KAYPRO®

MITE™

MITE

Version 2.7

A Data Communications System for Microcomputers

Mycroft Labs, Incorporated

COPYRIGHT

DISCLAIMER

Kaypro Corporation and Mycroft Labs make no representations or warranties with respect to the contents hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose.

TRADEMARKS

MITE is a registered trademark of Mycroft Labs.
KAYPRO is a registered trademark of Kaypro Corporation.

MYCROFT LABS
Box 6045, Tallahassee, Florida, 32314

SOFTWARE LICENSE AGREEMENT

IMPORTANT: All Mycroft Labs programs are sold only on the condition that the purchaser agrees to the following license. *Read this license carefully.* If you do not agree to the terms contained in this license, return the package to your dealer and your purchase price will be refunded. If you agree to the terms contained in this license, complete the SOFTWARE REGISTRATION and return it by mail.

==========

MYCROFT LABS agrees to grant and the Customer agrees to accept on the following terms and conditions nontransferable and nonexclusive licenses to use the software program(s) (Licensed Programs) herein delivered with this agreement.

TERM:

This agreement is effective from the date of receipt of the above-referenced program(s) and shall remain in force until terminated by the Customer upon one month's prior written notice, or by Mycroft Labs as provided below.

Any license under this Agreement may be discontinued by the Customer at any time upon one month's prior written notice. Mycroft Labs may discontinue any license or terminate this Agreement if the Customer fails to comply with any of the terms and conditions of this Agreement.

LICENSE:

Each program license granted under this Agreement authorizes the Customer to use the Licensed Program in any machine-readable form on any single computer system (referred to as System). A separate license is required for each System on which the Licensed program will be used.

This Agreement and any of the licenses, programs or materials to which it applies may not be assigned, sublicensed or otherwise transferred by the Customer without prior written consent from Mycroft Labs. No right to print or copy, in whole or in part, the Licensed Programs is granted except as hereinafter expressly provided.

PERMISSION TO COPY OR MODIFY LICENSED PROGRAMS:

The Customer shall not copy, in whole or in part, any Licensed Programs which are provided by Mycroft Labs in printed form under this Agreement. Additional copies of printed materials may be acquired from Mycroft Labs.

Any Licensed Programs which are provided by Mycroft Labs in machine-readable form may be copied, in whole or in part, in printed or machine-readable form in sufficient number for use by the Customer with the designated System, to understand the contents of such machine-readable material, to modify the Licensed Program as provided below, for backup purposes, or for archive purposes, provided, however, that no more than five (5) copies will be in existence under any license at any one time without prior written consent from Mycroft Labs. The Customer agrees to maintain appropriate records of the number and location of all such copies of Licensed Programs. The original, and any copies of the Licensed Programs, in whole or in part, which are made by the Customer shall be the property of Mycroft Labs. This does not imply, of course, that Mycroft Labs owns the media on which the Licensed Programs are recorded. The Customer may modify any machine-readable form of the Licensed Programs for his own use and merge it into other program material to form an updated work, provided that, upon discontinuance of the license for such Licensed Program, the Licensed Program supplied by Mycroft Labs will be completely removed from the updated work. Any portion of the Licensed Program included in an updated work shall remain subject to all other terms of this agreement, and Mycroft Labs retains the right to ownership of any such included portions of the Licensed Program, although no charge is made for such inclusion.

The Customer agrees to reproduce and include the copyright notice of Mycroft Labs on all copies, in whole or in part, in any form, including partial copies or modifications, of Licensed Programs made hereunder. The Customer further agrees to never remove any copyright notices from any Mycroft Labs software or documentation in either printed or machine-readable form.

PROTECTION AND SECURITY:

The Customer agrees not to provide or otherwise make available any Licensed Program, including, but not limited to, program listings, object code and source code, in any form, to any person other than Customer or Mycroft Labs employees, without prior written consent from Mycroft Labs.

DISCONTINUANCE:

Within one month after the date of discontinuance of any license under this Agreement, the Customer will furnish Mycroft Labs a certificate certifying that through his best effort, and to the best of his knowledge, the original and all copies, in whole or in part, in any form, including partial copies or modifications, of the Licensed Programs received from Mycroft Labs have been destroyed, except that, upon prior written authorization from Mycroft Labs, the Customer may retain a copy for archive purposes.

DISCLAIMER OF WARRANTY:

Mycroft Labs makes no warranties with respect to the Licensed Programs. The sole obligation of Mycroft Labs shall be to make available all published modifications or updates made by Mycroft Labs to Licensed Programs which are published within one (1) year from date of purchase, provided Customer has returned the Software Registration delivered with the Licensed Program. A charge for the modifications or updates may or may not be made at the sole discretion of Mycroft Labs.

LIMITATION OF LIABILITY:

The foregoing warranty is in lieu of all other warranties, expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. In no event will Mycroft Labs be liable for consequential damages even if Mycroft Labs has been advised of the possibility of such damages.

GENERAL:

If any of the provisions, or portions thereof, of this Agreement are invalid under any applicable statute or rule of law, they are to that extent to be deemed omitted.

This Agreement is adapted from an original Agreement developed by the Legal Department of Digital Research, Inc., and is used by permission.

CONTENTS

Contents
(Continued)

Contents
(Continued)

Contents
(Continued)

PREFACE

This manual has been designed for use with all Kaypro computers, so depending on which computer you have, you may have to paraphrase some of the instructions.

If you own a KAYPRO 10:
Any time this manual mentions the A or B drive, (e.g. *Place your MITE disk in drive A* or *Place a blank, formatted disk in drive B*) you may ignore it. You already have your MITE programs on the hard disk drive, and you may save files on either the A or B side of your hard disk drive.

You may want to read Appendix A titled *AUTOMITE* for information on running MITE from the Master Menu.

You should also read Appendix E titled *Installation Notes.*

If you own a floppy disk drive based CP/M computer:
This manual was originally written with you in mind, so you may use the instructions as they are.

You may want to read Appendix A titled *AUTOMITE* for information on running MITE.

You should also read Appendix E titled *Installation Notes.*

If you own a hard disk drive based MS-DOS computer:
You should ignore the section on installing MITE and any other references to installing MITE or to the program MINSTALL. The MS-DOS version does not need to be installed, but carefully read the manual for your particular type of modem to see if it needs a *dial prefix* and enter that in your parameter file (see the chapter titled *The Parameter Menu*).

Any time this manual mentions the A or B drive, (e.g. *Place your MITE disk in drive A* or *Place a blank, formatted disk in drive B*) you may ignore it. You already have your MITE programs on the hard disk drive, and you may save files on the hard disk drive as well. Remember that the drive designation should be C: any time you name a file.

If you own a floppy disk drive based MS-DOS computer:
You should ignore the section on installing MITE and any other references to installing MITE or to the program MINSTALL. The MS-DOS version does not need to be installed, but carefully read the manual for your particular type of modem to see if it needs a *dial prefix* and enter that in your parameter file (see the chapter titled *The Parameter Menu*).

FOREWORD

MITE is the flexible, menu-driven data communications and file transfer package for microcomputers.

MITE can give the user access to virtually any asynchronous ASCII system such as The Source, CompuServe, Dow Jones, corporate or university data center. It can exchange files with a large number of mainframe computers or other microcomputers with error checking and recovery. MITE can connect with another microcomputer via a telephone line using modems or via a cable connecting the two microcomputers.

MITE is available for over a hundred popular microcomputer systems and supports full dynamic modem control for most popular auto-dial/auto-answer modems. MITE provides an easy-to-use menu system. Multiple protocols such as MITE, XMODEM, Hayes, and CROSSTALK are supported.

The tutorial at the end of this manual covers the theory and practice of telecommunications and will answer most questions you may have about interfacing your KAYPRO with devices other than modems, etc.

General Information

This manual has four basic components:

1. An introduction including a section for the first-time data communications user and directions for getting started,

2. sections on each of the eight principal menus,

3. descriptions of advanced usage and technical information on non-standard installations,

4. informational appendices.

The beginning user should read this chapter and the chapter beginning on page 9 titled *Introduction for Beginners* before proceeding to the menu descriptions. The experienced user should read this chapter and skip to page 19, *Getting Started*. The experienced user may also wish to review the chapter titled *Examples of Usage* beginning on page 101.

Page 24 begins a description of the installation procedure. As part of the installation procedure, it is very important that you read the chapter titled Modem and Cable Installation *beginning on page 113. This chapter will show the correct switch settings for your modem and, in many cases, the proper cable configuration. Note: In most cases, the switches that are set at the factory are not* correct for use with MITE.

INTRODUCTION TO MITE

MITE

MITE is the Mycroft Labs intelligent terminal program. A terminal program is a program that will send over a modem what you type at your keyboard and will display on the screen characters it receives over the modem. An intelligent terminal program will also dial the phone for you, change to different data protocols, send and receive files, and allow you to do operating system-type functions, such as disk directories.

MITE serves two basic functions:

- **To allow you to use your KAYPRO as a terminal.**

  To give you the maximum flexibility and compatibility, MITE allows you to change almost every operating characteristic a terminal might have. There are parameters and options that you may not understand and that may be confusing at first, but don't worry. MITE has all the parameters and options already set up so they will work with almost all computers. Most of them you will never have to change. However, should the need ever arise, MITE can make your KAYPRO compatible with almost any dial-up computer you would ever want to call. The rule of thumb is, if you could access the computer with a Lear-Siegler ADM-3A terminal, then you can use MITE with that computer.

- **To transfer files.**

  With MITE, you can store information you get from other computers on diskette. You can also type in a file, save it on diskette, and then send it all at once over the modem to another computer. You can exchange any type of file (.COM files, data files, etc.) by using a binary file transfer. Using MITE, you can transfer files to and from almost any computer.

It is especially simple to learn and to use, even by non-technical personnel, and has a number of *user protection* features. It has four principal applications:

1. Accessing on-line timesharing systems, such as might be found at many corporate or university computing centers, or on-line data base utilities. Virtually any system that supports ASCII terminals can be accessed with full text file transfer capability. Some of the more popular compatible systems are:

DEC	All models, e.g., PDP-11, Vax-11
CDC	6000 series, Cyber 170 series, etc.
DG	Nova, Eclipse, etc.
PRIME	All models
HONEYWELL	Level 62, Level 6, etc.
HARRIS	All models
IBM	With TTY communications

   Most on-line data base utilities are compatible with MITE, including virtually every service available through TELENET and/or TYMNET. Examples include:

   The Source
   CompuServe
   Dow Jones

2. Accessing on-line remote CP/M systems (RCPM), Computerized Bulletin Board Systems (CBBS - (c) Ward Christensen), and many other such message and/or public domain software systems. MITE supports the standard *XMODEM* protocol, as used on most RCPM systems, for error-free transfer of any file, including raw object code (.COM files).

3. File exchange with other microcomputer systems running MITE, CLINK, Smartcom, CROSSTALK or various other *smart terminal programs*. MITE supports several protocols for compatibility with the greatest number of such programs.

4. Accessing the Western Union TWX (or indirectly, the TELEX) network or other *store and forward* systems (e.g., ITT).

WHAT YOU CAN DO WITH MITE

Using MITE, you can call computerized bulletin boards to leave messages for other computer users. You can subscribe to pay services, such as CompuServe, and The Source. These services provide on-line encyclopedias, libraries, shop-by-computer services, computer software, stock quotes, and much more. You can call a friend's computer, if it has a modem, and chat with your friend in terminal mode. You can exchange any legitimate CP/M file(s) with a friend over the phone. You can call up a CP/M-oriented bulletin board and get some of the thousands of public domain CP/M programs available. Many electronic mail services (such as MCI Mail and EasyLink) are available via MITE. The uses for MITE are limited only by your imagination.

MODEM COMPATIBILITY

MITE works with a great number of modems. For instance, some of the auto-dial modems it supports are:

Anchor Signalman Mark VII
Bizcomp 1012
CTS 212AH (Companion)
Datec 212
Hayes Smartmodems 300 and 300/1200
Incomm 212A
Kaypro Internal Modem
Novation Smart-Cat 300 and 300/1200
Novation Smart-Cat Plus
Novation Auto-Cat 1200
Novation J-Cat
Racal-Vadic VA212, VS212
Rixon 212A
Toyocom(BYTCOMM) 212AD
UDS 212A/D
U.S. Robotics 212.

It also supports a large number of manual dial modems. The specification of the modem to be used with MITE is menu-selected.

All communications parameters (including the phone number and any macro strings) specific to a given site can be easily selected via the menu options, then saved on a *parameter file* for future use. Once this is done, all parameters can be set in a single operation by specifying this file as an

argument on the command line (or via an option on the Main Menu). All options can be set with a minimum of effort, many with a single keystroke.

MITE has numerous *user protection* features to help prevent accidental loss of captured data or disk files. Any time a file is created (e.g., when capturing a file or saving parameters), the directory is checked to see if there is already a file with that name. If so, you are given the choice of overwriting it, or aborting the current operation so that a different file name can be specified. When exiting to the operating system, if a capture operation is in progress, the data will be automatically flushed to disk. Also, if the carrier is still present at that time, you are informed of this fact and given the option of hanging up.

System Requirements

MITE runs on Digital Research's CP/M versions 1.4, 2.2 or 3.0, and any version of MS-DOS or PC-DOS. The system must have a minimum of 56K of memory. The system must have at least one floppy disk drive.

HOW MITE IS ORGANIZED

MITE is organized into two main sections: terminal mode and the menus.

In terminal mode, MITE pretends that it is a terminal. A terminal is a device that does nothing but transmit and receive characters. A terminal receives characters from its input port (a modem) and displays them on the screen. It also receives characters from the keyboard and transmits them through its output port (a modem). When MITE is in terminal mode, it will display all the characters it receives from the modem and send all the characters you type to the modem.

In the menus, MITE shows you the current settings for its parameters and options and lets you change them if necessary. MITE has a main menu and eight sub-menus, each of which deals with a particular aspect of MITE's operation.

The menus are as follows:

Main Menu This is the menu from which you select the other menus. In addition to the eight sub-menus, there are five other options available:

G	Go into terminal mode.
H	Hang up the phone.
I	Enter a message for the Site ID.
L & S	Load and save a parameter file. A parameter file holds all the information about how MITE is set to operate. A default parameter set is built into MITE, so this feature is optional.

Parameter The Parameter menu contains the terminal-related information MITE uses. It is also where you tell MITE what telephone number you want it to dial. You need to set the terminal-related parameters to match those of the computer you are calling.

Option The Option menu is where you re-define the control keys for MITE, if necessary. There are also a few screen display options in this menu.

Text File Upload	When you send a file to another computer over a modem, it is called an upload. This menu deals with uploading a text file and the options involved in a text upload.
Text File Download	When you receive a file over a modem from another computer, it is called a download. This menu deals with downloading a text file and the options involved in a text download.
Binary File Xfer	This menu involves uploading or downloading a file using a special method for transferring files. You can transfer any kind of file this way--even program files.
Macro Definition	This menu allows you to assign a string of up to 62 characters to one key on the keyboard. This feature is provided for more advanced users.
Command Processor	The Command Processor lets you use many CP/M-type com mands without having to leave MITE.
Character Filter	The Character Filter allows you to selectively filter out any unwanted control characters another computer may send.

LOCAL/REMOTE COMMANDS

The Local/Remote commands are a mix between the terminal mode and the menus. Local/Remote command mode allows you to use menu-type commands from terminal mode. This mode is accessed by typing a control character from terminal mode. You can use all the Local/Remote commands without ever leaving terminal mode. This saves time and effort and makes complicated transactions much easier for the advanced user.

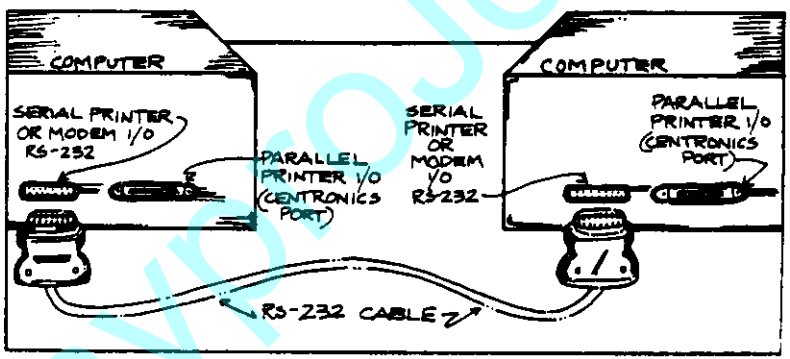
Introduction for Beginners

The purpose of this chapter is to provide a non-technical introduction to concepts and terms that are used in explaining the operation of the MITE program and to provide the information needed to run MITE on your computer. There is no one standard microcomputer configuration; however, since all microcomputers work more or less the same, a few general concepts are presented and then the details relating to specific computers.

Introduction to Data Communications

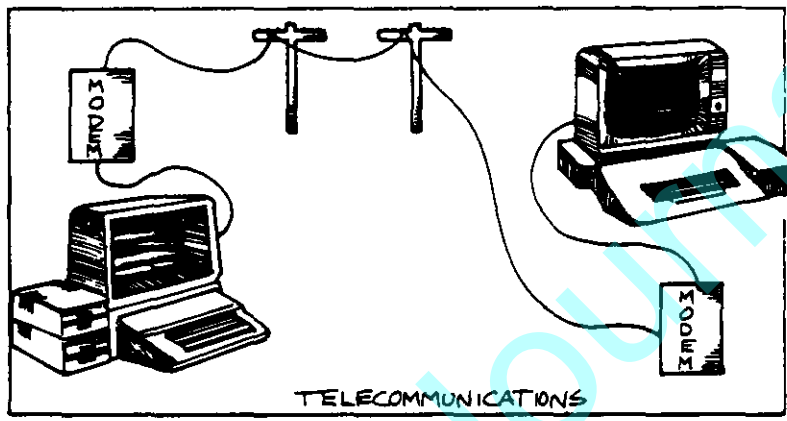
There are two ways to connect computers for communication:

1. **Hard wire** (also called **Direct Connect**) -- where the computers are in relatively close proximity (within 50 feet). With a hard wire connection the computers are connected via a cable to the computers' RS-232 serial ports (more about this later).



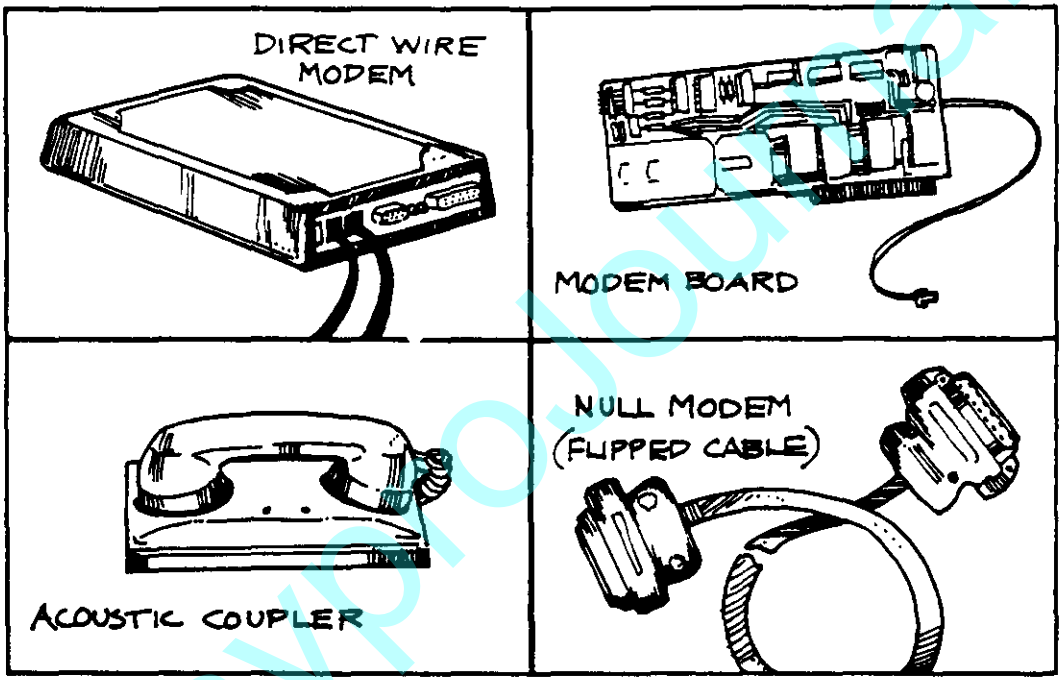
HARD WIRE OR DIRECT CONNECTION BETWEEN TWO COMPUTERS

2. **Telephone or modem connection** -- where the computers are connected via a telephone line. Additional equipment, usually a modem, is needed for this type of connection so that the data can be transmitted over the telephone lines.



TELEPHONE OR MODEM CONNECTION BETWEEN TWO COMPUTERS

The **modem** is a piece of equipment that translates the signal coming from the computer into an audio signal that can be sent over a phone line and vice versa. A modem is needed on both computers connected via the phone line. Pictured below are several types of modems which can be used with microcomputers.

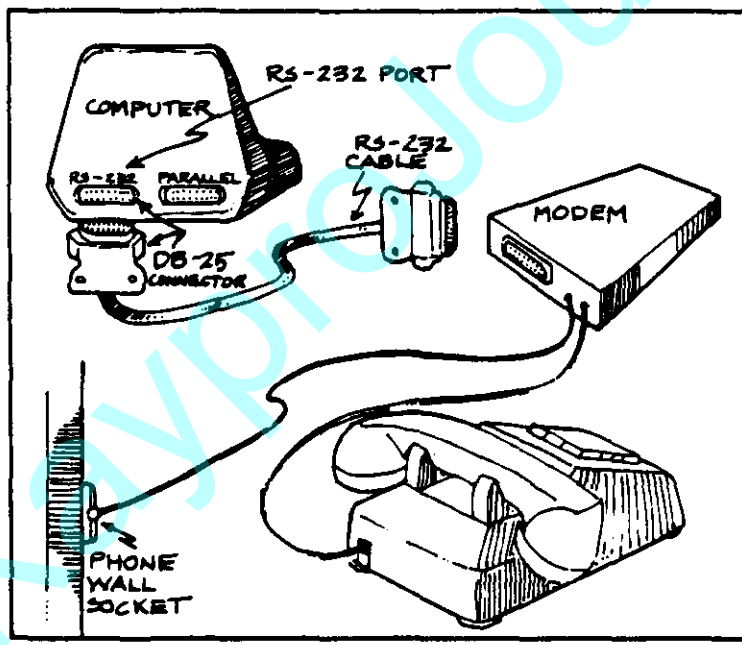


TYPES OF MODEMS

There are advantages and disadvantages to both types of connections. The advantage of the hard wire connection is that data can be transferred at a much faster rate than with a telephone connection, and the equipment cost is significantly less. The advantage of the telephone connection is that the computers can communicate over much greater distances; however, transmission speeds are slower and the additional equipment (modems) can cost between $100 and $500 each.

Data Communication Fundamentals

MITE is a serial communications program. That is, MITE uses a serial communications port of the computer to send and receive data. The serial communications ports on the computer are sometimes referred to as RS-232 ports. RS-232 refers to a standard used in most serial communications. Most RS-232 applications use a DB-25 connector. The **DB-25** connector has twenty-five pins or circuits - about eight of which are used in actually connecting computers to computers or computers to modems. Some computers come with serial communications ports and some do not. One of the first things you must determine is whether your computer has a serial communications port. To determine whether your computer has a serial communications port, look for a DB-25 connector on the back of your computer, or check the KAYPRO USER'S GUIDE.



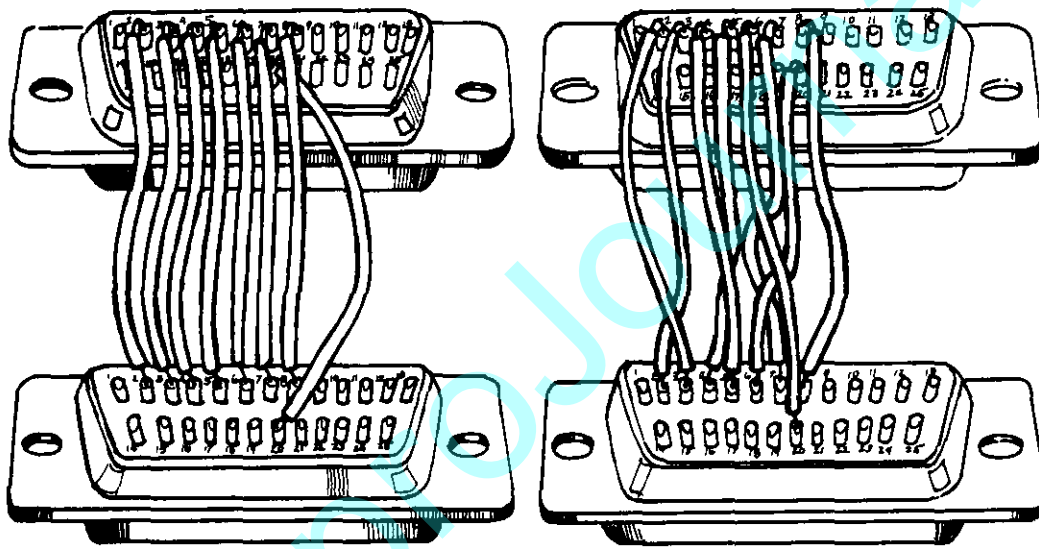
There are two types of RS-232 ports. One type, called the Data Communications Equipment (DCE) port, is configured to work with data terminal equipment (DTE) such as a terminal or a printer. The other type, called the Data Terminal Equipment (DTE) port, is configured to work with data

communications equipment (DCE) such as a modem. It really doesn't matter if your serial communications port is DTE (Data Terminal Equipment) or DCE (Data Communications Equipment), but it will be necessary to know which it is so that you can purchase or construct the proper type of cable. The cable will be used either to hard wire to another computer or to connect your computer to your modem.

