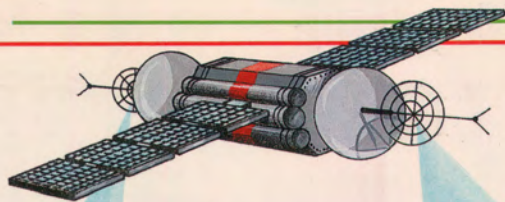


Oxxi



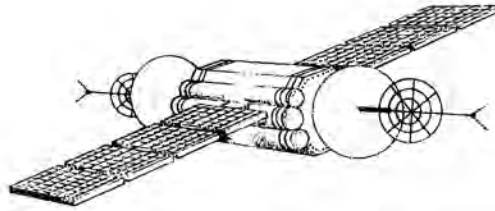
ATALKIIITM

DATA COMMUNICATIONS SYSTEM

- Communications Protocols
- Terminal Emulations
- Tektronix Graphics
- Script Language for Unattended Operation



Published by **Oxxi**
Developed by Felsina software



A-TALK IIITM
Advanced Communication
and Terminal Program
for the Amiga

User's Manual

Published by: Oxxi, Inc.
Developed by: Felsina Software
Manual written by: Piera Fumagalli

Copyright © 1986,1988 Felsina Software
All rights reserved worldwide.
Printed in the U.S.A.
Second Edition, January 1989.
ISBN 0-938385-06-2

The A-Talk III program described in this manual Copyright © 1986, 1988 Felsina Software, all rights reserved. Portions Copyright © 1986, 1987, Robert J. Mical. Portions of the glossary reprinted by permission, Black Box Corporation, Pittsburgh, PA © 1988, All Rights Reserved. Workbench and associated systems and software libraries Copyright © 1985, Commodore-Amiga, Inc. All rights reserved. You may not use, copy, modify, or transfer these programs or their documentation, or any copy except as expressly provided in the license agreement.

Oxxi, Inc.
P. O. Box 90309
Long Beach, CA 90809-0309
Telephone (213) 427-1227 FAX (213) 427-0971

OXXI, INC. PROGRAM LICENSE AGREEMENT

Carefully read all the terms and conditions of this agreement. If you do not agree to these terms and conditions, promptly return this package and the other components of this product to the Commodore dealer where the purchase was made.

LICENSE:

You have the non-exclusive right to use the enclosed programs only on a single computer. You may physically transfer the programs from one computer to another provided that the programs are used on only one computer at a time. You may not electronically transfer the programs from one computer to another over a network. You may not distribute copies of the programs or accompanying documentation to others. You may not modify or translate the programs or their documentation.

BACKUP AND TRANSFER:

You may duplicate the programs solely for backup purposes. You must reproduce and include the copyright notice on any backup copy. You may transfer and license the product to another party only if the other party agrees to the terms and conditions of this agreement and completes and returns a registration form to Oxxi, Inc.. If you transfer the programs you must at the same time transfer the documentation and the backup copy or copies, or transfer the documentation and destroy the backup copy or copies.

TERMS:

This license is effective until terminated. You may terminate it by destroying the programs and documentation and all copies thereof. This license will also terminate if you fail to comply with any term or condition of this agreement. You agree upon such termination to destroy all copies of the programs and documentation.

PROGRAM DISCLAIMER:

Oxxi, Inc. and Felsina Software make no warranties, either expressed or implied, with respect to the programs described herein, their quality, performance, merchantability, or fitness for any particular purpose. These programs are sold "as is". The entire risk as to their quality and performance is with the buyer (and not the creators of these programs Felsina Software; Oxxi, Inc.; their distributors and their retailers), who assumes the entire cost of all damages. In no event will Felsina Software or Oxxi, Inc. be liable for direct, indirect, incidental, or consequential damages. Some States or Countries do not allow the exclusion of limitation of implied warranties or liabilities for incidental or consequential damages, so the above limitation or exclusion may not apply. Further, Felsina Software and Oxxi, Inc. reserves the right to make changes from time to time in the

**DISKETTE
LIMITED
WARRANTY:**

content hereof without obligation of Felsina Software and Oxxi, Inc. to notify any person of such revision or changes.

Oxxi, Inc. warrants to the original licensee that the diskette on which the programs are recorded shall be free from defects in material and workmanship only for a period of ninety (90) days from the date of original purchase. If a defect covered by this warranty occurs during this 90-day warranty period, and it is returned to the dealer from whom it was purchased not later than five (5) days after the end of such 90-day period, the dealer shall, at the dealer's option, either repair or replace the diskette.

This warranty is in lieu of all other express or statutory warranties, and the duration of any implied warranty, including but not limited to the implied warranties, of merchantability and fitness for a particular purpose, is hereby limited to said ninety (90) day period. Oxxi, Inc.'s liability is limited solely to the repair or replacement of the defective product, in its sole discretion, and shall not in any event include damages for loss of use or loss of anticipated costs, expenses or damages, including without limitation any data or information which may be lost or rendered inaccurate, even if Oxxi, Inc. has been advised of the possibility of such damages.

Some states do not allow a limitation on how long an implied warranty lasts, so the above limitation may not apply to you. Some states do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Further, Oxxi, Inc. reserves the right to make changes from time to time in the content hereof without obligation of Oxxi, Inc. to notify any person of such revision or changes.

TRADEMARKS:

Aegis Draw is a trademark of Aegis Development, Inc.
A-Talk, A-Talk Plus and A-Talk III are trademarks of Felsina Software.
American People/Link is a trademark of American Home Network Inc.
AMIGA, Amiga Kickstart and Amiga Workbench are trademarks of Commodore-Amiga, Inc.
CompuServe is a registered trademark of CompuServe, Inc.
Courier HST is a trademark of U.S. Robotics.
Deluxe Paint II is a trademark of Electronic Arts, Inc.
Hayes is a registered trademark of Hayes Microcomputer Products, Inc.
IBM is a registered trademark of International Business Machines Corporation.
Kermit is a trademark of Henson Associates, Inc.
Macintosh is a trademark of Apple Computer, Inc.
NewsNet is a registered service mark of NewsNet, Inc.
PC-TALK III is a trademark of The Headland Press, Inc.
Tektronix is a registered trademark of Tektronix, Inc.
Telenet is a registered trademark of GTE Telenet.
UNIX is a registered trademark of AT&T.

ACKNOWLEDGEMENTS:

We would like to thank the following people for their contributions:

- **Rodger Anderson** for the callback script.
- **The Black Box Corporation** for portions of the glossary.
- **The Columbia University Center for Computing Activities (CUCCA)**, **Bill Catchings** and **Frank da Cruz**, for the Kermit protocol.
- **Chuck Forsberg** of **Omen Technologies** for the ZMODEM code.
- **William S. Hawes**, author of **ARexx**, and **Willy Langeveld** of **SLAC**, for **ARexx** support.
- **Dave Haynie**, of **Commodore-Amiga**, for hardware support.
- **Dan J. James** for the **WXMODEM** code.
- **Bjorn Knutsson** for **PAL** support.
- **James H. H. Lampert** for **Talk2Three** and his work on the mini-board script.
- **David M. Lucas** for **SpeechToy**.
- **Jim Mackraz**, of **Commodore-Amiga**, for software support.
- **Robert J. Mical** for the file and color requesters.
- **People/Link** for free access to their system for testing purposes.
- **Caroline Scheppner**, of **Commodore-Amiga**, for **IFF** support.

Contents

OXXI, INC. PROGRAM LICENSE AGREEMENT	iii
TRADEMARKS:	v
ACKNOWLEDGEMENTS:.....	v
ABOUT THIS MANUAL	xi
If you need Technical Support:	xii
1. WELCOME TO A-TALK III	1-1
TERMINAL EMULATIONS:	1-1
TEKTRONIX 4010/4014:	1-1
PROTOCOLS:	1-2
COMMUNICATIONS:.....	1-2
SCRIPT LANGUAGE:.....	1-2
SPEECH CAPABILITIES:.....	1-3
2. UP AND RUNNING IN THIRTY MINUTES	2-1
3. TOOLS	3-1
File Requesters	3-1
TalkToy	3-3
4. THE PROJECT MENU	4-1
Screen	4-1
New CLI	4-2
Colors	4-2
Quick Menu.....	4-3
Fun Keys	4-3
Icons	4-4
Printer	4-4
Voice	4-5
Start/Stop Timer.....	4-5
List Disks	4-6
Save.....	4-6
Quit	4-6
Exit.....	4-6
5. THE SETTINGS MENU	5-1
Baud Rate.....	5-1
Parity.....	5-2
Stop Bits.....	5-2
No. Bits.....	5-2
Handshake	5-3
Duplex	5-3

6. THE PHONE MENU	6-1
Modem Type.....	6-1
Line Type.....	6-1
Disconnect	6-1
Redial.....	6-2
Dial.....	6-2
Script.....	6-2
ARexx	6-2
Hangup.....	6-3
Send Break	6-3
Phone Book	6-3
The Function Keys.....	6-8
7. THE FILE MENU	7-1
Capture Options	
ASCII Send Options	7-1
Sending Mail	7-2
Line Delay	7-2
Character Delay.....	7-2
Receiving Mail or Capturing a Terminal Session.....	7-4
Creating a Login Script with Script Tutor	7-5
8. THE TRANSFER MENU	8-1
Protocol.....	8-1
WXMODEM	8-5
YMODEM-g	8-5
CRC	8-5
Get DIR.....	8-6
Type.....	8-7
Timeouts.....	8-7
Auto Chop.....	8-7
Examples of Protocol Transfers	8-7
Receiving a file with YMODEM Batch	8-12
Sending Files with ZMODEM.....	8-13
Using ZMODEM Resume.....	8-14
Using ZMODEM Remote.....	8-14
Sending Files with YMODEM and YMODEM Batch.....	8-15
Receiving a File with Kermit	8-16
Receiving a File with Kermit Server Mode	8-16
Sending a File with Kermit	8-17
9. THE TERMINAL MENU	9-1
Emulator.....	9-1
Auto Wrap	9-5
Font	9-5
Bell.....	9-5
Rows.....	9-5
Columns	9-6
Colors	9-6

10. THE TEKTRONIX MENU	10-1
Options	10-2
Display	10-3
Save IFF	10-4
Save Draw	10-4
Save Plot	10-4
Print	10-5
Zoom In	10-5
Replay	10-5
11. THE SCRIPT LANGUAGE	11-1
Writing Your Own Script	11-1
The Script Commands	11-2
ASCIISEND	11-2
ASK	11-2
AT	11-3
BAUD	11-3
BELL	11-4
BREAK	11-4
CD	11-4
CHAIN	11-5
CLEAR	11-5
CLOSE	11-6
COLUMNS	11-6
CRC	11-6
DELAY	11-6
DEPTH	11-7
DIAL	11-7
ECHO	11-8
EMULATOR	11-8
EXIT	11-8
FINISH	11-9
FONT	11-9
GETLINE	11-9
GOTO	11-9
HANGUP	11-10
HELP	11-10
ICONS	11-10
IF	11-10
IF TIMEOUT	11-11
INPUT	11-12
KGET	11-12
MODE	11-12
ON	11-12
OPEN	11-14
PRINTER	11-14
PROTOCOL	11-14
RECEIVE	11-14

REDIAL	11-15
REPLY	11-15
REWIND	11-15
ROWS	11-16
RUN	11-16
RX	11-16
SAY	11-17
SCREEN	11-17
SEND	11-17
SET	11-17
START	11-18
STOP	11-18
STRING	11-18
SUSPEND	11-19
TIMEOUT	11-19
VOICE	11-19
WAIT	11-20
WRAP	11-21
XON	11-21
A-Talk and A-Talk Plus to A-Talk III Script Conversion ..	11-21
SAMPLE SCRIPTS	11-23
Scripts Provided with the Phone Menu	11-23
Examples of More Complex Scripts	11-25
Appendix A: REQUIREMENTS	A-1
Appendix B: INSTALLATION OF A-TALK III	B-1
Contents of Disk 1 (named AT3)	B-1
Contents of Disk 2 (named ATInstall)	B-1
Hard Disk Installation	B-1
Floppy Disk Installation for Foreign Countries	B-3
Multiple Serial, Multiple Printer and Script Support	B-3
Appendix C: A-TALK AND AREXX	C-1
Return Codes	C-3
Quoted Strings	C-3
Appendix D: THE AMIGA KEYS AND SPECIAL FUNCTION KEYS	D-1
Special Function Keys	D-2
Appendix E: ASCII CHARACTER SET	E-1
International Character Sets	E-4
GLOSSARY	GL-1
INDEX	IN-1

ABOUT THIS MANUAL

This manual has been set up as a tutorial. If you read it from beginning to end, you will find that the chapters are arranged sequentially. First, make sure that your hardware and modem are set up properly; then read Chapters 1, **WELCOME TO A-TALK III** and 2, **UP AND RUNNING IN THIRTY MINUTES**. This will enable you to log onto an electronic bulletin board.

Chapter 3, **TOOLS** can be read to learn how to use the file requester and TalkToy.

More adventurous users who want to dial other host computers should read Chapters 4, **THE PROJECT MENU** and 5, **THE SETTINGS MENU**. The **Voice** option is described in Chapters 3, and 4. The **X-on/X-off** handshake can be found in Chapter 5. From there you go to Chapter 6, **THE PHONE MENU**, where you will be able to learn how to use the tested login scripts for sample hosts.

Once you are used to logging onto bulletin boards and sending mail, you will find it easier and cheaper to write your letters off-line and then send them as a file. Chapter 7, **THE FILE MENU** will help you figure out how to send and receive electronic mail with the **ASCII Send and Capture** features. The capture feature is useful if you want a record of your terminal session--a captured file can also be printed after you have logged off. Under 7, **THE FILE MENU** you will find the **Script Tutor**, which will enable you to automatically create your own login script as you login for the first time. This simple login script can later be modified and enhanced by more complex commands in the script language.

Files can also be transmitted with "protocols". When the phone lines are used to connect computers, noise and static on the lines could upset the reliable transmission of information. For this reason, "protocols" have been established to regulate the flow of information from one computer to another, and to ensure that transmissions are error-free. Under the **Protocol** option of **THE TRANSFER MENU** (Chapter 8) you will find: XMODEM, YMODEM, YMODEM Batch, ZMODEM, Kermit, WXMODEM and YMODEM-g. The protocols can be checked with Checksum, CRC-16 or CRC-32; autochop is available for binary files.

In Chapter 9, **THE TERMINAL MENU** you will find everything you need to emulate these terminals: VT100, VT52, H19, TTY, ANSI, TALK (sometimes called "CB" or "Chat" mode), and Tektronix

**If you need
Technical
Support:**

4010/4014. If you select the Tektronix option, you need to turn to Chapter 10, **THE TEKTRONIX MENU**.

In Chapter 11, **THE SCRIPT LANGUAGE**, you learn how to use A-Talk III's extensive script language, which allows you to set up totally automated operations.

Really adventurous users may find some interesting items in the Appendices. Here you will find **Appendix A: REQUIREMENTS** (for special setups or making your own cables), **Appendix B: INSTALLATION** (for hard disks or foreign languages), **Appendix C: A-TALK AND AREXX**, **Appendix D: THE AMIGA KEYS and SPECIAL FUNCTION KEYS**, and **Appendix E: ASCII CHARACTER SET**. If you are familiar with other computer communications programs, you may feel you don't need to read this manual from cover to cover. Answers to your questions, such as how to use the script language, can be found by checking the table of contents and/or the index.

We at Oxix don't expect you to have to spend hours trying to find the answers to technical questions, or give up entirely; to that end, we provide technical support for all our products. For technical support, call us between the hours of 10 am and 4 pm, Pacific Time at (213) 427-1227. To take full advantage of technical support, it is best to carefully document the problem before calling. Often solving a technical problem involves eliminating sources of errors. The more the problem is defined, the faster Technical Support will be able to arrive at a solution.

1. WELCOME TO A-TALK III

TERMINAL EMULATIONS:

In the April 1988 edition of *AmigaWorld*, Guy Wright and Bob Ryan wrote that "A-Talk from Oxix sports dozens of new options and is one of the most powerful telecommunications packages on the market. . . . A good package with few flaws, A-Talk [III] is probably the most complete telecommunications program available".

In the same article, *AmigaWorld* also says, "Terminal emulations are A-Talk [III]'s forte." Terminal emulations permit you to use your Amiga exactly as you would the fancy terminals at school or at work. A-Talk III provides you with these emulations:

- VT100
- VT52
- H19
- TTY
- ANSI
- TALK
- Tektronix® 4010/4014

You will even be able to create a small VT100 window by utilizing the small font (132 characters per line). This will allow you to multi-task on other parts of the screen while seeing the full VT100 emulation. Other available fonts include: VT100/H19 graphics fonts, and the IBM PC ANSI font. The emulation features include user-definable function keys, auto-wrap, 2/4/8 ANSI color support, and 24/48 lines. The VT100 emulator has been tested with the trickiest editors, including Emacs, Vi and EDT.

TEKTRONIX 4010/4014:

You won't be disappointed with our full Tektronix 4010/4014 emulator, which includes these features:

- Interlace and non-interlace modes
- Horizontal screen size of 640 to 744 pixels
- Vertical screen size of 400 to 480 pixels (interlace mode) or 200 to 240 pixels (non-interlace mode) or 592 pixels (PAL)
- Four character sizes (Micro, Small, Medium, Large)
- Stores screens in IFE, PLOT-10 or Aegis Draw™ formats
- Printer support
- Pan and zoom options to scroll over the 1024 x 786 pixel Tektronix screen
- ALPHA/GRAPH/GIN standard modes

PROTOCOLS:

- Enhanced graphics plot point and incremental plot modes
- Crosshair support
- VT640 escape sequences

As anyone knows, phone lines are subject to static and noise--no problem in most voice connections. But when the phone lines are used to connect computers, all that noise could upset the reliable transmission of information. For this reason, "protocols" have been established to regulate the flow of information from one computer to another, and to ensure that transmissions are error-free. A-Talk III provides you with these protocols:

- XMODEM
- YMODEM
- YMODEM Batch
- ZMODEM
- Kermit
- WXMODEM
- YMODEM-g
- Kermit Server Mode
- ASCII File Transfer

With these protocols you get the Checksum, CRC-16 or CRC-32 methods for error checking.

COMMUNICATIONS:

Other communications features include:

- Auto-chop for Amiga binary files
- CR/LF strip or translation
- X-on/X-off and RTS/CTS
- 300 to 57,600 bauds
- Half and full duplex
- ASCII capture
- Resizable and scrollable buffer
- Support for the Amiga "Clipboard"
- Concurrent printer support
- Character and line delays
- Selectable redial time and number of redials
- Relaxed timeouts for public networks
- International fonts
- Support for A500/A2000 extra keys and international keyboards
- Support for multiple serial and printer ports
- Clock/timer

SCRIPT LANGUAGE:

A login script is a set of commands that allows you to automatically access host computers, networks and databases. Beginners

SPEECH CAPABILITIES:

may use any of the tested login scripts included in A-Talk III (for BIX, CompuServe, DIALOG, Dow Jones, MCI, NewsNet, Official Airline Guides and People/Link); or they may use the "script tutor" to set up their own login script for a new host. With A-Talk III you also get a powerful script language with over fifty complex commands for advanced programmers. There are even commands for unattended operations, allowing you to leave your Amiga to do its work alone, for example, during off hours when phone rates are cheaper. The script language also includes:

- Interactive commands
- Flow control
- Pattern matching
- Multiple ON statements
- Timeouts
- File transfers
- User-defined menus
- Ability to leave out passwords for privacy
- Voice command
- ARexx support

A-Talk III uses the unique speech capabilities of the Amiga to permit you to have transmitted data read aloud in a variety of ways:

- Unfiltered
- Filtered (letters and numbers only)
- Reading of the voice buffer
- Requests (messages from requesters only)

With the special program, **TalkToy**, you can change the sample frequency, pitch, rate, volume, male/female and type of the voice to your needs and tastes.

2. UP AND RUNNING IN THIRTY MINUTES

IMPORTANT NOTICE:

WARNING: Before reading further, make a backup copy of A-Talk III to protect your investment. To do this from the Workbench, drag the A-Talk III icon over the icon for the disk that will receive the backup copy. Or you may use the CLI to copy A-Talk III from df0: to a destination disk in either df1 or df2:

```
1 > Diskcopy df0: to df1:
    (or Diskcopy df0: to df2:)
```

Please take a few minutes to complete the registration form, and then mail it to Oxxi, Inc. Registration is required for technical support, and will also allow us to keep you informed about updates for A-Talk III, and other useful information.

In order to use A-Talk III, you will need an Amiga model 500, 1000, or 2000 and a 1.2, or later, version of Workbench. You will also need 1) a modem and a serial cable or 2) a null modem cable. A null modem cable is used if you are connecting your Amiga to another personal computer in the same room.

If you are using a modem, read the instructions that come with it. Buy the recommended cable from your Amiga dealer. Your modem instruction manual will also tell you how to set the switches inside the modem.

Now you are ready to insert your A-Talk III diskette in one of the drives. The A-Talk III disk icon will appear on the screen. Double-click on it. A window will appear with an A-Talk III program icon inside.

Check to see if there is also a "Click Me First For Updates" icon on your A-Talk III diskette. In this file you will find news and updates for your particular version of A-Talk III. Later you can print it from the CLI by typing the following:

```
1 > COPY "For Updates" to PRT:
```

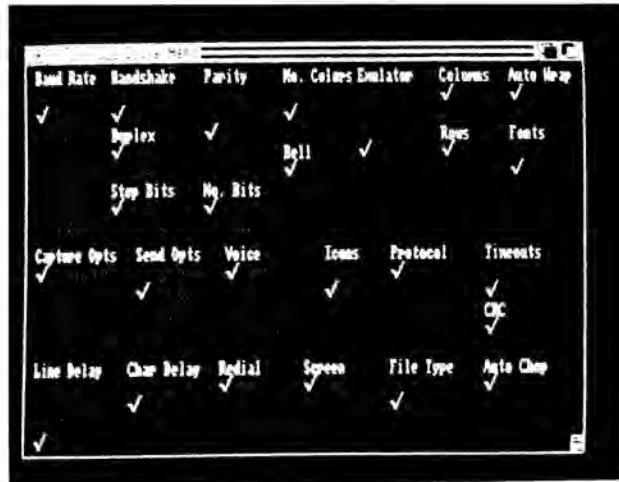
Now move your mouse to the window and double-click on the A-Talk III icon to load the program.

You may also start A-Talk III from the CLI by typing *run atalk3*.

To activate the menus, press the right button on your mouse. A horizontal menu bar will appear with these options:

Project Settings Phone File Transfer Terminal

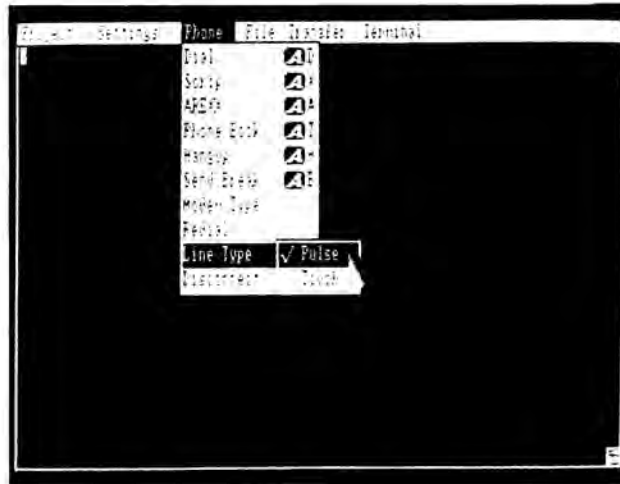
Move the cursor to the word **PROJECT**. Under the **PROJECT** menu you should find the option **Quick Menu**. Select it. This menu provides you with a fast way to change options that normally appear on the main menu:



In this example we will be connecting to an electronic bulletin board system (BBS). Many Amiga user groups have local bulletin boards that will keep you up-to-date on new Amiga software and equipment. It's also a great way to make friends who can help you learn about your Amiga. If you don't have any telephone numbers for local electronic bulletin boards, ask your Amiga dealer how to get in contact with an Amiga bulletin board or user group. Electronic bulletin boards are like clubs--there are many different varieties for different types of people. You may be interested in the more specialized bulletin boards, such as those that list real estate prices, etc.

As you can see, the default parameters (stop bits, parity, etc.) are already set up for communication with electronic bulletin boards. We are also assuming that you are using a Hayes compatible modem. The only option you might need to change is **Baud Rate**, which depends on your modem. For example, if you have a 2400 baud modem, click on the box next to 2400 under **Baud Rate**. Your selection will be indicated with a check mark. Exit from the **Quick Menu** by clicking on the closing gadget at the upper left-hand corner.

Another option you might need to change is the **Line Type** under the **PHONE** menu. Move the cursor to the **PHONE** menu.



Under the menu you will find the options **Pulse** and **Touch** after **Line Type**. If you have a rotary dial line, you must select **Pulse** by releasing the right-hand button on the mouse. The **Touch** selection is for the newer telephone lines set up for touch tone dialing. Remember that we are referring to the *telephone line*, not the telephone. Many touch tone phones have a rotary dial line option, so they can be used with rotary dial lines. Touch-tone lines can accept either the pulse or tone option.

Under the **PHONE** menu you will also find the **Dial** option. Choose **Dial** by releasing the the right-hand button on the mouse. A requester will pop up asking for a telephone number.



Type in the telephone number of your local electronic bulletin board. Check your modem instruction manual to see if you will have to include extra characters (such as pauses for long distance access codes) in order to dial properly. Click on **ok** or press **RETURN** when you are ready to dial.

With any luck, you have reached your electronic bulletin board. If the line is busy or for any other reason dialing cannot be completed, you will get a timeout or error message. Many bulletin boards are often busy, and it may take you a long time to make contact. Dial again, or set up A-Talk III to redial continuously. Do this by finding the **Redial** option under the **PHONE** menu. Select **continuously**. Then go back to the **Dial** option and start the process again.

When you have connected to your bulletin board, you will probably see the title of the bulletin board or some other introduction scroll up your screen. Then you will get a prompt asking for your name and possibly your password. Type these in the proper places, making up a password if you have to. Then you will usually be granted access to the bulletin board.

Browse through the various sections of the bulletin board. Most bulletin boards are self-explanatory, with menus, so this should be easy. Just make your selections and watch what comes up on your screen.

Most bulletin boards have sections for mail. You might want to write a letter introducing yourself to the Sysop, or system operator, who runs the bulletin board. Just find the mail section and read the instructions on how to send mail. When writing your letter, you may make changes on each line by backspacing over characters. But once you press the carriage return, you cannot go back to the previous line. Try to make each line no longer than eighty characters. Then send your letter by following the menu instructions of your bulletin board.

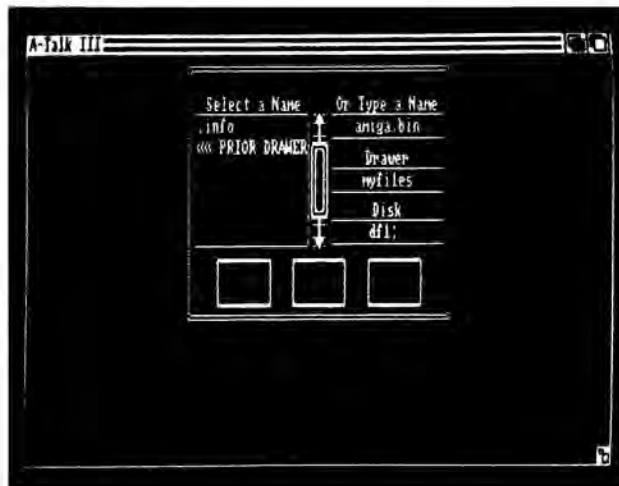
When you have finished your session, logoff. Again, you will do this by following the particular instructions given with each bulletin board. Once you have logged off, you must remember to hang up, so you won't be charged for extra connect time. Do this by selecting the **Hangup** command under the A-Talk III **PHONE** menu.

3. TOOLS

File Requesters

This chapter describes various A-Talk gadgets and requesters, and related Workbench tools.

Whenever you need to select or transfer a file, the file requester will pop up. It looks like this:



With the file requester, you can select a name from the list by clicking on it, or you can type the name of the file you want in the string gadget. Use the **CANCEL** gadget if you decide you don't want to select a file after all.

