62940076

CONTROL DATA CORPORATION

CDC® 110 VIKING SYSTEM

SOFTWARE USER'S MANUAL

62940076

CONTROL DATA CORPORATION

CDC® 110 VIKING SYSTEM

SOFTWARE USER'S MANUAL

REVISION RECORD

REVISION	DESCRIPTION
A (03-29-83)	Manual released
B (10-12-83)	Revised and reprinted to cover the enhancements made in release 5 of the 110 CP/M 2.2 software. This edition obsoletes the previous edition.

Publication No.

62940076

REVISION LETTERS I, O, Q AND X ARE NOT USED

Address comments concerning this manual to:

Control Data Corporation
Publications and Graphics Div.
2401 North Fairview Avenue
St. Paul, Minnesota 55113

or use Comment Sheet in the back of this manual.

LIST OF EFFECTIVE PAGES

New features, as well as changes, deletions, and additions to information in this manual are indicated by bars in the margins or by a dot near the page number if the entire page is affected. A bar by the page number indicates pagination rather than content has changed.

PAGE	REV	PAGE	REV	PAGE	REV
Front Cover	-				
Title Page	-				
ii thru xvi	B				
1-1 thru 1-4	B				
2-1	A				
2-2 thru 2-11	B				
3-1 thru 3-6	B				
4-1 thru 4-5	B				
5-1 thru 5-4	B				
6-1 thru 6-5	B				
7-1	A				
7-2 thru 7-4	B				
8-1	A				
8-2 thru 8-4	B				
9-1 thru 9-12	B				
10-1 thru 10-30	B				
11-1 thru 11-25	B				
12-1 thru 12-14	B				
13-1 thru 13-76	B				
A-1 thru A-9	A				
B-1	B				
B-2 thru B-24	A				
C-1 thru C-3	B				
D-1/D-2	A				
E-1 thru E-35	B				
F-1	B				
F-1 thru F-6	A				
F-7	B				
F-8 thru F-10	A				
G-1 thru G-6	B				
H-1 thru H-6	B				
I-1	B				
J-1 thru J-5	B				
Comment Sheet	B				
Mailer	-				
Back Cover	-				

This manual provides information on installing software and performing utility operations on the Control Data® 110 Viking System. The software products included are the CP/M® 2.2 Operating System, CBASIC®, BASIC-80*, and Pascal/M**.

This manual describes how to use the Control Data 110 system with CP/M. If you are using this system in other applications, you will have to use the manuals supplied with the application.

This manual is not intended to teach you how to use all of the capabilities of CP/M. If you want to learn how to use CP/M, there are a number of publications available in many bookstores. Books on CP/M include:

The CP/M Handbook by Rodnay Zaks
Published by SYBEX

Using CP/M by Judi Fernandez & Ruth Ashley
Published by Wiley

Osborne CP/M User Guide by Thom Hogan
Published by Osborne/McGraw-Hill

This manual is organized for two types of readers (a system user and a system software writer). The system user should read sections 1 through 12. These sections give detailed directions on how to create backup flexible disks of master flexible disks, how to install and test software, how to use the various flexible disk and rigid disk Control Data-supplied utility routines, and how to use the phone feature of the optional internal 1200/1200-baud modem.

The system software writer should read section 13 and the appendixes. These are organized as a reference manual and explain the unique features of the Control Data 110 CP/M implementation. They also provide useful system reference material.

CP/M and CBASIC are registered trademarks of Digital Research, Inc.
*BASIC-80 is a trademark of Microsoft, Inc.
**Pascal/M is a trademark of Sorcim.

Additional copies of this manual and the related publications in the list which follows are available from:

Control Data Corporation
Literature and Distribution Services
308 North Dale Street
St. Paul, Minnesota 55103

Title	Publication Number
CBASIC Reference Manual	62940021
Pascal/M User's Reference Manual	62940022
CP/M Operating System Manual	62940335
BASIC-80 Reference Manual	62940039
Microsoft BASIC Reference Book	62940040
110 Microcomputer System Owner's Manual which consists of a bookshelf case and the following four guides:	62940086
Customer Planning Guide	62940070
Installation Guide	62940071
Operating Guide	62940072
Problem Solving Guide	62940073
721-21/31 Owner's Manual	62950101*
721 Enhanced Display Terminal Hardware Reference Manual (supplemental programming information if terminal is to operate in CYBER mode under CP/M	62950102*

*These publications apply to your Type 3 display terminal if it is either a CC634-B/CC638-B unit or a CC634-A/CC638-A unit with a YR109-A enhanced firmware option installed.

Title	Publication Number
721 Display Terminal Operator's Guide/ Installation Instructions	62940019*
721 Display Terminal Hardware Reference Manual (supplemental programming information if terminal is to operate in CYBER mode under CP/M)	62940020*
721-301 Enhanced Graphics/Firmware Option Reference Manual (supplemental programming information if terminal is to operate with graphics)	62950116**
721-301 Graphics/Firmware Option Reference Manual (supplemental programming information if terminal is to operate with graphics)	62940095†

---

*These publications apply to your Type 3 display terminal if it is a CC634-A/CC638-A unit without the YR109-A enhanced firmware option installed.

**This publication may be used for reference if the terminal controlware on your CP/M disk is version 5.00 or higher.

†This publication may be used for reference if the version of the terminal controlware on your CP/M disk is below 5.00.

CONTENTS

11. PHONE UTILITY

12. UTILITIES

13. IMPLEMENTER'S INFORMATION

APPENDIXES

FIGURES

TABLES

The Control Data 110 Viking System is a standalone microcomputer system. It is designed to use the CP/M operating system. This section identifies your equipment, defines the system configuration, explains how to read this manual, and mentions the key terminology used in this manual.

YOUR EQUIPMENT

Your Control Data 110 has several possible configurations. The following criteria identify which Type of equipment you have. For future reference, enter the Type designations in the spaces provided. The Type-designating labels are on the corporate logo on the front of the equipment.

DISPLAY TERMINAL

You have one Type 3 display terminal:

__________ Type 3 terminal: separate keyboard and tilting display. Your terminal may include an optional internal 1200/1200-baud modem.

FLEXIBLE-DISK DRIVE

You have either one or two Types of flexible-disk drive:

__________ Type 1 (primary flexible-disk) drive: has a RESET button on the front of the unit.

__________ Type 2 (secondary flexible-disk) drive: does not have a RESET button on the front of the unit. If you have a Type 2 drive, you must also have a Type 1 drive.

RIGID-DISK DRIVE

You may have one to four rigid-disk drives. There are two Types of rigid-disk drives. You may have all one Type or a mix of two Types. The physical appearance of both types is identical.

__________ Type 3 (12.5-megabyte rigid-disk) drive: The model number is FA502-A/B as indicated on the label attached to the back panel.

__________ Type 4 (25.0-megabyte rigid-disk) drive: The model number is FA502-C/D as indicated on the label attached to the back panel.

PRINTER

These Types of printers are used with the Control Data 110.

__________ Type 1 (parallel graphics) printer: has two buttons on the front of the unit; uses parallel channel.

__________ Type 2 (matrix) printer: has four buttons on the front of the unit, but has no separate indicator lights.

__________ Type 3 (letter quality) printer: has four buttons and three separate indicator lights on the front of the unit.

__________ Type 4 (serial graphics printer): has a control panel and liquid crystal display (LCD) panel inset in the lower right corner of the unit.

Other serial-port printers can be used with the Control Data 110. When information applies only to a specific printer, the Type designation is used. When information applies to other printers as well, the term "serial-port printer" is used.

THE SYSTEM CONFIGURATION

Your Control Data 110 system must include the terminal and Type 1 disk drive. It may optionally include a printer, a Type 2 disk drive, up to four Type 3 or Type 4 disk drives, and an internal 1200/1200-baud modem.

HOW TO READ THIS MANUAL

Different sections of this manual pertain to different Control Data 110 configurations. The key areas of difference are in the number and Type of disk drives you have in your system, and whether your system contains an optional internal 1200/1200-baud modem.

All system users should read section 2, which introduces the equipment. Once you have used some of the utilities to install your software, read section 12, which discusses the utilities. You may also find it helpful to review the Operating Guide of your owner's manual. Additional reference material, including some books about CP/M, are listed in the preface of this manual.

The Type 3 terminal can display the character sets of various countries and may have matching optional keycaps installed. The operational descriptions in this manual assume that the standard (United States) character set is in use. If your terminal is using a different character set, some of the symbols displayed may be different than those mentioned in a description. In such a case, refer to the symbol cross-reference table in appendix I of this manual.

If you have only a Type 1 disk drive or if you have a Type 1 disk drive and one or more rigid-disk drives, you should also read:

- Section 3 on installing CP/M, testing the installation, and preparing your working copy of CP/M
- Section 4 on installing CBASIC, PASCAL/M, and BASIC-80
- Section 5 for general procedures used in copying software

If you have both a Type 1 and a Type 2 disk drive, you should also read:

- Section 6 on installing CP/M, testing the installation, and preparing your working copy of CP/M
- Section 7 on installing CBASIC, PASCAL/M, and BASIC-80
- Section 8 for general procedures used in copying software

If you have one or more Type 3 or Type 4 disk drives, you should also read:

- Section 9 for formatting the rigid disk
- Section 10 for making backup copies of information on the rigid disk

If you have the optional internal 1200/1200-baud modem, you should also read:

- Section 11 for instructions on using the phone feature of the modem.

System programmers should also read section 13 and the appendixes.

KEY TERMINOLOGY

The procedures often require that you press keys on your terminal. This manual uses angle brackets (< >) to set off the name of the key you should press. For example:

<next> means press the key with NEXT printed on the top of the key,

<CTRL S> means press the CTRL key and the S key,

<CTRL C> means press the CTRL key and the C key.

Software written for CP/M may refer to the CARRIAGE RETURN, CR, or RETURN key in the documentation and in the display messages. On your terminal, use the NEXT key. Since CP/M does not distinguish between uppercase and lowercase letters, CP/M software may use either. You may type responses to CP/M prompts in either uppercase or lowercase letters.

This section describes the possible configurations of the Control Data 110; provides hardware requirements and options, equipment characteristics, and information about operating modes; and introduces the terminal.

CONFIGURATIONS

The minimum Control Data 110 Viking System consists of a Type 3 terminal and a Type 1 disk drive. An optional Type 2 disk drive, printers, and up to four rigid-disk drives are available. Figure 2-1 shows the terminal, a Type 1 disk drive, and a rigid-disk drive.

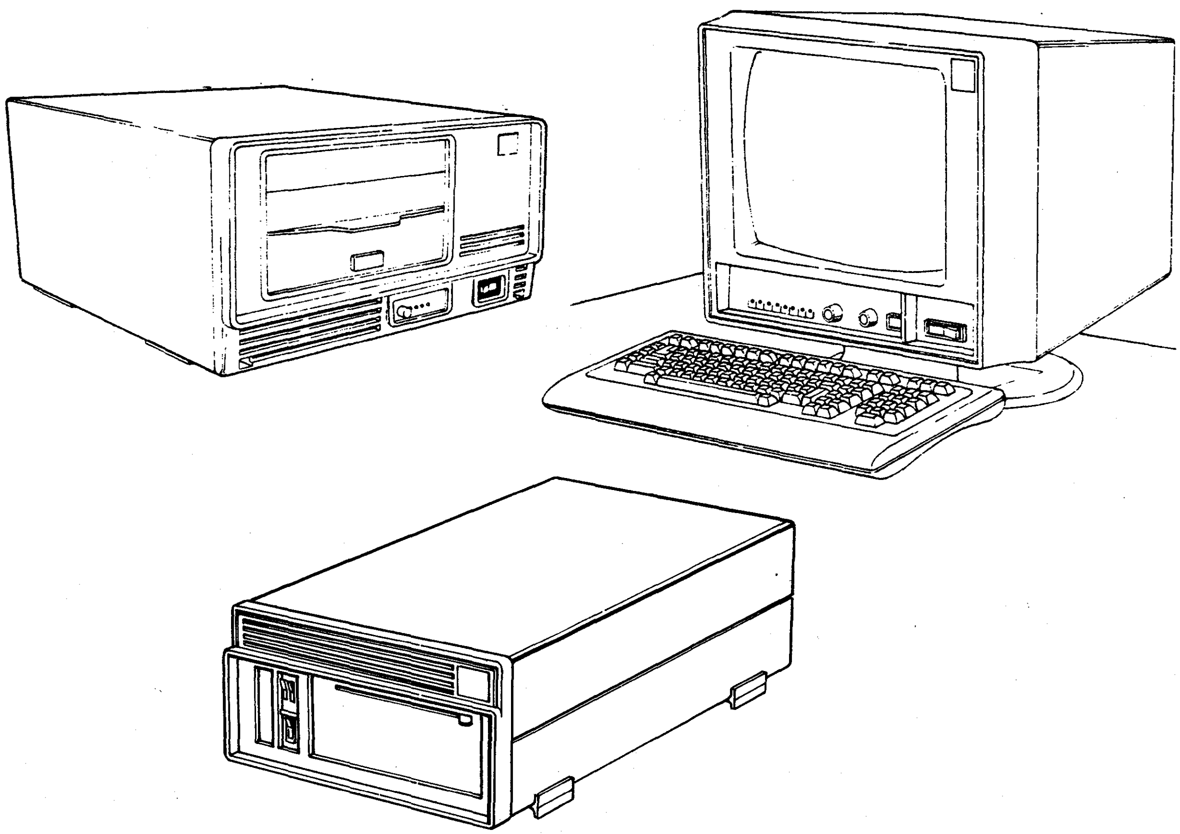


Figure 2-1. Control Data 110 System with Rigid-Disk Drive

The Control Data 110 system is a microcomputer. It can be used in standalone mode or in terminal mode. In standalone mode, it uses application software packages available on 8-inch flexible disks. Users can, however, write their own application software. In standalone mode, application software loads in via a flexible disk.

In terminal mode, the application software transfers from another computer. The Type 3 terminal with the graphics option may be used as a terminal on the Control Data Shared Network. The Control Data 110 system terminal has communications capabilities that allow it to operate with other computer systems in a distributed data-processing application.

In either mode, data variables and program control commands are entered through the keyboard. Processed data can be stored on a flexible disk, stored on a rigid disk, printed out if the system includes an optional printer, transferred to another computer if the system is in terminal mode, or transferred to another location via the modem.

HARDWARE REQUIREMENTS AND OPTIONS

The minimum hardware requirements for the Control Data 110 system are:

- A Type 3 terminal with parallel-channel option
- A Type 1 disk drive with 64K of memory

The options are:

- A Type 2 disk drive
- Up to four Type 3 and/or Type 4 rigid-disk drives
- A printer
- A dual asynchronous port for the Type 3 terminal (required to support an RS-232 interface printer)
- An internal 1200/1200-baud modem

EQUIPMENT CHARACTERISTICS

The equipment characteristics for the terminal, flexible-disk drive, rigid-disk drives, printers, modem, and for the computer characteristics of the system are as follows:

TERMINAL

- 80- or 132-column by 24- or 30-line display.
- Keyboard generates all 128 ASCII characters.
- Options provide special character sets for several languages.
- Special keys generate multiple character sequences.
- Numeric cluster.
- Start and stop screen output from keyboard.
- Typamatic keys generate repeated output.
- PRINT key causes a copy of the information on the screen to be printed.
- Inverse video, low intensity, or blinking fields may be selected by program to highlight fields.
- Touchpanel interface provided to computer programs (if the graphics/touchpanel option is included in the terminal).
- Operator may select block-form or underline-form, blinking or non-blinking cursor.
- Operator may choose to see green letters on black background, or black letters on green background.

FLEXIBLE-DISK DRIVE

- Format - Double-density, double-sided flexible disk or single-density, single-sided flexible disk.
- Capacity per flexible disk - 1.216-million bytes usable storage in double-density; 243-thousand bytes usable storage in single-density.
- System capacity with optional Type 2 disk drive - 2.432-million bytes maximum in double-density. 486-thousand bytes maximum with single-density.
- An optional Type 2 disk unit also makes copying flexible disks more efficient.

RIGID-DISK DRIVE

- 12.5- or 25.0-megabyte (Mb) storage capacity.
- 512 directory entry for both Type 3 and Type 4 disk.

PRINTERS

- CDC parallel channel for Type 1 printer.
- RS-232 serial interface for serial printers, such as the Type 4 serial graphics printer.

MODEM

- Optional telecommunications port.

COMPUTER

- 4-megahertz Z-80 CPU.
- Internal memory of 65 536 bytes. At least 48 000 bytes are available for user outside of operating system.
- Date and time clock maintained to the second.
- Serial full-duplex communications.

OPERATING MODES

The Control Data 110 system is capable of multiple operating modes. These modes are described briefly below.

CP/M 2.2 Computer System	The Control Data 110 system uses the CP/M 2.2 operating system. CBASIC, BASIC-80, Pascal/M, and other applications and programs are available from Control Data and other vendors.
Self-contained PLATO Station	The Control Data 110 system uses PLATO flexible disks. No access to the PLATO network is required.
Control Data Shared-Network Terminal	The terminal is connected to the Control Data Shared-Network. All the services of this network are then available including CDC PLATO.

This manual deals primarily with the CP/M 2.2 computer system mode of operation. If you wish to operate in a different mode, contact a Control Data sales representative.

TERMINAL

The Type 3 terminal is a multi-function/multi-mode terminal which must have CP/M mode installed. The standard terminal does not provide graphics or touchpanel features, or the internal 1200/1200-baud modem. Control Data 110 System CP/M supports the touchpanel, graphics, and the modem, if any of these options have been included. The following paragraphs discuss the terminal keyboard, the graphics option, and the modem features.

TERMINAL KEYBOARD

The terminal keyboard has tan, gray, and white keys (figure 2-2). Some of the keys have one legend on the top and another on the front skirt of the keycap. The legend on the top applies normally. The legend on the skirt applies if you have pressed the ⇧ (shift) key.

The tan and grey keys are similar to a standard typewriter layout. Character sets for various languages are available and are discussed in appendixes A, B, and I. The grey keys also form a numeric cluster. They perform differently when they are shifted (refer to appendix B). Some of the white keys are used by CP/M to generate special operations. The rest of the white keys generate two or three character sequences, which are not used in normal CP/M operations. The following list gives the CP/M use of the white keys.

Key	Action
NEXT	Carriage return
←	Backspace
⇧	When held down causes tan key to issue the uppercase character (shift key).
⇥	Tab

Dual-action key. Pressing once turns on red light on keycap and causes tan keys to issue uppercase alphabetic characters. Pressing again turns off red light and allows normal operation of tan keys (similar to a shift-lock key).

CTRL — When held down with another key causes selection of the Control case value for the key.

CR DEL — When shifted issues carriage return code. When unshifted causes delete code "DEL" to be issued.

PRINT — Causes contents of screen to be printed on the attached printer.

SETUP — Enters the setup mode of operation to allow you to change display characteristics.

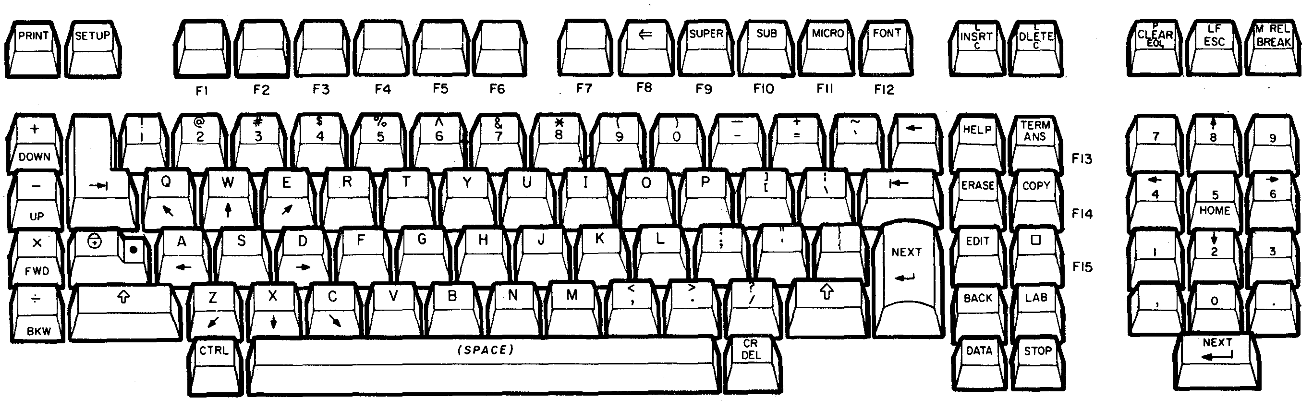


Figure 2-2. Terminal Keyboard with Standard Keycaps

GRAPHICS OPTION

When the touchpanel/graphics option is included, the terminal can produce and use graphics as though it were a Tektronix 401X (4010/4014) terminal. In graphics operation, the CRT screen displays dots. The entire screen has 512 dots across each line, and 512 dots up and down each row. The graphics option uses those dots, or points, to form characters, lines, and blocks. Each point has a mathematical address: the X coordinate specifies the point's horizontal position; the Y coordinate specifies the point's vertical position.

The following paragraphs discuss the use of the graphics option in standalone operations under CP/M. The modes of graphics operation are defined, instructions for preparing to use the option are provided, and methods of operator input to applications programs that use the graphics option are discussed. Further details of graphics operations can be found in the 721-301 manuals listed in the preface. The programs covered in the 721-301 manuals are similar to those on the CP/M disk.

Graphics Option Modes

The graphics option may be used to plot characters, dots, lines or rectangular blocks. One of seven modes can be chosen. These modes are alpha, graph, point-plot, special point plot, incremental plot, graphics input (GIN), and block mode.

Alpha Mode

In alpha mode, the terminal displays the 95 displayable ASCII characters. (The DEL key is ignored in alpha mode.) There are five character sizes - the standard PLATO characters, plus four character sizes that agree with the four sizes available on the Tektronix 4014 terminal.

A blinking cursor marks where the next character will be entered. After the symbol is entered, the cursor moves one position to the right. When the end of a line is reached, the cursor moves to the leftmost position on the next line. The exact cursor home position depends on the character size selected, but is always in the upper-left corner of the CRT screen.

You can use two keys to enter alpha mode. Pressing the ☐ key or the Page Clear key resets the terminal to alpha mode, positions the cursor to home position, and clears the screen. Pressing the shifted ☐ key or the HOME key resets the terminal to alpha mode, positions the cursor to home position, and selects size 1 characters.

Graph Mode

Graph mode is used to draw a line between two points. Graph mode can only be accessed by an applications program.

Point-Plot Mode

Point-plot mode is used to display single points not connected by lines. Point-plot mode can only be accessed by an applications program.

Special Point-Plot Mode

The special point-plot mode allows the display beam intensity to be varied on the Tektronix 401X series terminals. This produces a "grey scale" capability. The Type 3 terminal hardware, however, does not allow control of beam intensity, so the special point-plot mode is functionally identical to the point-plot mode on the Type 3 terminal.

Incremental Plot Mode

The incremental plot mode allows the Type 3 terminal to simulate the operation of an incremental digital plotter on the display screen.

Graphics Input (GIN) Mode

The graphics input mode is interactive in that it involves requests for information and the operator's response to the requests. The usual blinking cursor is replaced by the cross-hair cursor - two blinking, intersecting lines. The lines meet at a point which is recorded as a pair of X,Y coordinates.

The GIN Mode can only be accessed by an applications program. Then the cross-hair cursor can be positioned with the touchpanel or the keypad keys.

The touchpanel is divided into 256 areas, each 32 dots by 32 dots. When one area of the touchpanel is touched, the cursor is positioned at the center of the area.

The numeric keypad is used to move the cursor. The 1 key moves the cursor down and to the left; the 2 key moves it down; the 3 key moves the cursor down and to the right; the 4 key moves it left; the 6 key moves it right; the 7 key moves it up and to the left; the 8 key moves it up; the 9 key moves it up and to the right. (Refer to table 2-1.) Pressing a key on the keypad while holding down the CTRL key moves the cursor 64 dots. Pressing a keypad key while holding down the SHIFT key moves the cursor eight dots. Pressing an unshifted key on the numeric keypad moves the cursor one dot. When the cursor is in the position you want, press any tan non-keypad key to record its position.

TABLE 2-1. GIN-MODE CURSOR-POSITIONING KEYS

KEYPAD KEY PRESSED	DIRECTION OF CURSOR MOVEMENT
1	Down and Left
2	Down
3	Down and Right
4	Left
6	Right
7	Up and Left
8	Up
9	Up and Right

Block Mode

Block mode is used to write or erase rectangular areas. Two points are specified. These points are the bottom-left and upper-right corners of the rectangle that the graphics option displays on the CRT. Only applications programs can access block mode.

Using the Graphics Option

After you have loaded your applications program and the graphics option, follow these steps to use the graphics option:

1. If your applications program does not call up the graphics option, type GRAPHICS ON, then press NEXT. These things happen:

   - The screen is erased.
   - The terminal is in alpha mode.
   - The cursor is in the home position (upper-left corner).
   - Size 1 characters will be displayed.

2. If your applications program requires input, use the tan keyboard keys in alpha mode; the touchpanel or the keypad in GIN mode.

3. If you have a Type 1 or a Type 4 printer, your applications program can use the printer. Be sure it is on, has paper, and is ready. If your applications program does not direct printing, you can press the terminal PRINT key to print the contents of a screen display.

   CAUTION

   If the next step is done, wait until the CP/M prompt (>) appears before continuing keyboard input. Pressing a key before the prompt appears may cause the system to get hung up and require a reset.

4. If your applications program does not exit graphics mode, type GRAPHICS OFF, then press NEXT.

   NOTE

   If you want to use the graphics option with several applications, type GRAPHICS ON DEFAULT, then press NEXT. Then the graphics option is active whenever you load. When you are finished, type GRAPHICS OFF DEFAULT, then press NEXT.

   You can load the graphics option on your system, then use it with a remote host by typing GRAPHICS STANDALONE, then pressing NEXT.

If you try to initialize graphics and your terminal does not have the graphics option, the message "Graphics not available" is displayed on the screen.

MODEM OPTION

When the terminal includes the optional internal 1200/1200-baud modem, it provides access to the PHONE utility feature of CP/M. The utility can be used to initiate computer-to-computer linkups for data transmission, to provide online access with the terminal functioning as part of an online network, or to initiate human-to-human communication via the terminal itself.

Further information, including instructions for using the PHONE utility, is given in section 11 of this manual.

This section provides the procedure for installing CP/M if your Control Data 110 has only a Type 1 disk drive. You will need a supply of blank, formatted, double-density, double-sided flexible disks. Your 110 Operating Guide includes instructions for formatting flexible disks, or you may refer to section 12 of this manual. You will first make two copies of the CP/M master flexible disk (the backup copy and a second copy). Then you will verify both copies.

Use this section if you have a Control Data 110 system that has one flexible-disk drive. If your system has two flexible-disk drives, don't use this section, but go to section 7 for your instructions.

MAKING TWO COPIES OF THE CP/M 2.2 MASTER FLEXIBLE DISK

Making copies on a Control Data 110 with one flexible-disk drive involves removing and inserting flexible disks for each file copied. Since the CP/M flexible disk contains many files, copying is a time-consuming process. If possible, make copies on another Control Data 110 with two flexible-disk drives. Contact your nearest CDC representative for assistance.

If you must use a single-drive system, power up your system and load the CP/M master flexible disk. (Refer to your 110 Operating Guide for instructions.) Then perform these steps:

1. Type: SYSGEN <next>

2. SYSGEN program starts and the display shows:

```
A>sysgen
CDC SYSGEN for CP/M 2.2
version 2.ØØ

Source drive ? (or RETURN to skip)
```

3. Type: A

Display shows:

```
A>sysgen
CDC SYSGEN for CP/M 2.2
version 2.ØØ

Source drive ? (or RETURN to skip)a
Put source disk on A, then type RETURN
```

4. Type: <next>

Program will operate for about 10 seconds. Then display shows:

```
A>sysgen
CDC SYSGEN for CP/M 2.2
version 2.ØØ

Source drive ? (or RETURN to skip)a
Put source disk on A, then type RETURN
Function completed
Destination drive ? (or RETURN to terminate)
```

5. Remove your master CP/M flexible disk from drive and set it aside.

Type: A

Display is updated as follows:

```
A>sysgen
CDC SYSGEN for CP/M 2.2
version 2.ØØ

Source drive ? (or RETURN to skip)a
Put source disk on A, then type RETURN
Function completed
Destination drive ? (or RETURN to terminate)a
Put destination disk on A, then type RETURN
```

6. Insert blank, formatted flexible disk in drive.

   Type: <next>

   Program operates about 10 seconds and then additional display is added as follows:

```
A>sysgen
CDC SYSGEN for CP/M 2.2
version 2.ØØ

Source drive ? (or RETURN to skip)a
Put source disk on A, then type RETURN
Function completed
Destination drive ? (or RETURN to terminate)a
Put destination disk on A, then type RETURN
Function completed
Destination drive ? (or RETURN to terminate)
```

7. At this point, CP/M 2.2 Operating System has been copied from tracks 0 and 1 of master flexible disk to tracks 0 and 1 of copy flexible disk. To copy remainder of master flexible disk, continue as follows.

8. Remove copy flexible disk from drive and insert your master CP/M flexible disk into drive.

   Type: <next>

   This returns control to CP/M 2.2 Operating System and your display shows:

```
A>sysgen
CDC SYSGEN for CP/M 2.2
version 2.ØØ

Source drive ? (or RETURN to skip)a
Put source disk on A, then type RETURN
Function completed
Destination drive ? (or RETURN to terminate)a
Put destination disk on A, then type RETURN
Function completed
Destination drive ? (or RETURN to terminate)

A>
```

9. Type: FILECOPY <next>

FILECOPY program produces the following display:

```
        Control Data 11Ø

        Single disk drive file copy program
        Copyright (C) 1981, Control Data Corporation

          This program copies files from one diskette
          to another utilizing only 1 disk drive.  The
          program asks for the destination and source
          filenames in the same manner as pip, format:
          DESTINATION = SOURCE.  Drive codes are
          required and generalized filename templates
          with imbedded * and ? are legal.  The
          following are valid command lines:

                  A: = A:*.*
                  c:test.abc = a:test.abc
                  C:x*.* = A:test*.*


              Filenames:
```

10. To start actual copying of files

Type: a:*.*=a:*.* <next>

Prompt and reply appear as:

```
        Filenames:a:*.*=a:*.*
```

11. Display indicates which flexible disk to insert. Type NEXT. Master is source flexible disk and copy is destination flexible disk. The system checks flexible disks and prevents use of wrong flexible disk. When copying is complete, display shows:

```
        Filenames:
```

12. Type: <next>

Display shows:

```
        Insert system disk, type CR to exit
```

13. Insert system flexible disk and press <next>. System returns to CP/M. Make a second copy by repeating steps 1 through 13.

Remove the master flexible disk from the drive and file it. The master CP/M flexible disk provides serial number identification and proof of ownership. The master flexible disk should not be used again unless the copies are damaged.

CAUTION

File the master flexible disk in a safe place that is free from magnetic fields.