Microcomputers that have RS-232 connectors configure them as DCE or DTE devices. The problem is that the cable used for connecting a DTE device to a DCE device is different from the cable used to connect a DTE to DTE or DCE to DCE. First, look at the simplest case of connecting two computers together via a hard wire connection.

Hard Wire Connection Between Two Computers

Most serial communications applications use eight or fewer of the twenty-five available wires connected to pins on the DB-25 connector. There are two basic cable configurations, the standard cable and the flipped cable. The **standard cable** usually consists of two DB-25 connectors with the pins connected as shown in the diagram on the left below.



STANDARD CABLE **FLIPPED CABLE**

The **flipped or reversed** cable (also called a **null modem**) consists of two DB-25 connectors with the pins connected as shown in the diagram on the right above.

Do you see the difference between the two cables? The wiring is reversed or flipped on some pins in the second connector. The DB-25 connector is simply a connector or plug for connecting a wire into a circuit. What you are doing when you connect two computers with a cable is completing a number of circuits between their serial ports. That is, there is a convention (RS-232) for how the serial ports are wired to the DB-25 connector. What

is confusing is that there are two types of RS-232 interfaces. The Data Terminal Equipment (DTE) interface and the Data Communications Equipment (DCE) interface. These interfaces are explained in much more detail in *Appendix B - A Practical Guide to RS-232 Interfacing*, but for now, a simpler explanation would be to follow a data signal from one microcomputer to a second microcomputer.

The first microcomputer recognizes (through a program) that there is some information (hereon called a signal) to be sent to the second computer. The microprocessor sends the signal to the serial port. If the serial port is wired as a DTE interface, the signal is sent to pin two of the DB-25 connector on the computer. This signal goes out the wire connected to pin two on the computer cable and over to the DB-25 connector attached to the second computer.

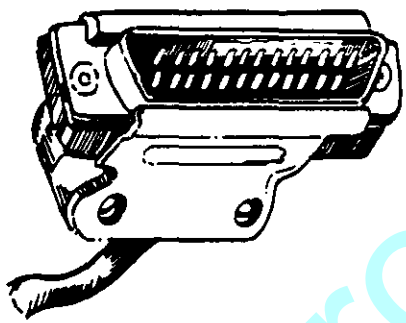
The serial port on the second computer can also be wired as a DTE or DCE interface. If it is wired as a DCE interface, it is expecting to receive data on pin two of its serial port. All we need is for the wire coming from pin two on the first connector to go to pin two on the connector on the second computer. The signal is accepted and processed by the program running on the second computer. But, what if the second computer is wired as a DTE interface? In that case, the second computer is also expecting to send data out pin two, not receive it there. It is expecting to receive data on pin three so the wire must be connected from pin two on the connector at the first computer (send data) to pin three on the connector at the second computer (receive data). This is called **flipping** or **reversing** the cable. There are other designations on pins four and five and pins six and twenty that must be reversed or flipped on a cable that connects a DTE interface to another DTE interface or a DCE interface to another DCE interface. There is a section titled *Modem and Cable Installation* in this guide, but this should explain the reason behind the *flipped cable.*

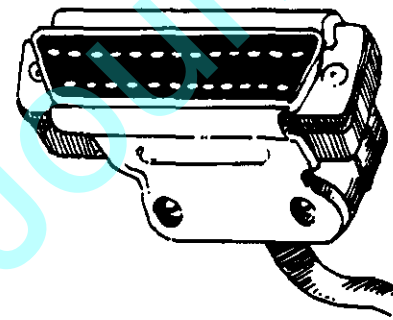
Which cable do you need? If you are connecting a DTE (Data Terminal Equipment) device to a DCE (Data Communications Equipment) device you use a standard cable. If you are connecting a DCE to DCE or a DTE to DTE, you will need a flipped cable. For a direct connection to another computer, you will most likely need a flipped cable since you are connecting two like devices together (DCE to DCE or DTE to DTE). To connect the computer to a modem, you will most likely need a standard cable since you are normally connecting a DTE (the computer) to a DCE (the modem).

In summary, a standard cable is used to connect two devices that have dissimilar types of interfaces, e.g., DTE to DCE. A flipped cable is used to connect two devices with similar interfaces, e.g., DTE to DTE or DCE to DCE.

You have also probably noticed that there are both *male* and *female* connectors. Most likely your computer and modem have female connectors, and the cables you will be purchasing will have male connectors at both ends. However, look at your equipment carefully before purchasing cables. Some computers have a male connector and will need a cable with a female connector on one end.

DB-25 MALE **DB-25 FEMALE**





Connecting the Computers Via a Phone Line (Modem Connection)

For this type of connection the computer is connected to a modem and the modem is connected to the phone line. The **modem** is a device that changes the digital electronic signals coming from the computer into audio signals that can be transmitted over the phone line. The modem at the receiving end changes the audio signals back into electronic signals that the other computer can interpret. The modem is generally a DCE device. If your computer port is a DCE, you will need a flipped cable because you will be connecting a DCE to a DCE. If your computer's port is a DTE, you will need a standard cable since you will be connecting a DTE to a DCE. However, there are many different types of modems and some require unique cable configurations. A list of cable configurations known to work

with MITE is given in the chapter titled *Modem and Cable Installation.* Before you buy or have a cable made up for your modem, you should look at this section to see if your modem is listed, and if so, what you have to do to use it.

There are different types of modems. First, there are **asynchronous serial modems** and **synchronous serial modems.** Synchronous or asynchronous refers to the timing of the data as it is sent out. There is a more elaborate discussion of this concept in *Appendix C - Introduction to Data Communications*, but suffice it to say that MITE is an *asynchronous* communications program and works with *asynchronous* modems. There are also **direct wire** modems and **acoustic coupled** modems. The direct wire modems have a modular type telephone plug that allows you to plug them directly into the telephone jack in your house. The acoustic modem has a set of rubber cups into which you place the handset of a standard desk telephone to make the connection between the modem and the phone line. Again, each type has its advantages and disadvantages. The advantage of the direct wire modem is more reliable data transmission. The disadvantage is that you may not have a modular plug jack where you wish to use the modem; this difference would require a modification to the telephone equipment. The advantage of the acoustic modem is that you can use it on any telephone that has a standard type handset. However, telephone handset designs are rapidly departing from the old standard, and you may not have a standard handset where you wish to use the computer.

Features of some direct wire modems are an **auto-dial** and/or an **auto-answer** capability. The auto-answer function allows the computer to detect an incoming phone call, answer the phone, and establish communications. When the user hangs up the phone, the auto-answer modem disconnects and hangs up the phone on the computer end. This feature is necessary if the user needs a computer that will be accessed by phone with no operator present. The auto-dial feature allows the modem to dial the number of the computer you are calling. This eliminates the need for a telephone instrument near the computer and is a convenience if the use of modem communications is frequent.

Modems also come in two basic speeds, 300 baud and 1200 baud. A 300 baud modem sends and receives characters at approximately 30 characters/sec. The 1200 baud modem sends and receives characters at approximately 120 characters/sec. The 1200 baud modem is four times faster than the 300 baud and in most cases about four times as expensive. You will have to determine if the higher speed justifies the higher cost in your situation. You also need to know the baud rate supported by the other computer(s) with which you wish to communicate. The best and most

expensive modem is one which supports 300 or 1200 baud and has auto-dial and auto-answer features. Presently, it is usually the case that most users with whom you will wish to communicate have 300 baud modems (many 1200 baud modems will also communicate at 300 baud as well).

The Role of MITE

MITE is the program that you will run to allow communications between computers. The sequence of events is generally this:

1. The originator of the communication runs MITE on his/her computer, configuring it to send data at a predetermined speed.

2. The originator calls another computer (either micro or mainframe) that has been configured to receive telephone communications.

3. The originator selects file transmission options from a menu and sends the other computer a file or requests a file from the other computer or the originator uses the computer as a terminal to another computer.

4. When all desired files have been transmitted and/or received or use as a terminal has been completed, the originator hangs up.

MITE is a very versatile program and it allows on-screen communication between the users before and after any file transmission process. There are other ways that MITE can be used in the direct connect situation and in connection via telephone to commercial information services such as CompuServe. These uses will be explained in detail later. The first step in using MITE (after your hardware is properly connected) is to install and copy the MITE master disk.

GETTING STARTED

WHAT YOU NEED

To use MITE, you will need the following items:

- a KAYPRO computer
- a modular telephone cord long enough to reach from the computer to the nearest telephone outlet
- the MITE program.
- (an external modem if your KAYPRO does not have one built in)

HOW IT WORKS

For the program to function, you must connect one end of the telephone cord into a telephone outlet and the other end into the telephone socket of your modem (on appropriately equipped KAYPROs, this is the socket marked MODEM in the back of the computer). This connects the computer to the outside world through the modem.

The modem works by sending a tone, called a carrier, over the telephone line to another computer's modem. The modem converts information from your computer into variations in the carrier. These variations are then converted by the modem at the other end into information the other computer can understand. The MITE program controls the modem and adjusts it so it sends information using the correct format.

EXTERNAL MODEMS

If you do not have a KAYPRO with a built-in modem, or if you are using a modem other than the one built into the KAYPRO, then you will have to install MITE so it will work with your modem. For instructions on how to install MITE, refer to Appendix E titled *Installation Notes.*

RUNNING MITE

IF YOUR COMPUTER DOES NOT HAVE A MASTER MENU:

1. Insert the working disk for MITE into drive A.

2. Reset the KAYPRO by pressing the red reset button on the back panel.

3. Type: MITE
   Press RETURN.

4. Optionally, a parameter file may be specified as the first argument. For example:

   A>mite source

   In this case, all parameters will be read from the file *SOURCE.PAR*. If a file type is specified on the argument, it will override the default file type of *.PAR*. In either case, at this time the Main Menu will be displayed, and any necessary changes to the parameters may be made and optionally saved for future use.

   A second parameter may also be specified, which, if present, is taken as the first option on the Main Menu. This parameter allows the user to start up directly in answer or originate mode. For example:

   A>mite cyber g

   would read all the parameters from the file cyber.par and then invoke the go command.

IF YOUR COMPUTER HAS A MASTER MENU:

1. Reset the KAYPRO by pressing the red reset button on the back panel.

2. From MASMENU, select the menu item, *Telecommunications.*

3. Press RETURN.

4. Select the menu item, *M.I.T.E.*

5. Press RETURN.

When you invoke MITE, you should see the following menu:
At the head of each menu the following three lines are displayed:

MITE *vx.yy* **- Copyright (c) 1983, Mycroft Labs, Inc.**
XXXXXXX. **Bytes Captured =** *nnnnn/nnnnn*. **Capture =** *XXX*.
Site ID = *xxx ... xxx*

MAIN MENU

G - Go Start Communications
H - Hang Up Phone
I - Enter Site ID
L - Load Parameters from Disk File
S - Save Parameters on Disk File

Sub-Menus:

P - Parameter	**O - Option**
U - Text File Upload	**D - Text File Download**
B - Binary File Xfer	**M - Macro Definition**
C - Command Processor	**F - Character Filter**

X - Exit to CP/M

Enter Option (? for help): _

- On the first line, the *vx.yy* is the current version number *x* and release level *yy* of MITE.

- The second line lists the status of three items as of the time the header was written. The first field (*XXXXXXX*) will contain DIRECT, ONLINE, or OFFLINE. This field reflects the current state of the carrier signal. When your computer is connected to another computer, this will read **ONLINE**. If you are using the direct connect option, this will read **DIRECT**. The second field tells how many bytes have been captured out of the number it is possible to capture in your particular system. For example, Bytes Captured = 1254/15725 would mean you have currently captured 1254 bytes out of a possible 15725. The third field tells whether capture mode is currently ON or OFF. This status line allows the user to get some quick feedback on these three items, and reassures you that the carrier has not been lost while you are in the menu system.

- The third line is the current Site ID.
- Further down is the main menu. This contains several options that directly affect program operation.
- Below the main menu are the sub-menus which allow you to change the operating characteristics of MITE.
- The last line is where MITE asks you for instructions. You give MITE instructions by typing the letter of the menu option you want. Note that either upper or lower case will be accepted.

Under the menu system, if the user tries to select an illegal option, the bell (audible tone) will be sounded, the illegal option request will be discarded, and control will be returned to option specification.

THE HELP FEATURE

Also on the last line, in parentheses, are the words **? for help**. MITE is very friendly and will help you with any questions you may have if you type question mark (?) at the appropriate menu. By using the help feature, you can learn a great deal about the operation of MITE. The following steps will familiarize you with the MITE help format:

1. Type: **?**
   Read the message.
2. Type ? again.
   You should see the general help page. Read this page carefully, as it contains valuable information about MITE.
3. Press any key (the space bar will do nicely).
   This returns MITE to the previous menu.
4. Type ? again.
5. Type: **P**
   This will display the help page for the Parameter Sub-menu. You don't have to read this one so carefully, as it deals with some things that haven't been covered yet.
6. Press any key.

7. Type: **P**
   This puts MITE at the Parameter menu.

8. Type ? again.

9. Type ? again.
   This will display another help page for the Parameter Sub-menu.

10. Press any key.
    This returns MITE to the previous menu.

11. Type ? again.

12. Type: **N**
    This will give you help on how to enter a new phone number to call.

13. Press any key.

14. Type: **X**
    This will return MITE to the main menu.

You may want to repeat the above steps a few times to make sure that you have a good grasp of how to operate the help feature. You can use this method to get help on most questions you might have about MITE.

Installing MITE

Before installing MITE, it would be best to decide how you plan to use it. If you do not have a modem or will only be using MITE in a hard wire situation, you should follow the next section entitled, *PROCEDURE FOR INSTALLING MITE FOR HARD WIRE (DIRECT CONNECT).*

HOWEVER, if you have a modem and plan to use MITE in both telephone connect and direct connect mode, you should skip to the modem installation procedure since the modem installation can also be run in DIRECT mode.

You could make a number of copies of MITE for different uses. For instance you could make one installation for direct connection and one installation for 300 baud modem connection. This way you could simply use the appropriate disk. The advanced user will quickly adapt to the procedure for configuring MITE for any number of applications and will not need multiple disks.

Procedure for Installing MITE for Hard Wire (Direct Connect)

1. Insert the MITE working disk you just made into drive A. (It may already be there).

2. Reboot the system.

3. When the system prompt shows, type **INSTALL<CR>**

4. Next you will be presented with a list of MODEMS. Since you are installing this copy to be a DIRECT or HARD WIRE copy, you should select option Z for *Direct Connect*

5. This will complete the first stage in installing MITE for DIRECT operation.

6. In order to complete the installation of MITE, you will have to run it. We will assume that you have the computers connected with the proper cable, and that the MITE disk you just installed is in drive A, reboot the computer.

7. At the system prompt, type **MITE<CR>**

8. You should see the following menu.

```
MITE v2.xx  -  Copyright (c) 1983, Mycroft Labs, Inc.
DIRECT.  Bytes Captured =     0/#####.  Capture = OFF.
Site ID =

MAIN MENU

     G - Go Start Communications
     H - Hang Up Phone
     I  - Enter Site ID
     L  - Load Parameters from Disk File
     S  - Save Parameters on Disk File

               Sub-Menus:

     P - Parameter                O - Option
     U - Text File Upload         D - Text File Download
     B - Binary File Xfer         M - Macro Definition
     C - Command Processor        F  - Character Filter

     X - Exit to CP/M

Enter Option  (? for help): _
```

Notice that in the upper left-hand corner is the word DIRECT. This means that this is a direct connect or hard wire installation. Beneath that you see the Main Menu and a list of the sub-menus. To complete the installation, we will have to set some parameters, so type a P to invoke the Parameter Menu.

9) You should now see the Parameter Menu.

```
MITE v2.xx  -  Copyright (c) 1983, Mycroft Labs, Inc.
DIRECT.  Bytes Captured =     0/#####.  Capture = OFF.
Site ID =

PARAMETER MENU

     B - Baud Rate              =  300
     D - Data Bits              = 7
     P - Parity                 = EVEN
     S - Stop Bits              = 1

     R - Role (ANS/ORG)         = ORG
     M - Mode (Duplex)          = FULL

     A - Auto Redial Count      = 0
     N - Phone Number           =
     I  - Modem Init String     =
     H - Dial Prefix            = AT DT

     X - Exit to Main Menu

Enter Option (? for help): _
```

This menu contains the information the program needs to effectively transfer data. The first parameter you will set is BAUD RATE. Notice that the baud rate is already set at 300. This is a normal value for modem connections, but for direct connections you can transmit data much faster. (Many computers will allow transfer at 9600 baud, but some will not.) For now set the BAUD RATE at 2400:

type: **B**
type: **2400<CR>**

You should see 2400 next to Baud Rate on the Parameter Menu.

10. For now ignore the other parameter settings, but note: the role of the computer is set in **ORG** (originate) mode. This parameter can be changed to ANS (answer) mode by pressing the **R** key. Notice this is a toggle operation. If you press **R** again, it will flip back to ORG. When direct connecting between two computers, one computer will have to be in ORG mode and the other will have to be in ANS mode.

11. Press **X** to exit to the Main Menu.

12. Before file transfers can take place you will need to change one more parameter in the Binary File Transfer Menu. Type **B** to invoke this menu.

13. You should see the Binary File Transfer Menu. Notice that the protocol is called XMODEM. You want to use MITE. Press **P** and choose **M** for MITE.

14. Press **X** to exit to the Main Menu

15. Look at the Main Menu and you will see an S option to save the parameters you just changed. Saving the parameters will make it easier to use MITE the next time. Type **S**, and when asked for the file name, call it DIRORG for direct connect, originate mode. From now on you can change the parameters by simply typing the L option from the Main Menu to *load* the parameter file, DIRORG, into the computer.

16. You are almost ready to start transferring data. But first you must install MITE on the other computer following the same steps as you did for the first computer, *EXCEPT*, the role of the second computer should be set to ANS (answer). See step 11 above. Save this parameter file as DIRANS (direct connect, answer mode).

    PLEASE NOTE: You need two copies of MITE, one for each computer or a program for the other computer which supports one of the MITE supported protocols, e.g., XMODEM.

You are now ready to start transferring files

Both computers are now connected and both are running an installed version of MITE; one is in ORG mode and the other is in ANS mode. The Main Menu should be showing on both computer screens. Type **G** for GO on *both* computers. Now you should have complete interaction. That is, anything typed on one computer will appear on the screen of the other. This is also true of two computers connected via telephone with MITE. This communication capability would allow two operators to *talk* by typing messages to the other's screen.

G puts MITE in Terminal Mode.

Try this out now. You should see the message appear on both screens simultaneously. This is the best way to verify that you have indeed connected the two computers properly.

Type ^J to return to the Main Menu.

Trouble Shooting

If you have problems, this is the first time they will show up. If you do not see the message on both computers, there is most likely a problem with the cable or the baud rate that you set in the Parameter Menu. Type a ^J to return to the Main Menu. Now type a P to see the Parameter Menu. Check to see if the baud rate on both computers is the same and that one computer is in ORG mode and the other is in ANS mode. If so, change the baud rate to 1200 on both computers. Go back to the Main Menu and type G again to see if the problem has been solved. Some computers can not transmit at the higher speeds. If this still does not solve the problem, you may have the wrong cable. If you presently have a standard cable, try a flipped cable, or if you are using a flipped cable, try a standard cable. Finally, if all else fails, get help from your dealer or by calling Mycroft Labs.

Transmitting Files in Hard Wire (Direct Connect) Mode

There are two types of files: text files and binary files. A text file is something that yields meaningful results when listed to the console with the TYPE command. An example of a *binary* file is a .COM executable object file, a .REL relocatable object file, Wordstar internal files, various data files created by programs like SuperCalc, etc. In order to transfer this kind of file, a more powerful transfer mechanism is needed. Note that simple text files are a special case of *binary file* and may also be transferred with this mechanism. If binary file transfer is possible, then that is the preferred mechanism since retransmission of blocks is performed automatically on detection of errors. This greatly improves the chances of getting the file to the remote site intact. For the most part, binary file transfer is only practical with another computer using the same operating system.

You will notice on the sub-menus that there are options for text file upload, text file download, and binary file transfer. Upload means to send a file from your computer to the other computer. Download means to transfer a file from the other computer to your computer. Both text files and binary files can be transmitted using the binary file transfer option; the

difference being that the text will be verified during transfer, but the text will not appear on the screen while it is being transferred. For direct connect applications, it is preferable to use the binary file transfer option.

Transmitting a File Using the Binary Transfer Option

1. From the Main Menu, press **B** for the Binary File Transfer Menu. (If you will be doing multi-file transfers (see step 2 below), the protocol in this menu should be set to MITE.)

2. Choose to send a file or files. If you choose to send a file, you will be asked for the file name you wish to send. You can send multiple files by using the wildcard syntax. For example, B:*.COM would send all the files on drive B with a .COM suffix. B:*.* would send all the files on B drive. If you are not using the MITE protocol, you must send one file at a time. Put the disk containing the file you wish to send in drive B and type the file name including the drive designation, e.g., B:myfile. When you press <CR> the computer will attempt to send the file.

3. The other computer will be receiving the file, so you first must be sure that there is enough room on the disk to receive the file. One simple way of doing this is to put a blank, formatted disk in drive B of the receiving computer. From the Main Menu, type C for *command processor*, when asked for a command, type **DRIVE B:**. This will change the default drive for receiving the file to drive B.

4. On the receiving computer, type **X** to exit to the Main Menu and type **B** for Binary File Transfer.

5. Finally, type **R** to receive a binary file. (If you are not in the MITE protocol, you will be prompted for a file name.)

6. You should see **ATTEMPTING TO SYNCHRONIZE** on both computers now and a series of dots should appear on the screen. This indicates a successful transfer is taking place. At the end of transmission, a message indicating completion will be displayed.

   If you get a series of characters, like RRRRRRR, the transfer is not working. After a short time the attempt to transfer will abort and a message to that effect will be displayed. The most common reason for failure is that the baud rate is set too high for your system. Go back into the Parameter Menu, reset the baud rate for 1200, and try again.

Procedure for Installing MITE for a Modem Connection

Before beginning the procedure for installing MITE for a modem connection you should be aware of some general features of MODEMS that may possibly cause you problems.

Manual Dial Modems:

Cables: Manual dial modems are typically wired as DCE devices. Please check your modem and computer to determine which cable you will need. Also check the suggested cable configurations in the chapter titled *Modem and Cable Installation.*

Switches: Most manual dial modems have a few switches you will have to be aware of. One common switch is an ORG/ANS switch. In most cases this should be set on ORG. Also, since most modems hook into ordinary telephone equipment, there is a TALK/DATA switch to allow you to use the telephone to dial the other computer, listen for a continuous high pitched tone and then by throwing the switch to put the modem on-line. This switch should be on TALK when dialing and switched to DATA when you hear the carrier tone from the other computer.

Operation: The manual dial modem may have a circuit that wishes to think it is on-line regardless of whether there is a carrier signal coming from the other computer. This is a minor annoyance but no real problem. The problem can be solved by putting the MITE program into DIRECT CONNECT MODE by using the **O** (Options) Menu. You should be aware that this is possible.

Auto-Dial/Auto-Answer Modems:

Cables: Auto-dial/auto-answer modems are typically DCE devices, but because of a number of functions most can perform, they may use a nonstandard cable. You should check to see if your modem is listed in the chapter titled *Modem and Cable Installation.*

The most typical problem MITE users encounter with auto-dial modems is the use of improper cables or switch settings.

Switches: The auto-dial/auto-answer modems typically have a number of internal *DIP* switches that have to be set a certain way to work with MITE. *Please note - the modems are not necessarily sent from the factory with the proper settings.* Again, the appropriate switch settings should be in the chapter entitled *Modem and Cable Installation.* These switches are sometimes not visible without removing the modem's cover. If you are unfamiliar or intimidated by hardware, you should be able to get your dealer to help you set these switches. Generally the directions in the modem manual should be sufficient.

