOUR LAST NAME:
55 L$="SMITH"
60 REM YOUR PASSWORD:
65 P#="PASSWORD"
70 REM AREA CODE CALLING FROM:
75 AC=213
80 REM -----
85 REM TSC PHONE NO.
90 PH$="1-213-516-9432"
95 IF AC=213 THEN PH$=RIGHT$(PH$,8)
100 REM INSTRUCTIONS?
110 IF P#="PASSWORD" THEN HOME : UTAB14 :
  PRINT"READ FOLLOWING DOCUMENTATION PRIOR TO RUNNING...";CHR$(7) :
  FOR I=1 TO 1500 : NEXT J : GOTO 2000
120 INPUT"DO YOU WANT INSTRUCTIONS? ";I$ : IF LEFT$(I$,1)="Y" THEN 2000
130 REM -----
200 REM DIAL TSC
210 HOME : INPUT"WANT TO CALL NOW? "; I$ : IF LEFT$(I$,1)<>"Y" THEN 280
220 D%=CHR$(4) : Q%=CHR$(17) : Z%=CHR$(26) :
  CHAR=1912 : SND=-16382 + 256 * SLOT : POKE 2040 + SLOT,0

```

```

230 HOME : PRINT D$*PR#0* : PRINT D$*IN#0* : PRINT D$*PR#*SLOT :
PRINT : POKE 1912 + SL,128 : REM POKE 129 FOR TONE DIALING
235 PRINT D$PH#
240 PRINT D$*PR#0* : PRINT D$*IN#*SLOT
250 IF PEEK (1656 + SLOT) > 127 THEN 300
260 PRINT D$*IN#0* : POKE -16368,0
270 PRINT : PRINT"NO ANSWER OR BUSY" : PRINT :
INPUT"TRY AGAIN? ";I$: IF LEFT$(I$,1)="Y" THEN 230
280 PRINT D$*PR#0* : PRINT D$*IN#0* : POKE 34,0
290 END
295 REM -----
300 REM LOG-ON
310 POKE 1912 + SLOT,128
320 O$="FIRST NAME? " : S$=F$ : GOSUB 1000 : IF NOT FD THEN 320
330 GOSUB 1100 : O$="LAST NAME? " : S$=L$ : GOSUB 1100
340 O$=F$+" "+L$ : INPUT "":I$ : INPUT "":I$ : I$=RIGHT$(I$,LEN(O$)) :
S$="N" : CT = (I$ = O$) : IF CT THEN S$="Y"
350 O$="CORRECT? " : GOSUB 1000 : GOSUB 1100
360 IF NOT CT THEN 320
370 O$="PASSWORD? " : S$=PW$ : GOSUB 1000 : IF FD THEN GOSUB 1100 : GOTO 380
375 O$="NUMBERS? " : GOSUB 1000 : IF NOT FD THEN 375
380 FLASH : PRINT : PRINT : PRINT : PRINT" *F*", TAKE OVER FROM HERE * :
NORMAL : POKE 1912 + SLOT,138
390 GET A$ : END
395 REM -----
1000 REM FIND QUESTION
FD=0 : N=0
1010 I$="" : N=N+1 : IF N>20 THEN 1040
1020 GET A$ : PRINT A$ : I$=I$+A$ : FD=(RIGHT$(I$,LEN(O$))=O$) :
IF FD THEN 1040
1030 IF A$=CHR$(13) THEN 1010
1035 GOTO 1020
1040 RETURN
1045 REM -----
1100 REM OUTPUT THRU MODEM
1110 S=LEN(S$) : INVERSE : FOR J=1 TO S : A(J)=ASC(MID$(S$,J,1)) : NEXT J
1120 FOR J=1 TO S : POKE CHAR,A(J) : CALL SND
1130 GET B$ : PRINT B$ : NEXT J
1140 NORMAL : POKE CHAR,13 : CALL SND : RETURN
1145 REM -----
2000 REM INSTRUCTIONS
2010 HOME : VTAB 8
2020 PRINT"THIS PROGRAM AUTOMATICALLY DIALS AND LOGS YOU ON TO:" : PRINT
2030 PRINT : PRINT"THE TELEPHONE SOFTWARE CONNECTION" : PRINT : PRINT
2040 PRINT"WHERE FULLY DOCUMENTED AND TESTED PROGRAMS FOR YOUR APPLE II
COMPUTER ARE AS CLOSE AS YOUR MODEM" : PRINT : PRINT
2050 PRINT : INPUT"HAVE YOU INSERTED YOUR PERSONAL INFO? ";I$
IF LEFT$(I$,1)<>"N" AND PW$<>"PASSWORD" THEN 200
2060 HOME : POKE 32,1 : HTAB 5 : INVERSE :
PRINT"INSERT APPROPRIATE INFORMATION: "CHR$(7) : NORMAL : LIST 30,78
2070 I$=" CALL TCC:AC" : PRINT "UNLOCK*I$ : PRINT "SAVE*I$ :
PRINT "LOCK*I$ : PRINT "RUN" : PRINT : INVERSE : HTAB 4 :