The copies made in the procedure above are exact copies of the CP/M 2.2 flexible disk. These copies should be serialized with the number on the original master flexible disk and they are subject to the conditions of the CP/M 2.2 licensing agreement. Label one copy BACKUP MASTER CP/M 2.2 and file it in a safe place. Label the other copy WORKING CP/M 2.2. Be sure to put the supplied copyright labels on both of the flexible disks.

VERIFYING BOTH CP/M COPIES

When you have finished installing CP/M, run the verification test to assure a complete and reliable installation. Verify both the backup master and the working copy.

Load the copy of the CP/M 2.2 flexible disk before performing the installation tests.

1. Type:

   SUBMIT CPMTST01 <next>

   Test starts.

2. This test takes about 2-1/2 minutes to complete. At the end of a successful test, the display shows:

```
*****VERIFICATION TEST COMPLETE
A>ERA CPMTST.M01
A>ERA CPMTST.BAK
A>
```

If the test fails, the display shows CP/M error messages.

3. Repeat steps 1 and 2 to verify the second copy.

This section tells how to copy CBASIC, Pascal/M, and BASIC-80 on a CP/M 2.2 flexible disk on a Control Data 110 system with one flexible-disk drive. If you have both a Type 1 and a Type 2 disk drive, do not use this section. Refer to section 7 instead.

These programs are delivered on double-density, double-sided, flexible disks. These flexible disks are master flexible disks and should only be used once to make a backup master. Then file the master flexible disks in a safe place free from magnetic fields.

You will first copy CBASIC and verify that the copy is correct. Then you will copy PASCAL/M and verify it. You will copy BASIC-80 and verify that. Finally you will remove some files to increase working space on your disk.

COPYING CBASIC

Make a backup flexible disk of CBASIC. Follow this procedure:

1. Load CP/M from WORKING CP/M 2.2 disk.
2. Type: FILECOPY <next>

   Display shows:

   ```
   Filenames:
   ```

3. Type: A:*.*=A:*.* <next>
4. Follow displayed prompts.

VERIFYING CBASIC

Follow these steps to verify that CBASIC was correctly copied:

1. Type: SUBMIT CBASTST1 <next>

   Test starts. Test takes about 18 seconds.

2. At end of successful test, display shows:

```
A>ERA TESTCBAS.INT
A>
```

   If test fails, display shows CP/M error messages.

COPYING PASCAL/M

Make a backup flexible disk of Pascal/M. Follow this procedure:

1. Load CP/M from WORKING CP/M 2.2 disk.
2. Type: FILECOPY <next>

   Display shows:

```
Filenames:
```

3. Type: A:*.*=A*.* <next>
4. Follow displayed prompts.

VERIFYING PASCAL/M

This test is a Pascal/M program, which must be compiled and run to test the installation of Pascal/M. Follow these steps:

1. Type: PRUN PASCAL TESTP <next>

   Program is compiled.

2. Type: PRUN TESTP <next>

   Test starts.

3. At end of successful test, display shows a list of files followed by prompt: A>

   Display shows Pascal/M error messages if test fails.

COPYING BASIC-80

Make a backup flexible disk of BASIC-80. Follow this procedure:

1. Load CP/M from WORKING CP/M 2.2 disk.
2. Type: FILECOPY <next>

   Display shows:

```
        Filenames:
```

3. Type: A:*.*=A:*.* <next>
4. Follow displayed prompts.

VERIFYING BASIC-80

This test must be run to test the installation of BASIC-80.

CAUTION

Follow these instructions carefully.

1. Type: MBASIC <next>

   Message appears on display, last line of which is:

```
    OK
```

2. Type: LOAD "RANTEST <next>

Display shows:

```
OK
```

3. Type: EDIT 10 <next>

Display shows:

```
10
```

4. Press space bar to move cursor until following appears on screen:

```
OPEN "R",1,"
```

5. Type: cA <next>

6. Type: EDIT 77 <next>

Display shows:

```
77
```

7. Repeat steps 4 and 5.

8. Type: SAVE "RANTEST2" <next>

Display shows:

```
OK
```

9. Type: SYSTEM <next>

Display shows:

```
A>
```

10. Type: MBASIC RANTEST2 <next>

    Test starts. Series of consecutive numbers appear on the display.

11. At the end of a successful test, display shows:

```
A>
```

INCREASING WORKING SPACE

This procedure tells how to remove unnecessary material to gain working space. Before proceeding, you should verify the copy of CP/M 2.2 (see section 3). This test must be run before you erase any material from the disk.

Take the flexible disk labeled WORKING CP/M 2.2 and insert it into the drive. Then perform these steps:

1. Type: <CTRL C>

   Display shows:

```
    A>
```

   System can now erase files from WORKING CP/M 2.2 disk.

2. Type: ERA *.ASM <next>

   Excess files are erased from WORKING CP/M 2.2 disk.

3. If flexible disk contains CBASIC 2.38 and Pascal/M 4.01 as well as CP/M 2.2, continue erasing files.

   Type: ERA CONFIG.PCO <NEXT>
ERA CONFIG.DAT <NEXT>
ERA INSTALLP.SUB <NEXT>
ERA READ.ME <NEXT>

4. When the files are erased, prompt: A> appears on the display.

5. To verify that this copy is a correct working copy, reload your new disk.

This section includes a procedure for copying files and a procedure for transferring software from single- to double-density disks. The procedures in this section apply to a Control Data 110 with one flexible-disk drive. If you have both a Type 1 and a Type 2 disk drive, refer to section 8.

The following definitions apply:

- The source flexible disk is the flexible disk from which data is read.
- The destination flexible disk is the flexible disk to which data is written.

COPYING FILES

Copying files is a time-consuming process on a single-drive system. If possible, make copies on a Control Data 110 with two flexible-disk drives, or, purchase additional copies. Contact a Control Data sales representative for assistance.

If you must copy files with only a Type 1 disk drive, perform these steps:

1. Load CP/M 2.2 and type: FILECOPY <next>
2. FILECOPY program introduces itself and then displays:

```
Filenames:
```

3. If your source is double-density, type: a:*.*=a:*.* <next>

   Prompt appears:

```
Filenames:a:*.*=a:*.*
```

or

If your source is single-density, type a:*.*=c:*.*<next>

Prompt appears:

```
Filenames:a:*.*=c:*.*
```

5. Follow prompts that tell you which flexible disk to insert and when to type <next>. Program checks for correct flexible disk, and prevents system from operating with wrong flexible disk.

6. When copy is complete, program displays:

```
Filenames:
```

7. Type: <next>

   Program displays:

```
Insert system disk, type CR to exit
```

8. To return control to CP/M 2.2, type: <next>

For more information, refer to the FILECOPY program in section 12.

TRANSFERRING SOFTWARE FROM SINGLE- TO DOUBLE-DENSITY DISKS

Software purchased from a source that provides standard CP/M 2.2-compatible flexible disks is recorded on single-density, single-sided disks. This software must be transferred to a formatted, double-density, double-sided flexible disk and a backup copy made. Follow these steps:

1. Load CP/M 2.2 and type: FILECOPY <next>

   FILECOPY program displays:

```
Filenames:
```

2. Type: a:*.*=c:*.* <next>

   Display shows:

```
Filenames:a:*.*=c:*.*
```

3. When program asks for source flexible disk, remove CP/M 2.2 disk and insert source single-density flexible disk in drive.

4. Type: <next>

5. When program asks for destination flexible disk, remove source disk and insert blank, formatted, double-density, double-sided flexible disk.

6. Insert flexible disks according to prompts until program again displays:

```
Filenames:
```

7. Type: <next>

8. Insert CP/M 2.2 disk and type: <next>

To make another copy of the double-density flexible disk, follow the procedure described under "Copying Files" at the beginning of this section.

To examine this double-density copy of your purchased software, you must make it into a CP/M disk and put several CP/M files on it. Perform these steps:

1. Load CP/M and type: SYSGEN <next>

   Display includes:

```
Source drive ?
```

2. Type: A <next>

Display includes:

```
Destination drive ?
```

3. Type: A

4. Remove CP/M 2.2 disk and insert a blank, formatted flexible disk.

5. Type: <next>

   When transfer is complete, display shows:

```
Function completed
```

6. Remove flexible disk in drive and insert CP/M 2.2 disk.

7. To return control to CP/M 2.2, type: <next>

8. Use FILECOPY to transfer files PIP.COM, TERMINAL.COM TERMINAL.PRB, TERMINAL.ALT and STAT.COM to new flexible disk. Insert CP/M and destination disks as prompts indicate. Answer filename prompts as follows:

```
Filenames:a:=a:pip.com <next>
Filenames:a:=a:stat.com <next>
Filenames:a:=a:terminal.* <next>
Filenames: (Type:  <next> at this point)
```

The application flexible disk now is a CP/M 2.2 flexible disk, which can be loaded and executed. To verify that this copy is a correct working copy, reload your new disk.

It is now possible to list the directory, use PIP to list files on the printer, and use STAT *.* to see the size of the files.

This section provides the procedure for copying CP/M if your Control Data 110 has both a Type 1 and a Type 2 disk drive. You will need a supply of blank, formatted, double-density, double-sided flexible disks. Your 110 Operating Guide includes instructions for formatting flexible disks, or you may refer to section 12 of this manual. You will first make two copies of the CP/M master flexible disk (the backup copy and a second copy). Then you will verify both copies.

Use this section if you have a Control Data 110 system that has two flexible-disk drives. If your system has only one flexible-disk drive, don't use this section, but go to section 3 for your instructions.

The following definitions apply:

- The source flexible disk is the flexible disk from which data is read.
- The destination flexible disk is the flexible disk to which data is written.

MAKING TWO COPIES OF THE CP/M 2.2 MASTER FLEXIBLE DISK

The most efficient way to make a copy of your CP/M master flexible disk is to use the utility DSKUTIL. Load the CP/M master flexible disk and then follow these steps:

1. Type: DSKUTIL <next>

   DSKUTIL program starts and display shows:

```
Control Data 110

DISK UTILITY
Version 2.09

This program allows you to:

Make a verified track for track copy of a flexible
disk using two disk drives.

 Verify a copy made by this program.
 Analyze a diskette for bad sectors.
**************************************************

Enter the number of the routine you desire.
1.     Track for track copy.
2.     Track for track verify.
3.     Disk analysis for bad sectors.
4.     Return to CP/M.  (Be sure system disk is in A)
```

2. Type: 1

   Display shows:

```
Disk copy utility.

This utility copies and verifies a disk
Mount original on drive A (Primary disk drive)
Mount blank copy on drive B (Secondary disk drive)
Type character when ready.
```

3. Insert blank, formatted flexible disk into Type 2 disk drive.

   Type: <next>

   Program takes about four minutes to make and verify the copy. Then display shows:

```
Disk copy utility.

This utility copies and verifies a disk
Mount original on drive A (Primary disk drive)
Mount blank copy on drive B (Secondary disk drive)
Type character when ready.


Unit A: is Double Sided
Unit B: is Double Sided
Double Density Verify

ROUTINE completed  ************************************


Enter the number of the routine you desire.
1.    Track for track copy.
2.    Track for track verify.
3.    Disk analysis for bad sectors.
4.    Return to CP/M.  (Be sure system disk is in A)
```

4. Although the copy routine did verify the copy, you can double-check by typing: 2

   Display shows:

```
Disk verify utility.

Mount original on drive A (Primary disk drive)
Mount blank copy on drive B (Secondary disk drive)
Type character when ready.
```

5. Type: <next>

   Program operates for about three minutes and then display shows the following:

```
Disk verify utility.

Mount original on drive A (Primary disk drive)
Mount blank copy on drive B (Secondary disk drive)
Type character when ready.


Unit A: is Double Sided
Unit B: is Double Sided
Double Density Verify

ROUTINE completed  ************************************


Enter the number of the routine you desire.
1.     Track for track copy.
2.     Track for track verify.
3.     Disk analysis for bad sectors.
4.     Return to CP/M.  (Be sure system disk is in A)
```

The preceding procedure provides one copy of the master CP/M 2.2 flexible disk. Make a second copy by repeating the procedure. Remove the master flexible disk from the Type 1 drive and file it. The master CP/M flexible disk provides serial number identification and proof of ownership. The master flexible disk should not be used again unless the copies are damaged.

CAUTION

File the master flexible disk in a safe place that is free from magnetic fields.

The copies made in the preceding procedure are exact copies of the CP/M flexible disk. These copies are serialized with the number on the original master flexible disk and they are subject to the conditions of the CP/M 2.2 licensing agreement. Label one copy BACKUP MASTER CP/M 2.2 and file it in a safe place. Label the other copy WORKING CP/M 2.2. Be sure to put the supplied copyright labels on both of the flexible disks.

VERIFYING CP/M

When you have finished copying CP/M, run the verification test to assure a complete and reliable installation. This test must be run before you remove any files from your CP/M 2.2 flexible disk. Verify both the backup master and the working copy.

Load the copy of CP/M 2.2 disk, then perform these steps:

1. Type: SUBMIT CPMTST01 <next>

   Test begins.

2. Test takes about 2-1/2 minutes to complete. At end of successful test, display shows:

```
***** VERIFICATION TEST COMPLETE
A>ERA CPMTST.M01
A>ERA CPMTST.BAK
A>
```

   If test fails, display shows CP/M error messages.

3. Repeat steps 1 and 2 for the second copy.

This section tells how to copy CBASIC, Pascal/M, and BASIC-80 on a CP/M 2.2 flexible disk on a Control Data 110 system with two flexible-disk drives. If your system has only one flexible-disk drive, refer to section 4.

These programs are delivered on double-density, double-sided, flexible disks. The flexible disks are master flexible disks and should only be used once to make a backup master. Then file the master flexible disks in a safe place free from magnetic fields.

You will first copy CBASIC and verify that the copy is correct. Then you will copy PASCAL/M and verify it. You will copy BASIC-80 and verify that. Finally, you will increase working space on your disk by removing some files.

The following definitions apply:

- The source flexible disk is the flexible disk from which data is read.
- The destination flexible disk is the flexible disk to which data is written.

COPYING CBASIC

First make a backup copy of the master CBASIC flexible disk as described in section 8. Then follow this procedure:

1. Put WORKING CP/M 2.2 disk in Type 1 disk drive and backup CBASIC disk in Type 2 disk drive.
2. Type: pip a:=b:*.* <next>

   CBASIC 2.38 files transfer to CP/M 2.2 flexible disk.

VERIFYING CBASIC

Follow these steps to verify CBASIC installation:

1. Type: SUBMIT CBASTST1 <next>

   Test begins.

2. Test takes about 18 seconds. At end of successful test, display shows:

```
A>ERA TESTCBAS.INT
A>
```

   If test fails, display shows CP/M 2.2 error messages.

COPYING PASCAL/M

Make a backup Pascal/M flexible disk as described in section 8. Then follow these steps:

1. Put WORKING CP/M 2.2 disk in Type 1 disk drive and put backup Pascal/M 4.01 disk in Type 2 disk drive.

2. Type: SUBMIT B:INSTALLP <next>

   Pascal/M files transfer to CP/M 2.2 flexible disk. A test program is compiled and run. Terminal displays general information about Pascal/M system.

VERIFYING PASCAL/M

This test is a Pascal/M program which must be compiled and run to test the installation of Pascal/M.

1. Type: PRUN PASCAL TESTP <next>

   Program is compiled.

2. Type: PRUN TESTP <next>

   Test starts.

3. At end of successful test, display shows list of files followed by:

```
A>
```

   If test fails, display shows Pascal/M error messages.

COPYING BASIC-80

Make a backup BASIC-80 flexible disk as described in section 8. Then follow these steps:

1. Put WORKING CP/M 2.2 disk into Type 1 disk drive and backup BASIC-80 disk into Type 2 disk drive.

2. Type: PIP a:=b:*.* <next>

   BASIC-80 files transfer to CP/M 2.2 flexible disk.

VERIFYING BASIC-80

Follow these steps to verify BASIC-80 installation:

1. Load CP/M from Type 1 disk drive.

2. Insert BASIC-80 backup flexible disk in Type 2 disk drive.

3. Type: B:MBASIC B:RANTEST <next>

   Test begins. Series of consecutive numbers appears on display. At end of successful test, display shows:

```
A>
```

INCREASING WORKING SPACE

This procedure tells how to remove unnecessary material from your CP/M disk to gain additional working space. Before doing this, verify the CP/M 2.2 copy (see section 6). The test will not work if you have erased any material from your CP/M disk.

Take the flexible disk labeled WORKING CP/M 2.2 and insert it into the Type 1 disk drive. Follow these steps:

1. Type: <CTRL C>

   Display shows:

   ```
   A>
   ```

   System can now erase files from WORKING CP/M 2.2 disk.

2. Type: ERA *.ASM <next>

   Excess files are erased from WORKING CP/M 2.2 flexible disk.

3. If system flexible disk contains CBASIC and PASCAL/M as well as CP/M 2.2, continue erasing files.

   Type: ERA CONFIG.PCO <next>
   ERA CONFIG.DAT <next>
   ERA INSTALLP.SUB <next>
   ERA READ.ME <next>
   ERA CBASTST1.SUB
   ERA TESTCBAS.INT

4. When the files are erased, the A> prompt appears on display.

5. If you need to make more copies of WORKING CP/M 2.2 flexible disk, use DSKUTIL utility.

6. To verify that this copy is a correct working copy, reload your new disk.

This section includes a procedure for copying files and a procedure for transferring software from single- to double-density flexible disks. The procedures in this section apply to a Control Data 110 with two flexible-disk drives. If you have only a Type 1 disk drive, refer to section 5.

The following definitions apply:

- The source flexible disk is the flexible disk from which data is read.
- The destination flexible disk is the flexible disk to which data is written.

Three cases are considered:

- Copying files from a flexible disk to the working CP/M disk.
- Copying files from a double-density non-CP/M disk to another double-density non CP/M disk.
- Transferring software from single- to double-density disk.

COPYING FROM A FLEXIBLE DISK TO THE WORKING CP/M DISK

To copy all the files from one flexible disk to the working CP/M disk, perform Steps 1 and 2. To copy a single file, perform steps 3 and 4. Note that any file on the destination disk with the same name as a file on the source disk is replaced.

1. Load CP/M from Type 1 disk drive. Insert source flexible disk into Type 2 disk drive.

2. Type: pip a:=b:*.*[v] <next>

   This transfers all files from flexible disk in Type 2 disk drive to CP/M flexible disk in Type 1 disk drive.

3. To copy an individual file to system flexible disk, place source flexible disk in Type 2 disk drive and load CP/M from Type 1 disk drive.

4. Type PIP A:=B:, then file name, and then <next>. For example, to copy TERMSET.COM file:

   Type: PIP A:=B:TERMSET.COM <next>

COPYING FROM A DOUBLE-DENSITY NON-CP/M DISK TO ANOTHER DOUBLE-DENSITY NON-CP/M DISK

To copy files from a double-density non-CP/M flexible disk (such as a data flexible disk) to another double-density non-CP/M flexible disk (a new, formatted flexible disk), do the following:

1. Place destination flexible disk in Type 2 disk drive and CP/M 2.2 flexible disk in Type 1 disk drive.

2. Type: <control C>

3. Type: pip <next>

4. PIP program loads and display shows:

```
        *
```

5. Remove CP/M 2.2 flexible disk from Type 1 disk drive and put source flexible disk in Type 1 disk drive.

6. Type: b:=a:*.*[v] <next>

7. All files on source disk in Type 1 disk drive transfer to destination disk in Type 2 disk drive. System verifies transfer.

8. PIP program displays the * prompt. Place CP/M flexible disk in Type 1 disk drive and type: <next>

   Control returns to CP/M.

The flexible disk in the Type 2 disk drive is now a copy of the source flexible disk.

TRANSFERRING SOFTWARE FROM SINGLE- TO DOUBLE-DENSITY DISK

Software purchased from a source that provides standard CP/M 2.2 compatible flexible disks is recorded on single-density, single-sided disks. This software must be transferred to a formatted, double-density, double-sided flexible disk and a backup copy made. Perform these steps:

1. Load CP/M from Type 1 disk drive.

2. Place a blank, formatted, double-density, double-sided flexible disk in Type 2 disk drive. Then type: <CTRL C>

3. To transfer all files from single-density source disk to double-density destination disk in Type 2 disk drive, type: PIP <next>

4. PIP program displays the * prompt. Remove CP/M disk from Type 1 disk drive and insert single-density source disk into Type 1 disk drive. Then type: B:=C:*.*[v] <next>

This transfers all the files on the source flexible disk to the double-density flexible disk.

File your single-density flexible disk in a safe place and use the double-density copy as a master copy. Many application flexible disks have xxx.SUB files to be used for installation. These should operate correctly using the double-density flexible disk as the source.

To make this double-density copy into a CP/M 2.2 flexible disk that can be loaded and executed, use the following procedure:

1. Load CP/M from Type 1 disk drive and put new double-density application disk in Type 2 disk drive. Then type: <CTRL C>

2. Type: SYSGEN <next>

3. Answer the prompts from SYSGEN one at a time as follows:

   A <next>
   B <next>
   <next>

4. Transfer necessary system files using PIP.

   Type: PIP B:=PIP.COM <next>
   PIP B:=STAT.COM <next>
   PIP B:=TERMINAL.* <next>

Your application flexible disk can now be loaded and executed. To verify that this copy is a correct working copy, reload your new disk.

Now you can list the directory, use PIP to list files on your printer, and use STAT *.* <next> to see the size of the files.

The format utility is used to format and analyze the rigid disk. It also permits you to see the total of disk surface faults. This section provides a brief description of the utility and procedures for using it. To use the format utility you must have the following equipment:

- Type 3 terminal
- Type 1 disk subsystem
- Rigid-disk interface option board
- Between one and four Type 3 or Type 4 rigid-disk subsystems

Each rigid disk is divided into two or three fixed size logical subdisks. A 12.5 megabyte rigid disk consists of one 8 Mb (8.38 megabyte) logical disk and one 4 Mb (4.19 megabyte) logical disk. A 25 megabyte rigid disk consists of three 8 Mb (8.38 megabytes) logical disks. Each logical disk has a specific CP/M disk designator (from E to P). Every rigid disk occupies three CP/M disk designators regardless of disk size. On 12.5 megabyte rigid disks the third disk designator is not used. The following is an example of disk designators for four separate rigid disks.

DISK E, F	12.5 megabyte (8 Mb/4 Mb)
DISK H, I, J	25 megabyte (4 Mb/8 Mb/8 Mb)
DISK K, L	12.5 megabyte (8 Mb/4 Mb)
DISK N, O, P	25 megabyte (8 Mb/8 Mb/8 Mb)

It takes about 1 hour and 48 minutes to format and analyze an entire 12.5-Mb rigid disk, and 3 hours and 36 minutes to format and analyze an entire 25-Mb rigid disk. The system displays a message during any lengthy process, telling you what part of the utility process it is performing.

While formatting a logical disk, the entire 512/256 tracks are formatted including the CP/M system information area.

While reformatting a sector, the sector is read in first. If no error occurs during the read, the system does not allow the reformat. If the sector is reformatted, all previous data in the reformatted sector is destroyed.

While reformatting a file, only a file space is reformatted and the file is deleted.

Any errors that may be detected while the utility is executing are displayed. You may then press NEXT to retry the step; BACK to return to the last display; or press the CTRL and C keys to exit the utility.

Five procedures for using the format utility are provided:

- formatting an entire rigid disk
- formatting a logical disk
- reformatting a single file
- reformatting a single sector
- accessing the error log

FORMATTING AN ENTIRE RIGID DISK

Remember that formatting destroys any data on the area being formatted. Use the following procedure for formatting an entire rigid disk:

1. Type: diskinit <next>

   Diskinit program starts and display shows main menu (figure 9-1)

```
Control Data 110

DISKINIT
Version 2.00

    ****************************************************

             You may select one of the following options:

                1. Format
                2. Error Report
                3. Return to CP/M

             Enter your choice... >
```

Figure 9-1. Main Menu

2. To format, type: 1

   The following prompt appears:

```
Enter the disk code of device to format
(From E to P) or press BACK to return to menu ...>
```

3. Type the letter designating the disk you want to format. A message then appears showing the disk code of the disk to be formatted and the capacity of the disk. For example, if E is selected and it is part of a 12.5-Mb rigid-disk drive, the following message appears:

```
***E - 8-megabyte logical disk***
Press BACK to retry or NEXT to continue.>
```

4. Type: <next>

   Display shows the format menu (figure 9-2):

```
You may choose one of the following formats:

   1. Full disk format.
   2. Format a logical disk
   3. Reformat a file.
   4. Reformat a sector.
   5. Return to previous option menu.

Enter the number of your choice...>
```

Figure 9-2. Format Menu

5. Type: 1 <next>

   The following warning appears:

```
  *  *  *  *  *        WARNING        *  *  *  *  *

     FORMATTING DESTROYS ALL PREVIOUSLY EXISTING DATA.

     Press BACK to retry or NEXT to continue. >
```

6. Verify that you have no needed data on the disk.

7. If you want to save data on the disk, press BACK to return to utility Main Menu. If you do not want to save data on the disk, press NEXT.

   This displays a message like the following:

```
 The Format and Surface Analysis
 of your 12.5 megabyte Rigid Disk
      will take approximately
       1 hour and 48 minutes

Press NEXT to continue or BACK to return to menu. >
```

8. If you do not want to format, press BACK to return to main menu. If you want to format, press NEXT.

   This displays:

```
              FORMATTING IS IN PROGRESS

    * * This will take approximately 3 minutes * *

                 PLEASE BE PATIENT...
```

9. Surface analysis automatically follows formatting. When surface analysis is underway, a message like the following appears:

```
Surface analysis is now in progress for logical disk E;
This will take approximately 1 hour and 10 minutes.


     TOTAL NUMBER OF TRACKS TO ANALYZE...   512

        NUMBER OF TRACKS COMPLETED...    XX
```

   When the current logical disk (disk E in this example) has been surface analyzed, the analysis of the next logical disk automatically begins.

10. When format and analysis is complete, a report like the following appears:

```
                    ANALYSIS REPORT

Logical Disk E-0 bad sectors encountered and disabled

Logical Disk F-XX bad sectors encountered and disabled

FULL DISK FORMAT OPERATION IS COMPLETE.

Press NEXT to continue. >
```

If 10 percent or more of the disk surfaces are unusable, the following warning accompanies the report:

```
                * * * * WARNING * * * *


   XX% of your disk surface is unusable.  Your disk may
   require repair.
```

FORMATTING A LOGICAL DISK

Remember that formatting destroys all data on the logical disk being formatted. Use the following procedure for formatting a logical disk:

1. Type: diskinit <next>

   Diskinit program starts and display shows main menu (figure 9-1).

2. To format, type: 1 <next>

   The following prompt appears:

```
   Enter the disk code of the device to format (from E to
   P) or press BACK to return to menu...>
```

3. Type the letter designating the disk you want to format. A message appears showing the disk code and capacity of the disk. For example, if you select disk E, the following message appears:

```
***E - 8-megabyte logical disk***
Press BACK to retry or NEXT to continue>
```

4. Type: <next>

   Display shows the format menu (figure 9-2).

5. Type: 2 <next>

   The following warning appears:

```
             * * * * * WARNING * * * * *
FORMATTING DESTROYS ALL PREVIOUSLY EXISTING DATA.
Press BACK to retry or NEXT to continue.>
```

6. Verify that you have no needed data on the disk.

7. If you have needed data on the disk, press BACK to return to utility Main Menu. If you do not have needed data on the disk, press NEXT.

   This displays:

```
          The Format and Surface Analysis
          of your 8-megabyte Logical Disk
              will take approximately
               1 hour and 17 minutes

Press NEXT to continue or BACK to return to menu.>
```

8. If you do not want to format, press BACK to return to main menu. If you do want to format, press NEXT.

   This displays:

```
FORMATTING IS IN PROGRESS

**This will take approximately 7 minutes**

PLEASE BE PATIENT...
```

9. Surface analysis automatically follows formatting. When surface analysis is underway, a message like the following appears:

```
Surface analysis is now in progress for logical disk E;
This will take approximately 1 hour and 10 minutes.

      TOTAL NUMBER OF TRACKS TO ANALYZE... 512
          NUMBER OF TRACKS COMPLETED... XX
```

10. When format and analysis is complete, the system displays one of the following messages:

```
              ANALYSIS COMPLETE
    XX bad sectors encountered and disabled
```

(or, if 10 percent or more of the logical disk is unusable)

```
XX bad sectors encountered and disabled.


          * * * * WARNING * * * *

      XX% of your disk surface is unusable.
          Your disk may require repair.
```

REFORMATTING A SINGLE FILE

Remember that reformatting destroys all data in the file being reformatted. Use the following procedure for reformatting a single file on a rigid disk:

1. Type: diskinit <next>

   Diskinit program starts and display shows main menu (figure 9-1).

2. To format, type: 1 <next>

   The following prompt appears:

```
Enter the disk code of device to format (from E to P)
or press BACK to return to menu...>
```

3. Type a letter designating the disk you want to format. Message appears showing the disk code of the disk to be formatted and capacity of the disk. For example, if F is selected and it is part of a 12.5-Mb rigid-disk drive, message appears as follows:

```
***F - 4-megabyte logical disk***
Press BACK to retry or NEXT to continue.>
```

4. Type: <next>

   Display shows the format menu (figure 9-2).

5. Type 3 <next>

   The following prompt appears:

```
Enter the name of the file to be reformatted;
(do NOT include the disk code) ... >
```

6. Type name of file to be reformatted. Do not include disk code (E to P).

   The following warning appears:

```
* * * * *            WARNING            * * * * *

FORMATTING DESTROYS ALL PREVIOUSLY EXISTING DATA.

Press BACK to retry, or NEXT to continue. >
```

7. Verify that you have no needed data on the file.

8. If you do have needed data on the file, press BACK to return to main menu. If you do not have needed data on the file, press NEXT.

   This displays a message like the following:

```
The name of the file to be reformatted is

              YOURFILE.TYP

If the entry is correct, press NEXT to proceed.
If you wish to retry, press BACK.
```

9. Verify that filename is correct and that file contains no essential data.

10. If filename is incorrect or if file should not be reformatted, press BACK. To format, press NEXT.

    While the file is being reformatted, a message like the following appears:

```
      Reformatting File YOURFILE.TYP...
will take approximately X hour X minute X second
```

11. When reformatting is complete, a message like the following appears:

```
              ANALYSIS COMPLETE
    0 bad sectors encountered and disabled

     FILE REFORMAT OPERATION IS COMPLETE.

            Press NEXT to continue
```

REFORMATTING A SINGLE SECTOR

Remember that reformatting destroys any data in the area being reformatted. Use the following procedure for reformatting a single sector on a rigid disk:

1. Type: diskinit <next>

   Diskinit program starts and display shows main menu (figure 9-1).

2. To format, type: 1 <next>

   Display shows:

```
Enter the disk code of device to format (from E to
P)...>
```

3. Type the letter designating the disk you want to format. Message appears showing the disk code of the disk to be formatted and capacity of the disk. For example, if H is selected and it is part of a 25-Mb rigid-disk drive, the following message appears:

```
*** H - 8-megabyte logical disk ***
Press BACK to retry or NEXT to continue.>
```

4. Type <next>

   Format menu (figure 9-2) appears.