You might notice that the ZZZ-cloud wait pointer looks different from the ones you are used to on the Amiga. The small tab at the top left allows you to continue pointing and making selections even though the ZZZ-cloud is displayed.

Select a Name

When you call up the file requester, the file names will start appearing one by one in the **Select a Name** gadget. These files are listed in alphabetical order, and you can select one even before all the names have been added to the list. If you see the file you want, you can stop the building of the file name list by clicking on a spot inside the requester but outside all the gadgets (for example, in the space between the **OK!** and **CANCEL** gadgets). Then select the file by pointing at the name and clicking on the left mouse button.

The name will be highlighted and will appear in the box to the right of the **Select a Name** gadgets. To restart the building of the file name list, click on either the **Drawer** or **Disk** string gadget and press the carriage return.

Once you have selected a file name, you will have to click on the **OK!** gadget to confirm your choice. Another way to select a name and confirm your choice is by double-clicking on the file name in the same way you double-click on a Workbench icon.

To the right of the box with the file names you will see a slider gadget with a knob that you can move to scroll through the file names. The knob will be different sizes depending on the number of names in the file list. If the knob is small, this means you can see only a small number of total file names. If the knob is so big it fills up the slider, you won't be able to move it because all the names on the list will already be displayed.

To view the entire list of file names, "grab" the slider knob and move it up and down by pointing at it, pressing and holding the left mouse button. Move the pointer to scroll through the list of file names.

To page through the list of file names, click in the area of the slider outside the knob. The up and down gadgets let you adjust the list of names one at a time.

At the bottom of the list of file names, there are special entries which allow you to move around through the drawers of a disk much as you move around through drawers on the Workbench display. The entries that start with the ">>>>" characters allow you to open a drawer and examine the names of the files in that drawer. If you select, for example, ">>>>download", a new list of file names from the ">>>>download" drawer will appear after a short pause. In the ">>>>download" drawer you will also find a special entry "<<<<PRIOR DRAWER". By selecting this entry, you will go back to the drawer that contained the ">>>>download" drawer.

The Next Disk Gadget

To view the files of a different disk, select the **Next Disk** gadget. The name of the next disk will be displayed in the **Disk** string gadget, and the disk is then searched. If this isn't the disk you wanted, you can press **Next Disk** again even though you may be seeing the wait pointer. You can press **Next Disk** as many times as you want until you see the names of the disk you want in the **Disk** string gadget.

Type a Name

If you don't want to use the **Select a Name** gadgets, just type the file name into the **Or Type a Name** string gadget. Confirm your file selection by clicking on the **OK!** gadget or by simply pressing the carriage return. You can start typing the file name as soon as the requester appears--you don't have to select the string gadget

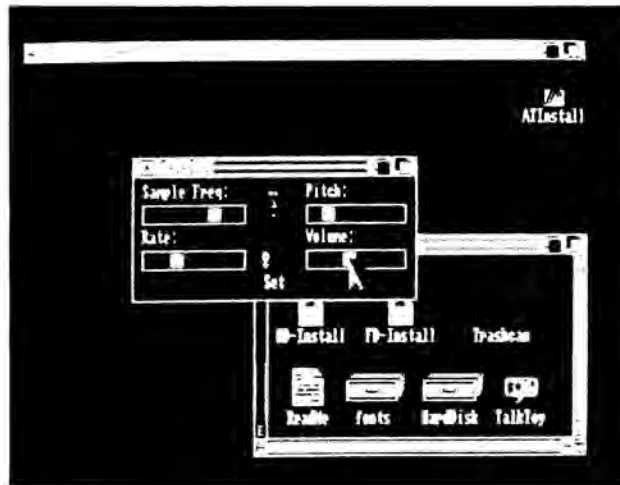
TalkToy

before typing, and you don't have to wait for the wait pointer to change before pressing the carriage return.

Change drawers by typing the name of the drawer you want in the **Drawer** string gadget. Select the gadget, type in the name and press the carriage return.

To change disks, type the name of the disk you want directly into the **Disk** string gadget.

This is a separate program provided with A-Talk III and included on the second diskette. It allows you to program the Amiga voice to your specifications. Once you click on the TalkToy icon, the program will pop up.



As you can see, you have six options which you can set: sample frequency, type of voice (robot or human), pitch, rate, male or female, and volume.

Sample Frequency is a complex concept. In order to represent the continuous quality of sound by digital values in the Amiga, *sampling* is used to convert continuous values into a series of numbers which can be easily stored by a computer. By noting, or sampling, the values present at various points in an interval of sound and converting these values into numbers, a numerical representation of sound is created for use by a computer. *Sampling Frequency* refers to how often the sound values are noted in a sound cycle. Different effects can be produced by moving the **Sample Freq** slide gadget between the extremes of 5,000 and 28,000. The default value is 22,200.

The **type of voice** can be altered by clicking on the little face in the middle of the gadget. This works like a toggle, changing from robot (monotone) voice to human-sounding voice.

Pitch refers to the pitch, or frequency range of the voice. The range of the Pitch is from 65 to 320. Move the sliding gadget to adjust this range.

Rate refers to the number of words per minute. You may vary this from 40 to 400 words per minute. The default is 150.

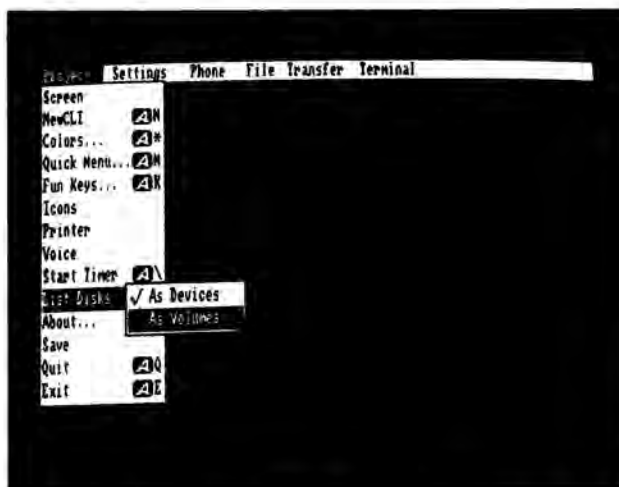
The next gadget with the male and female symbols can be used to make your computer speak like a man (heavier and generally lower-pitched voice) or a woman (lighter and generally higher-pitched voice). Click on the symbol of your choice.

The **Volume** device refers to how loud the sound is. It ranges from 0 to 64. The default is set at 64.

In order for your changes to take effect, you must have A-Talk III running at the same time. When you have finished programming the voice to your specifications, click on **Set** at the bottom of the TalkToy requester.

4. THE PROJECT MENU

Move the cursor to the word **PROJECT** on the horizontal bar menu. The following menu will appear:



Screen

You have two options under **Screen**: Workbench or Full. By default, your Amiga will be set in the Workbench screen mode. This means that you will have a four-color screen with a maximum of 77 characters filling the screen horizontally and 23 rows filling the screen vertically. This screen size is smaller than that of a normal terminal because the menu bar and gadgets are visible at all times. If you want to work on other programs while your file is being transmitted, you may resize the window with the dragging gadget so only the transmission requester is visible.

You may also want to resize the window while you are reading mail or transmitting data. From the moment you reduce the window size, the data displayed will wrap to fit the new window. Any data transmitted previously will be cut off at your new right margin and you will not be able to see it even if you enlarge the window to its original size. This data is lost unless you have been using the **FILE Capture** option, in which case it is stored under the name you provided.

To simulate a full screen terminal, select the **Full Screen** option. The Full screen is in two, four or eight colors and defaults to an 80x24 borderless window without sizing gadgets. This window

cannot be resized. You can drag the screen down and the Workbench screen will appear behind it if it is open.

To view the menu bar while using the Full screen, press the right button on your mouse. The menu options will appear when the cursor is moved over the menu titles.

Since all full screen terminal emulators need at least an 80x24 window, the Full screen must be used with all emulators except TTY and TALK in order to obtain full screen emulation unless you are using Workbench interlace mode with the small font (see Chapter 9, THE TERMINAL MENU, *Full VT100 Emulation in a Small Workbench Window*).

New CLI

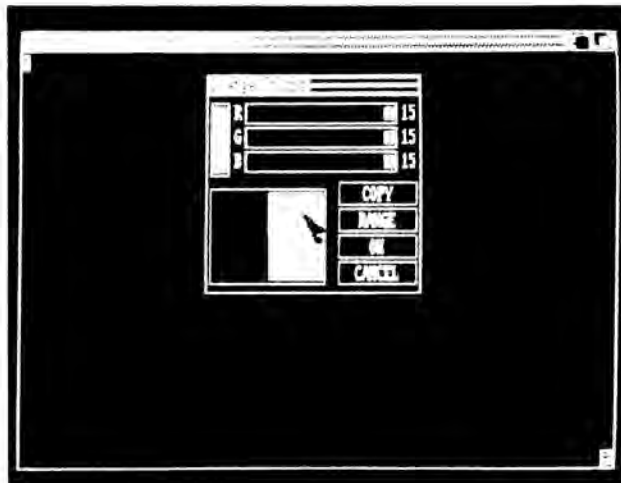
You may invoke a separate CLI (Command Line Interface) by selecting this option. The CLI window will appear on top of the Workbench screen. If you are using the Full screen, the Workbench screen will be brought to the front. You can go back to the A-Talk III custom screen by clicking on the Workbench Back Gadget. When using the Full screen, you can go back to Workbench by clicking on the A-Talk III Back Gadget. You can close the CLI window by typing:

```
1 > endcli
```

If you started A-Talk III from the CLI your new CLI will inherit your PATH and ASSIGNS. If you started A-Talk III from Workbench, no PATH and no ASSIGNS will be inherited.

Colors

This selection creates a color requester. It has gadgets that allow you to change the colors of the A-Talk Full or Workbench screen. With the color requester, you can modify the colors using either an RGB or an HSL technique.



The RGB technique involves three proportional gadgets, one each for Red, Green and Blue. By adjusting one of these gadgets, you adjust the amount of the associated color component in the final color. Some people feel that this is the most intuitive way to change colors on the Amiga, as this is how colors are represented internally by the Amiga.

Other people, however, prefer the HSL technique, which allows you to adjust the Hue, Saturation and Luminance of the final color. This technique is based on color theory; these are the types of controls found on color television sets.

No matter which technique you specify, you can always switch to the other technique by clicking on the RGB/HSL characters that appear to the left of the proportional gadgets.

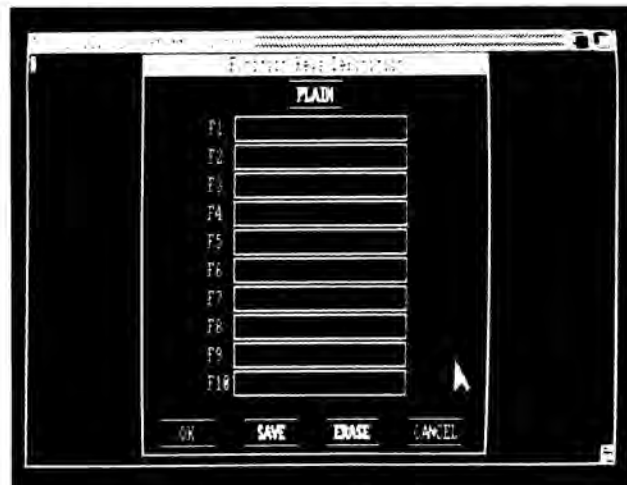
Quick Menu

If you select this option, you will get the **Quick Menu** requester. This provides you with a fast way to change options that normally appear on the main menu. On a single screen, the **Quick Menu** will allow you to change or set the **Screen**, **Icons**, and **Voice** options under the **Project Menu**; the **Baud Rate**, **Parity**, **Stop Bits**, **No. of Bits**, **Handshake** and **Duplex** options under the **Settings Menu**; the **Redial** option under the **Phone Menu**; **ASCII Capture Options**, **ASCII Send Options**, **Line Delay** and **Character Delay** options under the **File Menu**; the **Protocol**, **File Type**, **CRC**, **Timeout** and **Auto Chop** options under the **Transfer Menu**; and the **Emulator**, **Auto Wrap**, **Font**, **Bell**, **Rows**, **Columns** and **No. Colors** options under the **Terminal Menu**.

Click to make your choice. It will be indicated with a check mark. Close the **Quick Menu** by clicking on the closing gadget in the upper left-hand corner.

Fun Keys

Program the function keys with this option. When you make this selection, a requester will pop up. It looks like this:



The **Fun Keys** requester enables you to define twenty function keys: F1 through F10 plus the shifted keys F1 through F10. Click on the gadget at the top of the requester to toggle between **PLAIN** and **SHIFT**. A maximum of 30 characters per key is allowed. A-Talk III has already defined the function keys for sample hosts. If you want to define the function keys for a new host, use normal ASCII characters (letters, numbers and punctuation) and the following special, or control characters, where a caret (^) is used to indicate them:

^M	sends a carriage return
^J	sends a line feed
^I	sends a horizontal tab
^H	sends a backspace
^[sends an escape
^Q	sends an X-on
^S	sends an X-off

For a more complete list of all the control characters and their corresponding symbols, check the ASCII table in the appendix.

Once you have defined the function keys, click on the **SAVE** gadget to save your definitions. **OK** will take you back to the main menus and make your newly defined function keys available. **ERASE** will erase your definitions so you can start over. **CANCEL** will not only erase any changes you have made, but will also take you immediately back to the main menus.

Icons

If you select **Icons On**, A-Talk III will automatically create icons for all files received. You have the option to turn this function off.

Printer

Select **On** under **Printer** to print everything that comes in through the serial port. This will slow your session down to the speed of the printer. Select **Off** when you no longer want a printout.

A-Talk III will format the output for the printer according to the printer type and printer options (left margin, right margin and page length) selected in Preferences on the Workbench diskette. Moreover, A-Talk III will format your output with 5 blank header lines at the top of each page and 5 blank footer lines at the bottom of each page. Make sure to align your first page properly.

Getting a Cheap Printout

Using the printer option during your terminal session will be expensive because your session will slow to the speed of the printer. It will be cheaper to capture your mail or terminal session by opening the capture buffer (see Chapter 7). Close the capture buffer when all the data you want have been received. If your session is over, you can disconnect from your host. Then click on the **New CLI** in the **PROJECT** menu and type:

1 > copy Filename to PRT:

Voice

"Filename" is the name of the captured file. While your file is being printed, you can use other programs or continue your terminal session at the maximum speed. Remember to close the CLI window.

With audio equipment, you may also choose to have your mail or other transmitted data read aloud by your Amiga. Select the **On** or the **Filtered** option. When using the **On** option, every word and every transmitted printable character will be read aloud. Use the **Filtered** option if you would like a more natural reading of words with only the letters A to Z and the numbers 1 to 9.

The option **Requests** is for those who would like the voice for messages only. Messages from requesters such as "File Transfer Successful" will be spoken.

Use the **Now** option to empty the voice buffer. Since the voice is activated by receiving an "end of line" character (a carriage return or line feed) at the end of each line, you will not hear what is there until the line is finished. This is usually no problem unless you are receiving a prompt from your host. The voice will be activated only after you have typed your response to the prompt. If you want to hear the prompt, select **Now** for a reading of what is in the voice buffer.

Using the voice option without the X-on/X-off handshake is not recommended because your Amiga speaks at a rate of 165 words per minute, unless it is programmed differently with TalkToy. To slow transmission, use the X-on/X-off option if your host supports it.

The voice option is not recommended for transmission of files that contain a lot of control characters.

Since AmigaDOS is a multi-tasking operating system, other programs may be running at the same time as A-Talk III. If these programs use too much computing power, the voice will be distorted and will sound hoarse.

An automatic expunge of the narrator device and translator library will also be performed when selecting the **Voice Off** option. This can free valuable memory for Amigas short on memory.

Start/Stop Timer

This selection works as a toggle, either starting or stopping a timer in the screen or window bar. It reads **Start Timer** when the timer is off, and **Stop Timer** when the timer is on. The timer displays the date, the time, the amount of chip and fast memory, and the value of the timer (starting at zero when it is turned on). The timer can be toggled from the keyboard by pressing the right Amiga key and "\".

The timer is not automatically updated during file transfers and while there is no activity on the serial port. To update the timer, press the right mouse menu button.

List Disks

By default, the disk names used in requesters will be device names (for example, DF1:). You can change to volume names. If, for example, you have the Workbench diskette in the DF1: drive, the volume name is "Workbench:", and the device name is "DF1:".

Save

Use this selection to save all the options under the **SETTINGS** and **TERMINAL** menus in addition to **Modem Type**, **Line Type** and **Redial** options under the **PHONE** menu; **Screen**, **Icons** and **Voice** under the **PROJECT** menu; **Capture Options**, **ASCII Send Options**, **Line Delay**, and **Character Delay** under the **FILE** menu; **Protocol**, **Type**, **CRC**, **Timeout** and **Auto Chop** under the **TRANSFER** menu. This option will not exit A-Talk III.

Quit

Use this option to exit from your A-Talk III program. When you decide to quit, a requester will appear asking you whether or not you want to save your settings. If you say yes, all the options under the **SETTINGS** and **TERMINAL** menus will be saved in addition to **Modem Type**, **Line Type** and **Redial** options under the **PHONE** menu; **Screen**, **Icons** and **Voice** under the **PROJECT** menu; **Capture Options**, **ASCII Send Options**, **Line Delay**, and **Character Delay** under the **FILE** menu; **Protocol**, **Type**, **CRC**, **Timeout** and **Auto Chop** under the **TRANSFER** menu.

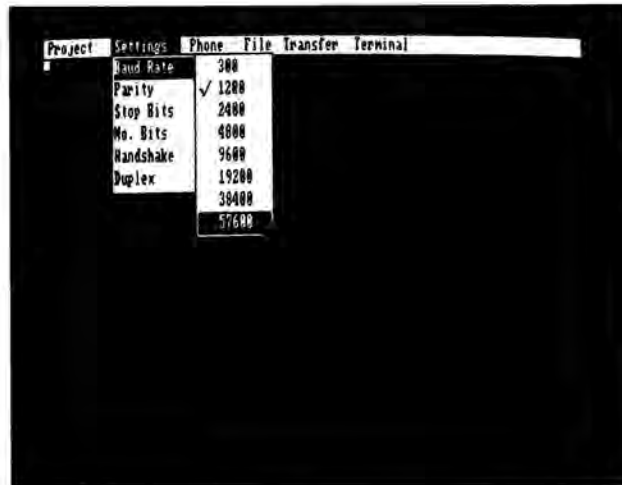
Note: After exiting from A-Talk III, wait for the disk drive light to turn off before removing the diskette. You will damage the diskette if you remove it when the disk drive light is on.

Exit

You may exit A-Talk III with no questions asked by selecting the "Exit" command in the **PROJECT** menu. None of the changes in the options are saved.

5. THE SETTINGS MENU

Move the cursor to the word **SETTINGS**. The following menu will appear:



You will have to set all of these options. In most cases the electronic bulletin board (BBS) or computer you are going to communicate with will specify what settings you need. All Amiga BBS's require eight bits, no parity and one stop bit.

Baud Rate

Baud rate refers to the speed at which data is transmitted. Drag the pointer down the **SETTINGS** menu and stop at **Baud Rate**. A-Talk III enables you to select any of the seven baud rates shown in the submenu. The one you choose will depend on your modem and the host computer or micro you plan on communicating with. Select the baud rate by moving the cursor to your choice and releasing the right mouse button. Use the higher baud rates of 38,400 and 57,600 only when you have two computers connected with a null modem cable. Even then the receiving computer may have trouble keeping up if it does not have enough power. The Amiga cannot receive at rates higher than 19,200 without getting a lot of errors. Do not multi-task at these higher speeds because all the power is needed by the serial device.

Let's say you'd like to connect to NewsNet. Most hosts will give you a guide to help you program your software. If you sub-

scribe to NewsNet, you will get a "Quick Reference Card". Take it out and look at the section on how to program your software. NewsNet accepts communication at 300, 1200 or 2400 baud. If you have a 1200 baud modem, select **1200**. Although your equipment will also permit you to communicate at 300 baud, in most cases you will be happier using the maximum speed allowed by your modem.

When two computers are connected directly with a null modem cable, you should be able to use 9600 baud or more. See Appendix A: REQUIREMENTS for an explanation of cables.

Parity

The other options in this menu refer to how the computer encodes and transmits data. Again, you will have to check with your host computer or micro.

A computer transmits data through electrical impulses. For a computer, an electrical impulse is equal to "1", and the lack of an impulse is equal to "0". All computer communication is based on combinations of zeroes and ones. A zero or one is called a "bit". Bits combine in groups of seven or eight to make up "bytes". "Parity" refers to whether the sum of the zeroes and ones in a byte will be odd or even. The phrase "seven bits **even** parity" means that the first seven bits will be added together and combined with the eighth bit to make an even number. For example, if the sum of the first seven bits is odd, a "1" will be added as the eighth bit to make an even number. If the sum of the first seven bits is even, a "0" will be added as the eighth bit to keep the number even. "Seven bits **odd** parity" means that the first seven bits will be added together and combined with the eighth bit to make an odd number. The option **mark** means that a "1" is always added no matter what the sum of the first seven bits is; **space** means a "0" is always added no matter what the sum of the first seven bits is. **None**, usually used in "eight bits no parity", means that no extra bit is added to the byte.

Many host computers will accept either 8 bits no parity or 7 bits even parity. Both setups will work for transmitting text files; but if you are transmitting a binary file, you **must** use 8 bits no parity.

Let's return to the NewsNet example. The NewsNet "Quick Reference Card" informs you that NewsNet accepts even parity. You will be transmitting a text file. Select "Even".

Stop Bits

NewsNet will permit you to select either one or two stop bits. If you are communicating at 300 baud, you may want to choose two bits. Otherwise flip a coin to make your choice.

No. Bits

Remember that most hosts will accept either 8 bits no parity or 7 bits even parity. If you are transmitting a binary file, you **must** use 8 bits no parity.

Handshake

In our NewsNet example, we are transmitting a text file. NewsNet asks for a 7-bit word. Select the number 7.

Since one computer may send data faster than the other can receive it, data could be lost unless something is done to control the send/receive operations. The X-on/X-off handshake delays the transmission of data until the receiving computer is ready to receive.

X-on/X-off should not be used with XMODEM, YMODEM or YMODEM Batch protocols. It may be used with Kermit and ZMODEM protocols, captures using the FILE menu, and in terminal emulation.

The X-on/X-off handshake is very useful when transferring files with the FILE Menu at speeds faster than 300 baud. Since A-Talk III sets the input buffer of the serial port at 8,000 characters (8Kbytes), data will be lost when the buffer overflows. The handshake will prevent data loss.

Going back to our example, you see that NewsNet supports the X-on/X-off handshake. Since we will be using the FILE Menu to transmit our data, we may use X-on/X-off. Select it.

A-Talk III also supports the RTS/CTS handshake, which is faster than X-on/X-off. It is recommended for ZMODEM transfers if your host modem supports it. You can select the RTS/CTS through the Amiga Preferences. After invoking Preferences, choose the "Change Serial" icon, then select Handshaking RTS/CTS. Click on **OK** and save or use. For file transfers at high speeds, you can also select a buffer size of 16,000 bytes (the default is 8,000 bytes). After making these selections from Preferences, you can then run A-Talk III and the serial port will be opened with the RTS/CTS handshake.

Duplex

When connected to a host computer, any command typed on your Amiga keyboard will be received by the host. In **full duplex**, the host will retransmit each command back to your Amiga. In **half duplex**, the host receives but does not echo your command; your Amiga then has to make it appear on your monitor.

If you make a mistake and select full duplex when you should have selected half duplex, you will see each character doubled like this: "mmiissttaakkee".

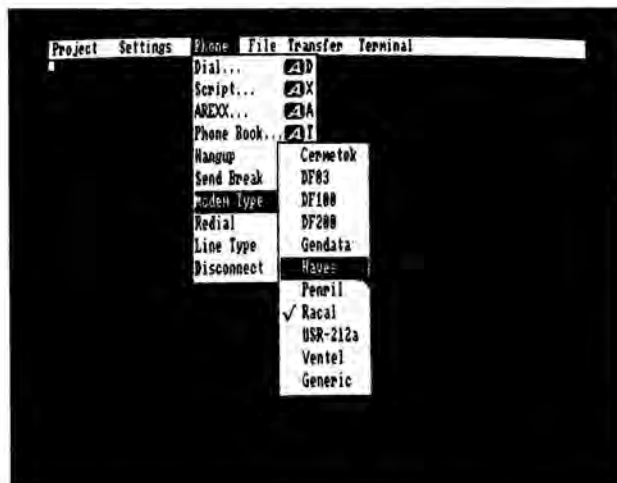
Half duplex should be used when connecting to another PC since micros usually do not echo the characters sent.

NewsNet supports **full duplex**. Select it.

Full duplex is sometimes called "local echo off". Half duplex is also referred to as "local echo on".

6. THE PHONE MENU

Under the word **PHONE** you will find this menu:



Modem Type

Select your modem type from those shown. If your modem is a Hayes compatible, select **Hayes**. If your modem is not listed, you may select **Generic**. The **Generic** option will work with any modem that correctly handles the carrier detect (CD) signal on pin 8 of the serial cable. If this pin is not connected to the modem, the generic modem option will not work since the software looks for the carrier detect signal. If your modem does not assert the carrier detect signal on pin 8, you will have to dial manually.

The US Robotics HST modem is a high-speed modem that is very popular with IBM PC-based bulletin boards. If you have this modem, optimum performance is obtained with the RTS/CTS handshake, a 16,000 byte serial buffer and 19,200 baud.

Line Type

Choose **Pulse** or **Touch** depending on what type of telephone line you have. If you have a rotary dial line, select **Pulse**.

Disconnect

When using any modem **other** than a Hayes compatible, pin 20 (DTR), as explained in section 2, **must** be connected for hang up to work. There seem to be quite a few early Amiga 1000's with a defective integrated circuit (IC) chip, the 1488 DTL quad line driver. This causes improper hang up; consequently the redial feature is inhibited. If you think your Amiga might have this defective chip, con-

sult your Amiga dealer. Unfortunately, there is no other way of disconnecting with non-Hayes compatible modems besides pin 20.

Due to this problem with early Amiga 1000's, A-Talk III gives you a choice for hang up with Hayes compatibles that support a hang up command (+ + +ATH). Check your modem manual to find out if yours does. For Hayes compatibles, you may choose either **Drop DTR** or + + +ATH for hang up. Some Hayes compatible modems do not support hang up with the "+ + +ATH" sequence. With these modems you should use the **Drop DTR** option.

Redial

If your modem supports an automatic dial, you can decide how many times you would like to dial: once, ten times, or continuously. The default is once. If your host is difficult to reach, you may want to try **ten times** or **continuously**. Make your choice before dialing or going into the **Phone Book**.

Between each redial is a delay of five seconds. If connection is not established after thirty seconds, there will be a timeout that will trigger the next redial. The length of the redial timeout can be programmed in the **Phone Book**.

Dial

Make this selection when you want to dial manually, without going into the **Phone Book** requester. The telephone number you dial will not be saved after each session. For setting up a permanent telephone directory, see the section on **Phone Book**.

When you choose **Dial**, a requester will pop up asking for a telephone number. Type it in. Check your modem instruction manual to see if you will have to include extra characters (such as pauses for long distance access codes) in order to dial properly. Click on **ok** or press **RETURN** when you are ready to dial.

If the line is busy or for any other reason dialing cannot be completed, you will get an error message. The progress of the call will be spoken if the voice option is on.

Script

Make this selection when you don't want to call up the **Phone Book** requester to use your login script. When you make this selection, the file requester will pop up asking for the name of your script file. After the selection, your login script will be executed. Before you use this option, you have to write a login script. See the section on **Script Tutor** in Chapter 7, **THE FILE MENU** or **11, THE SCRIPT LANGUAGE** for an explanation of how to write scripts. If you would like to end the execution of your script file before it is finished, you can select the **End Script** option which will appear in place of the **Script** option while your script is being executed.

ARexx

ARexx is a high-level language, an implementation of REXX, which can extend the capabilities of other programs used on the Amiga. If you have the ARexx program installed on your Amiga, A-

Hangup

Send Break

Phone Book

Talk III's script language can be extended and customized by users who take advantage of the ARexx option.