PRINT "RIDE THRU EACH LINE WITH CURSOR" : NORMAL : POKE32,0 : VTAB 6
2080 END
2090 REM -----

```

SPECIAL KEYS - The following keys have special meaning while in Terminal Mode:

←	Sends a Backspace character
Shift-Ctrl-M	Sends a RUBOUT character

DIALING (either Terminal Mode or from within a BASIC program) - The + symbol will search for another dialtone. The search can be aborted by typing a Ctrl-Z (otherwise the phone will be hungup if no dialtone is found after 20 seconds). The # symbol produces a 3 second pause before the next digit is dialed. The * and # symbols are meaningful only during tone dialing.

Use the mini-switches on the Apple-Cat board to setup the default dialing mode (either pulse or tone). You can switch back and forth between these modes with a P (for Pulse) or T (for Tone) character as part of the phone number.

It is possible to use the handset for voice communication after the Apple-Cat has dialed the number. While in Terminal Mode this is done either by using the Ctrl-A Ctrl-V command or by not hitting RETURN after typing in the phone number (use Ctrl-Z to hangup). From within a BASIC program just use a Linefeed character at the end of the phone number: PRINT CHR\$(17);"123-4567";CHR\$(19).

80 COLUMN BOARDS - The Firmware works with 80 column boards so you can display the maximum amount of information on the screen while using your Apple-Cat. Turn on both cards with the commands: PR#(80 column card slot) and IN#(Apple-Cat slot). The order of these commands is IMPORTANT and must not be reversed. This is because most 80 column cards effectively perform an IN# of their own.

The Firmware fully supports the VIDEOTERM and FULL-VIEW 80 cards. Other 80 column cards may work as well or with some limitations. The SMARTERM, for example, suffers from a missing cursor unless a small driver program is used to restore it. (ALS provides an application note which describes a method to restore the cursor on their SMARTERM card.)

LOWER-CASE DISPLAYS - The Firmware normally displays all incoming data as uppercase characters. If your Apple can display lowercase characters because it has either a lowercase display 'chip' (like the Paymar lowercase adapter) or because you are using an 80 column board then it possible to disable the upper case conversion by POKEing a 0 into location 1784+SLOT.

1-WIRE SHIFT KEY MOD - The Apple keyboard normally can input only uppercase characters. The popular '1-wire shift key modification' is frequently used in programs like word processors to provide both upper and lower case characters. This modification has been described in several Usergroup newsletters and consists of a single wire from the shift key switch to Pin 4 (PB2) of the Game I/O socket.

If your Apple has this modification then is possible to send lowercase characters by POKEing 128 into location 1528+SLOT.