5. Type: 4 <next>

   The following prompt appears:

```
Enter the track and sector number to be reformatted...

                Track (0 through 511) = >XXX

               Sector (0 through 127) = > XX
```

6. Type the track number, then sector number to be reformatted. The following warning appears:

```
* * * * *            WARNING            * * * * *

  FORMATTING DESTROYS ALL PREVIOUSLY EXISTING DATA.

  Press BACK to retry, or NEXT to continue.>
```

7. Verify that the sector does not contain needed data. If the sector contains needed data, press BACK. To reformat, press NEXT.

8. When reformat and analysis is complete, one of the following displays appears:

```
    Sector was read without error.
    *** SECTOR WILL NOT BE REFORMATTED ***
```

(or)

```
    Sector number XX has been reformatted.
```

(or)

```
    BAD SECTOR ENCOUNTERED

    Track...XX, Sector...XX was disabled

    ***********************************

         SECTOR XX WAS DISABLED...
         REFORMATTING COMPLETE.

    ***********************************

    Press NEXT to continue
```

ACCESSING THE ERROR LOG

The error log for a rigid disk shows the number of read/write errors that have occurred on all of its logical disks. Use the following procedure to access the error log for a rigid disk:

1. Type: diskinit <next>

2. Diskinit program starts and display shows main menu (figure 9-1).

3. To access error counter, type: 2 <next>

   The following prompt appears:

```
Enter the disk code of device to check (from E to P)... or
press BACK to return menu.>
```

4. Type the letter designating the disk whose error count you want to see. A message then appears showing the disk code of the disk and its capacity. For example, if E is selected, the following message appears:

```
    *** E - 8-megabyte logical disk ***
    Press BACK to retry or NEXT to continue.>
```

5. Type: <next>

   If the disk code entered was E and the rigid disk is a 12.5 Mb unit, an error count appears like the following:

```
    Total Read/Write  Errors for Disk E and F = XXXX

    Press NEXT or BACK to return to main menu.
```

6. Type: <next>

   Main menu appears (figure 9-1).

7. To return to CP/M, type: 3

The BACKUP utility is used to copy files from the rigid disk to flexible disks for backup storage. In addition, it permits you to see information about the backup itself, about the backed up files, and about the recovery of backed up information from flexible disks to the rigid disk.

The utility also offers a verification option. If you choose the option, the system verifies the file after backing it up to the flexible disk. The system displays the status of the verification, and stops the BACKUP utility if any errors are detected during verification. When this happens, the backup disk is good for the files that were backed up before the error occurred.

This section provides a brief description of the utility and the procedures for using it. To use the BACKUP utility, you must have the following equipment:

- Type 3 terminal
- Type 1 disk subsystem
- Rigid disk option
- Between one and four Type 3 or Type 4 rigid disk subsystems

It takes about 10 minutes to transfer 1.2-Mb of data (the contents of one double-density, double-sided flexible disk, approximately) without the verification option. With the verification option, the process takes about 14 minutes. You will notice that the system displays a message during any lengthy process. This message tells you what part of the utility process the system is performing.

Many of the displays you will see include the statement "HELP is available." You may press the HELP key to see more information about the displayed material. The backup process requires newly formatted flexible disks to store back-up files. Newly formatted flexible disks should be at hand before you begin the operation.

Any errors that may be detected while the utility is executing are displayed. You may then press NEXT to retry the step; BACK to return to the last display; or CTRL-C to exit the utility. You can also press the HELP key to learn more about the error message. Note that CTRL-C will only work if you place the system disk in the Type 1 drive.

CAUTION

> Although CTRL-C will work anywhere in the BACKUP utility, using it during a backup or recovery option may cause damage to the files being worked with.

Procedures for the following are provided:

- Backup using directory option
- Backup using text file option
- Continue a backup that was interrupted
- See backup information
- Restore files to rigid disk

BACKUP USING DIRECTORY OPTION

This option shows you a list of the files that are on any rigid disk in the system. Then you can choose the files from that disk that you want to back-up. Use the following procedure to create backups using a disk directory.

Because of the efficient use of the flexible disks by this program, the order that they are transferred will not be the order in which they reside on the rigid disk.

"HELP is available" indicates that you may press the HELP key for assistance.

Follow these steps to make backups using the directory:

1. Type: BACKUPS <next>

   Backup program starts and display shows main menu (figure 10-1).

```
                                          HELP is available.

        CONTROL DATA CORPORATION 110 BACKUP SYSTEM


  1. Go to backup creation menu.

  2. Continue a backup operation that was interrupted.

  3. See backup disk information.

  4. Restore backed up files from flexible disk to rigid
     disk.

  5. Return to CP/M (BACK key also returns).

Type the number of the option you want to use and press NEXT.
>
```

Figure 10-1. Main Menu

2. Type: 1 <next>

   Display shows backup creation menu (figure 10-2).

```
  0 Files currently selected              HELP is available.


                      BACKUP CREATION


  1. Select files from a disk directory.

  2. Select files named in a text file.

  3. Backup the selected files to flexible disk.
     (Will only work if > 0 and < 249 files selected)

  4. Save the selected file names in a text file.
     (Will only work if > 0 and < 249 files selected)

  5. Return to the main menu without doing anything.
     (BACK key will also return to the main menu)

Type the number of the option you want to use and press NEXT.
> 1
```

Figure 10-2. Backup Creation Menu

3. Type: 1 <next>

Display shows disk select message (figure 10-3).

```
                                          HELP is available.

      Specify a disk from which to select files.

    Type a valid disk letter and press NEXT.
    Press BACK to leave this option.
    Valid disk letters are from E to P.
    >
```

Figure 10-3. Disk Select Message

4. Select letter (from E to P) designating the logical disk from which you want to select files.

   Type: the selected letter <next>

   This displays:

```
    Please wait.  I am scanning the disk directory.
```

5. The names of the first 32 files on the selected disk are then displayed in a list like the one shown in figure 10-4.

```
SELECT FILES FOR BACK UP ON FLEXIBLE DISK(S) HELP is available.
0 Files currently selected                   Disk E

   # NAME      TYPE               # NAME      TYPE
    0 PIP       COM
    1 DISKINIT  COM
    2 STAT      COM
    3 BACKUPS   COM

Press DATA to see the file sizes in sectors.

Select a file, press HELP to find out how.  Type a disk letter
then a ':' to select a different disk.  Type '+' or '-' to page
forward or backward.  Press BACK to return to the previous
menu.
> 1,2,3
```

Figure 10-4. Example File-Select List

6. To see the next page of the disk directory, type: + <next>

   To page backward, type: - <next>

   To see files on another rigid disk, type: the letter designating that disk-drive unit, then ':'

   To see lengths of files, type: <data>

   You can only type one of these commands per line.

7. Once you are ready to select a file, enter selection options in a string separated by commas or by spaces. You may select files by number, or by name and type, or you may select all files on the disk, as follows:

   - NUMBER: To select a single file, type its number.

   - NUMBER-NUMBER: To select a set of files numbered in sequence, type first number, then -,then last number (example 3-11 for files 3 through 11). The last number may be a number larger than the number of files (2-999 would select file 2 through the last file).

   - FILENAME.TYPE: To select a file or set of files using generic filenames or types, type the filename or filetype. Do not use disk-designating letters with filenames.

   - *.* To select all files on the disk, type: *.*

   NOTE

   Files are deselected by entering the number or filename/filetype of a previously selected file. The first two characters or a filename cannot be 00-99. If two disks have the same filename, one of the files may not be selected for backup.

8. Press <next>. Display is updated after each selection. Selected files have * before filename.

9. When you have finished selection, press <back>. Backup creation menu (figure 10-2) appears. You may select more files using either option.

10. It is good practice to save names of selected files in a text file. To do this, return to backup creation menu (figure 10-2) and type: 4 <next>

This displays the write-to-text file message (figure 10-5).

```
                                             HELP is available.

   WRITE NAMES OF SELECTED FILES TO A TEXT FILE

   Type in the name of the file to write to
   and press NEXT.
   Press BACK to leave.
   >
```

Figure 10-5. Write-to-Text File Message

Type the disk designating letter and filename you have chosen for text file and then type <next>

This displays:

```
   Writing the filenames to disk.  Please wait.
```

Then this line is added:

```
   List written to file.  Please press NEXT.
```

When the above line appears, type: <next>

11. When you are ready to backup files, type 3 <next>

This displays:

```
   Hold on please I am getting file lengths from disk.
```

12. When the system has finished getting file lengths from the disk, the following appears:

```
|-------------------------------------------------------------|
|                                          HELP is available.|
|                                                             |
|    Please enter today's date in MM/DD/YY format.            |
|    >                                                        |
|-------------------------------------------------------------|
```

13. Type the date as follows:

    a. Month - 2 digits - use leading 0 for numbers 1 through 9

    b. Slash

    c. Day - 2 digits - use a leading 0 for numbers 1 through 9

    d. Slash

    e. Year - 2 digits - use last two digits of year

    If 06/21/83 is entered, display echoes as follows:

```
|-------------------------------------------------------------|
|        > 06/21/83                                           |
|-------------------------------------------------------------|
```

14. Type: <next>

    This displays:

```
|-------------------------------------------------------------|
|                                          HELP is available.|
|                                                             |
|    Please enter the current time in HH:MM format.           |
|    >                                                        |
|-------------------------------------------------------------|
```

15. Enter the time as follows:

    a. Hour - 2 digits - use leading 0 for numbers 1 through 9. Use 24-hour clock in which 1 pm is 13:00.

b. Colon

c. Minutes - 2 digits - use leading 0 for numbers 1 through 9

If 9 a.m. is entered (09:00), display echoes as follows:

```
> 09:00
```

16. Type: <next>

    This displays:

```
                                    HELP is available.

Please enter a backup ID (up to 8 characters).
>
```

17. Type your own chosen identifier for the backup as a whole. Use 8 characters or less. If 110unit1 is entered, display echoes as follows:

```
> 110unit1
```

18. Type: <next>

    This displays:

```
After backing up the files, do you want the
files verified (y or n)?
```

19. Type: y or n <next>

    This displays:

```
HELP is available.

What kind of diskette are you going to use?

1. Single-density single-sided.

2. Double-density double-sided.


Type the number of the option you want to use and press
NEXT.

>
```

20. If flexible disk to which you want to backup files is a single-sided, single-density disk, type: 1 <next>

    If flexible disk is a double-sided, double-density disk, type: 2 <next>

    A Message like the following then appears:

```
Now allocating files to flexible disks.

18 of 124
```

    On short backups this type message may just flash on the screen and then the next display appears.

21. When the system has finished assigning each file for backup, a display like the following appears:

```
This backup will use XX newly formatted disks and will
take about XX to XX minutes.

Press NEXT to continue, press BACK to quit.
```

22. Verify that you have the displayed number of flexible disks of the density chosen in step 20. Also verify that the time required for backup is suitable for you. If the disks or required time is not available, press <back> to quit. To continue the backup operation, press <next>.

This displays a message like the following:

```
                                        HELP is available.


    Please insert backup disk 1 into the Type 1 drive and
    press NEXT.  This disk must be a newly formatted
    [double] density flexible disk.  Press BACK to stop
    this backup operation.
```

23. Use a newly formatted flexible disk as designated in message and insert the disk into Type 1 disk drive.

    Then type: <next>

24. In case an incorrect flexible disk is inserted, a message like the following appears:

```
    The disk in the drive is the wrong fromat.  Press NEXT
    to retry.
```

    Insert correct flexible disk as in step 23 and type: <next>

25. During backup, the system displays the files being backed up in one of the following ways (all are examples):

    If verification option was not chosen:

```
       Now backing up E:BACKUPS.COM.
       Sectors 320-335
       Now backing up E:DISKINIT.COM.
       Sectors 208-223
```

If verification option was chosen:

```
   Now backing up A:MINCE.SWP.
Sectors 113-164
      Successfully verified
   Now backing up A:PASCAL.PCO.
   Sectors 206-242
      Successfully verified
```

If verification option was chosen and an error occurred in verifying:

```
   Now backing up A:PIP.COM.
   Sectors 274-297
      Error verifying file

   Backup processing stopped due to verification
   error.  Press NEXT to return to main menu.
```

26. If more than one disk is required, a message like the following appears when the system has completed backing up the first disk:

```
                                           HELP is available.


Please insert backup disk 2 into the Type 1 drive and
press next.  This disk must be a newly formatted
[double] density flexible disk.  Press BACK to stop
this backup operation.
```

27. Use a newly formatted flexible disk as designated in message. Remove the just-used flexible disk from the Type 1 disk drive and insert the new flexible disk in Type 1 disk drive.

28. Type: <next>

The system continues to display the files being backed up. If a message specifies, replace the used disk with a new disk.

29. When backup is complete, the following message appears:

```
Backup operation completed.

   Please press NEXT to return
```

30. Type: <next>

The main menu appears.

31. To return to CP/M, type: 5 <next>

BACKUP USING TEXT FILE OPTION

This option will select (never deselect) all the files listed in a text file. This text file must be created before entering the backups, and can be created using any text editor or by selecting item 4 on the backup creation menu (figure 10-2) during a prior backup. The text file must have only one filename per line. The filenames must be in this format:

E:FILENAME.TYPE

Where:

E:designates disk on which file resides; any valid CP/M disk designator, but NEVER generic designator (never *:).

FILENAME is any filename up to eight characters that would be valid if used in FILECOPY or PIP.

TYPE is any file type up to three characters that would be valid if used in FILECOPY or PIP.

Use the following procedure to backup disks using the text-file selection option:

1. Type: BACKUPS <next>

   Backup program starts and the display shows the main menu (figure 10-1).

2. To begin backup operation, Type: 1 <next>

   The display then shows the backup creation menu (figure 10-2).

3. Type: 2 <next>

   This displays:

```
                                              HELP is available.

                 SELECT FILES LISTED IN A TEXT FILE

   Enter the name of the text file containing the list
   and press NEXT.
   Press BACK to leave.
   >
```

4. Type the disk designating letter and the filename of the file that contains the list of files you want to select for backup.

   Type: <next>

5. A message indicates which file is being processed as in this example:

```
   Now processing list in file 'FILES.LST'.
      FileA*.* selected
      Test1*.* selected
      Test3*.* selected
```

6. When processing is complete, a message like the following appears:

```
Completed.  There are now 3 total files selected.
Please press NEXT to return.
```

7. When you have finished selection, press <next>. The backup creation menu (figure 10-2) then appears. You may select more files using either the directory option or the text file option.

8. If you have added more files, you may want to make a new text file. To do this, return to backup creation (figure 10-2) and type: 4 <next>

   This displays the write-to-text file message (figure 10-5).

   Type the filename you have chosen for text file and then type <next>

   This displays:

```
Writing the filenames to disk.  Please wait.
```

   Then this line is added:

```
List written to file.  Please press NEXT.
```

   When the above line appears, type: <next>

9. When you are ready to backup files, type 3 <next>

   This displays:

```
Hold on please I am getting file lengths from disk.
```

10. When the system has finished getting file lengths from the disk, the following appears:

```
                                          HELP is available.

    Please enter today's date in MM/DD/YY format.
    >
```

11. Type the date as follows:

    a. Month - 2 digits - use leading 0 for numbers 1 through 9

    b. Slash

    c. Day - 2 digits - use leading 0 for numbers 1 through 9

    d. Slash

    e. Year - 2 digits - use last two digits of year

    If 06/21/83 is entered, display echoes as follows:

```
      > 06/21/83
```

12. Type: <next>

    This displays:

```
                                          HELP is available.

    Please enter the current time in HH:MM format.
    >
```

13. Type the time as follows:

    a. Hour - 2 digits - use leading 0 for numbers 1 through 9. Use 24-hour clock in which 1 pm is 13:00.

    b. Colon

    c. Minutes - 2 digits - use leading 0 for 1 through 9

    If 9 a.m. is entered (09:00), display echoes as follows:

```
>09:00
```

14. Type: <next>

    This displays:

```
                                            HELP is available.

    Please enter a backup ID (8 characters or less).
    >
```

15. Type your own chosen identifier for the backup as a whole. Use 8 characters or less. Display echoes identifier after prompt.

16. Type: <next>

    This displays:

```
    After backing up the files, do you want the files
    verified (y or n)?
```

17. Type: y or n <next>

This displays:

```
|                                                              |
|                                           HELP is available.|
|                                                              |
|      What kind of diskette are you going to use?             |
|                                                              |
|      1. Single-density single-sided.                         |
|                                                              |
|      2. Double-density double-sided.                         |
|                                                              |
|                                                              |
|Type the number of the option you want to use and press NEXT.|
|                                                              |
|      >                                                       |
|______________________________________________________________|
```

18. If flexible disk to which you want to backup files is a single-sided, single-density disk, type: 1 <next>.

    If flexible disk is a double-sided, double-density disk, type: 2 <next>

    A message like the following then appears:

```
|                                                              |
|     Now allocating files to flexible disks.                  |
|                                                              |
|     18 of 124                                                |
|______________________________________________________________|
```

19. When the system has finished assigning each file for backup, a display like the following appears:

```
|                                                              |
|     This backup will use XX newly formatted disks and        |
|     will take about XX to XX minutes.                        |
|                                                              |
|     Press NEXT to continue, press BACK to exit.              |
|______________________________________________________________|
```

20. Verify that you have the displayed number of flexible disks of the type chosen in step 18. Also verify that time required for backup is suitable for you. If the flexible disks or required time is not available, press <back> to quit. To continue the backup operation, press <next>

    This displays a message like the following:

```
                                        HELP is available.

  Please insert backup disk 1 into the Type 1 drive and
  press NEXT.  This disk must be a newly formatted
  [double] density flexible disk.  Press BACK to stop
  this backup operation.
```

21. Use a newly formatted flexible disk the of density designated in message and insert disk in the Type 1 disk drive.

    Then type: <next>

22. In case an incorrect flexible disk is inserted, a message like the following appears:

```
  The disk in the drive is the wrong format.  Press NEXT
  to retry.
```

    Insert correct flexible disk as in step 21 and type: <next>

23. During backup, the system displays the files being copied in one of the following ways (all are examples):

    If verification option was not chosen:

```
  Now backing up E:BACKUPS.COM.
  Sectors 320-335
  Now backing up E:DISKINIT.COM.
  Sectors 208-223
```

If verification option was chosen:

```
Now backing up A:MINCE.SWP.
Sectors 113-164
   Successfully verified
Now backing up A:PASCAL.PCO.
Sectors 206-242
   Successfully verified
```

If verification option was chosen and an error occurred in verifying:

```
Now backing up A:PIP.COM.
Sectors 274-297
   Error verifying file

Backup processing stopped due to verification error.
Press NEXT to return to main menu.
```

24. If more than one flexible disk is required, a message like the following appears when the system has completed backing up the first flexible disk:

```
                                         HELP is available.

Please insert backup disk 2 into the Type 1 drive and
press next.  This disk must be a newly formatted
[double] density flexible disk.  Press BACK to stop
this backup operation.
```

25. Use a newly formatted flexible disk as designated in message. Remove the just-used flexible disk from the Type 1 disk drive and insert the new flexible disk in Type 1 disk drive.

26. Type: <next>

    The system continues to display the files being backed up. If a message specifies, replace the used disk with a new disk.

27. When backup is complete, the following message appears:

```
Backup operation completed.

Please press NEXT to return
```

28. Type: <next>

    The main menu (figure 10-1) appears.

29. Type: 5 <next> to return to CP/M.

CONTINUE A BACKUP THAT WAS INTERRUPTED

In case you have to stop the backup operation during backup processing, you may press CTRL-C to exit the utility. When you want to continue a backup that was interrupted, use the following procedure. In case the last copy was actually completed despite the interruption, you will be informed and returned to the main menu.

NOTE

If system configuration was changed since the interruption occurred, this utility may not work correctly.

"HELP is available" indicates that you may press the HELP key for assistance.

Use the following procedure to continue an interrupted backup operation:

1. Type: BACKUPS <next>

   Backup program starts and the main menu (figure 10-1) appears.

2. Type: 2 <next>

   This displays:

```
                                              HELP is available.

     CONTINUE A BACKUP OPERATION THAT WAS INTERRUPTED

     Please insert the last disk that was being backed
     up to into the Type 1 drive.

     Press NEXT to continue.
```

3. Insert the flexible disk into the Type 1 disk drive that was in it when the interruption occurred.

4. Type: <next>

   The system completes the backup as originally requested. If the following message appears, you have inserted the wrong flexible disk. Find the flexible disk that was in the Type 1 disk drive when the interruption occurred. Then repeat steps 3 and 4.

```
     This is not a backup disk.


     Press NEXT to try another disk.
     Press BACK to return without doing the backup.
```

5. If backup requires only one flexible disk, skip to step 8. Otherwise, when system has completed backing up the first disk, a message like the following appears:

```
                                               HELP is available.

     Please insert backup disk 2 into the Type 1 drive and
     press next.  This disk must be a newly formatted
     [double] density flexible disk.  Press BACK to stop
     this backup operation.
```

6. Remove the just-used flexible disk from the Type 1 disk drive and insert a new flexible disk in the Type 1 disk drive.

7. Type: <next>

   The system continues to display the files being copied. If a message specifies, replace the used disk with a new disk.

8. When backup is complete, the following message appears:

```
     Backup operation completed.

     Please press NEXT to return
```

9. Type: <next>

   The main menu (figure 10-1) appears.

10. To return to CP/M, type: 5 <next>

SEE BACKUP DISK INFORMATION

This section of the backup utility displays information concerning the backup operation itself and about the files that were backed up. Use the following procedure to see backup disk information:

1. Type: BACKUPS <next>

   Backup program starts and the main menu (figure 10-1) appears.

2. Type: 3 <next>

   This displays the backup disk information menu (figure 10-6) as follows:

```
                                                   HELP is available.

                      BACKUP DISK INFORMATION

                       What would you like to do?


   1. See general backup information.
      (Date, time, backup ID, etc...)

   2. See the list of files backed up to flexible disk.

   3. Return to main menu.

Type the number of the option you want to use and press NEXT.
>
```

Figure 10-6. Backup Disk Information Menu

3. To see list of files backed up on flexible disk, press 2, then <next> and skip to step 7. To see general information about the backup:

   Type: 1 <next>

   This displays:

```
                                                   HELP is available.



      Please place one of the backup flexible disks in the
      Type 1 drive.

      Press NEXT when ready to continue.
      Press BACK to leave.
```

4. Insert the backup flexible disk containing the information you want to see in the Type 1 disk drive.

5. Type: <next>

   This displays backup disk information like shown in the following example:

```
                                              HELP is available.


                 BACKUP DISK INFORMATION

   The backup ID for this disk is BACKUP.1.

   This is disk 1 in a set of 1 disks.

   The files contained in this backup copy are from
   01/25/83 at 15:30.

   There are 3 total files included in this backup.

   This backup disk contains XX files.


Please press NEXT to return.
```

6. Press NEXT to return to backup disk information menu and skip to step 11.

7. If you pressed 2 to see list of files, this message appears:

```
                                              HELP is available.


      Please place one of the backup flexible disks in the
      Type 1 drive.

      Press NEXT when ready to continue.
      Press BACK to leave.
```

8. Press NEXT. A display similar to the following example appears:

```
LIST OF FILES ON BACKUP DISK                    HELP is available.
3 total files.

   # NAME       TYPE          SIZE  SOURCE DISK  START-END DISK
    1 DISKINIT   COM          327         E            1- 1
    2 STAT       COM          224         E            1- 1
    3 BACKUPS    COM           41         E            1- 1


   Press '+' or '-' to page forward or backward.
   Press BACK to return.
>
```

9. To see the next page of the disk directory, type: + <next>

   To page backward, type - <next>

10. To return to the backup disk information menu, press <back>.

11. To return to the main menu, type: 3 <next>

    This displays:

```
Insert the system disk into the Type 1 drive and press NEXT.
```

12. To return to CP/M, type: 5 <next>

RESTORE FILES TO RIGID DISK

This section of the backup utility allows you to restore to the rigid disk any files that have been backed-up on a flexible disk. If the program finds that a file with the same name is already on the specified rigid disk, you will be asked for authority to delete the file and write in the new one.

"HELP is available" indicates that you may press the HELP key for assistance.

Use the following procedure to recover information from backup files:

1. Type: BACKUPS <next>

   Backup program starts and the main menu (figure 10-1) appears.

2. Type: 4 <next>

   This displays:

```
|------------------------------------------------------------|
|                                          HELP is available.|
|                                                            |
|                                                            |
|   Please place one of the backup flexible disks in the     |
|   Type 1 drive.                                            |
|                                                            |
|   Press NEXT when ready to continue.                       |
|   Press BACK to leave.                                     |
|------------------------------------------------------------|
```

3. Type <next>

   This displays a message like the following example:

```
|-------------------------------------------------------------|
|LIST OF FILES ON BACKUP DISK              HELP is available.|
|3 total files                             0 files selected.|
|                                                             |
|-------------------------------------------------------------|
|  # NAME      TYPE        SIZE  SOURCE DISK  START-END DISK |
|   1 DISKINIT  COM         327       E            1- 1       |
|   2 STAT      COM         224       E            1- 1       |
|   3 BACKUPS   COM          41       E            1- 1       |
|                                                             |
|                                                             |
|                                                             |
|  Type in the numbers of the file you would like to recover.|
|  HELP is available for examples.                            |
|  Press LAB alone when ready to continue.                    |
|  Press BACK to exit without file recovery.                  |
|>                                                            |
|-------------------------------------------------------------|
```

4. If you want to return to the backup information menu without backing up files, press <back>.

5. When you are ready to select a file, enter selection options in a string separated by commas or by spaces. You may select files by their individual number, or you may select all the files on the disk as follows:

   - NUMBER: To select a single file, type its number.

   - NUMBER-NUMBER: To select a set of files numbered in sequence, type first number, then -,then last number (example; 1-10 for files 1 through 10).

NOTE

Files are deselected by entering the number of a previously selected file.

6. Press <next>. Display is updated after each selection. Selected files have a * before the filename.

7. When you have finished selecting the files, type: <lab>

   If you press <lab> without selecting a file, this message appears:

```
You must select at least one file to be recovered.

Press NEXT to continue.
```

   Otherwise this message appears:

```
                                   HELP is available.


To what disk would you like to recover these files?

Press NEXT to recover to the original source disks.

Valid disks to select to are from E to P.

>
```

8. Enter the letter designating the rigid-disk drive to which you want the files restored. To restore files to their original source disks, type: <next>

9. If you have selected file destination and the following type message appears, place the required disk in the Type 1 disk drive and type: <next>

```
                                         HELP is available.

        Disk not loaded.
        Please place disk N in Type 1 disk drive.
        Press NEXT when ready.
```

10. In case a file is already on the chosen rigid disk, a display appears like in following example:

```
                                                HELP is available.

   The disk you have chosen to recover this file to already
   has a file E:DISKINIT.COM.

   1. Choose a different file name.

   2. Choose a different disk for recovery.

   3. Leave the recover utility.

   4. Skip just this file and continue.

   5. Remove the duplicate file and replace with this backup.

      Type the number of the option you want to use and press
      NEXT.

      >
```

a. If you choose to rename the file, so that both files are on the rigid disk, type: 1 <next>

This displays:

```
Enter your new file name and press NEXT.

>
```

Type new filename and then press <next>. Proceed to step 11

b. If you choose a different disk for recovery, type: 2 <next>

This displays:

```
To what disk would you like to recover these files?
Valid disks to select are from E to P. >
```

Type the new disk designator and then press <next>. Proceed to step 11.

c. To stop the recovery procedure without copying this or any subsequent files on the rigid disk:

Type: 3 <next>

Proceed to step 11.

d. To skip just this file and copy all subsequent files on the rigid disk:

Type: 4 <next>

Proceed to step 11.

e. If you choose to replace an old file with a new file of same name:

Type: 5 <next>

Proceed to step 11.

11. The following type message appears at the end of the process:

```
Recovery operation complete.

X files have been recovered.

Press NEXT to continue.
```

12. Type: <next>

This displays a message like the following:

```
Insert the system disk into the Type 1 drive and press NEXT.
```

13. Remove disk from Type 1 disk drive and insert the required disk.

14. To return to main menu, type: <next>

15. To return to CP/M, type: 5 <next>

The PHONE utility is used to create and maintain a directory of up to 99 phone number entries. The utility is also used to communicate, via your terminal, with the computer or terminal locations that correspond to the phone number entries. This section points out considerations concerning the utility and provides instructions on its use.

To use the PHONE utility, you must have the following equipment:

- Type 3 terminal
- Internal 1200/1200-baud modem
- Type 1 disk subsystem

CONSIDERATIONS

Considerations concerning the PHONE utility include:

- The PHONE utility only exists on versions 5.00 and higher of the CP/M® 2.2 Operating System. If the version of your CP/M disk is below 5.00, you do not have access to the PHONE utility. Contact your Control Data representative to obtain the current disk.

- This item only applies to terminals that have firmware older than (below) revision 4.0. Before using the PHONE utility, refer to the 110 Operating Guide and set the F5, position 2 CP/M mode parameter to a 1. This enables cursor biasing. This cursor biasing may stay in effect except when the 110 system diagnostics are run. Whenever the diagnostics are run, the F5, position 2 parameter must be temporarily changed back to a 0. A YR109-A Enhanced Firmware Option, which may be ordered through your Control Data representative, makes this parameter change unnecessary.

- The phone numbers you want to use with the PHONE utility go in a directory on the selected disk. The directory (file PHONE.DIR) must reside on the same disk as the PHONE utility (file PHONE.COM). Do not attempt to execute the PHONE utility from any disk except the one which is currently selected.

- The number of rings the PHONE utility uses in a phone call is selectable. At least two rings should be selected because the ringing at a called number does not match the ringback at the other end and sometimes fewer rings may actually occur than are selected.

ACCESSING PHONE UTILITY

You may begin using the PHONE utility by following this procedure:

1. Turn on the power for your terminal and disk subsystem.
2. Load your CP/M disk into the disk subsystem.
3. Press the F3 key in the top row of your keyboard. The terminal message appears, indicating the version of your disk.

   This prompt also appears:

```
A>
```

4. Type: PHONE <next>

   The Phone Directory Display Screen (figure 11-1) appears.

You are now ready to start creating a phone number directory. Before you start, read the following paragraphs for a brief description of the individual areas of the display screen. Then go on to read the command procedures, beginning with the ADD command. In case you should make any errors, this section ends with a listing and explanation of the error messages displayed by the system, and the actions you must take to correct the errors.

PHONE DIRECTORY DISPLAY SCREEN

The Phone Directory Display Screen is divided into four horizontal parts. They are designated by the letters A through D in figure 11-1. Each part, the Message Area (A), the Menu/Command Area (B), the Entry Being Built Line (C), and the Phone Directory Area (D), plays a different role in the use of the utility. Each part is described separately. Underlined terms or commands included in the following descriptions are highlighted on the display screen.