Using ARexx to enhance your A-Talk III login scripts is not recommended for the beginner. It is a very complex process, and therefore we have dealt with it in **Appendix: C, A-Talk and ARexx**.

Briefly, if you choose this option, a requester will pop up asking you for the name of an ARexx program. Programs using only ARexx commands should be given the file extension ".rexx" to distinguish them from programs written in other languages. Programs using ARexx and A-Talk III script language commands should be given the file extension ".atk".

Use this command to hang up, or disconnect your modem according to your **Disconnect** option in the **PHONE** menu.

This option simulates the break key, which is usually used to disconnect a terminal from the host computer. The break may have to be repeated several times.

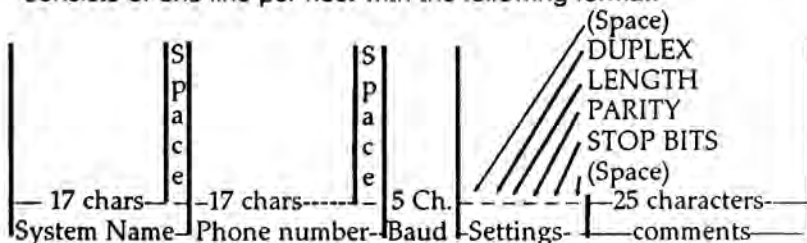
The A-Talk III **Phone Book** requester looks like this:



The **Phone Book** requester has sixty entries for sixty different host computers that you regularly log into. As you can see, only thirty boxes are visible at any one time. At the left under the boxes you will see an arrow gadget. Click on it to toggle back and forth between the first and second pages of the **Phone Book**.

Under the boxes you will see an information area. In this area there will be some pertinent information about the host: system name, phone number, baud rate, set options, a comment, queue index and long distance code. This is the sort of basic information

about hosts that you can easily download from many electronic bulletin boards or databases. This format, developed by Chet Solace, is compatible with files available on many bulletin boards. The file consists of one line per host with the following format:



For example, these are 4 entries:

```
Color America 1,619-364-3816 2400 F8N1 Chet Solace's BBS!
FAUG          1,415-595-5452 2400 F8N1 First Amiga User's Group
SFVAUG        1-818-368-4248 2400 F8N1 San Fernando Valley U.G.
BIX           587-0030       1200 F8N1 Byte Information Intchnng.
```

At the bottom of the **Phone Book** you will find a 35-character requester labeled **Long Distance Code**. This will allow you to enter an MCI, PC PURSUIT or SPRINT access code to be prefixed to the actual phone number. For example, it could contain the number:

1-800-347-6578W-12345-617-920-2050

The first part is the access code, which is followed by the Hayes "W" command (this waits for a dial tone). Then comes the phone company card secret access code and home phone number. The destination phone number for the host that you are calling is stored in the **Phone Number** string gadget.

At the bottom of the **Phone Book** you will also find a 2-digit string requester labeled **Queue**. This will allow you to link one host with another. Notice that each entry in the **Phone Book** is preceded by a host number. These numbers are used to link hosts. If a host is busy, the host indicated in the **Queue** requester will be tried. Circular lists are allowed (1->2->3->1).

Note: A-Talk III will look for the Phone Book and the saved settings in the subdirectory "Settings" of the current directory. A-Talk III will then look in AT3:Settings. If no files are found in either place, A-Talk III will put up a requester indicating that the Phone Book was not found. If you get this message, look in Appendix B to make sure you have installed A-Talk III correctly.

The Set Gadget

The **SET** box refers to four options: full or half duplex (F or H), number of bits (7 or 8), parity (E for even, O for odd, N for no parity, M for mark, and S for space), and number of stop bits (1 or 2).

The Phone Book is stored in the file AT3.Phone in the Settings drawer. While it can be edited manually, it is easier to edit entries using the A-Talk III Phone Book gadgets.

When downloading a phone book file from a BBS, the file can be installed as the phone book for A-Talk III by copying it to the file named AT3.Phone. The current version of A-Talk III will load and save the first 60 entries of the phone book.

As you can see, the only difference between the files downloaded from bulletin boards and the string gadget in the A-Talk III Phone Book requester is that the Phone Book requester also includes the script name and redial time for each host. And, of course, A-Talk III also has two more requesters, **Long Distance Code** and **Queue**. In order to merge the downloaded files with the A-Talk III Phone Book, we have put the script name, the redial time, queue index and long distance code for each host in a separate file, named AT3.Scripts, stored also in the Settings drawer. Its format is as follows:

```

      Delay (2 chars)      (Space)
      (Space)             Queue index (2 chars)
      (Space)             (Space)
|-----30 characters-----| |dd| |qq| |-----35 characters-----|
|-----Script Name-----| |dd| |qq| |-----Long Distance Code-----|

```

For example, these are two entries:

```

      genie.script      60
      bix.script        40

```

For each host in the downloaded phone book file, you might want to add a script name and, if you want to change the default timeout, a redial time; you may also add a long distance code or queue index number. There is a line by line correspondence between the AT3.Phone file and the AT3.Scripts file. The easiest way to add the script name, default timeout, long distance code or queue index for each phone book entry is to edit the Phone Book requester.

The default selection, which appears in the string gadget, is the host in the first box of the **Phone Book**. You can select another host from your **Phone Book** by clicking on the box that contains it. The host you have selected will then appear in the string gadget below the boxes.

Add an entry to your **Phone Book** by selecting the first empty box. Type in the name, phone number, and any other information into the string gadget. You must add new entries to the first empty box in the requester. Then click on the **EDIT** gadget at the right of the requester. If you want your edited changes to go into a permanent file, you must also click on the **SAVE** gadget.

The Edit Gadget

	<p>To edit an entry, click on the box containing the host name you want to edit. The host you want to edit will appear in the string gadget at the bottom of the requester. Make any changes by typing in the string gadget. This is the time to indicate or change the name of the login script file you want to be called up after dialing. If you would like to change the default redial delay of twenty seconds, do it now by typing another number (seconds) into the Redial Delay box. Then click on the EDIT button. If you want your edited changes to go into a permanent file, you must also click on the SAVE gadget.</p>
The Colors Gadget	<p>Click on the COLORS gadget to invoke the color requester, which is explained in Chapter 4, THE PROJECT MENU. You must click on SAVE to save the colors for each host.</p>
The Delete Gadget	<p>Delete an entry by first clicking on the box containing the host name, and then clicking on the DELETE button at the right of the requester. If you delete an entry, a box will remain empty. You can then add another host to that box.</p>
The Fkeys Gadget	<p>The FKEYS button will pull up the Fun Keys requester, which enables you to define twenty function keys: F1 through F10 plus the shifted keys F1 through F10. A-Talk III has already defined the function keys for sample hosts. If you want to define the function keys for a new host, refer to Chapter 4, THE PROJECT MENU.</p>
The Save Gadget	<p>The SAVE button will save the entire Phone Book directory and the edited host data (colors and parameters).</p>
The Quick Gadget	<p>Set all the options for a particular host by first clicking on that host and then on the QUICK button. The Quick Menu requester will pop up. This is a fast way to set options that normally appear on the main menu. For an explanation, see the Quick Menu section of Chapter 4. After setting the options, remember to click on the SAVE button.</p>
The Dial and Cancel Gadgets	<p>At the bottom of the Phone Book requester, you will see these options: DIAL and CANCEL. In order to dial you must first click on the host of your choice. Then click on the DIAL gadget. You may interrupt the dial procedure by pressing the escape key (ESC) on your keyboard.</p> <p>You may exit from the Phone Book requester at any time by clicking on CANCEL.</p>
The Script File Gadget	<p>Before using the Phone Book requester for automatic login, you will have to write a login script. A login script is a program that simulates the same commands that you would be typing in response to prompts from your host when you login manually. For example, if your host asks Name? and then Password?, you can write a program that will wait for Name? and then send the reply, your user name, after which it will wait for Password? and then send the reply, your password.</p>

A note to former A-Talk and A-Talk Plus users: A-Talk and A-Talk Plus scripts cannot be used with A-Talk III because the script languages are incompatible. If you are interested in converting your A-Talk or A-Talk Plus scripts for use with A-Talk III, see the end of chapter 11, THE SCRIPT LANGUAGE for a conversion table and documentation on the automatic translator program, Talk2Three.

Some sample hosts with login scripts have already been set up. All of the sample login scripts must be edited before use. You may edit them with Ed, TextCraft or your favorite text editor.

Let's say you would like to call NewsNet. The NewsNet login script has been named "newsnet.telenet". You will have to edit the NewsNet login script file in order to plug in your own ID and password. To use Ed to edit the file, call it up from the CLI with the command:

```
1 >Ed at3:scripts/newsnet.telenet
```

In the newsnet.telenet file, you will find the following login script which uses Telenet to access NewsNet:

```
DELAY 1
REPLY "^M^M"
WAIT "TERMINAL="
REPLY "^M"
WAIT @
REPLY "C 21566 ^M"
WAIT >
REPLY "ID NET12345 ^M"
WAIT "Password? "
REPLY "yourpassword ^M"
```

Type in your ID number in place of "12345" and your password in place of "yourpassword". When using Ed, the directional cursor keys work normally. Other Ed commands which you may need are:

Backspace	-Delete character to left of cursor
DEL key	-Delete character at cursor
CTRL-O	-Delete word or spaces
CTRL-M	-Return
ESC Q	-Quit without saving text
ESC X	-Exit, writing text into memory

Check your Amiga manual for additional Ed commands. When you have finished editing your script with Ed, save your script and exit with the command "ESC X".

The Function Keys

Now you may call up the **Phone Book** requester and make the NewsNet selection by clicking on the box containing NewsNet. Information about NewsNet will appear in the string gadget at the bottom of the requester. The parameters have already been set. Notice that the name of your login script, NewsNet, appears in the script file string gadget. Now all you have to do is type in your local NewsNet telephone number as you would dial it. If pauses are necessary, check your modem manual to see how to put them into your telephone number sequence. Then click on the **EDIT** button. Save the telephone number by clicking on **SAVE** at the right of the requester. This saves the entire phone book.

Then go to the bottom of the requester and click on **DIAL**. If your modem does not support an automatic dial, leave the telephone number string gadget blank. Dial manually by typing the modem command for dial, or literally dialing manually. When the phone connection is established, call up the **Phone Book**, select the box containing NewsNet, and then click on **DIAL**. Only the login script file will be processed automatically.

You may interrupt the dial procedure by pressing the escape key (ESC) on your keyboard.

If you would like to write your own login script for a new host, please refer to chapter 11, **THE SCRIPT LANGUAGE**.

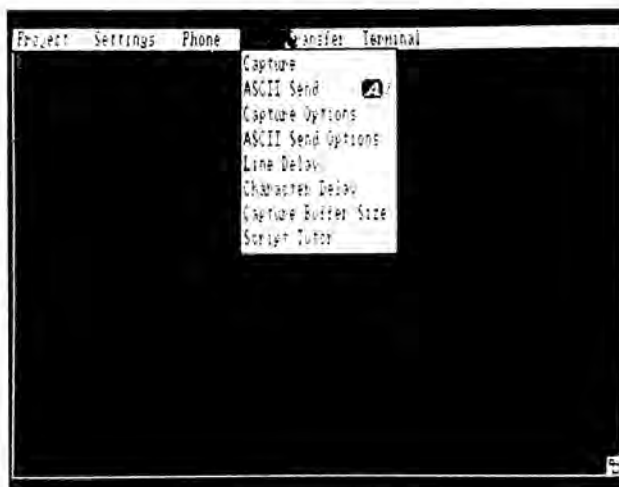
Program the function keys for a particular host by first clicking on the host and then on the **FKEYS** button. A requester will pop up, which enables you to define twenty function keys: F1 through F10 plus the shifted keys F1 through F10. A-Talk III has already defined the function keys for sample hosts. If you want to define the function keys for a new host, refer to Chapter 4, **THE PROJECT MENU**.

7. THE FILE MENU

The **FILE** menu is available for systems that do not support Kermit, XMODEM, YMODEM, YMODEM Batch or ZMODEM. Since **FILE** uses no error checking protocols, data transmission with this menu is always risky. Before selecting this option, make sure that:

- 1) Your host or remote micro supports the X-on/X-off handshake or you are receiving a small file of less than 4,000 characters (4Kbytes).
- 2) You are *not* transmitting a binary file.
- 3) You do not have a noisy telephone line. **FILE** is most often used for sending and receiving mail or capturing incoming data. In such cases a few errors in transmission won't make much difference.

You will see the following options under **FILE**:



Capture Options
ASCII Send
Options

The **Capture Options** and the **Send Options** must be set before trying to receive or send a file. These options refer to how various computers store the "end of line" character in their files. Some computers store the "end of line" as a line feed (LF); others

store it as a carriage return (CR); and still others store it as a carriage return and a line feed (CR/LF). For example:

- Amiga uses LF only.
- UNIX uses LF only.
- IBM PC uses CR/LF.
- Macintosh uses CR only.
- Apple II uses CR only.

Therefore you will have to set the **Receive Options** and the **Send Options** in accordance with the computer you are connecting to. The following charts will help you:

RECEIVE OPTIONS IF YOU ARE RECEIVING FROM:

Amiga	Strip CR
UNIX	Strip CR
IBM PC	Strip CR
Macintosh	CR→LF
Apple II	CR→LF
Public Networks	Strip CR

SEND OPTIONS IF YOU ARE SENDING TO:

Amiga	LF only
UNIX	LF only
IBM PC	CR/LF
Macintosh	CR only
Apple II	CR only
Public Networks	CR only

Once you have set the options to receive or send, you may begin transmission. After sending or receiving, remember to exit and logoff from the host.

Sending Mail

You will save connect time if you write and edit your mail off line, then connect to your host and send your letter. If you use this method, some hosts will expect a slower transmission that approximates normal typing speed. To deal with this problem, you can modify one or both of these two options: **Line Delay** or **Char Delay**.

Line Delay

A **Line Delay** is a delay at the end of each line. This is useful for hosts that pause to scroll at the end of the line. You may choose a delay of .25 (one quarter of a second) to one second.

Character Delay

To slow transmission character by character, modify **Char Delay**. You may choose delays from .05 to .2 of a second.

Let's try sending mail to the American People/Link network. First, you should use ED to write your letter to Ima Hacker. When

using ED, the directional keys work normally. Other ED commands you may need are:

Backspace	- Delete character to left of cursor
DEL key	- Delete character at cursor
CTRL-O	- Delete word or spaces
CTRL-M	- Return
ESC Q	- Quit without saving text
ESC X	- Exit, writing text into memory

Check your Amiga manual for additional ED commands. When you have finished editing your letter with ED, you have to signal the end of the transmission. According to People/Link conventions, your message should conclude with **/END** on a line by itself followed by a carriage return (↵). Put this at the end of your letter to signal the end of the transmission. Then you can save your script and exit with the command "ESC X".

Now you can login to People/Link. Select "Mail" from the main menu. In all examples given in this manual, everything you must type in will be *italicized*, and everything the computer types will be in **boldface**. A carriage return will be symbolized by the character (↵). The mail menu will scroll past and then you will get the usual prompt:

```
Please enter number or command:  
MAIL> /SEND  
SEND TO: IMA HACKER ↵  
Enter the subject (50 character limit):  
-> CAN WE TALK?↵  
Please begin your letter and type:  
/END on a new line when done  
Press <Return> after every line.
```

Now go to your **FILE** menu and select the **ASCII Send** option. A requester will pop up asking you to enter the file name. Type in the name under which you have stored the letter to Ima Hacker. Then click on **OK**.

While you are transmitting, the entire menu is ghosted except for the option **End Send** which will appear in place of **ASCII Send**. If you select **End Send**, the transmission will be interrupted. If you want to complete the transmission, just sit back and wait for an **End of File Send** requester.

If you have included a **"/END↵"** at the end of your letter, you will get the "MAIL" prompt. If not, you will have to exit by typing a **"/END↵"**. Then logoff from People/Link.

Receiving Mail or Capturing a Terminal Session

If you want to save your session on CompuServe, for example, or receive mail while still having access to all the other A-Talk III menus, you will use the capture buffer.

Under the **Capture** option you will see:

Open
Close
Clear
View
Cut
Copy
Paste

Open

Use the **Open** option to start the capturing process. Once you have selected **Open**, a requester will appear asking you for the name of the file in which you wish to store your session, or the name of the file you wish to transfer. By default the buffer is 16K bytes (about 16,000 characters). Change the default size by selecting the menu item **FILE Capture Buffer Size**, then type a new number in the requester. The file will be automatically saved to disk every 4K bytes. The buffer will enable you to preview your file once the capture has been closed.

Close, Clear and View

You will not be able to Clear or View the buffer until you have closed it. The menu will be ghosted until you select **Close**, which you may do any time after opening the capture buffer. If you select **Close** during your terminal session, you may reopen the capture buffer at any time--the additional text will be appended to your file.

Once you have cleared the capture buffer, it is emptied. The next time you open the capture buffer, the requester will pop up asking you for a new file name.

Once the buffer is closed, you can preview it while it is in memory by using the **View** option. What you see may be smaller than what is actually stored in your file because the buffer is circular. This means that when the buffer is filled, it will clear and begin filling up again from the top. Use the space bar on your keyboard to scroll to the next page, or use the vertical scrolling gadget at the right of your screen.

Let's say you would like to read your electronic mail. Ima Hacker will probably be sending a reply to your letter. Login to People/Link and select "/READ" from the mail menu. Then you may begin capturing at any time before the mail starts scrolling in front of you.

Go to your **FILE** menu and select **Capture Open**. A requester will appear asking you to name the incoming file. You may type "reply.hacker" and then click on **ok**.

The entire menu will be ghosted except for the **Close** option. The message will scroll as it is received. When you see that all of

Cut, Copy and Paste

the message has been received, select **Capture Close**. At this point, you can view your message by selecting **View**, or you may reopen your capture buffer and append another message to it.

The **Cut**, **Copy** and **Paste** options of the file menu take advantage of the Amiga's special "Clipboard" holding area, so portions of data downloaded with the A-Talk III **FILE Capture** function can be easily interchanged with other programs. Data put on the "Clipboard" can survive reboots and powerups because it is saved in a file in the devs/clipboard directory on the Workbench disk.

When using the **Cut** option, the entire contents of the capture buffer will be cut and put on the "Clipboard". It will no longer be in the capture buffer. The **Copy** option will copy the entire contents of the capture buffer onto the "Clipboard", while leaving the contents of the capture buffer intact. The **Paste** option will put the entire contents of the "Clipboard" into the capture buffer.

Probably the most common use of these options would be to take downloaded data from the capture buffer with the **Cut** or **Copy** options and put it on the "Clipboard", where it then could be transferred (pasted) to other programs that use the "Clipboard" feature, such as the Notepad. This will enable you to insert downloaded data in a Notepad file, or get your downloaded data printed with the fancy fonts, type sizes and type styles offered by Notepad. The Notepad is found in the Utilities drawer on the Workbench disk; it can be opened by double-clicking on its icon. The notepad has five easy-to-use menus (Project, Edit, Font, Style and Format).

Creating a Login Script with Script Tutor

This option is to be used to capture your login procedure and put it in a file that can then be used for automatic login. The file can also be edited to include more complex script commands, such as those for unattended operations. See chapter 11, **THE SCRIPT LANGUAGE** for an explanation of more complex script commands.

After selecting **On** from the **Script Tutor** option in the **FILE** menu, select **Capture Open**. The requester will pop up. Enter a new file name in the Scripts drawer. This will be the name of your new login script. Then dial and login in as usual. After you have successfully logged in, select **Capture Close** and **Script Tutor Off**. Now you can read your file by selecting the **View** command, or you can select the **Clear** command and read or edit your file off-line. It will look something like this:

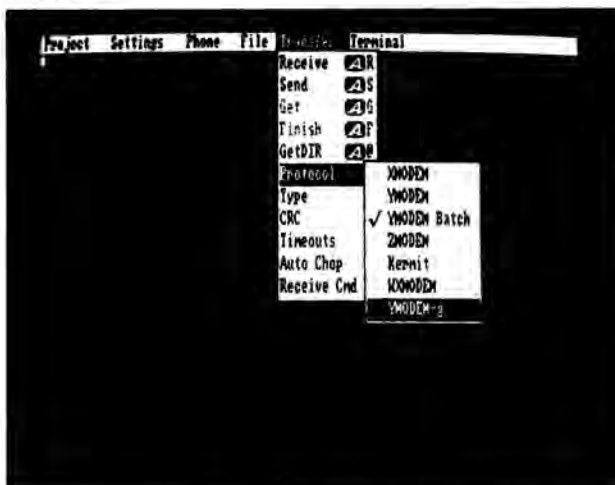
```
WAIT characters-from-host
REPLY characters-typed-at-the-keyboard
WAIT characters-from-host
REPLY characters-typed-at-the-keyboard
WAIT characters-from-host
```

When executed from the **Script** command or the **Phone Book** under the **PHONE** menu, the script file will automatically log you in.

The **Script Tutor** option works on the assumption that all replies typed in at the keyboard end with a carriage return. If you are communicating with a system that does not require carriage returns, the **Script Tutor** option may not work.

8. THE TRANSFER MENU

Move the cursor to the word **TRANSFER**. The following menu will appear:



Protocol

When two computers are communicating, they are transmitting and receiving electrical impulses. Even a small amount of static on the line may garble the message unless a protocol is used to ensure reliable transmission. The most widely used protocols are Kermit, XMODEM, YMODEM, YMODEM Batch and ZMODEM. Bulletin Board Systems usually support XMODEM, YMODEM, YMODEM Batch or ZMODEM, while mainframes and minicomputers often use Kermit. Some systems support YMODEM-g and WXMDEM.

XMODEM

XMODEM is a protocol used for the reliable transmission of files. It transfers information in blocks that are always the same size (128 bytes). Clearly, most files are not exact multiples of 128 bytes, so XMODEM pads the last block with zeros. In a text file transfer, those extra zeros can be edited out; but in a binary file transfer any extra zeros will make your entire file meaningless. That is why A-Talk III will strip the extra zeros from binary files transferred with XMODEM if you select the **Auto Chop On** option. Text files should be received with **Auto Chop Off**.

ARC and ZOO files are compressed binary files that do **NOT** need chopping. When using XMODEM or YMODEM to send or receive an ARC or ZOO file, select the **Send** or **Receive** option with **Type Binary** and **Auto Chop Off**.

YMODEM

XMODEM should be used with the communication parameters set at 8 bits no parity with no X-on/X-off handshake. Since some public networks require transmission with 7 bits even parity when not using XMODEM, A-Talk III will automatically save the current communication settings when an XMODEM transfer is invoked, switch to 8-bit/no parity/no X-on/X-off, transmit the file, and then restore the old settings.

XMODEM has to be run on both the host computer and your Amiga. Briefly, the procedure goes like this: login to your host and request a file to be sent or received using XMODEM, then go to your A-Talk III **TRANSFER** menu and request that the file be received or sent. If you are sending a text file, select the **Text** option under **Type**, then select **Send**; if you are receiving a binary file, select the **Binary** option under **Type**, then select **Receive**. After the transfer has been completed, you must remember to logoff from your host. XMODEM returns you to the host prompt immediately after the transfer. This means you do not have to make a formal exit from XMODEM before logging off from your host. For the same reason, you can transfer only one file at a time. Wildcard characters cannot be used.

XMODEM does not send the original file name along with the file; you have to designate a name under which the transferred file will be stored.

Since XMODEM does not send the file size along with the file, **Auto Chop On** is needed when receiving executable programs. This will eliminate padding characters. The padding characters, or extra zeros, will make your executable file meaningless unless these extra characters are chopped off.

Set the **Auto Chop Off** for transferring ARC or ZOO files since these are compressed binary files that do not need chopping.

This is a variation on the XMODEM protocol: rather than transfer data in blocks of 128 bytes, YMODEM transfers data in blocks of 1,024 bytes. This allows for faster transmission, unless your phone line has a lot of static. Static (or noise) will cause errors in transmission, and the entire block will have to be retransmitted until it is error-free.

YMODEM is sometimes called XMODEM-1K. It should not be confused with True YMODEM, or YMODEM Batch, which is described below.

Since YMODEM, like XMODEM, does not send the file size along with the file, **Auto Chop On** is needed when receiving executable programs. This will eliminate padding characters. The padding characters, or extra zeros, will make your executable file meaningless unless these extra characters are chopped off.

YMODEM Batch

Set the **Auto Chop Off** for transferring ARC or ZOO files since these are compressed binary files that do not need chopping.

Sometimes called True YMODEM, YMODEM Batch allows the batch transfer of information with wildcard characters. These characters ("*" and "?") may be used in order to designate groups of files that you would like transferred all at once. The "*" stands for any sequence of characters. The "?" stands for any one character. Using wildcard characters will increase the amount of time needed to locate your files.

With YMODEM Batch, data is transmitted in packets of 1,024 bytes. YMODEM Batch can use CRC-16. Information about each file (file name, file size, file date, etc.) is sent before each file transfer, and extra padding characters are automatically chopped off. When sending a file from your Amiga to a host, do not include the file name with the receive command that you give to your host.

ZMODEM

ZMODEM works exactly like YMODEM Batch, except that it is a more complex protocol. By acknowledging only the bad packets, transmission is speeded up. Also important is the fact that ZMODEM can use a more powerful cyclic redundancy check (CRC-32). See the section on CRC in this chapter for a further explanation.

ZMODEM also permits you to resume an interrupted file transfer both when invoked by the host and when selected locally on the Amiga. If the destination file exists (which is the case if the transfer has been interrupted), a requester will come up which will ask you if you want to **resume** or **replace** the file. Selecting **resume** will append the received data to the destination file, and **replace** will overwrite the old file with the received data (the old file's data is lost). As described in Chuck Forsberg's ZMODEM documentation, the **resume** option will only be allowed for binary transfers, when no text conversion option has been requested by the host ZMODEM program.

ZMODEM supports remote commands from a host computer, generally a UNIX based system or another Amiga running A-Talk III release 1.1. ZMODEM commands can be sent and received by A-Talk III. When receiving ZMODEM remote host commands, the commands must provide their own windows if they need to do input/output on the Amiga.

The send and receive ZMODEM speed has been improved by 60% over A-Talk III release 1.0. This will be noticeable only at speeds of 9600 baud and higher. At 9600 baud you will be able to transfer files at over 1000 characters per second; at 19,200 bauds you will be able to receive to the RAM: disk at over 1500 characters per second when using ZMODEM and the RTS/CTS handshake.

The Kermit file transfer protocol was designed at the Columbia University Center for Computing Activities (CUCCA) in 1981-82 mainly by Bill Catchings and Frank da Cruz. It is available for many systems for only a nominal fee from Columbia and from various user group organizations such as DECUS and SHARE. Kermit has been sent from Columbia to hundreds of sites in this country and abroad, reaching hundreds or thousands more through various user groups and networks. For an in-depth treatment of Kermit, refer to the book, *Kermit: A File Transfer Protocol* by Frank da Cruz, published by Digital Press. The Kermit protocol was named after Kermit the Frog, star of the television series THE MUPPET SHOW, and is used by permission of Henson Associates, Inc.

The Kermit protocol works by transferring data in "packets" of control information. The packets are not of any particular size. Lost or garbled packets are detected and retransmission is requested. With Kermit, you do not have to use the **Auto Chop** option since Kermit transfers files without adding padding characters.

In order for the Kermit protocol to work, a Kermit program must be running on your Amiga and on the host or micro you are communicating with. When you are using the Kermit protocol of A-Talk III, you are automatically in connect mode; unlike other Kermits, there is no connect command and no escape sequence to go back to your Amiga.

Kermit will transfer the file name along with the file. If you want to change the name, you will have to do that after the transfer. By default, Kermit will strip the disk and directory names and convert the file name to capital letters (if you are sending a file from your Amiga to a host) or small letters (if you are receiving a file from a host). For example, a file named:

```
df1:mydir/myfile.txt
```

will be sent to a host computer as:

```
MYFILE.TXT
```

In order to transfer a file, you must connect to your host with A-Talk III. Once you are connected to your host, log in. Then run Kermit on your host. From now on you will be running two Kermits (one on your host and one on your Amiga), so you will have to tell each Kermit what to do. First, tell your host to send or receive a file. Then run Kermit on your Amiga to receive or send. If you are receiving a text file, select **Protocol Kermit, Type Text**, and then select **Receive**; if you are sending a binary file, select **Protocol Kermit, Type Binary**, and then select **Send**.