Operation: Most common auto-dial/auto-answer modems are supported by MITE and can be installed by following the procedure that follows. However, there will always be new products or obsolete products that may not be supported. If you are considering purchasing a new auto-dial/auto-answer modem for use with MITE, you might first check to see what modems are supported. If you go through the following installation procedure and do not see your modem in the list, you might have to call your dealer or Mycroft Labs for help installing MITE.

The following steps should help you install MITE for a modem connection. It is assumed that you have the computer connected to the modem with the proper cable and that the modem is connected to the phone line.

1. Make a copy of the MITE Master disk following the directions in your USER'S GUIDE.

2. Insert the working disk you just made into drive A. (It may already be there).

3. Reboot the system.

4. When the system prompt shows, type **INSTALL<CR>**

5. You are now presented with a list of specific Auto-dial (and many are also auto-answer) Modems and a choice for Manual Dial Modems. If your modem is in this list, press the appropriate key. This will complete the first stage in installing MITE for modem operation.

6. To complete the installation of MITE, you will have to run it. Assuming that you have the computer connected to the modem with the proper cable, and that the MITE disk you just installed is in drive A, reboot the computer.

7. At the system prompt, type **MITE<CR>**

8. You should see the Main Menu.

```
MITE v2.xx  -  Copyright (c) 1983, Mycroft Labs, Inc.
OFFLINE.  Bytes Captured =     0/#####.  Capture = OFF.
Site ID =

MAIN MENU
     G - Go Start Communications
     H - Hang Up Phone
     I  - Enter Site ID
     L - Load Parameters from Disk File
     S - Save Parameters on Disk File

               Sub-Menus:

     P - Parameter                O - Option
     U - Text File Upload         D - Text File Download
     B - Binary File Xfer         M - Macro Definition
     C - Command Processor        F - Character Filter

     X - Exit to CP/M

Enter Option  (? for help): _
```

Notice that in the upper left-hand corner is the word OFFLINE. This means that this is a modem installation and that the program does not detect connection with another computer. Beneath that you see the Main Menu and a list of sub-menus. To complete the installation you will have to set some parameters, so type a **P** for Parameter Menu.

NOTE: If the word **ONLINE** comes up in the upper left-hand corner at this point, you have a slight problem. This message means that the program is getting a signal from the modem that says it detects another computer (in spite of the fact that you haven't called one yet). This is not uncommon with manual-dial modems and can be solved by typing an **O** for the Option Menu and selecting **D** for direct mode. Otherwise all other steps remain the same.

9. You should now see the Parameter Menu.

```
MITE v2.xx  -  Copyright (c) 1983, Mycroft Labs, Inc.
OFFLINE.  Bytes Captured =     0/#####.  Capture = OFF.
Site ID =

PARAMETER MENU

     B - Baud Rate              =  300
     D - Data Bits              = 7
     P - Parity                 = EVEN
     S - Stop Bits              = 1

     R - Role (ANS/ORG)         = ORG
     M - Mode (Duplex)          = FULL

     A - Auto Redial Count      = 0
     N - Phone Number           =
     I  - Modem Init String     =
     H - Dial Prefix            = AT DT

     X - Exit to Main Menu

Enter Option  (? for help): _
```

This menu contains the information the program needs to effectively transfer data. The first parameter to set is BAUD RATE. Notice that the BAUD RATE is already set at 300. This is a normal value for MODEM connections. But some modems can operate at 1200 baud. For now leave the baud rate set at 300. However, should you decide to change the baud rate you would;

type: **B**
type: **1200 <CR>**

You would see 1200 appear next to Baud Rate on the Parameter Menu.

10. For now you can ignore the other parameter settings, but note: the role of the computer is set in **ORG** (originate) mode. This parameter can be changed to **ANS** (answer) mode by pressing the **R** key. Notice this is a toggle operation. If you press **R** again, it will flip back to ORG. When calling another computer, your microcomputer will be in ORG mode.

If another computer is going to be calling you, you will have to be in ANS mode.

Also note that there is a place to type in a phone number. If you have an auto-dial modem, you would type the number of the computer you want to call.

See special instructions regarding entering the phone number for auto-dial modems in the *Parameter Menu* section of this manual.

11. Press **X** to exit to the Main Menu.

12. Before you can begin you will need to change one more parameter in the Binary File Transfer Menu. Select option B.

13. You should see the Binary File Transfer Menu. Notice that the protocol is called XMODEM. XMODEM is a commonly used protocol on remote computers. However, if you are going to be calling another computer that will be running the MITE protocol, you will want to change the protocol to MITE so that you can do multi-file transfers with a single command. To do this:

    type: **P** (this brings up a list of protocols)
    type: **M** (to choose the MITE protocol)

14. Press **X** to exit to the Main Menu.

15. Look at the Main Menu and you will see an S option to save the parameters you just changed. Saving the parameters will make it easier to use MITE the next time. Type S, and when asked for the file name, call it MODORG for modem connect, originate mode. From now on you can change the parameters by simply typing L from the Main Menu to *load* the parameter file, MODORG, into the computer.

16. You are almost ready to call the remote computer. The other computer should be running a communications program (MITE or something similar), it should be in answer mode, and it should be awaiting your call.

There are several ways you can use MITE in the modem mode. For instance you can call a mainframe, and assuming you have proper access, you can operate the mainframe using your computer as a terminal. You can also save the results of a session with the mainframe using MITE's file capture option. These options are explained in more detail in the *General*

Information section. What follows is a sample session with MITE. It involves calling another computer. In this session, you will transfer a text file from your computer to the computer you are calling. This is referred to as **uploading** a text file.

Sample Session in Modem Connection Mode

First, you must call the other computer.

For a manual dial modem, follow steps 1 & 2. For an auto-dial modem, skip to step 3.

1. Dial the phone number of the computer you want to call.

2. When the computer answers you should hear a continuous high pitched tone in the telephone receiver. On your manual modem (or telephone in some cases), you will flip a switch to cause the modem to transfer from talk to data mode. At the next re-painting of the screen, you should see the message on the screen change from OFFLINE to ONLINE. You are now connected to the other computer. Pressing **G** with the Main Menu showing should put you into communication with the remote computer. Go to Step 4.

   Note: Some manual modems will always show the **ONLINE** message when MITE is run. This phenomena is due to the way the modem was constructed. In this case it will be necessary to go into the Option Menu (the O option from the Main Menu) and put MITE into the **DIRECT** mode. You are still going through the modem, but you have told the system to ignore the carrier detect message from the modem. All the other instructions remain the same.

3. If you have an auto-dial modem, and if you entered the number of the computer you wanted to call in the Parameter Menu, press **G** (for GO) with the Main Menu showing and MITE will dial the remote computer and establish the connection.

4. You should see a sign-on message from the remote computer at this point. If you were connecting with a mainframe, you would have to enter your user number and password.

5. You are in *terminal mode*; you can return to the Main Menu by pressing **^J**. This will not disconnect you from the other computer. Try this now.

6. From the Main Menu select option **U**, *text file upload*. This will display the Text File Upload Menu.

7. At this point, type **U** and you will be prompted for the file name to be uploaded.

8. Type in the name of the text file you wish to send (include the drive designator, e.g., B:SAMPLE).

9. When you press <CR> after the file name, the computer will send the text file to the other computer. To retrieve a text file from another computer you would use option **D** from the Main Menu (text file download). See the *Text File Download* section of this manual.

SUMMARY

This is only the beginning of learning to use MITE. The following sections contain detailed information about the menus and how to use them. As you become more familiar with MITE you will want to incorporate some of the advanced features described in the *Advanced Usage* section. You should skim the next section to familiarize yourself with the contents and use it as a reference guide to help you in your use of MITE. For other sample sessions and tips on usage, please review the *Examples of Usage* section. The *Examples of Usage* may prove to be valuable templates for your use of MITE.

SAVING AND LOADING PARAMETERS

PARAMETER FILES

MITE allows you to store all of its parameters in a file. The parameter file will contain all the settings and options that MITE is currently set for when the file is created. This means you can set up MITE with the telephone number, the duplex, the parity, and macros (we will talk about those later) to call a specific location and save all of it, so the next time you want to call, you don't have to enter all that information again. You simply load the parameter file, and it's all there.

USING THE SITE ID

Once you have a set of parameters entered, you may want to give it a name. This is useful for keeping track of your parameter files, and it also serves as a reminder of who it is you are set up to call. To give your list of parameters a name:

1. From the main menu, type: **I**

2. Enter an appropriate message of up to 60 characters. This message will appear in all future headings, after the words **Site ID.**

MAKING A PARAMETER FILE

Once you have your parameters set up and named, you are ready to save them in a parameter file:

1. From the main menu, type: **S**

2. Enter a file name (8 characters or less with an optional 3 character extension).

   If you do not specify an extension, MITE will put .PAR (for parameter) on the end for you. Pick a file name that will be easy to remember.

Now all the parameters that MITE had in its memory are on that file, and you can call them up whenever you need them.

LOADING A PARAMETER FILE

To retrieve a parameter file:

1. From the main menu, type: **L**

2. Type the file name of the parameter file you wish to use.

   Again, if you do not specify an extension, MITE will put .PAR on the end.

Now MITE has loaded all the parameters and instructions from the diskette file and put them in its memory. You are now ready to make a call.

MITE will also automatically load a parameter file when you first run it. To do this:

1. From the A0> prompt, type: **MITE**

2. Type a space.

3. Type the name of the parameter file you want MITE to load. If you don't specify an extension, MITE will assume that it is .PAR.

4. Press RETURN.

LOADING A PARAMETER FILE

To retrieve a parameter file:

1. From the Main Menu, type: **L**

2. Type the file name of the parameter file you wish to use. If you do not specify an extension, MITE will put .PAR on the end.

Now MITE has loaded all the parameters and instructions from the diskette file and put them in its memory. You are now ready to make a call.

MITE will also automatically load a parameter file when you first run it. To do this:

1. From the A> prompt, type: **MITE**

2. Type a space.

3. Type the name of the parameter file you want MITE to load. Again, if you do not specify an extension, MITE will assume that it is .PAR.

4. Press RETURN.

THE FIRST CALL

ENTERING THE PHONE NUMBER

Now that you have gone through the preliminaries, you are ready to use your modem. First, you are going to need a computer to call. If you haven't done so already, it would be a good idea to get the telephone number of a local Bulletin Board System or BBS. A BBS is a computer set up expressly for other people to call with their own computers. On a BBS you can post messages for other users of the system to read and send electronic mail to other people. Your local computer users' group should be able to provide you with a list of BBS's in your area.

For example, assume you are calling a fictitious BBS in New York called Kaypro Network System, or KNS, at 1-914-555-1212. Remember, it is entirely fictitious; do not try to call it.

(Note: The following procedure applies only if you are using the KAYPRO built-in modem or another intelligent modem.)

First you must tell MITE what number you want it to call. To do this:

1. From the main menu, type: **P**
   This selects the Parameter menu, which is where MITE stores the phone number it is going to call.

2. From the Parameter menu, type: **N**
   This selects the option to change the phone number.

3. Type the telephone number you want to call. For our example, that would be 1-914-555-1212. The dashes are optional and just add to readability; the computer ignores them.

   Type the number *EXACTLY* as you would dial it if you were dialing a telephone:

   - If you are at an office where you must first dial a 9 to get an outside line, put a 9 before the telephone number: 9-1-914-555-1212.

- If you are at a place where you cannot use a tone phone, you will have to put a P (for pulse) before the number: P-1-914-555-1212.

  (This feature may not work on all modems. In case of doubt, read your modem manual carefully)

- If the number you are calling is local to you, you do not need to enter the 1 + area code: 555-1212.

4. Press RETURN.

5. To return to the main menu, type: **X**

MAKING THE CALL

Once MITE knows where you are going to call, it is ready to go ahead and start dialing:

1. Make sure that the computer is connected to a telephone outlet with a proper cord.

2. From the main menu, type: **G**
   This tells MITE to go ahead and dial.

If you are not using the KAYPRO built-in modem or other intelligent modem:

1. Make sure that the computer is hooked up to a telephone outlet with a proper cord.

2. Dial the number you wish to call. Use a telephone that is connected to the same phone line as the computer.

   Wait for the number to answer. You should then hear a high pitched tone.

3. When you hear the tone, type: **G**

4. Carefully replace the telephone receiver on the switchhook.

The screen will clear, and at the top of the screen you will see the heading again. The message that said OFFLINE should still say OFFLINE. That message reflects the state of MITE when the heading is printed, and it does not change until the heading is printed again.

The words **Now Dialing:** will then appear, followed by the telephone number you entered. This means that your computer is now making the telephone call. A few seconds later, the words **Awaiting Carrier (ESC to abort)** will appear on the screen. MITE is now waiting for the computer at the other end to answer the phone. If MITE does not detect a computer in 45 seconds, it will hang up and return to the main menu. This will happen if the line is busy, the other computer is not hooked up, or you entered the wrong phone number.

If for any reason you want MITE to stop the call before it connects with the other computer, you can press the ESC key, and MITE will hang up and return to the main menu.

When MITE connects with the other computer, it will print **Carrier Detected.** At this point, you are connected with the other computer. This is an example of what a call to a BBS (in our case, Kaypro Network System) might be like:

1. If you don't see anything happen after a few seconds, press the RETURN key a couple of times. Sometimes you need to get a computer's attention that way.

2. The other computer may then print something like:

   YOU HAVE REACHED THE KAYPRO NETWORK SYSTEM
   IN NEW YORK.

   ENTER PASSWORD (CR FOR NEW USER):_

3. The computer has told you that you should press RETURN if you haven't called before (CR is short for *press Carriage Return*). Since you are a new user, press RETURN.

4. The computer then may ask you for some information about yourself:

   First Name: Pat
   Last Name: Smyth
   City: Anytown

   The BBS needs this information to keep track of its users and to know who gets which electronic mail messages.

What you should type next depends entirely on the other computer; every computer and every BBS is set up differently. There are a few rules of thumb to follow, though:

- If you ever get hopelessly lost, quite often you can get help or instructions by typing ? or **HELP**. Almost all dial-up computers have some way of helping out confused users.

- If you are using a service such as Dow Jones or CompuServe, read their manual carefully before calling. On a system where you pay by the minute, any time wasted making mistakes costs you money.

- If the BBS or other system asks you a question about your computer's format that you don't understand, try just pressing RETURN. Many systems have pre-defined default values that work with most computers. They will use these values if you don't enter any.

- If you do not get a carrier after several calls, try calling the number on a telephone. After about two rings you should hear the phone being answered; then you should hear a high-pitched tone. If the line is busy, then someone else is using the computer. If the number just keeps ringing, their computer may not be hooked up, or you may have a wrong number.

When you are done communicating, and want to end the connection:

1. To return to the main menu, type: **CTRL--J**
2. To hang up the phone, type: **H**

Note: You may press **CTRL--J** at any time during a connection. This will put you back at the main menu. From there you can change almost any parameter, and you should not lose the connection. To return to the connection, press **G** from the main menu.

THE MAIN MENU

```
MITE v2.xx  -  Copyright (c) 1983, Mycroft Labs, Inc.
OFFLINE.  Bytes Captured =     0/#####.  Capture = OFF.
Site ID =

MAIN MENU
     G - Go Start Communications
     H - Hang Up Phone
     I  - Enter Site ID
     L - Load Parameters from Disk File
     S - Save Parameters on Disk File

       Sub-Menus:

     P - Parameter                  O - Option
     U - Text File Upload           D - Text File Download
     B - Binary File Xfer           M - Macro Definition
     C - Command Processor          F - Character Filter

     X - Exit to operating system

Enter Option  (? for help):_
```

This is the principal menu which starts MITE and provides access to its other features through the sub-menus.

The G option takes you from the menu system to the terminal mode. The action taken depends upon:

1. whether you are in the Answer(ANS) or Originate(ORG) role,
2. whether or not you have an auto-dial and/or auto-answer modem,
3. whether you have the direct connect option ON.

If you have not changed the default settings of MITE and you have an auto-dial modem, you simply type G to initiate the terminal mode in the ORG role.

The ORG role causes MITE to attempt to establish a connection with the other system. If you type G and the carrier is already present, the message **Now resuming previous call** is displayed and you are placed in the terminal mode. If you have an auto-dial modem and you have entered a phone number (the N option on the Parameter Menu), MITE will dial the phone and wait up to forty-five seconds for the modem to detect a carrier signal indicating that the connection has been established. If you have a manual-dial modem and are in the ORG role, you should dial the phone and wait for the remote system to answer. When it has answered, enter the G command and you will be placed in terminal mode.

If you are in the ANS role, MITE waits for another system to establish communications. If carrier is already present and you enter a G command, MITE responds with **Now resuming previous call** and you are placed in the terminal mode. If the carrier signal is not present, MITE displays **Awaiting incoming call - abort with escape**. If you have an auto-answer modem, it will answer the phone and bring up the carrier detect signal. This will tell MITE to enter the terminal mode. If you do not have an auto-answer modem, you must answer the phone and turn your modem on to accomplish the connection.

If you are in the direct connect mode (two computers connected without modems) and in either the ORG or ANS role, MITE assumes that the connection has been accomplished and responds to the G by placing you in the terminal mode.

Once the connection has been established, MITE enters the terminal mode. In this mode, any character typed on the console keyboard will be sent to the remote system, and any character received from the remote system will be written to the console display. The only exceptions to this are the *trigger* characters (escape, macro, etc.) and the unwanted characters listed in the unwanted character filter (see the *Unwanted Character Filter Menu* section in this manual). For example, the *macro trigger* is used in conjunction with a second character to invoke one of the ten macro strings.

The H option can be used to hang up the phone at any time. With some on-line systems, this option may be the only way to terminate a session. Not all such systems support a *BYE* or *OFF* command that causes their carrier to go away. If you try to exit to the operating system with carrier still present, you will be reminded that it is still present and asked if you wish to hang up at that time. Normally, when the carrier is lost, the phone is automatically hung up and control returns to the Main Menu. It is then possible to exit to the operating system. The proper functioning of this option depends on the actual implementation. Not all hardware systems will allow the software to hang up the phone.

The I option allows you to enter a one-line description of your site or the site you will be using. This *site ID* will be printed on the third line of each menu page. When using MITE in the ANSWER role, this site ID should be set to something identifying YOUR site. This ID will be sent to the user dialing into your system, along with the standard MITE greeting. If you are in the originate role, the site ID is purely informational.

The L option allows you to load parameters from a previously saved parameter file. You will be prompted for a file name, which should be entered in the *d:fn.ft* format (e.g., B:SOURCE.PAR, RATOFF.1, B:CPMNET, etc.). If a file type is not specified, the default file type .PAR will be used. If the specified file is not found, you will be notified, and control will return to the Main Menu. If the file is found, the following parameters will be loaded:

Byte	Size	Contents
0	1	ASCII character M
1	1	Version number * 16 + release number
2	20	Modem Initialization String
22	2	Baud Rate (high byte first)
24	1	Parity (0=NONE, 1=ODD, 2=EVEN)
25	1	Number of Data Bits (0=SEVEN, 1=EIGHT)
26	1	Number of Stop Bits (0=ONE, 1=TWO)
27	1	Duplex (0=HALF, 1=FULL)
28	1	Escape Trigger Character
29	1	Wait-For-Echo Option (0=OFF, 1=ON)
30	1	Flow Control Option (0=OFF, 1=ON)
31	1	Flow Control Start Character
32	1	Flow Control Stop Character
33	1	End-Of-Line Handshaking Option (0=OFF, 1=ON)
34	1	Auto-LF Option (0=OFF, 1=ON)
35	1	Caps Option (0=OFF, 1=ON)
36	1	Macro Trigger Character
37	1	Protocol(0=XMODEM, 1=CLINK,2=HAYES, 3=IBM, 4=MITE, 5=TEXT, 6=XMODEMB)
38	60	Site ID
98	1	Garbage Character Count
99	1	Turnaround Character
100	1	Intercharacter Delay, msec
101	1	Role (0=ORG, 1=ANS)
102	1	Break Trigger Character
103	1	TWX Mode Flag (0=OFF, 1=ON)
104	1	Printer Echo (0=OFF, 1=ON)
105	1	Local Command Trigger Character
106	1	Remote Command Trigger Character
107	10	Unwanted Characters
117	1	Direct Flag
118	10	Modem Dial Header
128	640	Macro Strings (10 strings, 64 characters each)
768	30	Number to be Dialed
798	1	Auto Redial Count
799	1	Capture Indicator Flag
800	1	CRC Option (0=OFF, 1=ON)

If anything other than a valid parameter file is specified, MITE will abort the load command and issue the following message:

Invalid Parameter File

If a parameter file created with an earlier version of MITE is loaded, then you will be warned:

Warning - old parameter file.

At this point you should check all parameters, change if necessary, and resave them (on the same file, normally) with the new version of MITE.

The S option allows you to save the current parameters on a disk file for future use as a command line argument or as input for the L option. You will be prompted for a file name, which should be entered in the *d:fn.ft* format. If no file type is specified, the default file type .PAR will be used. All parameters listed under the L option will be saved.

Sub-Menus

The remaining options allow you to transfer control to any of the sub-menus. From a sub-menu, you will normally return to the Main Menu once you have accomplished the desired operation(s) on that sub-menu.

The P option selects the PARAMETER MENU. From this menu, you can easily check or set various communications parameters such as the baud rate, the number of data bits, etc. Once control is transferred to this menu, it remains there until you exit to the Main Menu with the X option.

The O option selects the OPTION MENU. From this menu, you can select the *trigger characters*, as well as several other options. Once control is transferred to this menu, it remains there until you exit to the Main Menu with the X option.

The U option selects the TEXT FILE UPLOAD MENU. From this menu, you can initiate the uploading of a text file to the remote system or select various options which affect how this is to be done. Once control is transferred to this menu, it remains there until you exit to the Main Menu

with the X option (with the exception of the *upload* function itself, which automatically returns control to the terminal mode once the upload is complete).

The D option selects the TEXT FILE DOWNLOAD MENU. From this menu, you can turn the text file capture mode ON or OFF. You can decide whether *flow control* (XON/XOFF handshaking) is used, and if so, what characters are used to start and stop the flow of data from the remote system. Once control is transferred to this menu, it remains there until you exit to the Main Menu with the X option.

The B option selects the BINARY FILE TRANSFER MENU. From this menu, you can initiate a transfer of any file (including executable files, etc.) to or from another system running MITE (or various other intelligent terminal programs) or an RCPM system. You can also select the protocol to be used. Once control has been transferred to this menu, it remains there until you exit to the Main Menu with the X option. As with the text file *upload*, once the actual Send or Receive functions have completed, control is automatically returned to the terminal mode.

The M option selects the MACRO STRING DEFINITION MENU. From this menu, the user can view or change any of the ten macro strings which may be invoked via the *macro trigger* character. Once control has been transferred to this menu, it remains there until you exit to the Main Menu with the X option.

The C option selects the SYSTEM COMMAND PROCESSOR, from which the user can issue a number of commands similar to those available in the *Console Command Processor* (i.e., command mode). These include such commands as DIR and TYPE. Help is available by typing a ?. Once control has been transferred to this menu, it remains there until you exit to the Main Menu with the X option or enter an *empty* command line (i.e., immediate CR after prompt).

The F option selects the UNWANTED CHARACTER FILTER MENU. From this menu, you can view and modify up to 10 ASCII characters that should be discarded immediately upon receipt. These unwanted characters will not be displayed to the console or saved in memory. The first two of these characters default to 7FH (DEL) and 1AH (CTRL--Z). The NULL

character (00H) is automatically discarded. Once control has been transferred to this menu, it remains there until you exit to the Main Menu with the X option.

The X option allows you to exit to the operating system. You will be asked to confirm this action before the exit is done:

Are you sure (Y/N)?

If you wish to exit at this time, type a Y or y. Any other response will return control to the Main Menu. If you do elect to exit, and a capture file is currently open, MITE will automatically close it for you and inform you of this operation with the following message:

Capture Complete. Now closing file d:fn.ft

If the carrier is still present, MITE will inform you with the following message:

Warning... Carrier still present. Hang up (Y/N)?

If you are finished with the session, reply with anything starting with Y or y, and MITE will hang up before exiting. If you merely wish to return to the operating system temporarily and plan to resume this link, reply with anything else (typically N or n) and the hang up function will not be performed.

On MS-DOS computers, there is an additional menu item on the main menu. The *T - Special Features* item allows you to change the color of the screen display, and to change the communications port used by MITE.

To perform either operation, from the main menu, type: T

The Special Features Menu will appear.

The first three items on the menu, *Comm Port*, *Port Base Address*, and *Interrupt Vector Address*, are for changing the serial port MITE uses. *You should not change these values under normal circumstances!* The default values are already set to insure proper operation on your KAYPRO. If you install a modem card or another serial interface, you may have to change these values. For the proper values, refer to the instructions provided by the manufacturer of the modem/serial card.