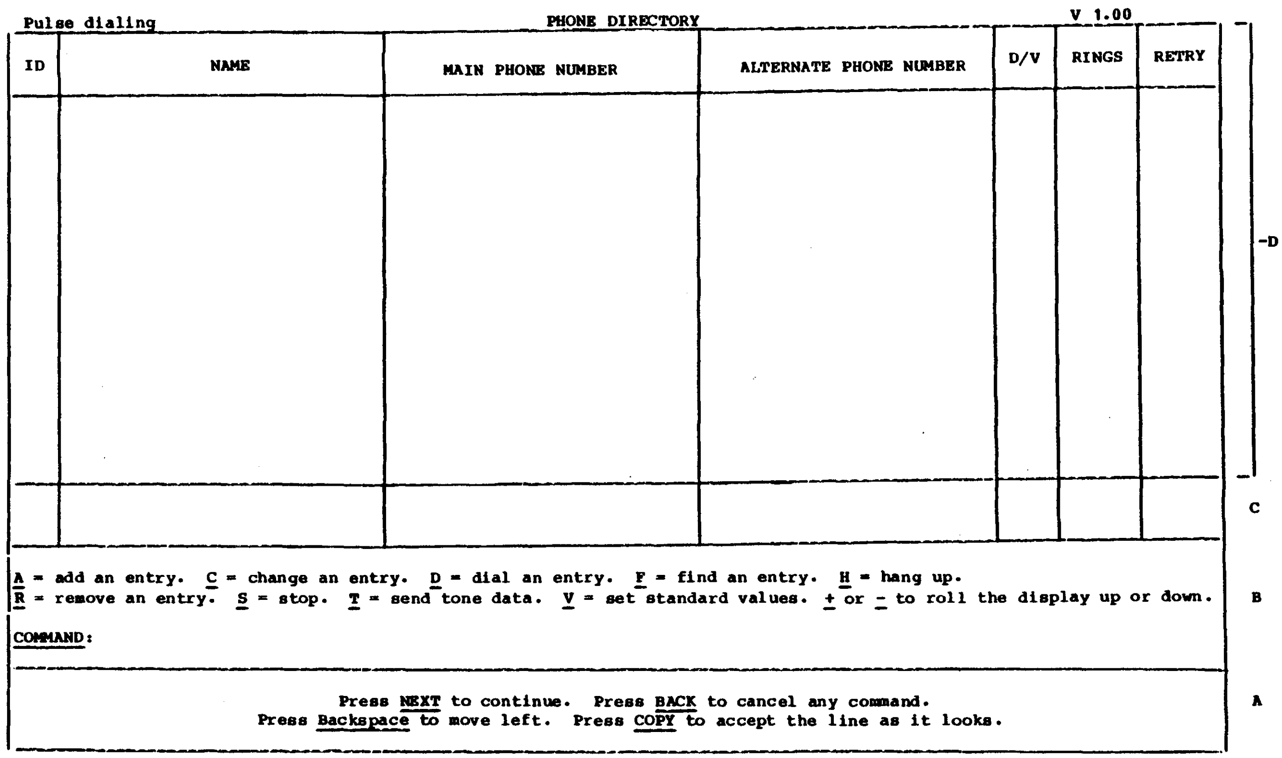


Figure 11-1. Example Phone Directory Display Screen

MESSAGE AREA

The message area consists of two lines. These lines may contain standard system messages or error messages.

Standard System Messages

These messages appear when you begin using the utility. The four messages are:

> Press NEXT to continue. Press BACK to cancel a command. Press Backspace to move left. Press COPY to accept the line as it looks.

The messages remain on the screen during the use of any of the commands in the utility, unless you make an error. Then an error message replaces the standard messages.

Error Messages

Error messages differ depending on the type of error you make. For example, if you try to enter a command not included in the utility, such as X, the message:

There is no such command, please try again.

appears in the message area. When you correct the error, the standard system messages reappear.

MENU/COMMAND AREA

The menu/command area consists of three lines. The lines may contain four different types of information. The types of information are the listing, or menu, of commands included in the utility; the individual commands you select; the instructions given by the system during any of the command procedures; and messages displayed by the system when you attempt to dial any of the numbers in the directory.

Menu

The menu of all the available commands appears on the first two lines of the menu/command area. The menu appears when you first begin using the utility. The ten commands are arranged horizontally as follows:

A = add an entry. C = change an entry. D = dial an entry.
F = find an entry. H = hang up. R = remove an entry.
S = stop. T = send tone data. V = set standard values.
+ or - to roll the display up or down.

Command

Before you select the command, the third line of this area contains the word COMMAND: followed by a blinking cursor. You type the chosen command on this line. For example, if you choose to add an entry, the line would appear like this:

COMMAND: A

Once you press NEXT, the command line disappears and the system begins giving you the instructions you need to complete the procedure.

System Instructions

After you enter a command, the first two lines of the menu/command area display the instructions you must follow and the questions you must answer to complete the command procedure.

For example, after you enter the A command to add a new entry, the following instructions appear:

Adding a new entry.
Type the name.

The blinking cursor is positioned in the spot where you type whatever information is requested. After you type what is requested and press the NEXT key, the system displays another instruction or question. This continues until you complete the command procedure. When you have finished, the menu of commands and the COMMAND: statement reappear.

System Messages

When you use the Dial command to dial a number in your directory, the system displays messages from the internal modem on the first line of the menu/command area, telling you the status of your call. For example, the following three internal modem messages may appear one at a time in the following order (these messages are offset from the rest of the display):

DIALING PHONE NUMBER
RINGING
NO ANSWER

At the end of the dialing process, the menu of commands and the COMMAND: statement reappear.

ENTRY BEING BUILT LINE

The Entry Being Built Line has several different functions, depending upon the command you are using.

- With the Add command, the blinking cursor appears in the first space of the name section of the line. The new entry appears on the line as you are typing it. When you complete the command procedure, the new entry appears highlighted in the Phone Directory area of the screen, and the Entry Being Built Line is again blank.
- With the Change, Dial, Find, and Remove commands, the entry you choose after you begin the procedure appears on the Entry Being Built Line.
- With the Values command, the standard (default) values set for the number of rings per call, number of retries per call, and data or voice communication appear on the Entry Being Built Line.

The remaining commands do not directly affect the Entry Being Built Line.

PHONE DIRECTORY AREA

The first time you use the PHONE utility, the Phone Directory Area is blank. As you use the Add command to create your directory, the system displays from one to 15 entries in alphabetical order within the directory area. The entry in the center of the directory is always highlighted.

Once you have put a number of entries into your directory, you may want to look through the directory before selecting a specific entry. The + and - commands allow you to do this. These commands move the directory up or down by the number of lines you specify. More detailed instructions for using the + and - commands appear later in this section.

COMMANDS

The PHONE utility contains the commands that are listed in table 11-1. Instructions on each command are on the following pages in the same order as the commands listing on the Phone Directory Display Screen. If you have not made any entries in the phone directory yet, it is recommended that you read the remainder of this and the command instructions before beginning.

TABLE 11-1. PHONE UTILITY COMMANDS

COMMAND	FUNCTION
Add	To add an entry to the directory.
Change	To change an entry in the directory.
Dial	To dial an entry automatically.
Find	To find and display an existing entry.
Hangup	To hang up the phone when a data call is finished.
Remove	To delete an entry from the directory.
Stop	To stop the utility.
Tone	To send tone-coded data.
+ or -	To roll the display up (+) or down (-).
Values	To change up to four of the default values which are relevant when adding an entry to the directory. The original default values are: ● The number of rings to be used in calls is five. ● The number of call retries is to be zero. ● Calls are for exchanging data, not a voice call. ● Pulse-dialing is to be used in calls, not tone-dialing.

You may interrupt use of the commands at anytime by typing <CTRL C> simultaneously. When you do this, the phone directory is not updated with any of the information you typed before typing <CTRL C>. The system displays the A>, just as it does when you first load the CP/M disk.

If you should make an error while using the commands, the system makes a "beep" sound and does not let you type any additional information until you correct the error. The error messages you may encounter and instructions for correcting the errors appear at the end of this section.

ADD COMMAND

Use the Add command to add entries to the phone number directory.

1. Type: A <next>

   Message appears:

   ```
   Adding an entry
   ```

   Prompt appears:

   ```
   Type the name
   ```

2. Type the identification name for the new number. Use up to 40 characters including blanks (spaces).

3. Type: <next>

   Prompt appears:

   ```
   Type the phone number
   ```

4. Type the first phone number for this entry. Use up to 30 characters selected from the following:

   0-9; any alphabetical character; special characters ( ) - and space

   * and # for special application in tone dialing;

   ! to wait for a dial tone; ? to wait for an absence of a dial tone

5. Type: <next>

   Prompt appears:

Type the alternate phone number

6. Type the alternate phone number for this entry. Use the same format as the first phone number.

7. Type: <next>

   Prompt appears:

This entry will be used for data calls. Type V if it should be used for voice calls.

or

This entry will be used for voice calls. Type D if it should be used for data calls.

8. Type V or D

   Type: <next>

   Prompt appears:

If there is no answer after X rings this entry will be retried unless you change this value.

   X is the default value.

9. Type a number from 1 to 9 (2 or more is recommended).

   Type: <next>

   Prompt appears:

This entry will be <u>retried</u> X times unless you change this value.

   X is the default value.

10. Type a number from 0 to 9, indicating the number of times you want the system to retry the call.

    Type: <next>

    Prompt appears:

An ID number has been assigned. If you wish to replace it with another, do so now.

11. Type an ID number only if you do not want the number assigned by the system:

    Type: <next> whether or not you type an ID number

    The system displays the directory portion of the screen with the new entry highlighted in the center. Alphabetically adjacent entries are placed above and below the new entry.

    Prompt appears:

Do you want to <u>ADD</u> another entry? (Y).

12. Type: <next> if you want to add another entry

    or

    Type: N <next> if you are finished.

CHANGE COMMAND

Use the Change command to change one or more aspect of an existing phone number entry. After you type the name or ID of the entry you want to change, the system displays the entry and updates it as you make the changes. Each subsequent message displays the current values for the entry. You may type directly over the current values to make your changes. Follow the procedure below to change an entry:

1. Type: C <next>

   Message appears:

   ```
   Changing an entry.
   ```

   Prompt appears:

   ```
   Type the ID or name of the entry to be changed.
   ```

2. Type the ID or name as requested.

   Type: <next>

   Prompt appears:

   ```
   Type the new name for the entry.
   ```

3. Type a new name if needed, or leave the entry as it is.

   Type: <next>

   Prompt appears:

   ```
   Type the new phone number.
   ```

   If you changed the name or ID, the system displays the directory portion of the screen with the entry you are changing highlighted in the center.

4. Type a new phone number if necessary.

   Type: <next>

   Prompt appears:

Type the new alternate phone number.

5. Type a new alternate phone number if you need to make a change.

   Type: <next>

   Prompt appears:

This entry will be used for data calls. Type V if it should be used for voice calls.

   or

This entry will be used for voice calls. Type D if it should be used for data calls.

6. Make any necessary changes.

   Type: <next>

   Prompt appears:

If there is no answer after X rings this entry will be retried unless you change this value.

7. Change the number of rings if necessary.

   Type: <next>

   Prompt appears:

This entry will be retried X times unless you change this value.

8. Change the number of retry attempts if necessary.

   Type: <next>

   Prompt appears:

Type the new ID number, if any.

9. Type an ID number if you do not want the number assigned by the system.

   Type: <next>

   The system displays the directory portion of the screen with the updated entry highlighted in the center. Alphabetically adjacent entries appear above and below the updated entry.

   Prompt appears:

Do you want to CHANGE any more entries? (Y).

10. Type: <next> if you want to make more changes

    or

    Type: N <next> if you are finished.

DIAL COMMAND

The Dial command performs a dial-up of an entry according to the information you enter. Follow the procedure below to use the dial command.

1. Type: D <next>

   Message appears:

Dialing an entry.

Prompt appears:

Type the ID or name of the entry to be dialed.

2. Type the ID or name as requested.

   Type: <next>

   Prompt appears:

Do you want to change any of the current Values for this entry?

   If you do not need to make changes, type N and go to step 3.

   If you need to make changes:

      Type: Y <next>

      Answer the questions as the system presents them, pressing <next> after each answer.

3. Type: <next>

   For data calls, the system attempts to dial the entry, as well as any alternate entry.

   For voice calls, the system attempts to dial the entry and displays the following message:

Press BACK to stop trying, or hang up the phone and press NEXT to try the alternate number. Trying main number.

   Follow the display prompts for the voice calls.

   System messages are displayed on line 24 in the Menu/Command area of the display screen.

   If the system fails to make a connection for either voice or data calls, it cancels the command.

   If the system makes a connection for a data call, the utility stops and control of the system returns to CP/M.

FIND COMMAND

Use the Find command to locate a specific entry in the phone number directory.

1. Type: F <next>

   Message appears:

Finding an entry.

   Prompt appears:

Type the ID or name of the entry to be found.

2. Type the ID or name of the entry you need.

   Type: <next>

   The system displays the selected number in the center of the directory display area of the screen. The number is highlighted, and is preceded and followed by alphabetically adjacent entries.

HANGUP COMMAND

The Hangup command breaks an established phone connection of a data call. When you type "H <next>" the system displays the message "Hanging up" for five seconds. During this time, the phone connection is broken.

REMOVE COMMAND

Use the Remove command to delete an entry from the phone number directory file.

1. Type: R <next>

   Message appears:

Removing an entry

   Prompt appears:

Type the ID or name of the entry to be removed.

2. Type the ID or name as requested.

   Type: <next>

   The system displays the entry you have chosen, highlighted in the center of the directory. It also displays the question:

Are you SURE you want to remove this entry? (N)

3. Type: (Y) <next>

   The number is removed and the ID recycled for reassignment.

   or

   Type: (N) <next>

   Prompt appears:

Do you want to REMOVE any more entries? (Y).

4. Type: <next> if you have more entries to remove

   or

   Type: N <next> if you are finished.

STOP COMMAND

The Stop command immediately ends the use of the phone utility. The phone directory file is updated, and any active phone connection remains connected.

To use the Stop command:

Type: S <next>

Message appears if file has been newly created or changed:

```
Stopping.  Writing new phone directory to disk
```

The system clears the screen and displays A>.

TONE COMMAND

Use the Tone command to transmit a tone message on an already established phone connection.

1. Type: T <next>

   Prompt appears:

```
Type tone-coded data (Up to 30 characters which appears
on a touch-tone phone):
```

2. Type the data.

   Type: <next>

   The system checks each character for validity and then sends the data over the phone line.

   Message appears:

```
Now sending tone data
```

VALUES COMMAND

The Values command allows you to change three default values: the voice/data default (originally data), the rings count default (originally 5), and the retry count default (originally 0). It also enables you to change the dialing mode parameter (originally pulse-dialing), which has no direct relationship to any individual directory entry.

Follow the procedure below to change the default values.

1. Type: V <next>

   Message appears:

```
Changing standard Values
```

   Prompt appears:

```
Calling will use tone-dialing.  Should this be changed
to use pulse-dialing?  (N)
```

   or

```
Calling will use pulse-dialing.  Should this be changed
to use tone-dialing?  (N)
```

2. Change (N) to (Y) if necessary.

   Type: <next>

   Prompt appears:

```
Calling is set for data calls.  Type V if it should be
used for voice calls.
```

   or

```
Calling is set for voice calls.  Type D if it should be
used for data calls.
```

3. Change V or D if necessary.

   Type: <next>

   Prompt appears:

> If there is no answer after X rings all entries will be retried unless you change this value.

   X is the default value.

4. Change the number of rings if necessary.

   Type: <next>

   Prompt appears:

> All entries will be retried X times unless you change this value.

   X is the default value.

5. Change the number or retry attempts if necessary.

   Type: <next>

   All the values have now been altered.

DISPLAY SCROLLING

You may use the plus (+) and minus (-) signs to scroll the directory portion of the display screen. The plus sign causes the directory to scroll upwards, while the minus sign causes the directory to scroll downwards. You may specify the number of entries you want to scroll, from one to 99. If you do not type a specific number on the screen, the directory scrolls seven entries.

Follow the instructions below to scroll the directory display.

1. Type: (+) or (-) <next>

   Prompt appears:

Enter line count: 7

2. Change the line count number if necessary.

   Type: <next>

   The directory display scrolls the chosen number of entries. The middle entry on the directory display remains highlighted.

ERROR MESSAGES

Error messages generated by the phone utility describe the error clearly in as much detail as possible. The messages appear in a separate area of the screen and leave the entry containing the error intact. In most cases the error message is generated at exactly the time the error occurs, for example, when you enter an illegal character. You cannot type any additional information until you correct the error.

Certain errors may occur that you will not be able to correct without the aid of your CDC representative. All the error messages for these errors begin with the words "System error:"

The following is a list of error messages and an explanation of each message. Correctable errors appear first, followed by system errors.

CORRECTABLE ERRORS

A name cannot be one- or two-digits because it would look like an ID number.

Encountered on entry of a one-digit name on an ADD or CHANGE command. No repositioning of the command processing takes place; re-entry of the name is permitted.

Call incomplete for above reason. Press any key to continue.

Encountered whenever a DIAL command could not be completed by the modem firmware. The firmware's message ("BUSY," "DIALING ERROR," or another) will still be displayed on the line preceding this error message.

Cursor biasing is not set. Refer to your Terminal Owner's Guide before retrying PHONE. Press any key to terminate.

Encountered prior to the screen being formatted, when cursor biasing is not set and the terminal firmware being used is older than Revision 4.0. With 4.0 firmware, cursor biasing can be set dynamically by the software; prior revisions of firmware do not permit it. This is essentially a fatal error, since no direct recovery is permitted. The only actions that can be taken to correct the problem is to manually set the cursor biasing bit in the Mode Installation Parameters (see the manual referred to in the message) or to obtain Revision 4.0 firmware.

NOTE

The CDC 110 keyboard diagnostic test does not expect cursor biasing to be set and will not perform correctly if it is. Cursor biasing must therefore be cleared prior to running the diagnostics.

PHONE cannot run on this terminal.

Encountered when you attempt to use the PHONE utility on anything but a Type 3 terminal. This error totally prevents PHONE from executing.

Phone directory not found on disk. Assuming empty directory.

Encountered during sign-on only if the PHONE.DIR file is not present on the disk. This error should not occur at any other time.

Please answer Y or N.

Encountered upon answering any yes-or-no question with other than a Y, N, or NEXT. Corrective action consists of backspacing and entering the legal character.

That ID number is already assigned.

Encountered on input of an ID number on an ADD or CHANGE command. The original or auto-assigned ID is displayed in the entry-being-built area of the screen and the cursor is positioned to its first digit.

There is no such command, please try again.

Self-explanatory.

There are too many characters in this field.

Self-explanatory.

There is no modem present.

Encountered on a DIAL, HANGUP, or TONE command when run on a terminal without a 1200/1200-baud modem.

There is no such ID number assigned. Was it typed correctly? (N)

Encountered on a CHANGE, FIND, DIAL, or REMOVE command when a one- or two-digit value has been entered in response to the name/ID question. Selection of YES as the response cancels the command. Selection of NO results in the repositioning of the command processing to the point of entry of the name/ID.

There is no more room in the directory (the limit is 99 entries).

Encountered on an attempt to input the 100th entry with an ADD command. Corrective action is obvious; the command will have been already cancelled.

There is no such name on the phone directory. Is it spelled right? (N)

Encountered on input of an unknown name on a FIND, REMOVE, or DIAL command. The section of the directory closest in alphabetical order to the name is brought to the screen for visual inspection, which may well reveal such a misspelling. Selection of YES as the response cancels the command. Selection of NO results in repositioning of the command processing to the point of entry of the name.

There is no such name on the phone directory. Should this be an ADD? (N)

Encountered on input of an unknown name on a CHANGE command. The section of the directory closest in alphabetical order to the name is brought to the screen for visual inspection. Selection of YES as the response results in conversion of the command from CHANGE to ADD. Selection of NO results in elimination of the entered name and repositioning of processing to just before entry of the name.

This name already exists. Should this be a CHANGE? (N)

Encountered on an ADD command upon entry of a name which already exists. The portion of the directory containing the name is displayed with the name highlighted at the vertical center of the screen for visual inspection. Selection of YES as the response results in conversion of the command to CHANGE. Selection of NO as the response results in elimination of the entered name and repositioning of processing to just before entry of the name.

This name already exists as shown. Please retype.

Encountered on entry of a new name during a CHANGE command. Corrective action is self-explanatory.

Warning: 1200/1200 baud modem not installed. Press any key to continue.

Encountered prior to formatting the screen, when the Type 3 terminal firmware does not detect the presence of the internal 1200/1200-baud modem. If an internal modem is installed, correct by setting the parameters for the modem as described in the terminal Installation Guide.

You may only use digits between 0-9.

Encountered on input of an explicit ID on an ADD or CHANGE command. Corrective action consists of entry of valid characters or cancellation of the entire command.

You may only use a number between 0 and 9.

Encountered on input of a retry count on an ADD, CHANGE, DIAL, or VALUES command. Corrective action consists of entry of a permissible number or cancellation of the entire command.

You may only use a number between 1 and 9.

Encountered on input of a ring count on an ADD, CHANGE, DIAL, or VALUES command. Corrective action consists of entry of a permissible number or cancellation of the entire command.

You may only use a-z, 0-9, +, and -.

Encountered on input of an illegal character during command entry. Corrective action consists of entry of a permissible character or cancellation of the command.

You may only use a-z, 0-9, -, *, #, (,), ?, ! and space. (? would mean 'wait for no tone', and ! would mean 'wait for a tone')

Encountered on input of a phone number on an ADD, CHANGE, or DIAL command. Corrective action consists of entry of a permissible character.

You may only use printable characters.

Encountered on input of a name on an ADD, CHANGE, DIAL, FIND, or REMOVE command. Corrective action consists of entry of valid characters or cancellation of the command.

You may only use D or V.

Encountered on input of an illegal character in response to a data/voice question on an ADD, CHANGE, DIAL, or PARAMETERS command. Corrective action consists of re-entry of one of the permissible characters or cancellation of the entire command.

You may only use A-Z, 0-9, *, and #.

Encountered on input of a tone-coded number on a TONE command. Corrective action consists of entry of a permissible character or cancellation of the command.

SYSTEM ERRORS

System Error: Can't create new phone directory.

Encountered during execution of a STOP command. The old directory is left intact, but the error is fatal. The exact reason for the error has not been determined.

System Error: Conflict in ID assignment status. Press any key to terminate.

Encountered on an ADD, CHANGE, DIAL, FIND, or REMOVE command, when the system has reserved an ID but no entry can be found for that ID.

System Error: Disk out of data space.

Encountered during execution of a STOP command when a new phone directory is being written to the disk. The old phone directory remains intact but the new phone directory must be erased. The largest a phone directory can be is eleven thousand characters.

System Error: No more room for directory in memory. Press any key to terminate.

Encountered during sign-on or during an ADD, CHANGE, or DIAL command. The directory is designed to handle 99 entries. While you can remove entries to create space if the error occurs during an ADD, CHANGE, or DIAL command, you must report the error to your CDC representative if the error occurs during sign-on.

System Error: File I/O System Error #XX.

The majority of these errors are unlikely to occur. If they do, you must contact your CDC representative. The error numbers (XX) are detailed in the following list.

1	Reading unwritten data
2*	Disk out of data space
3	Can't close current extent
4	Seek to unwritten extent
5	Can't create new extent
6	Seek past end of disk
7	Bad file descriptor given
8	File not open for read
9	File not open for write
10	No file descriptor slots left
11**	File not found
12	Bad mode given to open
13*	Can't create file
14	Seek past 65535th record

*Described under System Errors.
**Described under Correctable Errors.

The utility programs make it possible to transfer data from one flexible disk to another, format flexible and rigid disks, perform date and time-clock operations, etc. You have already used some of these utilities in formatting flexible disks, and copying and transferring software. This section contains descriptions and general instructions for these utilities:

- FILECOPY
- FORMAT - to format flexible disks
- DISKINIT - to format rigid disks
- SYSGEN
- DSKUTIL
- BACKUP
- DATTIM
- PHONE

FILECOPY

The FILECOPY program lets you copy files from one flexible disk to another. The program operates with one flexible-disk drive. The following transfers are possible:

Double-density to double-density

Single-density to single-density

Single-density to double-density

Double-density to single-density

In FILECOPY, your response to the prompt "Filenames:" is the command line "destination = source:filenames." The format for the filenames in FILECOPY is the same as the format of filenames used in PIP destination = source command lines. Filename templates using asterisks and question marks are legal. Therefore, multiple files can be copied using a single filename template. The only restriction in the FILECOPY command line is that you must include the drive code. The drive code distinguishes single-density disks (drive code C) from double-density disks (drive code A).

Perform these steps to copy files using FILECOPY:

1. Type: FILECOPY <next>

2. Program prompts for filename input with following question:

```
Filenames:
```

   The following are valid command lines:

   A: = A:*.*
   c:test.abc = a:test.abc
   C:X*.*=A:test*.*

3. Program asks that source flexible disk be inserted into drive by prompting:

```
Insert source disk, type CR to continue
```

4. Type: <next>

   Program responds with name of file being copied and prompt:

```
Insert destination disk, type CR to continue
```

5. Repeat steps 3 and 4 in response to prompts.

6. After all files matching input template have been copied, program again displays filenames prompt.

7. To exit FILECOPY,

   Type: <next>

8. Display shows:

```
Insert system disk, type CR to exit
```

Type: <next>

The program automatically protects against writing to or reading from the wrong flexible disk. On the first write to the destination flexible disk, the file FCOPY.$$$ is created on the destination disk. From then on, before the program writes, it checks to find FCOPY.$$$. And before it reads, it checks that FCOPY.$$$ is not on the disk.

FILECOPY error messages are:

Filename is too long

No source filename entered

Invalid - name or extension is too long

Invalid drive

Unable to complete copy - random write error

Source file not found

The wrong flexible disk has been inserted

FORMAT - TO FORMAT FLEXIBLE DISKS

The FORMAT program operates on either a one or two flexible-disk drive system. It provides prompts to indicate when to remove the CP/M flexible disk when you are using the Type 1 disk drive to format a new flexible disk.

Before using FORMAT, be sure that the write-protect slot on the flexible disk to be formatted is covered with opaque tape. Refer to your 110 Operating Guide for detailed instructions.

The program formats single-density, single-sided flexible disks in the IBM standard format of 26 sectors (128 bytes per sector) on each of 77 tracks. It is not necessary to format flexible disks if they already have this format.

The program formats double-density, double-sided flexible disks in the format used on the CDC CYBER 120 equipment, which is 77 tracks, 16 sectors/track, 512 bytes per sector. You must format ALL double-density, double-sided flexible disks before using them on the Control Data 110 system.

Detailed examples of the use of FORMAT are given in your 110 Operating Guide.

The FORMAT program is self-prompting. Prompts appear in succession after operator responses until the display appears as follows:

```
CDC 110 Disk Format routine
Version 3.01
12 May, 1983

Which drive do you want to format?  (A or B) b

Place disk to format in drive B
Type any character when ready


SINGLE DENSITY FORMAT

Format is Complete


Do you want to format another disk? (Y,N) n

A>
```

To format in Type 1 drive, respond to the first prompt with A. To format in Type 2 drive, respond with B. Follow prompts very carefully. Refer to 110 Operating Guide if necessary.

DISKINIT - TO FORMAT RIGID DISKS

This utility is used to format and analyze the rigid disk. It also permits you to see the total of disk surface faults. This section provides a brief description of the utility and procedures for using it. To use the format utility your system must include a Type 3 or Type 4 disk drive and a rigid-disk interface option board.

Each rigid disk is divided into two or three fixed size logical disks. A 12.5-megabyte rigid disk contains one 8-Mb logical disk and one 4-Mb logical disk. A 25-megabyte rigid disk contains three 8-megabyte logical disks. Each logical disk has a specific CP/M disk designator (from E to P). Every rigid disk occupies three CP/M disk designators regardless of disk size. On 12.5-megabyte rigid disks the third disk designator is not used. The following is an example of disk designators for four rigid disks.

DISK E, F	12.5 megabyte (8 Mb/4 Mb)
DISK H, I, J	25 megabyte (8 Mb/8 Mb/8 Mb)
DISK K, L	12.5 megabyte (8 Mb/4 Mb)
DISK N, O, P	25 megabyte (8 Mb/8 Mb/8 Mb)

The following paragraphs provide a description and procedure for DISKINIT.

DESCRIPTION

Each format procedure has four sections:

- Disk initialization.
- Formatting, including reformatting capability.
- Surface analysis to identify and disable bad sectors.
- Error logging and reporting.

Before rigid-disk media can be used, it must be properly formatted with the address information and data-block construction. The formatting process (including surface analysis) takes about 14 minutes per megabyte. Options allow you to format the entire surface of a 12.5-Mb or 25-Mb rigid disk, or to format a logical disk. You may also reformat a single file or a single sector. During the reformat operation of a single file or sector, all existing data in every reformatted file or sector is destroyed.

To prevent the destruction of usable data when reformatting a single sector, the system checks each sector before reformatting it. The program attempts to read the sector. If the sector is not readable, it is reformatted and verified (or allocated and disabled if it is a bad sector). If the sector is readable, it is not reformatted. You are informed that the sector was readable and therefore was not formatted.

Any storage media is subject to deterioration and flaws. To protect your data, a surface analysis is done to identify and disable unusable areas of the disk. After each format routine, the utility automatically analyzes the media surface for bad sectors.

When a bad sector is identified, it is flagged and listed in a bad sector file, so that the system will not attempt to use the sector. Read/write errors are logged and cumulative totals made available to you. Unrecoverable error totals are automatically displayed after surface analysis. In addition, the total percentage of disk space that is unusable due to bad sectors is accumulated. If 10 percent or more of the disk is unusable, an error message is displayed.

PROCEDURES

It takes about 1 hour and 52 minutes to format and analyze an entire 12.5-Mb rigid disk. You will notice that the system displays a message during any lengthy process. This message tells you what part of the utility process the system is performing.

Any errors detected while the utility is executing are displayed. You may then press NEXT to retry the step; BACK to return to the last display; or STOP or CONTROL C to exit the utility.

To run the utility, type: DISKINIT <next>. Then follow the prompts as in the examples in section 9.

SYSGEN

SYSGEN is a system copy program that transfers CP/M 2.2 from an existing Control Data CP/M 2.2 flexible disk to another flexible disk. The program only operates on double-sided, double-density flexible disks that have been formatted on the Control Data 110 system by the FORMAT program.

Section 3 gives a detailed example on the use of this program to transfer the Control Data CP/M system from one flexible disk to another using one flexible-disk drive.

The following is an example of the display you see on transferring from a flexible disk in the Type 1 disk drive to another flexible disk in the Type 2 disk drive. Follow the prompts and select the letter A or B to specify the source and destination of the CP/M system to copy. A specifies Type 1 disk drive, and B specifies Type 2 disk drive.

```
CDC SYSGEN for CP/M 2.2
VERSION 2.00

Source drive ? (or RETURN to skip)a
Put source disk on A, then type RETURN
Function completed
Destination drive ? (or RETURN to terminate)b
Put destination disk on B, then type RETURN
```

DSKUTIL

DSKUTIL is a general-purpose utility for use on flexible disks used in the Control Data 110 CP/M system. You must have both a Type 1 and a Type 2 disk drive to use DSKUTIL. The utility program is menu driven and lets you:

1. Make and verify a track-for-track copy of either a single-sided, single-density flexible disk or a double-sided, double-density flexible disk.

