

CROSSTALKTM



Data Communications Software System
for CP/M®

User's manual — Version 3.0

Copyright 1981, 1984



1000 Holcomb Woods Parkway
Roswell, Georgia 30076

TABLE OF CONTENTS

Chapter 1 Introduction

1.1	How to Use This Manual.....	1
1.2	Conventions used in this manual.....	1
1.3	Organization of this manual.....	2

Chapter 2 An Introduction to CROSSTALK 3.0

2.1	What CROSSTALK does.....	5
2.2	Terminal Features.....	5
2.3	File Transfer Features.....	6
2.4	Differences between CROSSTALK 2.0 and CROSSTALK 3.0.....	6

Chapter 3 Getting ready

3.1	Making a working copy of CROSSTALK.....	9
3.2	Connecting your modem.....	10
3.2.1	Modem Cable Configuration.....	10
3.2.2	Setting the modem option switches.....	11

Chapter 4 Running CROSSTALK for the first time

4.1	Invoking Crosstalk.....	13
4.2	The Command File Menu.....	13
4.3	The Status Screen.....	14
4.4	Entering commands.....	14
4.5	Getting Help.....	16
4.6	Error messages.....	16
4.7	Special key settings.....	16
4.8	Setting up your CROSSTALK program.....	18

Chapter 5 Making Your First Call With CROSSTALK

5.1	Making a call with the NEWUSER script file.....	19
5.2	Placing a call interactively.....	19
5.2.1	Setting the phone number.....	20

TABLE OF CONTENTS

5.2.2	Setting the location name.	20
5.2.3	Selecting a baud rate.	20
5.2.4	Dialing the number.	21
5.3	Manually dialing.	22
5.4	Terminal mode.	22
5.5	Entering commands while connected.	22
5.5.1	The ATTENTION key.	22
5.5.2	The SWITCH key.	23
5.5.3	When to use SWITCH and ATTENTION.	23
5.6	What to do when you're finished.	24

Chapter 6 Terminal features

6.1	Viewing the disk directory.	25
6.2	Changing the default disk drive.	26
6.3	Reviewing disk files.	26
6.4	Erasing disk files.	26
6.5	Function Keys.	27
6.6	Special Function Keys.	28
6.7	Listing the current parameters.	28
6.8	Changing the dialing strings.	28
6.9	Printing data while online — The PRINTER command.	29
6.10	The FILTER and INFILTER command.	30
6.11	The DEBUG command.	31
6.12	Terminal mode status line.	31
6.13	Changing the BREAK key assignment.	32
6.14	The TURNAROUND command.	32
6.15	The ANSWERBACK command.	33

Chapter 7 Communications parameters

7.1	The DUPLEX command.	35
7.2	Changing the communications speed.	36
7.3	Changing the data word length.	36
7.4	Changing the parity.	37
7.5	Changing the number of stop bits.	37
7.6	The LFAUTO command.	37

Chapter 8 Capturing Data

8.1	Capturing Data to Memory.	39
8.1.1	Saving Captured Data.	40
8.1.2	Precautions on using capture-to-memory.	40
8.2	Capturing Data Direct to Disk.	40
8.3	Retro-capture.	41
8.4	Capture search and status.	42

TABLE OF CONTENTS

8.5	Erasing the capture buffer.....	43
8.6	Reviewing the capture buffer.....	43

Chapter 9 Sending a File to Another Computer

9.1	The SEND command.....	45
9.2	Commands which affect the SEND command.....	45
9.2.1	Waiting for slow systems — The LWAIT & CWAIT commands.....	46
9.2.1.1	Waiting between lines — The LWAIT command.....	46
9.2.1.2	Waiting between characters — the CWAIT command.....	48
9.2.2	Expanding tab characters — the TABEX command.....	48
9.2.3	Converting lower case to upper — the UONLY command.....	49
9.2.4	Expanding blank lines — the BLANKEX command.....	49
9.2.5	Removing line feeds — the OUTFILTER command.....	49

Chapter 10 Answer mode

10.1	How answer mode works.....	51
10.2	Password protection.....	52
10.3	Access protection.....	52
10.4	Greeting message.....	53
10.5	Remote commands.....	53

Chapter 11 Protocol Transfers

11.1	CROSSTALK protocol transfers.....	55
11.1.1	Transmitting files to another CROSSTALK system.....	55
11.1.2	Requesting files from another CROSSTALK system.....	56
11.1.3	Alternate drive usage.....	57
11.1.4	Error handling during transfers.....	57
11.1.5	The BKSIZE command.....	58
11.2	XMODEM file transfers.....	59
11.3	"Hard" errors during transfers.....	59

Chapter 12 Command files

12.1	Executing a Command file.....	61
12.2	Creating command files.....	62
12.3	Changing CROSSTALK's default settings.....	63
12.4	Script files.....	64
12.5	Organization of a script file.....	65
12.6	How script files are performed.....	65
12.7	Writing a script file.....	65
12.8	Special commands for script files.....	66
12.8.1	The ABORT command.....	66

TABLE OF CONTENTS

12.8.2	The ALARM command.....	67
12.8.3	The ASK command.....	67
12.8.3.1	Single character input.....	67
12.8.3.2	Setting function keys with ASK.....	67
12.8.4	The CLEAR command.....	68
12.8.5	The DO command.....	68
12.8.6	The LABEL command.....	69
12.8.7	The IF command.....	69
12.8.8	The JUMP command.....	70
12.8.9	The SKIP command.....	71
12.8.10	The MESSAGE command.....	71
12.8.11	The RWIND command.....	72
12.8.12	The SBREAK command.....	72
12.8.13	The REPLY command.....	72
12.8.14	The WAIT command.....	72
12.8.15	The WHEN command.....	74
12.9	A typical script file.....	76
12.10	Example script files included with CROSSTALK.....	78
12.11	Differences between Command and Script files.....	78

Chapter 13 Command Summary

13.1	The ABORT command.....	79
13.2	The ACCEPT command.....	79
13.3	The ALARM command.....	80
13.4	The ANSWERBACK command.....	80
13.5	The ASK command.....	80
13.6	The ATTENTION command.....	80
13.7	The BSIZE command.....	81
13.8	The BLANKEX command.....	81
13.9	The BREAK command.....	81
13.10	The BYE command.....	82
13.11	The CAPTURE command.....	82
13.12	The CLEAR command.....	83
13.13	The COMMAND command.....	83
13.14	The CSTATUS command.....	83
13.15	The CWAIT command.....	84
13.16	The DATA command.....	84
13.17	The DEBUG command.....	84
13.18	The DIR Command.....	85
13.19	The DO command.....	85
13.20	The DPREFIX command.....	86
13.21	The DRIVE command.....	86
13.22	The DSUFFIX command.....	86
13.23	The DUPLEX command.....	87
13.24	The ERASE command.....	87
13.25	The FILTER Command.....	87

TABLE OF CONTENTS

13.26	The FKEYS command.	87
13.27	The GO command.	88
13.28	The HELP command.	88
13.29	The IF command.	88
13.30	The INFILTER command.	89
13.31	The JUMP command.	89
13.32	The KEY command.	89
13.33	The LABEL command.	89
13.34	The LFAUTO command.	90
13.35	The LIST command.	90
13.36	The LOAD command.	90
13.37	The LWAIT command.	90
13.38	The MAGIC command.	91
13.39	The MESSAGE command.	91
13.40	The MODE command.	91
13.41	The NAME command.	92
13.42	The NO command.	92
13.43	The NUMBER command.	92
13.44	The OUTFILTER command.	92
13.45	The PARITY command.	92
13.46	The PRINTER command.	93
13.47	The PWORD command.	93
13.48	The QUIT command.	93
13.49	The RCVE command.	93
13.50	The RDIALS command.	93
13.51	The REPLY command.	94
13.52	The RQUEST command.	94
13.53	The RWIND command.	94
13.54	The RXMODEM command.	94
13.55	The SAVE command.	95
13.56	The SBREAK command.	95
13.57	The SCREEN command.	95
13.58	The SEND command.	95
13.59	The SKIP command.	96
13.60	The SPEED command.	96
13.61	The STOP command.	96
13.62	The SWITCH command.	96
13.63	The TABEX command.	97
13.64	The TSTATUS command.	97
13.65	The TURNAROUND command.	97
13.66	The TYPE command.	98
13.67	The UONLY command.	98
13.68	The USER command.	98
13.69	The VERSION command.	98
13.70	The WAIT command.	99
13.71	The WHEN command.	99
13.72	The WRITE command.	99

TABLE OF CONTENTS

13.73	The XCPM command.....	99
13.74	The XMIT command.....	100
13.75	The XXMODEM command.....	100
<hr/>		
Chapter 14 Examples		
14.1	Calling The Source.....	101
14.2	How to capture data.....	103
14.3	How to send a file with the SEND command.....	103
14.4	Transferring files to another CROSSTALK system.....	104
<hr/>		
Appendix A	105
Appendix B	107
Appendix C	109
Appendix D	111
Appendix E	115
Appendix F	117
Appendix G	119
Appendix H	121
<hr/>		
Index	127

COPYRIGHT NOTICE :

Your copy of CROSSTALK is licensed for use on a single computer system only. Please sign and return the enclosed user agreement if you have not already done so. Returning the agreement registers you as a legal user of the CROSSTALK program, and entitles you to updates and support. You must provide your CROSSTALK serial number when requesting support from Microstuf. The CROSSTALK program and this manual are copyrighted material. All material in this manual is proprietary to Microstuf, Inc., and no part of this manual or the CROSSTALK program may be reproduced, transmitted, transcribed, stored in any information retrieval system, or translated into any foreign language or computer language, in any form, by any means, whether mechanical, electronic, magnetic, optical, chemical, manual, or otherwise, without the express written permission of Microstuf, Inc.

DISCLAIMER :

Microstuf, Inc. makes no representation or warranties with respect to the contents hereof, and specifically disclaims any warranties, implied or otherwise, of merchantability or suitability for any particular purpose. Microstuf reserves the right to modify the CROSSTALK program and associated manual at any time without obligation on behalf of Microstuf, Inc. to notify any persons or party of such modifications.

USER AGREEMENT :

Please read the enclosed user agreement, and sign, date and return the user agreement acknowledgement card before you begin using this software. Returning the user agreement card entitles you to software support, and registers you as a CROSSTALK user.

RETURN FOR UPDATE/WARRANTY PROCEDURE

When returning a diskette for updating or repair, you must always send us your original distribution diskette. You must also have previously registered your software with us. Follow the procedure listed below when returning a Microstuf product.

1. Contact the Microstuf, Inc. technical support personnel for a return authorization number (RA). Note that this RA is only valid for 30 days.
2. Pack the diskette (or entire package at the discretion of our representative) in a protective envelope or other suitable packaging. Please do not use highly static material as they may harm the diskette. It is important to mark the package as fragile (magnetic media) so that it will not be folded, bent or X-RAYed.
3. When returning a product always include the following information:

NAME

ADDRESS

CITY, STATE, ZIP

TELEPHONE NUMBER

RETURN AUTHORIZATION NUMBER

PROBLEM DESCRIPTION (a short description is adequate.)

4. All products returned to Microstuf should be shipped UPS or USPS prepaid. It is recommended that the diskette be insured when shipped. **MICROSTUF WILL NOT ACCEPT MERCHANDISE THAT HAS BEEN SENT C.O.D.**

Mail package to:

Microstuf, Inc.

1000 Holcombs Woods Parkway

Suite 440

Roswell, GA 30076

RA Number (Please include RA number on mailing label.)

1

Introduction

Congratulations on your purchase of CROSSTALK 3.0. You are now the owner of the most sophisticated, powerful, and flexible communications program available for 8-bit computers.

In order to get the most out of your purchase, please take the time to read this manual and go through the example sessions shown in the tutorials. Most of the calls we receive in our customer service department are simple questions, which could have saved the caller a few dollars in long-distance charges if he had read the manual. If there is something you do not understand, or if you have a problem, it is probably covered in this manual.

CROSSTALK 3.0 represents the end result of thousands of hours worth of design and programming work. Don't let the size of this manual scare you; the program is designed to allow you to use only those features you need, and ignore the others. As your needs become more demanding, you'll find that the program will be able to meet those needs.

1.1

How to Use This Manual

This manual has been designed to accommodate both the experienced user and the computer neophyte. "Techno-jargon" has been avoided wherever possible. Data communications is, by its nature, a technical area. CROSSTALK, and this manual, have been designed to allow you to deal with the various technical aspects of communications as easily and as clearly as possible.

We suggest that you start right here at Chapter 1, and progress forward, regardless of your level of computer proficiency. If the material seems familiar, skip to the next section. The worst that can happen is that you'll have to come back and re-read the sections you skipped.

A word of warning to old (pre - 3.0) CROSSTALK users: This version is COMPLETELY different from previous versions. There are many new commands, many old commands have new names, and some commands have different functions depending on the circumstances under which they are used. Section 2.4 and Appendix B contain information outlining the differences between CROSSTALK 3.0 and previous versions of CROSSTALK.

1.2

Conventions used in this manual

There are numerous examples of how to enter CROSSTALK commands throughout this manual. Whenever user input is shown, it will be displayed in **bold-faced** type.

When entering commands to CROSSTALK, you will need to end your input with a carriage return so that CROSSTALK will know that you are done typing at the command line. This will be indicated by "**(CR)**", which stands for a carriage return. This key is usually marked RETURN or ENTER on most keyboards.

Part of the ASCII character set is made up of what is known as "control characters." These are the non-printing characters which generally have a special meaning to terminals and other peripherals. In this manual, these characters are shown in the form **^x**, where "x" will be the upper case character corresponding to the actual control character. For instance, **^A** represents control-A.

1.3

Organization of this manual

This manual is organized into fourteen chapters, and seven appendices.

The first two chapters are an introduction to CROSSTALK, and to data communications in general.

The third chapter provides preparatory information. This chapter will show you how to make a working CROSSTALK disk, how to run install, how to connect your modem to your computer, and how to test your system to be sure that everything is working properly.

The fourth and fifth chapters are a tutorial on CROSSTALK. This section starts out with the most elementary aspects of the program, and progresses through to some of the more advanced features.

Chapters six and seven detail CROSSTALK's operation as a terminal program. Chapter six explains how to modify CROSSTALK's operation in terminal mode and chapter seven deals with changing the various hardware-related parameters.

Chapters eight and nine deal with uploading and downloading text files to non-CROSSTALK systems. Chapter eight describes how to use CROSSTALK to capture data from another computer system, while chapter nine explains how to send text files to another computer.

Chapter ten explains CROSSTALK's operation in answer mode.

Chapter eleven explains how to exchange files with another CROSSTALK system.

Chapter twelve shows how CROSSTALK uses "command files" to store and re-call phone numbers and other information for frequently-called systems. This chapter also shows examples of typical command and script files.

Chapter thirteen contains a summary of all of CROSSTALK's 75 commands. Each command is listed in alphabetical order, along with a brief description of each command.

Chapter fourteen shows examples of typical uses for CROSSTALK, including capturing data,

sending a text file, and exchanging program files with another CROSSTALK system.

Appendix A is a glossary of data communications and computer terms.

Appendix B is a guide for users who have used an older release of CROSSTALK. This section outlines the major differences between CROSSTALK 3.0 and older releases of the program.

Appendix C is a "how-to" section, detailing how to connect two computer systems together and transfer files.

Appendix D shows several typical cable diagrams for connecting your computer to a modem or to another computer system.

Appendix E contains option switch settings for several popular modems, and Appendix F contains typical dialing commands for several modems.

Appendix G is an ASCII code chart. This chart shows the hex value and mnemonic name of each of the "special" keys on the keyboard.

Appendix H is provided for programmers who may wish to further customize CROSSTALK for their system.

2

An Introduction to CROSSTALK 3.0

CROSSTALK is a complete, self-contained data communications program. It is available for most popular 8 and 16-bit computer systems, including most CP/M, MP/M, CP/M-86, MS-DOS, and PC-DOS computer systems.

CROSSTALK 3.0 is a complete rewrite of the 8 bit version of CROSSTALK. It was designed to fully utilize the computing power of the new 8 bit microcomputers. **Please note that because of the vast number of 8 bit microcomputers on the market and the diversity in their design, all features of CROSSTALK do not work the same on all systems. This manual makes highlighted notes of these features in each section.**

CROSSTALK 3.0 requires at least 48K of RAM and 1 disk drive.

2.1

What CROSSTALK does

CROSSTALK performs two major functions. It is a "terminal program", allowing you to dial into a host computer system and act as a terminal to that system. Second, it is a "file transfer" program. This allows you to call up other CROSSTALK-compatible systems, and exchange files with that system. The file transfers are performed with an extremely accurate error check, assuring you that the file you sent is exactly identical to the file received at the other end of the transfer, even over noisy telephone lines.

2.2

Terminal Features

Earlier, we mentioned that CROSSTALK was a "terminal program". This means that your computer system running CROSSTALK can operate as a terminal to many other computer systems. A good example of this would be using your CROSSTALK system to call into a timesharing "mainframe" computer system. As far as the mainframe can tell, you are just another terminal.

Most terminals are just that; they are terminals. They display incoming data on a video screen, and they send characters typed on the keyboard to the host computer. Terminals do not generally offer much in the way of storage or retrieval of data. CROSSTALK, on the other hand, allows you to perform several functions not normally found in terminals.

First, CROSSTALK can "capture" incoming information from the host computer, and save the stored information onto a disk. This allows you to "download" text and program files from other computer systems to your CROSSTALK system. Since you can store the captured information on a disk, you are free to edit and manipulate the information in any way you wish.

Conversely, CROSSTALK can send files from your disk to a host computer system. This feature allows you to prepare text files off-line, using your favorite word processor or text editor program, and then call up another computer system and transmit the file to the host system at full speed.

2.3

File Transfer Features

When you use CROSSTALK as a file transfer program, you can call up any other CROSSTALK or compatible system and exchange any type of file with the other system. The other system doesn't have to be the same type of computer, and in fact, does not even have to be running the same operating system. CROSSTALK's file transfers are "transparent"; the user does not have to concern himself with any of the hardware incompatibilities between the two systems in question. As far as the program is concerned, one CROSSTALK system is the same as any other, regardless of the type of hardware involved.

Any type of file may be transferred to another CROSSTALK system, including 8-bit .COM files. Unlike many other programs, CROSSTALK does not distinguish between 7-bit and 8-bit files. Eight bit transfers do not take any longer than 7 bit transfers, and the user does not have to "convert" 8-bit files to 7 bit files before transferring them.

The only limit on the size of file that can be transferred with CROSSTALK is imposed by the capacity of your computer's disk drives.

Files may be transferred in logical groups. If you want to send all of the files on a disk, a single command tells CROSSTALK to send all of the files. Names of files to be transferred may also be placed in a command or script file, allowing a user to transfer dozens of files with one command.

2.4

Differences between CROSSTALK 2.0 and CROSSTALK 3.0

CROSSTALK 3.0 is a complete rewrite of the previous release of CROSSTALK. It is very similar to our 16 bit version of the program, CROSSTALK-XVI. The end result is a program which has many features not found in the original version.

During the design stages of development, we placed one thing above all others: the overall USABILITY of the program. The program is not menu-driven, nor is it entirely command driven. Yet it has the advantages of each. We have found that menu-driven communications programs tend to bog the user down in a maze of menus and prompts. Purely command driven programs, on the other hand, tend to assume that the user knows exactly what he is doing, and generally provide little in the way of guidance for new users.

We took a slightly different approach when designing CROSSTALK 3.0. The status screen is always available by typing a single keystroke. The status screen is both a display of all of CROSSTALK's parameters and settings, and a menu of the commands related to those settings.

Most commands are self-prompting, and an extensive built-in help system is available to guide you along. If you enter an incomplete command, the program will prompt you to enter the rest of the command, and in many cases, will offer the most likely choices. If you don't understand a question the program is asking, you may press the "?" key, and the program will offer a brief explanation of the question.

Some command names have been changed in this release of CROSSTALK. For example, the old READ command has been re-named to the SEND command. The commands with new names are usually equivalent to their older counterparts.

Appendix B contains a complete list of commands which are either new or different to this release of CROSSTALK.

Before you run CROSSTALK on your system, you will need to do a little preparatory work. You'll need to copy the CROSSTALK program and help files onto a disk, configure CROSSTALK for your machine, connect the modem to your computer, and set the option switches on your modem.

This chapter will guide you through all of these steps.

3.1

Making a working copy of CROSSTALK

The first thing you'll need to do is to make a copy of CROSSTALK for your day-to-day use. We suggest that you create a "work" disk specifically for CROSSTALK. Use this "work" disk every time you run the program.

The reason we suggest that you use a separate disk for CROSSTALK is this: The CROSSTALK program and accompanying help files take up about 72K of disk space. Additionally, you will probably want to keep your CROSSTALK command and script files (more about these later) on the same disk, so that they will always be available to you without having to search through a box of diskettes.

To make a CROSSTALK work disk, you will need to "format" a new diskette, put a "system" on it, and then copy all of the CROSSTALK files to the new disk. Follow your computer manufacturer's instructions for formatting and placing a system on a diskette, and then complete the following steps:

1. Once the blank diskette had been formatted and a system is on the diskette you then need to put PIP.COM on the diskette. Assuming that your CP/M work disk is in drive A, and the new diskette in B, from the "A>" prompt type:

PIP B:=A:PIP.COM<CR>

2. Now, place your newly formatted diskette in drive A, the original CROSSTALK diskette in drive B, and type ^C. Then, from the CP/M "A>" prompt, enter the command:

PIP A:=B:*.*<CR>

This will copy all of the files from drive B to drive A. After the copying is complete, remove the original CROSSTALK diskette and file it away in a safe place. Remember that if you should ever have to return your CROSSTALK diskette to us for updating, we require that you return the ORIGINAL diskette.