Wildcard characters ("*" and "?") may be used in order to designate groups of files that you would like transferred all at once.

Kermit Server Mode

The "*" stands for any sequence of characters. The "?" stands for any one character. Using wildcard characters will increase the amount of time needed to locate your files.

After the transfer has been completed, you must remember to exit from your host Kermit and logoff from your host.

Kermit server mode can only be used if your host supports server mode, in which case you do not have to invoke the send/receive commands on your host. All you have to do is type "server" after your host prompt in order to put the host Kermit in server mode. After that, all commands are given locally from your Amiga. Remember to select **Kermit** from the **Protocol** options on your A- Talk III menu. To send a file using Kermit Server Mode, use the standard **Send** selection on the **Transfer** menu. A requester will pop up asking for the file name. Select **Get** to receive a file, and a requester will pop up asking for the file name on the host. Shut down the Kermit Server Mode with **Finish**.

A more detailed description of the Kermit Server Mode procedure can be found at the end of this chapter under the section **Examples of Protocol Transfers**.

WXMODEM

WXMODEM (Windowed XMODEM) is a protocol designed by Peter Boswell and currently available on People/Link. It is a modified version of the XMODEM protocol that provides greater speed in uploads and downloads.

YMODEM-g

YMODEM-g is a high-speed modification of the YMODEM Batch Protocol; acknowledgements for data blocks are not used. Because it does not use error-recovery, YMODEM-g must be used with hard-wired connection or with a hardware error-correcting modem. When an error is detected, the transfer is aborted.

YMODEM-g requires unidirectional flow-control to work at high speeds. The protocol works by sending an X-off when the receiving input buffer is full; when an X-on is received transfer is resumed. The Amiga serial device (as implemented in versions 1.2 and 1.3 of the system software) only allows bidirectional flow-control. Therefore it is not possible to use flow-control with YMODEM-g in streaming (continuous) mode. As a result, we have found that YMODEM-g is reliable for receiving speeds up to 4800 bauds. There is no problem in sending files from the Amiga to another host at speeds up to 19,200 bauds. If your host BBS supports the RTS/CTS handshake, it should be used in place of X-on/X-off for faster baud rates.

CRC

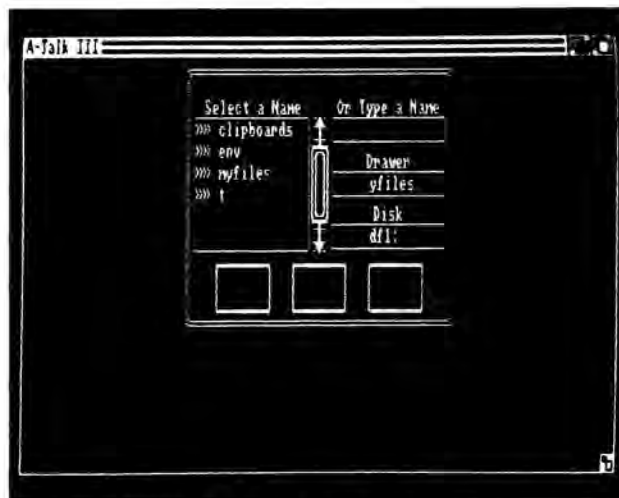
There are two ways of checking the XMODEM, YMODEM, YMODEM Batch and ZMODEM protocols: checksum and cyclic

redundancy check (CRC). Checksum makes a simple sum of each byte transmitted in the block. The cyclic redundancy check is more complex and more accurate, adding a character value to the bytes transmitted and verifying the sum. XMODEM is checked in groups of eight bits (checksum). YMODEM and YMODEM Batch are checked in groups of sixteen bits (CRC-16). ZMODEM is checked in groups of thirty-two bits (CRC-32). The larger the number of bits used for the cyclic redundancy check, the faster and more accurate the protocol.

If you would like to use CRC instead of the default checksum, select **CRC ON**.

Get DIR

There are only three cases in which, during the receiving process, no file requester comes up: receiving with ZMODEM, YMODEM Batch and Kermit Get. Since the file requester allows you to select the drawer and/or disk on which the received file is stored, these three cases require this special menu selection, **Get DIR**, for changing the drawer/disk on which the received file is stored. This option changes the disk and/or drawer where files will be stored and retrieved. When you select this option, a special file requester will pop up prompting you for a drawer/disk name. The **Type a Name** gadget will be ghosted. For example, if you want to save the next file you receive on the "DF1:" drive under a directory called "myfiles", you will select:



Since disk access is usually slower than memory access, you can speed up file transfers by sending from the RAM: disk or receiving on the RAM: disk. To receive, put in "RAM:" as your **Get DIR** option; after receiving a file, copy it to your diskette. When receiving, make sure you have enough memory on the RAM: disk before beginning the transfer.

Type

Before you start a transfer, it is important to select the type of transfer: binary or text. This will make a difference as to how your file is transferred. Use the binary selection for the transfer of executable programs or for the transfer of text files that require no translation (between two Amigas or between UNIX and an Amiga). Use the text selection for the transfer of text files that require the CR → LF translation (for example, PC-DOS files).

When using ZMODEM and YMODEM Batch, the type should be set to **text** ONLY when receiving text files from PC-DOS or CPM-based bulletin boards. When transferring text files between Amigas and between an Amiga and UNIX, the transfer type should be set to **binary** since both the Amiga and UNIX use the same end-of-line character and, unlike PS-DOS and CPM, they have no concept of an end-of-line character.

Timeouts

To prevent timeout problems when connecting to hosts through public data networks such as Telenet and Tymnet, two options are provided: **Standard** timeouts for use with Bulletin Boards (timeouts about every 10 seconds), and **Relaxed** for use with public networks (timeouts about every 23 seconds). Note that there is no extra delay introduced by choosing the **Relaxed** option in an error-free transfer. There will be some extra delay when transmission errors are encountered. If you notice that a transfer aborts with no indication of the type of error, it is probably a timeout error. In this case choose the **Relaxed** option and retry the transmission.

Auto Chop

Auto Chop is only needed when using XMODEM or YMODEM. When receiving executable programs with XMODEM or YMODEM, you will need to use the **Auto Chop On** option because these protocols add padding characters to all files transferred. The padding characters, or extra zeros, will make your executable file meaningless unless these extra characters are chopped off.

Set the **Auto Chop Off** for transferring ARC or ZOO files since these are compressed binary files that do not need chopping.

Examples of Protocol Transfers

Receiving a File with XMODEM

In this example we will be receiving a binary file from an IBM Personal Computer. The instructions below can be used whether the IBM PC is in the room with you, connected directly through cables; or miles away, connected through modems. You will have to initiate commands from both the IBM PC and the Amiga. In other words, both computers will have to be in the same room or you will have to get a friend to cooperate. One of you will need a modem with auto-answer mode.

In all examples given in this manual, everything you must type in will be *italicized*, and everything the computer types will be in

boldface. A carriage return will be symbolized by the character, "**␣**".

First go to your A-Talk III **TRANSFER** menu and set these options: **Protocol XMODEM, Type Binary, CRC On, Auto Chop On.**

Then go to the IBM PC and load a communication program such as PC-Talk III. You will see the prompt:

—PROCEED . . .

Alt-t (hold down the "Alt" key as if it were a shift key and type "t")

—TRANSMIT A FILE—

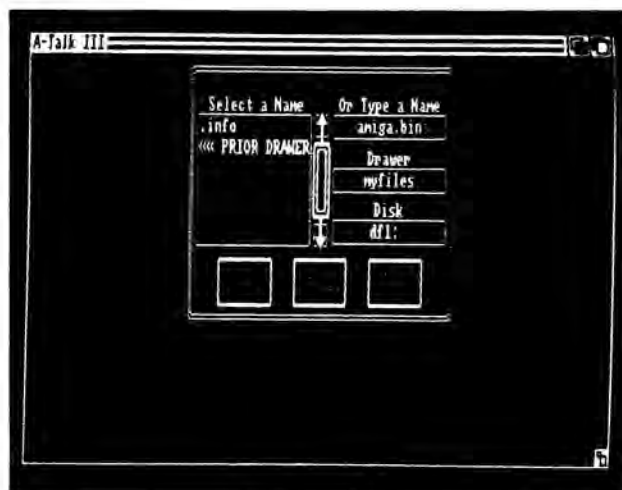
specification:sample.bin=x_␣

—TRANSMIT A FILE WITH XMODEM---

***Holding for Start . . .

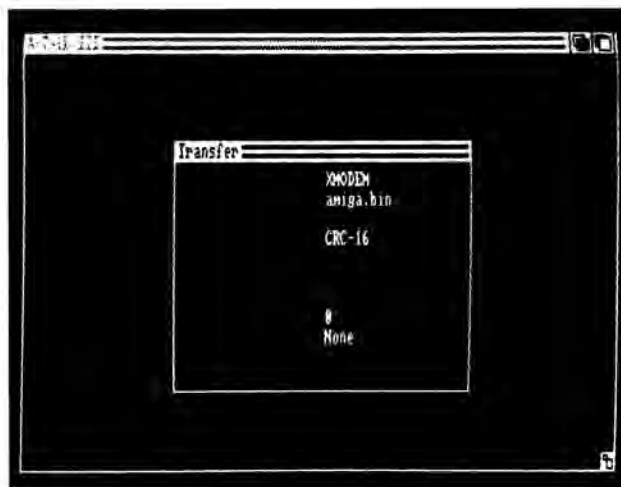
Note the PC-Talk III conventions: "Alt-t" to send a file and "=x" to run XMODEM. The particular file we are sending is named "sample.bin".

Now the IBM PC is waiting for your Amiga to start receiving. Go to your A-Talk III **TRANSFER** menu. Select **Receive**. The file requester will pop up. Use the requester to type in the name of the file you would like to receive. For a detailed explanation of how to use the file requester, refer to chapter 3, **TOOLS**.



In the above example, we have named the incoming file "amiga.bin". After you type the name, click on OK!. The transfer window will pop up. It is shown on the next page.

Sending a File with XMODEM



Your Amiga is in the process of receiving the file "sample.bin" and renaming it "amiga.bin". When possible, the transfer time is estimated. For XMODEM (as well as YMODEM, YMODEM Batch and Kermit) it is possible to estimate the transfer time and file size only when sending. The file size will also be shown when possible. In this example, the transfer time and file size cannot be shown. The numbers referring to the bytes and blocks transferred are updated as the file is transmitted. The error count is the number of errors that occurred while transmitting the file. The last message is an informative message about the last error that occurred. If communication is not initiated within the first twenty seconds, A-Talk III will automatically abort the transmission. You may choose to abort the transmission any time by clicking on **ABORT**.

Let's send a text file to a Bulletin Board System. Go to your A-Talk III **TRANSFER** menu and select: **Protocol Xmodem, Type Text, CRC On, Auto Chop Off**.

After logging in, you will see your BBS prompt:

Transfer Menu (XR = XMODEM Receive, XS = XMODEM Send)

BBS> XR _

—RECEIVE A FILE—

specification: text.bbs _

—RECEIVE A FILE WITH XMODEM—

*****Holding for Start**

You may see a string of "**^U^U^U**" on your screen. This means that the BBS is trying to initiate a transfer. Go to your A-

Receiving Files With YMODEM and ZMODEM

Talk III **TRANSFER** menu and select **Send**. The file requester will pop up asking for the name of the file you want to send. Type in the name of the file and click on **OK**. The transfer window will pop up, giving you the status of the transfer and the option to abort.

When finished, remember to logoff from your BBS and hang up.

YMODEM and ZMODEM are protocols commonly used on electronic bulletin boards. In the following example we will be logging onto a bulletin board run by an Amiga users' group. After dialing, you will see the logo:

OPUS-CBCS v1.03b



Now supporting San Fernando Valley Amiga Users' Group!

Home of the ^yL ^vn e B l i Fan Club!
 n a n

-Connected to 1939 Line Two 300-1200-2400 Baud

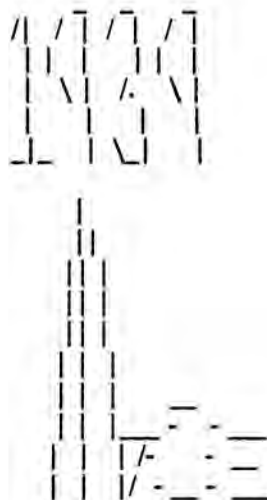
What is your FIRST name: *Userfirstname*

What is your LAST name: *Userlastname*

Userfirstname Userlastname [Y,n]? y

Password: *Userpassword*

Welcome to



Now go to your A-Talk III **TRANSFER** menu and set these options:

Protocol YMODEM, Type Binary, CRC On, Auto Chop Off.

MAIN MENU: Type '?' by itself for help

M)essage section F)ile section G)oodbye (logoff) S)tatistics

B)ulletin Y)ell at sysop C)hange setup U)ser list/search

R)eality Toggle

Select: f ↵

The FILES Section.

A)rea change L)ocate F)ile titles T)ype (show) G)oodbye

U)pload D)ownload S)tatistics M)ainmenu C)ontents

Select: d ↵

Select a method.

Z)modem

X)modem

Y)modem

M)odem7

S)ealink

Q)UIT (cancel the transfer)

Select: y ↵

What do you want to receive? *connect.arc* ↵

File: Connect.arc

Size: 40576 bytes (318 xmodem blocks)

Time: zmodem=2:49 xmodem/telink=3:45 sealink=3:22

Mode: Ymodem

Begin receiving now or send several CONTROL-X's to cancel.

Now the BBS is waiting for your Amiga to start receiving. Go to your A-Talk III **TRANSFER** menu. Select **Receive**. The file requester will pop up. Use the requester to type in the name of the file you would like to receive. You can use the name "connect.arc" or any other name you want to make up. For a detailed explanation of how to use the file requester, refer to chapter 3, **TOOLS**.

After you type the name, click on **OK!**. The transfer window will pop up. Your Amiga is in the process of receiving the file. When possible, the transfer time is estimated. For XMODEM (as well as YMODEM, YMODEM Batch and Kermit) it is possible to estimate the transfer time and file size only when sending. The file size will also be shown when possible. The numbers referring to the bytes and blocks transferred are updated as the file is transmitted. The error count is the number of errors that occurred while transmitting the file. The last message is an informative message about the last error that occurred. If communication is not initiated within the first twenty seconds, A-Talk III will automatically abort the transmission.

Receiving a file with ZMODEM

You may choose to abort the transmission any time by clicking on **ABORT**.

Now let's try receiving a file with ZMODEM. Go to your A-Talk III **TRANSFER** menu and set these options: *Protocol ZMODEM, Type Binary, CRC On.*

```
A)rea change L)ocate F)ile titles T)ype (show) G)oodbye
U)pload D)ownload S)tatistics M)ain menu C)ontents
Select: d _
Select a method.
Z)modem
X)modem
Y)modem
M)odem7
S)ealink
Q)QUIT (cancel the transfer)
Select: z _
What do you want to receive? missile.arc _
File: Missile.arc
Size: 84096 bytes (658 xmodem blocks)
Time: zmodem = 5:50 xmodem/telink = 7:46 sealink = 7:00
Mode: Zmodem
Begin receiving now or send several CONTROL-X's to cancel.
```

Now the BBS is waiting for your Amiga to start receiving. Go to your A-Talk III **TRANSFER** menu. Select **Receive**. Since ZMODEM transmits the file name along with the file, no file requester will pop up.

The transfer window will pop up. Your Amiga is in the process of receiving the file. With ZMODEM, it is possible to estimate the transfer time when sending or receiving. The file size will also be shown. The numbers referring to the bytes and blocks transferred are updated as the file is transmitted. The error count is the number of errors that occurred while transmitting the file. The last message is an informative message about the last error that occurred.

Transfer complete

```
A)rea change L)ocate F)ile titles T)ype (show) G)oodbye
U)pload D)ownload S)tatistics M)ain menu C)ontents
Select: g _
Goodbye?[y/n] y _
Logged off
```

You have now logged off from the users' group bulletin board.

Receiving a file with YMODEM Batch

We will be receiving two text files from a Sun Workstation running Berkeley UNIX 4.2bsd. With YMODEM Batch, you can send or receive a group of files all at once.

The first command you will type on the host is "ls", which is the UNIX command for listing a directory. The "pollux.usc.edu(4):" prompt refers to the name of the host.

```
pollux.usc.edu(4):ls
BUG1      BUG8 gz          rz.1      s.nr
BUG10     BUG9 minirb.1    rz.c      zm.c
BUG2      Bug1000         minirb.c  rz.nr    zmodem.h
BUG3      BugReport       ptest.sh  sb       zupl.t
BUG4      Makefile        rb        sx
BUG5      PRINT.MSG       rbsb.c    sz
BUG6      README          rx        sz.1
BUG7      UPDATE1.MSG     rz        sz.c
```

We are using Chuck Forsberg's "RBSB" program for UNIX to transfer the files with YMODEM Batch. We are using the wildcard character "*" to indicate all files with the suffix ".1". That would include the files "minirb.1", "rz.1" and "sz.1". The "RBSB" command for sending from the host with YMODEM Batch is "sb".

```
pollux.usc.edu(5):sb *.1
sb: 3 files 20138 bytes 1.8 minutes
```

Now the host is waiting for your Amiga to start receiving. Go to your A-Talk III **TRANSFER** menu. Select **Receive**. Since YMODEM Batch transmits the file name along with the file, no file requester will pop up.

The transfer window will pop up. Your Amiga is in the process of receiving the files one at a time. The file name will change as soon as a new file in the batch starts being received. With YMODEM Batch, it is possible to estimate the transfer time when sending or receiving. The file sizes will also be shown. The numbers referring to the bytes and blocks transferred are updated as the files are transmitted. The error count is the number of errors that occurred while transmitting the files. The last message is an informative message about the last error that occurred.

Sending Files with ZMODEM

We will be sending three binary files from an Amiga to a Sun Workstation running Berkeley UNIX 4.2bsd. With ZMODEM, you can send or receive a group of files all at once. As with UNIX, A-Talk III uses the wildcard characters "*.*".

We are using Chuck Forsberg's "RBSB" program for UNIX to transfer the files with ZMODEM. The "RBSB" command for receiving on the host with ZMODEM is "rz". Note that no file name is indicated in the command. This is because the file name on your Amiga will be sent along with the file.

pollux.usc.edu(6):rz_

rz ready. To begin transfer, type "sz file ..." to your
modem program **B0100000027fed4

Now the host is waiting for your Amiga to start sending. Go to your A-Talk III **TRANSFER** menu. Select **Send**. The file requester will pop up. Since we are sending all three binary files from the same directory with the extension ".uue", we will enter "*.uue" in the file name string gadget of the file requester. Then click on **OK!**.

The transfer window will pop up. Your Amiga is in the process of sending the files one at a time. The file name will change as soon as a new file in the batch starts being sent. With ZMODEM, it is possible to estimate the transfer time when sending or receiving. When sending multiple files, the total estimated transfer time will be shown. The file sizes will also be shown. The numbers referring to the bytes and blocks transferred are updated as the files are transmitted. The error count is the number of errors that occurred while transmitting the files. The last message is an informative message about the last error that occurred.

Remember to logoff from your host.

pollux.usc.edu(7):logout _
have a nice day!

Using ZMODEM Resume

ZMODEM permits you to resume an interrupted file transfer both when invoked by the host and when selected locally on the Amiga. If the destination file exists (which is the case if the transfer has been interrupted), a requester will come up which will ask you if you want to **resume** or **replace** the file. Selecting **resume** will append the received data to the destination file, and **replace** will overwrite the old file with the received data (the old file's data is lost). As described in Chuck Forsberg's ZMODEM documentation, the **resume** option will only be allowed for binary transfers, when no text conversion option has been requested by the host ZMODEM program.

Using ZMODEM Remote

ZMODEM supports remote commands from a host computer, generally a UNIX based system or another Amiga running A-Talk III release 1.1. ZMODEM commands can be sent and received by A-Talk III. When receiving ZMODEM remote host commands, the commands must provide their own windows if they need to do input/output on the Amiga.

*The ZMODEM command feature can be misused. For this reason, ZMODEM commands are rejected unless you have selected **Receive Cmd On** from the **Transfer** menu. This setting cannot be saved, so you must select it manually each time you want to use the remote feature.*

After selecting **Transfer Receive Cmd On**, you can have commands sent from a remote host to be executed on your Amiga. For example, "sz" is the send command on a Unix system. If you type:

```
sz -i Clock
```

on the UNIX system, and then select **ZMODEM Transfer Receive** on the Amiga, A-Talk III will execute the "Clock" program and return immediately upon successful startup.

When using ZMODEM, a new selection **Command** appears on your **Transfer** menu. This command is used to permit two Amigas running A-Talk to exchange commands. The remote A-Talk III user can select **ZMODEM Transfer Command**. A requester will pop up asking for a command. The remote user could type in, for example:

```
makedir RAM:mydir.
```

Then the local A-Talk III user can select **ZMODEM Transfer Receive**. The command is transferred and will instruct the local A-Talk III to run the "makedir" program, creating the appropriate directory and resuming execution only after the "makedir" command has exited.

Entire sets of directories and subdirectories can be transferred with ZMODEM when the same directory structure has been established on both the host and the Amiga, and GetDIR has selected the "root" of the corresponding Amiga directory tree.

The ZMODEM command entered in the Transfer Command requester can be prepended by a single character code:

Character	Meaning
none	return only after the command has completed on the "remote" host.
~	return immediately upon the receiving host's successful receipt of the command
+	interpret the following command as the name of an A-Talk III script instead of a DOS command

Sending Files with YMODEM and YMODEM Batch

Sending a file with YMODEM is identical to sending a file with XMODEM. Sending a file with YMODEM Batch is the same as sending a file with ZMODEM.

Receiving a File with Kermit

Let's pretend you would like to receive a group of text files with the suffix "amiga" from a VAX computer running UNIX located at the Fun Research Lab.

Go to your A-Talk III **TRANSFER** menu and select **Protocol Kermit, Type Text**.

After dialing the number for the VAX, your session will start:

```
FUNVAX
Fun Research Lab, VAX A (FUNVAX)--4.2 BSD UNIX
login: your name _
password: your password _
last login: Mon Apr 01 12:15:00 on ttyh2
$kermit _
C-Kermit, 4C(057) 31 July 88, 4.2 BSD
Type ? for help
C-Kermit>send *.amiga _
```

Now go to your A-Talk III **TRANSFER** menu and select **Receive**.

The transfer window will pop up. Your Amiga is in the process of receiving the files one at a time. The file name will change as soon as a new file in the batch starts being received. With Kermit, it is possible to estimate the transfer time only when sending from the Amiga. The file sizes will be shown only when sending.

Your first file of the group "*.amiga" is being transferred.

Now look at the line labeled **Blocks Tranferred**. As a file is transferred, the number of blocks increases. You may choose to abort the transmission at any time by clicking on **ABORT**.

You should also watch the line labeled **Error Count**. For Kermit, this represents the number of retries for any given packet. If the host computer transmits a garbled packet, Kermit will request retransmission. If the transfer is not successful after a number of retries, Kermit will abort.

In order to receive a file with Kermit Server Mode, log into your host and proceed normally until you get your Kermit prompt. Then type in *server* as follows:

```
$kermit _
C-Kermit, 4C(057) 31 July 88, 4.2 BSD
Type ? for help
C-Kermit>server _
Starting server on /dev/tty. Escape back to your
local Kermit.
Use FINISH command to shut down server.
```

Receiving a File with Kermit Server Mode

Sending a File with Kermit

Now go to your A-Talk III **TRANSFER** menu and select **Type Text** or **Type Binary** depending on what type of file you would like to receive. Then select **Get** to receive with Kermit Server Mode.

The file requester will appear asking for the name of the file on the host. Wildcard characters ("*" and "?") may be used to indicate groups of files. The usual A-Talk III transfer window will appear. After you have finished receiving, select **Finish** to shut down your host server Kermit.

Before logging off from your host, let's try sending a binary file. As you can see from the prompt in our example, the host Kermit is ready for your command:

```
C-Kermit> set file type binary ↵  
C-Kermit> receive ↵
```

Now go to your **TRANSFER** menu and select **Type Binary**. Then select **Send**. The file requester will pop up asking you for the name of the file you want to transfer. You can use wildcard characters ("*" and "?") to send multiple files.

For example, type in "test.?" to designate all the files you want to transfer with the prefix "test" and a single character suffix (for example, test.o and test.s). Once you have typed in the desired file name, you may click on **OK**.

Some characters may appear on your screen if your host Kermit is trying to establish contact before you have time to give the **Send** command.

The transfer window will pop up. Your Amiga is in the process of sending the files one at a time. The file name will change as soon as a new file in the batch starts being sent. With Kermit, it is possible to estimate the transfer time only when sending from the Amiga. The file sizes will be shown only when sending.

After a file transfer session, you must remember to exit from your host Kermit and logoff from your host. Kermit is waiting for your command:

```
C-Kermit> exit ↵  
$logout ↵
```

Then hang up your modem. As mentioned above, A-Talk III will automatically take care of exiting from your Amiga Kermit.

Sending a file with Kermit server mode is identical to sending a file without server mode. Remember to shut down the Kermit Server Mode with the **Finish** command on the **TRANSFER** menu.

9. THE TERMINAL MENU

If you are connected to a host and are not transferring files (with Kermit, XMODEM, YMODEM, YMODEM Batch, ZMODEM or the File Menu), your Amiga is (or can be) emulating a terminal.

You will see these options under the **TERMINAL** menu:



Emulator

You have seven options under **Emulator**: **VT100**, **VT52**, **H19**, **TTY**, **ANSI**, **TALK** and **TEK4014**. The emulator you choose will depend on what your host supports. You should use **TTY** or **TALK** for bulletin board systems; use **VT100** or **ANSI** with CompuServe; use **VT100**, **VT52** or **H19** with minis or mainframes. Use **TEK4014** (Tektronix 4010/14) for connecting to mini or mainframe hosts that support Plot-10 graphics programs.

VT100

The **VT100** is a very popular full-screen editor. If your host supports **VT100**, you are no doubt already familiar with it. The A-Talk III **VT100** emulator will support these fonts: **USA** (sometimes called **ASCII**), **UK**, National fonts for European countries, special graphics, alternate ROM 1 and alternate ROM 2 in both 80 and 132 characters per line. **Double height and double width characters are not supported.**

The National and USA character sets are supported by simulating 2 additional ROMS of a VT100 DEC terminal. The Select Character Set (SCS) command can be invoked as follows:

G0 Sets Sequence	G1 Sets Sequence	Selects Set
ESC (A	ESC) A	National
ESC (B	ESC) B	National/UK
ESC (0	ESC) 0	National Special Graphics
ESC (1	ESC) 1	USA ASCII (Alternate ROM)
ESC (2	ESC) 2	USA Special Graphics (Alt. ROM)

The default character set is the National character set.

A-Talk III supports the numeric and application mode escape sequences for the keypad keys. The VT100 four function keys (PF1 through PF4) are emulated by the top four keypad keys on the Amiga 500 and the Amiga 2000. Amiga 1000 users will have to set the F1 through F4 keys. Use the **Fun Keys** option under the **Project menu** for a global setting, or the **FKEY** option in the **Phone Book** to set the keys for a particular host. For the Amiga 1000, set the F1 through F4 function keys as follows:

F1	^[OP
F2	^[OQ
F3	^[OR
F4	^[OS

Since a VT100 needs an 80 x 24 window, the Custom screen should be used unless Workbench interface mode is used in a small window (see below).

Note for VMS users:

SET TERM/INQUIRE	sets VT100 terminal.
SET TERM/WIDTH=128	sets VT100, small font, 128 char/line
SET TERM/WIDTH=132	sets VT100, small font, 132 char/line

Set **Auto-Wrap Off** when using EDT.

Note for UNIX users: The standard "vt100" UNIX termcap should be used with the A-Talk III VT100 emulator in 80 column mode. The "vt100-w" termcap should be used in 132 column mode.

Set **Auto-Wrap On** when using vi or emacs on UNIX.

You will be able to get a small VT100 emulator in a window that takes up only a little more than one quarter of your screen. To do this, you must use the Workbench interlaced screen.