POTENTIAL TIMING PROBLEMS - It has been observed that not all combinations of Firmware (actually 2716 EPROMs), Apple-Cats and Apple-II's work properly. This is apparently due to the effect of component tolerances on timing. The symptoms of a timing problem can range anywhere from the Firmware not seeming to work at all or seeming to work most of the time but occasionally getting 'hung'. If you find this sort of thing happening on your system, it might easily be fixed by removing the Integrated Circuit labeled Z8 (its a 74LS30) on the Apple-Cat and replacing it with a 7430. This standard TTL device has a higher speed than the LS equivalent and is readily available from local electronics supply stores. We've found that changing this one component has solved most of the 'timing' problems observed to date. (WARNING: This change may violate your Apple-Cat warranty.)

Section 7 - APPLE-CAT vs COMM CARD

The following information is provided in an attempt to highlight the major differences between the Apple-Cat and Apple's Communication Card. See Novation's 'Advanced Programming Information' #800155 and Motorola's 6850 Data sheet for further details.

The Apple-Cat is not directly compatible with either Apple's Communication Card or the Hayes Micromodem II. This incompatibility stems from the Apple-Cat's assignment of certain hardware registers. Even though the Firmware makes the Apple-Cat respond to the same command strings and perform the same functions as these other modem cards, the Firmware CANNOT change either the Apple-Cat's hardware register addresses or their function. This is dictated solely by the Apple-Cat's circuitry. Why the designer's of the Apple-Cat chose to deviate from the Apple standard remains unclear.

Many communications programs currently available for the Apple read and write directly to several hardware registers on the Communication's Card (or the Micromodem). This is done to achieve fast response time and allow the program to perform more complex functions than are available on the modem card alone. The two most important registers are located within the 6850 ACIA (which stands for Asynchronous Communications Interface Adapter). This device is used on both the Communications Card and the Micromodem but not the Apple-Cat. Their addresses and functions are summarized in the following table:

ADDRESS	READ	WRITE
\$C0nE (1)	6850 Status	6850 Control
\$C0nF (2)	6850 Data Recv'd	6850 Data Xmit'd

(1) also \$C0nC for the Comm Card and \$C0n6 for the Micromodem
 (2) also \$C0nD for the Comm Card and \$C0n7 for the Micromodem

NOTE: n in the addresses shown represent the card slot number plus 8.

CONVERSION TO APPLE-CAT - It is possible to convert programs which have been designed for use with Apple's Communication Card (or the Hayes Micromodem) to run on the Apple-Cat. First it is necessary to establish equivalences between the 6850 registers and those in the Apple-Cat. The table on the following page lists these equivalences (if they exist).

The next step involves program modification. This task could range anywhere from being relatively simple to next to impossible. It all depends on the program's design and security. 'Protected' disks, for example, will prevent access to the program. Programs written in BASIC are probably the easiest to change. Those written in machine code, on the other hand, must first be disassembled to determine where the registers are written to or read from. A very cooperative software company might even tell you what locations to modify!

HARDWARE REGISTER EQUIVALENCES

6850 REGISTER ADDRESSES (1)

APPLE-CAT REGISTER ADDRESSES

STATUS (read only):

C0nE, Bit 0: Receive Register Full (2)
 C0nE, Bit 1: Xmit Register Empty
 C0nE, Bit 2: Reset (No Carrier on MMII)
 C0nE, Bit 3: unused (Reset on MMII)
 C0nE, Bit 4: Framing Error
 C0nE, Bit 5: Overrun Error
 C0nE, Bit 6: Parity Error
 C0nE, Bit 7: IRQ

C0nF, Bit 3
 C0nF, Bit 4
 no equiv (C0nD, Bit 5)
 ----- (no equiv)
 C0nF, Bit 6
 C0nF, Bit 7
 C0nF, Bit 5
 C0nF, Bit 0 = Ring IRQ
 C0nF, Bit 1 = Recv IRQ
 C0nF, Bit 2 = Xmit IRQ

CONTROL (write only)

C0nE, Bit 0: Clock Divide Ratio
 C0nE, Bit 1: & Master ACIA reset

 C0nE, Bit 2: Word Select
 C0nE, Bit 3: (# Data Bits, # Stop
 C0nE, Bit 4: Bits & Parity)

 C0nE, Bit 5: Xmit Control
 C0nE, Bit 6: (IRQ & Break)

 C0nE, Bit 7: Recv IRQ

(no equiv)
 (no equiv)

 C0nA, Bits 0 & 1 = Word Length
 C0nA, Bits 2 & 3 = # Stop Bits
 C0nA, Bits 4,5 & 6 = Parity

 C0nC, Bits 2 & 3

 C0nC, Bits 0 & 1

DATA RECEIVED (read only)

C0nF: Bits 0 - 7

C0nB: Bits 0 - 7

DATA TRANSMITTED (write only)

C0nF: Bits 0 - 7

C0nE: Bits 0 - 7

NOTES:

(1) n = Slot number + 8

(2) This bit is reset by reading the Data Received Register on the 6850. On the Apple-Cat, however, the equivalent bit is reset by performing a 'write' to location \$C0nD.