3. At this point, we need to configure CROSSTALK for your machine. This is done by running the program INSTALL.COM, which should be on your new CROSSTALK work disk. If this program is not on your diskette, CROSSTALK is already configured and ready to run; you can skip to #4. Otherwise, at the CP/M "A)" prompt, type:

INSTALL(CR)

Follow the instructions in INSTALL to create a working copy of CROSSTALK for your system.

4. You now have a working copy of CROSSTALK. We suggest that you make a back-up copy of your working diskette from time to time, since you'll probably save some CROSSTALK command and script files on it later, and you'll probably want to have them backed up.

3.2

Connecting your modem

Normally, this should be the easiest part of preparing your system. Unfortunately, it is one of the largest sources of calls to our customer service department.

Unlike many other communications programs (and the earlier releases of CROSSTALK), CROSSTALK 3.0 makes full use of all of the "handshaking" signals provided by the modem and computer. In order for the program to operate properly, these signals must be properly connected.

There are different types of modems available for microcomputers. The most commonly used type is often referred to as a "stand-alone" modem. This type of modem is connected to the computer through a cable to a "serial communications port" on the computer.

Another type of modem is a "plug-in" modem, generally found in S-100 bus machines. If you are using a "plug-in" modem, follow the manufacturers installation procedure, and skip to chapter 4.

3.2.1

Modem Cable Configuration

No matter which modem you use, you will have to follow the manufacturer's installation instructions for the modem. Many modems require that certain option switches be set; this is covered in the next section.

If you are using a stand-alone modem, you will need to acquire a cable to connect the modem to the computer's serial port. In order for CROSSTALK to operate properly, the following signals MUST be connected:

DTE Pin No.	Signal Name
1	Protective Ground
2	Transmit Data
3	Receive Data
7	Signal Ground
8	Carrier Detect
20	Data Terminal Ready

Some microcomputers will need other pins connected as well as the six shown above. Please see Appendix D for additional information on connecting your particular computer to modems and to other computer systems.

Additionally, there may be a CABLE.DOC file on your diskette. If so, locate this file and read it using the CP/M TYPE command (e.g.- "TYPE CABLE.DOC"). It will contain cabling diagrams and other information specific to your computer which may not appear in Appendix D. If you are still unsure about cabling for your machine, contact your local dealer or call our technical service department for further assistance.

3.2.2

Setting the modem option switches

As mentioned, CROSSTALK makes full use of the "handshaking" signals provided by the modem. Some modems, particularly the newer "auto-dial" modems, have option switches which allow the user to over-ride some of these signals.

Consult the table in appendix E for the proper switch settings for your particular modem. If your modem is not listed in the table, some guidelines for setting the switches follow:

CROSSTALK controls the DTR line. If your modem has a feature which allows DTR override, disable this feature.

Similarly, CROSSTALK expects to see the carrier detect signal only when a carrier is actually present. If your modem has a feature which "forces" the carrier detect signal to always be true, disable this feature.

Finally, many modems have a switch which allows you to disable auto-answer. Since CROSSTALK controls the DTR line, the program itself will insure that your modem will not answer the line unless CROSSTALK is in "answer" mode, awaiting a call. Set the switch to leave auto-answer enabled.

4Running CROSSTALK for the first time

This chapter will guide you through running CROSSTALK, and will show you the various ways to enter commands to CROSSTALK. At the end of the chapter, you will be able to configure CROSSTALK's default settings.

4.1

Invoking Crosstalk

Now that you have made a working copy of CROSSTALK and have connected your modem, you are ready to run the program for the first time. With your working copy of CROSSTALK in drive A, enter the command:

XTALK(CR)

After a few seconds, the screen will clear, and CROSSTALK will sign-on. The sign-on message will remain on the screen for about five seconds. To skip the sign-on message, press any key.

4.2

The Command File Menu

At this point, CROSSTALK will load the standard command file, STD.XTK, which contains the default settings. CROSSTALK will then clear the screen and give you a menu of the available command files, and ask you to enter the number of the file you wish to load. The screen will look something like this:

Available Command Files:

- 1) NEWUSER 2) SETUP 3) STD**

Enter the number of the file to use (1-3):

You may enter the number of the file to load or press RETURN to go to the status screen. For now, hit the RETURN key so that we can take a look at the STATUS screen.

4.3

The Status Screen

CROSSTALK's status screen will look much like this:

CROSSTALK — 3.0 Status Screen						Offline
Name Number			LOaded	None loaded		
Communication Parameters			CApture	Off, 20K free		
SPeed	1200	DAta	8	DEbug	Off	LFauto
STop	1	PArity	None	TABex	Off	BLankex
MOde	Call	DUplex	Full	INfilter	On	OUTfilter
Key Settings						Filter Settings
ATten	^A	KEY	^L	COmmand	^C	CWait
SWitch	ESC			BReak	^B	LWait
Miscellaneous Parameters						Send control settings
DRive	A:, 120K free	ACcept	Everything		PWord	
DPrefix	ATDT	DSuffix	I		TURNarnd	^M
PRinter	Off	UConly	Off		ANSwback	On
TStatus	Off	BKsize	1		RDials	10

Command?

The status screen serves two purposes. First, it is a display of all of CROSSTALK's major options, and each option's current setting. Second, it is a "menu" of commands. Notice that the first two letters of each command are highlighted; that is, they are displayed in "bright" letters (if, of course, your terminal supports this feature). There is a reason for this: when entering commands to CROSSTALK, the program only requires you to enter the first two letters of the command. You may enter the entire command name if you wish, but it isn't necessary.

4.4

Entering commands

Any command may be entered any time the "Command?" prompt appears on the screen by entering the two letter command name and pressing the RETURN key. If the command requires additional information, the program will ask you for the information by printing a question in the command line.

To see how CROSSTALK processes commands, let's enter a command. Enter "**NU**", and press the RETURN key. The following message will appear in the command line:

Enter number: _

Enter the number "**1-404-998-8048**", and press the RETURN key. Now look up at the status screen again. Notice that the word "NUmber" in the upper left now shows the phone number you just entered next to it.

Let's try that again, but this time, we'll ask CROSSTALK to help us. Enter "**NU**", and press RETURN. When the program asks you to enter a phone number, press the "?" key, and RETURN again. The program will display a "help" message describing the NUmber command.

Any time you need help with a command, you may press the "?" key for help with that command. The program will tell you about the command in question, and then repeat the question it was asking. Notice that the command line still says "Enter number to dial". For now, press RETURN. The program will clear the number at the top left of the screen.

Once you become familiar with the program, you will probably want to use the "answer ahead" method of entering commands. To see how this works, enter the command:

Command? **NU 1-404-998-8033(CR)**

Notice that the program did not ask you to enter a phone number, since you provided a phone number with the command. Also, notice that the phone number now appears next to the "NUmber" at the top left of the screen again.

Commands which set an ON/OFF state (such as the PRINTER command, which turns the printer on and off) will accept answers in several different ways. As an example, the commands:

Command? **PRINTER ON, PRINTER +, and PRINTER 1**

are all equivalent. Similarly, the commands:

Command? **PRINTER OFF, PRINTER -, and PRINTER 0**

are also equivalent. Use the form you find most convenient.

These "ON/OFF" commands have an additional option; they can be "toggled" to their opposite state by entering the command, followed by a slash (/) character. For example, if the printer was ON, then the command:

Command? **PRINTER /(CR)**

would turn it OFF. Conversely, if the printer was OFF, the same command would turn it ON.

4.5

Getting Help

Now that you know how to enter commands to CROSSTALK, one of the commands you may find particularly useful is the HELP command. As mentioned above, you can type "?" whenever you need help with one of CROSSTALK's commands. Additionally, you can get help on any of CROSSTALK's commands by simply asking for it with the HELP command. For instance, to get help on the NUMBER command, type:

Command? **HELP NUMBER(CR)**

CROSSTALK will display help for the NUMBER command, and return you to the screen you were previously at. Note that typing HELP alone at the "Command?" prompt will display a list of all of CROSSTALK's commands.

4.6

Error messages

In addition to asking questions and accepting commands, the command line is also used to display error messages. Enter the command:

Command? **ID(CR)**

The program will respond with:

Unrecognized command "ID". Press ENTER: _

Most error messages require that you press the RETURN key to acknowledge the error. If you make a mistake entering a command, just enter the command again. If you don't understand the command, press "?" when it asks you, and let the program tell you what to do.

4.7

Special key settings

CROSSTALK assigns a special meaning to several of the keys on the keyboard. These special keys are used in a number of ways. The table below lists each of the special keys, its default setting, and tells what each key does.

Key name	Default setting	Purpose
ATTENTION	[^] A	The ATTENTION key is used to get CROSSTALK's "attention" when you are connected to another computer. Normally, when you are online with another computer, all keys typed on your keyboard are transmitted to the other computer. When you press the ATTENTION key, CROSSTALK will respond by showing the "Command?" prompt on the bottom line of the screen. You may then enter commands to CROSSTALK.
SWITCH	ESC	The SWITCH key is used to switch between the status and terminal screens. Each time the SWITCH key is pressed, CROSSTALK will switch between the two screens.
BREAK	[^] B	The BREAK key causes CROSSTALK to transmit a special signal called a "break" signal to the other computer system. A break signal is used by many dial-up computer systems to interrupt printing. Do not confuse this key with the key marked "Break" on some keyboards.
COMMAND	[^] C	The COMMAND key is similar to the ATTENTION key, except that the COMMAND key is used to get the "attention" of an answering CROSSTALK system. An answering CROSSTALK system responds to the COMMAND key by sending the "Command?" prompt. We recommend that you keep this key set to [^] C for compatibility with other CROSSTALK users.
KEY	[^] L	This KEY command is used to emit the contents of any of the 10 programmable function keys. By hitting [^] L1 the contents of function key 1 will be sent to the host system.

You may re-assign the SWITCH, ATTENTION, BREAK, KEY and COMMAND keys to any "non-printing" key on the keyboard. By "non-printing", we mean those keys which do not display a character when they are pressed. The special key settings may be changed by entering the name of the key, and pressing RETURN. CROSSTALK will ask you to press the key you wish to use for that special key. Additionally, key settings may be changed by entering the command followed by the hex value of the new key, or by typing the new key (e.g.-AT C<CR> for [^]L as attention).

Note that with the exception of the COMmand key, no two "special" keys may have the same value.

4.8

Setting up your CROSSTALK program

Before you actually begin using CROSSTALK, you will probably want to change the way CROSSTALK is set up. As shipped from us, CROSSTALK "wakes up" with the following "default" settings:

Mode: CALL
SPeed: 300 baud
Modem type: Hayes Smartmodem

To change any of these settings, you may wish to use a special "script" file we have provided. A script file is actually a program written in CROSSTALK's own "language". Several script files are provided on your CROSSTALK disk. We will explain script files fully in chapter 12. For now, enter the command:

Command? **LO SETUP(CR)**

and CROSSTALK will load and "run" the SETUP script file. The SETUP file will ask you a series of questions about how you want to set up your CROSSTALK program. After you have answered all of the questions, your new default settings will be permanently saved on your disk in a file called STD.XTK. This file is then automatically loaded each time you bring CROSSTALK up.

Even if you do not wish to change your default settings, you may wish to run SETUP to see how script files work.

5

Making Your First Call With CROSSTALK

By now, you're probably tired of reading, and are ready to actually DO something. This section will show you two different ways to make a call with CROSSTALK.

In chapter four, you used a "script file" to help you set up CROSSTALK's default settings. In the first part of this chapter, we will use another script file to make a call. In the second part of this chapter, you will learn how to enter commands to CROSSTALK to make a phone call. After we make the call, we'll show how to save a command file, so that you can make a call to the same location later, without having to re-enter all of the information.

This section will assume that you are using a Hayes Smartmodem or other Hayes-compatible auto-dial modem. If you are using another type of auto-dial modem, you will need to set the dialing control strings with the DP and DS commands, or run the SETUP script file as shown in chapter 4. The DP and DS commands are described in Chapter 6.

If you are using a modem without an auto-dialer (such as an acoustic coupler), read this example, and see section 5.3.

5.1

Making a call with the NEWUSER script file

The easiest way to make your first call with CROSSTALK is to use the NEWUSER script file which was provided with your copy of CROSSTALK.

To load the NEWUSER script file, enter the command:

Command? **LO NEWUSER(CR)**

from the "command" prompt. The NEWUSER file will ask you a series of questions. After answering the questions, CROSSTALK will offer to save the settings you have entered for future use. If you will be calling the same computer often, you will probably want to save the settings.

After you run NEWUSER, you will be ready to make a call. The NEWUSER script file will tell you exactly what to do to begin your call. After you have established connection with the computer you are calling, turn to section 5.4, "Terminal mode".

5.2

Placing a call interactively

If you do not wish to use the NEWUSER script file, you may make a call by following the example presented in the remainder of this chapter. The steps outlined here are essentially the same as those performed by the NEWUSER script file.

5.2.1

Setting the phone number

Before you can make a call with CROSSTALK, the program has to know several things about the location you are calling. First, it needs to know the phone number, so that the program can dial the number for you.

To set the phone number, enter the command:

Command? **NU(CR)**

The program will ask you to enter the phone number to be dialed. For purposes of example, we will use the phone number of the Microstuf test center in Atlanta. In fact, you can call the test center if you wish, and check out your system. The number is (404) 998-8033. If you wish to call another number, enter the number to be dialed. Note that CROSSTALK does not care if you enter any punctuation or spaces. However, CROSSTALK sends exactly what you enter in the number command to the modem. If the modem cannot handle spaces and punctuation, do not enter any.

After entering the number, check the status screen display to ensure that you have entered the number correctly.

5.2.2

Setting the location name

Now that you have set the phone number, the computer knows what number to call. You may wish to set the NAme of the location, so that you'll know where the computer is calling. Enter the command:

Command? **NA(CR)**

The program will ask you to enter the name of the location you are calling. Enter "**Microstuf Test Center**", and press the RETURN key. Again, look at the status screen, and you'll see that the name is now shown on the screen.

5.2.3

Selecting a baud rate

When you first run CROSSTALK, the program is set for 300 baud operation. If you have a 1200 baud modem, and are calling our test center, you will probably want to call at 1200 baud.

To change the baud rate to 1200, enter the command:

Command? **SPeed 1200(CR)**

If you have a 300 baud modem, the program will already be set for 300 baud. Note that when entering baud rates, 1200 baud is entered as "1200", but 110 baud is entered as "0110".

5.2.4

Dialing the number

At this point, you have entered the two essential items necessary to place a call: a phone number and a baud rate. Even though you have entered this information, CROSSTALK will not dial the phone until you tell it to.

The GO command is used to tell CROSSTALK to begin dialing. There are several forms of the GO command. The simplest case is to enter GO and press RETURN. You may also begin the dialing process by pressing RETURN at the command prompt. The program will dial the number once, and attempt to connect to the other computer system. If the number is busy or doesn't answer, CROSSTALK will offer to re-dial the number for you.

The second form of the GO command allows you to specify in advance that you want CROSSTALK to re-dial until either the call gets through or the number specified in the RDial command is reached. For example, the command:

Command? **GO R90/30(CR)**

tells CROSSTALK to dial the number repetitively at 90 second intervals and to wait 30 seconds for a connection each time it dials the number. The "R" instructs CROSSTALK to sound an alarm tone after establishing the connection. If you do not wish to hear the alarm tone, use "Q" in place of "R".

You may change the maximum number of redials with the RDIAL command. The RDial default is 10, but can be anywhere from 0 to 65535. To change the maximum number of redials to 100, enter the command:

Command? **RDial 100(CR)**

The third form of the GO command is for use in non-standard applications. When issued, it tells CROSSTALK to ignore the state of the carrier detect signal. This is useful when using a modem which does not have an auto-dialer, using a modem which does not support the carrier detect signal, or when hardwiring two computers. To enter the terminal mode in this state, issue the command:

Command? **GO LOCAL(CR)**

Note that if you are using GO LOCAL when online to another computer via modems and for some reason the modems disconnect, CROSSTALK will not recognize that you are no longer online. Hence it is better to use the normal form of the GO command if your modem supports the carrier detect signal.

5.3

Manually dialing

Many modems, such as acoustic couplers, are not capable of auto-dialing. In this case, use the following procedure:

1. Set the baud rate using CROSSTALK's SPEED command.
2. Put CROSSTALK into "LOCAL" mode using the command **GO LOCAL**.
3. Dial the host computer using a standard telephone.
4. When the modems connect (there is usually an LED which shows the modem has connected), gently replace the handset on the telephone, and begin your session using CROSSTALK.

5.4

Terminal mode

Once CROSSTALK dials and connects to a distant computer system, your screen will clear and the program enters "terminal mode". At this point, your system looks like a terminal to the remote computer system. Any keys you type on the keyboard are sent to the other computer, and any characters received from the other computer are displayed on your screen.

What happens at this point depends largely on the computer you called. Some systems (such as our bulletin board at (404) 998-8048) require that you press RETURN a few times when you first connect, other systems immediately display a greeting message.

5.5

Entering commands while connected

Once you have connected to another computer, CROSSTALK will remain in terminal mode until you press either the ATTENTION or SWITCH key. Each of these keys has a special meaning to CROSSTALK.

5.5.1

The ATTENTION key

Pressing the ATTENTION key when CROSSTALK is connected to another computer will display the "COMMAND? " prompt on the bottom line of the screen. When the command

prompt appears, you may enter any command to CROSSTALK. While the command prompt is on the screen, CROSSTALK does not display incoming data, but will save the incoming characters (up to 2 screen fulls) and display them after you have entered the command.

[Note: Many 8 bit systems do not have interrupt driven I/O, and commands to CROSSTALK which display information (e.g.-DIR or TYPE) can cause loss of data from the modem. It is best to issue these commands when the host system has stopped transmitting data.]

As an example, let's say that you have called into another computer system, and connected. Once you call into the system, you decide that you want your printer on, so that you'll have a printed copy of your session. Press the ATTENTION key (usually the ^A key), and the "Command?" prompt will be displayed on the bottom of the screen. Enter the command "**PRINTER ON**", and press RETURN. CROSSTALK turns the printer on, and returns to terminal mode.

If you issue a command from terminal mode by pressing the ATTENTION key, CROSSTALK will return to terminal mode after completion of the command. If the command resulted in an error message, the error message will be displayed and you will return to terminal mode.

5.5.2

The SWITCH key

The SWITCH key operates in a manner similar to the ATTENTION key. When you press the SWITCH key, CROSSTALK switches between the terminal communications screen and the status screen. Any time the status screen is displayed, the "Command?" prompt will appear on the bottom line of the screen.

Please note that some CP/M machines cannot switch from the terminal screen to status screen and back without loss of characters due to hardware and/or system software limitations.

When the command prompt appears, you may enter any command to CROSSTALK. After the command is completed, the command prompt will re-appear. To return to terminal mode, press the RETURN key. The program will then switch back to the communications screen.

5.5.3

When to use SWITCH and ATTENTION

Generally, you will only use the SWITCH and ATTENTION keys when you are connected to another computer system. When you first run CROSSTALK, the status screen is displayed.

As soon as connection with another computer is established, the program automatically switches to the terminal screen.

As explained earlier, the SWITCH and ATTENTION keys are used to enter commands to CROSSTALK while you are connected to another computer.

When online, it is best to use the ATTENTION key whenever possible to avoid loss of data. However, if you have many commands to enter, or just feel more comfortable seeing the commands change, you will want to use the SWITCH key.

Note also that there are some commands that display information on the screen, such as the DIRECTORY command, which displays a disk directory. If this command is issued at the status screen, the screen will clear before the directory is displayed. CROSSTALK will pause after displaying the information, waiting for you to press the return key. CROSSTALK will then put the status screen back up when you are ready. If this command is issued from the terminal screen, the directory will be displayed, and communications will automatically continue. If characters are coming in from the host, there is a good chance that the directory would scroll off the screen before you had finished reading it.

5.6

What to do when you're finished

After you have completed a call, you will probably want to hang up. There are several commands relating to hanging up and exiting CROSSTALK. The table below shows these commands, and explains what each one does.

COMMAND	EFFECT
QUIT	Disconnects the current call, and returns you to CP/M. Use this command when you are finished with your call, and do not wish to make another call.
BYE	Disconnects the current call, but leaves you in CROSSTALK. Use this command when you are finished with your call, but wish to make another call.
XCPM	Leaves CROSSTALK without disconnecting the call in progress. Use this command when you wish to exit from CROSSTALK, run another program, and return to CROSSTALK without disconnecting.

To enter any of these commands while a call is in progress, press the ATTENTION or SWITCH key, enter the command, and press RETURN. Exercise caution when using the XCPM command — it is easy to forget that you are still connected.

In the last section, we explained the elementary aspects of using CROSSTALK: how to enter a phone number, how to enter commands, and how to use the SWITCH and ATTENTION keys. In this section, we will explain some of the more advanced features of CROSSTALK.

The features described in the beginning of this chapter are provided for the convenience of the user; they have no effect on the "hardware" aspects of communications. The commands described in the latter part of this chapter affect the way in which CROSSTALK operates in terminal mode.

6.1

Viewing the disk directory

CROSSTALK allows you to view the directory of any disk in the system in much the same manner as CP/M does. The DIR command is used to view disk directories. The default format for directories is very similar to the CP/M "DIR" directory display; only the file names are shown, four entries per line. Note that you may use the CP/M "wildcards" (i.e.- "*" and "?") when asking for a directory.