First, click on Workbench. Select the Preferences icon from the Workbench window and click twice on it. In the upper right-hand

Full VT100 Emulation
in a Small Workbench
Window

Non-Interlaced VT100 Window - Workbench VT100 Support

corner you will see a gadget called Workbench interlace. Go to it and select **on**.

You might want to change the colors of your screen to get a minimum of flicker. Good color choices are: a black background (color 0) with light green foreground (color 1) plus orange. Then click on **save** in the lower right-hand corner.

Your interlace mode will take effect from the next reboot on, so you will have to reboot your Amiga (CTRLA-A). Your screen will now be 640 x 400 pixels.

Run A-Talk III and make two selections: **VT100 Emulator** option and **Small** on the **Font** option (see below, section Font). You will have to resize your A-Talk III window. First, pull it down to make it about one half inch longer. Then decrease the width by about four inches. This is your small VT100 emulation window.

You have full use of the VT100 emulator when using A-Talk III with a Workbench screen window. This is useful for host systems like UNIX and VMS where the terminal screen size can be redefined to any number of rows and columns.

On an NTSC Amiga system, the A-Talk III Workbench window will come up with a size of 77 columns and 23 lines. When a "window resize" operation is performed, the number of lines and columns is recomputed.

On a DEC VMS system, you can use the following commands to reset the host terminal definition to a 77 x 23 screen:

```
SET TERM/WIDTH=77 SET TERM/PAGE=23
```

When connected to a UNIX 4.2/4.3bsd system, one can create a local termcap file with the following entries:

```
d0|vt100|vt100-am|vt100am|dec vt100:\
:do=^J:co#80:li#24:cl=50\E[;H\E[2J:sf=5\ED:\
:le=^H:bs:am:cm=5\E[%i%d;%dH:nd=2\E[C:up=2\E[A:\
:ce=3\E[K:cd=50\E[J:so=2\E[7m:se=2\E[m:us=2\E[4m:ue=2\E[m:\
:md=2\E[1m:mr=2\E[7m:mb=2\E[5m:me=2\E[m:is=\E[1;24r\E[24;1H:\
:rf=/usr/lib/tabset/vt100:\
:rs=\E\E[?3l\E[?4l\E[?5l\E[?7h\E[?8h:ks=\E[?1h\E=:ke=\E[?1l\E:\
:ku=\EOA:kd=\EOB:kr=\EOC:kl=\EOD:kb=^H:\
:ho=\E[H:k1=\EOP:k2=\EOQ:k3=\EOR:k4=\EOS:pt:sr=5\EM:vt#3:xn:\
:sc=\E7:rc=\E8:cs=\E[%i%d;%dr:
dt|vt100-at|vt100at3|vt100 in A-Talk III with 50 lines:\
:li#50:co#80:\
:tc=vt100-am:
dw|vt100-atw|vt100atw|vt100 in A-Talk III with 77x23 Wokbench:\
:li#23:co#77:\
:tc=vt100-am:
```

The termcap file above defines the standard VT100 termcap entry (vt100-am), a modified entry that allows 50 lines by 80

VT52 and H19

ANSI

TTY TALK

columns in Full Screen mode (vt100-at3), and a modified entry that allows 23 lines by 77 columns (vt100-atw) for Workbench Screen mode. Assuming the local termcap file is named "mytermcap", the definition for a 77x23 VT100 screen is installed with the UNIX commands:

```
setenv TERMCAP $HOME/mytermcap setenv TERM vt100-atw
```

The VT52 and H19 terminals are fully supported, including alternate keypads and graphics fonts. Set **Auto-Wrap On** when using *vi* or *emacs* on UNIX.

"ANSI" means that your terminal emulation will support escape sequences such as clear screen, erase line and cursor movement as defined by the American National Standard Code for Information Interchange. Amiga supports these escape sequences with the "console device".

ANSI can use up to 8 colors and supports bold face and italics. It can be used with IBM PC based bulletin boards such as OPUS.

A-Talk III now provides FULL support of all IBM PC ANSI escape codes, when selecting ANSI Terminal, IBM PC Font, 8 colors and Full screen. For complete emulation of the IBM CGA, the following colors should be used:

	R	G	B
Color 1: Black	0	0	0
Color 2: Red	15	0	0
Color 3: Green	0	15	0
Color 4: Yellow	15	15	0
Color 5: Blue	0	0	15
Color 6: Magenta	15	0	15
Color 7: Cyan	0	15	15
Color 8: White	15	15	15

Note: If you use the *IBM-PC* font, 25 rows and the above 8 colors, A-Talk III will completely emulate the *IBM ANSI.SYS* screen device driver, including *IBM-PC* graphics characters.

This terminal emulation works like a regular teletypewriter.

The **TALK** option is useful for electronic conferences and talking to Sysops (bulletin board system operators). It will allow "CB" or "chat" mode, opening a separate window for whatever you type. A-Talk III gives you a one line editing buffer, so you can use backspace and delete to correct your message before sending a line out. Once you press carriage return, the line will be sent to the host. Messages from others will appear, ungarbled, in a separate window.

Whatever is sent by the host goes into another window. This is called a "split screen" function. Since what you type will be echoed locally, turn echo off on your host.

Some hosts do not allow you to turn off the echo option. In those cases, you will see lines of what you type intermingled with the message being sent to you by someone else. Your message will also appear as you type it in a separate window. Your message is not being garbled, so don't worry about it.

Other hosts, such as CompuServe, will allow you to turn off the echo option. The command to use on CompuServe's CB Simulator is "/NOECHO". The command to use on People/Link's Party/Line is "/NOE".

TEK 4014

Select this option if you want to emulate the Tektronix 4010/4014. Once you have made the selection, the **Tektronix** menu will be activated. Turn to Chapter 10. **THE TEKTRONIX MENU** for a more detailed explanation of this terminal emulator.

Auto Wrap

By default, the Auto Wrap is off. This means that the 81st or 133rd character typed (depending on which font you select) will overwrite the last character in the line. When Auto Wrap is on, characters will automatically be placed at the beginning of the following line after the 80th or the 132nd character typed. Check the above section **Emulator** to see how you should set the Auto Wrap feature for your particular editor.

Font

There are three fonts available: large, small and IBM-PC. The default is the large font. You may choose the small font (132 characters per line) to use as you please. This will enable you to get an 80 character window without occupying the entire screen--use the Workbench screen option and reduce your window.

Use the **PC** font for IBM bulletin boards. *Note that the PC graphics characters are only available when using the ANSI terminal emulator.* Use the **Large** font for Amiga bulletin boards. *Note that the Amiga graphics characters, like the PC graphics characters, are only available when using the ANSI terminal emulator.*

Four different fonts are available when using the Tektronix 4010/14 emulator. See Chapter 10: **THE TEKTRONIX MENU**

Bell

If your Amiga is connected to audio equipment, either a monitor with a speaker or your own audio device such as a stereo, you may choose the audio version of the bell (a beep). With no audio equipment you should use the visual bell, which appears as a flash on the screen.

Rows

You may select either 24 or 48 lines. The 24-line screen may be used on a non-interlaced screen. Once you choose the 48-line option, the screen will automatically become interlaced.

Columns

PAL and MoreRows screen sizes are supported when using the **Full** screen option. When A-Talk III is started, it detects the current Workbench screen size. From that information it computes the maximum number of rows on the fly and generates two items in the **Terminal Rows** menu with the respective number of rows for interlaced and non-interlaced full screens. It also updates the corresponding values in the **Quick Menu** requester. When using the standard NTSC screen, A-Talk III will allow a maximum of 25 and 50 lines using vertical overscan.

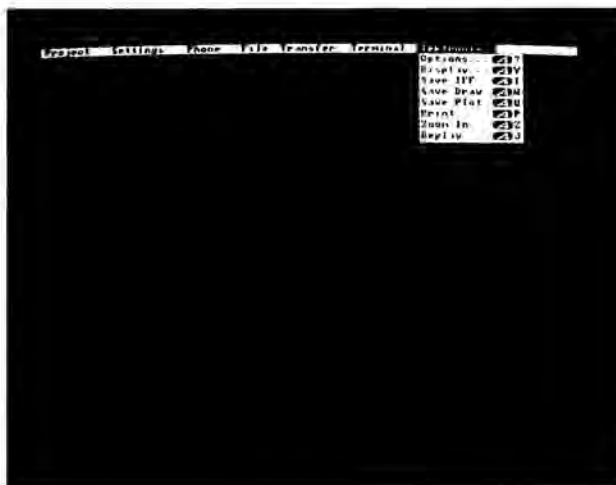
Colors

Choose either 128 or 132 columns (characters per line). When using 132 characters per line, the window will be larger than the Workbench window.

You may choose 2, 4 or 8 colors for the custom screen. A-Talk III will work faster with fewer colors. The Workbench always uses 4 colors.

10. THE TEKTRONIX MENU

A-Talk III supports full Tektronix 4010/4014 graphics emulation. The **TEKTRONIX** menu looks like this:



If you are using the **TEKTRONIX** menu, you are probably familiar with this terminal. The "4014 and 4014-1 Computer Display Terminal USERS MANUAL" is available from your local Tektronix Sales Office, part no. 070-1647-00. Call 1-800-835-9433 to find the local Tektronix Sales Office nearest you.

The **TEKTRONIX** menu is active only when you have selected the **TERMINAL** Emulator **TEK4014**. Once you make this selection, a screen will appear with the default colors: black (for the background), green (foreground drawings). It will be a 640 x 400 pixel interlaced screen. The left mouse button works like a toggle to bring up the screen drag bar. With the screen bar visible, you can drag down the Tektronix screen to see other screens behind it.

When using the interlace mode, you might want to go into Preferences and make some adjustments to avoid flicker. Put the brightness control in the middle position and reduce contrast until the flicker goes away.

Notice that the **PROJECT** Screen and **TERMINAL** Auto Wrap choices are ghosted; this is because they do not apply to the Tektronix emulator.

Fonts

Under the **TERMINAL Font** menu you will see the following choices: **MICRO** (tekmicro - 5 x 6 pixels), **SMALL** (teksmall - 5 x 7 pixels), **MEDIUM** (tekmedium - 8 x 11 pixels), **LARGE** (teklarge - 9 x 12 pixels).

You will be able to change the font from the A-Talk III menu as long as the program running on your host doesn't override this choice.

Keyboard

The Tektronix keyboard is like the VT100 keyboard except for these differences:

Key	Invokes	Action
ENTER	PAGE	Clear screen
Shift ENTER	RESET	HOME Cursor; select LARGE font

Retrographics VT640 Support

Two features of the RetroGraphics VT640 extension to the Tektronix escape sequences are included in A-Talk III:

•DATA LEVEL

All modes can enter data three ways--dots turned on, off, or toggled. The following sequences are recognized:

ESC /Ps d where Ps = 0 dots turned on
Ps = 1 dots turned off
Ps = 2 dots complemented (toggled)

Selective erase can be performed by turning the dots off and redrawing the vectors.

•LOAD CROSSHAIR

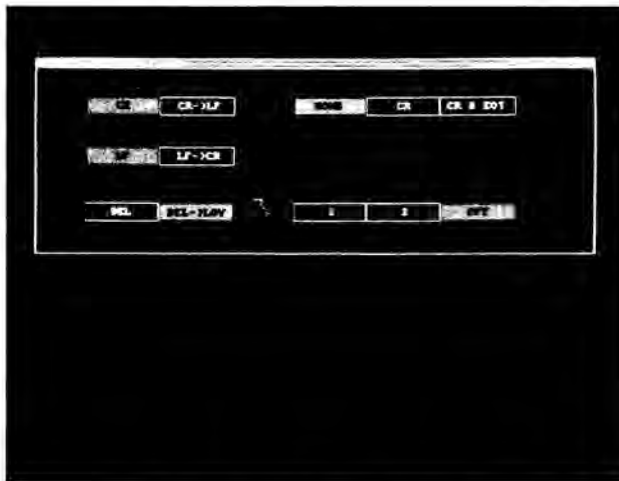
The sequence ESC /f takes the current graphics vector location and loads it into the crosshair location. This will work even when the A-Talk III screen is pulled down, and the cursor is currently over another screen. A "click" on the A-Talk III screen will also be simulated. When the GIN mode is entered, the crosshair will appear at the location loaded.

Options

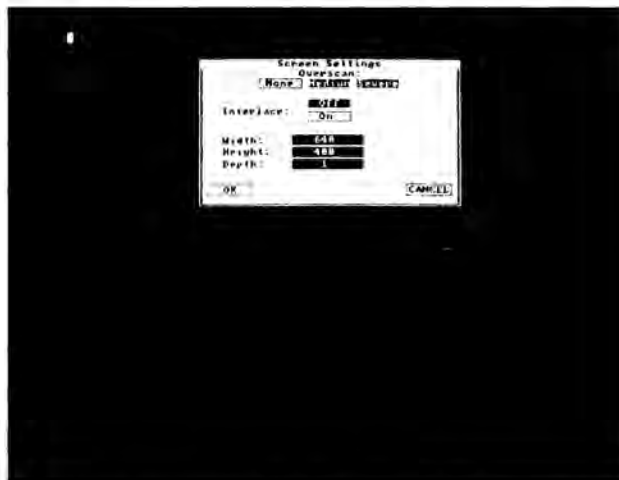
The choice **Options** refers to hardware options that would be available on a real Tektronix. These are explained in the Tektronix 4010 or 4014 user's manual. The "4014 and 4014-1 Computer Display Terminal USERS MANUAL" is available from your local Tektronix Sales Office, part no. 070-1647-00. Call 1-800-835-9433 to find the local Tektronix Sales Office nearest you.

This menu item calls up the **Tektronix Options** requester, which allows you to specify the *CR Effect*, *LF Effect*, *DEL implies LOY*, *GIN Termination*, and *Margin Control* options of the emulation. The requester looks like this:

Display



Use **Display** to change the size of your Tektronix screen. You can increase the size of the screen from 640 to 744 pixels horizontally, and from 400 to 480 pixels vertically, when using interlace mode and from 200 to 240 when using non-interlace mode. European PAL screens will allow you to get 112 more pixels vertically. Please note that you can actually see the maximum size *only* if your monitor will permit this. In all cases, the mouse cursor and the range of the crosshair will be limited to 640 x 400 pixels.



The **Interlace** gadget will let you choose **On** or **Off**.

You can change the size of the screen in two ways. By clicking on the **Medium** or **Severe** gadgets at the top of the requester, you will change the height and width of your screen as reflected in the **Width** and **Height** gadgets at the bottom of the requester. When

using overscan, the Tektronix screen will be automatically centered on your monitor. There is no need to use Preferences to reposition the upper left corner of the Workbench. For a more exact control of screen size, type the number of pixels you want in the **Width** and **Height** gadgets. The Display requester also allows you to change the number of colors (either 2 or 4). This is done by changing the **Depth** gadget from 1 (two colors) to 2 (four colors). The 2-color option is useful to display plots at the highest speeds and saves disk space when storing images in Deluxe Paint II format.

Save IFF

Use the option **Save IFF** to create a file that is compatible with Deluxe Paint II and other programs that accept the IFF format. Once you have selected **Save IFF**, a file requester will appear asking you for a file name. If you have **PROJECT Icons On**, saving a file in IFF format will create an icon identical to those created by Deluxe Paint II.

Note that the overscan images produced by A-Talk III are NOT compatible with Deluxe Paint (original version), but ONLY with Deluxe Paint II.

Display Program

For people that do not have Deluxe Paint II, we have included the Display Program in the Installation Disk. After having saved an image in IFF format, press SHIFT, with the left mouse button click once on the image icon and twice on the Display icon, then release the SHIFT key. The image will soon appear. Click the left button in the middle of the image to show the screen drag bar. Click in the upper left corner to exit Display. Display was written by Carolyn Scheppner of Commodore Business Machines.

From the CLI you can invoke Display by typing:

```
1 > display image-name
```

Save Draw

Save Draw works differently from **Save IFF**. Use this option if you want to create a file compatible with Aegis Draw. You must make this selection before drawing your image. The procedure to follow is this: choose the **Save Draw** option; give a command to your host which will send the image; when all of the image has been displayed, choose **End Draw**. Your plots will be temporarily saved in RAM and dumped to disk when they exceed 16K or when selecting **End Draw**. If you have **PROJECT Icons On**, this will create an icon identical to those of Aegis Draw. Please note that when plots include "clear screen" commands, they will be saved in different layers in different colors. If you have an Amiga with only 512K, it is advisable to save only one plot at a time since Aegis Draw will be able to handle only small plots in this case.

Save Plot

To save your plot on the disk in PLOT-10 format (the Tek 4014 format) select the option **Save Plot** just before starting to display the image that is being captured, and give a file name. The best

Print

time to do this is right **before** pressing the carriage return to display the image. After the image is completely displayed (and saved) select **End Plot**. Do this for each image to be stored locally for subsequent **Replay**.

To get a hard copy of your Tektronix screen, select the **PRINT** option. The type of output sent to the printer depends on the selections you have made in Preferences. When using the Tektronix default colors (black and green) these are the recommended printer selections:

HORIZONTAL ASPECT
NEGATIVE IMAGE
BLACK & WHITE SHADE
THRESHOLD 14

Note the selection of Negative Image. Since the default screen has a black background, the printer will print a black background and quickly use up your ribbon unless you ask for a negative image, which will print black on a white background.

If your screen colors are different from the default colors, you will probably have to change the threshold value. The threshold value refers to what is going to be considered black and what is going to be considered white.

Zoom In

The zoom and pan function initially appears as **Zoom In**. Note that this option uses a LOT of memory since it uses a SuperBitmap that matches the size (1024x768) of the Tek 4010 screen. To be sure there is enough memory on a 512K Amiga, you **MUST** use **ONLY** 1 bitplane (2 colors) and no windows or other programs running. If there isn't enough memory, an appropriate message will be displayed. On Amigas with expansion memory, you should also be able to use 2 bitplanes (4 colors) and still run other programs.

After selecting this option, a window with left and bottom scroll gadgets appears. Everything else will work the same, except that one can "Pan" over the image using the gadgets. Text will be the same size as in the other windows. One can return to the "full screen", by selecting the same menu item that now reads **Zoom Out**.

Replay

A-Talk III will let you capture plots in standard PLOT-10 format as well as Dpaint II and Aegis Draw format. You might want to use this to save connect charges. Once you have captured your plot in PLOT-10 format, you can "replay" the plot on the Amiga in order to save it in one of the other formats.

To "replay" a plot captured as previously described, select the **Replay** option from the TEKTRONIX menu, and give the file name

of the previously saved image. The plot will be generated at the speed of disk access, much faster than your communication line. You can abort the "replay" of the plot at any time, by pressing the SPACE BAR key on your keyboard.

11. THE SCRIPT LANGUAGE

Writing Your Own Script

A script is a program that simulates the same commands that you would be typing as responses to prompts from your host when you login or transfer files manually. You will have to write a program that will wait for those prompts and then send the proper reply, such as a carriage return, your user ID or your password.

Some sample hosts with login scripts have already been set up for you and are included on the disk. You will have to edit all of the sample scripts in order to plug in your own ID and password. All of the sample login scripts are stored in a drawer called "Scripts". You may edit them with Ed (see Chapter 6), TextCraft or your favorite text editor. We have printed these login scripts at the end of this chapter, so you can use them as examples. Also included are several more complex examples.

A note to former A-Talk and A-Talk Plus users: A-Talk and A-Talk Plus scripts cannot be used with A-Talk III because the script languages are incompatible. If you are interested in converting your A-Talk or A-Talk Plus scripts for use with A-Talk III, see the end of this section for a conversion table.

To write your own script, open a normal text file using TextCraft, Ed, or your favorite editor. Most commands must be on a separate line following this format:

COMMAND [number or string]

All commands (except the IF TIMEOUT command) can be abbreviated to three letters and can be in capital or lower case letters. A string is a group of characters. In most cases, strings must be set off by quotes. Single characters will not need quotes around them. When writing the strings that follow your commands, you will have to pay attention to whether your host uses small or capital letters. Also pay attention to spaces which may be required when waiting for prompts from your host or when replying to your host.

Beginning and ending double quotes are required when waiting for a string of characters. Quotes are not needed when waiting for a single character (case insensitive). Use the caret (^) to indicate control characters, as in ^C for a control C. Two carets (^ ^) indicate a caret character (^). When waiting for the double quote symbol ("), indicate it by a simple ".

A comment, or note to yourself which will not be executed by the computer, may be put into your script file preceded by the # sign. For example:

```
WAIT "SEND TO:"  
REPLY "IMA HACKER ^ M"  
# Enter the subject of your letter.  
WAIT "->"
```

Once you have written your script, you can start it in one of four ways: 1) from the **Script** option in the **Phone Menu**; 2) from the Phone Book; 3) from an ARexx script called up from the **ARexx** option in the **Phone Menu**; or 4) from the CLI with the command:

```
CLI> atalk3 FROM filename
```

where "filename" is the name of a script file.

The Script Commands

ASCIISEND

ASCIISEND will send an ASCII file from your Amiga to the host.

Format: **ASCIISEND** *filename*

Notice that the file name is not quoted!

Example:

```
WAIT "MAIL"  
REPLY "/SEND ^ M"  
WAIT "SEND TO:"  
REPLY "IMA HACKER ^ M"  
# Enter the subject of your letter.  
WAIT "->"  
REPLY "COPY PROTECTION. ^ M"  
WAIT "line."  
ASCIISEND df0:plink.mail  
WAIT "MAIL"
```

The example given is part of a script file for sending mail to People/Link. These commands will send your mail from disk drive df0: in an ASCII file to the host. The "WAIT" command tells your Amiga to wait for the prompt from the host. At the end of the file you will see a WAIT MAIL. If you have ended your letter with the proper People/Link conventions (with a /END ^ M), the host will send a MAIL prompt after the transmission of your letter.

ASK

The ASK command allows you to set up a question or a series of questions. The ASK command is used in conjunction with the IF command. This feature will allow you to set up script files with menus which will appear asking you to type your selection from the

keyboard. The keyboard character you type will not be sent to the host. It will serve to execute your command.

Format: **ASK** "*string*"

Example:

```
ASK "(K)ermit, X)MODEM, or Y)MODEM?"
IF K PROTOCOL K
IF X PROTOCOL X
IF Y PROTOCOL Y
```

With this example, you set up a little menu that will appear while your script is executing. It will ask you which protocol you want to use for a transfer: Kermit, XMODEM or YMODEM.

AT

The AT command will stop the execution of your script until a given time. It is very useful if you want to go to bed and have your computer make transfers at night when telephone rates are cheaper. Use the twenty-four hour format when putting in the time (for example, 8:00 PM would be 20:00).

Format: **AT** *hour:min*

Example:

```
AT 23:30
ON "NO CARRIER" GOTO login
login:
HANGUP
REPLY "ATDP000-0000 ^M"
TIMEOUT 30
WAIT "system?"
IF TIMEOUT GOTO login
```

This example of the AT command will stop the execution of your script until 11:30 PM. The script will start executing at eleven thirty PM and will dial and redial if a connection is not made within thirty seconds (the REPLY command is used with ON to dial--just add "ATDP" at the beginning and "^M" at the end of your phone number).

BAUD

To set the baud rate from your script, use this command.

Format: **BAUD** *baud rate*

Example:

```
BAUD 2400
SET F8N1
ROWS 24
COLUMNS 128
DEPTH 3
WRAP OFF
SCREEN Full
EMULATOR Ansi
FONT ibmpc
```

BELL

This example sets all parameters for connection to an IBM PC-based color Bulletin Board System, such as OPUS. It is most useful when used from an ARexx script, since all parameters can also be saved from the Phone Book QUICK gadget.

You can have the bell go off at any point in your script by using this command.

Format: **BELL**

Example:

```
DELAY 1
REPLY "^M"
WAIT "login:"
REPLY "yourname ^M"
WAIT "password:"
REPLY "yourpassword ^M"
WAIT "(vt100)"
REPLY "^M"
WAIT %
REPLY "source setamiga ^M"
WAIT %
REPLY "mail ^M"
BEL
```

This is an example of a login script for UNIX that you would use at home if you had a VT100 terminal at work. The bell will ring when the host begins transmitting your mail.

Notice that in the example only the first three letters of the BELL command have been used. All commands (except the IF TIMEOUT command) may be abbreviated to three letters.

BREAK

This command will send a break to your host, cutting off your connection. Any character that is about to be sent will be aborted by this command.

Format: **BREAK** [*value*]

Example:

```
BREAK 750000
```

The optional value for the break is given in microseconds. In this case the break is 750,000 microseconds or .75 seconds. If no value is given, the default break is .5 seconds (or 500,000 microseconds).

CD

Use CD to change the local directory for file transfers.

Format: **CD** *newdir*

Example:

```
WAIT "MAIL"
REPLY "/SEND ^M"
WAIT "SEND TO:"
```

```

REPLY "IMA HACKER ^M"
# Enter the subject of your letter.
WAIT "->"
REPLY "COPY PROTECTION. ^M"
WAIT "line."
CD df1:MYDIR
ASCIISEND more.mail
WAIT "MAIL"

```

This is the same example given under ASCIISEND for sending mail to People/Link, but this time your script file will change the directory to MYDIR on df1. It will send from a file called df1:MYDIR/more.mail.

CHAIN

Use the CHAIN command to link other script files to your login script. By linking one file at a time, you can link as many files as you want to your original login script. For example, script A can be linked to script B, which is in turn linked to script C and so on. If the CHAIN command is executed, the computer will read and execute the linked script(s) without returning to the original script.

Format: **CHAIN** *filename*

Example:

```

ASK "Select C)ompuServe or P)eopleLink"
IF C CHAIN df1:compuserve.script
IF P CHAIN df0:peoplelink.script

```

This example will link to and execute a script for logging onto CompuServe or People/Link, depending on whether "C" or "P" is selected from the keyboard.

Note: CHAIN cannot be executed from an AREXX script.

CLEAR

This command will clear the terminal screen, putting the cursor at the upper left-hand corner.

Format: **CLEAR**

Example:

```

DELAY 1
REPLY "^M"
WAIT "login:"
REPLY "yourname ^M"
WAIT "password:"
REPLY "yourpassword ^M"
WAIT "(vt100)"
REPLY "^ ^M"
WAIT %
CLEAR

```

This is a sample of a login script that you might use to access a UNIX system and emulate a VT100 terminal. Once the login is complete, and you have received the prompt (%) you might want to

CLOSE

clear the screen of the login procedure. Do this with the command **CLEAR**.

This will stop an ASCII file capture and close the file.

Format: **CLOSE**

Example:

```
WAIT "MAIL"
OPEN df0:mycapture.plink
REPLY "/READ ^M"
WAIT "MAIL"
CLOSE
```

This example is from a script file for capturing mail from People/Link. When the host sends the MAIL prompt, your mail has finished scrolling. You can then close the capture.

COLUMNS

This command selects the number of columns on your screen.

Format: **COLUMNS** 128/132

Example:

```
BAUD 2400
SET F8N1
ROWS 24
COLUMNS 128
WRAP OFF
SCREEN Full
EMULATOR Ansi
FONT ibmpc
```

COLUMNS is most useful when used from an ARexx script, since all parameters can also be saved from the PhoneBook QUICK gadget.

CRC

This command will set or reset the CRC option.

Format: **CRC** ON/OFF or: YES/NO

Example:

```
CRC ON
PROTOCOL Z
WAIT ">"
REPLY "sb hostfile ^M"
RECEIVE
```

This example shows a file download with ZMODEM and CRC-32, if supported by the host ZMODEM program (the UNIX program "sb" in this case).

DELAY

A DELAY will suspend the script file for a specified number of seconds. You may need a DELAY to slow down the execution of the script file in crucial points.

Format: **DELAY** *number*

DEPTH

Example:

```
DELAY 1
REPLY "^M"
WAIT "login:"
REPLY "yourname ^M"
WAIT "password:"
REPLY "yourpassword ^M"
```

This example will suspend the execution of the script for one second. A DELAY is needed to make sure the host is ready to receive the following REPLY, a carriage return.

This command will allow you to select the screen depth and therefore the number of colors (1 = two colors; 2 = four colors; 3 = eight colors).

Format: **DEPTH** *number (1/2/3)*

Example:

```
ROWS 24
DEPTH 3
SCREEN Full
EMULATOR Ansi
FONT ibmpc
```

DEPTH is most useful when used from an ARexx script, since all parameters can also be saved from the Phone Book QUICK gadget.