The C option toggles between Comm port 1, Comm port 2, and Comm port 3. The Base Address and Vector Address will change for each setting as well. These are the default addresses for the IBM-standard COM1, COM2, and COM3.

The B option allows you to enter, in hexadecimal notation, the base address of the serial port you wish MITE to use.

The V option allows you to enter, in hexadecimal notation, the interrupt vector address used for interrupt servicing.

These three parameters are saved with the rest of the parameters when you make a parameter file. This means you only have to enter these values once, and then save them in a parameter file.

The last four menu items allow you to change the color of the text and the background. To change one of these values:

1. Type the letter of the item you wish to change (N, M, H, or I). A menu of colors will appear.

2. Type the letter of the color you wish to select.

The colors are also included with the parameters saved on a parameter file.

THE PARAMETER MENU

```
MITE v2.xx  -  Copyright (c) 1983, Mycroft Labs, Inc.
 DIRECT.  Bytes Captured =     0/#####.  Capture = OFF.
 Site ID =

 PARAMETER MENU

      B - Baud Rate              =  300
      D - Data Bits              = 7
      P - Parity                 = EVEN
      S - Stop Bits              = 1

      R - Role (ANS/ORG)         = ORG
      M - Mode (Duplex)          = FULL

      A - Auto Redial Count      =  0
      N - Phone Number           =
      I  - Modem Init String     =
      H - Dial Prefix            = AT DT

      X - Exit to Main Menu

 Enter Option (? for help): _
```

The current value of the parameter is displayed to the right of the option description for most options. If the value is changed, it is immediately updated on the menu display. Some of the options prompt the user for input (e.g., Baud). Others merely toggle between two or three states when selected (e.g., Mode). This approach makes it easy to determine the current setting of all parameters at a glance and gives immediate feedback when they are being changed. With all parameters, options, macro strings, etc., any change(s) you make will stay in effect only until you exit to the operating system, unless you use the S option on the Main Menu to make the change(s) permanent.

The B option allows you to select a new baud rate. The Baud Rate determines how fast MITE will send and receive data. Almost all modems operate at a fixed baud rate, so you probably will not have to change this setting unless you buy a new modem. Most modems, including the one built into the KAYPRO, operate at 300 baud. A few modems operate at 1200 baud, however. If you are using MITE to communicate with a computer peripheral other than a modem, you may also need to adjust the baud rate. You will be prompted as follows:

Enter New Value:

Enter the new baud rate.

This rate may be entered in any base, with post radix, the default (and normal) base being decimal. If an illegal value is entered (one not supported by your implementation and/or hardware), the message **Illegal Value** will be displayed, and the rate will remain unchanged. If an empty line (i.e., immediate CR) is entered, the rate will remain unchanged. The default value is 300.

If you try to enter a baud rate that your KAYPRO can't use, MITE will return with the message, **Illegal Value**, and leave the baud rate unchanged.

The next three parameters (Data Bits, Parity, and Stop Bits) determine what format MITE uses to send and receive data. These three parameters are somewhat related--they all deal with which bits of data will have what significance.

These parameters MUST match those of the computer you are calling. If they do not, you will not be able send or receive information properly.

For more detail on these parameters, see *Appendix C - Introduction to Data Communications*

The D option allows you to select data bits. The Data Bit option determines how many bits will make a character. Most timesharing systems use 7 data bits, and most systems that support binary file transfers use 8. If you are doing a Binary File Transfer, both computers should be set to 8 data bits. This parameter toggles between 7 and 8. The default value is 7.

The P option allows you to select the parity of each character. Parity is a method of making sure that the character sent was the character received. If the Data Bits parameter is set to 7, then Parity probably should be set to EVEN. If the Data Bits parameter is set to 8, then Parity should be set to NONE. This parameter toggles between EVEN, NONE, and ODD. The default value is EVEN.

The S option allows you to select the stop bits. Stop bits mark the end of a character. If you have a 110 baud modem, you should probably change the Stop Bits to 2, otherwise leave it at 1. This parameter toggles between 1 and 2. The default value is 1.

The R option allows you to select the *role* that MITE will play in a connection. There are two different roles a computer can play in a modem connection. The computer that makes the call is in the originate role, and the computer that answers the phone is in the answer role. MITE is normally in the originate role, but, using this option, you can place it in the answer role. There are a few differences in the way MITE operates when in the answer role:

- When you type G from the main menu, MITE will respond with **Awaiting Incoming Call - abort with ESC**. If you are using an auto-answer modem, like the one built into the KAYPRO, it will answer the telephone and send an answer carrier automatically when the phone rings. MITE will then wait for the other computer to sent an originate carrier.

- When MITE first detects an incoming originate carrier, it will send the first line of the heading, the Site ID, and will send a message saying what the current Remote trigger character is.

- You or the originating party can use the Remote commands while MITE is in answer mode. Note that all remote commands affect the ANSWER system.

- MITE cannot dial outgoing calls while in the answer role.

The Role parameter toggles between ORG (originate), and ANS (answer). The default value is ORG.

The M option allows you to select the mode (or duplex) of the transmission. In half duplex, MITE sends every character you type directly to the screen. In full duplex, MITE waits for the other computer to return the character you just sent and then puts it on the screen. Full duplex insures that what you see on your screen is the same as what the other computer received, in case of transmission error.

If MITE is in full duplex, and the other computer is in half duplex, nothing will appear on the screen, because MITE will wait for the other computer to return the characters it has sent, and the other computer will not be returning characters. If MITE is in half duplex, and the other computer is in full duplex, every character you type will appear twice, because the other computer will be echoing back all the characters you send, and MITE will not be expecting the returned characters. In either case, you will have to change the duplex for normal communications.

The Mode (Duplex) parameter toggles between HALF and FULL. The default value is FULL.

The A option allows the user to specify the number of times an auto-dial modem will automatically redial the Phone Number (specified in the N option) if the connection is not completed in the first dialing. When MITE makes a telephone call and does not get an answer in 45 seconds, it will usually hang up and return to the main menu. If you use the Auto Redial parameter, however, you can have MITE call back repeatedly. The number in the Auto Redial Count is the number of times MITE will redial a call. If the Auto Redial Count is 2, MITE will make three calls (one original call, and two repeats). This feature is very useful if you are trying to reach a computer that is frequently busy.

The default is zero which means do not redial if the initial call is unsuccessful. The value can be set from zero to fifteen to indicate the number of desired retries before giving up. A number above fifteen is flagged as an illegal value and is ignored.

The N option allows the user to specify the phone number of the remote site. If a *smart modem* is being used, consult its user manual for specifics as to legal characters. To enter a blank phone number, enter at least one blank character. If the phone number field is blank, the dialing procedure will be skipped. Thirty characters are reserved for this string.

The I option permits the user to initialize the modem with up to twenty characters. This is for external modems that require certain commands before they will work properly. If you are using the built-in KAYPRO modem, you will not need to use this feature. Nevertheless, some smart modems require software switches to be set before they will operate properly with MITE. Many of these settings are preset via the MINSTALL program. If you did not select a specific modem type, you will have to consult your modem user manual for directions. The values entered here will be sent to the modem at initialization time. If control characters are required, they may be specified by entering the ^ symbol before the character. For example, carriage return would be indicated with ^M.

The H option allows the user to specify a string of up to ten characters which are to precede the number in the dial string. The Dial Prefix is primarily used for external modems as well, and you will not need to use it with the KAYPRO modem. For example, with the Hayes Smartmodem, an **AT D** must be sent to instruct the modem to dial a number. This string must be sent regardless of the number to be dialed. Therefore, the user may enter this string as the dial header string. Likewise, if a switchboard system is always used, part of the number may be included in the 10 characters allotted to the dial header, e.g., **AT DT9,**. The combination of the dial header and the number string gives the user 40 characters to express the dial string.

The X option allows control to return to the Main Menu.

THE OPTIONS MENU

```
MITE v#.## - Copyright (c) 1983, Mycroft Labs, Inc.
OFFLINE. Bytes Captured =     0/#####.  Capture = OFF.
Site ID =

OPTIONS MENU
        E  - Escape Trigger Char    = 0AH = ^J
        M  - Macro Trigger Char     = 1BH = ^[
        B  - Break Trigger Char     = 02H = ^B
        K  - Local Command Char     = 0BH = ^K
        R  - Remote Command Char    = 12H = ^R

        C  - Caps Lock              = OFF
        L  - Auto LF after CR       = OFF
        T  - TWX Mode               = OFF
        D  - Direct Connect Mode    = OFF
        Q  - Expand Tabs to CON     = ON

        X  - Exit to Main Menu

Enter option (? for help): _
```

The first five options on the menu deal with trigger characters. When you type a trigger character, MITE intercepts it and performs the function assigned to that character.

Since MITE intercepts trigger characters, it is impossible to send a character that is currently assigned as a trigger character. If, for example, you need to send a line feed (CTRL--J) to another computer, you will have to re-define the Escape Trigger character to something other than CTRL--J (the default value).

The value of the current trigger character is displayed in both hexadecimal and ASCII after the option it is assigned to. Control characters are represented with a caret (^).

The E option allows the user to specify an *Escape Trigger* character that will allow control to be transferred from *terminal mode* back to the Main Menu. Any time this user-specified character is typed while in link mode, the screen will be erased and the Main Menu will be displayed. From the Main Menu it is possible to initiate various functions, exit to the operating system. return to the link, or go to other menus. This *Escape Trigger* character should be specified as something not required on the remote system. The normal default value is 0AH (10 decimal), which is the ASCII Line Feed (LF) character, which is also a CTRL--J (^J). When this option is specified, the user will be prompted for a new ASCII character, which may be entered in any of three ways:

1. as a numeric value in any base with post radix (e.g., 10, 0AH, 00001010B, etc.) Note: the first character must be a decimal digit 0 to 9.

2. as a control code by entering a caret (^) followed by the character you are taking the *control* of (e.g., ^C for CTRL--C).

3. by entering an ASCII code directly (e.g., the TAB key). Simply pressing the key will result in the ASCII character with that value being entered. Note that if a numeric key is desired, the user must use the first method above for entering the value.

With the third method, certain characters cannot be entered because they have special meaning in the operating system and are processed according to their function. These include CTRL--P, CTRL--M (CR), CTRL--J (LF), CTRL--H (BS), CTRL--X, CTRL--U, and CTRL--R. Also, notice that both the hexadecimal value and the ASCII representation of the character are displayed in the menu. Control codes are represented as an upper case alpha character preceded by a caret (e.g. ^F for CTRL--F). Since many function keys use ESC as the first character of their output sequence, you should avoid using ESC as the *Escape Trigger* character. The Line Feed character (LF, 0AH, ^J) is a good alternative.

The M option allows you to specify the *Macro Trigger* character. The Macro trigger activates the macro routine. It is followed by the number of a macro. The Macro trigger character itself can be sent during communication by typing it twice in a row. This is the only trigger character to do so, however. The Macro trigger has a default value of ESC (^[).

The B option allows you to specify the *Break Trigger* character. Any time this user-specified character is typed while in *terminal* mode, a communications line BREAK function (SPACE condition for 150 milliseconds) will be performed. A BREAK is *not* a real character and is rather outdated, but is still required by certain computer systems (mostly IBM).

The K option allows you to specify the *Local Command Trigger* character. Any time this user-specified character is typed while in the link mode, you will be prompted with **Local Command?**. For further information, see *Local Commands* in the *Advanced Usage* Section. This function is disabled when the value 00H (null) is selected. The recommended value is ^K (0BH).

The R option allows you to specify the *Remote Command Trigger* character. Any time this user-specified character is typed on the ORIGINATE system while in the link mode, the ANSWER system will prompt you with **Remote Command?**. In response, the ORIGINATE user can issue *local* commands on the ANSWER system. This allows an ANSWER system to be put on-line in an unattended mode so that the ORIGINATE user can control both ends of a file transfer, check directory space, and perform other tasks all from the ORIGINATE system. The *Remote Command Trigger* character is specified by the ANSWER system operator. It is listed (if active, i.e., non-null) as part of the initial greeting when an incoming call is answered. Output of *local* commands invoked via the *Remote Command Trigger* character actually goes to both systems, and as a matter of convenience, remote commands may be invoked and/or entered from either console.

The C option allows you to select an automatic conversion of lower case characters to upper case both to and from the remote system. This affects only alphabetic characters and serves the same function as a CAPS LOCK key. The default value is OFF. Each time this option is selected, the value toggles between ON and OFF.

The L option allows you to select an Automatic LF (Line Feed) character to be sent on the local console any time a CR (Carriage Return) character is received. Some timesharing systems send only a CR at the end of each line, whereas most console terminals require both a CR and a LF to advance to the next line. The default value is OFF. Each time this option is selected, the value toggles between ON and OFF.

The T option allows you to enable or disable the Western Union TWX compatibility mode. The operation of MITE is modified to allow your KAYPRO to serve as a terminal on the TWX network. It is also possible to access any TELEX terminal in the world via either of two mechanisms supported by Western Union.

When the TWX mode of operation is selected, the following differences are in effect:

- Any time a CTRL--E is received, macro string number 8 will be sent as an *answerback* message.
- On establishing a connection, MITE will send a CTRL--E and await the returned *answerback*, echoing it to your console. Once this has been displayed, MITE will automatically ask for the name of the file to be uploaded and will send the file. Once the file is sent, control will be returned to the normal terminal mode.
- On detection of an incoming call, the normal header is suppressed.

It is recommended that you capture the entire session on disk or to your printer. *Be sure the baud rate is set to 110.* Since TWX mode is saved as a parameter, it should be sufficient to create a parameter file called TWX.PAR that will set up all parameters for accessing the TWX network.

The TWX Mode requires the following options to be set:

AUTO LF after CR	= ON	(option L on OPTION MENU)
Data Bits	= 7	(option D on PARAMETER MENU)
Parity	= EVEN	(option P on PARAMETER MENU)
Mode(duplex)	= HALF	(option M on PARAMETER MENU)
CR/LF Handshaking	= OFF	(option H on UPLOAD MENU)

The D option allows you to go into direct connect mode. The Direct Connect Mode makes MITE think that there is a carrier present, even if there is not. This is useful if you are connecting two computers together directly through modems (without using the telephone). When this option is on and you go into terminal mode, MITE will automatically start sending a carrier. It will not dial if it is in originate mode, and it will not wait for an incoming call if it is in answer mode.

Every time you press **D** from the Options menu, the Direct Connect Mode toggles from OFF to ON, or ON to OFF. The default value is OFF.

The Q option allows the user to expand tab characters to spaces or send the tabs themselves to the console. This will not expand tabs on transmission to the remote system, but only to the console. If this option is ON, any tab character sent to the console will be expanded to spaces to place the cursor at the next 8th character position. When receiving terminal control codes that MITE does not recognize, positioning of the cursor with tabs will cause invalid expansions. The escape codes will be counted as characters sent, but will have no effect on the cursor position. This option may be turned off to allow the terminal itself to position the cursor upon receipt of a tab. Each time this option is selected, the value toggles between ON and OFF. The default is ON.

The KAYPRO itself does not recognize tabs, so if this option is off, all tabs will be ignored, and since the MITE menus use tab characters for indentation, all the menus will be up against the left hand side of the screen.

The X option allows you to return control to the Main Menu.

THE TEXT FILE UPLOAD MENU

```
MITE v#.##  -  Copyright (c) 1983, Mycroft Labs, Inc.
OFFLINE.  Bytes Captured =     0/#####.  Capture = OFF.
Site ID =

TEXT FILE UPLOAD MENU
        U  - Upload Text File

        D  - Interchar.  Delay        = 000
        E  - Await Char.  Echo        = OFF
        H  - CR/LF Handshaking        = ON
        T  - Turnaround Char.         = 00H = ^@
        G  - Garbage Char.  Count     = 000


        X  - Exit to Main Menu

Enter option (? for help): _
```

A text file upload is a feature that allows you to send another computer a file from a diskette. The Upload feature will take a text file and send it to the other computer just as if you were typing it at the keyboard. This saves a great deal of time, which is especially important when calling long distance or when using a system that charges on a minute-by-minute basis.

A text file only contains characters you can enter from the keyboard. It is mainly composed of ASCII characters, such as letters, numbers, and punctuation. To see if a file is an uploadable text file, use the TYPE command to print it out on the screen. If you can read the file, and it does not produce garbage on the screen, then you should be able to upload it. If a file contains characters that you cannot enter from the keyboard, such as a .COM file, then you cannot send it using a text file upload.

The U option allows you to *upload* (send) a text file to the remote system as if it were coming from the keyboard of the local console. When this option is selected, you will be prompted for a file name, which should be entered in the *d:fn.ft* format. The specified file will be sent to the remote system using the conventions selected by the other options on this menu. Once the file has been sent, the communications link will be resumed automatically. If an XOFF character (13H or ^S) is received from the remote system during transmission, MITE will pause until an XON (11H or ^Q) is received before continuing to transmit the file. This pause will prevent many systems from losing data while transmitting. An upload can be aborted at any time by typing an ESC on the console keyboard.

You may also upload files from *terminal* mode. To do this:

1. From terminal mode, type the Local command trigger (CTRL--K).

2. Type: **READ**
   You will then see the prompt, **Enter Filename: _**.

3. Enter the name of the file you wish to upload.

4. Press RETURN.

At any time, you may abort the upload by pressing the ESC key.

The D option allows you to specify an *intercharacter delay*. The Intercharacter Delay allows you to slow down the rate of transmission by 0 to 255 milliseconds, so you do not send characters too fast for older, less sophisticated computers. You will probably never need to use this feature, but, like the Break trigger character, it is there in case you ever have a need for it. This function works in both FULL- and HALF-duplex. It is the only way to slow down text in HALF-duplex. The time starts with the actual transmission of the character, not counting the transmission time at whatever baud rate you are running. If the actual delay time is less than a single character time, there will be no effect. For example, at 300 baud, each character takes about 33 milliseconds to transmit, so values below 33 will have no effect. A value of 100 will result in about 10 characters per second being sent. This option does NOT affect baud rate; it merely inserts a variable length delay between characters which are being sent at the normal baud rate. The intercharacter delay works for text being sent through the macro strings as well as from a file. Since these characters are not being entered from the keyboard, the delay will be in effect. When this option is selected, you will be prompted for a new value. The default value is 0.

The E option allows you to enable or disable a *wait for character echo* mode. This option tells MITE to wait until it gets the echo of the character it just sent before sending the next one. This insures that you are not sending characters too fast for the other computer. Await Character Echo only works in full duplex mode, however. This option nearly doubles transmission time, so you may want to use it only if the character delay doesn't work.

When this option is enabled, MITE will compare each echoed character it receives against the character it sent and list the number of characters that didn't match at the end of the transmission (**nnnnn Compare Errors**). Each time this option is selected, the value will toggle between ON and OFF. The default value is OFF.

The H option allows you to select the end-of-line (CR/LF) handshaking mode. If MITE sends only one line of your file and then stops, you may need to turn off the CR/LF Handshaking (CR/LF stands for Carriage Return/Line Feed). If the CR/LF handshaking is on, MITE will wait after it sends a carriage return at the end of a line. When the other computer returns a line feed, MITE will send the next line. The reason for this is that many time-sharing computers have a slight pause after a line is entered, during which they are too busy to receive any more characters.

When they are ready to receive more characters, they will transmit a line feed.

However, many small computers, like the ones used for BBS's, (or online word processors, typesetters, etc.) do not send line feeds after a carriage return. This means that, if the CR/LF Handshaking is on, MITE will only send the first line and then wait for a line feed that will never come.

CR/LF Handshaking toggles between ON and OFF. The default value is ON.

The T option allows you to specify a *turnaround* character. When you upload text to some computers, they will give you a prompt at the beginning of each line; the prompt is usually something like a ? or a >. If you define the Turnaround Character as that prompt character, MITE will use it to detect the beginning of the next line. If the Turnaround Character is not 0 (^@), MITE will wait at the end of each line until it receives the turnaround character. After it receives the turnaround character, it will send the next line.

If MITE does not receive the prompt character after 20 seconds, it will go ahead and send the next line anyway. If you press the ESC key during that 20 seconds, MITE will automatically send the next line.

If you do not wish to use this feature, define the Turnaround Character as 0 (^@).

The G option allows you to set the *Garbage Character Count*. This count refers to the number of characters that MITE will wait for after transmitting a line and before starting to send the next line. Each such *garbage* character has a one-half (.5) second timeout. This feature allows the user to upload text to remote systems that send nulls, line numbers, or other extraneous characters at the start of each line before they are ready to receive the new line. Some computers will send line numbers, or some other information, at the beginning of every line you send. To make sure that MITE does not send the beginning of a line while the other computer is sending the line number, you can use the Garbage Character Count. A Garbage character is a character that has no meaning to MITE, but will still interfere with an upload. When MITE detects the start of a new line (either normally, or through the Handshaking and/or Turnaround options), it will wait for the other computer to send the number of characters specified in the Garbage Character Count. After it has received the

Garbage characters, MITE sends the next line.

MITE will wait for half a second for each Garbage Character. If it hasn't received a character for half a second, it will assume the character is not coming, and start waiting for the next Garbage Character.

If the Garbage Character Count is zero, this function is disabled.

When this option is selected, you will be prompted to enter a new value. The default value is 0.

Note: The H, T and G options can be used in combination. The H option is processed first, then the T option, and then the G option. This combination allows the user to select a mode such as *At the end of each line, wait for a ?, then ignore the next 2 characters with a one-half (.5) second timeout on each, and then send the next line.* Using the various options, it should be possible to upload text to virtually ANY on-line system.

The X option allows you to return control to the Main Menu.

THE TEXT FILE DOWNLOAD MENU

```
MITE v#.##  -  Copyright (c) 1983, Mycroft Labs, Inc.
OFFLINE.  Bytes Captured =     0/#####.  Capture = OFF.
Site ID =

TEXT FILE DOWNLOAD MENU
        C  - Capture Mode            = OFF
        I  - Capture Indicator       = ON
        A  - Append Captured Data
        W  - Write Captured Data
        R  - Reset Capture Buffer
        P  - Printer Echo            = OFF
        T  - Type Capture Buffer

        F  - Flow Control            = OFF
        Q  - Flow Start Char.        = 11H = ^Q
        S  - Flow Stop Char.         = 13H = ^S


        X  - Exit to Main Menu

Enter option (? for help): _
```

At some point, you will probably want to save some information you have gotten from another computer. Whether it is an encyclopedia entry from a computer data-base, a message sent to you on a BBS, or an article someone posted on a university mainframe computer, you will want a way of keeping a copy of whatever goes scrolling by on the screen. You can do this by using the Text File Download feature of MITE.

MITE has a part of memory, called a capture buffer, set aside so it can copy everything that appears on the screen. Later, you can tell MITE to save the buffer in a diskette text file.

The C option allows you to turn the text capture mode ON or OFF. The first time capture mode is enabled (or the first time after a Write operation), the user will be prompted for a file name:

Enter Filename:

This file name should be entered in the *d:fn.ft* format (e.g., FRED.TXT). Note that it is possible to specify any logical device (e.g., LST:) in addition to disk file names. If the specified file already exists, you will be notified of this fact and asked if you wish to overwrite it. If you later exit to the operating system without writing the captured data to disk with the W option, MITE will automatically write it for you at that time. Each time this option is selected, the value will toggle between ON and OFF. The initial value is OFF.

You may also turn the Capture Buffer on and off from *terminal* mode. To do this:

1. Type the Local command trigger (CTRL--K).
   You will then see the prompt, **Local Command (or ?): _**

2. Type: **CAPT ON**
   or type: **CAPT OFF**

3. Press RETURN.
   If you typed **CAPT ON**, you will see the prompt, **Enter Filename: _**

4. Enter the file name you wish to use to store the captured information.

5. Press RETURN.

The I option indicates to the user that the incoming data is being placed into the capture buffer. While downloading text, you may notice a colon (:) before every line. That is the capture indicator; it tells you that the capture buffer is on. It also warns you when the capture buffer is almost full. When the capture indicator changes from a colon to an asterisk (*), there is only 4k of memory space left in the buffer. When this happens, you should take the next possible opportunity to empty the buffer into the diskette file to avoid overflowing the capture buffer. The earliest opportunity will be when the sending system is waiting for input. If the sending system pauses before the capture buffer overflows, you can use the escape trigger character (normally a ^J in the Option Menu) to return to the Main Menu, select the Text File Download Menu, and use the A, W, or R option (of this menu) to effect the desired result. Alternatively, you can enter the local command trigger (normally ^K in the Option Menu) and then invoke one of the local commands such as Append, Write, or Reset (see details in the chapter titled *Advanced Usage* beginning on page 77).

If you want to turn the capture indicator off, you can do so by typing I at the Download menu.

If the I option is OFF, the first character of each line is the actual data transmitted. Each time this option is selected, the value toggles between ON and OFF.