Section 8 - FIRMWARE PROGRAMMING DETAILS

ENTRY POINTS - There are 4 separate entry points into the Firmware as shown in the following table:

COLD Input/Output	%Cn00 = 49152 + 256*SLOT
Alternate Warm Output	%Cn02 = 49154 + 256*SLOT
WARM Output	%Cn05 = 49157 + 256*SLOT
WARM Input	%Cn07 = 49159 + 256*SLOT

Following power-on, The COLD entry point is used by the FIRST character of an INPUT or PRINT statement following an IN#(slot) or PR#(slot) command, where (slot) is the Apple-Cat slot number. Subsequent inputs and outputs are automatically vectored through the warm entry points. An alternate entry point is provided to allow for special outputs from within a BASIC program. This is accomplished by Poking the character into location 1912 and calling this alternate output address. For example:

```
A$="ABCD" : SLOT=3 : SND=-16382 + 256*SLOT
FOR I=1 TO LEN(A$) : A=ASC(MID$(A$,I,1))
POKE 1912,A : CALL SND : NEXT I
```

INITIALIZATION - The first INPUT after the IN# (or PRINT after a PR#) performs a cold entry into the Firmware at %Cn00, where n is the slot number. Under certain circumstances this type of entry results in complete Firmware initialization. Among other things, initialization includes reading the mini-switches in the Apple-Cat board, turning off the modem transmitter and hanging up the phone line. Like most other firmware on peripheral cards, the input vector at \$38,\$39 (or output vector at \$36,\$37) is reset to the warm entry address. Thus subsequent inputs (or outputs) bypass the initialization sequence.

Whether or not the Firmware executes its initialization cycle depends on the RECBYT value (located at 2040+SLOT). If its value is anything other than the 2 valid ones (\$E4 or \$E5) then initialization is performed. Subsequent operations which turn the card on and off will not result in initialization. It can easily be forced, however, with a 'POKE 2040+SLOT,0' prior to the next IN# or PR#. Initialization can also be performed with the 'Ctrl-A Ctrl-I' Firmware command.

MEMORY USAGE - In addition to the memory space that is allocated to the Apple-Cat slot (%Cn00-%Cnff, \$C800-\$CFFF) the Firmware also uses a number of other locations in RAM.

Page Zero - Tone dialing uses \$00 to \$09 but those locations are not clobbered. The Monitor locations \$24 (CH), \$28 (BASL), and \$36 to \$39 (CSW/KSW) are used.

Page Two - Used as a FIFO buffer while in Dumb Terminal mode.