2. Verify a copy made with option 1.

3. Perform surface analysis of either a single-sided, single-density flexible disk or a double-sided, double-density flexible disk. Surface analysis reports any bad sectors and tracks on the flexible disk.

The following paragraphs provide operating instructions and discuss track-for-track copy-and-verify, track-for-track verify, and surface analysis.

OPERATING INSTRUCTIONS

To use this program reply to the system prompt as follows:

```
A>DSKUTIL <next>
```

This starts the utility. If necessary, you can then remove the flexible disk containing this program and run whatever utility option you choose. Simply follow the prompts. Section 6 gives examples of this program.

TRACK-FOR-TRACK COPY-AND-VERIFY

This routine makes and verifies an exact copy of a single-density flexible disk onto another single-density flexible disk or an exact copy of a Control Data 110 double-density flexible disk to another formatted double-density flexible disk. Any read errors are reported.

Before making a copy you must have formatted flexible disks of the correct type. Single-sided, single-density flexible disks are normally purchased with the correct format. Double-sided, double-density flexible disks must be formatted using the FORMAT utility.

The track-for-track copy reports any bad sectors. Data is copied as recovered after ten attempts to read the bad sector. A "hard" read error is reported as an unrecoverable read error if ten tries were made to read the data without a correct read. In this case, the bad data is copied. A "soft" read error is reported with the number of read retries made before a successful read. The data thus read is correct.

Follow the prompts to operate this routine. If you get bad sectors on the copy, the sector errors might be caused by a piece of dirt on the flexible disk. If so, the reported bad sector may disappear if you run the surface analysis utility on the source flexible disk several times. Once the errors stop appearing during the analysis, make a copy of the source flexible disk and retire it.

To recover from bad sectors you can also try using a head-cleaning flexible disk on the source disk drive and then trying the copy routine again.

TRACK-FOR-TRACK VERIFY

The copy made by the track-for-track copy-and-verify routine may be verified by running the verify routine after making the copy. This provides a check on the quality of the copy flexible disk and proves that it can be read correctly. The verify routine only reports tracks that do not verify and should give verify errors if there were read errors on the original flexible disk.

If the source flexible disk did not show any read errors, any errors shown by the verify option are probably due to a bad destination flexible disk. Replace the destination flexible disk and make a new track-for-track copy.

SURFACE ANALYSIS

This option reads each sector of the single- or double-density flexible disk and reports any sector read errors. Since it only reads the flexible disk, it can be used to check the quality of any Control Data 110 formatted and written-to flexible disk. You can run the surface analysis on flexible disks that were formatted using the FORMAT program. Surface analysis then checks the media quality of your newly formatted flexible disks.

This option may also be used to "sweep" debris from a flexible disk to try to correct any read errors on the flexible disk. Repeat the analysis several times to see if the read errors disappear.

If you analyze a flexible disk that has read errors and the errors disappear after several analysis passes, the flexible disk is probably dirty. Make a copy of the flexible disk and then retire the source disk.

BACKUP

The BACKUP utility is used to copy files from the rigid disk to flexible disks for backup storage or use. In addition, it permits you to see information about the backup, about the files backed-up, and to recover backed-up information from flexible disks to the rigid disk. To use the BACKUP utility you must have at least one Type 3 or Type 4 disk drive and a rigid-disk interface option.

Each rigid disk is divided into two or three fixed size logical disks. A 12.5-megabyte rigid disk contains one 8-Mb logical disk and one 4-Mb logical disk. A 25-megabyte rigid disk contains three 8-megabyte logical disks. Each logical disk has a specific CP/M disk designator (from E to P). Every rigid disk occupies three CP/M disk designators regardless of disk size. On 12.5-megabyte rigid disks the third disk designator is not used. The following is an example of disk designators for four rigid disks.

DISK E, F	12.5 megabyte (8 Mb/4 Mb)
DISK H, I, J	25 megabyte (8 Mb/8 Mb/8 Mb)
DISK K, L	12.5 megabyte (8 Mb/4 Mb)
DISK N, O, P	25 megabyte (8 Mb/8 Mb/8 Mb)

The following paragraphs provide a description and procedure for BACKUP.

DESCRIPTION

The rigid-disk backup utility permits copying files from the rigid disk to flexible disks as backup; accessing information about the backup; accessing information about the files backed-up; and recovering files from flexible disks to rigid disks.

The backup utility has two methods by which you can specify which files are to be backed-up. The first method is to select file name(s) from the disk directory. The second method is to specify a CP/M file that contains the name(s) of the files to be backed-up. The text file containing the list of files may have only one filename per line. The filenames must be in this format:

E:FILENAME.TYPE

Where:

E: designates disk on which file resides; any valid CP/M disk designator, but NEVER generic designator (never *:).

FILENAME is any filename up to eight characters that would be valid if used in PIP under CP/M (generic filenames are valid).

TYPE is any file type up to three characters that would be valid if used in PIP under CP/M (generic types are valid).

Any combination of these methods can be used provided no more than 248 files are selected in total.

Before actually copying the files to the flexible disk, the backup system analyzes the files to be backed-up and builds directory entries. Files are written on the flexible disk as standard CP/M files with standard directory entries. The first CP/M file on each backup flexible disk is the identifier file.

During the backup operation a pointer is kept showing the last file copied to flexible disk. If the operation is interrupted, this option reads the pointer and determines where to continue the operation.

The backup utility offers a verification option. If you choose this option, the file is verified after it is backed up to the flexible disk. The system displays the verification status for each file. If any error is detected during verification, the BACKUP utility is stopped. When this occurs, the backup disk is still good for the files that were backed up before the error.

Another option in the backup utility displays the information concerning the backup operation itself. It includes backup disk information: the time and date of the backup, the description of the backup, the backup ID, the numeric position of the disk in the backup series, the number of disks in the backup series, the total files in the backup series, and the total files in each individual disk.

This utility can also display information about the files that have been backed-up on the flexible disk. The following information is listed for each file that was backed-up to the disk:

- File name and type.
- File length in sectors.
- Disk number in the series of disks used for the backup on which the file begins.
- Disk number in the series of disks used for the backup on which the file ends.
- Origin disk of the file.

The BACKUP utility allows you to recover any files that have been backed up to a flexible disk. The files are recovered in the same order they were copied. Therefore, the system asks you to insert the first flexible disk in the backup series. That flexible disk has a list of all files copied during the series. You select the files you want to recover and specify to which rigid disk you want them copied. When recovery from the first flexible disk is complete, you will be asked to insert the second flexible disk, and so on.

If, during recovery, the system comes across a file that already exists at the specified location, the program asks for authority to delete the file and write in the new one. You will see an error message if there is not enough room for a file.

It takes about 10 minutes to transfer 1.2 Mb of data without verification (the contents of one double-density, double-sided flexible disk, approximately). You will notice that the system displays a message during any lengthy process. This message tells you what part of the utility process the system is performing. Many of the displays you will see include the statement "HELP is available." You may press the HELP key to see more information about the displayed material.

Any errors that may be detected while the utility is executing are displayed. You may then press NEXT to retry the step; BACK to return to the last display; CTRL C to exit the utility. You can also press the HELP key to learn more about the error message. To run the utility, type: BACKUPS <next>. Then follow the prompts as in the examples in section 10.

DATTIM

The CP/M 2.2 system has a date and time clock. The clock routine updates the time-of-day. It advances the date at midnight. The routine uses a 24-hour clock so PM times run from 12:00 to 23:59.

NOTE

> The month and the year are not automatically updated.

A time-setting utility is on the CP/M flexible disk. To use it:

Type: DATTIM <next>

Follow the prompts to set the date and time.

PHONE

The PHONE utility is used to create and maintain a directory of up to 99 phone number entries. The utility is also used to communicate, via your terminal, with the computer or terminal locations that correspond to the phone number entries. To use the PHONE utility, you must have the following equipment:

- Type 3 terminal
- Internal 1200/1200-baud modem
- Type 1 disk subsystem

The following paragraphs include a description and procedure for the PHONE utility.

DESCRIPTION

The utility contains ten different commands. Some commands are required to create and update the phone number directory while others are used to activate and disconnect phone connections to other terminals and computers. The ten commands and their functions are listed below.

Add - To add an entry to the directory.

Change - To change an entry in the directory.

Dial - To dial an entry automatically.

Find - To find and display an existing entry.

Hangup - To hang up the phone when a data call is finished.

Remove - To delete an entry from the directory.

Stop - To stop the utility.

Tone - To send tone-coded data.

+ or - - To roll the display up (+) or down (-).

Values - To change up to four of the various default values which are relevant when adding an entry to the directory.

PROCEDURE

To begin using the PHONE utility, you must follow this procedure:

1. Turn on the power for your terminal and disk subsystem.
2. Load your CP/M disk into the disk subsystem.
3. Press the F3 key in the top row of your keyboard. The terminal message appears, indicating the version of your disk. If the version of your CP/M disk is below 5.00, you do not have access to the PHONE utility.

   This prompt also appears under the terminal message:

   A >

4. Type: PHONE <next>

   The Phone Directory Display Screen appears.

The display screen is divided into four horizontal parts. These are the message area, the menu/command area, the entry being built line, and the phone directory area. You type commands in the menu/command area, and the new or modified entry appears on the entry being built line. The phone directory area displays any new or modified entry in the center of the area, highlighted and surrounded by alphabetically adjacent entries. System messages and error messages appear in the message area of the screen. All errors are accompanied by a "beep" tone, and most errors are correctable immediately.

Section 11 of this manual contains detailed information about the PHONE utility, including procedures for the use of each of the commands and a listing and explanation of all the possible error messages.

This section and the appendixes are written for the person who will implement application programs on the Control Data 110 Viking System using the CDC version of CP/M 2.2. It is assumed that you are an experienced programmer familiar with assembly-language programming. You should have the Digital Research set of manuals on CP/M and be familiar with them. (See the preface of this manual for publication numbers.)

This section provides information about the Control Data 110 Viking System, the system sign-on message, the SYSGEN program, the organization of Control Data 110 BIOS, terminal and printer characteristics, communications, customizing the Control Data 110, further communications programming information, automatic flexible-disk density sensing, date and time-clock operations, example programs, and the KEYTEST.COM routine.

CONTROL DATA 110 VIKING SYSTEM

The following paragraphs provide useful information about the equipment in the Control Data 110 Viking System and some details about input/output buffering.

VIKING SYSTEM EQUIPMENT

The Control Data 110 Viking System is a multi-processor system consisting of a Type 1 disk drive that contains a 4-mHz Z-80 processor with 64K of memory and a terminal which also contains a 4-mHz Z-80 processor with 64K RAM, along with optional equipment, such as a Type 2 flexible-disk drive, rigid-disk drives, printers, and modem.

The Type 1 disk drive uses a flexible-disk controller chip to access up to 2 double-density, double-sided flexible disks. The only communication from the disk controller is to the terminal.

The terminal provides access to the keyboard and display, and also has RS-232 ports to external equipment and a CDC parallel channel that communicates with the disk controller and the Type 1 (graphics) printer. If the system includes rigid-disk drives, the rigid-disk option (RDO) must be installed in the terminal.

The terminal has a Z-80 processor and some internal memory for the controlware. It also has a resident read-only memory (ROM), which provides keyboard and screen display characteristics. The terminal may be loaded from the disk controller with software (or controlware) that provides the unique CP/M characteristics.

The Control Data 110 Basic Input/Output System (BIOS) routine is resident in both the disk controller and in the terminal. The disk controller portion provides the interface to user programs, while the terminal controlware provides the I/O driver routines.

INPUT AND OUTPUT BUFFERING

All input to the system is interrupt-driven where the received bytes are stored in an input buffer. Input buffers are 264 bytes for communications lines and 40 bytes for the keyboard. All output is on a polled basis so that data is taken from the 40-byte output buffer when there is data available. (The screen has an 8-byte buffer)

An 8-byte stub of each buffer is contained in the disk controller, while the remaining bytes of the buffer are contained in the terminal. The terminal idle loop constantly attempts to make transfers between the disk controller board buffer and the terminal buffer as data and space are available.

SYSTEM SIGN-ON MESSAGE

The system sign-on message displayed when CP/M is bootstrapped contains vital information about the version of the BIOS routine, the version of the terminal controlware, and the available user space. If you have any problems with your system be sure to include this information from the sign-on message with any questions. The following paragraphs discuss version identification, terminal controlware, user memory space, and use of MOVCPM to change user memory space. This is an example of the sign-on message:

```
Control Data 110 using Type 3 (Display) terminal
Copyright 1982, Control Data Corporation
Terminal Controlware Vers.  5.00

64k CP/M vers 2.2 for Control Data 110  BIOS 5.10
52k user space

RIGID DISK(S) NOT READY

A>
```

VERSION IDENTIFICATION

Control Data 110 CP/M 2.2 is an evolving program, which is released at a number of different levels. The version is identified on your master flexible disk external label by an "L" number. The system also identifies itself by version numbers for the BIOS and the terminal controlware when it is loaded into your Control Data 110 system. The following versions exist as of the date of this manual:

"L" Number	BIOS Version	Terminal Controlware Version
L0280	2.12	2.07
L0330	2.20	3.08
L0390	4.03	4.04
L0470	5.12	5.02

TERMINAL CONTROLWARE

The controlware for the terminal is stored on the CP/M disk as file TERMINAL.COM. This is NOT a command file and should not be executed. It should not be renamed and must be resident on the CP/M disk for the system to boot.

USER MEMORY SPACE

The sign-on message when CP/M is loaded gives available user memory space in "K" units. This space definition includes all memory from location 0 to the start of the CP/M BDOS routine.

The exact user memory space can be calculated as in the following example:

- Sign-on gives 52K user space.
- 52 by 1024 = 53 248 bytes available.
- Subtracting 256 bytes for low memory not usable:

  53 248 - 256 = 52 892 actual bytes of memory for user program.

Note that the actual memory space may be less than shown here. To maintain compatibility with future updates of Control Data 110 CP/M 2.2 and also with future updates of CP/M, no user should plan on using more than 48K (48 896 bytes of actual memory). User programs should be designed to operate in this minimum amount of memory, but they can make use of additional memory as it is available in any particular system.

USE OF MOVCPM

The program MOVCPM can be used to create smaller CP/M systems for the Control Data 110. Normally you will use a 64K system for maximum memory, but you may wish to try programs using a smaller system to guarantee future compatibility. MOVCPM only works with BIOS version 2.00 and higher.

As an example: to generate a 60K CP/M system, type:

```
A>MOVCPM 60 * <next>
```

The new system is generated in memory ready to save or SYSGEN.

Immediately do a SYSGEN and don't read a system, but write it to a new flexible disk. If you write back to your system flexible disk on the Type 1 disk drive, be sure to reload CP/M. If you don't, the system will hang.

To generate a full size CP/M system type:

```
A>MOVCPM 64 * <next>
```

And then do a SYSGEN.

CAUTION

> The method just described is the only one that will not hang the system.

THE SYSGEN PROGRAM

The Control Data 110 CP/M flexible disk is not a standard CP/M 2.2 flexible disk. SYSGEN is a system generation program that transfers CP/M 2.2 from an existing flexible disk to other flexible disks. It only operates on flexible disks that have been formatted by the FORMAT program to the double-density, double-sided format.

To generate a system, use this document along with the CP/M 2.2 Alteration Guide. The program reads the system from tracks 0 and 1 and copies it to tracks 0 and 1 of the new system disk. Once the system has been read into memory (using MOVCPM, then DDT to get BOOT and BIOS), memory is allocated as follows:

980h	CP/M CCP
1180h	CP/M BDOS
2000h	BOOT
2200h	BIOS
4200h	end of BIOS

The disk format is as follows. The format is for double-density 512-byte sectors, 16-sectors per track (0...15).

Head	Cylinder	Track	Sector	Logical Sector	Routine
0	0	0	0	0..3	BOOT
0	0	0	1..7	4..27	BIOS
1	0	1	0..5	0..09	CCP,BDOS

ORGANIZATION OF CONTROL DATA 110 BIOS

The following paragraphs discuss the organization of the Control Data 110 BIOS. BIOS vector location and values are provided; using the IOBYTE function is explained. The rigid-disk BIOS interface and error logging and recovery are also discussed.

BIOS VECTOR LOCATION AND VALUES

The location of the BIOS vector that gives you access to the BIOS routines can be found by reading the contents of memory locations 1 and 2 as a word. This location contains the address of the second vector in BIOS.

The BIOS location depends on available user memory. The user memory is listed in the sign-on message when CP/M is loaded on the Control Data 110 system with BIOS version 2.00 or greater. For BIOS version 5.00 or greater the user memory is 52K.

If you have made a smaller CP/M system by running MOVCPM, then you must look at the contents of memory location 1 and 2 to find where the BIOS is located.

The following are the Standard CP/M BIOS vectors for a 52K user memory system.

BIOS Offset	Vector Name	Vector Description
00	CBOOT	(Don't use this entry)
03	WBOOT	Warm boot
06	CONST	Console status to A
09	CONIN	Console input to A
0C	CONOUT	Console output from C
0F	LIST	List output from C
12	PUNCH	Modem control output from C
15	READER	Modem status input to A
18	HOME	Set disk track to 0
1B	SELDSK	Select disk drive in C
1E	SETTRK	Set track number in BC
21	SETSEC	Set sector number in C
24	SETDMA	Set DMA address from BC
27	READ	Read disk data to DMA address
2A	WRITE	Write disk data from DMA address
2D	LISTST	List status to A
30	SECTRN	Translate sector number
33	(RESERVED)	
36	(RESERVED)	
39	(RESERVED)	
3C	(RESERVED)	
3F	(RESERVED)	
42	DSSIOP	Rigid-Disk I/O Packet
45	DSSDOT	Rigid-Disk Data Output
48	DSSDIN	Rigid-Disk Date Input
4B	DSSIST	Rigid-Disk Input Status
4E	MDCTRL	1200/1200 Modem Control

USING THE IOBYTE FUNCTION

For added flexibility, CDC CP/M 2.2 has implemented the IOBYTE function that allows reassignment of physical and logical devices. The IOBYTE function creates a mapping of logical to physical devices that can be altered during CP/M processing using the STAT command. The definition of the IOBYTE function is as follows:

A single location in memory (location 0003h) is maintained, called IOBYTE, which defines the logical-to-physical device mapping which is in effect at a particular time. The mapping is performed by splitting the IOBYTE into four distinct fields of two bits each, called the CONSOLE, READER, PUNCH, and LIST fields, as shown below:

	BITS	7 6	5 4	3 2	1 0
IOBYTE at 0003h		LIST	PUNCH	READER	CONSOLE

The value in each field can be in the range 0 to 3, defining the assigned source or destination of each logical device. The values that can be assigned to each field are given below with the corresponding device name used in STAT to set individual values.

CONSOLE (CON:) field (bits 1, 0)

- 0 - Console is assigned to the CRT and keyboard device (TTY:)
- 1 - Reserved (CRT:)
- 2 - Input from communications line, output to communications line (BAT:)
- 3 - Reserved (UC1:)

READER (RDR:) field (bits 3, 2)

- 0 - READER input is communications status (TTY:)
- 1 - READER input is from touchpanel (RDR:)
- 2 - Reserved (UR1:)
- 3 - Reserved (UR2:)

PUNCH (PUN:) field (bits 5, 4)

- 0 - PUNCH output is to communications control (TTY:)
- 1 - Reserved (PUN:)
- 2 - Reserved (UP1:)
- 3 - Reserved (UP2:)

LIST (LST:) field (bits 7, 6)

0 - LIST output is to system printer (TTY:)
1 - LIST output is to CRT screen (CRT:)
2 - LIST output is to system printer (LPT:)
3 - LIST status is communication output buffer status (UL1:)

Note that this organization of the IOBYTE function is unique to the Control Data 110 CP/M 2.2 system and is designed for use by the user programs. No standard CP/M system program makes use of the IOBYTE function except that PIP allows access to the physical devices and STAT allows the logical-physical assignments to be made and/or displayed.

CAUTION

Reserved values are used to prevent the system from hanging up. These assignments are subject to change in future versions of CP/M. To guarantee forward compatibility, no user program should use reserved values.

RIGID-DISK BIOS INTERFACE

The rigid-disk BIOS interface modifies the Basic Input/Output System (BIOS) of the CP/M to support the FINCH hard disk (rigid disk drive). The interface contains two parts:

- The CP/M Basic Disk Operating System (BDOS) interface
- The communication protocol between the terminal and the rigid disk interface board, necessary for accessing the rigid disk.

The BIOS routine is resident in both the disk controller and in the attached terminal. The controlware for the terminal is named as file TERMINAL.COM with revisions of 4.0 and up.

The following paragraphs discuss the CP/M initialization routines, the BDOS interface, and the communication needed to access the rigid disk.

CP/M Initialization Routine

At system start-up or re-boot (cold boot), CP/M brings in the rigid-disk initialization routine to test the Interface Option board and the rigid disk subsystem (DSS). CP/M issues an EXECUTE DIAGNOSTIC command to the Interface Option board during initialization. If there is no response after two seconds, CP/M assumes that this is not a rigid disk system, skips the rest of the initialization test, and continues with its cold boot operation.

The EXECUTE DIAGNOSTIC command starts the Interface Option board's self-tests to check its RAM and ROM memory. If any of the self-tests fail, the system displays the following error message:

RIGID DISK INTERFACE BOARD FAIL

and bypasses any further rigid disk tests.

After the Interface Option board completes the self-test, it tries to communicate with each of the rigid disks that is connected to the system. If the Interface Option board cannot communicate with any one of the rigid disks, the system displays the following message:

RIGID DISK(S) NOT READY

This message indicates that none of the disks is ready for operation. Either the power is off, or the disk(s) is performing the power-on self-test. The message may also appear if the terminator is missing.

After it completes the EXECUTE DIAGNOSTIC command, the CP/M issues a SELECT command to each rigid disk connected to the Interface Option board. The SELECT command supplies the CP/M with the rigid disk size, logical disk size, and write-protect switch status of each disk.

Each rigid disk is divided into two or three fixed size logical subdisks. A 12.5-megabyte rigid disk consists of one 8-Mb (8.38-megabyte) logical disk and one 4-Mb (4.19-megabyte) logical disk. A 25-megabyte rigid disk contains three 8-Mb (8.38-megabyte) logical disks. Each logical disk has a CP/M disk address designator (from E to P). Every rigid disk occupies three CP/M disk designators regardless of disk size. The third disk designator for a 12.5-megabyte rigid disk is not used.

During the initialization process, a disk description message appears on the screen after CP/M issues the SELECT command to the rigid disk(s). The message informs users of the actual rigid disk configuration(s). A sample message appears below:

```
DISK E,F                              12.5 Megabyte (8 Mb/4 Mb)
DISK H,I,J    WRITE PROTECT             25 Megabyte (8 Mb/8 Mb/8 Mb)
DISK K,L                              12.5 Megabyte (8 Mb/4 Mb)
DISK N,O,P                              25 Megabyte (8 Mb/8 Mb/8 Mb)
```

The message indicates that the system has two 12.5-megabyte rigid disks, two 25-megabyte rigid disks, DISK H,I,J is a 25-megabyte disk, and the WRITE PROTECT switch is set.

If the display message for a select disk is:

```
DISK E,F,G  ERROR ON SELECT
```

the rigid disk with device code 26 (octal) may have failed to respond to the SELECT command, or the disk may not be ready.

BDOS Interface

The BDOS interface provides the CP/M support for all the disk functions of the rigid disk in the same manner as for the flexible diskettes.

Each disk drive has an associated (16 bytes) disk parameter header for CP/M. The header contains information about the disk drive and provides the scratchpad area for certain BDOS operations. The format of the disk parameter header is as follows.

DISK PARAMETER HEADER

XLT	0000	0000	0000	DIRBUF	DPB	CSV	ALV
16 b	16 b	16 b	16 b	16 b	16 b	16 b	16 b

XLT — Address of the logical to physical translation vector. No translation is required in software for rigid disk operation.

0000 — Scratchpad values for use within the BDOS.

DIRBUF — Address of a 128-byte scratchpad area for directory operations within BDOS. The address is DIRBUF.

DPB — Address of a disk parameter block. The address for a 8-Mb logical disk is DPBD8, for a 4-Mb logical disk is DPBH4.

CSV — Address of a scratchpad. Area used for software check for changed disks. The software check for a hard disk is 00.

ALV — Address of a scratchpad area used by BDOS to keep disk storage allocation information.

The allocation unit for a 8-Mb logical disk is 8 kb. The CP/M disk characteristics are:

```
Record Capacity (128 byte) = 64640
Drive Capacity             = 8080 kb
32 Byte Directory Entries  = 512
Check Directory Entries    = 0
Record/Extent              = 512
Record/Block               = 64
Sector/Track               = 128
Reserved Tracks            = 7
```

The allocation unit for a 4-Mb logical disk is also 8 kb. The disk CP/M characteristics are:

```
Record Capacity (128 byte) = 31872
Drive Capacity             = 3984 kb
32 Byte Directory Entries  = 512
Check Directory Entries    = 0
Record/Extent              = 512
Record/Block               = 64
Sector/Track               = 128
Reserved Tracks            = 7
```

In the CP/M system, each rigid disk will be assigned to three logical disks which are associated with a physical address of the disk. The CP/M disk designating addresses and their physical addresses are listed below:

CP/M Disk Designator	Physical Disk Address
Disk E, F, G	26_8
Disk H, I, J	36_8
Disk K, L, M	46_8
Disk N, O, P	56_8

For a 12.5-megabyte rigid disk, the last disk designator is not used. If that disk designator is selected, CP/M will return a RDOS error on select.

Accessing the Rigid Disk

Four BIOS entry points are available through which you may directly interface the rigid disk. The following paragraphs discuss the four BIOS entry points. Further information about the command words may be found in appendix F.

Rigid-Disk I/O Packet

The rigid-disk I/O packet (DSSIOP) provides the command words for the rigid-disk adaptor. The BIOS entry point for the DSSIOP command set is:

Entry point:	BIOS + 42h	
Enter:	Reg B	- Upper 8-bits of first command word
	C	- Lower 8-bits of first command word
	D	- Upper 8-bits of second command word
	E	- Lower 8-bits of second command word
Exit:	A = 0	- Operation success

Rigid-Disk Data Output

With the rigid-disk data output (DSSDOT) command, up to 256 words (512 bytes) of data are shipped to the rigid-disk interface board. These words may either be written on the rigid disk or written into the RAM of the rigid-disk interface board for later execution. The H and L registers contain the first word address of the buffer area. Register B contains the total word count less 1. In case the WRITE/WRITE RAM command has not been issued by DSSIOP, DSSDOT returns FE in the A register to indicate an illegal operation. The BIOS entry point for the DSSDOT command is:

```
Entry Point:  BIOS + 45h

Enter:        Reg. B - Word count-1 (up to 256 words)
                H, L - Buffer data address

Exit:         A = 0  - Operation success
                = FE - Illegal operation
```

Rigid-Disk Data Input

With the rigid-disk data input (DSSDIN) command, up to 256 words (512 bytes) of data are input from the rigid-disk interface board and stored at the address specified in registers H and L. Register B contains the word count less 1. If the READ/READ RAM command has not been issued, this routine returns FE in the A register to indicate an illegal operation. The BIOS entry point for the DSSDIN command is:

```
Entry Point:  BIOS + 48h

Enter:        Reg. B - Word count-1 (up to 256 word)
              H, L   - Buffer data address

Exit:         A = O  - Operation success
                = FE - Illegal Operation
```

Rigid-Disk Input Status

At the end of some operations, the rigid-disk interface board sends a status word to the terminal driver. This status word can be returned to the user through the rigid-disk input status (DSSIST) routine. The highest bit of the status word is assigned for valid status bit. The valid status bit is 0 to indicate that the status is ready. This bit is reset to 1 after each DSSIOP is issued, but the rest of the status bits are not altered. Register A also is used to indicate the valid status word. The BIOS entry point for the DSSIST command is:

Entry Point:	BIOS + 4Bh	
Enter:	None	
Exit:	A = O	- Valid status word
	= FF	- Invalid status word
	H, L	- Status word

ERROR LOGGING AND ERROR RECOVERY

When an error is encountered while reading or writing data on the rigid disk, the same attempt is retried ten times. If after ten retries the read or write still fails, the user is informed of the error. A soft error occurs when an operation fails the first time but succeeds on the later retries. A hard error is encountered when all ten attempts fail. Hard errors cause processing to stop. The total accesses and the number of soft errors for each disk are logged when a read or write is operating. The access count and error count for each disk are located in a fixed address as follows:

Definition	Memory Location
Access count for Disk E, F, G	0044 - 0046 (hex)
Access count for Disk H, I, J	0047 - 0049 (hex)
Access count for Disk K, L, M	004A - 004C (hex)
Access count for Disk N, O, P	004D - 004F (hex)
R/W error count for Disk E, F, G	001C - 001E (hex)
R/W error count for Disk H, I, J	001F - 0021 (hex)
R/W error count for Disk K, L, M	0022 - 0024 (hex)
R/W error count for Disk N, O, P	0025 - 0027 (hex)

NOTE

CP/M cold boot clears all counters. The user must reset these counters as desired; the CP/M utility DDT can be used for this purpose.

MODEM BIOS INTERFACE

The CP/M BIOS provides asynchronous communication support through the Type 3 terminal. The communication driver routines in the terminal are part of TERMINAL.COM, which is automatically loaded when CP/M is booted. The COMM 3 channel is assigned for the optional internal 1200/1200-baud modem.

The BIOS provides one entry point for the application software to interface to the modem. The entry point for internal modem control is MDCTRL. In MDCTRL, the A register contains the modem entry point index and the B and C registers contain the parameter address. Upon the call return, if the modem is not installed, or if the dialed number does not end in "F," the A register receives a nonzero value.

The MDCTRL entry point is as follows:

Entry Point:	BIOS + 4EH	
Enter:	Reg A = 00	Autodial
	= 01	Get phone call status
	= 02	Set modem control parameters
	= 03	Auto-answer (continuous)
	= 04	Auto-answer (not continuous)
	= 05	Autodial (uses mode default parameters)
	= 06	Send DTMF information (tone dial only)
	Reg B,C =	(see individual routines)
Exit:	Reg A = FF	Operation unsuccessful

The following paragraphs describe the individual routines included in MDCTRL.

Autodial Routine

The autodial routine also includes four separate functions. Each is included in the autodial description. For autodial:

Enter: Reg A = 0
Reg B = The upper eight bits at the parameter list address
Reg C = The lower eight bits of the parameter list address

Parameter List

Address	List	Parameter	Function
(BC)+0	XX	1	Miscellaneous control
(BC)+1	XX	2	Ring count
(BC)+2	XXXX	3,4	Redial Attempts (Retries)
(BC)+4	XX	5	Phone number
(BC)+5	XX	6	Phone number
.	.	.	.
.	.	.	.
.	.	.	.
(BC)+N	46 (Hex)	N+1	Call terminator "F"

Miscellaneous Control Function

data byte

Bit	Function	7	6	5	4	3	2	1	0
7,6,5	(Not Used)	X	X	X					
4	D/V Select - Data				0				
	Voice				1				
3	Display Status Enabled					0			
	Messages Disabled					1			
2	Retry Number Specified						0		
	Continuous Retries						1		
1	Must be 0 (not used)							0	
0	Dial mode - Pulse								0
	Tone								1

Bit 4 is used for autodialing and voice communication.