The A option allows the user to write a partial file from the capture buffer. The data captured will be written to the file specified earlier and the buffer pointer will be reset to empty. Also, the capture mode will remain ON and the output file will remain open. This allows the user to capture several long sessions to the same file, flushing the buffer each time.

To do this from *terminal* mode:

1. Type the Local Command trigger (CTRL--K).
2. Type: **APPE <CR>**

The W option is used to write any captured data to the file that was originally specified when capture mode was first enabled. If flow control is enabled, this data may actually be the last (partial) buffer. When the write is complete, MITE will remind the user which file was being used:

Capture Complete. Now closing file d:fn.ft

The Write option will set the capture mode to OFF and reset the capture buffer to empty. When this option is selected, the write operation is performed at that time.

To do this from *terminal* mode:

1. Type the Local Command trigger (CTRL--K).
2. Type: **WRIT <CR>**

The R option allows you to reset the capture buffer to empty. This option is useful only when flow control is disabled.

The P option allows you to turn the *printer echo* function ON or OFF. Any time the printer echo is ON, any character that is written to the console during a link will also be written to the list device. In order for this option to function properly, the list device must operate at an effective baud rate that is higher than that of the communications link. For slower printers, it may be necessary to use the *capture to LST:* mechanism or to capture the data to a disk file for later printing. Each time this option is selected, the value toggles between ON and OFF. The initial value is OFF.

To turn on the printer echo from *terminal* mode:

1. Type the Local Command trigger (CTRL--K).
2. Type: **ECHO ON <CR>**

The T option allows you to list the current contents of the capture buffer to the console. The listing will pause every 23 lines (and at the end of file), until you hit a CR to continue or ESC to abort the listing. CTRL--S can be used to cause a pause in listing at any time. This feature is normally of use only when flow control is disabled (otherwise you can only list the last *partial buffer*).

The F option allows you to select whether *flow control* handshaking is to be used while capturing text. MITE takes advantage of the fact that many dial-up computers will stop their output when they receive a certain character, and continue output when they receive a certain character. When the Flow Control option and the Capture mode are on, MITE will try to stop the output of the other computer after receiving about 2000 characters. If the computer stops sending characters, MITE will then empty out the buffer into the diskette file. When it has emptied the buffer, it will send the character that tells the other computer to start sending again. With this feature, you do not have to constantly worry about how full the buffer is. XON/XOFF (CTRL--S/CTRL--Q) characters are usually used to accomplish this handshaking. Many systems will pause when you type CTRL--S and start back up when you type CTRL--Q. If flow control is enabled, MITE will attempt to use this mechanism to cause the remote system to pause while it writes to disk the data it has captured since the last pause. This is normally done every 2048 bytes. When the flow stop character is sent, MITE will wait until a full second has elapsed since the last character arrived before writing to disk. This is necessary since some systems have characters in transmission when the flow stop character is read. If the remote system supports this convention, it is highly recommended that you take advantage of it. Each time this option is selected, the value will toggle between ON and OFF. The default value is OFF.

Note that when flow control is enabled, some commands work somewhat differently. For example, the T option (Type capture buffer) will type only what is in the capture buffer currently, not what has been previously written to disk.

MITE will check to see if the start and stop characters are working. If they aren't, MITE will not use this option. It is a good idea, therefore, to leave this option on all the time.

The Q option allows you to specify the flow start character as described under the F option. Most systems that have this feature use CTRL--Q. When this option is specified, the user will be prompted for a new ASCII character, which may be entered in any of three ways:

1. as a numeric value in any base with post radix (e.g., 10, 0AH, 00001010B, etc.) Note: the first character must be a decimal digit 0 to 9.

2. as a control code by entering a caret (^) followed by the character you are taking the *control* of (e.g., ^C for CTRL--C).

3. by entering an ASCII code directly (e.g., the TAB key). Simply pressing the numeric key will result in the ASCII character with that value being entered. Note that if a numeric key is desired, the user must use the first method above for entering the value.

With the third method, certain characters cannot be entered because they have special meaning in the operating system and are processed according to their function. These include CTRL--P, CTRL--M (CR), CTRL--J (LF), CTRL--H (BS), CTRL--X, CTRL--U, and CTRL--R. Also, notice that both the hexadecimal value and the ASCII representation of the character are displayed in the menu. Control codes are represented as an upper case alpha character preceded by a caret (e.g., ^F for CTRL--F). The default value is ^Q.

The S option allows you to specify the flow stop character as described under the F option. Most systems that have this feature use CTRL--S. When this option is specified, the user will be prompted for a new ASCII character, which may be entered in any of the three ways described above. The default value is ^S.

The X option allows you to return control to the Main Menu.

THE BINARY FILE TRANSFER MENU

```
MITE v2.xx  -  Copyright (c) 1983, Mycroft Labs, Inc.
OFFLINE.  Bytes Captured =     0/#####.  Capture = OFF.
Site ID =

BINARY FILE TRANSFER MENU

     P - Protocol                    = XMODEM
     C - CRC Option (XMODEM)  = OFF

     S - Send File and Return to Link
     R - Receive File and Return to Link

     X - Exit to Main Menu

Enter option (? for help): _
```

A Binary File Transfer allows you to send or receive a file error free. You can send any type of file this way: text, programs, data, etc. MITE accomplishes this by using a standard file transfer format. MITE supports several different formats, but, by far, the most commonly used is XMODEM.

In order for you to use the Binary File Transfer, you must make sure that the other computer is capable of making a binary file transfer and that MITE is set to the same file transfer protocol as the other computer. If you are using an RCP/M (Remote CP/M) bulletin board, you should see XMODEM in the directory. If not, you may have to ask the system operator (SysOp) of the BBS whether or not it is capable of making a binary file transfer.

The P option allows you to select the protocol to be used in the transfer of binary files from one system to another. The protocols currently supported are:

MITE
The native MITE protocol (as used in the Mycroft Labs SEND/RECV utilities, see the August, 1982 *Dr. Dobbs Journal*). This protocol supports multi-file transfers, e.g., all files of type .ASM. No file name is required on the receive end. If you wish to receive files to a disk other than the current logged disk, you must first issue the DRIVE command in the System Command Processor.

XMODEM
The protocol used on most RCPM systems. It is compatible with XMODEM, MODEM7, MODEM80, and various other intelligent terminal programs. The XMODEM protocol checks each block of data it receives to make sure that there were no transmission errors. If there was an error in transmission, XMODEM will have the sending computer re-transmit the block. XMODEM also has the option of *Cyclic* Redundancy Checking (CRC). CRC increases the probability of catching an error, so it would be wise to use this option whenever possible.

XMODEM/B
This is the batch (multi-file) version of the XMODEM protocol. It supports batch style transfers using ambiguous file names, e.g., *.ASM. CRC error checking also is available by using the C option on this menu or through the local/remote command **CRC** (see the chapter on *Advanced Usage*). Receiving files with XMODEM/B is simpler than with XMODEM because the file name(s) is (are) not specified since it (they) is (are) sent along with the file(s). Files are received onto the current logged drive.

CLINK
The original Mycroft Labs binary file protocol. It is compatible with CLINK and CROSSTALK.

HAYES
The Hayes Terminal Program Verification protocol as used in the Smartcom communications package.

IBMPC

The IBM Asynchronous Support Package text file protocol. This is not strictly a binary protocol as only text files can be sent, and no error checking or retransmission is done.

TEXT

A simple ASCII text file protocol that can interact with a high-level language program on a minicomputer or mainframe to transfer text files with error checking and recovery. See Appendix E for an explanation of the TEXT protocol as used herein and for a copy of a sample FORTRAN program to use on the mainframe or mini.

When the P option is selected, the following display will appear. To change the protocol, you may enter one of the characters listed below. Enter a carriage return (CR) if you wish to leave the protocol unchanged.

Current protocol is: XXXXXX

X - XMODEM (Single)	**B - XMODEM/B (Batch)**
C - CLINK & Crosstalk	**H - HAYES (Smartcom)**
I - IBM Async	**M - MITE Multi-file**
T - TEXT (mainframes)	

Enter new protocol or CR for no change (? for help): _

Be sure that the system with which you plan to communicate supports at least one of these protocols. When the P option is selected, it will prompt for one of the supported protocols. The default value is XMODEM.

The C option allows the user to turn the CRC error checking ON or OFF. If you are using XMODEM or XMODEM/B, you will have to make sure the CRC (Cyclic Redundancy Check) option is OFF if the other computer is not using CRC, and that it is ON if the other computer is using CRC. This only applies to the XMODEM and XMODEM/B protocols. Each time this option is selected, the value will toggle between ON and OFF. The initial value is OFF. The CRC option on sending will default to OFF and will automatically switch to ON if the receiver is CRC-capable. The receiver needs to be set to have its CRC option set to ON or OFF depending on whether the sender is CRC-capable.

The S option allows a file to be sent using the currently selected protocol. When this option is selected, you will be prompted for a file name, which should be entered in the *d:fn.ft* format. At this time, MITE will start sending the specified file to the remote system.

You may also send files from *terminal* mode. To do this:

1. Type the local command trigger (CTRL--K).

2. Type: **SEND <CR>**
   You will now see the message **Current Protocol is:** followed by the name of the current protocol. Then you will see the prompt **Enter Filename:**

3. Type the name of the file you want to send, and press RETURN.

You will then see the message, **Attempting to Synchronize.** That means that MITE is waiting for the signal from the other computer to go ahead and send the file. When MITE gets the signal to go ahead, it will print a message saying that it is sending the file.

After that, MITE will print a period (.) for every block of 128 bytes sent successfully, an **R** for every block that had a transmission error, a **T** for a block that was not received in a certain amount of time, and a **U** for an unrecognizable command sent by the other computer.

When MITE is done sending the file, it will print the message, **All Files Sent. Resuming Link.**

The R option allows a file to be received using the currently selected protocol. When this option is selected, you will be prompted for a file name, which should be entered in the *d:fn.ft* format. At this time, MITE will start receiving the specified file from the remote system. A period (.) will be displayed on the console for each block received correctly. An **R** will be displayed for each block that is received in error. When the file has been completely received, MITE will display the message **File Received** and the terminal mode will be resumed automatically.

MITE will also permit the user to receive an ambiguous file list from a CROSSTALK system if several conditions are met. Since CROSSTALK does not include the file name in the protocol, but uses the remote command feature to indicate the beginning of a new transmission, MITE's remote command trigger must be set to ^C in order to be compatible with CROSSTALK. MITE must also be the answer system for the remote command to work. If MITE is the originating system, single file transmissions are required.

To receive a file from *terminal* mode:

1. Type the local command trigger (CTRL--K).

2. Type: **RECV**

3. Press RETURN.

   You will now see the message, **Current Protocol is:**, followed by the name of the current protocol. Then you will see the prompt **Enter Filename:** .

   If you are using XMODEM/B or MITE protocol, it will not ask for a file name because the name of the file will be sent along with the file itself.

4. Type the name you wish to use for the file.
   (Omit this step if you are using MITE or XMODEM/B.)

5. Press RETURN.

The sequence of events after that is almost identical to sending a file, with the exception that the messages say **receiving** instead of **sending**.

The X option returns control to the Main Menu.

UNATTENDED COMPUTERS

When you are connected to an unattended computer, like a BBS or a time-sharing mainframe, there will be no one at the other end to set up the computer for a file transfer. In most cases there are provisions to allow you to set up the other computer from your end of the connection.

On a BBS, there may be a menu option to upload or download a file. If you are not sure about how to do it, leave a message to the system operator (SysOp) of the BBS; he or she will be able to help you.

If you are using an RCP/M system, look for a file called XMODEM.COM in the directory. If it is there, then you can transfer files relatively easily.

To prepare an RCP/M system to receive a file:

1. From terminal mode, while the other computer is at the A0> prompt, type: **XMODEM R FILENAME.EXT** where **FILENAME.EXT** is the name of the file you are going to send.

2. Press RETURN.

You would then follow the MITE procedure for sending a file.

To prepare an RCP/M system to send a file:

1. From terminal mode, while the other computer is at the A0> prompt, type: **XMODEM S FILENAME.EXT** where **FILENAME.EXT** is the name of the file you are going to receive.

2. Press RETURN.

You would then follow the MITE procedure for receiving a file.

THE MACRO STRING DEFINITION MENU

```
MITE v2.xx  -  Copyright (c) 1983, Mycroft Labs, Inc.
OFFLINE.  Bytes Captured =     0/#####.  Capture = OFF.
Site ID =

MACRO STRING DEFINITION MENU
    0:  mailck^M
    1:  mail read^M
    2:  post scan cp/m^M
    3:  off^M
    4:
    5:
    6:
    7:
    8:
    9:  ^M@W^M@E@T=d1^M@T@c 30128^M@T>id tcm495 xxxxx^M

    X   - Exit to Main Menu

Enter Option or ? for help: _
```

This menu allows you to view and/or define up to ten pre-stored macro strings, each of which can be up to 62 characters in length. These strings are saved and loaded along with the parameters in the .PAR file. Typical uses for macro strings include semi- or fully automatic login or favorite commands. Macro strings may also be used to issue local commands or to modify menu options automatically. The example shown above might be used on The Source. To define the *n*th string, type the number of the desired string (0 to 9). You will be prompted for a new string, which will then be displayed in the menu following the string number. To exit from this menu, use the X option. Any printable ASCII characters can be entered directly.

To enter control codes (such as CR, LF, CTRL--X, CTRL--H), enter a caret (^) followed by the printable character corresponding to it. The following table will help in determining which characters to use for various control codes:

NUL	00H	^@	VT	0BH	^K	SYN	16H	^V
SOH	01H	^A	FF	0CH	^L	ETB	17H	^W
STX	02H	^B	CR	0DH	^M	CAN	18H	^X
ETX	03H	^C	SO	0EH	^N	EM	19H	^Y
EOT	04H	^D	SI	0FH	^O	SUB	1AH	^Z
ENQ	05H	^E	DLE	10H	^P	ESC	1BH	^[
ACK	06H	^F	DC1	11H	^Q	FS	1CH	^\
BEL	07H	^G	DC2	12H	^R	GS	1DH	^]
BS	08H	^H	DC3	13H	^S	RS	1EH	^^
HT	09H	^I	DC4	14H	^T	US	1FH	^_
LF	0AH	^J	NAK	15H	^U			

For example, to enter a Carriage Return (<CR> or 0DH) use the code ^M. Note that two carets in a row (^^) will be interpreted as a single caret. This means that the RS control code cannot be entered in this manner; therefore, it cannot be entered in a macro string.

In addition to standard and control characters, MITE provides several special characters which can control the way the macro is sent. They are designated by an @ followed by one or more characters. If you want your macro to print an @, you will have to enter an @@ so that MITE knows that you aren't using a special character. The special characters are as follows:

@E Turn on the *wait-for-echo mode*. When this mode is on, MITE waits for an echo of the character it just sent before it sends the next one. This insures that MITE will not send characters too fast for the other computer to process them. When MITE starts a macro, this mode is turned off.

@N Turn off the *wait-for-echo mode*. Most computers do not echo the initial carriage returns or passwords when you log in, so it is necessary to turn off the *wait-for-echo mode* for that part of the login sequence.

@W Wait until no characters have been sent for one second. This prevents MITE from sending characters while the other computer is still sending a message. After the other computer is through, MITE will wait one second, and if the other computer still has not sent any characters, then MITE will continue with the rest of the macro.

@P*n* Wait until no characters have been sent for *n*/10 of a second. This does the same thing as @W, except it allows a delay for a fractional part of a second. To cause MITE to stop sending your macro until no characters have been received for 3/10 of a second, you would put @P3 into your macro at the appropriate spot.

@T*x* Trap on the ASCII character *x*. This causes MITE to pause until it receives the specified character. If you wanted MITE to stop sending the macro until it received a question mark, you would put @T? into your macro at the appropriate spot.

@L*n* Link to macro string *n*. This causes MITE to activate the specified macro. Any characters remaining on a macro after this command will be ignored. If you wanted to make a macro 121 characters long, you could do so by putting the first 59 characters on macro #1, putting @L2 on the end, and putting the remaining 62 characters on macro #2.

If macro string 9 is defined, it will be sent automatically the first time a connection is established on an outgoing call. The console keyboard is also active when a macro string is being sent, and all characters except ESC will be sent to the remote system as usual. The ESC character is used to abort the current string.

NOTE: because remote system logins vary, none of the *auto login* features of currently available programs will work on all systems or even every time on a given system.

As an example of an auto login, the following string may be used to logon to The Source via TELENET:

^M@W^M@E@T=d1^M@T@c 30128^M@T>id tcm495 xxxxx^M

When invoked, this string performs the following functions:

^M	First CR to get attention (not echoed)
@W	Wait for 1 second
^M	Second CR to get attention (not echoed)
@E	Turn on wait-for-echo mode
@T=	Input and display characters until = is seen
d1^M	Response to **Terminal=** question on TELENET
@T@	Input and display characters until @ is seen
c 30128^M	Response to @ prompt on TELENET
T>	Input and display characters until > is seen
id ...^M	Source logon command

The *wait-for-echo* mode forces MITE to wait for the echo of each character that it sends before proceeding to the next character. This mode defaults to OFF at the start of each string. Some systems will lose characters if this wait is not enabled. However, many systems do not echo initial CR characters, passwords, etc., and require a means of turning this mode on and off on a character-by-character basis.

Macros are transmitted through a two-character sequence consisting of the *Macro Trigger Character* and one of the *index characters*. On many terminals and microcomputers, function keys are provided which will transmit a two-character sequence with a single keystroke. The macro facility of MITE is easier to use if the *Macro Trigger* and *index characters* are set up to reflect the machine being used. If INSTALL does not automatically patch MITE to work with the function keys on your machine, you may wish to patch the program to do so. While no assembly language programming knowledge is required, patching MITE does require the use of SID or DDT.

To invoke a macro string during a session, enter the *Macro Trigger Character* (see *Option Menu*), followed by the *n*th index character as set up for your terminal (see *Installation Notes*). Any time the *Macro Trigger Character* is typed during the session, if the next character is one of the ten installed *index characters*, then the corresponding macro string is sent as if it were coming from the keyboard. If the character following the *Macro Trigger Character* is NOT one of these index characters, then that second character is sent through normally. If the remote system needs the *Macro Trigger Character* itself, it can be sent by typing it twice. Note that this does not work on the *Escape Trigger Character*. See the Installation Notes for further tips on taking advantage of existing function keys.

Note: Macro string 8 is used as the *answerback* string in the TWX compatibility mode.

THE SYSTEM COMMAND PROCESSOR

MITE v2.xx - Copyright (c) 1983, Mycroft Labs, Inc.
OFFLINE. Bytes Captured = 0/#####. Capture = OFF.
Site ID =

SYSTEM COMMAND PROCESSOR

Enter command, HELP or ? for details: _

The Command Processor allows you to perform many operaitng system-type functions without having to exit MITE.

Available commands for CP/M computers:

COPY *new=old*	copy one file to another
DIR *x:afnx*	list directory of files
DRIVE *x:*	set default drive to *x*
ERASE *afnx*	erase file(s)
HELP	print available commands
LIST *x:ufn*	list file to printer
REN *old=new*	rename *old* file to *new*
RESET	make all drives read/write
SET *afnx atr*	set attribute on file(s)
SIZE *x:afnx*	display size of file(s)
SPACE *x:*	display space remaining on drive *x*
TYPE *ufn*	type file to console
USER *n*	select new user number

Available commands for MS-DOS computers:

DIR *x:afnx*	list files in directory
DIRS *afnx*	list all files in directory
CHDIR *newdir*	change current directory
DRIVE *x:*	select new drive as default
ERASE *x:afnx*	erase file(s)
HELP	print list of legal commands
LIST *x:ufn*	list file to printer
REN *old=new*	rename *old* file to *new*
RESET	make all drives read/write
SET *x:afnx $att*	give file(s) new attribute
SIZE *x:afnx*	list size of file(s) in k bytes
SPACE *x:*	show space available on drive x
TYPE *x:ufn*	list text file on console

The *system command processor* executes a number of commands similar to *direct* commands (e.g., DIR, TYPE), as opposed to transient commands (e.g., FORMAT, ED). When this option is selected, any number of these commands may be entered, one at a time. Once the command has been performed, you will be prompted as follows:

Enter <CR> to enter another command, X to exit to Main Menu:

The symbol *ufn* stands for **unambiguous file name**. An unambiguous file name can only represent a single file (e.g. FRED.ASM).

The parameters for many of the above commands can be specified in the same way as if you were typing them in at the system prompt, but in MITE they are actually more generalized. Anywhere you can specify an *afn* (ambiguous file name) in MITE you can specify an *afnx*. The symbol *afnx* stands for *ambiguous filename expression*, and may contain wildcard characters (? and *) as well as the + or - operators. For example, you can specify *all files of type .ASM and all files of type .COM except those files starting with the letter F* with the afnx ***.ASM+*.COM-F*.***. Note that if you want to refer to a single filename that includes the + and/or - characters, you must enclose that filename in single quotes. For example, to erase the file BASIC-E.COM, use the command: **ERA 'BASIC-E.COM'**

Any part of a command enclosed in brackets is optional.

If you need more detailed information on these commands, type a question mark (?) in place of a command, and an explanation will be displayed. The available commands are as follows:

CHDIR *newdir*

(For MS-DOS computers only.) Works the same way as the CHDIR or CD command does when typed in at the system prompt.

COPY *new=old*

(For CP/M computers only.) This command can be used to copy files from drive to drive, or to create backups on the same drive. You must specify both file names. If you specify a drive (A:) with no file name as the new file, the new file will have a blank file name; it *WILL NOT* be the same as the old file name.

DIR *x:afnx*

List the names of all of the files on drive *x* that satisfy the ambiguous file name expression (*afnx*). For example, **DIR *.ASM** will list the names of all files of type ASM. If the drive (x:) is not specified, the current drive is assumed. Names of R/W files will be preceded with the standard colon (:). Names of R/O files will be preceded with a greater-than sign (>). SYS files are not listed.

DRIVE *[d:]*

This command will change the current default drive. The command **DRIVE** by itself will tell you which drive is the current default drive. For example, **DRIVE B:** will set the default drive to B:.

ERA *x:afnx [q]*

(For CP/M computers only.) Erase all files on drive *x* that satisfy the ambiguous file name expression (*afnx*). For example, **ERA *.ASM** will erase all files of type ASM. If the drive (x:) is not specified, the current drive is assumed. If a second parameter of q is specified (ERA *.ASM Q), you will be asked YES or NO on each file before it is deleted. *CAUTION--If you type* **ERA *.***, MITE will not ask to make sure that you want to erase all files, as the operaitng system does; it will go ahead and erase the whole directory without question.

ERASE *afnx*

(For MS-DOS computers only.) Works the same way as described for the ERA command above, except there is no *q* option.

HELP

This command will print all available command processor commands (this list).

LIST *ufnx*

This command will output a file to the printer. It will not accept ambiguous file names--you must be specific.

REN *new=old*

(For MS-DOS computers, the syntax is REN *old=new*.) Give an existing file (*old*) a new name (*new*). Note that both *old* and *new* must be unambiguous file names. Drive designations (x:) may be included in *old* and/or *new*. For example, to change the name of the file VALERI.ASM to MANNI.MAC, type:

REN MANNI.MAC=VALERI.ASM (for CP/M computers)
REN VALERI.ASM=MANNI.MAC (for MS-DOS computers)

RESET

Make all drives Read/Write again. Equivalent to a CTRL--C (^C) in CP/M command mode. Should be done any time a new disk is mounted.

SET *x:afnx $att*

Give all files on drive *x* that satisfy the ambiguous file name expression (*afnx*) the new attribute *att*. If the drive (x:) is not specified, the current drive is assumed. Possible new attributes are:

$DIR - make files visible in directory
$SYS - make files invisible in directory
$R/W - make files Read/Write
$R/O - make files Read/Only

For example, to make all files of type COM on the current disk Read/Only, use the command: **SET *.COM $R/O.**

SIZE *x:afnx*

List size of specified file(s) in K bytes, then list total size of those listed. This can be helpful for estimating transmission time for a file.

SPACE *x:*

List space available on specified drive. If no drive is specified, the current drive is assumed.

TYPE *x:ufn*

List specified file to the console. CTRL--S (^S) may be used to pause (once paused, any character will restart the listing). Any other character will abort the listing.

USER *n*

(For CP/M computers only.) This command changes the current user area. User area *n* must be between 0 and 7 inclusive.

Note that all file name wildcard conventions (* and ?) used in command mode are supported.