DIAL

By using this command, you can dial the telephone number with your modem. The DIAL command is used in conjunction with the IF command. If you want to dial the telephone in conjunction with the ON command, you must use REPLY. See the REPLY command for an explanation of this.

Format: **DIAL** *"string"*

Example:

```
AT 23:30
retry:
DIAL "7435000"
IF "NO CARRIER" GOTO retry
IF "CONNECT 1200" GOTO conn1200
IF "CONNECT 2400" GOTO conn2400
IF TIMEOUT GOTO retry
conn1200:
ECHO "got connected at 1200"
GOTO done
conn2400:
ECHO "got connected at 2400"
done:
```

This example is for an automatic dial and redial in case the phone is busy or connection, either at 1200 or 2400 baud, is not

made within the default timeout period of thirty seconds. This script is set up to start at 11:30 PM.

ECHO

This command will make a string appear on the screen during the login script.

Format: **ECHO** *"string"*

Example:

```
WAIT "login:"  
REPLY "yourname ^ M"  
WAIT "password:"  
REPLY "yourpassword ^ M"  
WAIT "(vt100)"  
REPLY "^ M"  
WAIT %  
ECHO "Login successful"
```

By using the ECHO command in this example, the phrase "Login successful" will appear on your screen when the login has been completed.

EMULATOR

This command will set the terminal emulator type.

Format: **EMULATOR** *name*

Example:

```
ROWS 24  
SCREEN Full  
EMULATOR Ansi
```

EMULATOR is most useful when used from an ARexx script, since all parameters can also be saved from the Phone Book QUICK gadget.

To set the emulator type to other terminals, use simply the name of the terminal--only the first three letters are significant: VT100, VT52, H19, ANSI, TEK, TALK, AND TTY.

EXIT

This will close the script and end the execution of A-Talk III.

Format: **EXIT**

Example:

```
WAIT "MAIL"  
OPEN df0:mycapture.plink  
REPLY "/READ ^ M"  
WAIT "MAIL"  
CLOSE  
HANGUP  
EXIT
```

This example will exit A-Talk III after closing the ASCII capture procedure.

Note: EXIT cannot be executed from an AREXX script.

FINISH

This command will shut down a host Kermit server mode.

Format: **FINISH**

Example:

```
KGET file.c
WAIT $
FINISH
```

FONT

This command will set the font type.

Format: **FONT** *name (s//i) (small//large//bmpr)*

Example:

```
ROWS 24
DEPTH 3
WRAP OFF
SCREEN Full
EMULATOR Ansi
FONT ibmpr
```

FONT is most useful when used from an ARexx script, since all parameters can also be saved from the Phone Book QUICK gadget.

GETLINE

This command will get, from the host, the number of characters specified. If the number is not specified, this command will get the entire line up to the carriage return or a maximum of 128 characters.

Format: **GETLINE** *number*

Example:

```
GETLINE 5
```

```
IF "hello" GOTO login
```

GOTO

This command jumps to a label at a different part of the script file. A label can be any word that is not part of the script language. The jumps may be forward or backward.

Format: **GOTO** *label*

Example:

```
ON "NO CARRIER" GOTO login
login:
HANGUP
REPLY "ATDP000-0000 ^M"
WAIT "login:"
REPLY "yourname ^M"
WAIT "password:"
REPLY "yourpassword ^M"
WAIT "(vt100)"
REPLY " ^M"
WAIT %
```

This example sets up "login" as a label by putting it on a separate line with a colon after it. The GOTO command will search for that label, jump to it and begin executing at this point.

The example given could be used if you're logging onto an erratic host that is in the habit of dropping the line and giving you the message "NO CARRIER". If this happens, you would want to start all over, redial and try to login again. The GOTO feature would allow you to do that automatically.

Note: GOTO cannot be executed from an AREXX script.

This is a simple command. It will drop the modem connection.

Format: **HANGUP**

Example:

```
REPLY "^M"
WAIT "$"
REPLY "logout ^M"
DELAY 1
HANGUP
```

This example is taken from a script exiting a VAX system running UNIX.

This command will display the Quick menu.

Format: **HELP**

Example:

```
HELP
```

This command will set the saving of icons.

Format: **ICONS ON/OFF YES/NO**

Example:

```
REPLY "kermit ^M"
WAIT "C-Kermit>"
REPLY "server ^M"
ICONS ON
MODE binary
PROTOCOL K
GET "hostfile"
FINISH
WAIT "C-Kermit>"
REPLY "exit ^M"
```

This example shows a kermit download in server mode, which also creates a project icon for the file, named hostfile.info.

This command will look for and check characters, either those typed in from the keyboard, those sent from the host, or those

HANGUP

HELP

ICONS

IF

returned from the modem. It is used in conjunction with the ASK, GETLINE, WAIT or DIAL commands.

Format: **IF** *character command*
IF *"string" command*

Example:

```
ASK "Protocol (K,X,Y,Z)?"
IF K PROTOCOL K
IF X PROTOCOL X
IF Y PROTOCOL Y
IF Z PROTOCOL Z
GETLINE 9
IF "C-Kermit" SEND file1
```

This example will ask you to select (from the keyboard) a "K", "X", "Y" or "Z" to designate a protocol. If you select K, the protocol you will be using to transfer files will be Kermit. If the nine characters from the host happen to be the prompt "C-Kermit", your Amiga will send a file, beginning a Kermit transfer of "file1".

IF TIMEOUT

This is a variation of the IF command. Use the TIMEOUT command to set up the situation, as in this example:

Format: **IF TIMEOUT** *command*

The "IF TIMEOUT" cannot be abbreviated!

Example:

```
TIMEOUT 20
login:
HANGUP
DIAL "000-0000"
WAIT "login:"
IF TIMEOUT GOTO login
REPLY "yourname ^ M"
WAIT "password:"
IF TIMEOUT GOTO login
REPLY "yourpassword ^ M"
WAIT "(vt100)"
IF TIMEOUT GOTO login
REPLY "^ M"
WAIT %
IF TIMEOUT GOTO login
```

This example sets up "login:" as a label by putting it on a separate line with a colon after it. The IF TIMEOUT GOTO command will be executed if the prompts ("login:", "password:", "(vt100)", %) are not sent by the host after twenty seconds. This login script will start over with the label "login:".

INPUT

This command is useful if you want to set up your script files without, for example, the password. You might want to do this if your script files are stored on diskettes that other people have access to. You can leave out the password or other important information from your login script and add an INPUT command that will ask for you to type it in at the proper moment. If you use this command, you must be present at the keyboard to type in the required information. This command is not for unattended operation.

Format: **INPUT** "*string*"

Example:

```
WAIT "@"  
INPUT "Type in TELENET access code now:"
```

With this example, the message "Type in TELENET access code now:" will appear on your screen, prompting you for the access code.

KGET

The KGET command will get files from the host Kermit.

Format: **KGET** *file.name*

Example:

```
PROTOCOL K  
KGET *.c
```

This example of the command will transfer all of the files with the suffix ".c" to your Amiga.

MODE

With this command you choose the type of transfers to perform, binary or text.

Format: **MODE** *type* (binary/text)

Example:

```
MODE text  
PROTOCOL Y  
WAIT "$"  
REPLY "upload *.c ^M"  
RECEIVE
```

This example refers to a YMODEM text transfer.

ON

The behavior of the ON command depends on the command after it.

Format: **ON** "*string*" *command*

Example:

```
ON "NO CARRIER" GOTO login  
login:  
HANGUP  
REPLY "ATDP000-0000 ^M"  
WAIT "login:"
```

```

REPLY "yourname ^M"
WAIT "password:"
REPLY "yourpassword ^M"
WAIT "(vt100)"
REPLY "^M"
WAIT %

```

If a "NO CARRIER" is received from the host at any point in this script, the WAIT commands will be aborted and the GOTO command will be executed. That means that the login script will be started over whenever the telephone line is dropped, as indicated by the message "NO CARRIER".

Example:

```

ON "CSE" ECHO "found"
DELAY 30

```

If, as in this example, the command is not a REPLY, the DELAY is aborted and the command is executed, followed by the next command after DELAY.

Example:

```

ON "Ima" REPLY "Hacker"
DELAY 30

```

If, as in this example, the command is a REPLY, the DELAY is continued.

A maximum of **three** ON strings can be installed at any one time. They will be executed in a circular fashion, returning to the first when the last one has been executed. The START and STOP commands clear all ON strings.

Example:

```

ON "NO CARRIER" GOTO login
ON "--more--" REPLY " "
ON ">" REPLY "^M"
login:
HANGUP
REPLY "ATDP000-0000 ^M"
WAIT "Name?"
SEND "Ima Hacker ^M"
WAIT "Password?"
SEND "cracker ^M"

```

In this example, the script will look for and execute a different command depending on what is received from the host. If the line disconnects, giving the message "NO CARRIER", the script will start over again, beginning from the label "login". Every time the string "--more--" is received, the script will reply with a space. Every time the prompt ">" is received, the script will reply with a carriage return.

Note: ON cannot be executed from an AREXX script.

OPEN

This command will start an ASCII file capture.

Format: **OPEN** *filename*

Example:

```
OPEN df0:mycapture.c
REPLY "cat mail ^M"
WAIT "$"
CLOSE
```

This example will start an ASCII file capture, putting it in a file named mycapture.c on disk df0:. Then, the example will suspend the capture, and resume the capture with the command OPEN. If, as in this case, the file name is missing, the capture will be appended to the previously SUSPENDED file.

PRINTER

This command will turn the printer on or off, initiating a printer capture. Remember that when you use this command, your session will be slowed to the speed of the printer.

Format: **PRINTER ON/OFF** or: **PRINTER YES/NO**

Example:

```
PRINTER ON
WAIT "end"
PRINTER OFF
```

PROTOCOL

This command is used for selecting the protocol for file transfers. It is used in conjunction with the SEND/RECEIVE commands.

Format: **PROTOCOL** *single character* (X, Y, B, Z, K, W or G)

Example:

```
PROTOCOL X
SEND filename
```

In this example, the protocol XMODEM will be used to send a file called "filename". An "X" stands for XMODEM, a "Y" stands for YMODEM, "B" stands for YMODEM Batch, "Z" for ZMODEM, "K" for Kermit, "W" for WXMODEM and "G" for YMODEM-g.

RECEIVE

Use this command to receive files from a host in any case except when using Kermit server mode, in which case the command to use is KGET.

Format: **RECEIVE** *filename*

Example:

```
MODE text
PROTOCOL Y
WAIT "$"
REPLY "upload *.c ^M"
RECEIVE
```

REDIAL

In this example you will be receiving from the host all the text files with the suffix "c". You will be using the YMODEM protocol for this batch transfer.

This command will select the redial type (o = once; t = ten times; c = continuously).

Format: **REDIAL** type (o/t/c)

Example:

```
REDIAL t
DIAL "000-0000"
REPLY "^M"
WAIT "login:"
```

The above example will dial the number up to ten times until connection is established.

REPLY

This command will send a string or a character to the host.

Format: **REPLY** "string"

Beginning and ending double quotes are required to send a string of characters. Quotes are not needed to send a single character (case insensitive). Use the caret (^) to send control characters, as in "^C" to send a control C. Two carets will transmit a single caret character. The double quote symbol (") can be sent by \".

Example:

```
WAIT "Name?"
REPLY "Ima Hacker ^M"
```

This example will reply with the name, Ima Hacker, followed by a carriage return (^M).

Since the DIAL command does not work in conjunction with the ON command, it is sometimes necessary to use the REPLY command for dialing. You can do this by adding an "ATDP" (for pulse dialing) or "ATDT" (for tone dialing) at the beginning of the phone number and a "^M" at the end as follows:

```
REPLY "ATDP000-0000 ^M"
```

REWIND

The REWIND command is very similar to the GOTO command, except that the GOTO will search for a label. The REWIND command will restart the script from the beginning of the script file. It is faster than jumping to a label at the beginning of a file because the computer will not have to search for the label.

Format: **REWIND**

Example:

```
ON "NO CARRIER" REWIND
REPLY "ATDP000-0000 ^M"
WAIT "login:"
REPLY "yourname ^M"
```



```
WAIT "password:"  
REPLY "yourpassword ^ M"  
WAIT "(vt100)"  
REPLY " ^ M"  
WAIT %
```

Whenever the line to the host drops, giving you the message "NO CARRIER", this login script will go back to the beginning, starting the login procedure all over again.

Note: **REWIND** cannot be executed from an AREXX script.

ROWS

This command will select the number of rows on your screen.

Format: **ROWS** 24/48/25/50

Example:

```
ROWS 24  
COLUMNS 128  
SCREEN Full
```

ROWS is most useful when used from an ARexx script, since all parameters can also be saved from the Phone Book QUICK gadget.

When using MoreRows and PAL Amigas, the maximum number of rows will be larger than 25 and 50.

RUN

This command will run a given program, with the possibility that the program will crash the system.

Format: **RUN** "string"

Example:

```
RUN "df0:clock"
```

This command runs an Amigados program which puts a clock on the Workbench screen.

RX

This command will call up either an ARexx string program or another script written in the REXX language.

Format: **RX** filename.rexx

Example:

```
RX rexx:marquis.rexx
```

The above example shows you how to call up an ARexx program (filename.rexx) from an A-Talk III script.

Format **RX** "string"

Example:

```
RX "do err = 1 to 5; say err errortext(err); end"
```

The above example is a command to call up an ARexx string program (use quotes around the string program).

Note: **RX** cannot be executed from an AREXX script.

SAY

This command, used in conjunction with the VOICE command, will speak a string. The VOICE must be ON for the SAY command to work.

Format: **SAY** *"string"*

Example:

```
VOICE ON
SAY "I am connected."
VOICE OFF
```

SCREEN

This command will allow you to set the screen type, Full or Workbench.

Format: **SCREEN** *Full/WB*

Example:

```
ROWS 24
COLUMNS 128
DEPTH 3
WRAP OFF
SCREEN Full
EMULATOR Ansi
FONT ibmpc
```

SCREEN is most useful when used from an ARexx script, since all parameters can also be saved from the Phone Book QUICK gadget.

SEND

This command will send files to the host. Remember to specify a protocol before giving this command.

Format: **SEND** *filename*

Example:

```
PROTOCOL K
SEND file.??
```

Notice that no quotes are used around the file name. You may use wildcard characters (* and ?) when sending YMODEM Batch, ZMODEM and Kermit files. This example will send all files with the names file.10, file.11, file.12, etc.

SET

This command will set the comm (communications) parameters by using the same format as in the phonebook. The four digit format refers to full or half duplex (F or H), number of bits (7 or 8), parity (E for even, O for odd, N for no parity, M for mark, and S for space), and number of stop bits (1 or 2).

Format: **SET** *value*

Example:

```
SET F8N1
COLUMNS 128
SCREEN Full
```

START

SET is most useful when used from an ARExx script, since all parameters can also be saved from the Phone Book QUICK gadget. This example will set the comm parameters to full duplex, 8 bits, no parity and one stop bit.

This command will execute a new script file, returning to the first file once the new one has been executed. Only **one** level of nesting is allowed.

Format: **START** *filename*

Example:

```
START df1:peoplelink.script  
WAIT "MAIL>"
```

In this example, you run your usual login script for People/Link, then start sending or receiving mail. In this way you can use standard login scripts in larger, customized, special purpose scripts.

Note: **START** cannot be executed from an AREXX script.

STOP

This will stop the execution of the script file.

Format: **STOP**

Example:

```
ON "NO CARRIER" STOP
```

This example will stop the script if the line drops and you get the "NO CARRIER" message.

Note: **STOP** cannot be executed from an AREXX script.

STRING

This command will send a keyboard or host string to AREXX as a RESULT string.

Format: **STRING**

Example:

```
OPTIONS RESULTS  
ADDRESS ATK  
ASK "Type a key"  
STRING  
ADDRESS  
IF RC=0 THEN SAY "KEY IS" RESULT
```

This is an example of a script using both ARExx and A-Talk commands. "OPTIONS RESULTS" is an ARExx command requiring A-Talk to send back a result string as well as error codes. "ADDRESS atk" will start sending commands to A-Talk. "ASK 'Type a key'" asks for a key to be typed. "STRING" will take what was typed and send it as a string to ARExx. "ADDRESS" will toggle back to ARExx, and start sending commands to ARExx. Therefore the "IF" string is in the ARExx language and means that if there are no errors, then see what is sent back; it is stored in the variable

SUSPEND

called "RESULT". "SAY" is an ARexx command that means print on the ARexx window.

The SUSPEND command will suspend, or temporarily stop, capturing incoming data.

Format: **SUSPEND**

Example:

```
OPEN df0:mycapture
WAIT $
REPLY "cat file1 ^ M"
WAIT $
SUSPEND
REPLY "cat file2 ^ M"
WAIT $
OPEN
REPLY "cat file3 ^ M"
WAIT $
CLOSE
```

This example will start capturing incoming data, storing it all under a file called "mycapture" on disk df0:.

TIMEOUT

This will set the timeout for WAITs which come after it. Set the timeout value in number of seconds. Remember that if you do not put a number down, the default is *forever!* That means if you are waiting, for example, for a "password:" prompt which never arrives, and you have not set the timeout with a number of seconds, the script will lock up if the prompt is never received. You can get out of this locked situation by selecting the **End Script** option in the **PHONE MENU**.

Format: **TIMEOUT** *number*

Example:

```
TIMEOUT 30
WAIT "User Name:"
```

In this example, the script file will wait (for the string "User Name:") for a maximum of thirty seconds before timing out.

VOICE

This command will start or stop the voice synthesizer. If you want the computer to say something during the execution of the login script, you must use VOICE in conjunction with the SAY command.

Format: **VOICE ON/OFF** or: **VOICE YES/NO**

Example:

```
VOICE ON
SAY "I am connected."
VOICE OFF
```

WAIT

This command suspends the script file until a particular string is received.

Beginning and ending double quotes are required when waiting for a string of characters. Quotes are not needed when waiting for a single character (case insensitive). Use the caret (^) to indicate a WAIT for control characters, as in ^C when waiting for a control C. Two carets (^ ^) indicate a WAIT for a caret character (^). When waiting for the double quote symbol ("), indicate it by a simple \".

Format: **WAIT** "string" or: **WAIT**

A WAIT command with no string following it will be an endless wait. If you have no idea how long it will take for your string to be received, you can use the ON command in conjunction with the WAIT command with no string following it. Of course, if the string after the ON command is never received, the script will be locked, waiting endlessly. You can get out of this locked situation by selecting the **End Script** option in the **PHONE MENU**.

Example:

```
ON "NO CARRIER" GOTO retry
ON "CONNECT 1200" GOTO conn1200
ON "CONNECT 2400" GOTO conn2400
retry:
REPLY "ATDP7435000 ^M"
WAIT
conn1200:
ECHO "got connected at 1200"
GOTO done
conn2400:
ECHO "got connected at 2400"
done:
```

This example will do a WAIT for an unspecified length of time. The WAIT will go on until one of three strings is received, "CONNECT 1200", "CONNECT 2400" or "NO CARRIER". If "CONNECT 1200" is received, the script will jump to the label "conn1200:" and ECHO the string "got connected at 1200" so you will know you have been connected at 1200 baud. Then the script will jump to the label "done:", bypassing the ECHO command for the "CONNECT 2400" possibility. If, on the other hand, "CONNECT 2400" is received, the script will jump to the label "done:" and ECHO the string "got connected at 2400" so you will know that you have been connected at 2400 baud. If "NO CARRIER" is received, the script will dial again. If none of the three strings is received, the script will be locked, waiting endlessly. You can get out of this locked situation by selecting the **End Script** option in the **PHONE MENU**. Note that this script uses the "REPLY" command as an alternative to

WRAP

"DIAL". You can do this by including "ATDP" at the beginning of your telephone number and "^M" at the end.

This command will select the auto-wrap feature.

Format: **WRAP ON/OFF** or **YES/NO**

Example:

```
ROWS 24
COLUMNS 128
WRAP OFF
SCREEN Full
EMULATOR Ansi
```

WRAP is most useful when used from an ARexx script, since all parameters can also be saved from the PhoneBook QUICK gadget.

XON

This command will set the X-on/X-off handshake.

Format: **XON ON/OFF** or **YES/NO**

Example:

```
XON ON
ASCIISEND "localfile"
WAIT ". ^M"
XON OFF
```

This example sends an ASCII file to a "slow" host that supports X-on/X-off.

A-Talk and A-Talk Plus to A-Talk III Script Conversion

The easiest way for you to convert your old A-Talk and A-Talk Plus script files to A-Talk III compatible files would be to use the Talk2Three tool provided on the A-Talk III installation disk. Just click on the Talk2Three icon. You will be asked for the names of the source and destination files, and the tool will do the rest of the work for you. If your old scripts have icons associated with them, they will be copied to the new scripts created by Talk2Three; if not, you will receive a message notifying you that no icon was found, and the new script will be created with no icon. Talk2Three can be used either from the workbench or from the CLI or SHELL, with little or no difference in performance.

If for some reason you do not want to use the Talk2Three tool, the table shown on the next page will help you make your conversions.

A-Talk or A-Talk Plus:

WAIT string

SEND string

DELAY number

TIMEOUT sec

WAITSEND string1-string2

DEBUG

VOICE ON/OFF

SAY string

A-Talk III:

WAIT "string"

REPLY "string"

DELAY number/1000
(divide your number by 1000)

TIMEOUT sec

TIMEOUT 20
begin:
WAIT "string1"
IF TIMEOUT GOTO continue
GOTO found
continue:
REPLY "string2"
GOTO begin
found:

no command

VOICE ON/OFF

SAY "string"

In your A-Talk and A-Talk Plus scripts you may have used special characters as well as the normal ASCII characters in your strings. To convert these to normal ASCII characters, use the chart in Appendix E. For example, \r for a carriage return should be converted to ^M.

SAMPLE SCRIPTS

Scripts Provided with the Phone Menu

Login script for
accessing CompuServe

```
DELAY 1
REPLY "^M"
WAIT "User ID: "
REPLY "12345,6789 ^M"
WAIT "Password: "
REPLY "yourpassword ^M"
```

(Type in your ID number in place of "12345,6789" and your password in place of "yourpassword".)

Login script for
accessing DIALOG:

```
DELAY 1
REPLY A
WAIT "Service: "
REPLY "dialog2 ^M"
WAIT "LOGON:"
DELAY 1
REPLY "X123YYYY"
```

(Type in your ID number in place of "X123YYYY".)

Login script for
accessing Dow Jones:

```
DELAY 1
REPLY "^M"
WAIT "name: "
REPLY "USERNAME ^M"
WAIT "Password: "
REPLY "yourpassword ^M"
```

(Type in your name in place of "USERNAME" and your password in place of "yourpassword".)

Login script for
accessing MCI Mail:

```
DELAY 1
REPLY "^M"
WAIT "name: "
REPLY "USERNAME ^M"
WAIT "Password: "
REPLY "yourpassword ^M"
```

(Type in your name in place of "USERNAME" and your password in place of "yourpassword".)

**Login script for
accessing NewsNet:**

```
DELAY 1
REPLY "^M^M"
WAIT "TERMINAL="
REPLY "^M"
WAIT @
REPLY "C 21566 ^M"
WAIT
REPLY "ID NET12345 ^M"
WAIT "Password?"
REPLY "yourpassword ^M"
```

(Type in your ID number in place of "12345" and your password in place of "yourpassword".)

**Login script for
accessing OAG
(Official Airline
Guide):**

```
DELAY 1
REPLY "^MA"
WAIT "log in: "
REPLY "OAG ^M"
WAIT "PASSWORD"
DELAY 1
REPLY "USR12345;PASSWD ^M"
```

(Type in your ID number and password in place of "USR12345;PASSWD".)

**Login script for
accessing OAG
through Dialog:**

```
DELAY 1
REPLY "^M"
DELAY 1
REPLY "^M"
WAIT @
REPLY "OAG ^M"
WAIT "PASSWORD"
DELAY 1
REPLY "USR12345;PASSWD ^M"
```

(Substitute your ID number and password, as above.)

**Login script for
accessing People/Link**

The following is a login script which accesses People/Link through Telenet. Comment out or delete the lines with the Telenet entries that don't apply to your modem setup. Note that the delay between characters is needed since Telenet won't recognize a sequence if it is entered too fast.

(Type in your ID number in place of "XYZ123" and your password in place of "MYPASS").

```
DELAY 1
# 2400 baud, 8 bits, no parity, 1 stop bit :
REPLY "@"
DELAY 1
REPLY "D"
DELAY 1
REPLY "^M"
```

```

# 2400 baud, 7 bits, even parity, 1 stop bit :
REPLY "@"
DELAY 1
REPLY "^M"
# 1200 baud, 8 bits, no parity, 1 stop bit :
REPLY "^M"
DELAY 1
REPLY "D"
DELAY 1
REPLY "^M"
# 1200 baud, 7 bits, even parity, 1 stop bit :
REPLY "^M"
DELAY 1
REPLY "^M"
#
WAIT "TERMINAL ="
REPLY "D1 ^M"
WAIT @
REPLY "C PLINK ^M"
WAIT "USER ID:"
REPLY "XYZ123 ^M"
WAIT "PASSWORD: "
REPLY "MYPASS ^M"

```

Examples of More Complex Scripts

Unattended downloads from UNIX

This is a script for unattended operation that will access a VAX running UNIX and will receive and capture files. We have used the "AT" command so that you can set up your computer to do this when phone rates are cheaper in the middle of the night.

```

AT 23:30
ON "NO CARRIER" GOTO login
login:
HANGUP
REPLY "ATDP000-0000 ^M"
TIMEOUT 30
WAIT "system?"
IF TIMEOUT GOTO login
REPLY "pollux ^M"
WAIT "login:"
IF TIMEOUT GOTO login
REPLY "username ^M"
WAIT "Password:"
IF TIMEOUT GOTO login
REPLY "userpassword ^M"
ECHO "Logged in! ^M"
WAIT "pollux.usc.edu(1):"

```

Script for accessing
BIX through Tymnet,
for reading mail and
conferences.

```
#
# Send a file with Ymodem batch
#
WAIT ")."
PROTOCOL B
REPLY "sb rbsb.c ^M"
WAIT "minutes"
RECEIVE
#
# Capture a file
#
OPEN file1
REPLY "cat mail.1M ^M"
WAIT ")."
CLOSE
REPLY " ^M"
WAIT ")."
REPLY "logout ^M"
DELAY 1
HANGUP
```

```
#-----
# BIX.script -- read mail and conferences
#-----
TIMEOUT 30
#
# Wake up Tymnet
#
ON "please log in: " GOTO login
DELAY 1
Begin:
REPLY a
DELAY 2
REPLY ^M
DELAY 2
GOTO Begin
#
# Login
#
login:
DELAY 1
REPLY "bix ^M"
WAIT "Name? "
REPLY "username ^M"
WAIT "Password: "
REPLY "userpassword ^M"
#
# Now logged in. Read mail, if any.
```

**Script for accessing an
IBM PC BBS**

```

ON ".More.." REPLY "^M"
# synchronize
WAIT "Conf/Topic"
WAIT ":"
#
# Read BIX mail
#
OPEN vd0:bixmail
REPLY "mail ^M"
# loop this for each message
moremail:
WAIT "Mail:"
REPLY ^M
ON "No more unread" GOTO donemail
WAIT "read/action:"
REPLY "quit ^M"
GOTO moremail
donemail:
REPLY "quit ^M"
WAIT ":"
REPLY "bye ^M"
WAIT "come back soon."
# close capture file
CLOSE
DELAY 2
HANGUP
#
# All done!
#
end:
STOP

```

```

CLEAR
ECHO "^M"
ECHO "
ECHO " / ^M"
ECHO " | ^M"
ECHO " | ^M"
ECHO " | ^M"
ECHO " | ^M"
ECHO " | ^M"
ECHO " | ^M"
ECHO " | ^M"
ECHO " | ^M"
ECHO " \ ^M"
ECHO "^M"
retry:
ECHO "^M ^M Dialing ... ^M"
REPLY "ATDT1-818-355-1336 ^M"

```

CABBS

The totally happening
L.A. Amiga hangout

```

#
next2:
ON "NO CARRIER" GOTO retry
on "CONNECT 2400" GOTO next3
on "ERROR" GOTO retry
WAIT
GOTO next2
#
next3:
REPLY "^M"
REPLY "^M"
WAIT "First"
#####
# Put your name and password here as shown #
#####
REPLY "Userfirstname Userlastname test ^M"
WAIT "call."
REPLY "^K"
WAIT "[y/n]"
REPLY "n ^M"
REPLY "^M"

```

Script for modem call back

Call back scripts are useful when getting in contact with companies that want to make sure their system is not infiltrated by hackers. These companies will not let your call go directly through, but will call you back after initial contact is made. This is a script to call work through the LeeMah dial back modem. Modem commands should be Hayes compatible, although the actual modem used was supra 2400.

```

#
# callback.scr by Rodger Anderson
# (rodger@hpdml93)
#
BAUD 2400
SET F8N1
# We don't want to wait forever.
TIMEOUT 30
#
Start:
HANGUP
REPLY "ATZ ^M"
#
# Now dial work
#
DELAY 1
# Replace phone number and code as appropriate.
REPLY "ATDT#####@#####; ^M"
# The @ is to tell the modem to wait for 5 seconds of

```

```

# quiet before sending more numbers.
WAIT "OK"
IF TIMEOUT GOTO Start
#
# We are onhook, hang up, and wait for callback.
#
REPLY "ATH ^M"
WAIT "RING"
IF TIMEOUT GOTO Start
#
# The phone rang, so answer it and hope its a
# modem. We answer in
# answer mode, although you may have to answer in
# originate mode.
REPLY "ATA ^M"
WAIT "CONNECT"
IF TIMEOUT GOTO Start
#
# We are connected.
# Remainder of script could be used to login to
# specific system, etc.