Bit 3 enables/disables the display of status messages on the screen.

Bit 2 controls how often the modem will redial a number.

Bit 1 must be reset (must = 0).

Bit 0 controls the dialing mode.

Ring Count Function (1 byte)

The ring count parameter may be in ASCII characters 0 through 9. J4 indicates the number of rings allowed before a call attempt is aborted. A value of 0 defaults to five rings.

Retries Function (2 bytes)

The retry parameter may be in ASCII characters 00 through 99. It controls the number of times the modem redials a number before returning to the caller.

The first byte contains the units digit, and the second byte contains the tens digit. Bit 2 of the MISC CONTROL byte must be set to 0. An ASCII value of 99 causes continuous retries until a connection is established, or until the call attempt is aborted by a reset.

NOTE

> Applications written for use in Canada are restricted to a maximum of ten retries under DOC regulations.

Phone Number Function

The phone number must be in the following format, using ASCII characters only:

0-9 (digits)
* or # (valid only for tone dial)
!, ?, A, F (control characters)

Control characters may be imbedded within a phone number. An exclamation point (!) causes a "wait until tone detected" condition before the dial sequence is continued. A question mark (?) causes a three second delay after the tone is no longer detected before the dial sequence is continued.

The last character of the sequence must be A or F. An A indicates that the number to follow is an alternate. An F indicates that no additional numbers follow.

The following are several examples of phone numbers:

1.	16124436019F	One number
2.	16124436019A7765043F	One number and alternate
3.	93314408!009301569187496O6016	Network access (the ! indicates pause for tone)

Get Phone Call Status Routine

The routine to get phone call status is as follows:

Enter: Reg A = 01
Reg B = The upper 8 bits of the status table address
Reg C = The lower 8 bits of the status table address

Upon return, the BC register points to the beginning of a six byte status table. The table has the following format and contents:

Byte 1 - Miscellaneous Status
Byte 2 - Call Status
Byte 3 - Self-Test Status
Byte 4 - ROM Revision
Byte 5 - Dial Retries (units digit)
Byte 6 - Dial Retries (tens digit)

Each byte is described separately in the subsequent paragraphs.

Miscellaneous Status Byte

The miscellaneous status byte has the same functions as those described under Autodial Miscellaneous Control.

Call Status Byte

The call status byte has the following format and functions.

Bit	Function	7	6	5	4	3	2	1	0
		data byte							
7	Primary number dialed last	0							
	Alternate number dialed last	1							
6	(Not Used)		X	1					
5	Local loopback (OFF)			0					
	Local loopback enabled (ON)			1					
4	No ring indicator				0				
	Ring indicator active				1				
3	Not waiting for carrier					0			
	Waiting for carrier (no answer)					1			
2	Not busy						0		
	Busy						1		
1	Not ringing							0	
	Ringing							1	
0	On hook (not connected)								0
	Off hook - connected or data only mode								1

Self-Test Status Byte

The self-test status byte provides information about modem tests, UART tests, ROM checksum test, and modem ready. The bit functions for the self-test status byte are shown below.

Bit	Function	7	6	5	4	3	2	1	0
		data byte							
7	Always = 0	0							
6	Modem Loopback Test (pass)		0	0					
	(fail)		1						
5	UART Loopback Test (pass)			0					
	(fail)			1					
4	ROM Checksum Test (pass)				0				
	(fail)				1				
3	Always = 0					0			
2	Always = 0						0		
1	Always = 0							0	
0	Modem Ready (no)								0
	(yes)								1

ROM Revision Byte

The ROM revision byte has the following format.

status byte 4

Function	7	6	5	4	3	2	1	0
Release (bit 7-4)	X	X	X	X				
Revision (bit 3-0)					X	X	X	X

Number of Dial Retries Byte

The Number of Dial Retry bytes (5 and 6) contain the ASCII values of the number of retries. The count will wrap around to 00 after a count of 99.

Set Modem Status Routine

The routine to set modem status is as follows:

Enter: Reg A = 02
Reg B (upper 8 bits of the set modem status parameter address)
Reg C (lower 8 bits of the set modem status parameter address)

Upon entry, the BC register contains the address of the set modem status parameter table. Each byte of the set modem status parameter is described separately below.

Modem Status Byte 1

status byte 1

Bit	Function	7	6	5	4	3	2	1	0
7	(Not used)	X							
6	Messages - Enabled		0						
	Disabled		1						
5	(Not used)			X					
4	LED - on				0				
	off				1				
3	Self-test - do not run					0			
	Run (causes hangup)					1			
2	Not used						X		
1	Local loopback mode - disable							0	
	enable							1	
0	On/off hook - no hangup (no action)								0
	Hangup (break connection)								1

Modem Status Byte 2

status byte 2

Bit	Function	7	6	5	4	3	2	1	0
7-4	Hex value of the number of rings to wait for answer (a value of 0 defaults to 2 rings)	X	X	X	X				
3	(Not used)					X			
2	(Not used)						X		
1*	Deactivate Auto Answer (no)							0	
	(yes)							1	
0	Activate Auto Answer (no)								0
	(yes)								1

*Note - Bit 1 has priority over bit 0.

If bit 0 is set (that is, =1) and bit 1 is not set (that is, =0) (indicating 'Activate Auto Answer'), then the firmware checks the ring indicator.

If a ring is present, the modem firmware will remain in control and connects the incoming call after the selected number of rings, and then returns control to the caller.

If no ring is present, the firmware will return with no other activity.

The user may determine the presence of a call by checking for either 'carrier on' or an 'off-hook' condition.

Auto-Answer (Continuous) Routine

The routine for auto-answer (continuous) is:

Enter: Reg A = 03

This call causes the modem firmware to remain in control until a ring status is present and a connection is made (answered in two rings).

Auto-Answer (Noncontinuous) Routine

The routine for auto-answer (noncontinuous) is:

Enter: Reg A = 04

This call causes the modem firmware to read the status of the ring indicator. If a ring is present, the firmware connects to the incoming call after two rings and then returns. If no ring is present, the firmware returns immediately.

Autodial (Uses Mode Default Parameters) Routine

The routine for autodial is:

Enter: Reg A = 05

This call allows the internal modem to autodial through use of the active mode parameters.

Send Dual Tone Modulated Frequency (DTMF) Information (Tone Dial Only) Routine

The routine to send DTMF information is as follows:

```
Enter:  Reg A = 06
        Reg B = Upper 8 bits of the list address
        Reg C = Lower 8 bits of the list address
```

Tone code information may be sent over the phone line at anytime by defining a list of bytes in memory, loading the BC register with the address of the first byte, and calling the Send DTMF Information routine. The format of the required list is illustrated below.

```
                                                   Data
    ---|---                               ------------------------------
       |                    (BC)--->     | BYTE 1               (ASCII) |
       |                                 |------------------------------|
       |                                 | BYTE 2               (ASCII) |
       |                                 |------------------------------|
     LIST                                |                              |
       |                                 |                              |
       |                                 |------------------------------|
       |                                 | BYTE N-1             (ASCII) |
       |                                 |------------------------------|
     LAST BYTE = ASCII "F" TO --->       | BYTE N           (ASCII "F") |
     INDICATE END OF "LIST"              |                              |
                                          ------------------------------
```

Data is stored as ASCII characters to represent the characters 0 through 9, *, and # as they appear on the touch-tone pad on the telephone. The last byte must be an ASCII F (hex 46) which indicates the end of the list.

This routine does not check any status (for example, Off Hook, Carrier Detect, etc.) prior to execution. It is the user's responsibility to ensure that a valid connection exists.

Use of the exclamation point (!) or the question mark (?) in the list will prohibit tone information from being sent.

MODEM HANG UP

The internal modem hangs up when:

- The BREAK key is pressed during the dialing process or when the BREAK key is pressed to exit from auto-answering mode

- HANGUP is entered on the keyboard and the NEXT key is pressed to hang up a data call

TERMINAL CHARACTERISTICS

The following paragraphs discuss the terminal keyboard and screen characteristics and changing the screen characteristics. The graphics option and the terminal touchpanel are described.

TERMINAL KEYBOARD

The terminal keyboard layout and the codes generated for each keystroke are shown in appendix B. The utility routine KEYTEST may be used to display the codes generated by the keyboard. The PRINT and SETUP keys on the terminal have special functions associated with Control Data 110 CP/M 2.2.

PRINT Key

Pressing the PRINT key (lowercase) or the SHIFT and COPY keys causes the printer to make a copy of the current contents of the display screen.

SETUP Key

Pressing SETUP causes the terminal to enter the Parameter Change Mode, where most screen characteristics can be temporarily changed. Such changes are effective until the terminal is powered off or RESET.

SCREEN CHARACTERISTICS

The terminal under Control Data 110 CP/M 2.2 has two modes of operation: compatibility mode and CYBER or advanced mode. The following paragraphs discuss these modes.

Compatibility Mode

The system begins running in compatibility mode whenever a cold boot is performed. This mode should always be used when running the Console Command Processor (CCP) and all utility routines under CP/M.

The location of data displayed on the screen and movement of the cursor is controlled by characters output from the computer to the terminal. The following paragraphs define all legal character strings and their action on the terminal screen.

The main terminal characteristics are based on the Televideo* 920 terminal. Some of the characteristics of the Hazeltine** 1420 terminal and the Soroc*** IQ 120 terminal are included. These characteristics of the Control Data 110 terminal permit configuration of purchased applications that use terminal characteristics and provide configuration programs having predefined terminals in them.

A list follows showing functions and their associated character strings for each of the three terminals. These abbreviations for the terminals are used:

TVI	Televideo 920
SOR	SOROC IQ 120
HAZ	Hazeltine 1420
all	All three terminals

Character strings are given as hexadecimal values of 2 digits, or as the corresponding ASCII character. A lower case d following a number denotes a decimal value. The Hazeltine escape sequences can be preceded by the ESC or (tilde).

Character String	Terminal	Function
07	all	Ring bell.
08	all	Backspace cursor.
09	all	Tab cursor to next multiple of 8 columns.
0A	all	Cursor down 1 line.
0B	TVI	Cursor up 1 line.
0C 10	TVI HAZ	Cursor forward 1 space.
0D	all	Carriage return. Cursor moves to beginning of line.

---

*Televideo 920 is a registered trademark of Televideo Systems Inc.
**Hazeltine 1420 is a registered trademark of Hazeltine Corp.
***Soroc IQ 120 is a registered trademark of Soroc Technology, Inc.

Character String	Terminal	Function
1A (TVI uses 1E,1A)	TVI	Clear screen. Cursor to upper-left corner.
1B,1C	HAZ	
1B,2A	SOR	
1E	TVI	Home cursor. Cursor to upper-left corner.
1B,12	HAZ	
7E,12	HAZ	
1E	SOR	
1B,41,ROW+20,COL+20	TVI	Set cursor to column, row.
1B,11,row(mod 32d)†,col(mod 96d)	HAZ	
7E,11,row(mod 32d),col(mod 96d)	HAZ	
1B,3D,row+20,col+20	SOR	
1B,42	TVI	Clear from cursor to end of line.
1B,0F	HAZ	
7E,0F	HAZ	
1B,54	SOR	
1B,59	SOR	Clear to end of screen from current cursor position.
1B,3F	TVI	Read cursor position return row+20, col+20, 0D
1B,6A	TVI	Set inverse video.
1B,6B	TVI	Clear inverse video.
7E, 1A	HAZ	Insert line.
1B, 1A	HAZ	Position cursor at first column of current row. Insert a blank row. Move following rows down.
7E, 13	HAZ	Delete line.
1B, 13	HAZ	Position cursor at first column of current row. Delete current row. Move following rows up.

The following pertains to the Control Data 110 system only:

1F		Cursor to beginning of next line and clear that line.

---

†Row value must be between 0 and 31 decimal. If value is greater than 31d, modulus 32d is taken and used as row value.

CYBER or Advanced Mode

The CYBER or advanced mode can be selected by writing an escape sequence from a user program. In this mode, the terminal has the screen characteristics described in appendix E. In order to use the full display characteristics of the terminal CYBER mode, send an escape sequence to the terminal as follows:

ESC, ESC, 'b', 'b'

This output sequence sets the terminal program in the CYBER mode of operation so that the keyboard and terminal characteristics are as defined in appendix E. Returning the terminal program to the CP/M compatibility mode is accomplished by the following sequence:

ESC, ESC, 'c', 'c'

Any user program using the full CYBER terminal characteristics should issue the initial ESCape sequence as it begins execution and the final sequence before reloading the CP/M by performing a jump to memory location 0.

CHANGING SCREEN CHARACTERISTICS

You can change screen characteristics either from the operator's console, or by character output sequences from the user program.

You can select different screen characteristics at any time while running CP/M by pressing the SETUP key. This returns you to the Parameter Selection Mode of your terminal through which you can change line size, characters per line, cursor type, background color, and other display-related parameters. After making the changes, press the F1 key to return to your program.

The following modes should not be changed while operating in CP/M.

Setup No. 1 F9 selects CYBER mode as (Large) or (Small). CP/M requires (Large) CYBER mode. The (Small) CYBER mode changes the keyboard translation so that the Carriage return key becomes a new line key and the code for TAB is changed. This is unacceptable to CP/M.

Setup No. 2 F6 selects half duplex or full duplex transmission. You should leave this in full duplex. Selecting half duplex causes all keyboard characters to be displayed twice on the screen.

GRAPHICS OPTION

When the graphics option is included, the terminal can produce and use graphics as though it were a Tektronix 401X (4010/4014) terminal. The terminal then operates as an ASCII graphics terminal. It is compatible with the Tektronix 401X communications protocol, but is not completely compatible with the Tektronix 401X family. For example, the screen formats are slightly different.

The entire screen has 512 dots across each line, and 512 dots up and down each row, each of which may be on (bright) or off (dark). The graphics option uses those dots, or points, to form characters, lines, and blocks. Each point has a mathematical address: the X coordinate specifies the point's horizontal position; the Y coordinate specifies the point's vertical position.

The graphics option can be used when the terminal is connected to a remote host as in the Control Data Shared Network. The following paragraphs discuss the use of the graphics option in standalone operation under CP/M. Display operation is explained; character sizes are listed; the modes of graphics operation are defined; input devices, print options, and the bypass condition are discussed; use of the graphics option is explained; and differences between the Tektronix terminals and graphics mode are listed.

Display Operation

The display may be used to plot characters, dots, lines or rectangular blocks. The display screen has a resolution of 512 by 512 dots. All graphics data is displayed within this area. Received data coordinates may be either scaled or unscaled.

Display Coordinate Scaling

The escape sequence ESC < from CP/M selects scaled coordinates with Y bias. Escape sequence ESC = selects scaled coordinates without Y bias. Escape sequence ESC > from CP/M selects unscaled coordinates. The operator can select scaled or unscaled coordinates as an operator parameter.

When operating with scaled coordinates without Y bias, the display screen is addressed as shown in figure 13-1 for 10-bit host coordinates, or figure 13-2 for 12-bit host coordinates. X coordinates run horizontally in the range of 0 through 1023 or 0 through 4095, respectively. Y coordinates are similar in the vertical direction.

X = 0, Y = 1023	X = 1023, Y = 1023
X = 0, Y = 0	X = 1023, Y = 0

Figure 13-1. Display with 10-Bit Scaled Coordinates (No Y Bias)

X = 0, Y = 4095	X = 4095, Y = 4095
X = 0, Y = 0	X = 4095, Y = 0

Figure 13-2. Display with 12-Bit Scaled Coordinates (No Y Bias)

When operating with scaled coordinates with Y bias, the display screen is addressed as shown in figure 13-3 for 10-bit host coordinates, or figure 13-4 for 12-bit host coordinates. X coordinates run horizontally in the range of 0 through 1023 or 0 through 4095, respectively. Y coordinates are similar in the vertical direction, except that they are biased upward on the screen by 122 pixels. This creates two logically separate areas on the screen. Unpredictable results may occur if these areas are not treated separately and independently.

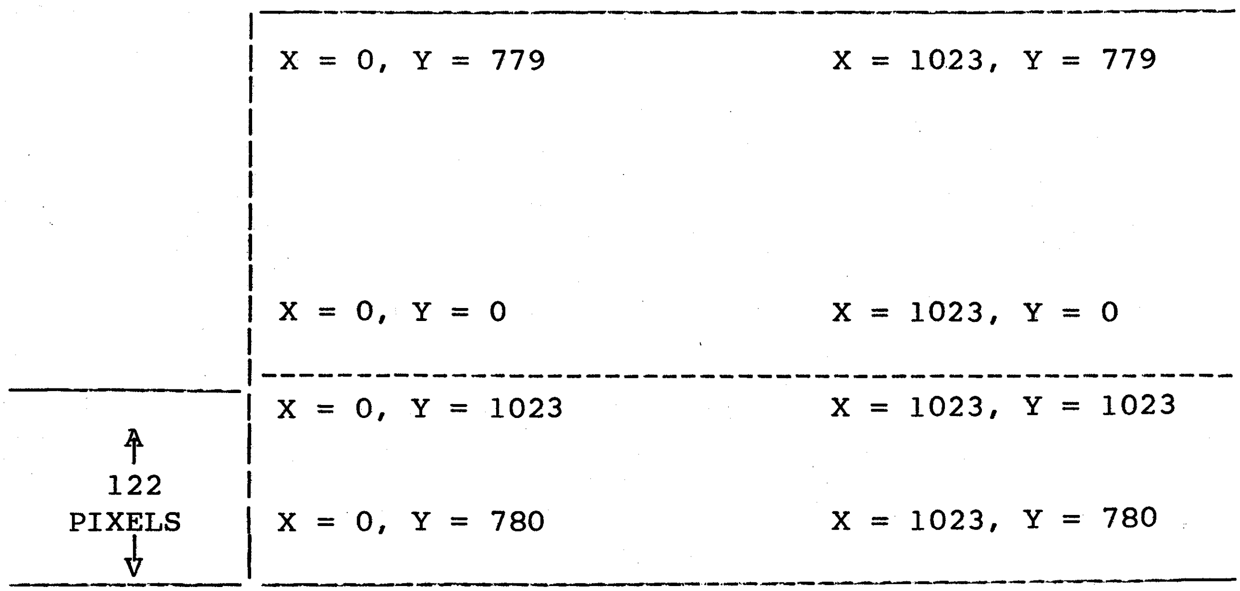


Figure 13-3. Display with 10-Bit Scaled Coordinates (With Y Bias)

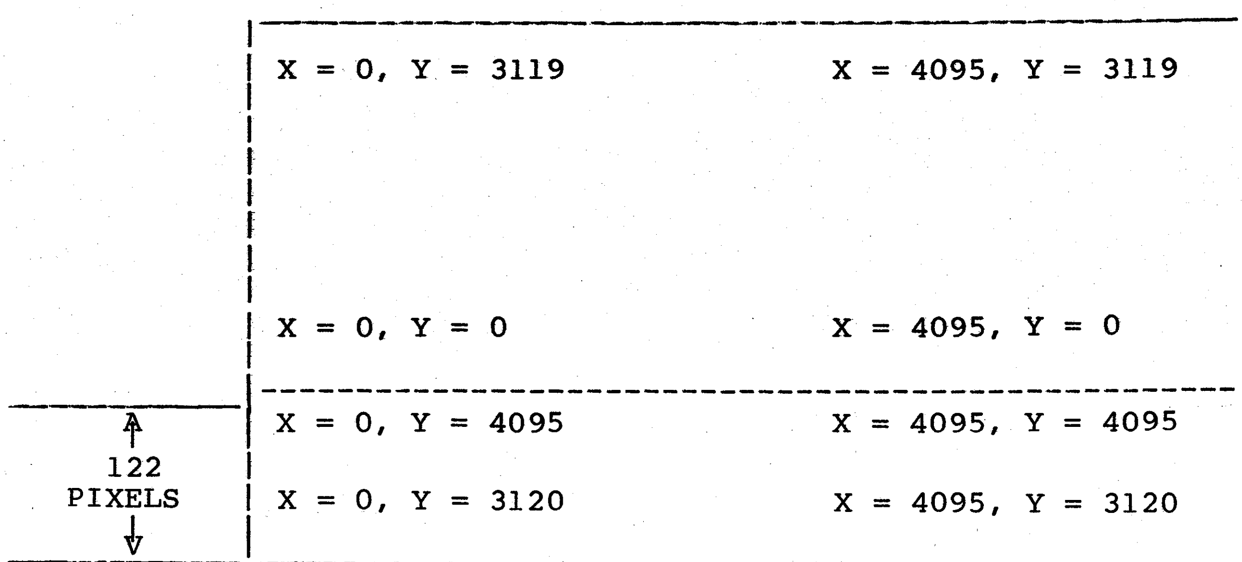


Figure 13-4. Display with 12-Bit Scaled Coordinates (With Y Bias)

When operating with unscaled coordinates, the display screen is addressed as shown in figure 13-5 for 10-bit host coordinates, or figure 13-6 for 12-bit host coordinates. X coordinates run horizontally in the range of 0 through 511 or 0 through 2047, respectively. Y coordinates are similar in the vertical direction. Unpredictable results may occur if host coordinates exceed these defined ranges.

X = 0, Y = 511	X = 511, Y = 511
X = 0, Y = 0	X = 511, Y = 0

Figure 13-5. Display with 10-Bit Unscaled Coordinates

X = 0, Y = 2047	X = 2047, Y = 2047
X = 0, Y = 0	X = 2047, Y = 0

Figure 13-6. Display with 12-Bit Unscaled Coordinates

Character Sizes

There are five character sizes - the standard PLATO characters plus four character sizes which agree with the four sizes available on the Tektronix 4014 terminal.

Character Size 0

Size 0 is the standard PLATO character with a most characters built on a 7-by-9-dot matrix and with no character larger than 8-by-16 dots. The screen format for size 0 characters has 32 lines of 64 characters each. A Tektronix-emulating format of 24 lines of 64 characters each may also be used.

Character Size 1

Size 1 characters are built on a 5-by-7-dot matrix with no character larger than 7-by-10 dots. The screen format for size 1 characters has 51 lines of 73 characters each. A format of 39 lines of 73 characters each may also be used and this corresponds to the Tektronix screen format of 35 74-character lines.

Character Size 2

Size 2 characters are built on a 5-by-7-dot matrix with no character larger than 6-by-9 dots. The screen format has 56 lines of 85 characters each. A format of 43 lines of 85 characters each may also be used and this corresponds to the Tektronix screen format of 38 81-character lines.

Character Size 3

Size 3 characters are built on a 3-by-5-dot matrix with no character larger than 4-by-7 dots. The screen format has 73 lines of 128 characters each. A format of 55 lines of 128 characters each may also be used; this corresponds to the Tektronix screen format of 58 121-character lines.

Character Size 4

Size 4 characters are built on a 3-by-5-dot matrix with no character larger than 4-by-6 dots. The screen format has 85 lines of 128 characters each. A format of 65 lines of 128 characters each may also be used; this corresponds to the Tektronix screen format of 64 121-character lines.

GRAPHICS OPERATION MODES

When Graphics operation is selected, one of seven modes can be chosen. These modes are alpha, graph, point plot special point-plot, incremental point-plot, graphics input (GIN), and block mode. The primary use of each mode is:

- Alpha - display characters.
- Graph - draw a line between two sets of coordinates or perform random positioning.
- Point-Plot - display one point at the intersection of two coordinates.
- Special Point-Plot - varies display beam intensity.
- Incremental Point-Plot - allows simulation of an incremental digital plotter.
- GIN - displays cross-hair cursor and permits operator to input coordinates.
- Block - write or erase rectangular blocks.

Release 5 summary level 047 includes 401X enhancement firmware for the graphics mode. The graphics mode of your system is compatible with the 401X firmware.

Alpha Mode

In alpha mode, the terminal displays the 95 displayable ASCII characters. The DEL (rubout) code is ignored in alpha mode.

A blinking cursor occupies the lower dots of the character matrix and marks the position on the display screen where the next character will be entered. After the symbol is entered, the cursor moves one position to the right. When the end of a line is reached, the cursor moves to the leftmost position on the next line. The cursor is nondestructive. Cursor home position is shown in table 13-1.

TABLE 13-1. CURSOR HOME POSITION VERSUS CHARACTER SIZE

	CURSOR HOME POSITION		
CHARACTER SIZE	SCALED W/O BIAS	SCALED WITH BIAS	UNSCALED
0	X = 0, Y = 992	X = 0, Y = 748	X = 0, X = 496
1	X = 0, Y = 1004	X = 0, Y = 760	X = 0, Y = 502
2	X = 0, Y = 1006	X = 0, Y = 762	X = 0, Y = 503
3	X = 0, Y = 1010	X = 0, Y = 766	X = 0, Y = 505
4	X = 0, Y = 1012	X = 0, Y = 768	X = 0, Y = 506

Note: Coordinate values shown are for 10-bit host coordinates. Multiply values by four for 12-bit host coordinates.

Characters are written on the display screen in clear-write mode unless otherwise selected by the applications program.

Control characters and keys that can be used for transition to alpha mode are:

- CR - This control character resets the terminal to alpha mode, positions the alpha cursor to the effective margin position in the current line, and clears the bypass condition.

- ESC FF - This control character sequence resets the terminal to alpha mode, selects margin 1, positions the alpha cursor to the leftmost top line (home position), clears the display, and clears the bypass condition.

- US - This control character resets the terminal to alpha mode, leaves the alpha cursor at the last graph-mode address, and clears the bypass condition. This character is nonfunctional in GIN mode, so it cannot be used to transfer from GIN mode to alpha mode.

- Pressing the □ key or the P CLEAR key resets the terminal to alpha mode, selects margin 1, positions the alpha cursor to home position, and clears the display.

- Pressing the shifted ☐ key or the HOME key resets the terminal to alpha mode, selects margin 1, positions the alpha cursor to home position, and selects size 1 characters.

In alpha mode, the following ASCII control characters are operational. All other control characters are ignored.

- CR - This character moves the cursor to the effective margin position in the current line and clears the bypass condition. A line-feed operation is also performed in offline mode or if auto line-feed is selected.
- LF - This character moves the cursor down one line and clears the bypass condition. When the cursor reaches the bottom line, the cursor moves to the same column position in the top line.
- BS - This character moves the cursor one position to the left and clears the bypass condition. When the beginning of the line is reached, the cursor moves to the last position of the line above. When the first position of the top line is reached, the cursor moves to the last position of the bottom line.
- HT - This character moves the cursor one position to the right and clears the bypass condition. Spacing past the end of a line causes the cursor to move to the beginning of the next lower line. If the cursor is in the last position of the bottom line, it moves to the home position.
- VT - This character moves the cursor up one line and clears the bypass condition. When the top line is reached, no further cursor movement occurs.
- BEL - This character sounds the audible alarm and clears the bypass condition.

Escape sequences allow deselection of previously selected functions and transmission of status and cursor position. Valid sequences are:

- ESC DC1 - While in alpha mode, this control code sets inverse video for characters. The inverse of the character's dot pattern (represented by the ASCII code) is written into display memory.

- ESC DC2 - While in alpha mode, this control code sets character overstrike write. The dot pattern represented by the ASCII code is superimposed in display memory.

- ESC DC3 - While in alpha mode, this control code sets character overstrike erase. The dot pattern represented by the ASCII code is erased from display memory.

- ESC DC4 - While in alpha mode, this control code sets clear write. The dot pattern occupying the display position is erased and the dot pattern represented by the ASCII code is written into display memory.

- ESC CAN - This sequence selects the bypass condition to inhibit data from being processed by the display.

- ESC ENQ - This sequence causes the terminal to transmit terminal status and the X-Y address of the left end of the alpha cursor to the applications program. Bypass is not affected. If a screen-copy operation is active when this sequence is received, the terminal transmission is deferred until the copy operation terminates. The character sequence the terminal transmits is shown in figure 13-7.

Byte	Item	Data							
1	Terminal status*	P1	0	1	R	V	A	M	1
2	High bits of X address	P2	0	1	X9	X8	X7	X6	X5
3	Low bits of X address	P3	0	1	X4	X3	X2	X1	X0
4	High bits of Y address	P4	0	1	Y9	Y8	Y7	Y6	Y5
5	Low bits of Y address	P5	0	1	Y4	Y3	Y2	Y1	Y0
6	Carriage Return (CR)	P6	0	0	0	1	1	0	1

*R = 0 if graphics printer attached, ready to print, and text print not selected. Otherwise R=1.
V = 1 if graph mode selected. Otherwise V = 0.
A = 1 if alpha mode selected. Otherwise A = 0.
M = 1 if margin 2 is active. M = 0 if margin 1 is active. If M = 1, X9 must equal 1 regardless of the transmitted value.
P1 through P6 are the parity bits of bytes 1 through 6.

Figure 13-7. Response to ESC ENC Call

- ESC ! X - This three character sequence is recognized when the graphics tablet is under arming control.

- ESC _ A - This three character sequence selects the graphics tablet for continuous operation.

- ESC _ B - This three character sequence puts the graphics tablet under control of subsequent ESC ! X arming sequences.

- ESC _ C - This three character sequence deselects the graphics tablet.

- ESC _ D - This three character sequence causes the terminal to transmit graphics firmware revision. The revision is sent as four ASCII characters where #.# is the revision and CR is an ASCII carriage return.

- ESC ETB - This sequence initiates a screen copy operation. Bypass condition and selected mode are not affected. The alpha cursor is inhibited until the copy operation terminates.

- ESC FF - This sequence clears the display, clears the bypass condition, and positions the alpha cursor to the home position.

- ESC 7 - Selects size 0 characters.

- ESC 8 - Selects size 1 characters.

- ESC 9 - Selects size 2 characters.

- ESC : - Selects size 3 characters.

- ESC ; - Selects size 4 characters.

Alpha mode contains two effective margin positions, referred to as margin 1 and margin 2. Margin 1 is at the left side of the graphics display, and margin 2 is at the horizontal center. Margin 2 is effective only in alpha mode.

If margin 1 exists, margin 2 is ignored while writing characters to the screen. If two column formatting is to occur, margin 2 information must be kept to one-half or less of the maximum number of characters per line for the selected character size.

In alpha mode, a page-full condition may occur when line-feeding past the bottom line. This condition depends upon the selection in the "PGFULL" (F2) box on operator line 2.

- When OFF is selected, a page full cannot occur.

- When MARG1 is selected, a page full occurs when line-feeding past the bottom line of either margin 1 or margin 2.

- When MARG2 is selected, a page full occurs only when line-feeding past the bottom line of margin 2.

When a page-full condition occurs, the ALERT indicator on the front of the terminal lights. No further data is processed until the page full is cleared by pressing the F1 key, without the operator parameter line displayed, or by transmitting a character from the keyboard. When the page full is cleared, the ALERT indicator turns off.

Graph Mode

In graph mode, the terminal writes and erases vectors in response to ASCII code sequences. To set vector coordinate positions, the X coordinate and Y coordinate must be converted to ASCII characters as shown in the coordinate conversion chart in appendix G.

These ASCII characters are interpreted by the terminal to define a vector. The reception of the low X coordinate begins a vector operation. Appendix G shows which coordinates must be sent when coordinate values change. Any number of vectors may be drawn. Each new vector uses the previous vector endpoint as a start point. (Coordinates sent with a preceding Group Separator (GS) control character specify a new start point.) Received LF characters are ignored in graph mode.