THE UNWANTED CHARACTER FILTER MENU

```
MITE v2.xx  -  Copyright (c) 1983, Mycroft Labs, Inc.
OFFLINE.  Bytes Captured =     0/#####.  Capture = OFF.
Site ID =

UNWANTED CHARACTER FILTER DEFINITION MENU

    0:  7FH    =
    1:  1AH    = ^Z
    2:  00H    = ^@
    3:  00H    = ^@
    4:  00H    = ^@
    5:  00H    = ^@
    6:  00H    = ^@
    7:  00H    = ^@
    8:  00H    = ^@
    9:  00H    = ^@

    X  - Exit to Main Menu

Enter Option (? for help): _
```

This sub-menu gives the user the capability of ignoring certain characters received from the remote system. This feature may be useful when a microcomputer is communicating with a mainframe that uses the delete character (7FH) as a pad character. Many microcomputers will do a character delete from the screen when the delete character is received. In order to prevent this deletion, the user would select the 7FH as a character to ignore from the remote system. The 7FH character is a standard character to be ignored.

Another example involves the Dow Jones Service. This service will sometimes send a 1EH code which will cause certain computers to home the cursor without clearing the screen. This action can cause havoc for the user.

For situations like these, MITE has a character filter. The character filter allows you to ignore, or filter out, certain characters. Whenever MITE receives a character that is in the character filter, it blocks it out, and

keeps it from passing on to the screen. The character filter can hold up to ten characters.

Three methods of entering the value are allowed. The ASCII value of the character may be entered in decimal or in hexadecimal. The control character may be entered by pressing the Control key along with the appropriate character. In some cases the operating system will trap this information and will not return it to MITE. For these cases, a third method of entry is needed. Enter a carat (^) followed by the character (e.g., ^M to indicate a Carriage Return, 0DH).

ADVANCED USAGE

Local Commands

Experienced users frequently find themselves leaving the terminal mode and going back through the menus to check their directory, initiate file transfers, etc. The constant interruption of the screen display by menus can be an inconvenience for the more sophisticated user. Local Commands help you avoid these interruptions. The Local Commands are intended to duplicate menu functions in a way that the more practiced user will find to be faster and less intrusive. Local Commands are, therefore, shortcuts; they provide a somewhat simpler way to accomplish certain commonly needed communications functions. However, Local Commands can only be used when MITE is in terminal mode. Terminal mode is when your microcomputer, through MITE, is acting as a terminal to a remote system. Terminal mode can also be described as the condition where a character typed on your console is sent to the device on the other end of the communications link without interpretation by your system.

There are several exceptions to this non-interpretation in the terminal mode by your system. One exception is the CAPS LOCK feature. If it is ON, any lower case characters will be converted to upper case before they are sent to the other system. Another exception occurs when a *trigger character*, for example the MACRO trigger, is typed. The *trigger character* will be interpreted by your system and will invoke the prescribed option of MITE. The *trigger character* will not be transmitted to the device at the other end of the communications link.

You establish terminal mode by typing a G (for GO) in the Main Menu of MITE. If the device on the other end of the communications line is a system also running MITE, it also must initiate terminal mode by typing G with the Main Menu showing. In other cases, the device on the other end of the communications link may or may not need an overt action to establish communications from its end.

To invoke the use of a Local Command in terminal mode, the user would type the *Local Command Trigger* character. The default (preset) and recommended character is ^K. This *Local Command Trigger* can be changed through option K on the OPTION MENU. Do not set the trigger value to 00H since this effectively disables the function. Whenever the *Local Command Trigger* character is typed on the Local console while in terminal link mode, the prompt **Local Command?** is displayed. At this time, any

system command (see *System Command Processor*) or any one of the Local Commands listed below may be entered. Once the command has been processed, terminal mode will be automatically resumed. Note that only the first four characters of the Local Commands need be entered. If a file name is omitted, MITE will ask for it later.

Remote Commands

Remote Commands allow you to put a MITE answer system on-line so that other MITE users can dial into it and request directories, erase files, check space available, change the protocol, and/or send and receive files. In fact, all operations can be achieved from the originating system. Therefore, the answer system can be left totally unattended. The *Remote Command Trigger* character (if non-null) is displayed in the MITE header which appears on the screen of a calling system at the time a call is accepted by the answer system. This capability need only be present on the answer system; hence, earlier versions of MITE (or other terminal programs) can serve as the originate system.

The Remote Commands have the same syntax as the Local Commands. In other words, they are the same with the exception of the EXIT command. Exit cannot be issued as a Remote Command. There is a *Remote Command Trigger* character which invokes the Remote Command feature. The default (preset) and recommended character is ^R. This *Remote Command Trigger* can be changed via the R option on the *Option Menu*. When the *Remote Command Trigger* character is typed by either the originate or answer system user, the prompt **Remote Command?** will be displayed. In response, either user can enter a command, which is then executed by the answer system and the output from which is sent to **both** systems.

Syntax of Local and Remote Commands

APPEnd

Allows the user to write the captured data to the disk file without closing the file. Capture mode must be ON before you use APPEnd. The capture mode is left ON after the writing operation is completed. Since the file is not closed by APPEnd, another append or write to the same file is allowed. This is the same as the A option on the TEXT FILE DOWNLOAD MENU.

BUFFer

Prints the capture buffer statistics (i.e., number of bytes captured out of total space available).

CAPTure ON or CAPTure OFF

Allows the user to turn text capture mode ON or OFF. The first time capture is enabled or the first time after a WRITe, MITE will ask for a file name. This is the same as the C option on the TEXT FILE DOWNLOAD MENU.

CRC ON or CRC OFF

Allows the user to turn CRC error checking for either of the two supported XMODEM protocols ON or OFF. This is the same as the C option on the BINARY FILE TRANSFER MENU.

ECHO ON or ECHO OFF

The echo mode, when turned on, will send everything that appears on the screen to the printer also, so you can have a permanent record of the call. It is the same as the P option on the Text File Download menu.

EXIT

Exits to the operating system, closes open files, but does not hang up the phone. This command may *not* be issued as a Remote Command.

HELP Local or HELP L

Lists available Local Commands to the console.

HELP System or HELP S

Lists available system commands to the console.

MACRos
Lists all non-blank macro strings to the console.

PROTocol
Displays the current binary protocol, displays the list of available protocols, and allows a new protocol to be selected. This is the same as the P option on the BINARY FILE TRANSFER MENU.

READ *x:ufn*
Reads specified file from disk and sends it as if it were coming from the local keyboard without interpretation by your system. *x* is the drive name and *ufn* is an unambiguous filename. The drive name is optional; if it is not specified, the drive name is assumed to be the current logged drive. This is the same as the U option on the TEXT FILE UPLOAD MENU.

RECV *x:ufn*
Receives file transmission onto specified file using the currently selected binary protocol. *x* is the drive name and *ufn* is an unambiguous file name. If you are using the MITE binary transfer protocol, the *x:ufn* (drive and unambiguous filename) is ignored and the sending system will supply the file name. This is the same as the R option on the BINARY FILE TRANSFER MENU.

SCREen OFF or SCREen ON
This option allows the user to turn screen output OFF or ON. When OFF, characters received over the communications link will not be displayed on the screen. It is useful on those systems whose video displays are too slow to keep up with communications. This phenomenon most often occurs on machines without a separate terminal. Problems with losing characters at the beginning of lines would indicate this option might be necessary when uploading or capturing files.

SEND *x:afnx*
Sends specified file from disk using the currently selected binary protocol. *x* is the drive name and *afnx* is an ambiguous filename expression. This is the same as the S option on the BINARY FILE TRANSFER MENU.

USE *x:ufn*

Redirects input to the console from the keyboard to the file specified. *x* is the drive name and *ufn* is an unambiguous file name. This file may be used to modify MITE settings through the menu modes as well as for communicating with a remote system. Once this command has been issued, console input will be obtained from the file rather than the keyboard. This action will continue until the end-of-file is encountered, regardless of the correctness of the input file. At the end of file, control will be returned to the keyboard. If a macro string is active at the same time as the USE file, the macro string has precedence. That is, the *console* input character will be obtained first from a macro string if a macro is active. If there is no macro string active, the character will be obtained from the USE file if one is active. If neither a macro or USE file is active, the character will be obtained from the keyboard.

The following items should be taken into account when creating a USE file:

- All control characters in the file are ignored. This includes carriage returns and line feeds.
- To issue a control character, use the caret (^) followed by the character. For example, a carriage return (CTRL--M) is specified by ^M.
- To delay 1 to 9 seconds, use a **@D***n* where *n* is the number of seconds.
- To send an ampersand (@), send two, e.g., **@@**.
- An ampersand (@) followed by any character other than another ampersand or a **D** will cause the ampersand and the other character to be ignored.
- An **@D** followed by anything other than **1** to **9** will cause the **@D** and the other character to be ignored.

WRITe

Allows the user to write the captured data (or the last part of it if flow control is enabled) to disk. Capture mode must be ON before using WRITe. A WRITe closes the file and turns the capture mode OFF. This is the same as the W option on the TEXT FILE DOWNLOAD MENU.

EXAMPLES OF USAGE

Parameter Files

A parameter file is a type of file used by MITE to hold useful relatively static information. It usually has a file name like SOURCE.PAR or DIR.PAR, where the first part of the file name describes the content of the file (at least the creator of the file knows what it stands for), and the suffix .PAR identifies it as a parameter file.

Parameter files are used to save the redundant typing of needed information from the keyboard every time you want to run a program. Once you have saved the needed information in a parameter file, you can have the program *read* the information from the disk. This is much faster and easier than typing the information in every time.

Creating Parameter Files

When you run MITE the first time, you will notice that most parameters are preset. These are called the *default* parameters and they are contained within the MITE program itself. However, these default settings do not work in all situations. Therefore, you may need to change certain parameters so that communication between your computer and the remote computer can proceed properly. You may also wish to use MITE in many different configurations. For example, you might want to use it with a hard wire connection to rapidly transfer files, or you might want to use it to access an information utility such as The Source. Each of these uses requires that different parameters be set within MITE.

Fortunately, once you set the parameters in MITE for a particular application you can save them in a parameter file on the MITE disk using the S option from the Main Menu. The next time you wish to use that application, you can reset all the necessary parameters by simply loading the parameter file by using the L option from the Main Menu. Or even easier, you can have MITE automatically read the parameter file as it is being run by typing MITE *NAME* (where *NAME* is the name of your parameter file) at the system prompt (A>).

To change a parameter, you change the value of the default from within the proper menu. For instance, if you were going to communicate with another microcomputer running MITE, you would probably want to use the MITE protocol. This protocol is set from within the Binary File Transfer Menu. The default protocol is XMODEM. To change to the MITE protocol, you would:

1. Press **B** to go to the Binary Transfer Menu from the Main Menu. (Remember **B** means press the B key - either upper or lower case.)

2. Press **P** to indicate you want to change the protocol.

3. Press **M** to select the MITE protocol.

This sequence changes the protocol to MITE temporarily; that is, once you exit MITE, the change will be gone. If you reenter MITE, the default protocol, XMODEM, will again be in effect.

If you wish to use the MITE protocol consistently, you could save the parameters you changed above by returning to the Main Menu and typing S for *Save parameters on disk file.* At this point, you will be asked for a file name. If you typed MPRO (for Mite PROtocol), the new parameters would be saved in a file called MPRO.PAR on the MITE disk. The next time you wanted to use these parameters, you could run MITE by typing MITE MPRO. This would automatically load the new parameters into the MITE program. Another way to load a parameter file into MITE is to use the L option from the Main Menu to *Load parameters from the disk file.* When prompted for a file name, you would type MPRO and the new parameters would be loaded into the MITE program. This is a trivial example of using the save and load parameters option since it is just as easy to set a single parameter from within MITE as it is to load a parameter file. However, this capability is valuable when there are a number of parameters to be set, as would be the case when using an auto-dial modem to access an information utility such as The Source.

You should take time to read through each of the menu descriptions in this User's Guide. They describe the parameters you can set and give examples of when the defaults would need to be changed. You should also try changing some of the parameters in MITE, saving the changes in a parameter file, and loading the file, until you feel comfortable with the procedure.

In the next section, the use of MITE with a typical information utility, The Source, will be described.

A Typical Session on The Source

Preliminaries

Before you can sign-on The Source, or any information utility, there are certain preliminary steps you must take. These steps are listed below:

1. Get a subscription to The Source, CompuServe Information Service, Dow Jones, etc. You can purchase a subscription at most retail computer stores.

   When you buy your subscription you will get the following items:

a) an identifier code	e.g., **d1**
b) a system number	e.g., **30128**
c) a personal account number	e.g., **str555**
d) a password	e.g., **orpyak**

2. Determine if you are going to use TELENET to access the utility or if you are going to dial it directly. The access method dictates the sign-on procedure you will use and how you will set MITE up for auto-dial and auto-logon.

3. This assumes that you have MITE properly installed for your terminal and modem.

Installing the New Parameters

If you have a manual-dial modem, you simply dial the TELENET number (or dial the utility directly). When you hear the high pitched tone, switch your phone to the modem and follow the logon procedures in the introductory guide you were given at the time you purchased your subscription.

If you have an auto-dial modem, you can let MITE dial the utility and send the answering computer the proper logon messages. MITE does this through the use of **macro** strings that are stored in a parameter file. The following procedure is used to add the information you need for this automated process.

1. Run MITE and from the Main Menu type **P** for the Parameter Menu.

2. Press **N** to add the phone number you have to dial. Enter the phone number followed by a <CR>.

3. Press **X** to exit to the Main Menu.

4. Press **M** to get to the Macro String Definition Menu.

   NOTE: This menu allows you to view and/or define up to ten pre-stored macro strings, each of which can have up to 62 characters. When saved, these strings become part of the .PAR file. Typical uses for the strings include semi- or fully automatic login or favorite commands. Macro strings may also be used to issue local commands or to modify menu options automatically. To define a macro string, type the number of the slot where you want the string to go (0 to 9). You will be prompted for a new string. When you finish typing the string in and press return, the new string will be displayed in the menu following the string number you chose. To exit from the Macro Menu, use the X option. Any printable ASCII characters can be entered directly. To enter control codes (such as CR, LF, CTRL--X, CTRL--H), enter a caret (^) followed by the printable character corresponding to it. For example, to put a carriage return into a macro string, type in a caret ^ followed by an M (^M). See the Macro String Definition Menu Chapter for a complete table of control characters.

   There are several special control codes that may be used to automate the logon process on many remote systems. The characters currently available are:

@E	Turn on the wait-for-echo mode
@N	No echo wait (default)
@W	Wait until no characters received for one full second
@P*i*	Wait until no characters received for *i*/10 second
@T*n*	Trap on the ASCII character *n*
@L*i*	Link to macro string *i* (0-9)
^M	Carriage return

If macro string #9 is defined, it will be sent automatically the first time a connection is established on an outgoing call. Your keyboard is active when the macro string is being sent and you can use the ESC key to abort the current string.

5. Press 9 to change or add macro string #9.

6. Type the macro string that will send the proper sign-on information and press return to enter it into the menu. (The procedure for determining the correct macro string is presented below). Type X to return to the Main Menu.

Constructing an Auto-Logon String

As an example of how to construct an auto logon string, let's look at a typical sequence of operations for manually signing onto The Source. It consists of seven steps:

1. Press the return key twice, e.g., <CR><CR>

2. A **TERMINAL =** prompt should be displayed on the screen.

3. Type in the identifier code given to you when you received your account number. e.g., **d1<CR>**

4. An @ prompt should be displayed on the screen.

5. Type in your assigned computer system number preceded by a c followed by a space, e.g., **c 30128<CR>**

6. The display will show that you are successfully connected.

7. Send the logon command: type id and press the space bar, then type your 6 digit account number and press the space bar, then type your password and press the return key, e.g., **id str555 orpyak<CR>**

   You should now be connected to The Source.

Now let's translate the logon steps into a macro string that will allow the computer to send the proper information at the proper time. The example below constructs the macro string for each step.

SIGN-ON STEP	MACRO STRING CHARACTERS
1. Send 2 carriage returns (wait 1 sec between each). The time delay allows the remote computer time to process the first return before receiving the second.	**^M@W^M**
2. Turn on the wait-for-echo mode and wait for an equal sign (=) to be sent by the computer.	**@E@T=**
3. Send the terminal identifier code and a carriage return.	**d1^M**
4. Input and display characters until the @ is seen.	**@T@**
5. Send the assigned system number and a return.	**c 30128^M**
6. Input and display characters until the > is seen.	**@T>**
7. Send The Source logon command and a carriage return.	**id str555 orpyak^M**

The macro string that goes into slot #9 looks like this:

^M@W^M@E@T=d1^M@T@c 30128^M@T>id str555 orpyak^M

To end construction of the macro definition string, type a carriage return.

To save the new parameters you have just set, type X to return to the Main Menu, and use the S option to save the parameter file (you could name it SOURCE to remember what it is for).

Other Macros

You can also put commands to be used on the utility systems into macro strings. When you are on the system and you want to issue a command, you type the macro escape trigger character, usually a CTRL--L (^L), and the number of the macro string you wish to send. For example, the command MAILCK is used to check your electronic mail in The Source. MAILCK can be put into the zero macro string slot and when needed, a ^L0 will issue the command instead of you having to type it out. This is a simple example, but it is how the macro strings can be used.

Capturing a Session on a Disk File

It is easy to capture a session on The Source or any other utility on a disk file for later review. Before typing G (for GO) from the Main Menu, perform the following steps:

1. Put a formatted blank disk in drive B.
2. Type C to bring up the Command Processor.
3. Type **DRIVE B:<CR>**
4. Type X (to return to the Main Menu).
5. Type D (to bring up the Text Download Menu).
6. Type C (to turn the capture feature on).

   You will be asked for a file name.
   Type in **B:session.txt** (or any file name you wish).
7. Type X (to return to the Main Menu).
8. Type G to start the auto-dial sequence.

All the characters received will be stored in the file session.txt on drive B as long as there is enough room on the disk.

The following section provide more information about uploading and downloading text files.

Uploading and Downloading Text Files

At any point, you can turn on the text capture mode (see the *Text File Download Menu*). The first time you do this, you will be asked for a file name. This is the name of the file to which you want the captured data written. Until you terminate this mode with the W command or by exiting to the operating system, any character written to the screen while in link mode will be captured on that file. It is possible to temporarily disable the capture mode by turning it OFF, then back ON later. This mechanism can be used to make a permanent record of an entire session. If flow control is enabled, you may notice short pauses every 2,000 characters or so. Do not let these worry you - this is MITE taking a second to write a buffer of data to disk. Characters defined in the unwanted character filter are automatically discarded. The ASCII characters 00 (NUL), 1AH (CTRL--Z or logical EOF), and 7FH (DEL) are defaults in the unwanted character filter. To change or delete these characters from the filter or to add other characters to the filter refer to the *Unwanted Character Filter Menu.*

It is also quite easy to send text files to the remote system as if they were being typed on your keyboard. This feature can save a great deal of time on long-distance calls or on systems that charge on a *connect-time* basis. It can also allow you to leave nicely formatted messages (which you have created with a word processor before making the connection). To send files, put the remote system into a mode where it would normally expect you to start typing a message or text file (e.g., insert mode in an editor), then go to the Text File Upload Menu and issue the U command. You will be asked for a file name. This is the name of the file that you wish to send. DO NOT SPECIFY A BINARY FILE. It would probably confuse the remote system rather badly. You have several options concerning how the file is to be uploaded. See the *Text File Upload Menu* description for details.

Sending and Receiving Binary Files

Not all files are simple *text files*, i.e., a file that yields meaningful information when displayed on the console with the *TYPE* command. Some files are *binary* files. Examples of *binary* files are executable object files, various data files created by programs like SuperCalc. In order to transfer binary files, a more powerful transfer mechanism is needed. Simple text files are a special case of *binary files* and may also be transferred with this mechanism. If binary file transfer is possible, then it is the preferred mechanism since retransmission of blocks is performed automatically on detection of errors. This greatly improves the chances of getting the file to

the remote site intact. For the most part, binary file transfer is only practical with another microcomputer system.

There are numerous *on-line* systems around the country that have a large number of public domain programs on them. These systems are referred to as *Remote CP/M* (or RCPM) systems. They all have a program called XMODEM that supports file transfers in either direction, with error checking and automatic recovery. Any kind of file may be transferred using this program. In order to transfer files, you must have a program on your computer that knows how to communicate with XMODEM.

One of the nicer features of MITE is support for the file transfer protocol used by XMODEM. In order to use this feature, you must bring up the Binary File Transfer Menu and select the XMODEM protocol via the P option. This protocol may also be used for file transfer between two MITE systems as long as both systems have selected the XMODEM protocol. Most of these on-line systems have documentation on how to use XMODEM. Very briefly, it is run by specifying either an R (to receive) or an S (to send) as the first parameter and a file name as the second parameter (e.g., **XMODEM R FRED.ASM** to receive a file from you and write it to FRED.ASM on the RCPM system).

To receive a file from an RCPM system onto your disk:

1. Login to the RCPM system like you would with any other on-line system. Few, if any, of these systems have passwords.

2. Start the XMODEM program (in Send mode) on the RCPM system: **XMODEM S fn.ft.**

3. Escape from the terminal mode and bring up the Binary File Transfer Menu (option B on the Main Menu). Select the R option (to Receive a binary file) and specify the file name under which you want this file saved.

To send a binary file to an RCPM system from your disk:

1. Login to the RCPM system like you would with any other on-line system.

2. Start the XMODEM program (in Receive mode) on the RCPM system: **XMODEM R fn.ft.**

3. Escape from the terminal mode and bring up the Binary File Transfer Menu (option B on the Main Menu). Select the S option (to Send a binary file) and specify the name of the file you want to send.

   In either case, a period (.) will be printed on your console for each 128 byte block that is transferred successfully. An **R** will be printed for each block that had to be retransmitted. Once the transfer is complete, control will be returned automatically to the terminal mode.

To exchange files with a CROSSTALK user, a similar procedure is used. Sending files TO a CROSSTALK system is especially easy, if the CROSSTALK system has enabled a remote escape character. In any case, the CROSSTALK protocol must be selected (via the P option on the Binary File Transfer Menu). For the following examples, it is assumed that the CROSSTALK and MITE remote command escape (trigger) characters are set to ^C.

To receive a file from a CROSSTALK system onto your disk:

1. Initiate the telephone call from the CROSSTALK system to the MITE system. It is assumed that MITE is the answer system and the other system is the originate system. This is required in order to use the Remote Command facility in MITE.

2. After the connection has been established and the two users have agreed on the file to be sent, the CROSSTALK user should enter a ^C. The MITE system will respond to both systems with **Remote command:**

3. At this time, the CROSSTALK user should issue a remote receive command to the MITE system: **recv fn.ft<CR>**.

4. Enter ESC on the CROSSTALK system. It will respond with **Command:**. Enter **xm fn.ft<CR>**.

To send a file to a CROSSTALK system from your disk:

1. Establish a connection with the CLINK system as above.

2. After the connection has been established and the two users have agreed on the file to be sent, the CROSSTALK user should enter a

^C. The MITE system will respond to both systems with **Remote command:**

3. At this time, the CROSSTALK user should issue a remote send command to the MITE system: **send fn.ft<CR>.**

4. Enter **PgUp** on the CROSSTALK system. It will respond with **Command:**. Enter **recv fn.ft<CR>.**

In either case, a period (.) will be displayed for each block (typically 1024 bytes in length) transferred intact, and an R will be displayed for each block that had to be retransmitted. Various other characters may be listed under certain circumstances, such as T for timeout or C for CRC error.

Setting Up an Unattended Answer System

It is possible to set up an unattended answer system for other users to dial into so that they can send files to or receive files from your system. An unattended answer system can be useful for distribution of public domain software in clubs, etc. The user dialing into such a system can request a directory of files on any of your disks, erase files, type files, check file size, check space on the disk, check/set protocol, or start a file transfer in either direction.

To set up an unattended answer system, you set up a normal answer system and then define a non-null *remote trigger* character (R command on the Option Menu). Obviously an auto-answer modem is required for this type of operation. The recommended value (and normal default value) is CTRL--R (^R or 12H). Any time this character has a non-null value, its value will be displayed to any incoming user along with the normal greeting.

When someone dials into an unattended system, he can issue *local* commands on the ANSWER system by typing the *remote trigger* character. The phrase **Remote Command?** will be displayed on his system, at which time he may enter any *local* or *system* command. The command will be performed on the ANSWER system. The results of that command will be displayed on BOTH consoles. When in doubt, the dialed-in user may use the HELP command.

MODEM AND CABLE INSTALLATION

The following information will aid in the installation of MITE with various modems. The first part lists various modem types and any special configuration or cabling for these modems. If your modem is not included in the first section, read the modem manual and set the following switches:

1. DTR - Active, not forced on. This setting will allow the hang up to disconnect the modem from the telephone line.

2. DCD - Active, not forced on. This setting will allow MITE to detect the carrier signal from the modem.

If the computer or modem you are using does not support the DTR signal, MITE will not be able to disconnect (Hang up option) the modem from the phone line. A manual disconnection will be required.