```

A mini-BBS script for unattended file transfers

To use MiniBBS, first modify the sign-on code and password to suit your preferences, and the modem control commands to suit your modem (these are set up for Hayes protocols, using an Okidata Okitel 1200). If your remote is unable to use ZModem, the script can easily be set up to use Kermit or YModem Batch; if your remote can only use XModem or YModem, that can also be accommodated; use the RCV: paragraph's send filename acceptor as a model for writing a receive file acceptor for the SND: paragraph.

Once the script is set up, set the CAPTURE OPTIONS to "CR-LF" and the BAUD RATE to whatever speed you would like to use, hook up your modem, and execute the script. It will then set the modem up for auto-answer on the first ring, and sleep until the modem answers the phone.

At your remote station, set up your communications program for the same baud rate as you set on your host system, ZModem protocol (unless you changed that), and HALF DUPLEX! The A-Talk III MiniBBS does not ECHO what you type. Call your modem, and if the sign-on message does not appear immediately, hit return.

Type in the sign-on code (the example in this script is "/ID FS47", which MUST be typed in all caps) and your password (AARDVARK is supplied, again in all caps). At this point, you will see the Main menu. Source for the MinBBS script is provided in the Script drawer and can be further enhanced, for example with the addition of ARexx support.

Please note that some of the lines of the script are wider than the page; such lines are continued on the following line, indented slightly.

```
#####
#
# MiniBBS 2.0, by J. Lampert & M. Papa #
#
# An A-Talk III mini-board script #
# for unattended file transfers #
#
#####
##### DECEMBER 18, 1988 #####
ECHO "***** IMPORTANT: Make sure CAPTURE OPTIONS is
      set for 'CR-LF' ^M ^J"
ECHO "***** IMPORTANT: Make sure ASCII SEND OPTIONS is
      set for 'CR/LF' ^M ^J"
ECHO "***** IMPORTANT: Make sure CHAR and LINE DELAY
      are set to 0 ^M ^J"
ECHO "***** IMPORTANT: Make sure HANDSHAKE is set for 'X-
      on/X-off' ^M ^J"
REPLY "AT S0 = 1 ^M"
WAIT "OK ^M"
#Modify these as needed for your modem's auto-answer cmd &
      confirmation signal.
ECHO "Modem set for auto-answer, 1 ring. ^M ^J"
SLEEP:
WAIT ^M
REPLY "^[[2J ^[[H"
ECHO "USER SIGNING ON! ^M ^J"
REPLY "*A-Talk III MiniBBS 2.0 READY. Sign on. ^M ^J"
REPLY "User ID: "
WAIT "/ID FS47 ^M"
ECHO ^J
#This can be modified to accept whatever sign-on
#code you may want to use, or to allow for multiple
#users (with their own accounts and passwords)
#or to eliminate the sign-on code entirely and simply
#ask for a password.
#The traditional "password grids" are in case you call
#from a TTY or TEK.
REPLY"*PASSWORD? ^M ^J##### ^H ^H ^H ^H ^H ^H ^H
      H ^HHHHHHHHHH"
REPLY"^H ^H ^H ^H ^H ^H ^H ^H ^H ^H ^H ^H ^H ^H ^H ^H
      ^H ^H ^H"
# WAIT "AARDVARK ^M"
#This should be modified to wait for whatever password you may
      wish to use.
REPLY "^J>Welcome to A-Talk III MiniBBS. ^M ^J"
DELAY 2
```

```

# MAIN Menu
MAINMENU:
REPLY "^[[2J^[[H"
REPLY "^M^J^J^J"
REPLY "MAIN MENU: Type '?' by itself for help ^M^J^J"
REPLY "F)ile section G)oodbye (logoff) S)hutdown ^M^J"
REPLY "B)ulletin    C)hange setup ^M^J"
REPLY "Select: "
GETLINE 1
IF F GOTO FILEMENU
IF G GOTO SIGNOFF
IF B GOTO BULLETIN
IF C GOTO CHANGE
IF S GOTO SHUTDOWN
IF ? GOTO GIVEHELP
GOTO MAINMENU

# Help menu
GIVEHELP:
REPLY "^M^J"
REPLY "HELP: ^M^J^J"
REPLY "Type F to switch to the file section ^M^J"
REPLY "Type G to logoff and leave the MiniBBS up ^M^J"
REPLY "Type S to logoff and shut down the MiniBBS ^M^J"
REPLY "Type ? to get this message ^M^J"
REPLY "Type B to get information about MiniBBS ^M^J"
REPLY " Type any key to return to the Main menu: "
GETLINE 1
GOTO MAINMENU

# Setup menu
CHANGE:
REPLY "^M^J"
REPLY "^J^J^J"
REPLY "CHANGE SETUP Menu. ^M^J^J"
REPLY "1) F8N1 ^M^J"
REPLY "2) F7E1 ^M^J"
REPLY "3) F7O1 ^M^J"
REPLY "Select: "
GETLINE 1
IF 1 SET F8N1
IF 2 SET F7E1
IF 2 SET F7O1
GOTO MAINMENU

```

```

# FILE Menu
FILEMENU:
REPLY "^[[2J^[[H"
REPLY "^J^J^J"
REPLY "The FILES Section. ^M^J^J"
REPLY "A)rea Change F)ile Titles T)ype      P)rotocol ^M^J"
REPLY "U)pload      D)ownload      S)how      M)ain menu ^M^J"
REPLY "Select: "
GETLINE 1
IF U GOTO RCV
IF D GOTO SND
IF P GOTO SELECT
IF T GOTO TPEMENU
IF S GOTO SHOWMENU
IF M GOTO MAINMENU
GOTO FILEMENU
#
# Remember to set ASCII SEND Options to CR/LF, no delays and
# X-on/X-off
#
# Show Menu
SHOWMENU:
REPLY "^M^J"
REPLY "^J^J^J"
REPLY "Select File Name ^M^J^J"
REPLY "*Please type in the word 'ASCII,' a space, and the name
      of the ^M^J"
REPLY " file you want MiniBBS to show. End with a CARRIAGE
      RETURN ^M^J"
REPLY "": "
OPEN RAM:SHOWFILE
WAIT ^M
REPLY "^M^J"
CLOSE
REPLY "Starting to send file -- Press CARRIAGE RETURN at end
      of file. ^M^J^J"
DELAY 5
START RAM:SHOWFILE
WAIT "^M"
REPLY "File Sent -- Press any key to continue: "
GETLINE 1
GOTO FILEMENU
# Type Menu
TPEMENU:
REPLY "^M^J"
REPLY "^J^J^J"
REPLY "Select File Type: ^M^J^J"
REPLY "T) Text ^M^J"

```

```
REPLY "B) Binary ^M ^J"
REPLY "Select: "
GETLINE 1
IF T MODE Text
IF B MODE Binary
GOTO FILEMENU
```

```
# Protocol Menu
SELECT:
REPLY "^M ^J"
REPLY "^J ^J ^J"
REPLY "Select a Protocol: ^M ^J ^J"
REPLY "B) YMODEM Batch ^M ^J"
REPLY "Z) ZMODEM ^M ^J"
REPLY "K) Kermit ^M ^J"
REPLY "G) YMODEM-g ^M ^J"
REPLY "Select: "
GETLINE 1
IF B PROTOCOL B
IF Z PROTOCOL Z
IF K PROTOCOL K
IF G PROTOCOL G
GOTO FILEMENU
```

```
SND:
REPLY "^M ^J"
REPLY "*Please type in the word 'SEND,' a space, and the name
      of the file ^M ^J"
REPLY " you want MiniBBS to send. End with a CARRIAGE
      RETURN ^M ^J"
REPLY ": "
OPEN RAM:SENDFILES
WAIT ^M
REPLY "^M ^J"
CLOSE
DELAY 5
START RAM:SENDFILES
REPLY "Transfer Complete ^M ^J"
GOTO FILEMENU
```

```
RCV:
REPLY "^M ^J"
REPLY "Begin receiving now ^M ^J"
RECEIVE
REPLY "Transfer Complete ^M ^J"
GOTO FILEMENU
```

```

SIGNOFF:
REPLY "^M^J"
REPLY "Goodbye? [y/n] :"
GETLINE 1
IF n GOTO MAINMENU
REPLY "Logged OFF HOLD ***** ^M^J"
REPLY "*GOOD-BYE ^M^J"
HANGUP
GOTO SLEEP

SHUTDOWN:
REPLY "^M^J"
REPLY "Logged OFF ***** ^M^J"
REPLY "*GOOD-BYE ^M^J"
REPLY "Shutting down modem Auto-Answer"
HANGUP
#Modify this line as needed to match your modem's auto-answer
#off command.
REPLY "AT S0=0 ^M"
STOP

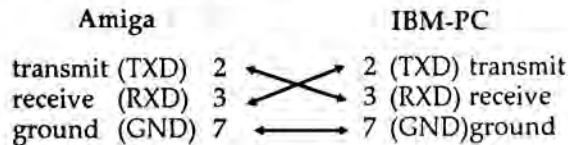
# Bulletin
BULLETIN:
REPLY "^M^J^J^J"
REPLY "    A-Talk III MinBBS ^M^J^J"
REPLY "This is a demonstration of the power of ^M^J"
REPLY "the A-Talk III script language. ^M^J"
REPLY "Instead of compiling the code to create ^M^J"
REPLY "a BBS-like program, the A-Talk III language ^M^J"
REPLY "can be used to create a full-featured ^M^J"
REPLY "BBS program, which can be completely ^M^J"
REPLY "customized to your own taste. ^M^J"
REPLY "This version provides file upload and ^M^J"
REPLY "download, and display of the contents ^M^J"
REPLY "of text files. Enjoy ^M^J"
REPLY "^M^J"
REPLY " Type any key to return to the Main menu: "
GETLINE 1
GOTO MAINMENU

```

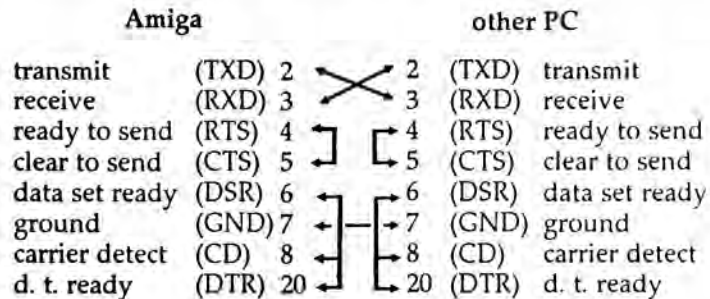
Appendix A: REQUIREMENTS

In order to use A-Talk III, you will need 1) a modem and a serial cable or 2) a null modem cable.

Use a null modem cable and no modems if you are connecting your Amiga to another personal computer. A null modem cable has, at minimum, three wires connected to pins 2, 3 and 7. If you would like to connect your Amiga to an IBM PC, you will need a null modem cable with a male connector for your Amiga and a female connector for the IBM PC. The following diagram will show you how this connection is made:



If you are using another type of personal computer and find that the above configuration doesn't work, you could try a more complicated connection of handshaking lines:



If you are using a modem, read the instructions that come with it. Buy the recommended cable from your Amiga dealer. Your modem instruction manual will also tell you how to set the switches inside the modem.

The **Generic Modem** option in the **DIAL** menu will work with any modem that correctly handles the **carrier detect** (CD) signal on pin 8 of the serial cable. If this pin is not connected to the modem, the generic modem option will not work since the software looks for

the **carrier detect** signal. When using the generic modem option, or the U. S. Robotics HST modem, the recommended cable connection looks like this:

Amiga Pin #		Modem Pin #
2	← (TXD) →	2
3	← (RXD) →	3
4	← (RTS) →	4
5	← (CTS) →	5
6	← (DSR) →	6
7	← (GND) →	7
8	← (CD) →	8
20	← (DTR) →	20

WARNING: On the Amiga 1000 serial connector, pins 14, 21 and 23 are used for power. Connecting these pins might fry your modem.

Appendix B: INSTALLATION OF A-TALK III

You have been provided with two A-Talk III diskettes. The first diskette (AT3) contains, among other things, the A-Talk III program itself. The second diskette (ATInstall) contains, among other things, installation programs.

Contents of Disk 1 (named AT3)

- 1) **Workbench 1.2**
- 2) **ATalk3** (A-Talk III executable)
- 3) **A-Talk III Project Icon**
- 4) **fonts:** (VT100, H19, Tek4014 USA fonts)
- 5) **Settings:** (Settings Drawer)
- 6) **Scripts:** (Scripts Drawer)

Contents of Disk 2 (named ATInstall)

- 1) **fonts:** (VT100, H19, Tek4014 foreign fonts)
 - Swedish/Finnish
 - Danish/Norwegian
 - German
 - French
 - Italian
- 2) **FD-Install** (Floppy Disk installation)
- 3) **HD-Install** (Hard Disk installation)
- 4) **TalkToy** (A-Talk voice option setting tool)
- 5) **Display** (IFF picture display tool)
- 7) **Talk2Three (A-Talk Script Translator)**
- 8) **Tools:** (Drawer with PD & Shareware comm programs)

Disk 1 (AT3) comes preinstalled with USA fonts. If you are using a floppy-based USA-NTSC Amiga you can boot and use a backup copy of this disk. If you are using a PAL Amiga or if you are installing A-Talk III on a hard disk, you must execute one of the installation programs provided in Disk 2.

Hard Disk Installation

When using A-Talk III with a hard disk, whether the Amiga is NTSC or PAL, you must use this installation procedure.

Place a copy of Disk 2 in any drive. The program will ask you to insert Disk 1 at the proper time.

Open the disk icon by double clicking on it. A number of tools will appear.



Double click on the HD-Install icon. The program will ask you the following questions:

On which <partition> do you want to install A-Talk III?

Select Hard Disk Partition [dh0:, dh1:, etc..]

Which country-dependent fonts do you want to install on your system disk?

[U] USA

[S] Swedish/Finnish

[D] Danish/Norwegian

[G] German

[F] French

[I] Italian

Enter country-code [press RETURN after code]:

After these selections, a new drawer named "at" will be created in the selected hard disk partition if not already present. Subsequently the A-Talk III executable and supporting directories (Settings, Scripts) will be copied into the new drawer. The A-Talk III fonts will also be copied to your hard disk FONTS: directory.

To complete the installation, you must modify your Startup-Sequence file in the S: directory. Add the following lines to your startup file:

```
IF EXISTS disk:at
  Assign AT3: disk:at
  Path AT3: add
ENDIF
```

The partition named *disk:* is the one that you entered at the beginning of the hard disk installation. *This is required so that A-Talk can find its settings, phone book and script files.*

Floppy Disk Installation for Foreign Countries

When using A-Talk in a non-NTSC USA Amiga, you must use this installation procedure. Place a copy of Disk 2 in any drive. Open Disk 2 by double clicking on it. A number of tools will appear.

Double click on the FD-Install icon. This program will ask you the following question:

Which country-dependent fonts do you want to install in the A-Talk III disk?

- [U] USA
- [S] Swedish/Finnish
- [D] Danish/Norwegian
- [G] German
- [F] French
- [I] Italian

Enter country-code [press RETURN after code]:

After you choose the country-code, the selected set of fonts will be copied to Disk 1.

In order to use the distributed disk in a foreign country, you will also have to select the proper keyboard key map for your country. To do this, you will have to use the SetMap tool in the system drawer on Workbench. First, open the system drawer and select the SetMap icon; choose Info from the Workbench menu, and select the Add gadget. Then select the String Gadget between the Add Gadget and the words Tool Types and type "KEYMAP=" followed by the code name of the new key map you want. The key maps available are:

- d for German keyboards
- e for Spanish keyboards
- f for French keyboards
- gb for British keyboards
- i for Italian keyboards
- is for Icelandic keyboards
- s for Swedish/Finnish keyboards
- usa for standard United States keyboards
- usa0 for the key map provided with the 1.1 release of the Amiga system software. Some programs may require this key map.
- usa2 for Dvorak keyboards

Press Return when you have finished entering the name of the key map. Select the Save Gadget. When the Workbench reappears, point to the SetMap icon and double-click the selection button.

Multiple serial ports, multiple printer ports, and startup scripts are supported, whether invoked from the CLI or Workbench.

Multiple Serial, Multiple Printer and Script Support

•Tool Types

A new serial device can be selected by setting the DEVICE and the UNIT in the icon ToolType. Use the Info Workbench command to set one or both of DEVICE and UNIT.

The syntax is as follows:

a) Different Serial DEVICE

DEVICE = name

Example:

DEVICE = sevenserial.device

If not defined, the "default" serial device is "serial.device".

b) Different Serial UNIT Number

UNIT = n

where n = 0,1,2,3 ...

Example:

UNIT = 2

If not defined, the default serial unit is 0 (zero).

c) Different Printer DEVICE

PRINTDEV = name

Example:

PRINTDEV = newpar.device

If not defined, the "default" printer device is "printer.device".

d) Different Printer UNIT Number

PRINTUNIT = n

where n = 0,1,2,3 ...

Example:

PRINTUNIT = 2

If not defined, the default printer unit is 0 (zero).

e) Startup Script

FROM = name

where name is the pathname for your startup script file

Example:

FROM = COMPUSERVE.SCR

if not defined, no script will be invoked on startup.

•Tool and Project Icons

The selection of "options" such as DEVICE and UNIT can be done on the A-Talk III TOOL icon or on a separate PROJECT icon. In this case the DEFAULT TOOL must also be set to AT3:ATalk3. A sample PROJECT icon that uses this feature is provided on the distribution disk, named "MultiSer".

If you have 7 ports, for example, it is possible to use all seven of them, AT THE SAME TIME, just by duplicating the Multiser icon and setting a different UNIT number for each one of them.

CLI Selection of Options

- **Format:**

(using the same notation as the AmigaDOS User's Manual)

ATALK3 [FROM name] [DEVICE name] [UNIT]
[PRINTDEV name] [PRINTUNIT]

- **Template:**

ATALK3 "FROM/K,DEVICE/K,UNIT/K,PRINTDEV/K,PRINTUNIT/K"

- **Purpose:**

Run A-Talk III from the CLI and select initial script, serial and printer devices and respective units. Typing

ATalk3 ?

in a CLI shows the Format of the A-Talk III command line.

- **Path:**

AT3:

- **Example:**

ATALK3 FROM bix.scr DEVICE sevenserial.device UNIT 2

Everything in the brackets "[]" is optional. "/K" must be specified.

Appendix C: A-TALK AND AREXX

ARexx is a high-level language, an implementation of REXX, which can extend the capabilities of other programs used on the Amiga. Macros, scripts or execs can be created with ARExx to customize programs and combine different applications. If you want to install ARExx on your Amiga, you can order it from:

William S. Hawes
P.O. Box 308
Maynard, MA 01754
(617) 568-8695

A-Talk III sets up a command interface to communicate with the resident process of ARExx. A-Talk's own script language can be extended and customized by users who take advantage of ARExx.

The resident process of ARExx must be up and running before any ARExx programs can be run. Use the command "rexxmast" to start it up. A small window will pop up, and then ARExx will disappear into the background until it is used. If you use ARExx often, you might want to put the rexxmast command in the start-up sequence that resides in the system S:directory. The resident process can be closed with the command "rxc". Since ARExx waits in the background, there is no reason to close ARExx unless you are very short on memory.

An ARExx macro program can be executed from the CLI by typing the command "rx" followed by the program name and any arguments. To avoid confusion, it would be best to give all ARExx programs the file extension ".rexx" to distinguish them from programs written in other languages. ARExx commands and commands from the A-Talk III script languages should be given the file extension ".atk".

ARExx macro programs can also be executed from the A-Talk III menu. Choose **ARExx** from the **PHONE** menu. A requester will pop up asking you for a file name. Again, use the extension ".rexx" for programs containing only ARExx commands, and the file extension ".atk" for programs containing ARExx commands and commands from the A-Talk III script language.

Once the REXX: device is defined, it will be searched after the current directory when looking for an ARExx program. You can

designate one directory for all ARExx programs with the assign command. "assign rexx: sys:rexx" defines the REXX device as the :rexx directory on the system disk.

By default, all commands in an ARExx macro will be interpreted as functions or instructions of the REXX programming language. When writing programs that contain ARExx commands and commands for other programming languages, you will have to "address" the other "host" (message port associated with the other application) that the command is intended for. Whenever a new host is addressed, the new host becomes the "current" host, and the previously "current" host becomes the "previous" one. Other previous host values are lost; ARExx maintains only two host addresses at once. If, for example, you are writing an ARExx script that also uses commands from the A-Talk III script language, by default all commands will be addressed to the ARExx programming language. When you want a command to be addressed to A-Talk III, use this command:

```
address ATK
```

When you want commands to revert back to interpretation by ARExx, simply use the command:

```
address
```

And the commands will then be directed to ARExx.

When writing scripts with the A-Talk III script language, it may be useful to take advantage of the powerful ARExx language by inserting commands directed at the ARExx interpreter. This is especially useful if you want to use interactive commands and variables. From an A-Talk III script, you can call up either an ARExx string program or another script written in the REXX language. Both are called up with the command "RX".

The following is an example of how to call up an ARExx program (filename.rexx) from an A-Talk III script:

```
RX rexx:marquis.rexx
```

This is an example of the command to call up an ARExx string program (use quotes around the string program):

```
RX "do err = 1 to 5;say err errortext(err);end"
```

When calling up ARExx from A-Talk III for interactive procedures, no ARExx window will come up for messages unless A-Talk III has been started from the CLI. You can remedy this by using the following ARExx program, which will open a special window for ARExx input/output messages:

```
/* test of interactive window */
trace
open('out','con://640/60/Interactive A-Talk Command
Selection')
do i = 1 to 20
```

```

writech('out','Please enter A-Talk Script Command: ')
instr = readln('out')
writech('out', 'Sending' instr '....')
address ATK
instr
address
writeln('out', 'Done')
end

```

This example will first open an ARexx window, then will ask the user to type in an A-Talk script command, which will be sent to A-Talk III. The script then reverts to ARexx. This script has a loop which will execute at most 20 times unless aborted with ^C. The script allows you to test any A-Talk script command from ARexx.

Return Codes

A-Talk III will send return codes back to ARexx at the end of the execution of each script command, so that processing actions can be altered when errors occur. The return codes sent by A-Talk III follow the ARexx model of error severity level. The return codes are:

- 0 = no errors occurred
- 1 = invalid script command
- 5 = error executing script command
- 10 = ARexx library not found

Errors of severity 5 would indicate conditions such as "timeout on WAIT," "DIAL could not get carrier," "Cannot open capture file."

The error severity level returned by A-Talk III will be assigned to the special variable RC in the ARexx macro program.

Quoted Strings

When you want to send an A-Talk script command from an ARexx macro and the command includes a "quoted" string, you should prepend and append single quotes to the command, as in:

'WAIT "hello world"'

The same is true for commands such as SAY and ECHO that are valid in both the A-Talk and ARexx script languages:

SAY 'hello'	ARexx command
'SAY "hello"'	A-Talk command

or

ECHO 'bye'	ARexx command
'ECHO "bye"'	A-Talk command

Appendix D: THE AMIGA KEYS AND SPECIAL FUNCTION KEYS

By now you have no doubt noticed the letter "A" followed by another letter after various menu items. These are the Amiga keys. These keys will allow you to use your keyboard instead of your mouse to give commands. Press the RIGHT Amiga key and the letters shown below.

For Menu Item: Press:

•Project

NewCLI	N
Colors	*
QuickMenu	M
Fun Keys	K
Printer	Off 0
	On 1
Voice	Now .
Start/Stop Timer	\
Quit	Q
Exit	E

•Phone

Dial	D
Script	X
ARexx	A
Phonebook	T
Hangup	H
Send Break	B

•File

Capture	Open O
	Close C
	Clear L
	View V
	Cut -
	Copy =
	Paste +
ASCII Send	/

For Menu Item: Press:

•Transfer

Receive	R
Send	S
Get	G
Finish	F
Get DIR	@

•Tektronix

Options	?
Display	Y
Save IFF	I
Save Draw	W
Save Plot	U
Print	P
Zoom In	Z
Replay	J

Special Function Keys:

DEC VT-100
Emulation:

Various terminals have a variety of special function keys, such as PF1-PF4 on the DEC VT-100, and the RESET/PAGE key on a TEKTRONIX 4014. These keys are implemented as follows:

The VT-100 emulator redefines the keypad of an AMIGA 500/2000 keyboard as if it were the standard VT-100 keypad

PF1	PF2	PF3	PF4
7	8	9	—
4	5	6	,
1	2	3	Enter
0		.	

(above). On the AMIGA 1000, the keys perform as labeled; since the keypad does not include the top row or "+" key, the PF keys, if needed, must be set up as function keys, as shown in Chapter 9.

DEC VT-52, ANSI,
TTY, TALK, and H19
Emulations:

These emulations all use the same keypad arrangement as the VT-100; they differ only in PF key and "application mode" escape sequences.

TEKTRONIX 4014
Emulation:

This emulation is as above, except that the keypad Enter key is assigned the RESET/PAGE function, as below. PAGE clears the

7	8	9	—
4	5	6	,
1	2	3	RESET PAGE
0		.	

screen; RESET (Shift-PAGE) resets the terminal without clearing.