CROSSTALK provides two options to the DIR command which are NOT found in CP/M. These are the /S (size) and /T (transmission time) options. A typical disk directory is shown below, using the /T option with the baud rate set to 1200. To display this, enter the command:

Command? **DIR /T<CR>**

CROSSTALK will then display:

```
XTALK COM ( 5m) XTALK HLP ( 4m) XTHHELP DIR ( 0m) STD XTK ( 0m)
SETUP XTK ( 0m) SETUP XTS ( 0m)
```

6 File(s), 70K total, 161K free, transmission times for 1200 baud.

Note that the transmission times calculated for each file are at the current baud rate. Changing the baud rate will always affect the time required to send a particular file.

When using the "/T" option, note that transmission times are rounded down to the nearest minute. Files which will take less than one minute to transmit are displayed as 0 minutes. When using the "/S" option, all sizes are rounded up to the nearest K.

6.2

Changing the default disk drive

When you first run CROSSTALK, the program "looks" to see what your current default disk drive is. The program then sets its own default drive to the same drive.

You may change CROSSTALK's default disk drive with the DRIVE command. For example, the command:

Command? **DR B:(CR)**

tells CROSSTALK to use drive B: for any disk operations, unless you provide a specific drive name. For example, if you have set the default drive to B:, you can still view the directory for drive A: by entering the command "DIR A:".

6.3

Reviewing disk files

CROSSTALK has a feature, the TYPE command, which allows you to inspect the contents of a disk file without having to return to CP/M. The TYPE command is similar to the CP/M TYPE command. To review the file BERT, for example, enter the command:

Command? **TYPE BERT(CR)**

If you are capturing information, the command **TYPE** alone will display the contents of the capture buffer.

If the TYPE command is issued at the status screen, CROSSTALK will pause every 23 lines, waiting for you to press a key before displaying the next page. If you issue the command at the terminal screen, you may need to press ^S to pause the display, and ^Q to begin it again. To cancel the TYPE command, press ^C.

6.4

Erasing disk files

Disk seem to have a habit of filling up at the most inconvenient time. The ERASE command may be used to erase files from your disk, allowing you to free up space on a full disk by erasing unwanted files.

You may provide any legal file name to the ERASE command. For example, the command:

Command? **ERASE *.BAK(CR)**

will erase all of the files with a name ending in ".BAK". CROSSTALK will show you the name of each file to be erased, and ask you to confirm that you want to erase the file. You may avoid being asked for confirmation by using the "/Y" option in the command (e.g.- ERase *.BAK /Y).

The command ERASE with no argument will erase the contents of the capture buffer. As with erasing a file, you are asked to confirm the erase unless you specify "/Y" in the command line. Note that you may only erase the capture buffer when you are using the capture to memory method of capture (see chapter 8).

6.5

Function Keys

CROSSTALK can store up to 10 function key assignments with up to 40 characters per key. These keys are function keys 0-9, and are invoked by pressing the TRIP key (default is $\wedge L$ which can be reset with the KEY command) followed by the number of the function key.

CROSSTALK utilizes function keys in two ways. Each key may be assigned either a string of text or a CROSSTALK command. This allows you to program the function keys to perform any number of different functions.

The FKeys command is used to both set and review function key settings. The command "FK" by itself will display the current contents of the keys.

To program function key number 1 with the text "Hello", enter the command:

Command? **FK 1 Hello<CR>**

There are several characters which have special meaning when placed in a function key string. The table below lists each of the special characters.

Character	Effect
	The form separator is used to imbed a carriage return (ENTER key) inside of a function key definition.
@	If the @ character is the first character in a function key string, it tells CROSSTALK that the contents of this function key are to be taken as a command to CROSSTALK. For example, setting F1 to @BYI will cause CROSSTALK to hang up the phone (perform the BYE command) when TRIP 1 is pressed.
\wedge	The caret is used to place control characters in a function key. The character following the \wedge is sent as a control character.

The following examples show some typical function key settings:

Command	Effect
FK 1 @PR /!	Sets F1 to turn printer off or on when pressed. Note that this key assignment begins with the @ character.
FK 2 ID TCA123 ^A^B^C !	Sets F2 to send "ID TCA123", followed by control-A, control-B, control-C, and a carriage return. Note the use of the caret (^) before the control characters, and the use of the vertical bar for RETURN.

6.6

Special Function Keys

There are three special "trip" features within CROSSTALK. They are trip C, P and T. Trip C toggles the state of Capture; it is identical to the CROSSTALK command CA /(CR). Likewise, trip P toggles the state of the printer. Trip T TYPES the contents of the capture buffer. These special features may be invoked either in terminal mode, or when at the command prompt, and they are invoked by typing the current trip KEY, followed by either C, P, or T. To send the current trip KEY, simply type it twice.

6.7

Listing the current parameters

The LIST command has no effect in terminal mode or at the status screen, and will NOT generate an error message. It is valid only when used as a remote command to CROSSTALK.

6.8

Changing the dialing strings

CROSSTALK can auto-dial with most auto-dial modems. The program comes shipped set up for a Hayes Smartmodem or compatible modem. The compatible modems include the U.S. Robotics Auto Dial 212A, and several others.

If you have another type of modem, or if you wish to alter the default set-up for the Hayes-compatible modems, you can do so by either running the SETUP script file (supplied on the CROSSTALK diskette) or by using the DPREFIX and DSUFFIX commands.

The DPREFIX command allows you to specify the dialing prefix for your modem. The default setting for DPREFIX is "ATDT". The DPREFIX string is sent to the modem BEFORE

the phone number. The default setting "ATDT" instructs the Hayes compatible modems to dial with tones. If for, example, you wish to pulse dial, the command:

Command? **DP ATDP<CR>**

will instruct a Hayes modem to dial with pulse dialing.

There are several characters which have special meaning when placed in a dialing string. The table below lists each of the special characters.

Character	Effect
	Imbeds a carriage return (RETURN key) inside of a dialing string.
^	Used to place control characters in a dialing string. The character following the ^ is sent as a control character.
~	Instructs CROSSTALK to "wait" for one second when dialing. Use as many of these as needed for delaying between dialing characters.

The DSUFFIX command is the counterpart to the DPREFIX command. The DSUFFIX command tells CROSSTALK what characters to send to the modem AFTER the phone number.

For example dialing commands for several different types of modems, see Appendix F.

6.9

Printing data while online — The PRINTER command

CROSSTALK can send a "copy" of all terminal screen activity to a printer while online. When the printer is enabled, you can still see incoming data on the screen.

The PRINTER command tells CROSSTALK to turn on or off the output to the printer. The command:

Command? **PRINTER ON<CR>**

will cause CROSSTALK to duplicate any terminal screen information on the printer.

The command "PR /" will switch the printer back and forth from ON to OFF. CROSSTALK is supplied with this command as a special trip feature, trip P. Pressing the trip KEY followed by a P (upper or lower case) will switch the printer on and off with two keystrokes.

Note that "what you see is what you get," and there is no way to disable terminal screen output. Hence, the FILTER and INFILTER commands (see next section) affect what is sent

to the printer, since they affect the screen output. Also, if you have DEBUG on, the "debugged" characters will be printed as they appear on the terminal screen.

It is also very important to note that your printer must be at least as fast as the speed at which you are communicating. That is, if you are connected at 300 baud, your printer must be capable of printing 30 characters per second (carriage returns included), or there will be severe loss of data. If you experience loss of data with the printer on, it is best to capture the data to a disk file (see chapter 8), and print the file after you have finished communicating.

6.10

The FILTER and INFILTER command

Some computer systems transmit special characters, called control-characters. These characters are often used by host computers for various different functions, and are generally not needed by your CROSSTALK system.

These characters can cause some problems, since they show up as graphics characters on many microcomputers. CROSSTALK has two features, FILTER and INFILTER, which allow you to selectively discard these unwanted characters.

The INFILTER command is used to enable or disable the control character filtering. The FILTER command is used to decide which control characters are discarded.

There are a few control characters that you will almost always want to keep. These characters are listed in the table below:

Character	Function
^G	Bell
^H	Backspace
^I	Tab
^J	Line Feed
^M	Carriage Return

Use the FILTER command to change the table of allowable control characters. Enter the command:

Command? **FILTER(CR)**

and CROSSTALK will display the table of all of the possible control characters, much like this:

Filtered: ^@^A^B^C^D^E^F^K^L^N^O^P^Q^R^S^T^U^V^W^X^Y^ZESC^`^]^^^_DEL
Passed: ^G^H^I^J^M

If a character is listed as **Filtered**, then CROSSTALK will "throw away" any of these incoming characters. If it is listed as **Passed**, then the character will appear on the screen.

Note that the FILTER and INFILTER commands affect the data displayed on your screen, the data printed on your printer, and the data captured to the capture buffer.

6.11

The DEBUG command

As mentioned in the last section, some computer systems send "control-characters". CROSSTALK's DEBUG command lets you "see" incoming control characters graphically on the screen. There are two different display modes available.

Option	Description
Off	Control characters are not "debugged".
Hex	All characters are shown as hex numbers. For example, a bell is shown as [07].
Char	Control characters are displayed in the form [^X].

The DEBUG feature is useful in identifying unknown control characters. Note that the command "DEBUG ON" will select CHAR debug mode.

6.12

Terminal mode status line

When you are in terminal mode, CROSSTALK has a feature that will allow you to see the current value of the SWITCH and ATTENTION keys, the status of CAPTURE, and whether the PRINTER is on or off on a "status line". This status line is displayed at the top of the screen when in terminal mode. Although this is a very handy feature, you may find that using the status line is causing your computer to drop characters. For this reason alone, we have made the use of the status line optional via the TSTATUS command. To enable the status line, issue the command:

Command? **TStatus On(CR)**

To disable this feature, use "off" in place of "on". Note that you may also use "/" in place of on or off to toggle the status of TSTATUS.

6.13

Changing the BREAK key assignment

A break is a special signal used by some computer systems to interrupt an operation. You can send a break by pressing the key designated as the BREAK key. The BREAK key can be any non-printing key on your keyboard. For example, the default setting for the BREAK key is ^B. Anytime the BREAK key is pressed when in terminal mode, CROSSTALK will send a BREAK signal to the other computer.

You can change the key assigned to the BREAK key by using the BREAK command. For example, to change the break key assignment to ^Z, enter the command:

Command? **BREAK(CR)**

CROSSTALK will ask you to press the key you wish to use to send a break. To set the BREAK key to control-Z, press ^Z.

6.14

The TURNAROUND command

Some mainframe computer systems use a key other than the RETURN key to signal end-of-line. An example of this is the Univac 9000 series, which uses ^C for end-of-line. The TURNAROUND command tells CROSSTALK what character to send when the RETURN key is pressed.

The default setting for TURNAROUND is the RETURN key. To change the TURNAROUND character to another key, enter the command:

Command? **TU(CR)**

CROSSTALK will ask you to press the key to be sent when the enter key is pressed.

6.15

The ANSWERBACK command

Some computer systems and networks require that all terminals calling into the system be equipped with an answerback. An answerback is an identifier code which identifies your particular terminal to the computer system.

Answerbacks are most commonly used on the TELEX network. Each terminal on the network has its own unique identifier code, and the network can send a code called ENQUIRE to each terminal, asking the terminal to send its answerback.

The ENQUIRE character is ^E. If desired, CROSSTALK can respond to a received ^E by transmitting the contents of trip key number 4. To enable this feature, use the ANSWBACK command.

When ANSWBACK is on, any received ^E character will cause the contents of trip key 4 to be sent to the modem. When ANSWBACK is off, received ^E characters are ignored.

Note that answering CROSSTALK and TRANSPORTER systems send a ^E character to request your password. If you will be calling into a CROSSTALK or TRANSPORTER system which will require a password, turn ANSWBACK on, and enter your password in trip key 4. When the system answers, your password will automatically be sent to the answering system.

Data communications, by it's nature, is an area filled with technical details. Before two computers can even begin to communicate with one another, they must agree on HOW they will do the communicating.

A good analogy to this is two people talking on the telephone. One person must call the other to establish the communications link. When they begin talking, they must both be speaking the same language. Finally, they will usually speak one at a time, so that the other person can hear what is being said.

Communications between two computers is a little more complicated than two people speaking on the phone. Again, one computer must initiate the call (CALL mode), and the other computer answers the call (ANSWER mode). Like the two people in our analogy, they must each "speak" in a manner that the other computer understands.

It is important to remember that when you are calling into another computer system, you will generally have to adjust YOUR communications parameters to suit the requirements of the system you are calling. Most dial-up systems are designed to be accessed by a large number of people, with different types of equipment. For the most part, each terminal calling into a dial-up system must conform to the standards for that system; the system will not configure itself to conform to your parameters.

CROSSTALK provides several commands which control all of the communications parameters. This chapter will explain each of these commands.

7.1

The DUPLEX command

Not all host computer systems operate in the same manner. Some computer systems will echo every character the user types, while others do not echo at all.

When communicating with a computer which does echo your data, you will want to set CROSSTALK to FULL duplex. This is the most commonly used of the two possible settings, and is the default value.

If you call into a computer and can not see yourself typing, the host system is probably not echoing your data. In this case, you would want to set duplex to HALF.

If you see two of everything you type, set duplex to FULL. Note that when you select ANSWER mode, CROSSTALK automatically selects HALF duplex. When you select CALL mode, the program automatically selects FULL duplex.

7.2

Changing the communications speed

CROSSTALK allows you to communicate with modems and other computers at several rates of speed. The rates available on most computer systems are 110, 300, 600, 1200, 2400, 4800, and 9600 baud.

The SPEED command allows you to select the communications speed at which CROSSTALK will operate. You may enter the command:

Command? **SP(CR)**

and CROSSTALK will prompt you to select a speed from a list of choices. Alternatively, the command SPEED 1200 will select 1200 baud. Only the first letter of the speed must be entered. (To select 110 baud, use SPEED 0110 since SPEED 1 selects 1200).

NOTE: Not all micro-computers are capable of "software selectable" baud rates. If this is the case, CROSSTALK can only select between a high speed and a low speed. They are shown as 1200 baud and 300 baud respectively.

7.3

Changing the data word length

Some computer systems require that you communicate with them using a certain combination of data, stop, and parity bits. (For a full explanation of these terms, see the glossary.)

The DATA command allows you to select the number of data bits to be used. You may select 7 or 8 bits. The command:

Command? **DATA 8(CR)**

would set CROSSTALK up for 8 bit data. There is no hard and fast rule about when to use 7 bits, and when to use 8. If you are in doubt, try 8 bits first.

Note that a CROSSTALK protocol transfer (See chapter 11) requires that 8 data bits be used, and that CROSSTALK automatically selects 8 bits whenever a protocol file transfer is performed.

7.4

Changing the parity

CROSSTALK allows you to select the parity to be used by the communications interface. (For a full explanation of parity, see the glossary.) The available options are EVEN, ODD, and NONE. For example, to set the parity to even , enter the command:

Command? **PARITY EVEN(CR)**

Most computer systems ignore parity, and this setting is not usually critical. However, many mainframe computers require that your parity be set correctly. In such cases, consult the mainframe system operator to determine the correct parity setting for the system in question.

7.5

Changing the number of stop bits

As with the PARITY and DATA commands, it is sometimes necessary to change the number of stop bits. The STOP command is used to select either one or two stop bits. As a general rule, most computers use one stop bit at speeds of 300 baud and above, and two stop bits at 110 baud.

7.6

The LFAUTO command

Some computer systems do not send line feed characters at the end of each line. This can cause a problem, since CROSSTALK expects each line to begin with a carriage return and a line feed.

If you call into a computer system and each line "overtypes" the previous line, try turning LFAUTO on. When each carriage return is received, CROSSTALK will add a "fake" line feed. The lines will no longer overprint on your screen.

Before we explain HOW to capture data, we will first explain WHAT it means.

Let's say that you subscribe to a dial-up information system, such as Dun & Bradstreet. You have called in to the system, and requested a credit check on a new customer. Let's also say that you need to include that credit information in a report to your boss.

By using CROSSTALK's capture features, you can save that customer's credit information on your disk, and then use your word processor program to edit the information and include it in your report.

CROSSTALK offers two ways to capture data, and each has its own advantages and disadvantages. The two methods are called "Capture to disk" and "Capture to memory". We'll describe capture to memory first.

8.1

Capturing Data to Memory

Capture to memory does just that — it saves incoming data in your system's memory. After you have captured the data you want, you can write the data to a disk file of your choice.

You can capture as much data as your system's memory will hold. CP/M uses up to 8K of memory, and CROSSTALK uses about 36K, so a machine with 64K of memory can capture about 20K of data.

Capture can be turned off and on at any time, allowing you to selectively capture only the data you want. There are several commands related to enabling and disabling capture to memory. They are:

Command	Effect
CA +	Turns capture ON
CA -	Turns capture OFF, clears capture buffer
CA /	Switches capture between ON and PAUSED. If capture is OFF, this will turn it ON. If it is ON, it will change it to PAUSED. If it is PAUSED, it will change it to ON. This is useful turning capture on or off quickly, especially when assigned to a trip key.

Note that capture may be toggled from ON to PAUSED and PAUSED to ON using "trip C" from either terminal mode or the command prompt.

8.1.1

Saving Captured Data

After using capture-to-memory to capture some data, you'll probably want to save that data in a disk file. The WRite command is used to write the captured data into a disk file of your choice.

As an example, if you had captured a financial report on a prospective new customer, you could issue the command:

Command? **WR B:NEWCUST.RPT(CR)**

to write the captured data into the file NEWCUST.RPT on drive B:. If there is not enough space remaining on the disk to save the entire file, the program will print a warning message, and leave the capture buffer as is. You may then WRITE the data to an alternate drive, or change disks, issue the DRIVE command, and WRITE the data to the new diskette.

As a precaution, if you have captured some data and attempt to either leave the program or erase the contents of the capture buffer without first saving the data, the program will warn you and ask if you wish to save the data to a disk file before proceeding.

8.1.2

Precautions on using capture-to-memory

If you attempt to capture more data than your system can hold in its memory, the program will warn you, and turn capture OFF. If you have a system with a small amount of memory, you may wish to use capture-to-disk as an alternative.

You should be careful of this when capturing large amounts of data at one time. In such cases, it is much safer to use capture-to-disk, since the data will be written to the disk every time the buffer fills up, thus avoiding any unpleasant surprises.

One last word on using capture-to-memory: Remember that the captured data is not written to the disk until you explicitly tell the program to save the data. In the event of a power outage, any data which has been captured will be lost. Using capture-to-disk avoids this problem, since captured data is saved fairly often.

8.2

Capturing Data Direct to Disk

As an alternative to capturing to memory, you may wish to have the program write the captured data directly into a disk file.

To begin capturing data to a disk file, enter the command:

Command? **CA (filename)(CR)**

If the file name given does not exist, it will be created. If the file does exist, the program will give you the option of erasing the old file, or of adding the new data to the end of the old file. In either case, the program will then begin capturing incoming data into the file.

As we outlined in the previous section, capturing to memory poses some potential problems which can be overcome by capturing directly to disk.

When capturing data directly to disk, the program will write the captured data to the disk every time the capture buffer fills up. This is done by sending an XOFF (^S) to the host system, waiting 5 seconds for the host to stop sending, and then writing the buffer to the disk.

If your diskette becomes full when using the capture to disk method, the capture file will be closed automatically, and the captured data will be left in the capture buffer with capture paused. You then have several options. You may use the ERASE command to remove unwanted files, and then WRITE the capture buffer to a new filename, or to the same file you were originally capturing data to using the append option. If you need all the files on your diskette, then you will want to WRITE the data to an alternate drive or to another diskette. If you change diskettes, remember to issue the DRIVE command so that CP/M knows you have changed diskettes.

As a general rule, use capture to disk when:

- You have a small capture buffer (less than 10K)
- You are in an area which is susceptible to power outages
- You will be capturing large amounts of data at one time
- You are downloading a specific file from a host

You may be wondering WHY there are two different methods of capturing data, when it seems that capture-to-disk has several advantages. The answer is that while capture-to-disk is inherently more reliable, one may not be sure that the information to be captured is worth saving. In this case, the capture buffer may be erased, and capture turned off without having to waste time writing data to the disk. Also, one may be capturing small amounts of data which is not at all related. In this case, several WRITES to different files may prove to be a better method.

8.3

Retro-capture

CROSSTALK has a unique feature called RETRO-CAPTURE, which allows you to capture data which has already scrolled off of the screen. The program maintains a "circular" buffer, and all incoming data is routed through this buffer. The buffer holds 4k of characters. As new data comes in, the oldest data is "bumped" out of the buffer to make room for the new data. RETRO-CAPTURE allows you to retrieve data out of this buffer and store it in the

main capture buffer. Since the screen holds 1920 characters (24 lines of 80 characters), the retro-capture circular buffer can hold over 2 "screenfuls" of text.

To activate retro-capture, enter the command:

Command? **CA <CR>**

The program will respond with a message telling you that RETRO-CAPTURE is complete, and will turn capture ON. Note that RETRO-CAPTURE may only be used when capture is OFF.

If you wish to capture a specific number of lines, you may enter the command:

Command? **CA <xx<CR>**

where xx is the number of lines you wish to capture.

8.4

Capture search and status

When capturing data to memory, you can search through the capture buffer for a specific word or phrase. You can also ask CROSSTALK to tell you how much data has been captured.

Both of these functions are accomplished through the CStatus command. The command "CS" alone will show you the status of the capture buffer — how many "K" of characters have been captured, and how much room (in "K") is free.

To search through the buffer for a specific word or phrase, enter the command:

Command? **CS text<CR>**

where "text" is the word or phrase you wish to search for. For example, if you had captured some data regarding wheat commodities, the command:

Command? **CS WHEAT<CR>**

would search the capture buffer for all occurrences of the word "wheat". If the word "wheat" exists in the capture buffer, the program will display the contents of the capture buffer surrounding the located word. After displaying the word, the program will offer to continue searching through the buffer.