The control character used for transition to graph mode is Group Separator (GS). It sets the terminal to graph mode and restores any previous graph-mode coordinates.

Coordinates following the GS character define a new vector start point. Therefore, no vector is written between them and previous graph-mode coordinates. This dark or unwritten vector can be written if GS is immediately followed by a BEL character. The GS character can be sent at any time to change the vector start point without writing a solid-line vector.

Escape sequences begin with an ESC control character. Valid escape sequences for graph mode are:

- ESC DC1 - Causes the terminal to erase the specified vector(s). All other characteristics of graph mode remain the same.
- ESC DC2 - Causes the terminal to write the specified vector(s). All other characteristics of graph mode remain the same.

- ESC DC3 - Causes the terminal to erase the specified vector(s). All other characteristics of graph mode remain the same.

- ESC DC4 - Causes the terminal to write the specified vector(s). All other characteristics of graph mode remain the same.

- ESC \ or ESC h or ESC p selects solid vectors.

- ESC a or ESC i or ESC q selects dotted vectors.

- ESC b or ESC j or ESC r selects dot-dashed vectors.

- ESC c or ESC k or ESC s selects short-dashed vectors.

- ESC d or ESC l or ESC x selects long-dashed vectors.

- ESC ETB - Initiates a screen-copy operation. Bypass condition and selected mode are not affected.

- ESC ENQ - Causes the terminal to transmit terminal status and graph-mode X-Y beam position. Bypass is not affected. If a screen-copy operation is active when this sequence is received, the terminal transmission is deferred until the copy operation terminates. The character sequence the terminal transmits is shown in figure 13-7.

Point-Plot Mode

In point-plot mode, a point is written or erased at the specified X,Y coordinate. Received LF characters are ignored in point-plot mode.

Point-plot mode is selected by the File Separator (FS) control character. It sets the terminal to point-plot mode and restores any previous point-plot mode coordinates.

All addresses received are then displayed as a written or erased point.

Writing or erasing is selected by one of four escape sequences:

- **ESC DC1 - Erase a point(s)**

- **ESC DC2 - Write a point(s)**

- ESC DC3 - Erase a point(s)
- ESC DC4 - Write a point(s)
- ESC ETB - Initiates a screen-copy operation. Bypass condition and selected mode are not affected.
- ESC ENQ - Causes the terminal to transmit terminal status and point-plot X-Y beam position. Bypass is not affected. If a screen-copy operation is active when this sequence is received, the terminal transmission is deferred until the copy operation terminates. The following character sequence the terminal transmits is shown in figure 13-7.

Special Point-Plot Mode

In special point-plot mode, the terminal varies the display beam intensity, producing a "grey scale" capability. This is only possible on Tektronix 401X series terminals. Type 3 terminals do not allow control of beam intensity, so this mode is functionally identical to point-plot mode.

Special point-plot mode is selected by the two character sequence ESC FS. This sequence establishes the most recently defined coordinates, regardless of mode, as the initial special point-plot mode coordinates.

An intensity control character must immediately follow the ESC FS selection sequence, and must also precede each successive point address. The intensity character must be in the range of 20 hex to 7D hex. The character is functionally ignored by the firmware.

In special point-plot mode, a point is written or erased at the specified X, Y coordinate. The terminal returns the last high Y, extra byte, low Y, high X, and extra byte address set by the mode when the terminal is switched to another mode. When entering special point plot mode, only low X must be received to reset the terminal to the previous coordinates.

Valid escape sequences for the special point-plot mode are the same as those listed for point-plot mode in the preceding pages.

Incremental Point-Plot Mode

Incremental point-plot mode is selected by the Record Separator (RS) control character.

This mode inhibits the normal X/Y addressing that occurs during the other graph modes. It interprets each data byte as a command to move one step in one of eight directions, or to blank or unblank the writing beam while the beam is being moved. A step changes the internal X and/or Y coordinate by one if scaled or by two if unscaled.

Initial coordinate conditions are those that were in effect before the incremental plot mode was selected.

The following table shows what happens when you use certain command characters while the incremental point-plot mode is active.

ASCII Character	Hexadecimal Value	Action
Space	20	Beam off
P	50	Beam on
D	44	Move up (+Y)
E	45	Move up and right (+X, +Y)
A	41	Move right (+X)
I	49	Move down and right (+X, -Y)
H	48	Move down (-Y)
J	4A	Move down and left (-X, -Y)
B	42	Move left (-X)
F	46	Move up and left (-X, +Y)

When the beam is off, positioning occurs but no writing is performed on the screen. When the beam is on, positioning occurs and writing is performed on the screen in the selected writing code, either write or erase.

Escape sequences begin with an ESC control character. The following are the valid escape sequences for incremental point-plot mode.

- ESC DC1 or ESC DC3 - Erases a point.
- ESC DC2 or ESC DC4 - Writes a point.

- ESC ENQ - Causes the terminal to transmit terminal status and X-Y beam position. Bypass is not affected. If a screen-copy operation is active when this sequence is received, the terminal transmission is deferred until the copy operation terminates. The character sequence the terminal is shown in figure 13-7.

- ESC ETB - Initiates a screen copy operation. Bypass condition and selected mode are not affected.

Graphics Input (GIN) Mode

The graphics input mode is interactive in that it involves requests for information and the operator's response to the requests. A cross-hair cursor, which appears as a blinking plus sign, is enabled in GIN mode. The intersect address of the cross-hair cursor is the position of the X and Y coordinates. The cursor is nondestructive. Cursor position can be changed using the touchpanel or the numeric keypad.

During GIN mode, an ESC ETB sequence initiates a screen-copy operation. Bypass condition and selected mode are not affected. The cross-hair cursor is inhibited until the copy operation terminates. ESC FF, ESC ENQ and ESC ETB are the only escape sequences allowed in GIN mode. LF characters received in GIN mode are ignored.

The ESC SUB sequence selects GIN mode, activates the bypass condition, enables the cross-hair cursor, and enables the touchpanel.

Both of the following methods of initiating transmission to the applications program also reset the terminal to alpha mode. Bypass is not cleared until a subsequent character or character sequence that clears bypass is received. Terminal status is not transmitted while in GIN mode.

- Character Keys other than cursor positioning keys - All character keys other than the cursor positioning keys cause transmission of the entered character and X-Y position of the cross-hair cursor. The following character sequence is transmitted.

Byte	Item	Data
1	Keyboard Key	Keyboard Character
2	High bits of X address	5 MSB X + 20 hex
3	Low bits of X address	5 LSB X + 20 hex
4	High bits of Y address	5 MSB Y + 20 hex
5	Low bits of Y address	5 LSB Y + 20 hex
6	CR	0D hex

- ESC ENQ - Causes transmission of the X-Y position of the cross-hair cursor. The transmitted character sequence is the same as shown above except that byte 1 is status.

Block Mode

In block mode, a rectangular area is written or erased. Two sets of X-Y coordinates at diagonally opposite corners of the rectangle are required to define the rectangle. After writing or erasing the rectangle, the current cursor position is set to X=X1 and Y = Y1-15 (unscaled), Y = Y1-30 (scaled), where X1-Y1 is the first of the two X-Y coordinates received. Appendix G specifies the format of the coordinates. Writing or erasing is determined by escape sequences received prior to the receipt of the coordinates.

Block Mode is selected by the End of Message (EM) control character.

Escape sequences begin with an ESC control character. Valid escape sequences for block mode are:

- ESC DC1 - Erase a rectangle.

- ESC DC2 - Write a rectangle.

- ESC DC3 - Erase a rectangle.

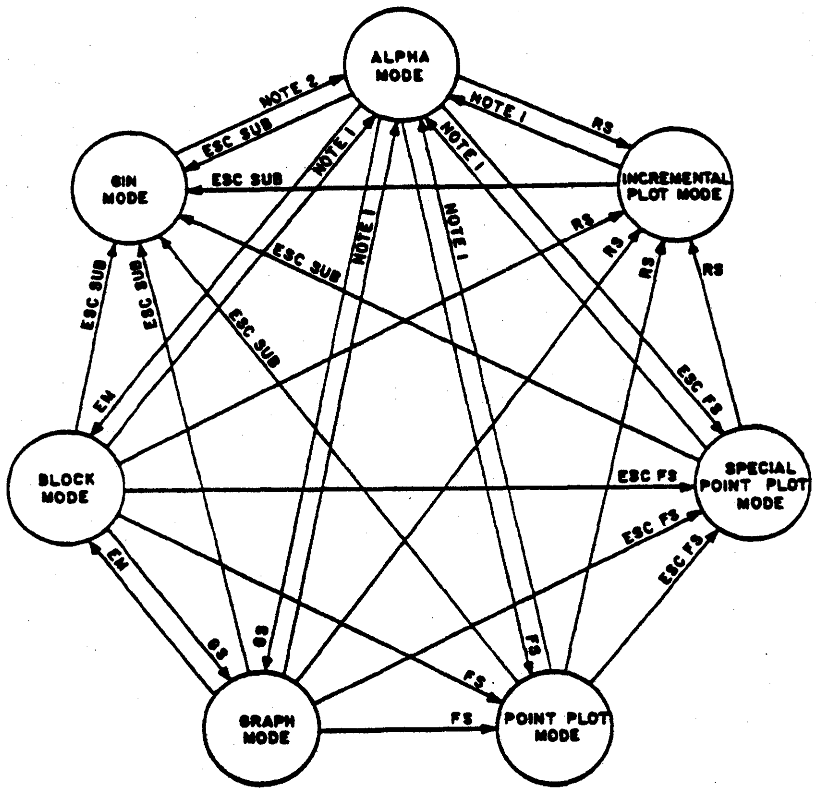
- ESC DC4 - Write a rectangle.

- ESC ETB - Initiates a screen-copy operation. Bypass operation and selected mode are not affected.

- ESC ENQ - Causes the terminal to transmit terminal status and X-Y beam position. Bypass is not affected. If a screen-copy operation is active when this sequence is received, the terminal transmission is deferred until the copy operation terminates. The character sequence the terminal transmits is shown in figure 13-7.

Mode Selection

Control codes are used to select a mode and to select certain functions within that mode. Appendix H is a summary of the ASCII operation control codes used with the graphics option. Figure 13-8 shows those control codes and sequences used to select a mode. The following paragraphs provide detailed descriptions of the operational modes and define the control codes that select functions within the modes.



NOTE 1: US, CR, ESC FF, Keyboard Page Clear, and Keyboard Home
NOTE 2: CR, ESC FF, Keyboard Page Clear, and Keyboard Home.

Figure 13-8. Permissible Operating Mode Changes

Input Devices

The following paragraphs discuss the input devices that can be used with the graphics option. These include the keyboard, the touchpanel, the numeric keypad, and the graphics tablet.

Keyboard

The tan keyboard keys are used in alpha mode. Each key displays its usual ASCII displayable symbol. The specific symbol and control codes are displayed or transmitted as in any CP/M application.

Touchpanel

The touchpanel consists of 256 touch-sensitive areas overlaying the display screen in a 16-by-16 square array. In GIN mode, touching the touchpanel moves the cursor to the center of the touched square.

Numeric Keypad

Touching the numeric keypad keys moves the cursor in GIN mode. No data is transmitted until the operator presses a non-cursor key.

- Coarse Positioning - Pressing a 1, 2, 3, 4, 6, 7, 8, or 9 key on the numeric keypad while holding down the CTRL key moves the cursor 64 dots. Table 13-2 shows the direction of movement.

- Medium Positioning - Pressing a shifted 1, 2, 3, 4, 6, 7, 8, or 9 key on the numeric keypad moves the cursor eight dots. Table 13-2 shows the direction of movement.

- Fine Positioning - Pressing an unshifted 1, 2, 3, 4, 6, 7, 8, or 9 key on the numeric keypad moves the cursor one dot. Table 13-2 shows the direction of movement.

TABLE 13-2. GIN MODE CURSOR-POSITIONING KEYS

KEYPAD KEY PRESSED	DIRECTION OF CURSOR MOVEMENT
1	Down and Left
2	Down
3	Down and Right
4	Left
6	Right
7	Up and Left
8	Up
9	Up and Right

Graphics Tablet

Graphics input operation is supported using a Summagraphics bit-pad (graphics tablet) or equivalent digitizing tablet. The stylus on the graphics-tablet device can be used to position a cross-hair cursor on the terminal screen and to send the cross-hair cursor coordinates to the applications program.

The graphics-tablet device is connected to serial port A or B using an RS-232 communications cable. The dual serial-port option must be installed in the terminal.

Graphics tablet data rate can be up to 19 200 bits per second asynchronous with continuous transmission to the terminal whenever the stylus is in close proximity to the tablet. Slower data rates may be selected if desired. Data rate selection is made with the port A or B terminal installation parameter. This parameter must match the rate selected on the graphics tablet.

Graphics Tablet Cursor Control

It is assumed that the graphics tablet is set up to provide continuous coordinate sampling and transmission while the stylus is in close proximity to the digitizing tablet. A cross-hair cursor appears on the screen when this occurs and stays on the screen until the stylus is moved away from the tablet. The cursor follows movements made by the stylus as long as the stylus is near the tablet. When a graphics-tablet cursor coordinate is sent, the audible alarm sounds and a dot is plotted at the cursor position as feedback to the operator.

The previous cursor position is automatically saved while the graphics-tablet cursor is active, and is restored when the graphics-tablet cursor disappears.

An applications program may override the graphics tablet cursor at any time without loss of position. That is, output to the terminal may occur whether or not the graphics tablet cursor is active.

The X/Y position of the graphics tablet cursor on the CRT screen may be transmitted to an applications program by pressing on the tablet stylus so that the switch in the stylus is activated. The following applicable package of data is sent each time the stylus is pressed to the tablet by the operator. This is done no matter what data mode is active in the terminal.

10-Bit Mode

	7	6	5	4	3	2	1	0	
Byte 1	P1	0	0	1	1	1	0	1	(ASCII GS)
Byte 2	P2	0	1	Y9	Y8	Y7	Y6	Y5	(High bits of Y)
Byte 3	P3	0	1	Y4	Y3	Y2	Y1	Y0	(Low bits of Y)
Byte 4	P4	0	1	X9	X8	X7	X6	X5	(High bits of X)
Byte 5	P5	0	1	X4	X3	X2	X1	X0	(Low bits of X)
Byte 6	P6	0	0	0	1	1	0	1	(ASCII CR)

12-Bit Mode

	7	6	5	4	3	2	1	0	
Byte 1	P1	0	0	1	1	1	0	1	(ASCII GS)
Byte 2	P3	0	1	Y11	Y10	Y9	Y8	Y7	(High bits of Y)
Byte 3	P2	0	1	0	Y1	Y0	X1	X0	(Extra X/Y bits)
Byte 4	P4	0	1	Y6	Y5	Y4	Y3	Y2	(Low bits of Y)
Byte 5	P5	0	1	X11	X10	X9	X8	X7	(High bits of X)
Byte 6	P6	0	1	X6	X5	X4	X3	X2	(Low bits of X)
Byte 7	P7	0	0	0	1	1	0	1	(ASCII CR)

The X and Y coordinates are the screen coordinates of the graphics tablet cursor (scaled or unscaled).

NOTE

The graphics tablet sends the Y coordinate first. All other graphics option input devices send the X coordinate first.

Print Options

The Type 1 and Type 4 printers are supported by this option. Printing may be in a screen-copy mode or in a text-print mode. Both print modes can be active at the same time with screen copy overriding text print.

The baud rate, number of data bits, and parity of the terminal's serial printer port and the Type 4 printer must match. The buffer status on the Type 4 printer must be set to X on/X off and the printer error status must be set to DTR.

Screen-Copy Mode

In screen-copy mode the printer makes a dot-for-dot reproduction of information on the terminal display screen. The screen-print controlware is available for use at any time during graphics operation.

The operator makes a screen copy by pressing the PRINT key on the terminal keyboard. Prior to this the printer must have been made ready for operation by pressing the PRINT switch on the printer so that the indicator in that switch is lit. When the copy is complete, the printer form-feeds. The printer is then ready to make another copy.

In graphics operation, the applications program may initiate the screen copy with an ESC ETB character sequence. Operation of the screen copy is the same as if the operator had started the operation.

Nominal time for a screen copy to be made is 35 seconds. High-density (dark) images on paper may slow the printer by one-third to one-half its rated speed. This is to prevent the printer solenoids from overheating. The maximum print time for a totally dark image is about 120 seconds. The fastest possible screen-copy time is about 4 seconds.

Screen-copy time depends on the number of horizontal rows of the screen which have dots turned on. If a row has one or more dots turned on, that row will have a print time of 65 milliseconds. If no dots are turned on in a row, that row will be printed in about 5 milliseconds.

Text-Print Mode

In text-print mode, the printer prints alphanumeric data as it is displayed on the screen. The following paragraphs explain selecting and deselecting text-print mode.

The operator selects text-print mode by pressing F3 while the operator parameter line is selected. While the mode is active, ASCII characters received from the applications program (and also those generated from the keyboard if in half-duplex) are sent to the printer for printing and are displayed on the screen.

Since the printer is a line printer, rather than a character printer, actual printing of a line of characters does not occur until the printer receives a CR or LF character. Once a line of characters is printed, the printer is advanced to the next line, so that overstrike printing of characters is not allowed. Lines may be printed at 6 or 8 lines to the inch as determined by a switch setting in the printer.

The operator deselects text-print mode by pressing SETUP, then the F3 key. Terminal operation may then proceed as if no printer were present. If an error condition is present in the printer, the mode may be terminated by pressing the STOP key.

Bypass Condition

The bypass condition allows data to be transmitted to the computer without it being written on the screen. The following program commands set and clear the bypass condition.

- Set bypass

  ESC CAN
  ESC SUB

- Clear Bypass

BEL	ESC BEL
BS	ESC BS
CR	
HT	ESC HT
LF	
VT	ESC VT
	ESC FF
US	ESC US

Using the Graphics Option

The option requires the file ALTPERS.COM. Do not rename this file or delete it from your system. While the graphics option is operating under CP/M control all, normal CDC 110 CP/M operations are active, except as required for the input devices. (Serial port B (J2) is reserved for use with the bit pad.) The following paragraphs discuss initializing the graphics, initializing, stopping and starting the graphics, and reinitializing to the regular modes. The initializing functions are executed either under CP/M control or under application program control, while the stopping and starting can only be done under application program control.

NOTE

> Once the graphics personality is turned on, any output not intended as graphics will have unpredictable results. Most text however, will be displayed.

Initializing

Initialization of the graphics is performed before any graphic display codes are sent to the terminal. Initialization turns off the current mode, erases the screen, sets the terminal to display the graphic memory, and causes the terminal to act upon all subsequent 7-bit display codes as a Tektronix 401X terminal would. The same events occur if you initialize while already using the graphics option. Initialization is performed either with CP/M commands or under application program control. The graphics option may also be initialized as default.

The CP/M command "GRAPHICS ON" initializes the graphics. This command is used when accessing a program that does not contain the initialization sequences for accessing the graphics option.

Initialization by an application program is done by a set of codes sent to the display. These codes each have a function in initializing the terminal.

- The initialize code (80 hex) should always be sent first. This code sets up the terminal to act as a Tektronix 401X terminal and erases the screen.

- The return communications code (98 hex) causes the communications line input to be accessable through the CP/M calls described in this section. Without this code all communications line input will be lost.

- The return rigid-disk interrupts code (9B hex) is required if graphics and rigid-disk operations are to be running at the same time.

- The get printer input code (A3 hex) causes all input from the printer to be given to the regular personality. Without this code the Xon/Xoff feature of the printer will not work.

The graphics personality can be initialized as default on terminal load. Two CP/M commands control this initialization process. These commands do not affect, in any way, the GRAPHICS ON and GRAPHICS OFF commands.

- GRAPHICS ON DEFAULT causes all subsequent terminal loads to be immediately followed by a GRAPHICS ON command. The graphics are not turned on at the time of the command but on the next load.

- GRAPHICS OFF DEFAULT causes any previous GRAPHICS ON DEFAULT to be nullified. If turned on the graphics will remain on, but any subsequent terminal loads will not be followed by the GRAPHICS ON command.

Initial Conditions

The following initial conditions exist after the graphics option has been loaded and initialized:

- The screen is erased.

- The terminal is in alpha mode.

- The alpha cursor is in the home position (upper-left corner).

- Size 1 characters will be displayed.

- Character writing is in clear-write mode.

- Margin 1 is selected.

- The terminal is set to online or offline, to half- or full-duplex, as determined by currently selected operator parameters.

- Coordinate scaling is on.

Stopping and Starting the Graphics

While using the graphics option, it is sometimes necessary to switch temporarily back to the regular personalities without initializing the screen (for example, to use the escape functions). This can be done by switching off the graphics and then switching on the graphics.

The switch-graphics-off function causes the graphics option to be turned off temporarily. While the graphics are switched off the display still shows whatever was last generated by the graphics. Only such things as escape sequences are active. Switching off is done by outputting 92 hex to the display.

The switch-graphics-on function switches graphics back on after it has been switched off. If the application program trys to switch on without having previously switched off nothing will happen. Switching on is done by outputting a 91 hex to the display.

Reinitializing

Reinitializing causes the screen to erase, character graphics to be enabled, and all inputs and outputs to be restored to their normal status. It restores the mode that was active before the graphics option was entered. Unlike the stop graphics code, once reinitialization is done the graphics must be initialized again in order to be used. It cannot be just started. Reinitialization without graphics enabled erases the screen. Reinitialization can be done either by a CP/M command or under application program control.

The CP/M command GRAPHICS OFF reinitializes to the regular mode. This command is used when the application program does not contain the reinitialization code.

Reinitialization to the regular personality is done by the application sending a reinitialize code (90 hex) to the display.

Graphics Operation Codes

In summary, application programs control the operation of the graphics by sending the following display codes:

- 80 hex - Initialize graphics.
- 98 hex - Return communications input to be handled by the regular personalities.
- 9B hex - Return fixed disk interrupts to be handled by the regular personalities.
- A3 hex - Return printer input to be handled by the regular personalities.
- 92 hex - Stop graphics without initialization.
- 91 hex - Restart graphics without initialiation.
- 90 hex - Reinitialize the mode active before graphics was initialized.

Graphics under Communications Line Control

Graphics under communications line control eliminates all CP/M control of the graphics. This feature allows you to use the fast load from the disk to load the graphics for use with the an outside system. This feature is best used by those who own a 110 system but use graphic applications on large systems, such as the Control Data Shared Network. (A load from the shared network takes much longer than one from disk.)

The GRAPHICS STANDALONE CP/M command or the display code 81 hex initialize graphics under communications line control. To return to CP/M, reset the system.

DIFFERENCES BETWEEN TEKTRONIX TERMINALS AND GRAPHICS MODE

The following differences exist between the Graphics mode of this terminal and that of the Tektronix terminals.

a) The screen resolution of this terminal is 512-by-512 dots, while that of the Tektronix terminals is 780-by-1024 dots. Coordinates received by the terminal are scaled down by a factor of 2 before being used for display. Y coordinates are increased by a value of 122 after scaling but before being used by the display screen.

b) The cross-hair cursor in GIN mode on this terminal is displayed as a + sign, rather than as vertical and horizontal lines that extend the full height and width of the screen. Since there are no thumbwheel controls on this terminal, the method of manually positioning the cross-hair is also different. It is accomplished by using the touchpanel and arrow keys on the keyboard.

c) Character sizes relative to screen size are different from the Tektronix terminals. Five character sizes are supported, but it is recommended that sizes 3 and 4 not be used in future applications since they are not very legible.

d) Alternate character set is not supported.

e) Intensity control of the graphics display beam is not provided, due to hardware limitations.

f) Automatic characters after transmission of status or cursor position can be selected via hardware jumpers on the Tektronix 401X series of terminals. The CR-only option is supported by this option.

TERMINAL TOUCHPANEL

The following description of touchpanel operation applies to CP/M implementation on your terminal if it has the touchpanel option installed. Note that this differs from the operation of the touchpanel when the terminal is used as a standalone terminal or with another computer.

The coordinates of each touch are converted into an X and a Y value. The origin of the touchpanel is the lower-left corner. X values increase left to right with values from 20 hex to 2F hex. Y values increase from bottom to top with values from 30 hex to 3F hex.

The values for each touch are stored in a 256-byte buffer where they wait for input requests from the computer. The values are stored in the buffer in the order X first and then Y. The contents of the touch buffer can be read by an assembly-language program. The program uses BDOS calls to read and set the IOBYTE at location 3 in memory to a 1 for the reader selection bits. Then a call to BDOS Reader input returns the values in the touch buffer. If the buffer is empty, a 0 is returned; otherwise, the X and then the Y value are returned.

An example program, TOUCHTST, is provided on the flexible disk. This program provides an example of accessing the touchpanel from assembly code. The source is provided in file TOUCHTST.ASM, while the assembled code is in TOUCHTST.COM. Examine the source listing and run the program to see how this interface is used. Note that TOUCHTST is only an example program, it is neither documented nor supported by CDC.

PRINTER CHARACTERISTICS

The TERMINAL.COM program operating in the terminal contains the printer drivers for CP/M 2.2. Print characters are transferred from the CP/M 2.2 program in the disk drive to the terminal. The current drivers support the Type 1 printer on the parallel channel and serial printers, such as the Type 2 printer on the RS-232 channel. Serial RS-232 interface printers are plugged into the J1 connector of the dual-port option board on the terminal.

As delivered, the system is set to use either type of printer. Selection of the printer is automatic. If the Type 1 printer is connected and has power on, it is selected. If there is no Type 1 printer connected, or it has power off, the serial printer is selected.

Refer to the 110 Installation Guide for printer installation information (see preface for publication number).

COMMUNICATIONS ON THE CONTROL DATA 110 VIKING SYSTEM

The Control Data 110 CP/M system provides communications driver support in the Basic Input/Output System code for asynchronous-only communications over the DB-25 connectors in the terminal. Your terminal can support four channels of communications. The communications channels are referred to as Comm 0, Comm 1, Comm 2, and Comm 3. The connector names associated with the communications channels are:

Channel	Use	Type 3
Comm 0	HOST	Host
Comm 1	PRINTER	Dual port A
Comm 2		Dual port B
Comm 3	Internal Modem	Internal Modem

The Comm 1 channel is normally assigned to a RS-232 interface printer and is driven for output only as the LIST device. If you do not use a RS-232 interface printer on your system then the Comm 1 channel may be used for communications.

Any writer of communications programs for the Control Data 110 should note that this is a multi-processor system. CP/M programs and memory reside in the memory of the flexible-disk controller, while the communications driver routines reside in the terminal with a separate Z-80 CPU. Data received from a communications channel is stored in a 256-byte input buffer in the terminal. Received characters are then transferred to an 8-byte input buffer in the flexible disk unit memory where they are processed as input. Output from the flexible disk unit memory goes to an 8-byte buffer and then is automatically transferred to a 32-byte output buffer in the terminal before being transmitted on the communications channel.

Communications input and output takes place over the CONIN, CONOUT, and CONST in BIOS or BDOS calls. The user program must set the IO byte in location 3 to 2 for the console redirection. This causes all subsequent calls to CONIN, CONOUT, and CONST to receive data, send data, and check input status of the communications lines.

Selection of the comm channel is determined by the COMM byte at location 0E (hex) with the values:

COMBYTE Value	Comm Channel To Use
0	Comm 0
1	Comm 1
2	Comm 2
3	Comm 3

The recommended programming practice is to set the COMBYTE value first, then modify the IOBYTE to access the communications devices. Next use BIOS calls to CONIN, CONOUT to transfer data and use BIOS calls to CONST to see if there is any data in the input buffers. The similar BDOS calls can be used, but note that it is then impossible to receive the character NUL or to transmit the character 0FF(hex).

CUSTOMIZING THE CONTROL DATA 110 SYSTEM

The Control Data 110 CP/M 2.2 system provides for customization of terminal characteristics, printer characteristics, and communications line characteristics. Terminal characteristics are items such as number of lines per page. You can change background color, cursor type, and scrolling mode from the keyboard of the terminal. In addition, the codes of most of the white keys can be altered to meet special needs. Printer characteristics are the type of printer (Type 1 or serial printer, such as the Type 2, 3, or 4 printer) and line size. For the serial printer you can change transmission speed, byte size, parity checking, and feedback method. The serial communications port parameters are baud rate, byte size, and parity.

This customization information is contained in the TERMINAL program TERMINAL.COM. Remember that TERMINAL.COM is essential to your system and must not be deleted or renamed.

You can change the default values in TERMINAL.COM to customize your system in two ways: using the program TERMSET, or using ESCape sequences. (You must use TERMSET to change printer characteristics.) The following paragraphs discuss customizing with TERMSET including use of the SETUP and BREAK keys, enabling of CTRL C or CTRL P options, and configuration of the Type 4 serial graphics printer into the system.

Paragraphs describing dynamic alteration appear after the TERMSET information.

CUSTOMIZING WITH TERMSET

The TERMSET program allows you to set the customization parameters for your terminal. A change may be temporary, or the file TERMINAL.COM may be permanently changed so the new parameters will be used the next time the Control Data 110 System is loaded. To use TERMSET, reply to the CP/M prompt by typing:

TERMSET <next>

The program is self-prompting. It asks for your terminal type and then extracts the current parameters from the TERMINAL.COM file and displays them as shown below. If you select a parameter value out of the given range, TERMSET gives you an error indication. If you get an error indication, you must select a value in the legal range or else the illegal values will be stored in the TERMINAL.COM file when you update it.

The TERMSET program changes any parameters in a temporary storage area. You have the option of updating the TERMINAL.COM file for permanent changes. You may load the revised values into the terminal without making permanent changes. This allows you to test the parameter changes without committing to a permanent change in the TERMINAL.COM file. You may exit TERMSET without affecting the terminal or the TERMINAL.COM file.