Common I/O Slot RAM -

TMP1	\$478 = 1144	Temporary location
TMP2	\$4F8 = 1272	
TMP3	\$578 = 1400	
??	\$5F8 = 1528	(unassigned)
??	\$678 = 1656	(unassigned)
NO	\$6F8 = 1784	Slot number constant
CHAR	\$778 = 1912	Data sent/received
CN	\$7F8 = 2040	Slot number constant

Slot dependent RAM -

OUT.L0	\$388 + CN = 1144 + N
OUT.HI	\$438 + CN = 1272 + N
XMTBYT	\$488 + CN = 1400 + N
SHIFT	\$538 + CN = 1528 + N
CRDLY	\$538 + CN = 1528 + N
MODEM	\$588 + CN = 1656 + N
LOCSE	\$638 + CN = 1784 + N
FLAGS	\$688 + CN = 1912 + N
RECBYT	\$738 + CN = 2040 + N

Current output - Lobyte
 Device address - HiByte
 Image of Xmtreg (\$C08D + N0)
 1-Wire Shiftkey mod (Bit 7 enables)
 No. of 10 Msec delays (127 max)
 See below for details
 LC -> UC (\$20 enables, \$00 disables)
 See below for details
 Image of Recreg (\$C089 + N0)
 (also used as initialization flag)

MODEM (\$588 + CN = 1656 + N)

B7	Hook	0=0n *	1=0ff	Hungup or picked up
B6	Carrier	0=0ff*	1=0n	Valid carrier indicator
B5	undefined			
B4	Buff	0=0ff*	1=0n	Printer buffer control
B3	undefined			
B2	Mode	0=Ans*	1=0ng	Controls xmit & rcv freq
B1	Voice	0=0n *	1=0ff	Communication or voice mode
B0	Baud (Sw 2)*	0=110	1=300	10 or 30 chars per second

FLAGS (\$688 + CN = 1912 + N)

B7	Echo	0=0n *	1=0ff	Full/Half duplex
B6	Dialing?	0=No *	1=Yes	Dialing in progress
B5	NRING (Sw 4)*	0=1st	1=9th	Number of rings before pickup
B4	LF1	0=0ff*	1=0n	Xmit linefeed & delay
B3	Term	0=0ff*	1=0n	Terminal mode control bit
B2	Tran	0=0ff*	1=0n	Transparent mode
B1	Keyboard Enable	0=No	1=Yes*	
B0	Dial Mode (Sw 1)*	0=Pulse	1=Tone	

* COLD ENTRY DEFAULTS

Name	Location	Slot						
		1	2	3	4	5	6	7
OUT.L0	1144+SLOT	1145	1146	1147	1148	1149	1150	1151
	\$478+SLOT	\$479	\$47A	\$47B	\$47C	\$47D	\$47E	\$47F
OUT.HI	1272+SLOT	1273	1274	1275	1276	1277	1278	1279
	\$4F8+SLOT	\$4F9	\$4FA	\$4FB	\$4FC	\$4FD	\$4FE	\$4FF
XMTBYT	1400+SLOT	1401	1402	1403	1404	1405	1406	1407
	\$578+SLOT	\$579	\$57A	\$57B	\$57C	\$57D	\$57E	\$57F
SHIFT/CRDLY	1528+SLOT	1529	1530	1531	1532	1533	1534	1535
	\$5F8+SLOT	\$5F9	\$5FA	\$5FB	\$5FC	\$5FD	\$5FE	\$5FF
MODEM	1656+SLOT	1657	1658	1659	1660	1661	1662	1663
	\$678+SLOT	\$679	\$67A	\$67B	\$67C	\$67D	\$67E	\$67F
LOCSE	1784+SLOT	1785	1786	1787	1788	1789	1790	1791
	\$6F8+SLOT	\$6F9	\$6FA	\$6FB	\$6FC	\$6FD	\$6FE	\$6FF
FLAGS	1912+SLOT	1913	1914	1915	1916	1917	1918	1919
	\$788+SLOT	\$789	\$78A	\$78B	\$78C	\$78D	\$78E	\$78F
RECBYT	2040+SLOT	2041	2042	2043	2044	2045	2046	2047
	\$7F8+SLOT	\$7F9	\$7FA	\$7FB	\$7FC	\$7FD	\$7FE	\$7FF