Note that the CS command may be issued with either method of capture active. However, if capture to disk is active, the CS command does not search data that may have previously been written to disk; it only searches through what is currently in the capture buffer.

8.5

Erasing the capture buffer

It is often necessary to erase the contents of the capture buffer. The ERASE command allows you to erase all of the data in the capture buffer. The command:

Command? **ERASE<CR>**

instructs CROSSTALK to discard all data in the capture buffer. As a precaution, the program asks you to confirm that you really want to erase the buffer before the data is actually discarded. As with the ERASE filename option, you may use the "/Y" option to tell CROSSTALK not to verify the erase. To do this, issue the command:

Command? **ERase /Y<CR>**

8.6

Reviewing the capture buffer

CROSSTALK allows you to review the contents of the capture buffer. The TYPE command is used to display the capture buffer data on the screen. Alternately, the special trip sequence, trip key-T, will display the contents of the capture buffer. The TYPE command may be entered at either the status or the terminal screen. Recall that when entered at the status screen, CROSSTALK will pause every 23 lines. To cancel the TYPE command, press ^C. Note that you may review the capture buffer contents with either method of capture active.

9

Sending a File to Another Computer

In the last chapter, we looked at the capture feature which provides a means of capturing data from a remote computer system and saving it on your disk. In this chapter, we will show you how to use the SEND command to send ASCII data from your disk to another computer system, and we will describe all of the commands associated with the SEND command. It is these commands which make it possible for CROSSTALK to upload text files to virtually any other computer.

Before we proceed any farther, please keep in mind that you will use the SEND command ONLY when you are communicating with a system which is NOT running CROSSTALK or a compatible file transfer program. The process of exchanging files with another CROSSTALK compatible system will be covered in the next section.

9.1

The SEND command

As mentioned above, CROSSTALK can send a text file (any file composed of only the valid ASCII characters) to another computer which is not running CROSSTALK. To send a text file to another computer, enter the command:

Command? **SEND <filename><CR>**

This will cause CROSSTALK to send the file named by <filename> to the host computer. However, CROSSTALK does not "know" how to make any particular host system capture and save the information you are sending it. You will need to issue the appropriate command(s) to your host system interactively before issuing the SEND command so that the computer you called will know what to do with the information you are sending it.

Note that CROSSTALK DOES honor the XON/XOFF protocol when SENDING a file.

9.2

Commands which affect the SEND command

Sending a text file to another computer system is not always as simple as merely issuing the SEND command. Since no two computer systems operate in exactly the same way, it is often necessary to modify the manner in which CROSSTALK transmits text when SENDING a file. There are several commands which affect the SEND command. In many cases, it will be necessary to use one or more of these commands before issuing the SEND command.

These commands are: CWAIT, LWAIT, OUTFILTER, TABEX, BLANKEX, and UCONLY. Each of these commands has its own distinct purpose, and each may be used in conjunction with any of the others to afford you the maximum flexibility.

The remainder of this section will explain each of these commands, and show how to use them.

9.2.1

Waiting for slow systems — The LWAIT & CWAIT commands

Many computer systems require some sort of an inter-line delay when accepting text files. For example, many systems will send a prompt when they are ready to accept a line of text, others may simply require that you wait for a specified period of time between lines. The reason for this is that most host computer systems are designed to service terminals with people typing at those terminals. Since very few of us can type at 120 characters per second, the systems don't have to respond to input at full speed.

When you use CROSSTALK's SEND command to send a text file to another computer system, your system looks like a very fast typist to the host system. Unfortunately, many computer systems simply can not keep up. If you send text faster than the host system can accept it, some characters may be lost. To avoid losing any information, CROSSTALK has several methods for "waiting" for slower computer systems.

CROSSTALK has two commands which control this waiting. The CWAIT command provides a wait between each transmitted character, and the LWAIT command can provide a delay between each line. In most cases, you will need to use only the LWAIT command.

9.2.1.1

Waiting between lines — The LWAIT command

The LWAIT command is used to select a method of waiting between transmitted lines. There are several options available, and each has its own distinct function. The options are explained below.

LWAIT PROMPT X option:

With this option selected, CROSSTALK will send a line of text, wait for a prompt of X characters, and then send the next line. This method is most useful when sending text files to systems which send a fixed number of characters as a prompt.

LWAIT LEARN option:

This option is very similar to the PROMPT option, except that it counts the number of characters in the prompt for you. When LEARN is selected, CROSSTALK sends the first line

of the file, and then asks you to press the space bar when the host system is finished sending its prompt. While the program is waiting for you to press the space bar, it is also counting the number of characters received from the modem. After you press the space bar, the program sets up LWAIT PROMPT mode, using its count of characters as the prompt length.

This method is the simplest to use, but again, will only operate properly with systems which send a fixed-length prompt.

LWAIT MANUAL option:

In this mode, CROSSTALK sends a line of text, asks you to press the space bar, and then sends the next line of text. This process is repeated until there are no more lines to send. Since you can see any data being returned from the host, you will be able to control the flow of data to the host system.

This option is useful when sending small files to systems which do not always respond to input in a consistent manner. Since you can see the host's response between each line, you can control the flow of data accordingly.

LWAIT QUIET X option:

In this mode, CROSSTALK sends a line of text, and then waits until the line is "quiet" (i.e. - no characters received) for a period of "X" tenths of seconds. The next line is then sent, and the process is repeated until there are no more lines to send.

This option is most useful when sending text to a system which sends a random-length prompt between lines, and has an inconsistent delay in processing text input.

LWAIT DELAY X option:

This option is similar to the QUIET option described above, except that the program waits for "X" tenths of a second, regardless of what was received from the host. It is most useful for introducing a short inter-line delay on systems which can not quite take full-speed transmissions, yet is consistent in the delay necessary to accept text.

LWAIT CHARACTER "?" option:

When this option is selected, CROSSTALK sends each line of text, and then waits for the specified character before sending the next line. This option is most useful when sending text to systems which send a random-length prompt, but only when the prompt ends in a consistent character. It is also especially useful for sending to many half-duplex mainframes, such as IBM systems running TSO. These systems typically do not echo received data, but send a single character prompt when the system is ready for the next line. By waiting for the specific character, you are assured that you haven't begun sending the next line of text before the host system is ready for it.

LWAIT ECHO option:

When LWAIT is set to ECHO, CROSSTALK sends each line of text, and then waits for the carriage return character to be echoed back from the host computer. The next line is then sent.

This option is useful for sending text files to systems which send a carriage return to indicate that the system is ready to accept another line of text.

9.2.1.2**Waiting between characters — the CWAIT command**

The CWAIT command is similar to the LWAIT command. While LWAIT tells CROSSTALK how to wait between each line of text, CWAIT tells CROSSTALK how to wait between each transmitted character. These two commands operate independently of each other which enable CROSSTALK to upload data to virtually any host computer.

For most applications, inter-character delays will not be necessary when uploading a file, and CWAIT may be left as NONE. CWAIT is typically used when sending text files to bulletin boards and older mainframes which are slow to respond to incoming data.

The CWAIT command has two options: Delay and Echo. The DELAY X option instructs CROSSTALK to wait X thousandths of a second between characters. This option is most useful for introducing a slight inter-character delay to accommodate systems which can not take full-speed input.

The ECHO option tells CROSSTALK to wait for the last transmitted character to be echoed back from the host before transmitting the next character. Since the character must make a full "round trip" before the next character is sent, the effective transmission speed is cut in half. Use this option sparingly, only on systems which are very slow to respond to input. Note that this option can not be used when CROSSTALK is set to half duplex, since characters are not echoed back in half duplex operation.

9.2.2**Expanding tab characters — the TABEX command**

CP/M systems interpret the tab key to mean "move the cursor to the next column position which is a multiple of eight". Some systems do not allow the use of tab characters in text files. If you are transmitting a text file containing tabs to a host system which does not support tabs, you will need to use the TABEX command.

The TABEX command tells CROSSTALK that if a tab character is encountered in a text file, send the appropriate number of spaces instead. Setting TABEX to ON enables this feature, setting it OFF disables it.

If the system you are sending to does support tabs, you can save considerable transmission time by leaving TABEX OFF. When TABEX is on, each tab will be transmitted as up to eight spaces.

9.2.3

Converting lower case to upper — the UCONLY command

Some older computer systems do not support lower case letters. If you have a text file containing lower case letters, and you need to send the file to a system which does not support lower case, the UCONLY command can help.

When UCONLY is ON, all lower case letters are converted to upper case before they are transmitted. Upper case letters and punctuation marks are not affected.

9.2.4

Expanding blank lines — the BLANKEX command

Some computer systems take a blank line to mean "I'm finished sending text now". If you are trying to send a file containing blank lines to a system which doesn't want to see any blank lines, use the BLANKEX command.

When BLANKEX is ON, blank lines are converted to a one character line, consisting of a single space character. This will usually alleviate the problem.

9.2.5

Removing line feeds — the OUTFILTER command

Text files created with word processors and text editors contain a carriage return and a line feed at the end of each line. Many computer systems do not need the line feed character, and some systems may create an additional blank line from the line feed character.

When OUTFILTER is ON, no line feed characters will be transmitted. If a text file contains carriage returns and line feeds at the end of each line, only the carriage return will be sent.

CROSSTALK can operate in one of two major modes. These two modes are CALL and ANSWER. CALL mode is used for making calls, and ANSWER mode is used for answering calls.

Besides the obvious difference, there are numerous other differences between the two modes. First, and most importantly, when you are in call mode, you know that YOU are going to be entering all of the commands. In answer mode, either end (the caller or the answerer) may enter commands to CROSSTALK, although the caller will usually be entering the commands.

To complicate matters just a little more, some commands respond differently depending on who entered the command. Some commands which have potentially harmful effects are not valid when entered by a remote caller.

Further, an unattended answer-mode system is "wide open" to abuse and malicious mischief. CROSSTALK has several built-in safeguards to protect you and your system from abuse.

This section will explain the operation of answer mode, the protection features, and will describe the commands which operate differently in answer mode.

To set CROSSTALK to answer mode, enter the command:

Command? **MODE ANSWER<CR>**

When you select answer mode, three things happen: First, the program switches to answer mode. When the program is in answer mode and a GO command is issued, the program watches the modem, waiting for a call, rather than dialing a number as it does in CALL mode.

Second, selecting answer mode causes the program to select HALF duplex (local echo ON). This is done so that you can see what you are typing.

Third, remote echo is enabled so that persons calling into your system can see themselves typing.

When someone calls into an answer mode system, he may enter commands to CROSSTALK, just as you can enter commands from your end.

10.2

Password protection

After CROSSTALK is set to answer mode, and the GO command is issued, the program waits for an incoming call. When an incoming call comes in, the program checks to see if a password is required. If an answer password has been set (with the PWord command), the caller is required to enter the proper password before being allowed access to the system.

If the caller does not provide the correct password after three tries, the program hangs up, and waits for another call. If PWORD is blank, then the caller is not required to enter a password.

The first time the program asks for a password, it sends the message "Enter Password:", followed by a control-E. The control-E character will automatically request an "answerback" from another calling CROSSTALK or TRANSPORTER system. If the caller has his answerback set to the correct password for the system, he will automatically be logged in.

10.3

Access protection

CROSSTALK has a feature which allows you to restrict the amount of access a caller has to your answer-mode system. The ACCEPT command is used to set this level of access. The options to the ACCEPt command are:

Option	Action
Everything	Allows caller full access to system.
Creates	Allows caller to append data to or create new files, but not to overwrite existing files.
Appends	Allows caller to append captured data to an existing file, but not to create any new files.
Nothing	Prohibit caller from writing to any files.

10.4

Greeting message

Once a caller has entered the correct password, or if no password is required, the program prints the contents of the NAME command as a "greeting" message.

Use of this feature is entirely optional. If no name has been entered, no message will be sent.

10.5

Remote commands

Once a caller has called into a CROSSTALK system, he may enter commands to the answering system. When the caller types the COMMAND character (usually ^C), the answering CROSSTALK system sends the prompt:

Command ?

When the caller sees the prompt, he may enter a command to CROSSTALK. Note that there are many commands (QUIT and XCPM to name two), which the caller is not allowed to enter. If the caller enters one of the "forbidden" commands, the program sends him a reminder message, saying that he can not enter that particular command from his end.

The COMMAND character is normally ^C. We recommend that you leave COMMAND set to ^C to maintain compatibility with other CROSSTALK and TRANSPORTER users.

CROSSTALK provides two methods of performing error checked file transfers: the CROSSTALK protocol transfer and the XMODEM, or MODEM7 protocol transfer. The CROSSTALK protocol should be used whenever you are communicating with another CROSSTALK or CROSSTALK compatible system for two reasons: 1) the error checking method used by the XMODEM protocol is not nearly as accurate as CROSSTALK's and 2) the XMODEM file transfer method can be very cumbersome to use. It is important to remember that an error checked transfer can be done ONLY between two compatible systems.

11.1

CROSSTALK protocol transfers

There are two commands directly related to the CROSSTALK protocol transfers. These two commands are XMIT (transmit) and RQUEST (request). XMit is used to send files from your system to another CROSSTALK system, and RQuest is used to request files from another system.

Protocol transfers are done "transparently". That is, when a transfer is being performed, you do not actually see the file on your screen. Instead, a transfer summary is displayed, which shows you the progress of the transfer.

It is important to note that CROSSTALK will automatically select 8 DATA BITS and NO PARITY when a protocol transfer is initiated, and that CAPTURE must be OFF. Also, when performing protocol transfers between two CROSSTALK systems, one of the systems must be in ANSWER mode, the other in CALL (or Originate) mode, even if the computers are hardwired. It is important to remember that CROSSTALK assumes that an answer mode system is entirely unattended; that is, that no commands will be issued from the answer mode system. **All transfer commands MUST be issued by the call mode system.**

11.1.1

Transmitting files to another CROSSTALK system

The XMIT command is used to transmit files to another CROSSTALK or TRANSPORTER system. Transfers may be done on a single file, or on a logical group of files. For example, the command:

Command? **XMIT *.BAT<CR>**

will send all of the files with the type ".BAT" from the calling system to the answering system.

As an example, let's say that you have called into an answer-mode CROSSTALK system, and that you have a file named FRED.TXT that you want to send to the other system. After placing your call and establishing communications, press your ATTENTION key, and enter the command:

Command? **XMIT FRED.TXT(CR)**

The program will initiate the transfer, and then print a summary message showing the progress of the transfer like this:

Sending ... A:FRED.TXT

Block	% Complete	Errors	Consec. Errors
5	14	0	0

After the transfer is complete, the program will tell you it is finished. If the program can not complete the transfer, it will print an error message in the command line, and cancel the transfer.

If you wish to cancel a transfer at any time, you may do so by pressing the ATTENTION key. The transfer will be cancelled, and the incomplete file will be deleted from the answering system.

11.1.2

Requesting files from another CROSSTALK system

The RQUEST command is the opposite of the XMIT command. Use the RQUEST command to "ask for" files from an answering system. If, for example, you have called into another CROSSTALK system, and wish to get all of the BASIC programs from the other system, enter the command:

Command? **RQ *.BAS(CR)**

CROSSTALK will ask the other system to send it all of the files with the type ".BAS". If no such files exist on the answering CROSSTALK system, an error message will be printed. If any files exist, the program will begin the transfer. The transfer summary is displayed, just as in the XMIT command.

You may cancel a transfer at any time by pressing the ATTENTION key.

11.1.3

Alternate drive usage

In the examples above, we did not specify a drive name. If no drive name is given, the program assumes that you want to send or receive using the current default drive.

You may send and receive files to and from any drive on the system. CROSSTALK allows you to specify both a source and a destination drive name, much in the same way as the CP/M PIP command. For example, the command:

Command? **XM B:FRED.TXT A:(CR)**

will cause the file FRED.TXT to be sent from drive B: on the sending system and written to drive A: on the receiving system.

Similarly, the command:

Command? **RQ B:FRED.BAS A:(CR)**

will cause the file FRED.BAS to be sent from drive B: on the sending system to drive A: on the requesting system.

11.1.4

Error handling during transfers

CROSSTALK automatically detects and corrects errors during protocol transfers. If a block of data is received incorrectly, the receiving system sends a message to the sending system, asking that the block of data be re-transmitted.

This process is done automatically. When a block of data is received incorrectly, the program shows an error in the transfer summary.

If you notice an unusually large number of consecutive errors during a transfer, you may wish to cancel the transfer. A transfer can be cancelled at any time, from either end of the transfer, by pressing the ATTENTION key.

A large number of errors is usually indicative of some type of hardware problem. When transferring files via modem, the most likely cause is a bad telephone connection. The best remedy is to disconnect and try again.

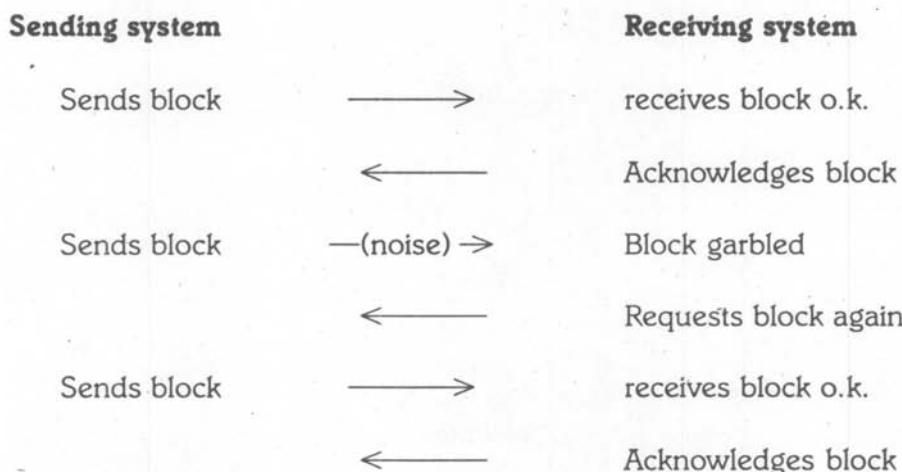
11.1.5

The BKSIZE command

When transferring files, CROSSTALK sends data in groups called "blocks". Each time a block is sent, the program performs a mathematical check on the block, called a CRC check. If the CRC calculation does not match at both ends of the transfer, the program re-transmits the block again.

The larger the block of data is, the more time it takes to re-transmit the block. If telephone lines were perfect, we wouldn't have to worry about CRC checks, block sizes, or protocol transfers. Unfortunately, the telephone network is far from perfect. Noise and "static" can create big problems when transferring files via telephone.

CROSSTALK defines a "block" as being the BLOCK SIZE times 256 bytes (1/4 Kilobytes) of data. After every block, the program asks the other end if it received the block correctly. The dialogue between the two systems looks like this:



This process is repeated until the entire file has been sent. In the case of an error, more data will have to be re-transmitted if the size of the block is large.

CROSSTALK allows you to select the block size with the BKSIZE command. The block size may be set from 1 to 10, which corresponds to 256 to 2560 bytes per block.

On an extremely quiet line, such as two computers wired together with no modems, increasing the block size will slightly increase the transmission speed. On a noisy phone line, decreasing the block size may increase the transmission speed.

The reason for this is that on a quiet line, you will never have to re-transmit any blocks. With a larger block size, the program spends less time checking, since the checks are performed at longer intervals, and the two CROSSTALK systems have to converse less often. With a small block size, the program spends a little more time checking and must handshake more frequently, but has to re-transmit fewer characters in the event of an error.

The default value for block size is 1. If most of your transfers will be performed over a modem, we suggest that you leave the block size set at 1. If you are transferring files over a cable to another computer, you may wish to use a larger block size.

11.2

XMODEM file transfers

Two commands not found in the previous release of CROSSTALK, XXMODEM and RXMODEM, add full XMODEM file transfer capabilities. XMODEM is a "public domain" file transfer protocol which is available on a large number of public-access "bulletin-board" systems as well as in other communication programs.

We have added XMODEM capabilities so that CROSSTALK users may exchange files with these other communications programs which use the XMODEM protocol. XMODEM is not nearly as easy to use as CROSSTALK's own file transfer protocol, and the error checking method used by XMODEM is not as accurate as CROSSTALK's own, so we recommend that you use the CROSSTALK protocol file transfer whenever possible.

Since there is no "standard" implementation of XMODEM, it is a little difficult for us to describe exactly how to use XMODEM for file transfers. The wide variety of other communications programs which use XMODEM vary in their command structure. Most XMODEM bulletin board systems are "menu-driven", so it is usually a fairly simple matter to determine the correct procedure for beginning a transfer.

There are two CROSSTALK XMODEM-related commands: the RXMODEM command is used to tell CROSSTALK to receive an XMODEM file transfer, and the XXMODEM command is used to send a file with the XMODEM protocol. When initiating an XMODEM transfer, CROSSTALK will automatically select 8 data bits and no parity, just as in the CROSSTALK transfers. Unlike the CROSSTALK protocol transfer, however, the XMODEM file transfer can only send one file per command.

Due to the wide variety of other programs which use the XMODEM protocol and the frequency with which public domain software changes, **our customer service personnel cannot answer questions regarding the use of XMODEM file transfers.** Our implementation of the protocol has been tested extensively with bulletin boards nationwide with no reported problems.

11.3

"Hard" errors during transfers

There are several types of errors which may be encountered during a transfer that CROSSTALK can do nothing about. We call these "hard" errors.

The hard errors are:

- No disk in drive or physical disk error
- Carrier lost during transfer
- Other computer's disk is full
- Requested file not found at other computer
- Other computer's operator cancelled transfer

When CROSSTALK encounters any of these errors during a transfer, the program will report the error on the status line, and cancel the transfer. Correct the problem, and try the transfer again.