If your computer does not support the DCD signal (e.g., those using Intel 8251), you may switch the DCD and DSR signals in the cable and modify the CIOS to use the DSR signal to indicate whether carrier has been detected.

If your modem does not support DCD, or if the modem holds it in the *ON* state all the time, you will not be able to use the Auto-Dial option of MITE. This restriction is caused by the method used to determine if there has been a modem to modem connection after dialing. If DCD is *ON* all the time, MITE assumes that you are already in communication with the remote system. See Appendix B and Appendix C for more on RS-232 interfacing and data communications.

These settings are known to work for the following modems:

Modem Type & Number	Switch # and Setting	Cable Mods
■ Anchor Signalman Mark I	None	Standard
■ Anchor Signalman Mark VII	None ¤	20-25 »
■ Anchor Signalman Mark XII	None	Standard &
■ BIZCOMP 1012	5,7,8,9 - ON (Up) 1,2,3,4,6 - OFF (Down)	Standard
■ CTS 212AH	2,5,6,7 - UP 1,3,4,8 - DOWN	Standard
■ Datec 212	1,5,6 - CLOSED 2,3,4,7,8 - OPEN	Standard
■ Hayes Smartmodem 300	1,2,4,5,6,7 - UP 3,8 - DOWN	Standard
■ Hayes Smartmodem 1200	1,2,4,5,6,7 - UP 3,8 - DOWN	Standard
■ Incomm 212A	RNG,BAL,DTR - UP 8HI - DOWN	Standard
■ Novation Auto-Cat 212	None	20-25 #
■ Novation J-Cat	None *	Standard
■ Novation Smart-Cat	2,4,5,6 - ON (Down) 1,3 - OFF (Up)	Standard
■ Novation Smart-Cat Plus	1,3,6,7 - OFF (Down) 2,4,5,8 - ON (Up)	Standard
■ Racal-Vadic VA212	None §	Standard

Modem Type & Number	Switch # and Setting	Cable Mods
■ Toyocom (BYTCOMM) 212AD	SW6 2,9 - ON 1,3-8,10 - OFF SW7 2,3,5,9 - ON 1,4,6,7,8,10 - OFF	Standard
■ U. S. Robotics 212	All OFF	Standard
■ UDS 103LP Direct Connect	None	Standard
■ UDS 212 A/D	1 - ON, 2,3,4 - OFF	Standard

» Remove modem connector pin 20 and insert into position 25. Connect modem connector pin 20 to modem connector pin 7.

& Does not support hang up.

* This modem must have another line connected to the DB-25 connector at the modem end. Please insert the pin on the orange wire into position #8 and the blue wire into the connector pin position #20 to allow auto-dial and auto-hang up. In the CIOS, to allow for auto-dialing, the WAIT50 subroutine should be changed for a 70 msec delay.

¤ To use pulse dialing with this modem, the CIOS source code must be modified to reverse the polarity of the SETOH routine. To take the modem off hook, the SETOH routine must bring the DTR status signal low instead of its normal high.

§ This modem must be software programmed by the user to activate the Data Carrier Detect signal from the modem. The user must set option #1 to 2 and then change option #16 to 2. As long as the user does not change option #1 to either 1 or 3, the DCD signal will be properly set (even after a power down). Please consult your modem instruction manual for more details.

Appendix A

AUTOMITE

(For CP/M computers only)

About AUTOMITE

AUTOMITE is a program that simplifies the use of MITE*, the Mycroft Labs terminal program, by taking care of the details involved in using a terminal program like MITE. With AUTOMITE and MITE, you can access most of the major database services (The Source, CompuServe, Dow Jones, etc.) by using your modem. All you have to do to use AUTOMITE, once you have subscribed to a service, is press a few arrow keys. Then AUTOMITE will automatically have MITE call the service you selected, and perform all the steps necessary to get you on the system. You only have to enter a small amount of information once (the telephone number, etc.) for each service you subscribe to, and after that, AUTOMITE will remember all the information for you.

AUTOMITE also allows you to set up your own menu items for other places you call frequently. This allows you to call your favorite Bulletin Board System (BBS), or other computer system, by using only a few keystrokes.

External Modems

If you have a KAYPRO that does not have an internal modem, or if you are not using the modem built into the KAYPRO, then you will have to install MITE so it will work with your modem. You must do this before you run AUTOMITE, or else MITE and the modem will not function properly together.
Your computer may or may not have a *Master Menu*, so to install MITE, choose the instructions from the following list that fit your needs.

* MITE is a trademark of Mycroft Labs, Inc.

IF YOUR COMPUTER HAS AN EXTERNAL MODEM AND DOES NOT HAVE A MASTER MENU

1. Insert the working disk for MITE into drive A.

2. Reset the KAYPRO by pressing the red reset button on the back panel.

3. Type: MINSTALL
   Press RETURN.

   You will then see a menu of the different modems you can use with MITE.

4. Type the letter or number of the modem you are using.

   The program will then modify MITE so it will work with your modem.

IF YOUR COMPUTER HAS AN EXTERNAL MODEM AND HAS A MASTER MENU

1. Reset the KAYPRO by pressing the red reset button on the back panel.

2. From MASMENU, select the menu item, *Telecommunications.*

3. Press RETURN.

4. Select the menu item, *Install new modem.*

5. Press RETURN.

   You will then see a menu of the different modems you can use with MITE.

6. Type the letter or number of the modem you are using.

   The program will then modify MITE so it will work with your modem.

RUNNING AUTOMITE
IF YOUR COMPUTER DOES NOT HAVE A MASTER MENU

1. Insert the working disk for AUTOMITE into drive A.
2. Reset the KAYPRO by pressing the red reset button on the back panel.
3. Type: AUTOMITE
   Press RETURN.

RUNNING AUTOMITE
IF YOUR COMPUTER HAS A MASTER MENU

1. Reset the KAYPRO by pressing the red reset button on the back panel.
2. From MASMENU, select the menu item, *Telecommunications.*
3. Press RETURN.
4. Select the menu item marked *AUTOMITE*
5. Press RETURN.

THE AUTOMITE MENU

You should now see the AUTOMITE menu. This is where you give AUTOMITE its instructions.

- On the left-hand side of the screen is the list of menu items; to select a menu item, use the up-arrow and down-arrow keys to move the dark bar over the item you want and press the right-arrow key or RETURN.

- On the right-hand side of the screen is a short description of the selected menu item on the left.

- On the bottom of the screen is a short help message. If you ever try to enter something AUTOMITE will not accept, look in this box for help.

USING AUTOMITE

Before you can use a database service, such as Compuserve or The Source, you must first subscribe to the service. Information on how to subscribe to each service is contained in the information box for that menu item.

When you subscribe to a service, they will send you the telephone number used to access the service, an account or ID number, and a password. You will also get instructions on how to use the service.

To set up AUTOMITE once you have subscribed to a service:

1. Select the service you wish to set up, and press RETURN.

   AUTOMITE will then ask you a few questions. These questions are specifically tailored for the service you chose.

2. Answer the questions.

   After you have answered the questions, AUTOMITE will place the bar on the menu item marked *Connect.*

3. Check your modem to make sure it is ready to dial.

4. Press RETURN.

   AUTOMITE will then save all the information you have entered, and will go call the service you set up.

To use AUTOMITE once you have a service already set up:

1. Select the service you wish to call and press RETURN.

   You will then see a menu with all the information you originally entered. If you ever need to change any of those items, you can do it from this menu.

2. Press RETURN.

AUTOMITE will then invoke MITE, which will automatically dial the number and enter all the information required for you to use the service.

When you are connected, there are two special key sequences:

- To get a directory of the current disk, type: ESCape 0
- To log off of the service you are using, type: ESCape 1

When you have logged off the system, you will see the message **Carrier Lost. Type CR to continue.** At this point, press RETURN to return to the AUTOMITE main menu. Then, if you wish to exit to CP/M or return to MASMENU, press ESCape.

OTHER OPTIONS

On the main menu is the menu item called *Other Options*. To use the Other Options, select that item and press RETURN. The *Other Options* include *Set up a New Site*, *Change a Site Name*, *Modify a Site*, and *Erase a Site*. These allow you to make any changes you need to make to existing files, or to create new ones of your own.

To set up a site of your own:

1. From the AUTOMITE Master Menu, select the *Other Options* item and press RETURN.

2. Select the item, *Set up a new site*, and press RETURN.

3. Enter the file name you want to use to save the site on disk and press RETURN.

4. Enter a short description of the site (up to 50 characters) and press RETURN.

5. Select the menu item, *Phone Number*, and press RETURN.

6. Enter the phone number and press RETURN.

7. If necessary, change the Baud and Parity/bits settings by following the instructions provided in the right-hand box on the screen.

   NOTE: Do not change these settings unless you are certain they must be changed for you to be able to call the site.

8. Select the menu item, *Save on disk*, and press RETURN.

Your new site will now appear on the AUTOMITE Master Menu with the rest of the sites. You can call it using the same procedure for calling the other sites.

The last item on the Other Options menu, *Run MITE Directly*, allows you to use MITE, which is the terminal program that AUTOMITE uses. If you feel you have the experience required to use a terminal program directly, you may want to try this option.

Appendix B

A Practical Guide to RS-232 Interfacing

The following information is intended to compile and explain in relatively simple terms enough of the details of the RS-232 standard to allow a technician to construct and/or debug interfaces between any two *RS-232 Compatible* devices. A more detailed coverage of the subject may be found in the book Technical Aspects of Data Communication by John E. McNamara (1977, Digital Press).

This guide is necessary because of the casual way that vendors implement *RS-232* interfaces, sometimes omitting required signals, requiring optional ones, or worse, implementing signals incorrectly. Because of this casual treatment and a lack of readily available information about the real EIA standard, there is often considerable confusion involved in trying to interface two RS-232 devices.

BACKGROUND

RS-232-C is the most recent version of the EIA (Electronics Industry Association) standard for low speed serial data communication. It defines a number of parameters concerning voltage levels, loading characteristics, and timing relationships. The actual connectors which are almost universally used (DB-25P and DB-25S, sometimes called *EIA connectors*) are recommended but are not mandatory. Typical practice requires mounting the female (DB-25S) connector on the chassis of the communications equipment and the male (DB-25P) connectors on the cable connecting two pieces of data communications equipment.

There are two main classes of RS-232 devices: **DTE** (Data Terminal Equipment), such as terminals, and **DCE** (Data Communication Equipment), such as modems. Typically, one only interfaces a DTE to a DCE, as opposed to one DTE to another DTE or one DCE to another DCE, although there are ways to do the latter two by building non-standard cables. Rarely, if ever, are more than two devices involved in a given interface. A serial port on a computer may be implemented as either DTE or DCE, depending on what type of device it is intended to support.

RS-232 is intended for relatively short (50 feet or less), relatively low speed (19,200 bits per second or less) serial communications. Both asynchronous and synchronous serial encoding are supported. As *digital* signals (switched D.C. voltage, such as square waves) are used, a very wide bandwidth channel (such as direct wire) is required. A limited bandwidth channel (such as a phone circuit) would cause severe and unacceptable distortion and consequent loss of information.

RS-232 will support **simplex**, **half-duplex**, or **full-duplex** type channels. In a simplex channel, data will be traveling in one direction only, e.g., from DCE to DTE. An example might be a *Receive Only* printer. In a half-duplex channel, data may travel in either direction, but only in one direction at any given time and the line must be *turned around* before data can travel in the other direction. An example might be a Bell 201 style modem. In a full-duplex channel, data may travel in both directions simultaneously. An example might be a Bell 103 style modem. Certain of the RS-232 *handshaking* lines are used to resolve problems associated with these modes, such as which direction data may travel at any given instant.

If one of the devices involved in an RS-232 interface is a real modem (especially a half-duplex modem), the *handshaking* lines must be supported, and the timing relationships between them are quite important. These lines are normally much easier to deal with if no modems are involved. In certain cases, these lines may be used to allow one device (which is receiving data at a higher rate than it is capable of processing indefinitely) to cause the other device to pause while the first one *catches up*. This use of the handshaking lines was not really intended by the designers of the RS-232 standard, but it is a useful by-product of the way such interfaces are typically implemented.

Much of the RS-232 standard is concerned with support of **modems**. These are devices which can convert a serial digital data signal into an analog signal compatible with a narrow bandwidth (e.g., 3 kHz) channel such as a switched telephone circuit, and back into serial digital data on the other end. The first process is called MO*dulation*, and the second process is called DEM*odulation*, hence, the term **MODEM**. The actual process used (at data rates of up to 1200 bits per second) is Frequency Shift Keying (FSK), in which a constant frequency sine wave (called the *carrier*) is shifted to a slightly higher or slightly lower frequency to represent a logic 0 or logic 1, respectively. In a half-duplex modem, the entire available bandwidth is used for one direction. In a full-duplex modem, the available bandwidth is divided into two sub-bands. There is both an *originate carrier* for data from the terminal to the computer and an *answer carrier* for data from the computer to the terminal. The actual frequencies (in Hertz) used on the Bell 103A full-duplex modem are:

Signal	State	Originate	Answer
logic 0	SPACE	1270	2225
carrier		1170	2125
logic 1	MARK	1070	2025

THE STANDARD CIRCUITS AND THEIR DEFINITIONS

For the purposes of the RS-232 standard, a **circuit** is defined to be a continuous wire from one device to the other. There are 25 circuits in the full specification, less than half of which are likely to be found in a given interface. In the simplest case, a full-duplex interface may be implemented with as few as three circuits. There is a certain amount of confusion associated with the names of these circuits, partly because there are three different naming conventions (common name, EIA circuit name, and CCITT circuit name). The table below lists all three names, along with the circuit number (which is also the connector pin with which that circuit is normally associated on both ends). Note that the signal names are from the viewpoint of the DTE (e.g., Transmit Data is data being sent by the DTE but received by the DCE).

PIN	NAME	EIA	CCITT	DTE/DCE	FUNCTION
1	CG	AA	101	---	Chassis Ground
2	TD	BA	103	-->	Transmit Data
3	RD	BB	104	<--	Receive Data
4	RTS	CA	105	-->	Request To Send
5	CTS	CB	106	<--	Clear To Send
6	DSR	CC	107	<--	Data Set Ready
7	SG	AB	102	---	Signal Ground
8	DCD	CF	109	<--	Data Carrier Detect
9*				<--	Positive Test Voltage
10*				<--	Negative Test Voltage
11					(usually not used)
12§	SDCD	SCF	122	<--	Secondary Data Carrier Detect
13§	SCTS	SCB	121	<--	Secondary Clear To Send
14§	STD	SBA	118	-->	Secondary Transmit Data
15¤	TC	DB	114	<--	Transmit Clock
16§	SRD	SBB	119	<--	Secondary Receive Data
17¤	RC	DD	115	<--	Receive Clock
18					(usually not used)
19§	SRTS	SCA	120	-->	Secondary Request To Send
20	DTR	CD	108.2	-->	Data Terminal Ready
21*	SQ	CG	110	<--	Signal Quality
22	RI	CE	125	<--	Ring Indicator
23*		CH	111	-->	Data Rate Selector
		CI	112	<--	Data Rate Selector
24*	XTC	DA	113	-->	External Transmit Clock
25*				-->	Busy

In the preceding table, the character following the pin number means:

- * rarely used
- § used only if secondary channel implemented
- ¤ used only on synchronous interfaces

The direction of the arrow indicates which end (DTE or DCE) originates each signal, except for the ground lines (---). For example, circuit 2 (TD) is originated by the DTE and received by the DCE. Certain of the above circuits (11, 14, 16, and 18) are used only by (or in a different way by) Bell 208A modems.

A **secondary channel** is sometimes used to provide a very slow (5 to 10 bits per second) path for return information (such as acknowledge (ACK) or negative acknowledge (NAK) characters) on a primarily half-duplex channel. If the modem used supports this feature, it is possible for the receiver to accept or reject a message without having to *turn the line around*, a process that usually takes 100 to 200 milliseconds.

On the above circuits, all voltages are with respect to the Signal Ground (SG) line. The following conventions are used:

Voltage	Signal	Logic	Control
+3 to +25	SPACE	0	On
-3 to -25	MARK	1	Off

Note that the voltage values are inverted from the logic values (e.g., the more positive logic value corresponds to the more negative voltage). Note also that a logic 0 corresponds to the signal name being *true* (e.g., if the DTR line is at logic 0, that is, in the +3 to +25 voltage range, then the Data Terminal IS Ready).

ELECTRICAL CHARACTERISTICS OF EACH CIRCUIT

The following criteria apply to the electrical characteristics of each of the above circuits:

1. The magnitude of an open circuit voltage shall not exceed 25V.

2. The driver shall be able to sustain a short to any other wire in the cable without damage to itself or to the other equipment, and the short circuit current shall not exceed 0.5 ampere.

3. Signals shall be considered in the MARK (logic 1) state when the voltage is more negative than -3V with respect to the Signal Ground. Signals shall be considered in the SPACE (logic 0) state when the voltage is more positive than 3V with respect to the Signal Ground. The range between -3V and 3V is defined as the transition region, within which the signal state is not defined.

4. The load impedance shall have a DC resistance of less than 7000 ohms when measured with an applied voltage of from 3V to 25V, but more than 3000 ohms when measured with a voltage of less than 25V.

5. When the terminator load resistance meets the requirements of Rule 4 above, and the terminator open circuit voltage is 0V, the magnitude of the potential of that circuit with respect to Signal Ground will be in the 5V to 15V range.

6. The driver shall assert a voltage between -5V and -15V relative to the Signal Ground to represent a MARK signal condition. The driver shall assert a voltage between 5V and 15V relative to the Signal Ground to represent a SPACE signal condition. Note that this rule in conjunction with Rule 3 above allows for 2V of noise margin. Note also that in practice, -12V and 12V are typically used.

7. The driver shall change the output voltage at a rate not exceeding 30 volts per microsecond, but the time required for the signal to pass through the -3V to +3V transition region shall not exceed 1 millisecond or 4 percent of a bit time, whichever is smaller.

8. The shunt capacitance of the terminator shall not exceed 2500 picofarads, including the capacitance of the cable. Note that when using standard cable with 50 picofarads per foot capacitance, this limits the cable length to no more than 50 feet. Lower capacitance cable allows longer runs.

9. The impedance of the driver circuit under power-off conditions shall be greater than 300 ohms.

Note that two widely available integrated circuit chips (1488 and 1489) implement TTL to RS-232 drivers (4 per chip) and RS-232 receivers to TTL (also 4 per chip) in a manner consistent with all of the above rules.

DEFINITION OF THE MOST COMMON CIRCUITS

1 **CG** Chassis Ground

This circuit (also called Frame Ground) is a mechanism that insures that the chassis of the two devices are at the same potential to prevent electrical shock to the operator. Note that this circuit is not used as the reference for any of the other voltages. This circuit is optional. If it is used, care should be taken not to set up ground loops.

2 **TD** Transmit Data

This circuit is the path whereby serial data is sent from the DTE to the DCE. This circuit must be present if data is to travel in that direction at any time.

3 **RD** Receive Data

This circuit is the path whereby serial data is sent from the DCE to the DTE. This circuit must be present if data is to travel in that direction at any time.

4 **RTS** Request To Send

This circuit is the signal that indicates that the DTE wishes to send data to the DCE (note that no such line is available for the opposite direction; hence, the DTE must always be ready to accept data). In normal operation, the RTS line will be OFF (logic 1 / MARK). Once the DTE has data to send and has determined that the channel is not busy, it will set RTS to ON (logic 0 / SPACE) and will await an ON condition on CTS from the DCE, at which time it may then begin sending. Once the DTE is through sending, it will reset RTS to OFF (logic 1 / MARK). On a full-duplex or simplex channel, this signal may be set to ON at initialization and left in that state. Note that some DCEs must have an incoming RTS in order to transmit (although this is not strictly according to the standard). In this case, this signal must either be brought across from the DTE or provided by a wraparound (e.g., from DSR) locally at the DCE end of the cable.

5 **CTS** Clear To Send

This circuit is the signal that indicates that the DCE is ready to accept data from the DTE. In normal operation, the CTS line will be in the OFF state. When the DTE asserts RTS, the DCE will do whatever is necessary to allow data to be sent (e.g., a modem would raise carrier and wait until it stabilized). At this time, the DCE would set CTS to the ON state, which would then allow the DTE to send data. When the RTS from the DTE returns to the OFF state, the DCE releases the channel (e.g., a modem would drop carrier) and then sets CTS back to the OFF state. Note that a typical DTE must have an incoming CTS before it can transmit. This signal must either be brought over from the DCE or provided by a wraparound (e.g., from DTR) locally at the DTE end of the cable.

6 **DSR** Data Set Ready

This circuit is the signal that informs the DTE that the DCE is alive and well. It is normally set to the ON state by the DCE upon power-up and is left in that state. Note that a typical DTE must have an incoming DSR in order to function normally. This line must either be brought over from the DCE or provided by a wraparound (e.g., from DTR) locally at the DTE end of the cable. On the DCE end of the interface, this signal is almost always present and may be wrapped back around (to DTR and/or RTS) to satisfy required signals whose normal function is not required.

7 **SG** Signal Ground

This circuit is the ground to which all other voltages are relative. It must be present in any RS-232 interface.

8 **DCD** Data Carrier Detect

This circuit is the signal whereby the DCE informs the DTE that it has an incoming carrier. It may be used by the DTE to determine if the channel is idle so that the DTE can request it with RTS. Note that some DTEs must have an incoming DCD before they will operate. In this case, this signal must either be brought over from the DCE or provided by a wraparound (e.g., from DTR) locally at the DTE end of the cable.

15 **TC** Transmit Clock

This circuit provides the clock for the transmitter section of a synchronous DTE. It may or may not be running at the same rate as the receive clock. This circuit must be present on synchronous interfaces.

17 **RC** Receive Clock

This circuit provides the clock for the receiver section of a synchronous DTE. It may or may not be running at the same rate as the transmit clock. Note that both TC and RC are sourced by the DCE. This circuit must be present on synchronous interfaces.

20 **DTR** Data Terminal Ready

This circuit provides the signal that informs the DCE that the DTE is alive and well. It is normally set to the ON state by the DTE at power-up and is left in that state. Note that a typical DCE must have an incoming DTR before it will function normally. This signal must either be brought over from the DTE or provided by a wraparound (e.g., from DSR) locally at the DCE end of the cable. On the DTE side of the interface, this signal is almost always present and may be wrapped back around to other circuits (e.g., DSR, CTS and/or DCD) to satisfy required handshaking signals if their normal function is not required.

Note that in an asynchronous channel, both ends provide their own internal timing, which (as long as they are within 5% of each other) is sufficient for them to agree when the bits occur within a single character. In this case, no timing information need be sent over the interface between the two devices. In a synchronous channel, however, both ends must agree when the bits occur over possibly thousands of characters. In this case, both devices must use the same clocks. Note that the transmitter and receiver may be running at different rates. Note also that BOTH clocks are provided by the DCE. When one has a synchronous terminal tied into a synchronous port on a computer via two synchronous modems, and the terminal is transmitting, the terminal's modem supplies the Transmit Clock. The Transmit Clock is brought directly out to the terminal at its end, is encoded with the data, and is sent to the computer's modem. The computer's modem recovers the clock and brings it out as the Receive Clock to the computer. When the computer is transmitting, the same thing happens in the other direction. Whichever modem is transmitting must supply the clock for that direction, but on each end the DCE device supplies both clocks to

the DTE device.

All of the preceding information applies to interfacing a DTE device to a DCE device. In order to interface two DTE devices, it is usually sufficient to provide a *flipped* cable (or a null modem) in which the pairs (TD, RD), (RTS,CTS), and (DTR,DSR) have been flipped. The TD of one DTE is connected to the RD of the other DTE and vice versa. It may be necessary to wrap various of the handshaking lines back around from the DTR on each end in order to have both ends work. In a similar manner, two DCE devices can be interfaced to each other.

An RS-232 **break-out box** is particularly useful in solving interfacing problems. This is a device which is inserted between the DTE and DCE. First, it allows you to monitor the state of the various handshaking lines (light on = signal ON / logic 0) and watch the serial data flicker on TD and/or RD. Secondly, it allows you to break the connection on one or more of the lines (with dip-switches) and make any kind of cross-connections and/or wraparounds (with jumper wires). Using this device, it is fairly easy to determine which lines are not functioning as required and quickly build a prototype of a cable that will serve to interface the two devices. At this point, the break-out box can be removed and a real cable built that performs the same function. An example of this kind of device is the International Data Sciences, Inc. Model 60 *Modem and Terminal Interface Pocket Analyzer* (also called a *bluebox*). Care should be taken with this type of device to connect the correct end of it to the DTE device or the lights and switches will not correspond to the actual signals.