Appendix E: ASCII CHARACTER SET

Dec	Oct	Hex	Character	Dec	Oct	Hex	Character
0	0	0	^@ NUL	39	47	27	'
1	1	1	^A SOH	40	50	28	(
2	2	2	^B STX	41	51	29)
3	3	3	^C ETX	42	52	2A	*
4	4	4	^D EOT	43	53	2B	+
5	5	5	^E ENQ	44	54	2C	,
6	6	6	^F ACK	45	55	2D	-
7	7	7	^G BEL	46	56	2E	.
8	10	8	^H BS	47	57	2F	/
9	11	9	^I HT	48	60	30	0
10	12	A	^J LF	49	61	31	1
11	13	B	^K VT	50	62	32	2
12	14	C	^L FF	51	63	33	3
13	15	D	^M CR	52	64	34	4
14	16	E	^N SO	53	65	35	5
15	17	F	^O SI	54	66	36	6
16	20	10	^P DLE	55	67	37	7
17	21	11	^Q DC1 X-ON	56	70	38	8
18	22	12	^R DC2	57	71	39	9
19	23	13	^S DC3 X-OFF	58	72	3A	:
20	24	14	^T DC4	59	73	3B	;
21	25	15	^U NAK	60	74	3C	<
22	26	16	^V SYN	61	75	3D	=
23	27	17	^W ETB	62	76	3E	>
24	30	18	^X CAN	63	77	3F	?
25	31	19	^Y EM	64	100	40	@
26	32	1A	^Z SUB	65	101	41	A
27	33	1B	^[ESC	66	102	42	B
28	34	1C	^\ FS	67	103	43	C
29	35	1D	^] GS	68	104	44	D
30	36	1E	^^ RS	69	105	45	E
31	37	1F	^_ US	70	106	46	F
32	40	20	Space	71	107	47	G
33	41	21	!	72	110	48	H
34	42	22	"	73	111	49	I
35	43	23	#	74	112	4A	J
36	44	24	\$	75	113	4B	K
37	45	25	%	76	114	4C	L
38	46	26	&	77	115	4D	M

Dec	Oct	Hex	Character
-----	-----	-----	-----------

78	116	4E	N
79	117	4F	O
80	120	50	P
81	121	51	Q
82	122	52	R
83	123	53	S
84	124	54	T
85	125	55	U
86	126	56	V
87	127	57	W
88	130	58	X
89	131	59	Y
90	132	5A	Z
91	133	5B	[
92	134	5C	\
93	135	5D]
94	136	5E	^
95	137	5F	_
96	140	60	`
97	141	61	a
98	142	62	b
99	143	63	c
100	144	64	d
101	145	65	e
102	146	66	f
103	147	67	g
104	150	68	h
105	151	69	i
106	152	6A	j
107	153	6B	k
108	154	6C	l
109	155	6D	m
110	156	6E	n
111	157	6F	o
112	160	70	p
113	161	71	q
114	162	72	r
115	163	73	s
116	164	74	t
117	165	75	u
118	166	76	v
119	167	77	w
120	170	78	x
121	171	79	y
122	172	7A	z

Dec	Oct	Hex	Character
-----	-----	-----	-----------

123	173	7B	{
124	174	7C	
125	175	7D	}
126	176	7E	~
127	177	7F	DEL
128 - 159			UNASSIGNED
160	240	A0	Non-BreakingSpace
161	241	A1	¡
162	242	A2	¢
163	243	A3	£
164	244	A4	¤
165	245	A5	¥
166	246	A6	¦
167	247	A7	§
168	250	A8	¨
169	251	A9	©
170	252	AA	ª
171	253	AB	«
172	254	AC	¬
173	255	AD	Soft Hyphen
174	256	AE	®
175	257	AF	¯
176	260	B0	°
177	261	B1	±
178	262	B2	²
179	263	B3	³
180	264	B4	´
181	265	B5	µ
182	266	B6	¶
183	267	B7	•
184	270	B8	¸
185	271	B9	¹
186	272	BA	º
187	273	BB	»
188	274	BC	¼
189	275	BD	½
190	276	BE	¾
191	277	BF	¿
192	300	C0	À
193	301	C1	Á
194	302	C2	Â
195	303	C3	Ã
196	304	C4	Ä
197	305	C5	Å
198	306	C6	Æ

Dec	Oct	Hex	Character
199	307	C7	Ç
200	310	C8	È
201	311	C9	É
202	312	CA	Ê
203	313	CB	Ë
204	314	CC	Ì
205	315	CD	Í
206	316	CE	Î
207	317	CF	Ï
208	320	D0	Ð
209	321	D1	Ñ
210	322	D2	Ò
211	323	D3	Ó
212	324	D4	Ô
213	325	D5	Õ
214	326	D6	Ö
215	327	D7	×
216	330	D8	Ø
217	331	D9	Ù
218	332	DA	Ú
219	333	DB	Û
220	334	DC	Ü
221	335	DD	Ý
222	336	DE	Þ
223	337	DF	ß
224	340	E0	à
225	341	E1	á
226	342	E2	â
227	343	E3	ã
228	344	E4	ä
229	345	E5	å
230	346	E6	æ
231	347	E7	ç
232	350	E8	è
233	351	E9	é
234	352	EA	ê
235	353	EB	ë
236	354	EC	ì
237	355	ED	í
238	356	EE	î
239	357	EF	ï
240	360	F0	ð
241	361	F1	ñ
242	362	F2	ò
243	363	F3	ó

Dec	Oct	Hex	Character
244	364	F4	ô
245	365	F5	õ
246	366	F6	ö
247	367	F7	÷
248	370	F8	ø
249	371	F9	ù
250	372	FA	ú
251	373	FB	û
252	374	FC	ü
253	375	FD	ý
254	376	FE	þ
255	377	FF	ÿ

A-Talk and A-Talk Plus character conversions

<u>A-Talk, Plus</u>	<u>A-Talk III</u>
\\	\
\r	^M
\n	^J
\t	^I
\b	^H
\s	space
\x	^Q
\###	(not applicable)
BREAK	(use BREAK command)
^	^^

International Character Sets

Most mainframe systems do not use the AMIGA or IBM-PC international character sets, but use special national customized versions of ASCII. These versions remap certain less-used characters of standard 7-bit ASCII into the various special characters required by the languages. These character sets are reproduced below.

Country

Decimal	Hex	US	France	Germany	UK	Denmark	Sweden	Italy	Spain
35	23	#	#	#	£	#	#	#	Pt
36	24	\$	\$	\$	\$	\$	Ö	\$	\$
64	40	@	à	§	@	@	È	@	@
91	5B	[°	Ä	[Æ	Ä	°	í
92	5C	\	ç	Ö	\	Ø	Ö	\	ñ
93	5D]	§	Ü]	Å	Å	é	¿
94	5E	^	^	^	^	^	Ü	^	^
96	60	,	,	,	,	,	é	ù	,
123	7B	{	é	ä	{	æ	ä	à	{
124	7C		ù	ö		ø	ö	ò	
125	7D	}	è	ü	}	å	å	è	}
126	7E	~	..	ß	~	~	ü	í	~

GLOSSARY

Portions of this glossary have been reprinted with permission from *Black Box Catalog*.

ACK - A control character sent to acknowledge that a transmission block has been received.

ALT keys (Alternative Keys) - Two keys at the bottom of the keyboard that work, like shift keys, when pressed with other keys.

AmigaDOS - The disk operating system for the Amiga.

Amiga Keys - The two keys beside the space bar that execute commands when pressed along with another key. The right Amiga key executes menu commands.

ANSI (American National Standards Institute) - The principal standard development body in the U.S.A. ANSI is a nonprofit, nongovernmental body supported by over 1000 trade organizations, professional societies and companies. U.S.A.'s member body to ISO (International Standards Organization).

ASCII (American Standard Code for Information Interchange) - Pronounced "as'kee". A seven-bit-plus-parity code established by ANSI to achieve compatibility between data services.

Asynchronous Transmission - Transmission in which time intervals between transmitted characters may be of unequal length. Transmission is controlled by start and stop bits at the beginning and end of each character.

Backspace - The key with the left pointing arrow at the top right of the keyboard. This key moves the cursor back, erasing data.

Baud - Unit of signaling speed. The speed in baud is the number of discrete conditions or events per second. If each event represents only one bit condition, baud rate equals bps. When each event represents more than one bit (e.g. dibit), baud rate does not equal bps.

BCC (Block Check Character) - The result of a transmission verification algorithm accumulated over a transmission block. It is normally added to the end; e.g. CRC, LRC.

BEL - A control character that is used when there is a need to call for attention; it may control alarm or attention devices.

Bell 212 - An AT&T modem-providing, full-duplex, asynchronous or synchronous, 1200 bps data transmission for use on the public telephone network.

Bit (Binary Digit) - Contraction of "binary digit", the smallest unit of information in a binary system; a one or zero condition.

Block Check Character - Used to check transmission accuracy, a character transmitted by the sender after each message block and compared with a block check character computed by the receiver.

BPS (Bits Per Second) - Unit of data transmission rate.

Boot - To start a computer with a master program that will enable it to use other programs.

Buffer - A temporary-storage device used to compensate for a difference in data rate and flow between two devices (typically a computer and a printer); also called a spooler.

Byte - A binary element string functioning as a unit, usually shorter than a computer "word". Eight-bit bytes are most common. Also called a "character".

CAN (Cancel) - A character indicating that the data preceding it is in error and should be ignored.

Carrier Detect - Same as Received Line Signal Detector. An RS-232 modem signal that indicates to an attached terminal that the modem is receiving a signal from a remote modem.

Character - Letter, numeral, punctuation, control figure or any other symbol contained in a message.

Clear To Send (CTS) - Modem interface signal that indicates to the DTE device to begin transmission.

CLI (Command Line Interpreter) - A way to operate the computer without the mouse. Commands are typed directly into the computer.

Command - Any instruction given to the computer, whether typed or selected from a menu.

Communications Protocol - The rules governing the exchange of information between devices on a data link.

Console - The device used by an operator, system manager or maintenance engineer to monitor or control computer, system or network performance.

CR (Carriage Return) - An ASCII or EBCDIC control character that moves the cursor or print mechanism to the left margin.

CRC (Cyclic Redundancy Check) - An error-detection scheme in which the block check character is the remainder after dividing all the serialized bits in a transmission block by a predetermined binary number.

CRT (Cathode Ray Tube) - A television-like picture tube used in a terminal; commonly used as a synonym for CA1 terminal.

CTS (Clear to Send) - An RS-232 modem interface control signal which indicates that the attached DTE may begin transmitting.

Cursor Keys - Arrow keys (on the keypad) which move the cursor around on the screen. By pressing these keys and the right Amiga key, you can also move the **Pointer**.

Default - The option that will be executed unless the user specifies another option.

DEL (Delete Key) - A key at the top of the keyboard which moves the cursor backwards, deleting data.

Delete - The removal of data by mouse selection (e.g. **Delete** under the **Phone Menu**), use of the CLI delete command, the **Delete** or **Backspace** keys.

Dibit - A group of two bits. The four possible states for a dibit code are 00, 01, 10, and 11.

Display - To show an image on the monitor screen, or the monitor itself.

Downloading - The process of sending configuration parameters, operating software or related data from a central source to remote stations.

Drawers - Amiga Workbench storage places for programs or files. Double click on the Drawer icons will open a Window displaying the icons inside it.

Drive - A hardware device (Floppy Disk Drive or Hard Disk Drive) for storing and retrieving data.

DTE (Data Terminal Equipment) - Devices acting as data source, data sink, or both.

DTR (Data Terminal Ready) - Modem interface signal which alerts the modem that the DTE device is ready for transmission.

Dumb Terminal - Both hard-copy and VDT type ASCII asynchronous terminals that do not use a data transmission protocol and usually send data one character at a time.

Emulation - The imitation of a computer system, performed by a combination of hardware and software, that allows programs to run between incompatible systems.

ENQ (Enquiry) - A transmission control character used as a request for a response from a remote station.

EOT (End of Transmission) - A transmission control character used to indicate the conclusion of the transmission of one or more texts.

Error Detection - An arrangement that senses flaws in received data by examining parity bits, verifying block check characters or using other techniques.

ESC (Escape) - 1) A control character which is used to provide additional control functions. It alters the meaning of a limited number of continuously following bit combinations. 2) The Escape Key, which interrupts certain procedures (such as Dial and Login).

ETB (End of Transmission Block) - A transmission control character used to indicate the end of a transmission block of data.

ETX (End of Text) - A transmission control character that terminates a text.

Even Parity - A "dumb" terminal data verification method in which each character must have an even number of "on" bits.

FDX (Full Duplex) - Simultaneous, two-way, independent transmission in both directions (4-wire).

File - A collection of related data on a disk or other mass storage.

File Name - The name used by the computer to call up or refer to a particular file.

Flow Control - The procedure for regulating the flow of data between two devices; prevents the loss of data once a device's buffer has reached its capacity.

Function Keys - The ten keys at the top of the keyboard. Twenty commands are executed by using these keys shifted (simultaneously pressing the shift key) and unshifted.

Gadget - A device, represented graphically on the Window or Requester, which communicates with the program. Use gadgets by clicking on them with the Select (left) mouse button.

Handshaking - Exchange of predetermined signals between two devices establishing a connection. Usually part of a communications protocol.

Help - The support accessed by the **Help Key**. Technical support is also available from Oxxi, Inc. at (213) 427-1227 from 9:00 AM to 5:00 Pacific Time, Monday through Friday.

HDX (Half Duplex) - Transmission in either direction, but not simultaneous (2-wire).

Icon - An image on the Workbench that represents a Disk Drive, Program, File, etc.

IFF (Interchange File Format) - An Amiga software standard for interchanging graphic and sound files among various programs.

Kickstart - The part of the operating system needed to boot the Amiga. The Amiga 500 and 2000 have Kickstart embedded in the hardware ROM, while the Amiga 1000 must have Kickstart loaded from disk into a protected area of RAM (RAM-ROM).

Load - To put a disk program into the computer.

Mainframe - A large-scale computer system that can house comprehensive software and several peripherals.

Mark - Presence of signal. In telegraph communication, a mark represents the closed condition or current flowing. A mark impulse is equivalent to a binary 1.

Menu - The list of available software functions for selection by mouse or Amiga Keys. A menu is displayed when you move the Pointer to the name of the Menu on the Menu bar while pressing the Menu (right) mouse button.

Modem (Modulator-Demodulator) - A device used to convert serial digital data from a transmitting terminal to a signal suitable for transmission over a telephone channel, or to reconvert the transmitted signal to serial digital data for acceptance by a receiving terminal.

Multi-tasking - The ability of a computer to run more than one program at a time.

NAK (Negative Acknowledgement) - A control character used to indicate that the previous transmission block was in error and the receiver is ready to accept retransmission.

Null Modem - A device that connects two DTE devices directly by emulating the physical connections of a modem.

Operating System - The software which runs the hardware devices (monitor, disk drives, etc.) and application programs.

Packet - A group of bits (including data and call control signals) transmitted as a whole on a packet-switched network. Usually smaller than a transmission block.

Parity Bit - A bit that is set a "0" or "1" in a character to ensure that the total number of 1 bits in the byte is even or odd.

Parity Check - The addition of noninformation bits that make up a transmission block to ensure that the total number of 1s is always either even (even parity) or odd(odd parity); used to detect transmission errors.

Password - A code word to ensure that a user's account (on a mainframe or network) remains private.

Pointer - The arrow, or object, which moves on the screen as you move the mouse. The Pointer is used for selecting icons and gadgets and for choosing menu items.

Port - A computer interface capable of attaching to a modem for communication with a remote terminal.

Protocol - A formal set of conventions governing the formatting and relative timing of message exchange between two communicating systems.

Public Switched Network - Any switching communications system-- such as Telex, TWX or public telephone networks--that provides circuit switching to many customers.

Pulse Dialing - Older form of phone dialing, utilized in rotary dial phones.

RAM (Random Access Memory) - Semiconductor read-write volatile memory. Data stored is lost if power is turned off.

Redundancy Check - A technique of error detection involving the transmission of additional data related to the basic data in such a way that the receiving terminal, by comparing the two sets of data, can determine the probability that an error has occurred in transmission.

Requester - A graphic box which appears on the screen when extra information is needed by the program. The user will type in the information or click to make selections. Then the user will have to select a gadget to close the requester before proceeding.

RTS (Request to Send) - An RS-232 modem interface (sent from the DTE to the modem on pin 4) which indicates that the DTE has data to transmit.

Response Time - The elapsed time between the generation of the last character of a message at a terminal and the receipt of the first character of the reply. It includes terminal delay and network delay.

ROM (Read-Only Memory) - Nonvolatile semiconductor memory manufactured with predefined data content, permanently stored.

RS-232 - Interface between data terminal equipment (computer) and data communication equipment (modem) employing serial binary data interchange.

RTS (Request to Send) - Physical modem interface control signal from DTE, requesting clearance to transmit.

Scroll Bar - A gadget which you move to display previously hidden areas of a project.

Scroll - To move through the contents of a project, displaying previously hidden areas.

Sizing Gadget - By dragging this gadget, the user is able to change the size of a window.

SOH (Start of Header) - A transmission control character used as the first character of a heading of an information message.

Space - Absence of signal. In telegraph communications, a space represents the open condition or no current flowing. A space impulse is equivalent to a binary 0.

Start Bit - In asynchronous transmission, the first bit or element in each character, normally a space, which prepares the receiving equipment for the reception and registration of the character.

Stop Bit - In asynchronous transmission, the last bit, used to indicate the end of a character; normally a mark condition, which serves to return the line to its idle or rest state.

Tool - A program running on the Workbench. A tool lets you create and change information.

Window - A box, or area on the screen. Tools and independently running programs use windows to accept and present information.

Workbench - A facility which allows you to perform various operations such as copying disks and opening, closing and moving projects, tools and drawers.

X-ON/X-OFF (Transmitter On/Transmitter Off) - Control characters used for flow control, instructing a terminal to start transmission (X-ON) and end transmission (X-OFF).

Zoom - This allows you to narrow in on a portion of the screen for a more detailed view.

INDEX

Numeric

7 bits, 5-2
8 bits, 5-2

A

ANSI, 9-1, 9-4
ARC Files, 8-1, 8-7
ARexx language, 6-2
ASCII Send, 7-3
ASCIISEND, 11-2
ASK, 11-2
AT, 11-3
Auto Chop, 8-1
Auto Wrap, 9-2, 9-4 ... 9-5
Auto-Chop, 8-7

B

Backup Copy, 2-1
BAUD, 11-3
Baud Rate, 5-1
BBS, 2-2
Bell, 9-5, 11-4
Bits

Number of, 5-2

Board, 2-2
BREAK, 11-4
Bulletin Board, 2-2

C

Capture, 7-4
Capture Buffer, 7-4
Clear, 7-4
Close, 7-4
Copy, 7-4 ... 7-5
Cut, 7-4 ... 7-5
Open, 7-4
Paste, 7-4 ... 7-5
View, 7-4
Capture Options, 7-1
Carriage Return, 7-2
CB, 9-4
CD, 11-4
CHAIN, 11-5
Char Delay, 7-2
Character Delay, 7-2
chat, 9-4
Chat Mode, 9-4
Checksum, 8-5
Clear, 7-4, 11-5

Clipboard, 7-5
Clock, 4-5
Close, 7-4, 11-6
Color Requester, 4-2
Color Selection
HSL, 4-3
RGB, 4-3

Colors, 9-6
Columns, 9-6, 11-6
Copy, 7-4 ... 7-5
CR, 7-2
CRC, 8-3, 8-5 ... 8-6, 11-6
CRC-16, 8-6
CRC-32, 8-3, 8-6
Crosshair, 10-2
Cut, 7-4 ... 7-5
Cyclic Redundancy Check, 8-3, 8-6

D

Data Level, 10-2
DELAY, 11-6
DEPTH, 11-7
Dial, 6-2, 11-7
Disconnect, 6-1
Duplex, 5-3
Full, 5-3
half, 5-3

E

ECHO, 11-8
Ed, 6-7, 7-3
EDT, 9-2
Electronic Bulletin Board System, 2-2
emacs, 9-2, 9-4
Emulator, 9-1
ANSI, 9-1, 9-4
H19, 9-1, 9-4
TALK Terminal, 9-1, 9-4
TEK4014, 9-1
Tektronix, 10-1 ... 10-6
Tektronix 4014, 9-1
TTY, 9-1, 9-4
VT100, 9-1
VT52, 9-1, 9-4
End Send, 7-3

End-of-line, 7-1
 Even Parity, 5-2
 Exit, 4-6, 11-8
F File Requester, 3-1
 FINISH, 11-9
 Font, 9-5, 11-9
 Fonts, 10-2
 Tektronix, 10-2
 Full Duplex, 5-3
 Full Screen, 4-1
 Full VT100 Emulation
 in a Small Workbench
 Window, 9-2
 Fun Keys, 4-3
 Function Keys, 4-3
G Get, 8-5
 Get Dir, 8-6
 GETLINE, 11-9
 GOTO, 11-9
H H19, 9-1, 9-4
 Half Duplex, 5-3
 Handshake, 5-3
 RTS/CTS, 5-3
 X-on X-off, 7-1
 X-on/X-off, 5-3
 Hangup, 6-3, 11-10
 HELP, 11-10
 HSL, 4-3
 HST, A-2, 6-1
I Icons, 4-4, 11-10
 IF, 11-10
 IF TIMEOUT, 11-11
 INPUT, 11-12
K Kermit, 8-1, 8-4
 Kermit Server Mode, 8-5
 Keyboard
 Tektronix, 10-2
 KGET, 11-12
L LF, 7-1
 Line Delay, 7-2
 Line Feed, 7-1
 Line Type, 2-3
 Pulse, 2-3, 6-1
 Touch, 2-3, 6-1
 List Disks, 4-6
 as Device Names, 4-6
 as Volume Names, 4-6

Load Crosshair, 10-2
 Local echo, 5-3
 Login Script
 OAG, 11-24
 Login Scripts, 6-2, 6-6, 7-5
 CompuServe, 11-23
 DIALOG, 11-23
 Dow Jones, 11-23
 MCIMail, 11-23
 NewsNet, 11-24
 PeopleLink, 11-24
 Long Distance Access Code,
 6-4
M Mail
 Sending, 7-2
 Mail, Receiving, 7-4
 MODE, 11-12
 Modem
 HST, A-2, 6-1
 type, 6-1
 MoreRows, 9-6
 Multiple Printers, B-3
 Multiple Serial Ports, B-3
N New CLI, 4-2
 No. Bits, 5-2
 Non -Interlaced VT100 Win-
 dow, 9-3
 Null Modem Cable, 2-1
O Odd Parity, 5-2
 ON, 11-12
 Open, 7-4, 11-14
 Options
 Tektronix, 10-2
P Padding Characters, 8-2
 PAL, 9-6
 Parity
 even, 5-2
 mark, 5-2
 none, 5-2
 odd, 5-2
 space, 5-2
 Paste, 7-4 ... 7-5
 Phone Book, 6-3
 Pitch, 3-4
 Printer, 4-4, 11-14
 Protocol, 8-1, 11-14

- Q** Queue, 6-4
Quick Menu, 2-2, 4-3
Quit, 4-6
- R** Rate, 3-4
RECEIVE, 11-14
Receiving Mail, 7-4
Redial, 6-2, 11-15
REPLY, 11-15
Retrographics VT640, 10-2
REWIND, 11-15
RGB, 4-3
Rows, 9-5, 11-16
RUN, 11-16
RX, 11-16
- S** Sample Frequency, 3-3
Sample Scripts, 11-23
Save, 4-6
SAY, 11-17
Screen, 4-1, 11-17
 Full, 4-1
 Workbench, 4-1
Script, 6-2, 6-6, 7-5,
 11-1 ... 11-34
 Sample Scripts, 11-23
Script Conversion Table
 A-Talk and A-Talk
 Plus to A-Talk III, 11-21
Script Tutor, 7-5
SEND, 11-17
Send break, 6-3
Send Options, 7-1
Sending Mail, 7-2
Serial Cable, 2-1
SET, 11-17
Split Screen, 9-5
START, 11-18
Startup Scripts, B-3
STOP, 11-18
Stop Bits, 5-2
STRING, 11-18
SUSPEND, 11-19
- T** TALK Terminal, 9-1, 9-4
Talk Toy, 3-3
Talk2Three, 11-21
TEK4014, 9-1
Tektronix, 10-1 ... 10-6
 Display, 10-3
 Fonts, 10-2
 Keyboard, 10-2
 Options, 10-2
 Print, 10-5
 Replay, 10-5
 Save Draw, 10-4
 Save IFF, 10-4
 Save Plot, 10-4
 Zoom In, 10-5
Tektronix 4014, 9-1
Tektronix Menu, 10-1 ... 10-6
Terminal Menu, 9-1 ... 9-6
Text Editor, 6-7
TIMEOUT, 11-19
Timeouts, 8-7
 Relaxed, 8-7
 Standard, 8-7
Timer, 4-5
Transfer Menu, 8-1 ... 8-18
True YMODEM, 8-3
TTY, 9-1, 9-4
Type, 8-7
- U** U. S. Robotics HST, 6-1, A-2
UNIX, 9-2
- V** vi, 9-2, 9-4
View, 7-4
VMS, 9-2
Voice, 4-5, 11-19
 Filtered, 4-5
 Normal, 4-5
 Requests, 4-5
Voice Buffer, 4-5
Volume, 3-4
VT100, 9-1, 9-3
VT52, 9-1, 9-4
- W** WAIT, 11-20
Wildcard Characters, 8-3 ...
8-4
Workbench Interlace, 9-3
Workbench Screen, 4-1
Workbench VT100, 9-3
WRAP, 11-21
WXMODEM, 8-5
- X** XMODEM, 8-1, 8-7
XON, 11-21
- Y** YMODEM, 8-1 ... 8-2, 8-10
YMODEM Batch, 8-1, 8-3

YMODEM, True, 8-3
YMODEM-1K, 8-2
YMODEM-g, 8-5
Z ZMODEM, 8-1, 8-3, 8-10
ZModem Remote, 8-14
ZMODEM Resume, 8-14
ZOO Files, 8-1, 8-7

MORE SOFTWARE SOLUTIONS FROM OXXI

Do Business, Not Accounting, with

NIMBUSTM

Record Keeping for Small Business

Easy To Use, Yet Powerful

NIMBUS from Oxxi is an easy-to-use yet powerful cash management accounting system for small businesses. Nimbus removes the drudgery of designing reports, conceiving a Chart of Accounts, and keeping track of debits and credits. The pre-defined reports, editable Chart of Accounts, and single entry method make Nimbus the first up-and-running-in-ten-minutes accounting system. Nimbus has been designed for the small businessman who is too busy to do accounting, but who wants to instantly know the payment status of a customer, outstanding balances to vendors, and his P/L statement.

Completely Multitasking

Nimbus's multitasking capabilities give it constant access to the General Ledger, Accounts Payable, and Accounts Receivable. When data is entered into either Accounts Payable or Accounts Receivable, Nimbus automatically updates the General Ledger.

Customers and Vendors Kept Track of by Name

Customers and vendors are tracked by name, rather than a customer number or vendor number. Thus the small businessman can find information in his language, rather than the computer's. Pre-defined invoices optionally include a summary of customer account aging. Thus, every invoice lets the small businessman know his customer's account status, and can act as a customer statement of prior invoices. Yet Nimbus is flexible. If separate statements are desired, they can be called up at will, and both invoices and statements can be printed either on plain paper, or on readily available stock forms.

No More Debits and Credits

Pop-up mini menus guide the user through each account activity. Nimbus's menus are written in commonly used business terms, rather than accounting jargon. Debits and credits are a thing of the past with Nimbus.

Do Business, Not Accounting

Nimbus is the first accounting software system that makes the computer do what it does best, keeping track of categories and calculations, and lets the users do what they do best, conduct their business. Practically all the Nimbus user has to do is make deposits to his checking account, write checks, and send out invoices. The same activities he is used to performing on a daily basis. The difference is that Nimbus automatically captures that information and converts it into easy-to-understand reports such as a Balance Sheet or Income Statement.

New for 1989! Version 1.3. List Price: \$159.00

NIMBUS is a trademark of Nimbus, Inc.

A SOFTWARE STORAGE SOLUTION from OXXI

Walnut Disk Box

Solid American Walnut Disk Box, with hand rubbed lacquer finish, mitered corners, recessed hinge, partitions, and handle. Holds 35 disks comfortably. Small footprint – 4 5/8 inches by 8 inches by 4 inches high.

Great gift, or a treat for yourself and your disks!

Only \$19.95! (a \$34.95 value)

Order direct from:

**Oxxi, Inc.
P. O. Box 90309
Long Beach, CA 90809-0309**

U. S. Orders add \$4.00 for shipping and handling.

California residents add \$1.30 sales tax.

International orders add air postage.

MORE SOFTWARE SOLUTIONS FROM OXXI

TAX BREAK™

Tax Preparation and Planning Tool

Tax Break™ is easy to use with on screen representation of forms and schedules, fast recalculation, and constant status of taxes due or refunds. Full integration of all forms and schedules allows constant updating of tax status for evaluation of different tax scenarios.

- Scrollable, on-screen lookalike forms and schedules
- Schedules can be opened from a menu or by double clicking on the schedule result line in form 1040
- Ongoing status of taxes due and refunds at the bottom of each form, or expanded status screen available with mouse click or F2 key.
- Instant updating of forms and schedules with each recalculation
- Line by line data entry prompts
- On-screen IRS booklet
- Prints IRS-approved facsimile forms
- Pop-up calculator
- Zoom in and out of interlace at will

US version 1.0 includes Form 1040, Schedules A-E, SE, 1906, 2441, IRA, and more!

List Price \$79.95

Amiga is a trademark of Commodore-Amiga, Inc.
TAX BREAK is a trademark of Oxxi, Inc.

Oxxi inc.

P.O. Box 90309
Long Beach, CA 90809-0309



**This was brought to you
from the archives of**

<http://retro-commodore.eu>