12**Command files**

If you frequently call the same computer system, you will find it convenient to create a command file for that system. Basically, a command file is a list of commands, stored in a disk file. CROSSTALK can load and perform the commands contained in a command file just as if the commands were entered from the keyboard.

Using command files saves you from having to enter the phone number, name, modem speed, trip key settings, and other information for a particular computer system each time you wish to place a call. Instead of entering all of the information each time you call a particular system, you enter it once, and then save the information in a command file.

12.1**Executing a Command file**

When you first invoke CROSSTALK, the program will search the disk for the command file STD.XTK, the "standard" command file. This is handy for setting CROSSTALK's default settings, since this file is loaded every time you run CROSSTALK (more about STD.XTK in section 12.3). When finished loading STD.XTK, or if STD.XTK is not found, CROSSTALK will search for all of the available command files on the disk, and ask you to choose one. You may hit the return key without selecting a file to go to the status screen, or you can select a file for CROSSTALK to LOAD, in which case the command file will be loaded and CROSSTALK will automatically dial the number.

If you know ahead of time which command file you want to use, from your system prompt, type:

XTALK filename(CR)

This tells CROSSTALK that you want to load the file named in (filename) as the command file. The program will load in the file, and begin dialing immediately.

Command files may contain any CROSSTALK command. They generally contain all of the information that CROSSTALK needs to establish communications with another computer system. In addition to the "essentials", command files also contain information on all of the trip key settings.

Command files may also pass control to another type of special file called a script file. We'll explain script files and their function a little later.

A typical command file looks like this:

(this example is broken into two columns to save space)

NAme	Big Mainframe Computer	DUplex	Full
NUmber	98765432	OUtfiter	On
ACcept	Everything	PArity	None
ANswback	On	PRinter	Off
ATten	^A	SPeed	1200
BReak	^B	STop	1
SWitch	^L	TAbex	Off
KEy	^L	BLankex	Off
CWait	None	UConly	Off
LWait	Learn	TStatus	Off
DEbug	Off	FK 0	""
DPrefix	"ATDT"	FK 1	"userid abc123!"
DSuffix	"!"	FK 2	"password ^A^B^C"
RDials	10	FK 3	"mail check!"
INfilter	On	FK 4	"directory!"
LFauto	Off	FK 5	""
MOde	Call	FK 6	""
PWord		FK 7	""
TUrnarnd	^M	FK 8	""
BKsize	1	FK 9	""
CApture	Off	GO	Q30/45
COmmand	^C		
DAta	8		
FIlter	-----+ + + + - - - - -		

(cont'd next column)

Note that the last item in this command file is a GO command. This will cause CROSSTALK to begin dialing the phone when this file is loaded.

12.2

Creating command files

There are two different ways to create command files. The easiest way is to use CROSSTALK's SAVE command. To create a command file with SAVE, follow these steps:

1. Bring up CROSSTALK
2. Set the phone number with the NU command.
3. Set the location name with the NA command.
4. Set any desired trip keys with the FK command.
5. Set the desired modem speed with the SP command.
6. Change any other settings you wish (parity, stop bits, etc.) with the appropriate commands.

7. Enter the command "SAVE filename", where filename is the name you wish to assign to this command file.
8. Your command file is now saved.

This procedure will create a command file much like the one in our example above. All of the command settings necessary to for a particular system have been saved. Note that CROSSTALK will automatically insert the LOad command at the end of the STD.XTK file, and will insert the GO Q30/45 command at the end of all other command files.

Command files are stored as ascii text, and may be created or altered with any text editor.

To create your own command file, start out by having CROSSTALK save a command file with the SAVE command, as above. After the file is saved, exit CROSSTALK, and bring up your text editor. Then use the text editor to change or add the necessary lines to the file.

Remember that any command may be part of a command file. By combining commands in a file, it is possible to set up an "automated" CROSSTALK session.

The following example shows a command file which will dial the Microstuf test center until it gets a connection, then request a file from the center, and send a file to the test center. After the file transfers have been completed, the program will hang up and dial another call.

Command	File contents:	Comments:
Name	Microstuf Test Center	
Number	1-404-998-8033	Set the phone number
Speed	1200	Set 1200 baud
Parity	none	
Data	8	Set 8 bits, no parity
GO r 30/30		Dial every 30 seconds
WA d 50		Kill 5 seconds
Rquest	MESSAGE.TXT	Get this file
Xmit	MORE.TXT	Send this file
BYe		Hang up
Load	ANOTHER.XTK	And go do another file

12.3

Changing CROSSTALK's default settings

All of CROSSTALK's default settings are stored in a file named STD.XTK. When you first run CROSSTALK, the program searches the disk for STD.XTK, and sets itself up accordingly.

The STD file is just like any other .XTK file with one exception: the last command in the STD file is a "LOAD" command. When CROSSTALK encounters this LOAD command with no file name, it presents you with a "menu" of available command files.

You may change any of CROSSTALK's default settings simply by entering or changing the appropriate command in STD.XTK. For example, if you want CROSSTALK to come up with the trip keys set a certain way, you can change the FKeys commands in the STD.XTK file. The next time you bring up CROSSTALK, the function keys will be set as you want them.

A special script file, SETUP, is provided with your copy of CROSSTALK. You can run this script file by entering the command "**DO SETUP**". CROSSTALK will ask you a series of questions, and then save the new default settings in STD.XTK for you. The STD.XTK file may also be edited with any text editor.

12.4

Script files

CROSSTALK provides a means for users to write and run "programs" in CROSSTALK's own "language". These programs are called **SCRIPT FILES**, and they add a great deal of flexibility to CROSSTALK.

A properly written script file, like any good program, allows the user to accomplish an objective without having to consider either the program or the language used. Like programs written in conventional programming languages, a CROSSTALK script file may be as simple or as complex as you wish it to be.

Script files are very similar to command files. In fact, a script file may be included inside of a command file, or it may be called from a command file.

The major difference between the two types of files is that command files are normally used to provide parameters to tell CROSSTALK how to dial a particular computer system, while script files tell CROSSTALK what to do once it has made the connection.

Script files are performed by entering the DO command. As script files are similar to command files, the DO command is similar to the LOAD command. Use DO to perform a script file, and use LOAD to perform a command file.

Any CROSSTALK command may be included as part of a script file. In addition, there are several commands which are useful ONLY in script files. The remainder of this chapter will describe those commands, and show examples of typical script files.

If you are proficient in BASIC, PASCAL, FORTRAN, or most any other high-level programming language, you should have no trouble learning the CROSSTALK command language. **Please keep in mind that our customer service department can NOT help you write script files for any particular purpose, any more than Microsoft's customer service department can tell you how to write a BASIC program. Programming in any language is an acquired skill, and we can not give programming courses over the phone.**

12.5

Organization of a script file

A script file is a disk file consisting of one or more CROSSTALK commands. Script file names must have the file type ".XTS". You can create a script file with any text editor program, such as the ED program included with CP/M. Multiple commands may be placed on the same line, as long as they are separated by a space, a colon, and another space. Commands in a script file are performed in the order they appear in the file. You may document a script file by using a semi-colon followed by any text as a comment, much like the basic REM command.

12.6

How script files are performed

When CROSSTALK first connects to another computer system, the program checks the current disk to see if a script file exists with the same name as the currently loaded command file.

For example, if you use a command file named "BERT.XTK" to make a call to another computer, CROSSTALK will look for "BERT.XTS" as soon as the call is connected. If the .XTS file exists, CROSSTALK will automatically perform the commands in the script file.

You may also perform a script file by entering the command "**DO BERT**", where "BERT" is the name of the file you wish to perform. If you enter "DO" with no file name, CROSSTALK will display a menu of the available script files, and ask you to choose one.

The number of script files you may have is limited only by the amount of available disk space. Script files may pass control to other script files, so that commonly used commands need not be repeated in several different files.

12.7

Writing a script file

Before you sit down to actually write a script file, it is a good idea to have in front of you a printed copy of what it is you want to do. For example, if you want to write a script file to call The Source, log in, and send some commands to The Source, you'll need to know exactly what questions The Source asks, how it expects you to answer them, and so on.

The easiest way to obtain a printed copy of such an exchange is to call the system first, and either capture the session to a disk file, or print it on your printer while you are on line.

After you know exactly what to do and when to do it, the task of writing a script file becomes much simpler.

12.8

Special commands for script files

There are a number of CROSSTALK commands which are useful only in script files. Many of these commands have counterparts in the BASIC language. These commands are:

Command:	Purpose:
ABORT	Cancels operation of a script file.
ALARM	Sounds an alarm tone on the computer.
ASK	Asks the user a question, and obtains an answer, much like BASIC's INPUT command.
CLEAR	Clears the current screen.
DO	Resumes a suspended script file.
IF	Logical operator, similar to the IF statement in BASIC and PASCAL.
JUMP	Causes CROSSTALK to begin performing commands in a script file at a label position. See LABEL.
LABEL	Marks the line containing the LABEL statement so that a JUMP command can locate it, akin to line numbers in BASIC, or to a label name in assembly language.
MESSAGE	Prints a message on the current screen, similar to BASIC's PRINT statement.
- (NOT)	Logical operator, similar to BASIC's NOT operator.
RWIND	Re-starts the script file from the beginning, usually used when an error is encountered.
SBREAK	Sends a BREAK to the remote computer. This command has exactly the same effect as pressing the BREAK key.
SCREEN	Allows you to switch to a particular screen (either the status or terminal screen) under script file control.
SKIP	Skips forward a number of lines in a script file, or forward to a label much like the JUMP command.
REPLY	Sends a single line of text to the modem
WAIT	Waits until a certain condition has been met.
WHEN	Tells CROSSTALK to perform an action upon receipt of a certain string of text.

Each of these commands will be explained fully in the rest of this section.

12.8.1

The ABORT command

The ABORT command causes CROSSTALK to cancel the currently active script file. This command is normally used when an error condition has been detected, and you wish to cancel the remainder of the script file.

12.8.2

The ALARM command

The ALARM command is used to sound an audible alert tone to let you know that CROSSTALK has done something. The command:

Command? **AL(CR)**

sounds the alarm tone. This is the same tone that you will hear when CROSSTALK connects when given the GO Rxx/yy command.

12.8.3

The ASK command

The ASK command is used to print a message on the screen and wait for a response. The response may either be a single character, or it may program a function key. The type of input required is dictated by the syntax of the ASK command.

12.8.3.1

Single character input

CROSSTALK is capable of accepting a single character input using the ASK command, and later acting on the user's input with the IF, JUMP or SKIP commands (explained later in this section). For instance, the command:

ASK Press any key to continue:

when executed in a script file would cause the prompt:

Press any key to continue:

to appear on the status line, and CROSSTALK would then wait for a single key input from the user. After a key is pressed, CROSSTALK will continue executing the script file at the next command.

12.8.3.2

Setting function keys with ASK

You may use the ASK command to set the value of any of CROSSTALK's function keys. As with the single character input mode of the ASK command, CROSSTALK will display the prompt and wait for input. To program a function key with the ASK command, use the following syntax:

ASK @x text...

where x is a number from 0-9 (corresponding to the function keys 0-9) and text is any text. For instance, to request input for function key 3, use the following command:

ASK @3 Enter your user ID:

This will cause CROSSTALK to display "Enter your user ID:" in the status line, and wait for input. The characters entered by the user will be assigned to function key 3, and input is terminated by a carriage return.

12.8.4

The CLEAR command

The CLEAR command is used to clear CROSSTALK's "terminal" screen. This command has no effect when issued at the status screen. The CLEAR command is useful when writing scripts that display messages or menus (see MESSAGE command in this chapter).

12.8.5

The DO command

The DO command tells CROSSTALK to perform the commands contained in a "script" file. Script files are files containing commands for CROSSTALK to perform.

There are four forms of the DO command. The table below shows how each works.

Command	Action:
DO -	Disables the "auto-linking" to a script file after a connection is established
DO	When no other script file is active, this command will display a menu of the available script files, and then ask you to choose one. The chosen file will then be performed.
DO	If a script file has been suspended by the user or by a WAIT MANUAL command, entering "DO" will resume the script file processing.
DO BERT	Performs the commands contained in the script file "BERT.XTS".

You may create a script file with any text editor. Script files must have the file name extension "XTS". The script files may contain any list of commands you wish for CROSSTALK to do. There are two commands which are especially useful in script files. These are the "REply" and "WAit" commands.

12.8.6

The LABEL command

The LABEL command is used to "name" a line in a script file. It performs no operation in and of itself, but it allows another command to refer to a certain point in a script file by name.

The command:

LABEL BERT

assigns the name "BERT" to that particular line in the script file. Note that a label name MUST start with a letter.

12.8.7

The IF command

The IF command is very important, since it is the only "decision making" command. Earlier, we showed how the ASK command can be used to print a message and get a single character response from the user. The IF command is used to make decisions, based on the result of an ASK command. As an example, these two lines ask the user if he wishes to clear the screen, and then clears the screen if he answers "Y".

ASK Do you wish to clear the screen?

IF Y clear

You may provide a series of letters to the IF command, as long as the letters are consecutive. No spaces or commas are allowed between the letters. For example, the commands:

ASK Do you wish to leave CROSSTALK?

IF XEY quit

will cause CROSSTALK to perform a QUIT command if the user replies by pressing the X, E, or Y keys.

There are two characters which have special meaning when used in conjunction with the IF command.

The minus sign (-) is used to logically negate the arguments supplied to the IF command — that is, it reverses the conditions specified in the IF command.

For example, the commands:

LABEL HERE
ASK Enter A,B, or C:
IF -ABC jump HERE

(commands to be executed if A,B or C pressed)

will ask the user to enter A, B, or C, and skip to the label HERE if any other key is pressed. Notice that we used the LABEL command in this example, to tell the JUMP command where to go. The JUMP and LABEL command will be explained in greater detail later.

The dollar sign (\$) is used to mean ONLINE. If you are connected to another system, the "IF \$" condition will be TRUE. For example, the command:

IF \$ ALARM

will sound the alarm ONLY if you are connected to another computer.

12.8.8

The JUMP command

The JUMP command causes CROSSTALK to go to a particular line in a script file and continue execution at that point. In the example above, we used a JUMP command to return to the label HERE. A JUMP command must always be followed by a label name. Note that the JUMP command CAN jump forwards or backwards.

The at-sign (@) has a special meaning when used with the JUMP command. The @ symbol may be used as an "ON-GOTO" condition. The @ tells CROSSTALK "replace the @ with the answer you got from the ASK command". For example, the commands:

LABEL askuser
ASK Enter A,B, or C:
IF -ABC jump ASKUSER
JUMP DO-@

LABEL DO-A
(some commands)

LABEL DO-B
(some more commands)

LABEL DO-C
(still more commands)

will cause CROSSTALK to jump to a certain point in a script file, based on a user's response to a question.

12.8.9

The SKIP command

The SKIP command is used to either skip a certain number of lines in a script file, or to move FORWARD to a label, as in the JUMP command. You can NOT skip past the end of the file, nor can you skip a negative number of lines. The statement:

SKIP 10

will cause CROSSTALK to skip over the next ten lines in the script file.

Note that if you will only be branching FORWARD in a script file, your script file will operate much more quickly if you use SKIP <label name> instead of the JUMP command. As with the JUMP command, you may use the special @ symbol with the SKIP command.

12.8.10

The MESSAGE command

This command is used to print a message for the user to read when executing a script file. Note that if you are at the status screen when the MESSAGE command is executed, CROSSTALK will automatically perform a SCREEN Terminal.

A MESSAGE command must be followed by a line containing a single period to mark the end of the message.

For example, the following commands will switch to the terminal screen, clear it (in case it was already at the terminal screen), and print a message in the window:

**SCREEN T
CLEAR
MESSAGE**

This message will appear on the terminal screen.

. (note that this period marks the end of the message)

You will find that the MESSAGE command, used in conjunction with the ASK and JUMP commands, is ideal for creating menus; you may want to take a look at the SETUP.XTS file to see how this is done.

12.8.11

The RWIND command

The RWIND command "rewinds" the current script file to the beginning, and starts over from the top. It is faster to use RWIND than it is to use a JUMP to a label at the start of a file.

12.8.12

The SBREAK command

The SBREAK command is used to send a BREAK signal under control of a script file. It has the same effect as pressing the BREAK key.

12.8.13

The REPLY command

The REPLY command is used to send a string of text to the host computer. In order to send control characters, simply precede the printable character with a caret (^). If you need to send a carriage return, you may use either ^M or `\r`. As an example, the command:

RE**P**ly ID TCA123 ^A^B^C!

will send the string "ID TCA123 ^A^B^C" to the host, followed by a carriage return.

You will find that the REPLY command used in conjunction with the WAIT command (next section) provides an elegant yet simple method for conversing with a host system; this allows you to perform automatic logons, text retrieval, etc.

Since the REPLY command is intended to be part of a script file, it would not normally be entered as a command from the command line.

12.8.14

The WAIT command

The WAIT command tells CROSSTALK to wait until a certain condition has been met. These conditions are given as arguments to the WAIT command. The conditions are:

Condition

Effect

Echo

Wait for a carriage return, then do the next line in the script file.

Quiet X	Wait until the line is "quiet" (i.e., no characters received) for X tenths of a second, then do the next line in the script file.
Delay X	Wait for a period of X tenths of a second, then do the next line.
Char "x"	Wait until the character "x" is received from the communications line, then execute the next line.
For "x"	Same as "Wait char", above.
Prompt X	Wait for X characters from the communications line, then do the next line in the file.
String "text"	Wait until the string "text" is received from the communications line. Note that case is ignored.
Manual	Suspend script processing until a "DO" command is issued.

Note that any of these WAIT commands may be cancelled without aborting the script file by pressing the SPACE BAR. CROSSTALK will cancel the current WAIT and go on to the next command in the file.

Notice that when the WAIT command is issued in a script file, CROSSTALK will display the WAIT condition on the terminal screen. For instance, when CROSSTALK executes the command:

WAIT STRING "TERMINAL = "

CROSSTALK will print **Waiting for "TERMINAL = "** on your screen. This is especially useful when writing and debugging script files. However, CROSSTALK will also display all of the incoming characters on the screen when executing a script file, thus jumbling the lines on the screen. This can become annoying once you have your script files working, so CROSSTALK provides a command to disable the WAIT prompts — the MAGIC command. The MAGIC command operates like all of the other ON/OFF commands in CROSSTALK. When MAGIC is on, no "wait messages" are displayed. When MAGIC is off, the messages will be displayed on the screen. You can use the MAGIC command in the script files which have been debugged with MAGIC ON as the first command, and MAGIC OFF as the last command.

As mentioned in the previous section, these WAIT conditions allow a great amount of flexibility and power when used together with the REPLY command. An example script file is shown at the end of this chapter.

12.8.15

The WHEN command

WHEN is a special command that allows CROSSTALK to do a command or series of commands WHENever a particular word or phrase is sent to you by a remote computer, thus helping to automate your terminal session even more. WHEN can be especially useful in script files, but can also be entered from the keyboard as a regular command.

The format of the WHEN command is as follows:

WHen "text" command-line

The quotes around the text are required. Once this command is given, CROSSTALK will perform the command or commands specified whenever "text" is received.

For example, The Source often sends "--More--" when more output is forthcoming. You could tell CROSSTALK to always send a carriage return when that happened by giving the command:

WHen "--More--" reply |

Remember that the REPLY command sends characters to the modem as if you had typed them, and that the vertical bar represents a carriage return (corresponding to the RETURN key on your keyboard).

As another example, let's say that you were working with a mainframe that would send "Done" when your job was done. You could give the command:

WHen "Done" alarm

and whenever CROSSTALK received the word "Done", the alarm would sound, alerting you to the fact that your job had been completed.

You can only have one WHEN condition active at a time. If you give a new WHEN command while another is active, the new condition supercedes the old one, and the old one is lost. You may deactivate a WHEN condition without specifying a new one by using "WHen -".

Combining the WHen command with the other CROSSTALK script file control facilities can give you great versatility in automating your communications. For example, the sequence

```
when "Password:" reply BERT | : do
wait manual
```

will cause CROSSTALK to wait for the word "Password:" to appear from the remote system, reply with the word "BERT", and then proceed with the script file processing. This happens as follows:

1. The WHEN command sets the condition to watch for, and script file processing continues.
2. CROSSTALK finds the 'wait manual' command, and pauses in its execution of commands from the file. It is important to note that the WHEN command does NOT cause CROSSTALK to wait for the condition specified — use the WAIT STRING command for this.
3. Eventually, the remote system asks for "Password:" and CROSSTALK activates the WHEN condition.
4. The commands specified on the WHEN condition first send "BERT" to the remote system (the REPLY command), and reactivates the script file (the DO command).

Keep the following conditions in mind when using WHEN:

- * The WHEN condition STAYS ACTIVE until explicitly deactivated. If you want something to happen only the FIRST TIME a string is received, remember to put a WHEN — at the end of the WHEN command line. For example:

```
when "System:" reply SYS10 | :when - : do
```

- * CROSSTALK IGNORES case and all blanks when checking for a WHEN condition. Therefore,

```
when "-More-" reply |
```

works just the same as

```
when " -M O R E-" reply |
```

or

```
when " -- more --" reply |
```

- * As we just mentioned, CROSSTALK is intelligent enough to account for miscellaneous blanks and changes in letter case while checking for WHEN conditions. However, it CANNOT be expected to act properly if a noisy communications line causes garbage to appear at random. Therefore, we do not recommend that you depend on WHEN (or other modem-dependent script file facilities like WAIT) in unattended operations when there is a likelihood of noise on the line. We cannot be held responsible for data lost or connect expenses incurred as a result of an unsuccessful script file.
- * If a WHEN condition occurs, and a script file containing a WAIT statement is waiting for a particular word or character, the WHEN condition takes precedence. After the WHEN condition has been met, and the WHEN commands executed, the program will continue with the command line that was interrupted.