Appendix C

Introduction to Data Communications

Basic Concepts - Information Encoding

One of the fundamental concepts in the data communications field is that of **encoding** information from human-sensible form(s) into machine-sensible form and **decoding** machine-sensible back into human-sensible. A teletype-compatible terminal is a simple hardware device which is capable of performing both of these actions. There are several common means of encoding information so that a machine can *sense* it, almost all of which involve reducing the information to **binary** as a first step. Information theory tells us that in the general case, any information can be reduced (reversibly) in this manner. A single binary digit (**bit**) can be represented by the presence or absence of a current, one of two possible voltages, one of two possible audio frequencies, etc. Likewise, there are several common conventions for associating a group of bits to represent a particular character, numeric value, machine instruction, etc.

From one point to another within a terminal (e.g., the link between the keyboard and the rest of the terminal), a simple encoding scheme would typically be used which is called **TTL level parallel**. The TTL stands for *Transistor-Transistor Logic*. Under the TTL encoding scheme, a logic "0" is represented by a voltage in the range 0 to 1.3 VDC, and a logic "1" is represented by a voltage in the range 3.6 to 5.0 VDC. The current involved is very small; hence, TTL links must be physically very short (typically 2 to 3 feet at most). There are no widely adopted conventions concerning how to arrange the signals on a TTL link or even which signals to have present.

From one device to another (e.g., terminal to modem) one would typically use either **current loop** (on older equipment) or **EIA RS-232-C** (on most all recent equipment). A current loop is implemented by physically breaking a wire loop (or equivalent) in which a 20 mA current is flowing. Here a logic "1" is represented by the presence of current (MARK) and a logic "0" by the absence of current (SPACE). Historically, a long chain of *half-duplex* teletypes were hooked up *serially* on a single loop (often with the "return" wire being one rail of a train track). When any one terminal was typed on, the entire loop was *made* and *broken*; hence, the current was present and absent in a binary representation of the character that was typed. Since all the printer mechanisms in the loop decoded the information present in the current and printed the character corresponding to whatever pattern(s) were detected, they all printed the character typed.

Once again, no real standards exist concerning the type of connectors to use; hence, moving a current loop device from one environment to another usually involves rewiring connectors.

As computer peripherals began to proliferate, and all of them had their own proprietary (and incompatible) interface conventions, the Electronics Industries Association (EIA) decided to provide a generalized interface standard. If a given manufacturer stuck reasonably close to it, he stood an excellent chance of being able to *plug* his equipment directly into other such devices using standard, readily available cables and/or connectors. Although there are a bewildering variety of *subsets* of this standard (virtually no one ever implements the entire standard), one can usually connect two **RS-232 compatible** devices together with a minimum of difficulty. The RS-232-C specification not only specifies the physical encoding scheme (logic "0" = +3 to +25 VDC, typically +12, logic "1" = -3 to -25 VDC, typically -12), it also specifies which pins of a *recommended* connector (DB-25P/DB-25S) are to carry which signals and how those signals interact (*handshaking*). As somewhat higher currents are used than in TTL, a simple RS-232 link may be somewhat longer, typically up to 50 feet.

Often, one needs to connect two devices over even longer distances (possibly even over thousands of miles). As the **switched telephone network** is already present and is reasonably inexpensive to use, engineers have designed devices which allow one to couple two RS-232 devices via a voice grade phone line, such that the information is encoded into one of two possible audio frequencies in the one to three kilohertz range and back again. Since this process involves both **MO***dulation* and **DEM***odulation* of a carrier frequency, these devices are called **modems**. With modems the distance is limited only by the extent of the phone network. As in all other areas of data communications, there are a number of similar, yet incompatible, mechanisms for encoding information. Each method is optimized for some combination of bandwidth, signal quality, and cost.

Basic Concepts - Information Structure

The next level above representing individual bits in some machine-sensible form is to impose some sort of structure on groups of bits. Once again, there are several commonly used conventions, each optimized for a given set of parameters.

Returning to the example of the keyboard TTL link, because of the short length of the connecting cable, chances are that such a link would use a *byte parallel* interface. That is, the bits (in this case seven) that represent a given character would all be transmitted simultaneously over separate wires. By maintaining an order among the various wires, the character can be received intact by the rest of the terminal's circuitry.

Often, one wishes to reduce the required number of wires over a link to a minimum. An RS-232 link can be established with as few as three wires (two if information flows in only one direction). This is achieved by allowing multiple bits to *time-share* a single wire. There is more than one convention for doing this: *asynchronous serial* and *synchronous serial.* The connection between a terminal and its modem is typically RS-232 asynchronous serial.

Asynchronous serial involves sending information with precise timing between bits of a given character, but with arbitrarily long (including zero-length) intervals between characters. In order to allow this variable length interval, certain overhead information is required on each character. Normally, during an intercharacter gap, the line is in the logic "1" (MARK) state. When a character starts, there is a single *start bit* which is always logic "0" (SPACE). Once the start bit has occurred, then the *data bits* occur, least significant bit first. One typically finds either 7 or 8 such data bits on recent equipment. Following these bits, one often finds a *parity bit*, which is a simple error detection mechanism. If running with *even parity*, this bit will be chosen so that the total number of bits (in the data bits + parity bit part of the character) is an even number. Following the parity bit (if it is present) are 1 or 2 *stop bits* (always logic "1" (MARK)). Stop bits simply insure that the line returns to the idle (MARK) state for at least one (or two) bit time(s), so that the hardware can recycle and be ready to detect the next start bit.

This technique is commonly employed in situations where a human determines the timing between characters (such as in most terminals with keyboards). It is suitable for relatively low speed applications (1 to 1000 characters per second). A single inexpensive LSI chip, called a UART (Universal Asynchronous Receiver / Transmitter), is available to convert

parallel data to or from asynchronous serial data. Note that all connections to the UART are at TTL levels; consequently, TTL to RS-232 and RS-232 to TTL level shifters (such as 1488 and 1489) are usually found in conjunction with a UART.

Synchronous serial involves sending characters not only with precise timing between bits of a given character, but at regular intervals (no gap between characters), typically in relatively long (e.g., 1024 character) blocks. No overhead is required on each character, although parity is still sometimes used. However, certain overhead information is required at the start and end of each block. There are two main types of synchronous serial: **byte control protocols** (BCP), such as DDCMP (DEC's Digital Data Communications Message Protocol) and BISYNC (the older IBM standard) and **bit oriented protocols** (BOP) such as SDLC (Synchronous Data Link Control) and HDLC (High-level Data Link Control). A BCP can function with conventional byte-oriented hardware, but a BOP requires special bit-oriented hardware.

The primary function of the overhead information is to achieve **synchronization**, such that both ends agree on the boundaries between bytes in the continuous stream of bits. Typically, this overhead information also supplies a certain amount of redundancy for sophisticated **error detection** (or even **error correction**). Synchronization is used in relatively high speed applications (100 to 100000 characters per second). Note that since the timing is more critical (and must be sent along with the data), and the speed is generally higher, modems for synchronous serial data are normally more complex and expensive that those for asynchronous serial data.

Usually even higher levels of organization than just described are imposed on synchronous serial data, up to the **message** level and beyond. This includes mechanisms to support acknowledgment of messages received intact, requests for retransmission of messages received with errors, end of transmission status, etc.

Any serial communications link can be set up in one of three modes: *simplex*, *half-duplex*, or *full-duplex*. The term *echoplex* is often used to refer to a full-duplex channel which is being used primarily in one direction, with only echoes of the primary data returning from the other end (the hardware is the same as in full duplex).

A **simplex** channel is one in which information travels in just one direction. Typical applications might include: receive only printers, plotters, etc. Simplex channels are usually very easy to set up and develop software for.

A **half-duplex** channel is one in which information travels in just one direction at any given time, with the possibility of *turning the line around* (for information to travel in the opposite direction). There is usually a relatively high overhead time associated with turning a line around. This type channel is typically found in older, less sophisticated equipment. Higher speed modems often employ half-duplex in order to obtain maximum utilization of a given bandwidth.

A **full-duplex** channel is one in which information may travel in both directions simultaneously. Note that such a channel may be used to emulate either of the simpler modes. This type channel is ordinarily the most demanding on the communications hardware and software. One usually must resort to assembly language in order to provide the sort of concurrency required to support a full-duplex link.

Data Communications Hardware

There are several hardware *building blocks* used to create a communications system. Among these are computer *ports*, terminals, multiplexors, line drivers, modems, and phone system interfaces.

Almost all computers have facilities for connection to one or more serial devices. These are generally implemented as **serial ports**. Such a port may be as simple as a single UART with RS-232 conversion circuits and some mechanism to allow the CPU to read and write parallel data and status from (to) the UART. Such ports are typically implemented as DTE (Data Terminal Equipment) so that they may be connected to a DCE (Data Communication Equipment, such as a modem) with a simple (un-flipped) RS-232 cable. To connect such a port directly to another DTE (such as a terminal), a special cable is required which *flips* several signal pairs (note that the DTE and DCE ends of an RS-232 interface are symmetric).

Some computers actually have smaller processors (**front ends**) which oversee the operation of some (or all) of the ports connected to the overall computer system. These front end processors typically free up the main CPU for more complex tasks and *interrupt* it only when a mass transfer of data between the main CPU and front end is required. In a reasonable front end system, the main CPU would be interrupted only when an entire line had been received from a given terminal, and the front end would be able to accept an entire line of output at once for a given terminal. This means that the front end processor must have at least enough memory for both input and output buffers for each terminal connected through it.

There are a wide variety of currently available terminals which may be used in communications systems. These may be asynchronous or synchronous, hard copy or CRT, intelligent or *dumb*.

By now, the distinction between asynchronous and synchronous should be clear. Suffice it to say that asynchronous terminals are typically character-oriented, whereas synchronous terminals are typically message (line or page)-oriented. This refers to the amount of information that is transmitted or received in a single operation. Most synchronous terminals are more *intelligent* than asynchronous terminals. On most synchronous terminals, a line (or page) of text may be entered and edited locally, and a line protocol may be supported.

Hard copy terminals use some mechanism for physically imprinting (impact, thermal or otherwise) characters on paper. A **CRT** terminal employs a Cathode Ray Tube much like a TV picture tube to display

characters and/or graphics. An **RO** terminal is Receive Only (no keyboard). A **KSR** terminal can both send (from the keyboard) and receive. An **ASR** (Automatic Send/Receive) terminal can also send and receive from some machine-readable media (e.g., paper tape, cassette tape, or diskette).

An **intelligent terminal** usually has various *local* capabilities, such as forms generation, screen editing, a calculator mode, and/or standalone operation as a general purpose computer. The most sophisticated intelligent terminals allow local (offline) creation and editing of text files and support transfer of such files to or from remote systems, often with error detection and automatic retransmission.

At times, it is necessary to *collapse* multiple low speed lines into a single high speed line and/or expand such a line back into multiple low speed lines. This is usually done to save on phone line costs or to take advantage of a wide bandwidth channel, such as coax cable. There are two mechanisms for **multiplexing** lines in this manner: time division and frequency division.

In a **Time Division Multiplexor** (TDM), the high speed line is *time sliced*, typically at the character level. Certain recent TDM's use a technique called *statistical multiplexing* whereby more than "n" terminals running at baud rate "m" can be multiplexed onto a single line running at baud rate "nm", as long as not all of them run at full speed all the time. Note that this is a valid assumption most of the time and that time division multiplexing is an ideal application for a dedicated microprocessor system.

In a **Frequency Division Multiplexor** (FDM), the bandwidth of an analog channel is divided into multiple sub-bands, and a separate channel is maintained in each sub-band through simple frequency shifting techniques. The sub-bands are split back into separate signals at the other end using bandpass filters and frequency down-shifters. This is similar to the method whereby the phone company multiplexes many 3 kilohertz wide voice channels onto a single physical wire.

A **concentrator** is a device which allows the first *n* phone lines out of *m* possible lines ($n<=m$) to actually connect to modems, and hence, to a computer system. The "n+1" dial-in gets a busy signal. This sort of device is quite useful at any site with more users than dial-in ports (e.g., a timesharing service).

Modems are used to transduce serial digital data to/from analog audio frequency tones which may then be sent over voice-grade phone lines

without undue distortion and/or disruption of other phone company services. Bell markets an extensive line of modems which basically have set the standards for the industry as a whole. Most available modems claim to be compatible with one or more of Bell's standard models; hence, it is sufficient to describe the characteristics of the Bell units.

Several different modulation techniques are used in modems, such as FSK (Frequency Shift Keying) and PM (Phase Modulation). FSK involves shifting a base (carrier) frequency up or down a given incremental frequency to represent a logic "1" or "0" respectively. PM is typically used in conjunction with FSK to achieve *multiple bit per baud* speeds. Most 4800 and 9600 bit per second modems actually run at 1200 baud, with either 4PM (4 phase) or 8PM (8 phase) modulation superimposed. Note that PM type modems tend to be quite expensive.

As modems are useful only in groups of two or more and are often *symmetric* in nature, the terms *originate* and *answer* are used to distinguish between the two possible roles. Some full-duplex modems are *originate only*, some are *answer only*, and some can swing either way. This primarily refers to which frequency they *listen* to and which they *talk* to.

An originate modem typically transmits on the lower carrier frequency and receives on the higher one. An answer modem is the mirror image (receives on the lower and transmits on the higher). With full-duplex modems, you need one modem operating as an Originate modem and one modem operating as an Answer modem to establish a connection. The Answer modem is typically at a central computer site, and the Originate modem is typically at a remote terminal site.

With half-duplex modems (on a single phone circuit), a given modem generally runs as an Originate modem for a while, then the line is turned around and it runs as an Answer modem for a while. Note that since data will only be travelling in one direction at any given time, only one carrier frequency is used.

Examples of Currently Available Modems

The **Bell 103A** is an asynchronous, originate/answer modem which has a maximum throughput of 300 bits per second. It uses simple FSK and can be run full-duplex. Most all modems used by hobbyists are compatible with this unit, such as the Novation Cat, D.C. Hayes Micromodem 100, and Apple modem.

The **Bell 201C** is a synchronous modem which uses 4PM. It has a maximum throughput of 2400 bits per second. With a single phone circuit (2 wire), it must be run half-duplex. With a dual circuit (4 wire), it can be run full-duplex.

The **Bell 202C** is an asynchronous modem which uses FSK. It has a maximum throughput of 1200 bits per second and may only be run half-duplex. The **202D** allows full-duplex operation at speeds up to 1800 bits per second over 4 wire circuits.

The **Bell 208B** is a synchronous modem which uses 8PM. It has a maximum throughput of 4800 bits per second and only runs half-duplex.

Vadic has a unit which is not compatible with any Bell model, yet which is in widespread use (the **VA3400** series). It is an asynchronous FSK modem with a maximum throughput of 1200 bits per second and can run full-duplex on a 2 wire circuit. This is used in many commercial timesharing systems, such as TYMNET and TELENET. Note that this device appears to the computer or terminal as a 103A type device, except for baud rate.

Bell has recently announced a **212A** model which can run full-duplex 1200 baud asynchronous over a 2 wire circuit (as well as 103A style).

Connection to the Switched Telephone Network

In order to protect your equipment from possible voltage spikes and other irregularities in the phone system and to protect the phone system from your equipment, a **DAA** (Data Access Arrangement) is inserted in the line between a modem and the actual wall jack. This is simply a circuit which will not allow any kind of damaging signals to get through in either direction. The phone company used to lease these units, but no longer provides this service. DAA's are available for purchase from a number of companies (such as UDS). Most new modems being sold now include the DAA as an integral part of the modem (usually referred to as *Direct Connect*).

Some modem/DAA combinations are capable of detecting an incoming call and either answering by themselves or alerting the device they are connected to, which can then request it to go *off-hook* (**Auto-answer**). Likewise, it is possible for a modem/DAA combination to originate a call (**Auto-dial**). This is sometimes done with touch tones, but usually with carefully timed clicks (on-hook / off-hook). If the device connected to the modem/DAA has a mechanism for going off-hook in order to answer an incoming call, it can also dial using timing loops. There is a Bell device called the 801C which can be used to implement Auto-dialing.

Some modems avoid the need for a DAA by using an **acoustic coupler**. This is a speaker / microphone pair which allows the audio signal to be coupled into the phone system through the microphone / speaker in a standard handset. This arrangement makes it easier to connect into the phone system, but has the undesirable side effect of being much lower level (by a factor of 8 or so) than a direct connect system; hence, it is much more subject to error. The reason for this lower level is to prevent interference of the outgoing signal with the incoming one. A standard handset feeds some of the signal from the mouthpiece microphone back into the earpiece speaker to make the user feel more comfortable. This unfortunately disturbs the acoustic coupler, so the transmit level must be held very low to prevent interference.

A normal, randomly chosen switched line on the phone system typically has a 3 kilohertz bandwidth and widely separated *burst* noise characteristics, usually 10 msec in length, with an average error rate of one bit in 100,000. Better error rates can be obtained by going to **leased lines** with or without **conditioning** (which is very costly).

Simple Communications Systems

The first example of a communications system is a simple DTE-DCE link, such as a terminal to modem connection. In this case, you would need a straight (un-flipped) 8 line cable, as described below:

2	TD	Transmit Data	DTE --> DCE
3	RD	Receive Data	DTE <-- DCE
4	RTS	Request To Send	DTE --> DCE
5	CTS	Clear To Send	DTE <-- DCE
6	DSR	Data Set Ready	DTE <-- DCE
7	SG	Signal Ground	DTE --- DCE
8	DCD	Data Carrier Detect	DTE <-- DCE
20	DTR	Data Terminal Ready	DTE --> DCE

If fewer lines are desired in the connecting cable, the absolute minimum is lines 2,3 and 7 with *wraparounds* on both ends to convince both parties that the other is really there. Note that in normal use, the terminal brings up DTR and the modem brings up DSR when they are powered on. Many terminals require an incoming DSR (and modems an incoming DTR) before they will operate. In an asynchronous, full-duplex environment, the terminal then brings up RTS and awaits CTS from the modem (which is brought up along with DCD by the modem, once the incoming carrier has been detected). At this point, the terminal may transmit data to the modem over line 2, and the modem may transmit data to the terminal over line 3. In a half-duplex system, the RTS and CTS lines are used to determine which direction the data will be going over the phone line.

If the terminal (DTE) were to be connected to another DTE (for example, a serial printer), the following pairs would have to be flipped:

(2,3) (4,5) (6,20)

Everything else should work as before.

As a slightly more elaborate communications system, you would connect a remote terminal to a local computer via the phone system in the following manner:

The first interface is a simple RS-232 DTE to DCE connection between the terminal and the originate modem (Bell 103A), as described in the previous example.

The second interface is between the modem and the phone system. In this case, you would use an acoustic coupler, into a standard phone handset, using manual dialing.

The third interface is between the phone system and the answer modem. In this case, you would use a direct connect auto-answer modem, such as the D-CAT (also Bell 103A).

The fourth interface is another simple DCE to DTE link from the answer modem to a computer port which is configured as a DTE, as described in the previous example.

Once software is installed to handle handshaking and auto-answer, this communications system allows the remote terminal to dial in and use the system as if connected directly to it at 300 baud.

Appendix D

THE TEXT PROTOCOL

The TEXT Protocol is used to allow verified transmissions between a microcomputer running MITE and a minicomputer or mainframe. The protocol consists of two parts: MITE and the FORTRAN program TEXTP. A sample version of TEXTP, written in FORTRAN, is provided on the MITE distribution disk. TEXTP, or a program which can do the same thing, should be executing on the mainframe or minicomputer at the time you give MITE a send or receive command. TEXTP will initially be in a read state awaiting a line of input from MITE.

Verified transmission to a mainframe is complicated by the mainframe itself. Typically the mainframe does not allow enough direct control of its input/output facilities to allow the implementation of a traditional protocol. The operating system of most mainframes is only intended to communicate with a relatively simple terminal and a human typist. For a mainframe to microcomputer protocol to operate successfully, the protocol must emulate a human typist.

Most of the flexibility required to communicate with a mainframe is already incorporated into the text file upload and text file download capabilities of MITE. These capabilities, with the addition of some means of error detection and block re-transmission, form the basis for the TEXT protocol.

Sending Files from MITE to TEXTP

The general procedure for sending a file from MITE to TEXTP is as follows:

1. Read a line from the file to be transmitted.
2. Compute a 16-bit unsigned checksum of the line.
3. Get the current message number (See Table 1 below).
4. Format the line as specified in Table 1.
5. Transmit the formatted line to TEXTP.
6. Await a response from the mainframe.
7. If the mainframe rejects the line, go to step 5.
8. Increment the message number.
9. Go to step 1.

Upon reaching the end of the file to be transmitted MITE sends an E instead of the formatted data line.

TABLE 1

Items in single quotes indicate literal values.

Character Position	Value
1	'D'
2	Message number (0-9)
3-5	No. of data characters in line (000-128)
6-10	Checksum (00000-65535)
11-?	Data characters

NOTES:

1. The message number is part of the error detection logic of the protocol. It cycles in the range 0-9 and makes it easier to detect errors that occur in the acknowledgment phase of a block transmission.

2. The checksum is computed based on characters in positions 1-5 and 11-?. It is a simple 16-bit unsigned sum of the ASCII representations of the characters involved.

3. All numeric values (message number, character count, and checksum) are transmitted as printable values. They are always right-justified and zero-filled. For example, a checksum value of 2104 is transmitted as the characters 02104.

The flexibility of the protocol is based on how MITE determines when to send the data line. MITE assumes that TEXTP is ready to receive the first data line. After sending a line to the mainframe, MITE waits for a response. The response/wait procedure is as follows:

1. If CR/LF handshaking is on (option H in the TEXT FILE UPLOAD MENU), MITE waits up to 2 seconds for a line-feed.

2. MITE then waits up to 45 seconds for a response from the mainframe. A response is defined as an R followed by a character, followed by a carriage return. If the automatic line-feed option is off (option L in the OPTIONS menu), MITE waits for a line-feed.

3. If the character following the R is an A, it is regarded as acknowledgment that the line was received properly. If the character is an X, the send is aborted. Any other character is treated as a negative acknowledgment and MITE prepares to re-transmit the line. If the mainframe does not respond within the time allotted, the send is aborted.

Before transmitting the next data line (or re-transmitting the previous line), MITE waits until the mainframe is ready to receive. This procedure is as follows:

1. If a non-null turnaround character is specified (option T in the TEXT FILE UPLOAD MENU), MITE waits up to 20 seconds for the mainframe to transmit the turnaround character.

2. If a non-zero garbage character count is specified (option G in the TEXT FILE UPLOAD MENU), MITE waits up to .5 seconds for each character until receiving the number of characters specified.

   If MITE does not receive the turnaround character within 20 seconds, it assumes that the character was lost in transmission. In this case MITE continues and transmits the next line.

While transmitting a data line, if wait-for-character-echo is ON (option E in the TEXT FILE UPLOAD MENU), the next character will not be transmitted until the current character has been echoed by the mainframe. If an intercharacter delay is specified (option D in the TEXT FILE UPLOAD MENU), MITE will wait the specified number of milliseconds before sending the next character.

Sending A File from TEXTP to MITE

It is a great deal easier to receive a file from a mainframe than to transmit a file to it. To send a file from TEXTP to MITE much of the preceding discussion is relevant, with the roles reversed. The general procedure is as follows:

1. Send TEXTP an acknowledgment message (**RA** followed by a carriage return).

2. Receive a line from the mainframe (formatted as in Table 1).

3. Compute a checksum and compare it with the received checksum.

4. If the checksums match, transmit an acknowledgment (**RA** and a carriage return). Also write the data portion of the line to the output file. If the two do not match, transmit a negative acknowledgment (**RN** followed by a carriage return).

5. Go to step 2.

Before sending the **RA** or **RN** to TEXTP, MITE waits for the mainframe to become ready to receive by following the same procedure discussed in the previous section.

Appendix E

Installation Notes

(For CP/M computers only)

If you have a computer and a modem currently supported by MITE, it is quite simple to install MITE using the INSTALL program. The installation procedure is as follows:

1. Make a working copy of the master distribution disk on one of your own diskettes. Place the master distribution disk in a safe place since you may need it at a later time. *Do not install onto the master distribution disk.*

2. Mount the new diskette in drive A and run MINSTALL.

3. Next, you are presented with a list of specific auto-dial/auto-answer modems and a choice for manual-dial modems. If no modem is to be used, select the direct connect option. The direct connect option will allow computers to be connected directly to each other without modems. If direct connection is desired, a special *reverse* cable or *null modem* may be required. Consult your nearest computer store for aid in determining what equipment is needed.

4. At this point, MINSTALL will write the installed version of MITE to the file MITE.COM. If you want this file to have a different name, include an **O=fn** parameter on the command line (e.g., INSTALL O=MYMITE.COM). The default input file is MITE/U.COM. If for any reason this needs to be changed, include an **I=fn** parameter on the command line.

INDEX

Index
(Continued)

Index
(Continued)

Index
(Continued)

Index
(Continued)

Index
(Continued)

Index
(Continued)

Index
(Continued)

Index
(Continued)

Index
(Continued)

Index
(Continued)