12.9

A typical script file

A properly written script file can add a great deal of flexibility to your CROSSTALK system. You can create script files to do any number of things. For example, if you frequently call another CROSSTALK system and exchange a group of files with the other system, you can place the XMIT and RQUEST commands in a script file.

A more common use of script files is to call into a timesharing system and automatically enter an account number and password to the host system.

Script files must have the file type ".XTS". The following example shows a combination of a command file and a script file which will call another computer system, log in, enter a password, and send some commands to the other computer system. Note that any line in a script file which begins with a semicolon is assumed to be a comment, and is not processed as a command. Note that blank lines do not affect the operation of a script file, and may be used to improve readability.

File EXAMPLE.XTK (command file)

Name	Example command file
Number	1-404-555-1212
SPeed	1200
Data	8
PArity	None
Mode	Call

(Additional
commands)

GO (dials number)

(after connecting, CROSSTALK searches the disk for EXAMPLE.XTS. If the file exists, CROSSTALK does the commands in the script file.)

File EXAMPLE.XTS (script file)

```
; Wait until the line has been "quiet" for one second  
Wait quiet 10  
  
; Send LOGIN command to host computer  
Reply LOGIN!  
  
; Wait for "Enter account number:" prompt from host  
Wait char ":"  
  
; Reply with our account number  
Reply ABC 123!  
  
; Wait for "password?" prompt  
Wait char "?"  
  
; Reply with our password  
Reply ^A^B^C^D!  
  
; Wait for system's "->" prompt  
Wait string "->"  
  
; Ask for our mail, and capture it to memory  
Reply MAIL READ!  
  
; When we get "-more-", send a carriage return  
When "-more-" reply:  
Capture ON  
; End of this script.
```

Each line of this particular script file has a comment before it. The comments are entirely optional, and are there only to aid your understanding of what the script file is doing. Note that the final line in the script file is "Capture ON". You may include any valid CROSSTALK command in a script file.

In this example, we showed the command and script files as two separate files, but you can just as well add the script commands to the end of the command file.

In most cases, you'll probably want to have the chores of entering your log-in and password done as part of a script file, rather than as part of a command file. In some cases, you may want to have several different script files for the same system. For example, if you use The Source frequently, you might want to create one script file to read your mail, another to read the POST messages, another to check your favorite stock reports, and another to sign you off.

You can create as many script files as you wish. The only limit is your imagination and the amount of storage available on your disks. If you have several script files, you may enter the command "DO" to have CROSSTALK show you a menu of all the available script files.

12.10

Example script files included with CROSSTALK

Your CROSSTALK distribution disk contains several script files. These files are provided as examples, and may be changed to suit your own needs.

Two files, SETUP and NEWUSER, are provided. SETUP allows you to change CROSSTALK's default parameter settings through a series of questions and answers. NEWUSER guides a new user through the process of making a call, again through a series of questions and answers.

While these script files are provided as examples, they are entirely functional script files. Both of these files illustrate how a script file may be used to provide a customized "front end" to CROSSTALK.

12.11

Differences between Command and Script files

Although any of CROSSTALK's commands may appear in either command or script files, you should keep a few things in mind when editing command and script files.

First, there are some very subtle differences between the DO and the LOad commands. As you will remember, the DO command may be aborted by the user or by a WAIT manual command, and re-started with the DO command with no argument; if a LOad command is terminated, it cannot just "pick up where it left off." Also, some commands will operate differently depending upon the type of file they are in. For instance, the command "NA" with no argument will simply clear the name field in a command file, whereas in a script file, CROSSTALK will prompt you for a name upon the command's execution.

Second, when you have LOADED a file, you will notice that the filename appears on the status screen as "LOADED." If you place the "scripts" at the end of the command file (in lieu of a script file), they will be executed after the GO command takes place. If you then disconnect (or did not connect the first time you dialed) and need to redial, you will notice that your "scripts" are not re-executed, since CROSSTALK has already finished the command file. If you had used a corresponding script file (i.e.- SRC.XTS corresponds to SRC.XTK), the script file would automatically be re-executed upon connection.

Last, if you need to change any of CROSSTALK's parameters (SPeed, DAta bits, INfilter, etc.) permanently, you will normally use CROSSTALK's SAVE command to overwrite the existing command file for the system you are calling. Note that if you put "scripts" at the end of the command file in lieu of using a corresponding script file, your scripts will be overwritten.

This section of the manual contains an alphabetical listing of all of CROSSTALK's commands. Each command is listed with a full explanation of what each command does, and what the possible options for each command are. The first line of each description indicates how the command may be used; either as a LOCAL command (from a call mode system), as a REMOTE command (on an answer mode system), as either REMOTE or LOCAL, or in a SCRIPT file only.

13.1

The ABORT command

SCRIPT

The ABort command is used to cancel execution of a script file. It is normally used when an error has been detected. See chapter 12.

13.2

The ACCEPT command

LOCAL

The ACcept command allows you to set the type of access a caller has when calling into an answering CROSSTALK system. By setting the access level, you can protect your system from unauthorized tampering.

The options to the ACcept command are:

Option	Action
Nothing	Prohibit caller from writing to any files
Appends	Allows caller to append captured data to an existing file, but not to create any new files.
Creates	Allows caller to append data to or create new files, but not to overwrite existing files.
Everything	Allows caller full access to system.

13.3

The ALARM command

LOCAL/REMOTE/SCRIPT

The ALarm command is used to sound the speaker on your terminal 3 times. This command may be used to alert the operator in a script file, or in a WHEN command.

13.4

The ANSWERBACK command

LOCAL

The ANsback command tells CROSSTALK whether or not to respond to a received ^E character. Many computer networks utilize this feature as a means of identifying a caller. Answering CROSSTALK and TRANSPORTER systems utilize the feature to request your password. If ANsback is ON, a received ^E will cause the contents of your F4 key (see FK command) to be sent. If ANsback is OFF, the ^E will be ignored.

13.5

The ASK command

SCRIPT

ASk is used to prompt the operator for information. It is normally used in script files, in conjunction with the MESSAGE command. See section 12.8.3 for examples.

13.6

The ATTENTION command

LOCAL

The ATTENTION command selects the key to be used for the "ATTENTION" key. The attention key is the key used to display the "Command?" prompt when you are in terminal mode. Note that this key can not be sent to the remote computer, since typing it causes the program to immediately enter the command mode. The default key used for attention is the ^A key. If the system you are calling requires you to send an ^A, you will have to select another key for the attention key.

There are several different ways to set the ATTENTION key. The chart below shows each of the possible methods.

Command	Effect
AT	CROSSTALK will ask you to press the key to be used for the ATTENTION key. You may NOT select the numbers 0-9 or the letters A-Z as the attention key.
AT 1b	Sets the attention key to the key with a hex value of 1b (in this case, the ESCAPE key).

Note that CROSSTALK will not allow you to set the ATTENTION and SWITCH keys to the same key setting.

13.7

The BKSIZE command

LOCAL/REMOTE

The BKSIZE command sets the size of the data block (in 256 byte increments) sent during protocol transfers. It is normally set to 1. If you are "direct connecting" two computers together without modems, you may wish to select a larger block size (up to 10). Using a larger block size marginally speeds up protocol transfers. We recommend the default setting of 1 when transferring files via modems.

13.8

The BLANKEX command

LOCAL

The BLANKEX command tells CROSSTALK how to handle "blank" lines when sending files to another computer with the SEnd command. If BLANKEX is ON, CROSSTALK will convert blank lines into lines consisting of one space. This may be useful when sending prepared text containing empty lines into a host computer system which assumes a blank line to mean "end of text".

13.9

The BREAK command

LOCAL

The BREAK command is used to select the key to be used for sending a BREAK. The BREAK key is normally assigned to the ^B key, but may be changed if desired.

There are several different ways to set the BREAK key. The chart below shows each of the possible methods.

<u>Command</u>	<u>Notes</u>
BR	CROSSTALK will ask you to press the key to be used for the BREAK key. You may NOT select the numbers 0-9 or the letters A-Z as the attention key.
BR 02	Sets the BREAK key to the key with a hex value of 02 (in this case, the ^B key).

13.10	The BYE command
-------	------------------------

LOCAL/REMOTE

The BYe command hangs up the phone line, and disconnects the current call. Use this command when you wish to hang up and make another call without exiting CROSSTALK.

13.11	The CAPTURE command
-------	----------------------------

LOCAL/REMOTE

The CApture command has two main modes: Capture-to-disk, and capture-to-memory. To capture directly to disk, enter "CA (filename)". If the file already exists, CROSSTALK will ask if you wish to over-write the file, or append the new data to the end of the old file. The command "CA -" is used to end capture-to-disk, and closes the capture file. You may use "CA /" to toggle capture to disk on and off.

To begin capturing data to memory, enter "CA +". After capturing your information, you will need to issue the WRITE command to save the data to a disk file. Note that the WRITE command does not turn capture off. It only WRITES the captured data to the specified file. While capturing data, the command "CA /" will toggle capture on and off, allowing you to selectively save incoming data. The command "CA -" turns capture off.

If you wish to capture some data which is already on your screen, and capture is OFF, you may use the "RETRO-CAPTURE" feature. Enter "CA <xx>", where xx is the number of lines you wish to "grab" off the screen. CROSSTALK will retroactively capture the data, add it to the capture buffer, and leave CApture ON. RETRO-CAPTURE may only be used when CApture is OFF.

13.12

The CLEAR command

LOCAL/SCRIPT

The CClear command is used to clear the terminal screen. Although it is normally used in script files, the CClear command may be used to clear the terminal screen when online.

13.13

The COMMAND command

LOCAL/REMOTE

The COMMAND command sets the COmmand key. The COmmand character is the key a caller would type if he had called into your system and wanted to enter commands to CROSSTALK. We recommend that you leave this key set to the default setting of ^C.

There are several different ways to set the COMMAND key. The chart below shows each of the possible methods.

Command	Notes
CO	CROSSTALK will ask you to press the key to be used for the COMMAND key. You may NOT select the numbers 0-9 or the letters A-Z as the command key.
CO 03	Sets the COMMAND key to the key with a hex value of 03 (in this case, the ^C key).

13.14

The CSTATUS command

LOCAL/REMOTE

The CStatus command has two functions. The command "CStatus" alone displays the status of the capture buffer — how many characters have been captured (in Kilobytes) and how much space is left (again, in Kilobytes) for capturing additional data.

The command "CS fred" will search the capture buffer for all occurrences of the text "fred". If the word "fred" is located in the capture buffer, the program will display the contents of the capture buffer surrounding the located word. After displaying the word, the program will offer to continue searching through the buffer.

Note that the CS command may be used with either method of capture active. Remember that with capture-to-disk active, the search will only take place on the characters in the capture buffer and not on the characters that may have previously been written to disk.

13.15

The CWAIT command

LOCAL

The CWait command tells CROSSTALK how to wait between characters when transmitting files with the SEnd command. Use this command when you are sending text to a computer system which can not accept text at full speed.

The CWAIT command has several options. The chart below illustrates the effect of each:

Option	Effect
NONE	Do not wait at all between characters.
Echo	Wait for each character to be echoed by the host before sending the next character.
Delay XX	Delay xx thousandths of a second, then send the next character.

Note that CWAIT can be used in combination with LWAIT to allow you to transmit text to virtually any dial-up computer system.

13.16

The DATA command

LOCAL

The DAta command sets the number of data bits used by CROSSTALK. The default value is 8. CROSSTALK automatically switches to 8 bits whenever a protocol transfer is performed, even if 7 bits had previously been selected. Some dial-up computer systems require 7 data bits.

13.17

The DEBUG command

LOCAL

The DEbug command allows you to select one of three "debug" modes. The debug modes allow you to display incoming control-characters graphically on your screen. This is often

useful when attempting to identify non-printing control characters transmitted by some dial-up computer systems.

The chart below shows how each of the debug modes operates:

Mode	Action
NONE	Incoming control characters are not altered.
HEX	All incoming characters are shown as hex numbers. For example, an incoming ^Z will be shown as [1A].
CHAR	Incoming control-characters are shown in the form [^X].

13.18

The DIR Command

LOCAL/REMOTE

The DIR command is used to view the disk directory, much like the CP/M "DIR" command. Like the CP/M DIR command, you may provide an optional file name, and the program will display only those files given in the file name. For example, the command "DIR *.ASM" will show only the .ASM files.

There are two options to the DIR command which are not found in the CP/M DIR command. These are the /S and /T options. The command "DIR /S" will show the size of each file, and "DIR /T" will show the amount of time required to send each file at the current baud rate.

13.19

The DO command

LOCAL/REMOTE

The DO command is used to begin or continue execution of a script file. The command "DO" by itself will display a menu of available script files. The command "DO (filename)" will cause the script file named "filename.xts" to be performed. See section 12.8.5 for further information.

13.20

The DPREFIX command

LOCAL

The DPREFIX command tells CROSSTALK how to dial your modem. When you instruct CROSSTALK to dial, the program sends the following data to the modem:

DPREFIX string | NUMBER string | DSUFFIX string

By changing the DPREFIX string, CROSSTALK can dial a number of different types of auto-dial modems. There are several characters which have special meaning when placed in a DPREFIX string. These are:

Character	Effect
	Places a carriage return (ENTER key) in the string.
~	Causes CROSSTALK to delay one second when dialing.
^	The character following the ^ character is sent as a control character. For example, "^T" would place the ^T character in the dial prefix.

13.21

The DRIVE command

LOCAL/REMOTE

The DRive command has two functions. Entering "DRive" alone will "re-log" the drives on your system. This will allow you to change diskettes while in CROSSTALK without CP/M making the disk READ ONLY. Entering "DR" followed by a drive name (i.e.-DR B:) will change the "default" drive to the specified drive and "re-log" all the drives on the system.

13.22

The DSUFFIX command

LOCAL

The DSUFFIX command sets the dialing suffix string. (see explanation of the DPREFIX command). The same special characters apply to DSUFFIX as apply to DPREFIX.

13.23

The DUPLEX command

LOCAL

The DUplex command sets duplex to full (local echo OFF) or half (local echo ON). Most dial-up computer systems require that your system be set for FULL duplex.

Note that when you select ANSWER mode, CROSSTALK automatically selects HALF duplex. When you select CALL mode, the program automatically selects FULL duplex. See the glossary for an explanation of "duplex".

13.24

The ERASE command

LOCAL/REMOTE (limited)

The ERase command has two functions: The command "ER" alone will erase the contents of the capture buffer. The command "ER FILE.NAM" will erase the file "FILE.NAM". You are always asked to confirm an erase before the operation actually takes place. The command "ER *.*" will allow you to selectively erase any or all files on a disk.

Note that an answer-mode caller may only use the ERASE command to erase the contents of the capture buffer. He may NOT erase files from the disk.

13.25

The FILTER Command

LOCAL

The FIIter command is used to display and change the table of allowable control characters. Enter the command "FILTER", and CROSSTALK will display the table of all of the possible control characters.

Any characters listed as "Filtered" are discarded by CROSSTALK. The characters listed as "Passed" are treated as any other character by CROSSTALK. You may change whether a control character is "filtered" or "passed" by simply typing that character.

13.26

The FKEYS command

LOCAL

The FKeys command is used to set and display the contents of the programmable trip keys. The command "FK" will show the settings of the trip keys.

To set a trip key, enter "FK n string" where n is the number of the key to be set, and "string" is the string of text assigned to that key. Several characters have special meaning when used in trip keys. The vertical bar (|) places a carriage return in the string. Any key which begins with an at-sign (@) will be taken as a command to CROSSTALK.

For Example, the command "FK 1 @PR / | " will set key F1 to "@ PR | ". When the F1 key is "tripped", CROSSTALK will toggle the printer on or off.

13.27

The GO command

LOCAL

The function of the GO command depends on which mode the program is in. In CALL mode, The GO command tells CROSSTALK to dial a number and establish a connection. You may instruct the program to re-dial automatically if a connection is not established on the first try. The command "GO R30" would tell CROSSTALK to re-dial every 30 seconds until a connection is established, or until the RDIALS limit is reached.

The command "GO LOCAL" tells CROSSTALK that there is no modem connected to the system. Use this option when using CROSSTALK to transfer files between two computers "hard-wired" together. When the local option is used, the carrier detect signal on the RS-232 interface is ignored and the word "local" is shown on the status line.

In ANSWER mode, the GO command tells CROSSTALK to begin waiting for incoming calls.

13.28

The HELP command

LOCAL/REMOTE

The HELP command provides help on all of CROSSTALK's commands. Entering "HElp" with no option displays a list of all of CROSSTALK's commands. If you need help with a command, type "HELP XX", where "XX" is the command you need help with. If CROSSTALK asks you a question you don't understand, enter a "?", and the help for that command will be displayed.

13.29

The IF command

SCRIPT

The IF command is used in script files to make a decision based on the results of an ASk command. It allows "branching" to different locations in a script file, based on the user's response to a question. See section 12.8.7 for more details.

13.30

The INFILTER command

LOCAL

The INfilter command tells CROSSTALK to discard incoming control characters when in terminal mode. Some host computer systems may send control-characters which are meaningless to your computer. Turning INfilter ON will discard these characters. If you see extraneous "junk" characters on the terminal screen, try turning INfilter ON. INfilter also "strips" the top bit off of 8-bit characters. With INFILTER OFF, these characters may appear as "graphics" characters. Turning INFILTER ON will display these characters correctly.

13.31

The JUMP command

SCRIPT

The JUmp command is used to branch control to a particular location in a script file. See section 12.8.8 for more information.

13.32

The KEY command

LOCAL

The KEY command is used to set CROSSTALK's "trip" key. The default KEY is ^L. You will use the trip KEY to send the contents of a CROSSTALK function key by pressing the current KEY followed by 0-9. This KEY is also used to execute one of the special trip functions: C (capture toggle), P (printer toggle) or T (type the capture buffer).

You can send the current KEY to the host system by pressing it twice.

13.33

The LABEL command

SCRIPT

The LAbel command is used to assign a label name to a particular spot in a script file, so that it may be referred to by the JUMP command. See section 12.8.6 for more information.

13.34

The LFAUTO command

LOCAL

When LFauto is on, CROSSTALK will print a line feed on the screen after each received carriage return. This is useful when communicating with systems which do not send line feeds. Conversely, when LFAUTO is ON, CROSSTALK will print a carriage return each time a line feed is received. This is useful when communicating with systems which send line feeds without carriage returns.

13.35

The LIST command

REMOTE

The LIST command is used to display CROSSTALK's parameter settings to a remote system. This command has no effect when issued locally.

13.36

The LOAD command

LOCAL

The LOad command is used to load CROSSTALK command files. Command files are files containing commands for CROSSTALK to perform. The command "LO" will display all of the available command files, and allow you to choose one to load. The command "LO FRED" will load in "FRED.XTK", and perform all of the commands contained in that file.

13.37

The LWAIT command

LOCAL

The LWait command tells CROSSTALK how to wait between lines when transmitting text files with the SEnd command. There are several options to the LWAIT command. The table below shows how each option works:

Option	Effect
NONE	Each line is sent with no waiting.
Prompt X	Send a line of text, wait for a prompt of X characters, then send the next line.

Char "X"	Send a line of text, wait until the character "X" is received, then send the next line.
Delay X	Wait X tenths of seconds between lines.
Manual	Prompts the user to press the space bar between each line.
Echo	Wait until a carriage return is echoed from the host system.
Learn	Send the first line of text, count incoming characters, wait until the user presses the space bar, then send each successive line with a wait for a prompt of "count" characters.

13.38**The MAGIC command****SCRIPT**

The MAGIC command is used to inhibit the display of the script file WAIT messages when a script file is active. Use "MAGIC ON" after debugging a script to make script files work like magic.

13.39**The MESSAGE command****SCRIPT**

The MEssage command is used in script files to display a message on the screen. This is particularly useful used in conjunction with the ASK command. See section 12.8.10 for detailed information.

13.40**The MODE command****LOCAL**

The MOde command tells your CROSSTALK program whether to Make or Answer calls. If you wish to MAKE a call, select "CALL" mode. To ANSWER calls, use "ANSWER" mode. CROSSTALK automatically selects HALF Duplex (see DUplex command) when the answer mode is selected, and selects FULL duplex when call mode is selected.

13.41

The NAME command

LOCAL

The NAme command is used to set the name of a called location. Use of this command is optional. If a NAme has been given to CROSSTALK, the name will be sent as a "greeting" message whenever a remote terminal calls into CROSSTALK in the answer mode.

13.42

The NO command

LOCAL/REMOTE

This command is used internally between two CROSSTALK systems. It is sent from an answer system to a call system at the end of a file transfer, to tell the calling system that there are no more files to be transmitted.

This command is not normally entered as a command by the user.

13.43

The NUMBER command

LOCAL

The NUmber command sets the phone number to be dialed. The number may be up to 40 characters long.

13.44

The OUTFILTER command

LOCAL

The OUtfilter command instructs CROSSTALK to remove line feeds when sending text files with the SEnd command. If OUtfilter is ON, line feed characters are discarded.

13.45

The PARITY command

LOCAL

The PArity command is used to set the parity bit to the appropriate value. Some systems require that parity be set correctly, other systems ignore parity entirely. The possible settings for the PARITY command are None, Even, and Odd.

13.46

The PRINTER command

LOCAL/REMOTE

The PRinter command tells CROSSTALK to echo all data displayed on the terminal screen to the printer. Turning PRinter ON does not turn the terminal video display off. The command "PR /" will toggle the printer to the opposite state of what it currently is.

13.47

The PWORD command

LOCAL

The PWord command is used to set CROSSTALK's answer password. The password may be up to 12 characters long. If you have set a password, then any persons calling into your system will be required to enter the password before they can access your system. Setting PWord to blank disables the password feature.

13.48

The QUIT command

LOCAL

The QUIT command tells CROSSTALK to immediately disconnect and return to CP/M. If any information is in the capture buffer, CROSSTALK will ask if you wish to save the information before exiting.

13.49

The RCVE command

LOCAL/REMOTE

This command is used internally by CROSSTALK, during file transfers. It is not normally entered as a command by the user.

13.50

The RDIALS command

LOCAL

The RDials command is used to set the maximum number of times CROSSTALK will redial a number when told to GO. The default is 10 redials, but may be from 0 to 65535.

13.51

The REPLY command

SCRIPT

The REPLY command is used in a script file to send a message to the other computer system. See section 12.8.13 for more information.

13.52

The RQUEST command

LOCAL

The RQuest command is used to request a transfer of files from another CROSSTALK system. This command should only be issued from a CALL mode system when calling into an ANSWER mode system.

You may specify an optional target or destination drive name when issuing the RQUEST command. Some examples of this are shown below:

Command	Result
RQ *.ASM	Request all files with the type .ASM
RQ B:FRED	Request the file FRED from drive B:, place the file on the default drive.
RQ B:BERT C:	Request the file BERT from drive B:, place the file on drive C:.

13.53

The RWIND command

SCRIPT

The RWind command is used to cancel and re-start a script file. The current script file is terminated and re-started from the beginning, just as if a new DO command has been issued. See section 12.8.11 for more information.

13.54

The RXMODEM command

LOCAL/REMOTE

The RXMODEM command is used to tell CROSSTALK to receive a file using the XMODEM protocol. This command does NOT make the host system SEND you a file using the

XMODEM protocol; you will need to issue the proper commands to the host you are calling to have them send a file with XMODEM in order for CROSSTALK to receive it.

13.55**The SAVE command**

LOCAL

The SAve command saves a CROSSTALK command file for future use. All of CROSSTALK's operating parameters, including all function key definitions, are saved in the command file.

13.56**The SBREAK command**

SCRIPT

The SBreak command is used to send a BREAK signal to the modem from within a script file. See section 12.8.12.

13.57**The SCREEN command**

SCRIPT

The SCREEN command is used to change screens under control of a script file. Use "SCreen S" to go to the status screen, "SCreen T" to go to the terminal screen.

13.58**The SEND command**

LOCAL

Use the SEnd command to send a text file from your disk to another computer system. This command is most often used when communicating with another computer which is not running CROSSTALK. If the system you are communicating with is running CROSSTALK, use the "XMit" command instead.

Several other commands are used to modify the method the SEND command uses to send files. For information on all of the commands affecting the operation of the SEnd command, see the LWait, CWait, BLankex, and OUtfilter commands.

13.59

The SKIP command

SCRIPT

The SKip command is used to skip over a number of lines in a script file, or forward to a specific label much like the JUMP command. See section 12.8.9 for more information.

13.60

The SPEED command

LOCAL

The SPeed command selects the speed at which your computer will communicate over the phone line. Only the first digit of the desired speed must be entered. Note that 1200 baud is 1; 110 baud is 0.

13.61

The STOP command

LOCAL

The STop command sets the number of stop bits used to make up the data word. Most systems use 1 stop bit at speeds of 300 baud, and above, and 2 stop bits at 110 baud.

13.62

The SWITCH command

LOCAL

The SWitch command selects the key to be used for switching between the terminal and status screens. This key is normally the ESC key but can be changed to another key if desired.

There are several ways to set the SWITCH key. The table below shows each of the possible methods:

Command	Notes
SW	CROSSTALK will ask you to press the key to be used for the SWITCH key. You may NOT select the numbers 0-9 or the letters A-Z as the SWITCH key.
SW 02	Sets the SWITCH key to the key with a hex value of 02 (in this case, the ^B key).

13.63

The TABEX command

LOCAL

The TABEX command is used to turn tab expansion on and off. When TABEX is on, transmitted tab characters are sent as spaces to the next tab position. This is useful when sending files to systems which do not interpret tabs.

Note that the TABEX command only affects the SEND command. It has no effect on normal "terminal mode" operation.

13.64

The TSTATUS command

LOCAL

The TSTATUS command is used to toggle the appearance of CROSSTALK's terminal screen status line. With TSTATUS ON, CROSSTALK will display the current SWITCH and ATTENTION keys, the state of the PRINTER and the status of the CAPTURE buffer. With TSTATUS OFF, the status line is not displayed.

Note that CROSSTALK may drop characters on some systems with TSTATUS ON. If this is the case, you will need to have TSTATUS OFF.

13.65

The TURNAROUND command

LOCAL

Some computer systems use a key other than RETURN to mean "end of line". The TURNAROUND command tells CROSSTALK which character to send when the RETURN key is pressed.

There are several ways to set the TURNAROUND key. The table below shows each of the possible methods:

Command	Notes
TU	CROSSTALK will ask you to press the key to be used for the TURNAROUND key.
TU 03	Sets the TURNAROUND key to the key with a hex value of 3 (in this case, the ^C key).

13.66

The TYPE command

LOCAL/REMOTE

The TYPE command has two functions: it may be used to review the contents of the capture buffer, or it may be used to view the contents of a disk file.

When given with no name, the TType command types the contents of the capture buffer to the screen. The command "TType (filename)" will type the contents of the file on the screen.

13.67

The UONLY command

LOCAL

The Uonly command is used to set upper case mode on and off. When UOnly is on, all lower case letters encountered during a SEnd will be converted to upper case. Received characters are unaffected.

13.68

The USER command

LOCAL/REMOTE

The USER command is used to change or display the current CP/M user number. Typing "UUser 3", for example, will set your CP/M user number to 3. Typing "UUser" will display the current user. Remember that any disk activity (e.g.-DIR, TYPE, etc.) will take place under the current user. Also, CROSSTALK will return you to your original DRIVE and USER when exiting CROSSTALK.

Note that you may NOT change users when capturing information directly to a file. Note also that your CROSSTALK HELP files must be under the current user in order to get help.

13.69

The VERSION command

LOCAL/REMOTE

The VERSION command will return CROSSTALK's current version and the DATE it was last modified.

13.70

The WAIT command

LOCAL/REMOTE/SCRIPT

The WAit command causes a script file to be paused until the condition specified by the WAIT command has been met. See section 12.8.14 for more details.

13.71

The WHEN command

LOCAL/SCRIPT

The WHEN command is used to tell CROSSTALK to execute a command(s) when the specified string is received from the modem. See section 12.8.15 for detailed information.

13.72

The WRITE command

LOCAL/REMOTE

The WRite command stores captured data into a file of your choice. If a disk error occurs, or if the capture buffer is empty, an error message will be printed.

Use the WRITE command when you have captured data using the capture-to-memory command, and you wish to save the captured data on your disk. It is not necessary to use the WRITE command if you are using capture-to-disk.

13.73

The XCPM command

LOCAL

The XCPM command tells CROSSTALK to exit to CP/M without hanging up the phone line. You may run any other program and return to CROSSTALK, and your connection will still be intact. If any information is in the capture buffer, CROSSTALK will ask if you wish to save the information before exiting.

13.74

The XMIT command

LOCAL

The XMit command transmits files to another CROSSTALK system. "Wildcard" file names are allowed. An optional target drive name may be given after the file name.

Note that the XMIT command should only be used when calling into an answer-mode CROSSTALK or TRANSPORTER system.

The chart below shows some example XMIT command:

Command	Result
XMIT *.ASM	Sends all the files with the type .ASM to the other system.
XMIT BERT	Sends the file BERT from the default drive to the other system.
XMIT B:FRED C:	Sends the file B:FRED to drive C: on the other system.

13.75

The XXMODEM command

LOCAL/REMOTE

The XXMODEM command is used to tell CROSSTALK to send a file using the XMODEM protocol. This command does NOT ready the host system to receive a file using the XMODEM protocol; you will need to issue to proper commands to the host before issuing the XXMODEM command to CROSSTALK.

This section of the manual gives some how-to-do-it examples of typical uses of CROSSTALK. Please remember that these are only examples. Every situation is different, and what is shown here may not work for every system in the world. Any names, phone numbers, account numbers or passwords shown in this section are fictitious. Any similarity to any real names, phone numbers, account numbers or passwords is purely coincidental.

14.1

Calling The Source

Judging from the feedback we get from users of CROSSTALK, a lot of you are using CROSSTALK to call The Source. CROSSTALK can help you cut down on connect time by making your call to The Source more efficient. In this example, we will show a command and script file which will call The Source through TELNET, log you in, and send the MAIL READ command to The Source. In addition, we'll set some function keys to some common Source commands.

Once these command and script files have been created, all you have to do to call The Source is enter **XTALK SOURCE** from your system prompt. CROSSTALK will make the call, wait and re-dial if necessary, and log you in.

File SOURCE.XTK**Contents of file:****Comments:**

NAme	The Source	This is who we're calling
NUmber	555-1212	and this is the phone number
ANswback	Off	
ATten	^A	Set our normal AT key
BReak	^B	and our normal BREAK key
SWitch	ESC	and our normal SWITCH key
CWait	None	Don't need CWAIT
LWait	Delay 1	LWAIT 1 works well
DEbug	Off	No DEBUG
DPrefix	ATDT	Dial with tones
DSuffix	!	
INfilter	On	Filter junk chars. out
LFauto	Off	Don't need extra LF's

MOde	Call	We are making a call
TUrnarnd	^M	Send CR when we press CR
CApture	Off	
DAta	8	8 bit data
DUplex	Full	Full duplex
OUtfiltr	On	Filter outgoing LF's
PArity	None	
PRinter	Off	Start with printer OFF
SPeed	1200	1200 baud
STop	1	1 stop bit
TAbox	On	Expand tabs
BLankex	On	Expand blank lines
UConly	Off	Lower case is ok.
FK 1	Post read IBM !	Set these function keys
FK 2	Post read BULLETIN-BOARD !	
FK 3	UPI !	
FK 4	n !	
FK 5	@CA / !	F5 toggles capture on/off
FK 6	@PR / !	F6 does same for printer
FK 7	@TY !	F7 will review the buffer
FK 8		
FK 9		
FK 10		
GO R30/30		Re-dial until we connect

File SOURCE.XTS**Contents of file:**

Wait delay 5
 Reply !
 Wait delay 10
 Reply !
 Wait string "TERMINAL = "
 Reply D1 !
 Wait for "@"
 Reply C 30147 !
 Wait char ">"
 Reply id abc123 ^A^B^C^ !
 Wait char ">"
 Reply mail read !

Comments:

wait for 1/2 second
 send a carriage return
 wait for a second
 send another carriage return
 wait for TERMINAL = prompt
 send terminal type
 wait for TELENET's @ prompt
 and ask for the Source
 wait for "log in >" prompt
 send our log-in & password
 wait for Source's ">" prompt
 and ask for our mail

14.2

How to capture data

This example will show you how to capture data from a host computer system, and save the data on your disk. This example will assume that you have already made a call and established communications with the host system.

Once you have made your call and established communications with another computer, any data sent to your computer from the host system can be captured and saved on your disk. In this example, we will show you how to save your mail from The Source.

You type:**What happens:****MAIL READ**

Source MAIL system signs on

^A CA +

CROSSTALK begins capturing

(answer MAIL system prompts)

Source MAIL is displayed (end of mail)

^A WRITE B:MAIL

CROSSTALK saves captured data in file B:MAIL.

^A CA off -

Turn capture off.

14.3

How to send a file with the SEND command

This example will show you how to send a text file to a host computer system with the SEND command. This example will again assume that you are already connected to the host computer. The example will show you how to send a pre-edited text file into The Source MAIL system.

You type:**What happens:****MAIL SEND TCA123**

Source MAIL system signs on

(answer MAIL system prompts)

Source replies "Enter Text:"

^A LWAIT LEARN

CROSSTALK sets up "learn" mode

^A SEND "filename"

CROSSTALK sends first line of text, asks you to press the space bar

[SPACE BAR]

CROSSTALK sends remainder of text file to The Source.

.S

MAIL program sends mail.

14.4

Transferring files to another CROSSTALK system

This example shows how to transfer files with another CROSSTALK or TRANSPORTER system. We will assume that you are the caller, and the other system answered. We will also assume that you have already made the call and established communications with the other system.

In the example, we will send the file "FRED.TXT" to the other system, and request the file "ERNIE.BAS" from the other system.

You type:	What happens:
[ESC] XM FRED.TXT	CROSSTALK begins transfer of file FRED.TXT to other CROSSTALK system.
[ESC] RQ ERNIE.BAS	CROSSTALK begins transfer of file ERNIE.BAS from the other CROSSTALK system.

Word	Definition
Bit	A bit is the single unit of a data word. Normally, 7 data bits are used to represent the 96 commonly used alphanumeric characters used by most computer systems.
Baud Rate	The speed at which a computer communicates with another device. Baud is a technical term for "bits per second". If you are transmitting a 10 bit data word at 300 baud, you are sending 30 characters per second.
CRC check	A method for verifying the integrity of data. The CRC (cyclic redundancy check) is computed by adding together the numeric total of the value of each character sent. This sum is then compared to the total sum of the characters received. If the totals do not match, an error has occurred, and the block of data is automatically re-transmitted.
Data word	The combination of start bits, data bits, parity bits, and stop bits is called the data word. One data word is used to represent each character of transmitted data.
Duplex	Duplex is a term describing the method of communication between two computers. In full duplex, characters sent from point a to point b are echoed back from point b before they appear on point a's terminal. In Half duplex, characters sent from point a to point b are sent directly to point a's terminal, as well as to point b. Point b does not echo the characters back to point a.
Modem	An acronym for MOdulator/DEModulator, a modem is a device which converts digital data into audio tones suitable for transmission over regular telephone lines.
Parity	The parity bit is an additional data bit, the purpose of which is to provide a simple means of error checking. When parity is even, the total sum of all the bits (including the parity bit) will be an even number. When parity is odd, the total will be an odd number.

Word	Description
Protocol	A pre-defined method of exchanging data between computers. The protocol defines the order and method of how the data and CRC information are exchanged.
Start bit	A start bit is used to signify the beginning of a data word.
Stop bits	The bits used to signify the end of a data word. Normally, one stop bit is used at 300 baud and above, and two stop bits are used at 100 baud.
Word length	The word length is the numeric total of the number of start, data, parity, and stop bits used to make up the data word.

Appendix B**Guide For CROSSTALK 2.0 Users**

The following table lists those commands which are either new or different in release 3.0 of CROSSTALK.

New command	Old command	Difference
ACCEPT		New feature
ANSWERBACK		New feature
BKSIZE	BLOCK	New name; command "BL" now has different meaning.
BLANKEX		New feature
BREAK		New feature
CLEAR		New feature
CSTATUS		New feature
CWAIT	FLOW	Replaces "FLOW CHAR" mode
DATA		New feature
DEBUG	DEBUG	Several new options available
DIR	DIR	" "
DO		New feature
DPREFIX		New feature
DRIVE		New feature
DSUFFIX		New feature
ERASE		New feature
FILTER		New feature
FKEYS	L1-L4	FKEYS 0-9 now programmable
GO		New feature
INFILTER		New feature

KEY	TRIP	New name, same action
LWAIT	FLOW	Replaces "FLOW LINE" mode.
MAGIC		New feature
MODE	MODE	Originate is now CALL mode, old "MODE O" command still accepted.
OUTFILTER	SCREEN	New name
PARITY		New feature
PWORD		New feature
RDIALS		New feature
RXMODEM		New feature
SEND	READ	New name - same operation
SPEED		New feature
STOP		New feature
SWITCH		New feature
TSTATUS		New feature
TURNARND		New feature
TYPE	TYPE	TYPE command may now be used to review disk files.
USER		New feature
WAIT	WAIT	WAIT now has a completely different meaning, old WAIT command replaced by LWAIT and CWAIT commands.
WHEN		New feature
XXMODEM		New feature

Appendix C

Connecting Two Systems Through Cables

Many of our customers use CROSSTALK to wire two systems together in order to transfer files between the two computers. Here are a few tips which will make this process much easier.

First, before you can transfer files between two systems, you must be able to communicate between the two systems. This may sound elementary, but many people overlook this fact when trying to transfer files.

In order to communicate properly, your cable between the two systems MUST be connected correctly. Unfortunately, there are so many different types of systems on the market that we can not even begin to tell you exactly what type of cable you'll need to connect a brand X to a brand Y. A list of all the possible cable combinations would be as large as the rest of this manual, and would be out of date by the time it was printed. The general rule is this: All computers have one of two types of communications interface, either a DCE port or a DTE port.

If you are connecting two computers with DISSIMILAR port types, you'll need a straight cable; this is the "easy" one. If BOTH computers have either a DTE or DCE port, you need one of three things: either A) a "null" modem, B) a "breakout" box, or C) a special cable. In any case, the desired result is to connect the TRANSMIT data on one computer to the RECEIVE data on the other. An example DTE-to-DTE connection is shown below:

	Computer A Pin #	Computer B Pin #	
Ground	1	1	Ground
Transmit	2	3	Receive
Receive	3	2	Transmit
Ground	7	7	Ground

Notice that pins 2 and 3 are "crossed". This is necessary to connect the transmit data on one side to the receive data on the other. Note also that other than the 4 pins shown, no other connections between the two computers are necessary.

Once you have connected the systems together, you will need to test to see if they are actually communicating with one another. Since the systems are wired together, there is no need to dial a phone number to establish a connection. Remember that when transferring files, one system must be in ANSWER mode, the other in CALL mode. The system in CALL mode ALWAYS issues the commands to the ANSWER mode system.

To test the connection, place one machine in ANSWER mode by issuing the command:

Command? **MODE ANSWER<CR>**

Place the other system in CALL mode by issuing the command:

Command? **MODE CALL(CR)**

Make sure that both systems are set for the same baud rate. On each of the machines, issue the command "GO LOCAL". Now type on the keyboard of the computer which is in CALL mode. You should see what you type on both screens. If you do, you're ready to begin transferring files. If not, you probably have a cable problem. Check the cable, and try again.

Once you have established that the two systems are talking to one another, you may begin transferring files. Remember that the commands to transmit or request files are always entered to the CALL mode system.

Appendix D Cabling Information

This appendix shows cabling diagrams for many of the CP/M machines. Additionally, there may be a CABLE.DOC file on your diskette. This file contains information on how to configure a cable for your machine with CROSSTALK, as well as other information relevant to your particular implementation. Locate this file and read it using the CP/M TYPE command (e.g. — TYPE CABLE.DOC(CR)) or any text editor. Information in this file takes precedence over any other information in this manual.

Please note that this version of CROSSTALK takes advantage of two of the modem handshaking lines: CD (pin 8) and DTR (pin 20). For previous users of CROSSTALK, this means that you will most likely have to alter your current cable in order to use CROSSTALK 3.0.

This appendix is organized into 2 parts. The first is a list of computers. This list will show the computer, the cabling diagram to follow, and any additional information necessary in configuring a cable. The second part contains several cable diagrams. To find out how to configure your cable, follow these steps:

1. Check for the CABLE.DOC file on your disk as above. Use this information if it is there.
2. Find your computer in the list of computers. Note which cable diagram you will need.
3. Now find the cable diagram for your computer in the second part of this appendix.

When making your cable, you will use the diagram you found above in conjunction with any additional information under "Notes". When instructed to "jumper" certain pins, this means to connect these pins on the computer side of the cable.

If in doubt about how to make a cable, bring this information to a computer dealer specializing in MODEMS. They will be able to configure a cable for you.

Computer	Diagram	Notes
Apple w/SSC card	1	Header to DTE
Apple w/CCS card	5	Jump 6-20
Apple w/CPS card		
Apple w/SSM AIO 2	2	Header to DTE
Apple w/SSM ASIO	2	Header to DTE
Altos	3	
Compupro w/Interfacer	?	MASTE R=1 Slave =
Eagle 2,3,4	5	
Intertec Superbrain		
Modem Port	1	Jump 5-20
Printer Port	7	
Kaypro 2,4,10	1	
Morrow Micro-decision 1,2,3	6	
Morrow Micro-decision 11	6	
NEC PC8000	3	

Northstar Advantage	8
Northstar Horizon	8
Osborne Executive	1
Modem Port	1
Printer Port	4
Otronra Attache	1
Sanyo MBC-1000	3
Sony SMC-70	1
Televideo 801	7
Televideo 802,803,TPC,800A	1
Teletek	4
TRS-80 Model 2,12,16	2
Vector Graphic 3,4	3
Wave Mate Bullet	4
Xerox 820	1

Using Otronra DTE Cable

Cable Diagrams

#1

Computer	Modem
1	1
2	2
3	3
7	7
8	8
20	20

#2

Computer	Modem
1	1
2	2
3	3
7	7
8	8
4	20

#3

Computer	Modem
1	1
3	2
2	3
7	7
20	8
6	20

#4

	Computer	Modem
	1	1
	3	2
	2	3
	7	7
	4	8
	6	20

#5

	Computer	Modem
	1	1
	3	2
	2	3
	7	7
	5	8
	4	20

#6

	Computer	Modem
	1	1
	3	2
	2	3
	7	7
	20	8
	5	20

#7

	Computer	Modem
	1	1
	3	2
	2	3
	7	7
	20	8
	8	20

#8

	Computer	Modem
	1	1
	3	2
	2	3
	7	7
	4	8
	5	20

This section describes the option switch settings for several popular modems.

Hayes Smartmodem & Smartmodem 1200:

Switch #	Position	Function
1 = 8 = off = UP		Allows CROSSTALK to maintain control over the DTR (data terminal ready) signal.
2 = 7 = . N/A		Not used by CROSSTALK.
3 = 6 = N/A		Not used by CROSSTALK.
4 = 5 = on = DOWN		Tells Smartmodem not to echo "command" characters.
5 = 4 = off UP		Allow auto-answer on first ring. Note that since CROSSTALK maintains control over DTR, the modem will not answer the phone unless CROSSTALK is running, and set to answer mode.
6 = 10 = off = UP		Provides carrier detect signal only when carrier is actually present. Since CROSSTALK uses this signal, this switch MUST be set properly.
7 = 2 = on ?		Set UP for use with single-line home telephone jack (type RJ-11), DOWN for multi-line business phone (RJ-12 & RJ-13) jacks.
8 = 1 = on DOWN		Enables command recognition (Smartmodem 1200 ONLY)

U.S. Robotics Auto-dial 212A:

Switch #	Position	Function
RNG	OFF	Only used in test mode.
8HI	ON	Provides carrier detect signal only when carrier is actually present. Since CROSSTALK uses this signal, this switch MUST be set properly.
BAL	?	Set according to local telephone line conditions, normally down.
DTR	OFF	Allows CROSSTALK to maintain control over the DTR (data terminal ready) signal.

Novation SMART-CAT modems:

Switch #	Position	Function
1	UP	Do not force "Transparent mode"
2	UP	Do not send response codes
3	??	Not used by CROSSTALK
4	DOWN	Default to 1200 baud at power-up
5	DOWN	Enable Auto-answer
6	DOWN	Show carrier detect signal ONLY when carrier is present.

Note: Due to the design of the Smart-Cat, it will be necessary to send the command "% u 0" before performing any protocol files transfers.

Multi-tech MT212AH

Switch #	Position	Function
3-DIP/#1	OFF	Enable Automatic Answer
3-DIP/#2	OFF	DTR Dependent on outside device
3-DIP/#3	ON	CD and DSR act normal
4-DIP/#1	??	Not used by CROSSTALK
4-DIP/#2	ON	Suppress Command Mode Responses
4-DIP/#3	ON	Disable Echo of Command Mode Characters
4-DIP/#4	OFF	Enable Command Mode

This appendix shows typical modem dialing strings for several modems.

Modem	DPrefix	DSuffix	Comments
Bizcomp 1012	V~D~		Set tone-dial mode, then dial number.
Hayes & Robotics	ATDT		Dial with tones.
Hayes & Robotics	ATS11=40DT		"Fast" dial with tones.
Hayes & Robotics	ATDP		Dial with pulse-dialing
Novation Smart-Cat	% D~		Dial with tones. Note - There MUST be a blank space after the "~".
SSM Transmodem & Racal 345x	^E ~D ~		"Wake up" modem, then dial number.
UDS 212A/D	OG0~EN~D~		Put modem to sleep, wake it up, then dial the number.
Multi-tech MT212AH	ATDT		Dial with Tones
Multi-tech MT212AH	ATDP		Pulse dial

This table is a list of each of the standard ASCII control characters. The list shows the character, the hex value of the character, and the mnemonic name of each character.

Keyboard Character	Hex Value	Ascii Mnemonic
^@	00	NUL
^A	01	SOH
^B	02	STX
^C	03	ETX
^D	04	EOT
^E	05	ENQ
^F	06	ACK
^G	07	BEL
^H	08	BS
^I	09	TAB
^J	0A	LF
^K	0B	VT
^L	0C	FF
^M	0D	CR
^N	0E	SO
^O	0F	SI
^P	10	DLE
^Q	11	DC1 (X-ON)
^R	12	DC2
^S	13	DC3 (X-OFF)
^T	14	DC4
^U	15	NAK
^V	16	SYN
^W	17	ETB
^X	18	CAN
^Y	19	EM
^Z	1A	SUB
^[1B	ESC
^`	1C	FS
^]	1D	GS
^~	1E	RS
^_	1F	US

This appendix is provided for programmers who may wish to modify the CROSSTALK driver and/or screen attribute areas of CROSSTALK. It is divided into two sections: the first describes the communications drivers and its responsibilities and the second details the patchable areas for the video drivers.

CROSSTALK I/O DRIVERS

The drivers for CROSSTALK 3.0 are much more complex than the drivers for the previous releases of CROSSTALK, and extend from 100h to 3ffh. The reserved areas are defined as follows (all addresses are in hex):

0100	jmp	0400	; jump to XTALK
------	-----	------	-----------------

; The following bytes are reserved for UART parameters:

0103	db	3	; baud rate
0104	db	8	; data bits
0105	db	1	; stop bits
0106	db	0	; parity (0=n, 1=e, 2=o)
0107	db	1	; dtr
0108	db	1	; local flag

; The following is a jump table to the driver routines.

0109	jmp	CHARIN	138	; get modem char.
010C	jmp	CHAROUT	141F	; send modem char.
010F	jmp	CARRIER	160	; check modem carrier
0112	jmp	BREAK	178	; send break
0115	jmp	INIT	192	; initialize port (w/ reset)
0118	jmp	SFTINIT	192	; soft initialization
011B	jmp	DEINIT	1D0	; de-initialize port
011E	jmp	SETBAUD	192	; set baud rate
0121	jmp	SETDTR	1DD	; set dtr
0124	jmp	WAITTONE	1EC	; wait for dial tone
0127	jmp	DIALDIGIT	140	; dial a digit
012A	jmp	WAITCARR	1EF	; wait for carrier
012D	jmp	HANGUP	1ES	; hang up the phone
0130	jmp	ANSWER	1F1	; answer the modem

The space from 133h to 3ffh will contain the actual routines to do the modem I/O. The routines may be in any order, just make sure that the correct position in the jump table (above) points to the corresponding routine. Also, all registers used must be preserved except the accumulator, unless otherwise noted. The routines are documented below, and the following conventions are used:

A-E, H&L are the respective registers in the 8080 & Z-80
C = carry flag
Z = zero flag

CHARIN

Check carrier, and if no carrier, return C set. Else, get a character from the port, if one is ready.

ENTRY:

EXIT: C set if no carrier, reset if carrier present
 Z set if no character
 A = character, Z reset

CHAROUT

Check carrier, and if no carrier, return C set. Else, send the character in A to the port.

ENTRY: A = character to transmit
EXIT: C set if no carrier, reset if carrier present

CARRIER

Check to see if the port has carrier. If the LOCAL flag is true (1) then assume that carrier is true. Note that you will check to see if the LOCAL flag is true before checking the status of the port. If CHAROUT calls CARRIER, be careful to preserve the character in A.

ENTRY:

EXIT: C set if NO MODEM CARRIER
 C reset if CARRIER PRESENT

BREAK

This routine is required to send a 250-300 ms. break. Note that a call to 403h will delay BC milliseconds.

ENTRY:

EXIT:

INITPORT & SFTINIT

It is the responsibility of these routines to initialize the port in accordance with the values set from 104h to 107h. The only difference between these routines are that INITPORT is called ONLY once (for the first initialization). You will usually reset the chip on this pass. All other calls are to the SFTINIT jump in the above table.

The bytes from 104 to 107 are defined below:

104 DBITS -	8 = 8 data bits, 7 = 7.
105 SBITS -	1 = 1 stop bit, 2 = 2 stop bits
106 PARITY-	0 = No parity, 1 = Even parity, 2 = Odd parity
107 DTR -	0 = No DTR, 1 = DTR

If any of the above combinations are impossible, it is best to provide a default (i.e.- If baud is not selectable other than varying the clock mode, then select either 3 or 12).

ENTRY:
EXIT:

DEINIT

This routine will be called before CROSSTALK exits to CP/M. If you need to re-enable handshaking, shut off interrupts, or "chain" to other programs, this is the place to do it.

ENTRY:
EXIT:

SETBAUD

This routine must set the baud rate in accordance with the byte at 103h. If the baud rate is not software selectable, you can usually vary the baud by changing the clock factor in the chip. If an impossible baud rate for your machine is being asked for, simply select a default value (make sure that you set the byte at 103 to reflect the correct baud rate). The baud rate should be set as follows:

0 = 0110, 1 = 1200, 2 = 2400, 3 = 300, 4 = 4800, 6 = 600, 9 = 9600

ENTRY:
EXIT:

SETDTR

This routine must SET DTR on the chip, and store a 1 in the DTR byte (107h).

ENTRY:
EXIT:

WAITTONE

This routine is called just before dialing the modem. It should wake the modem up and wait for acknowledgment of the dial tone. If none is present, it should time out and return with C set. If a dial tone is detected, return with C reset. If this feature is not implemented, just assume that there is dial tone.

ENTRY:

EXIT: C set if no dial tone

C reset if dial tone detected (or not implemented)

DIALDIGIT

This routine is called to dial the modem. It is called with the character to dial in A, and is called for every character in the DPREFIX, NUmber, and DSuffix strings. If you are using an "intelligent" modem, it is sufficient to let the "jmp" in the table go directly to CHAROUT (Note: CROSSTALK sets the LOCAL flag before dialing.)

ENTRY: A = character (digit) to dial.

EXIT:

WAITCARR

This routine is called directly after dialing the modem. If carrier is implemented on your computer, it is sufficient to set the Z flag (C reset) and let XTALK poll the CARRIER routine for you.

If carrier is not implemented, some modems (eg-Hayes) send result codes when carrier is detected. If this is the case, loop for HL times 100 milliseconds (call 403h w/ BC = 100 HL times) while waiting for this code. Return with Z not set and C not set to show carrier detected. Otherwise, set C to reflect no carrier.

ENTRY: HL = SECONDS to wait / 10. For 30 secs., HL = 300.

EXIT: Z set, C reset - CROSSTALK will wait for carrier

Z reset, C reset = CARRIER detected.

Z reset, C set = NO CARRIER detected.

HANGUP

This routine is called when a BYE or a QUIT is issued. It should set the DTR byte to 0, drop DTR on the modem port and hang up the modem. It is usually sufficient to just drop DTR for the modem to hang up. If your port or modem does not support this, usually a string to the modem will tell it to hang up (eg- send + + + ATH with the proper guard times to hang up a Hayes, or Hayes compatible modem).

ANSWER

This routine is provided for modems which need special code in order for it to answer an incoming call. If not used, just C reset, Z set tells CROSSTALK to wait for carrier. Z reset tells CROSSTALK that this routine is implemented, with the following results: C set means this routine was aborted, C reset means that the phone was answered and carrier is present.

ENTRY: A = current ATTEN char. If pressed, abort this routine.
EXIT: Z set = CROSSTALK waits for carrier
Z reset, C set = routine aborted.
Z reset, C reset = phone answered, carrier present.

Machine specific attributes

The following areas are user patchable areas; they should be set to reflect your system and terminal attributes. The areas are defined below, with all address expressed in hex. Note that all strings (anything longer than 1 byte) must be terminated with a null (00h).

406-406 CPU Speed in MHz. (1 = 1, 2 = 2, ... 7 = 7)

The following set the terminal attributes:

407-40F Clear screen codes.
410-418 Normal intensity: terminal mode.
419-421 Enhanced video: Command? prompt & terminal mode msgs.
422-42A Status screen enhance: 1st 2 letters of each command.
42B-433 Status screen normal: end of each command and args.

The following define cursor positioning:

434-434 ANSI Standard flag (1 = ANSI, 0 = non-ANSI)
435-435 Row or column set first. (1 = row set first)
436-43A Set cursor position lead-in. (i.e.-ESC = for TVI)
43B-43F Delimeter between row/column, column/row.
440-440 Bias for cursor positioning. (i.e.-20h for TVI)
441-441 Number of rows for your terminal

442-xxxx Start of CROSSTALK main code.

Index

? key	15, 16
Abort command	66, 79
Accept command	52, 79
Alarm command	67, 80
Answer mode	51, 83, 91, 94
Answerback	33
Answerback command	80
Answerback request	52
Ask command	67, 69, 80, 88
Attention command	80
Attention key	22, 23, 31, 80
Automatic log-ons	65, 72
Baud rate	20, 36, 96
Bksize command	58, 81
Blankex command	49, 81
Branching in script files	70, 71, 89, 96
Break	32, 82, 95
Break command	81
Break key	32
Break signal	72
Buffer space	39
Bye	24
Bye command	82
Cables	10
Call mode	91
Capture buffer	39
Capture command	82
Capture search	42, 83
Capture status	42, 83
Capture to disk	40
Capture toggle	28
Capturing data	103
Carrier detect	70
Carrier detect signal	88
Changing default drive	86
Changing default settings	63
Changing function keys	87
Clear command	68, 83
Clearing the terminal screen	68
Command character	53
Command command	83
Command files	61, 90
Command line	14
Conditionals in Script files	88

Control characters	30
Creating command files	62
CROSSTALK protocol	55
Cstatus command	42, 83
Cwait command	46, 48, 84
Data bits	36
Data command	36, 84
Debug command	31, 84
Default settings	63
Dial Prefix	28
Dial string	86
Dial Suffix	28
Dialing	21
Dialing strings	28
Dialing with CROSSTALK	88
Dir command	25, 85
Directory command	25, 85
Display file	26, 98
Displaying control chars.	31
Displaying files	26, 98
Displaying function keys	27, 87
Displaying the capture buffer	43
Do command	64, 68, 78, 85
DPrefix command	28, 86
Drive command	26, 86
DSuffix command	28, 86
Duplex	35, 87
Duplex command	35, 87, 91
Echo	35, 87
Enquire	33
Entering commands	14
Erase command	26, 43, 87
Executing script files	65
Exiting CROSSTALK	24, 93, 99
Expanding tabs	48, 97
File transfer	5
Filter	29, 30, 87
Filter command	30, 87
Filtering control characters	30, 87, 89
Fkeys command	27, 87
Function keys	27, 67, 87, 89
Go command	21, 88
Greeting message	53, 92
Hardwiring computers using CROSSTALK	88
Help	15, 16, 88

Rcve command	93
Rdials command	21, 93
Re-dial maximum	21, 93
Re-dialing	21
Remote List command	28
Reply command	72, 94
Retro-capture	41, 82
Return re-definition	32, 97
Review capture buffer	28, 43, 98
Reviewing files	26, 98
Rewinding a script file	72, 94
Rquest command	55, 56, 94, 104
Rwind command	72, 94
Rxmodem command	55, 59, 94
 Save command	62, 95
Saving captured data	40, 99
Sbreak command	72, 95
Screen command	66, 71, 95
Script file commands	66
Script file example	76
Script files	64, 85, 91, 94, 95
Searching the capture buffer	83
Send command	45, 84, 90, 92, 95, 97, 98, 103
Sending a Break in script files	72
Sending a File to Another Computer	45
Sending text from a script file	72
Setting the Block size	81
Setting the phone number	92
Skip command	71, 96
Speed command	20, 36, 96
Standard settings	63
Starting a script file	68, 85
Status line	31
STD.XTK	63
Stop bits	37, 96
Stop command	37, 96
Stripping line feeds	49, 92
Switch command	17, 96
Switch key	17, 23, 31, 96
 Tabex command	48, 97
Terminal mode	5, 22
Terminal status	31
Terminal status line	31, 97
Trip Key	27, 89
TStatus command	31, 97
Turnaround command	32, 97
Type command	26, 28, 43, 98

Help command	16, 88
Hex representation of characters	31
If command	69, 88
Infilter	30, 89
Infilter command	30, 89
Inter-character delay	48, 84
Inter-line delay	46, 90
Jump command	70, 89, 96
Key command	27, 89
Key settings	17
Label command	69, 89
LFauto command	37, 90
Line feeds	37, 49, 90, 92
List command	28, 90
Load command	61, 90
Local	21
Local echo	87
Local mode	21, 88
Lwait command	46, 90
Magic command	73, 91
Message command	71, 91
Mode command	91
Modem commands	28
Modem switch settings	11
Name	20, 92
Name command	92
New user script file	19
No command	92
Number	20
Number command	20, 92
On line	70
On/off toggle	15
OUtfilter command	49, 92
Parity command	37, 92
Password	93
Phone number	92
Printed information	30
Printer command	29, 83
Printer toggle	28
Protocol transfers	36, 81, 92, 93, 94, 100, 104
PWord command	52, 93
Quit command	24, 93

UConly command	49, 98
Upload	6
Uploading ASCII data	45
Uploading ASCII text files	45, 95
User command	98
Version command	98
Wait command	72, 91, 98
Word length	36
Work disk	9
Write command	40, 82, 99
Xcpm command	24, 99
Xmit command	55, 100, 104
XMODEM protocol	55, 59, 94, 100
XMODEM transfers	94, 100
Xxmodem command	100

Trademarks

The following trademarks appear in this manual:

CP/M, CP/M-86, and MP/M are trademarks of Digital Research, Inc.

MS-DOS is a trademark of Microsoft

IBM is a trademark of International Business Machines, Corp.

CROSSTALK, CROSSTALK-XVI, TRANSPORTER and Microstuf are trademarks of Microstuf, Inc.

The Source is a service mark of Source Telecomputing Corp.

Dun & Bradstreet is a trademark of The Dun & Bradstreet Corp.

Microsoft is a trademark of Microsoft Corp.

Stack Smartmodem is a trademark of Hayes Microcomputer Products, Inc.

Smart-Cat is a trademark of Novation, Inc.

Auto-Dial 212A is a trademark of U.S. Robotics, Inc.

Univac 9000 is a trademark of Sperry Univac

Telenet is a trademark of GTE

; CROSSTALK "Script" file to log onto the Source via Telenet

; NOTE: This script is provided as a working example of a CROSSTALK script
; file for logging on to the SOURCE via TELENET. You will need to
; set your "id" string and password where indicated below.

; MICROSTUF, INC. assumes no responsibility and makes no warranties
; for the accuracy or use of this script file.

magic on
reply :
wait delay 5
reply :
wait string "TERMINAL="
reply D1!
wait character '@'

; connect to the node for the SOURCE
reply C 30147!
wait character ">"

; You will need to replace this dummy string with your ID and PASSWORD
reply id abc123 password!
wait string "->"
ask You are now logged in. Do you wish to invoke the SRC menu?
if -y abort

; This rest of this script is designed to make using the source very easy

; make sure we are at the terminal screen, kill any waiting prompts
magic on
screen terminal

label MAIN_MENU

clear

message

To use any of the following SOURCE services, simply press the corresponding letter of your desired selection.

SERVICE (action)	Press:
-----	-----
Check CP/M posts	C
Check IBM posts	I
Leave this script	L
Check stock quotes	Q
Get the latest in SPORTS	S
Exit the Source	X

```
label WRONG
Ask Enter your choice:

if lqiosx skip ACT_@
jump WRONG

; Here, we will just do an abort to get out of the script file.

label ACT_L
; Send an extra return so user will see the source's "->" prompt
reply :
; and now, abort this script
abort

; ***** GET STOCKS *****
label act_Q

; tell the source that we want the stockcheck feature
reply stockcheck!

; wait for ending prompt
wait string "<H>ELP:"

reply 1!
wait string "previous menu"

; now, turn capture on, using automatic append option
capture STKQUOTE /A

; here, you will add "reply"s with the symbol you want a quote of.
; Examples follow which can be modified to check the quotes you want.

; get IBM (Note that we already got the "previous menu" prompt
reply ibm!

; Wait for prompt again and then get COMPAQ's
wait string "previous menu"
reply cmpq!

; Wait for prompt again and then get Home Depots
wait string "previous menu"
reply homd!

; Wait for prompt again and then get LOTUS'
wait string "previous menu"
reply lots!

; Note - use the same sequence as above for your stocks, or read the
; Sources help on the stockcheck for easier ways.

; now, wait for prompt again, then turn capture off, back to ->
wait string "previous menu"
capture off
```

```
re :
```

```
wait string "<H>ELP:"  
re q:
```

```
wait string "->"
```

```
ask Your stock's have been saved in STKQUOTE. Press Enter to return to menu  
jump MAIN_MENU
```

```
; ***** DO THE <<POST>> stuff *****
```

```
label act_I
```

```
label act_C
```

```
message
```

```
When you are finished using POST, simply type a "q" at the
```

```
"<N>ext, <PO>st, or Return for text-"
```

```
prompt and you will be returned to this menu.
```

```
skip PO_@
```

```
; post read IBM:
```

```
label po_I
```

```
reply post read ibm!
```

```
skip PO_X
```

```
; post read CP/M:
```

```
label po_C
```

```
reply post read cp/m!
```

```
skip PO_X
```

```
; here, set the rest of the post conditions:
```

```
label PO_X
```

```
wait string "all :"
```

```
reply :
```

```
; what we do is wait for the user to type the q @ the text- prompt using  
; the when condition. The result is a do, w/ no args.  
; note that the line after the when condition is a wait manual, which  
; suspends this file. When the WHEN occurs, it issues the DO, which  
; picks up at the following RWIND
```

```
when "text-q" do
```

```
wait manual
```

```
reply :
```

```
wait string "->"
```

```
jump MAIN_MENU
```

```
; ***** here, we do the sports stuff, using the when as above *****
```

```
label act_S
```

message

To end the SPORTS section of the SOURCE, type the letter q at the
"Enter item number or Help"
prompt and you will be returned to the main menu.

```
reply sports!
when "Help q" do
wait manual
reply :
wait string "->"
jump MAIN_MENU
```

; ***** EXIT THE SOURCE *****

```
label act_X
ask Are you sure?
if -y jump WRONG
re off!

wait string "disconnected from"
wa d 20
sc s
by
lo -
```