The initial display from TERMSET is shown below. You must type 3 since your terminal is Type 3.

```
CONTROL DATA 110

TERMINAL customization program.
Version 3.04

This program will allow you to create a new copy of the TERMINAL program
specialized to your particular equipment.
Values given in () are the delivered parameters for the TERMINAL program.

You may make temporary changes or permanent changes in the terminal program.
The following special prompt characters have the following action:
1       Causes the current modified values to be displayed.
2       Will abort the program with no changes.
3       Will download the current changed values to the terminal
        to test the changes.
4       Will record a new file with the permanent changes.

What type terminal do you have?  (1,2,or 3).
If your terminal has a separate keyboard, you have a type 3 terminal.
```

You will see the following display of parameters which can be changed. Parentheses enclose the default values.

```
Item  Value   Description
  e     0     Printer Type      (0)=PG/S, 1=PG, 2=S, 3=None, 4=PG/SG, 5=SG
                                PG/SG=Parallel/Serial Graphic, S=ASCII serial
  f     3     Printer baud      0=150, 1=300, 2=600, (3)=1200, 4=2400, 5=4800,
                                6=9600, 7=19200
  g     1     Printer byte size         0=7 bits, (1)=8 bits
  h     0     Printer parity    (0)=None, 1=Odd, 2=None, 3=Even
  i     0     Printer feedback method   (0)=DSR, 1=X on,X off, 2=Both
  j     0     Printer width+1   Values from 1 to 255
  k    50     Printer width+1 div 256
  l     1     Comm. baud rate   0=150, (1)=300, 2=600, 3=1200, 4=2400, 5=4800,
                                6=9600, 7=19200
  m     1     Comm. byte size   0=7bit, (1)=8 bit
  n     0     Comm. parity      (0)=None, 1=Odd, 2=None, 3=Even
  o     0     Operator BREAK key        (0)=Enable, 1=Disable
  q    30     Lines on display  (30) May have 3 to 30 lines
  r     0     Operator SETUP key        (0)=Enable, 1=Disable
  s     3     Comm 2 baud       0=150, 1=300, 2=600, (3)=1200, 4=2400, 5=4800,
                                6=9600, 7=19200
  t     1     Comm 2 byte size  0=7 bits, (1)=8 bits
  u     0     Comm 2 parity     (0)=None, 1=Odd, 2=None, 3=Even
  v     3     Comm 3 baud       0=150, 1=300, 2=600, (3)=1200, 4=2400, 5=4800,
                                6=9600, 7=19200
  w     1     Comm 3 byte size  0=7 bits, (1)=8 bits
  x     0     Comm 3 parity     (0)=None, 1=Odd, 2=None, 3=Even
  y           Display/change Function keys
  z           Display/change White keys
  {     0     Enable CTRL C/CTRL P      (0)=Both, 1=CTRL P, 2=CTRL C, 3=None
Enter item to change (alpha value under 'Item') or the following command
1=display, 2=abort, 3=temp, 4=permanent
```

To modify a function key in all operating states (lowercase, uppercase, lowercase with CTRL key pressed, and uppercase with CTRL key pressed), enter y. The following appears.

```
The function key values are:

KEY     HEX 1    HEX 2    HEX 3    Ascii 1 Ascii 2 Ascii 3

 F1      02       00       00      ^B
 F2      06       00       00      ^F
 F3      1B       42       00      ^[       B
 F4      1B       46       00      ^[       F
 F5      01       00       00      ^A
 F6      05       00       00      ^E
 F7      10       00       00      ^P
 F8      0E       00       00      ^N
 F9      1B       56       00      ^[       V
 F10     16       00       00      ^V
 F11     00       00       00
 F12     00       00       00

Enter function key (1..12) to change, or Q)uit ?
```

A modification to function key 11 is shown in the following example. Note that if you want less than three values generated by a function key, press the CTRL and @ keys to enter a null (00) for the unused values. Any value(s) following a null are invalid.

CAUTION

Entries of Control C or Control Z cannot be made correctly using this mode.

```
The function key values are:

KEY      HEX 1    HEX 2    HEX 3    Ascii 1 Ascii 2 Ascii 3

 F1       02       00       00       ^B
 F2       06       00       00       ^F
 F3       1B       42       00       ^[      B
 F4       1B       46       00       ^[      F
 F5       01       00       00       ^A
 F6       05       00       00       ^E
 F7       10       00       00       ^P
 F8       0E       00       00       ^N
 F9       1B       56       00       ^[      V
 F10      16       00       00       ^V
 F11      00       00       00
 F12      00       00       00

Enter function key (1..12) to change, or Q)uit ? 11
Do you want to enter hex values (Y,N)? y

Please type up to 3 hex values
Press space bar between values  01 02 00
```

To modify the white keys (one case of the function keys and the special purpose keys), enter z. The message which is then displayed asks for the hexadecimal value of the white key to be changed. These values are given in table 13-3. In this example, the codes for function key 12 in lowercase are to be changed. Note that if you want less than three values generated by a white key, press the CTRL and @ keys to enter a null (00) for the unused values. Any value(s) following a null are invalid.

```
WHITE KEY MODIFICATION

This portion allows you to examine and change each keycode
You must use your CDC 110 CP/M User's manual with this routine
to see which keycode matches each Keycap and shift

Do you want to use default values for keycodes (Y or N)? y

Enter hexadecimal value of white key to change, or Q)uit ? 0c

The White key value is:

KEY      HEX 1    HEX 2    HEX 3    Ascii 1 Ascii 2 Ascii 3

 0C       00       00       00

Do you want to change this value (Y,N)? y
Do you want to enter hex values (Y,N)? n

Please enter three keystrokes for the White key value
use <CONTROL @> for null value. <CONTROL C> will quit program.   F12
The White key value is:

KEY      HEX 1    HEX 2    HEX 3    Ascii 1 Ascii 2 Ascii 3

 0C       46       31       32       F       1       2


Enter hexadecimal value of white key to change, or Q)uit ?
```

To make any other changes, type the letter of the line you want to change. Then type the number of the new value you want to use. Type <next>.

If you wish to make permanent changes in the TERMINAL.COM program, type 4 in response to the "Enter item to change" prompt. Then the TERMSET display will look like this:

```
Item  Value   Description
  e    0      Printer Type       (0)=PC/S, 1=PG, 2=S, 3=None, 4=PG/SG, 5=SG
                                 PG/SG=Parallel/Serial Graphic, S=ASCII serial
  f    3      Printer baud       0=150, 1=300, 2=600, (3)=1200, 4=2400, 5=4800,
                                 6=9600, 7=19200
  g    1      Printer byte size        0=7 bits, (1)=8 bits
  h    0      Printer parity     (0)=None, 1=Odd, 2=None, 3=Even
  i    0      Printer feedback method   (0)=DSR, 1=X on,X off, 2=Both
  j    0      Printer width+1    Values from 1 to 255
  k   50      Printer width+1 div 256
  l    1      Comm. baud rate    0=150, (1)=300, 2=600, 3=1200, 4=2400, 5=4800,
                                 6=9600, 7=19200
  m    1      Comm. byte size    0=7bit, (1)=8 bit
  n    0      Comm. parity       (0)=None, 1=Odd, 2=None, 3=Even
  o    0      Operator BREAK key        (0)=Enable, 1=Disable
  q   30      Lines on display  (30) May have 3 to 30 lines
  r    0      Operator SETUP key        (0)=Enable, 1=Disable
  s    3      Comm 2 baud        0=150, 1=300, 2=600, (3)=1200, 4=2400, 5=4800,
                                 6=9600, 7=19200
  t    1      Comm 2 byte size   0=7 bits, (1)=8 bits
  u    0      Comm 2 parity      (0)=None, 1=Odd, 2=None, 3=Even
  v    3      Comm 3 baud        0=150, 1=300, 2=600, (3)=1200, 4=2400, 5=4800,
                                 6=9600, 7=19200
  w    1      Comm 3 byte size   0=7 bits, (1)=8 bits
  x    0      Comm 3 parity      (0)=None, 1=Odd, 2=None, 3=Even
  y           Display/change Function keys
  z           Display/change White keys
  {    0      Enable CTRL C/CTRL P      (0)=Both, 1=CTRL P, 2=CTRL C, 3=None
Enter item to change (alpha value under 'Item') or the following command
1=display, 2=abort, 3=temp, 4=permanent
```

Table 13-3 lists the hexadecimal values that identify the white keys and their operating states. The compatibility mode keycodes listed in appendix B are the default values of the white keys.

TABLE 13-3. VALUES FOR IDENTIFYING WHITE KEYS

LEGENDS ON WHITE KEYS*			HEXADECIMAL KEY VALUES				
LOWER	CENTER	UPPER	LOWERCASE	UPPERCASE	LOWERCASE WITH CTRL KEY	UPPERCASE WITH CTRL KEY	ALL OPERATING STATES
	(F1)		1	21	41	61	81
	(F2)		2	22	42	62	82
	(F3)		3	23	43	63	83
	(F4)		4	24	44	64	84
	(F5)		5	25	45	65	85
	(F6)		6	26	46	66	86
	(F7)		7	27	47	67	87
	⇐ (F8)		8	28	48	68	88
	SUPER (F9)		9	29	49	69	89
	SUB (F10)		A	2A	4A	6A	8A
	MICRO (F11)		B	2B	4B	6B	8B
	FONT (F12)		C	2C	4C	6C	8C
ANS	(F13)	TERM	D	2D	4D	6D	8D
	COPY (F14)		E	2E	4E	6E	8E
	□ (F15)		F	2F	4F	6F	8F
	+	DOWN***	10	30	50	70	90
	-	UP***	11	31	51	71	91

TABLE 13-3. VALUES FOR IDENTIFYING WHITE KEYS (CONTD)

LEGENDS ON WHITE KEYS*			HEXADECIMAL KEY VALUES				
LOWER	CENTER	UPPER	LOWERCASE	UPPERCASE	LOWERCASE WITH CTRL KEY	UPPERCASE WITH CTRL KEY	ALL OPERATING STATES.
	X	FWD***	12	32	52	72	92
	÷	BKW***	13	33	53	73	93
	HELP		14	34	54**	74**	94
	ERASE		15**	35	55**	75	95
	EDIT		16	36	56**	76**	96
	BACK		17	37	57**	77**	97
	DATA		18	38	58	78**	98
	LAB		19	39	59**	79**	99
	STOP		1A	3A	5A**	7A**	9A
	PRINT		1B	3B	5B**	7B**	9B
	⟝		1C	3C**	5C	7C**	9C

*White keys with codes that cannot be altered are not listed in table.

**No code is generated for this operating state of the key.

***Labeled on skirt of keycap.

SETUP and BREAK Keys

The TERMSET program makes both the SETUP and BREAK keys available to you. TERMSET provides key control as follows:

SETUP key	0-Enable 1-Disable
BREAK key	0-Enable 1-Disable

When you enable the SETUP or BREAK key, the TERMINAL.COM passes the raw codes "ID" and "18" respectively to the terminal resident firmware for processing. If you disable the key option(s), the TERMINAL.COM discards the raw codes.

Enabling CTRL C or CTRL P Option

The TERMSET program allows you to enable both the Control C and Control P functions of CP/M. The option is defined as follows:

```
ENABLE CTRL C/CTRL P  0 - Both (Enable CTRL C and CTRL P
                      1 - ^ P (Enable CTRL P)
                      2 - ^ C (Enable CTRL C)
                      3 - None (suppress CTRL C and CTRL P)
```

Configuring the Type 4 Serial Graphics Printer

The TERMSET program enables you to configure the Type 4 serial graphics printer into the system. The printer option of the program provides the following configurations:

```
PRINTER TYPE OPTION  0 - PG/S (Either parallel graphics or
                                ASCII serial printer)
                     1 - PG (Parallel graphics printer
                              only)
                     2 - S (ASCII serial printer only)
                     3 - None
                     4 - PG/SG (Either parallel graphics
                                 or serial graphics printer)
                     5 - SG (Serial graphics printer only)
```

If the Type 4 serial graphics printer is configured into the system, a code sequence ESC 4 is sent to the printer to disable the graphics mode when you cold boot the CP/M or press the PRINT key. This feature allows the printer to operate in the ASCII mode during normal CP/M operations.

DYNAMIC ALTERATION

The following paragraphs discuss the dynamic alteration of Control Data 110 parameters. Most parameters can be altered either through user programs or from the terminal keyboard. Terminal and communications characteristics can each be altered.

The terminal controlware allows a user program to change the terminal parameters by sending an escape sequence to the terminal. The following paragraphs describe the escape sequences for altering basic terminal parameters, altering codes of white keys, and dynamic alteration of communications parameters.

Altering Basic Terminal Parameters

The escape sequences for altering basic terminal parameters consist of the ASCII characters ESC, ESC, letter, value. The letter is one of the lowercase ASCII characters in the following list. The value is the binary value to be set for the terminal parameter. After the parameters have been changed, the escape sequence ESC, ESC, 'a', 'a' must be sent to the terminal to reinitialize the terminal and put the parameter changes in effect. This capability is provided in all terminal controlware of version 1.03 or higher. The escape sequences are as follows:

NOTE

Refer to the display from the program TERMSET for the meaning and permissible values for "value" in the following sequences.

ESC, ESC, 'e', value	Printer type selector.
ESC, ESC, 'f', value	Comm 1 baud rate for RS-232 interface to printer
ESC, ESC, 'g', value	Comm 1 byte size for RS-232 interface
ESC, ESC, 'h', value	Comm 1 parity for RS-232 interface
ESC, ESC, 'i', value	Printer feedback method
ESC, ESC, 'j', value	Printer width (1 - 255)
ESC, ESC, 'k', value	Printer width (Multiples of 256)
ESC, ESC, 'l', value	Comm 0 Communications baud rate
ESC, ESC, 'm', value	Comm 0 Communications byte size
ESC, ESC, 'n', value	Comm 0 Communications parity
ESC, ESC, 'o', value	Operator BREAK key

ESC, ESC, 'q', value	Number of lines on screen (3 to 30)
ESC, ESC, 'r', value	Operator SETUP key
ESC, ESC, 's', value	Comm 2 baud rate
ESC, ESC, 't', value	Comm 2 byte size
ESC, ESC, 'u', value	Comm 2 parity
ESC, ESC, 'v', value	Comm 3 baud rate
ESC, ESC, 'w', value	Comm 3 byte size
ESC, ESC, 'x', value	Comm 3 parity
ESC, ESC, '{', value	CTRL C/CTRL P
ESC, ESC, 'a', 'a'	Re-initialize terminal.

Some parameters take effect without reinitializing, but it is safer to reinitialize.

The screen may also be cleared from the keyboard. In CP/M, the terminal may be cleared by pressing the CTRL key and Z keys, then the NEXT key. In compatibility mode, the ☐ key clears the screen.

If the number of lines on the terminal is set to 29 or less, the user program can write in the additional lines up to line number 29 (lines are numbered from 0 to 29). This is done by issuing a cursor positioning command to the desired line and then outputting data. The data output must not contain a carriage return or a line feed.

A carriage return or line feed when the cursor is below the screen size causes the top of the screen to scroll up one line and the cursor to return to the bottom line specified in screen size. Also, output below the last line must not write to character number 79 in a line (characters are numbered 0 to 79). If this occurs, the same scrolling and return will occur.

Altering Codes of White Keys

A user program can alter the codes of the white function and special purpose keys. To alter the codes for the duration of the program, send the terminal the following sequence:

ESC,ESC,'y',hex key value,number of codes, code,code,code

To alter the codes until the system is reset, send the terminal the following sequence:

ESC,ESC,'y', hex key value, number of codes, code,code,code

In both sequences, the hex-key value indentifies which key is to be altered and in which operating state (lowercase, uppercase, lowercase with CTRL key pressed, uppercase with CTRL key pressed, or all four). Refer to table 13-3. The number of codes (0 through 3) indicates how many consecutive codes are to be generated by the key in the specified operating state. The code(s) at the end of the sequence (may be none or up to three) are the actual codes to be generated. The only codes which cannot be used are 00 and FF hex. The following codes cause special actions:

Hex Code	Special Action
80	Causes the next code to be converted to the Control case value (same as if the CTRL key was in use)
81	Starts a print of the contents of the screen
82	Clears the screen

Dynamic Alteration of Communications Parameters

The communications parameters can be set dynamically by outputting a series of ESC sequences to the console. This informs the terminal program of the desired communications channel charactersitics. The following are the escape sequences:

COMM 0 (host connector)

ESC, ESC, 'l', value	Set baud rate
ESC, ESC, 'm', value	Set byte size
ESC, ESC, 'n', value	Set parity

COMM 1 (printer connector)

ESC, ESC, 'f', value	Set baud rate
ESC, ESC, 'g', value	Set byte size
ESC, ESC, 'h', value	Set parity

COMM 2 (Dual port B connector)

ESC, ESC, 's', value	Set baud rate
ESC, ESC, 't', value	Set byte size
ESC, ESC, 'u', value	Set parity

COMM 3 (Internal Modem)

ESC, ESC, 'u', value	Set baud rate
ESC, ESC, 'w', value	Set byte size
ESC, ESC, 'x', value	Set parity

After sending the comm parameters you must send the sequence:

ESC, ESC, 'a', 'a'

Parameters which were previously sent are used to initialize the communications circuitry. Values sent are binary:

BAUD RATE FOR ALL COMMUNICATIONS CHANNELS

VALUE	BAUD RATE
0	150
1	300
2	600
3	1200
4	2400
5	4800
6	9600
7	19200

BYTE SIZE (ALL CHANNELS)

VALUE	BYTE SIZE
0	7 bit byte
1	8 bit byte

PARITY (ALL CHANNELS)

VALUE	PARITY GENERATED AND CHECKED
0	None
1	Odd
2	None
3	Even

The ESC sequences are defined as they would be written in an assembly program. The following code would initialize comm 0 channel to 300 baud, 7 bit byte, even parity.

```
          org     100h              ;Start of transient program area
          jmp     start             ;pass initial setup

esc:      equ     01Bh              ;define ESC character
iobyte:   equ     3                 ;location of IO byte
combyt:   equ     0Eh               ;location of Comm byte

setup:
          db      esc,esc,'l',1     ;300 Baud
          db      esc,esc,'m',0     ;7 bit byte
          db      esc,esc,'n',3     ;even parity
          db      esc,esc,'a','a'
          db      '$'               ;stopper for output

print:                              ;print string at HL stopping on
                                    '$' character
          mov     a,m               ;get character
          cpi     '$'
          rz                        ;return if stopper
          inx     h                 ;increment for next character
          push    h
          mov     c,a               ;character to C for BIOS call
          call    conout
          pop     h
          jmp     print

conout:           ;routine to make bios call to CONOUT
          lhld    1                 ;get base of Bios Vector
          mvi     l,0Ch             ;offset to CONOUT
          pchl    l                 ;jump to BIOS routine

start:            ;start of setup code
          mvi     a,0
          sta     iobyte            ;set iobyte to standard output
          sta     combyt            ;set comm byte to channel 0
          lxi     h,setup
          call    print
          jmp     0                 ;exit routine with comm setup.
```

The program can be assembled using the CP/M ASM assembler.

FURTHER COMMUNICATIONS PROGRAMMING INFORMATION

The following paragraphs provide additional information for communications programmers. Modem status and control signals, programming a break, connector configurations, and communications programming hints are discussed.

MODEM STATUS SIGNALS

The user program can detect certain modem status signals appearing on the RS-232 connector by doing an input from the Reader device in the BIOS with the IO byte for the Reader device set to 0.

The status value returned on the terminal is dependent on the channel used and is as follows:

Byte returned to user program.

```
Bit     7 6 5 4 3 2 1 0            Comm 0              Comm 1, Comm 2
Number  x   x   x x x x  From     Host Port              Dual Port
        |   |   | | | |
        |   |   | | | |__________DSR                    DTR
        |   |   | | |____________Ring Indicator         Ring Indicator
        |   |   | |______________0 = XMT buffer empty
        |   |   |                1 = data in XMB buffer
        |   |   |________________DCD                    Secondary RTS
        |   |
        |   |____________________CTS                    RTS
        |
        |________________________BREAK                  BREAK
```

MODEM CONTROL SIGNALS

The user program can set various modem control bits by outputting a byte to the PUNCH device via BIOS with the IO byte set to 0. The Comm byte should be set to the channel desired.

The modem control signals which can be sent to the terminal ports depend on the port selected. The data byte consists of following bits:

```
Bit       7 6 5 4 3 2 1 0                Comm 0         Comm 1, Comm 2
Number    x     x   x x     Control to  Host port         Dual Port
          |     |   | |
          |     |   | |________________ RTS              CTS
          |     |   |__________________ Sec. RTS         Carrier on
          |     |______________________ Break            Break
          |____________________________ DTR              DSR
```

PROGRAMMING A BREAK

If the user program sets the break bit in the control byte, it is up to the user program to leave the break bit set to 1 for the time necessary to generate a break condition on the communication line. The user program must then send a control byte with the break bit set to 0 to remove the break condition.

CONNECTOR CONFIGURATIONS

The following gives the pin location and signal directions for the DB-25 connectors on the Type 3 terminal.

HOST PORT (COMM 0)

Signal	Pin	Signal direction	
XMT	2	-------->	From terminal to device.
RCV	3	<--------	To terminal from device.
RTS	4	-------->	
CTS	5	<--------	
DSR	6	<--------	
DCD	8	<--------	
DTR	20	-------->	

DUAL ASYNCHRONOUS PORTS (COMM 1/COMM 2)

Signal	Pin	Signal direction	
XMT	2	<--------	From device to terminal.
RCV	3	-------->	To device from terminal.
RTS	4	<--------	
CTS	5	-------->	
DSR	6	-------->	
DCD	8	-------->	
DTR	20	<--------	

COMMUNICATIONS PROGRAMMING HINTS

To access the communications channels you must set two bytes in lower memory to provide selection of communications and the desired channel. Thereafter, you can use BIOS calls to transmit data, to receive data, to check if there is receive data available, and to check if there is room in the transmit buffer. The information can be summarized:

Memory Location	Name	Function
03 (hex)	IOBYTE	Select alternate device for BIOS calls Set this byte to 0C2 (hex) for communications.
0E (hex)	COMBYT	Selects communications channel 0 = host port 1 = printer port 2 = Dual port B 3 = Internal modem
BIOS call	vector offset	function when selected for communications
CONST	06 (hex)	Return A reg = 0 if no char ready FFh = character ready to read
CONIN	09 (hex)	Read character to A register
CONOUT	0C (hex)	Transmit character in C register
PUNCH	12 (hex)	Set Modem control from C
READER	15 (hex)	Return Modem status in A register
LISTST	2D (hex)	Return A reg = 0 output buffer full FFh = space for another output

AUTOMATIC FLEXIBLE-DISK DENSITY SENSING

The flexible-disk drives are numbered 0 for the Type 1 disk drive and 1 for the Type 2 disk drive. The current version of Control Data 110 CP/M (identified by a BIOS version of 2.00 or greater) determines the type of flexible disk inserted in the disk drive when the drive is addressed as A: or B:. If the Type 1 drive is addressed as C: or the Type 2 drive is addressed as D: then the flexible disk is assumed to be single-density, single-sided (for compatibility with earlier versions of Control Data 110 CP/M). This determination is made on execution of a select-disk command in the BIOS.

The following operating procedure should be used to insure that no errors are generated while using this feature in the Control Data 110 CP/M:

1. Whenever you replace a flexible disk in the Type 2 disk drive with a different density flexible disk, be sure to warm boot CP/M by pressing ETX (or CTRL and C) after you have replaced flexible disk.

2. If you are doing a transfer of files between two different density flexible disks using PIP, use the following procedure:

   a. Place the flexible disk that is to receive the files into the Type 2 disk drive.

   b. Type <CTRL C> to introduce this flexible disk to CP/M and allow it to be read/written. (This step is not necessary if this is the first time you have placed a flexible disk in Type 2 disk drive.)

   c. Select the Type 2 disk drive as the default flexible disk by typing B: so that display looks like:

```
A>B:
```

Then type <next>. You receive the following prompt from CP/M:

```
B>
```

Then type:

B>A:PIP <next>

This gets PIP program which prompts:

```
*
```

To transfer all files from the source disk, remove the system disk from the Type 1 disk drive and insert the source disk in the Type 1 disk drive. Then type:

B:=A:*.* <next>

All files will be transferred. Next insert the system disk in the Type 1 disk drive and type a carriage return. System operates.

3. As long as you use only double-density flexible disks, there are no special precautions required to use the system.

DATE AND TIME-CLOCK OPERATIONS

The CP/M 2.2 system has a date and time clock. Section 12 mentions using the DATTIM utility to set the clock. The program stores the date in memory as follows:

Location		Value
Decimal	Hexadecimal	
16	10	Month as two 4-bit fields
17	11	Day as two 4-bit fields
18	12	Year as two 4-bit fields
19	13	Hour as two 4-bit fields
20	14	Minute as two 4-bit fields
21	15	Second as two 4-bit fields
22	16	1/60-second counter

The CPM BIOS routine updates the 1/60-second counter on each transition of the ac line. The counter runs from 60 down to 1.

The clock routine updates the time-of-day, and advances the date at midnight. The routine uses a 24-hour clock so PM times run from 12:00 to 23:59.

All values are stored as packed BCD. For example, the value 23 is stored as hexadecimal 23 or decimal 35.

This time is accessed in Pascal/M by built-in functions. Assembly language programs can also reference time by reading the contents of the locations specified and using the time in packed BCD. Or the packed BCD can be converted to the values required.

CBASIC programs can access time by using the PEEK function and then doing the necessary conversion.

The date and time currently set for the system can be determined by executing the DATTIM program and typing <next> at the date and time prompts. The DATTIM program does not advance the month and the year.

EXAMPLE PROGRAMS

The following assembly source programs and assembled binary files are included on the CP/M 2.2 flexible disk as examples of system programs and use of the Control Data 110 system under CP/M. These programs can be assembled using ASM, loaded using LOAD, and executed by typing the name of the program at the CP/M prompt line.

The interested user can examine these programs as examples. Note, however, that these programs are neither documented nor supported by CDC.

COMLOOP0.ASM COMLOOP0.COM	An example of the method of programming communications using the host connector for connection to a remote communications device.
COMLOOP1.ASM COMLOOP1.COM	An example of the method of programming communications using the serial printer connector (when you are not using the serial printer) for connection to a remote communications device.
TERMTEST.ASM TERMTEST.COM	This demonstration program shows how to program the terminal display for the compatibility set of characteristics.
TOUCHTST.ASM TOUCHTST.COM	This demonstration program shows how to program input from the touchpanel (if included in your terminal).

THE KEYTEST.COM ROUTINE

The program KEYTEST allows you to check the operation of the keyboard in normal and numeric keypad mode. This program displays the character outputted and the hex code generated by the key. Note that some terminal keys will generate up to 3 characters of output.

Control characters are displayed with their ASCII mnemonic, the hex value, and the corresponding key that is pressed simultaneously with the CTRL (Control) key. For example, the DC3 ASCII code is generated by pressing the "S" or "s" key and the CTRL key simultaneously. It is displayed with an up arrow as $\wedge$ S.

This appendix contains an ASCII code chart and special code charts that are only applicable to the Type 3 terminal.

ASCII CODE CHART

The ASCII code chart is shown in table A-1.

TABLE A-1. ASCII CODE CHART

BITS												
				2^7	0	0	0	0	0	0	0	0
				2^6	0	0	0	0	1	1	1	1
				2^5	0	0	1	1	0	0	1	1
				2^4	0	1	0	1	0	1	0	1
2^3 ↓	2^2 ↓	2^1 ↓	2^0 ↓	COLUMN → / ↓ ROW	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	`	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	10(A)	LF	SUB	*	:	J	Z	j	z
1	0	1	1	11(B)	VT	ESC	+	;	K	[	k	{
1	1	0	0	12(C)	FF	FS	,	<	L	\	l	|
1	1	0	1	13(D)	CR	GS	-	=	M	]	m	}
1	1	1	0	14(E)	SO	RS	.	>	N	^	n	~
1	1	1	1	15(F)	SI	US	/	?	O	_	o	DEL

04074-1

SPECIAL CODE CHARTS APPLICABLE ONLY TO TYPE 3 TERMINAL

The following tables A-2 through A-9 give the special code sets which may be used in CYBER mode under CP/M.

TABLE A-2. LINE-DRAWING SYMBOL CODES

BITS						
b7 →					0	0
b6 →					1	1
b5 →					0	1
b4	b3	b2	b1	COLUMN → / ROW ↓	2	3
0	0	0	0	0	─	╝
0	0	0	1	1	│	╦
0	0	1	0	2	┌	╩
0	0	1	1	3	┐	╠
0	1	0	0	4	└	╣
0	1	0	1	5	┘	╬
0	1	1	0	6	┬	‖│
0	1	1	1	7	┴	│‖
1	0	0	0	8	├	▟
1	0	0	1	9	┤	▛
1	0	1	0	10 (A)	┼	▜
1	0	1	1	11 (B)	═	▙
1	1	0	0	12 (C)	║	▀
1	1	0	1	13 (D)	╔	▄
1	1	1	0	14 (E)	╗	
1	1	1	1	15 (F)	╚	█

02016-2

TABLE A-3. PLATO SYMBOL CODES

BITS B4	B3	B2	B1	B7 B6 B5 → / ROW ↓ COLUMN	1 Ø Ø 4	1 Ø 1 5	1 1 0 6	1 1 1 7
Ø	Ø	Ø	Ø	0		α	c̄	\
Ø	Ø	Ø	1	1	/	β	¨	\
Ø	Ø	1	Ø	2	≡	δ	□	/
Ø	Ø	1	1	3	~	λ	○	/
Ø	1	Ø	Ø	4	⇦	μ	◆	‾
Ø	1	Ø	1	5	≠	π	×	_
Ø	1	1	Ø	6	↑	ρ	´	|
Ø	1	1	1	7	→	σ	ˏ	|
1	Ø	Ø	Ø	8	↓	ω	ˇ	▒
1	Ø	Ø	1	9	←	≤	↕	▒
1	Ø	1	Ø	10 (A)	×	≥	|	=
1	Ø	1	1	11 (B)	Σ	θ		=
1	1	Ø	Ø	12 (C)	Δ	∢		◣
1	1	Ø	1	13 (D)	∪	°		◣
1	1	1	Ø	14 (E)	∩	⋗		◢
1	1	1	1	15 (F)	÷	⪢		◢

03415-3

TABLE A-4. UNITED KINGDOM CHARACTER CODES

BITS				b7 b6 b5 →	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b4 ↓	b3 ↓	b2 ↓	b1 ↓	COLUMN → / ↓ ROW	2	3	4	5	6	7
0	0	0	0	0	SP	0	@	P	`	p
0	0	0	1	1	!	1	A	Q	a	q
0	0	1	0	2	"	2	B	R	b	r
0	0	1	1	3	£	3	C	S	c	s
0	1	0	0	4	$	4	D	T	d	t
0	1	0	1	5	%	5	E	U	e	u
0	1	1	0	6	&	6	F	V	f	v
0	1	1	1	7	'	7	G	W	g	w
1	0	0	0	8	(	8	H	X	h	x
1	0	0	1	9	)	9	I	Y	i	y
1	0	1	0	10(A)	*	:	J	Z	j	z
1	0	1	1	11(B)	+	;	K	[	k	{
1	1	0	0	12(C)	,	<	L	\	l	¦
1	1	0	1	13(D)	-	=	M	]	m	}
1	1	1	0	14(E)	.	>	N	^	n	‾
1	1	1	1	15(F)	/	?	O	_	o	

02015-8

TABLE A-5. FRENCH CHARACTER CODES

BITS										
b7					0	0	1	1	1	1
b6					1	1	0	0	1	1
b5					0	1	0	1	0	1
b4	b3	b2	b1	COLUMN → / ROW ↓	2	3	4	5	6	7
0	0	0	0	0	SP	0	à	P	`	p
0	0	0	1	1	!	1	A	Q	a	q
0	0	1	0	2	"	2	B	R	b	r
0	0	1	1	3	£	3	C	S	c	s
0	1	0	0	4	$	4	D	T	d	t
0	1	0	1	5	%	5	E	U	e	u
0	1	1	0	6	&	6	F	V	f	v
0	1	1	1	7	'	7	G	W	g	w
1	0	0	0	8	(	8	H	X	h	x
1	0	0	1	9	)	9	I	Y	i	y
1	0	1	0	10(A)	*	:	J	Z	j	z
1	0	1	1	11(B)	+	;	K	°	k	é
1	1	0	0	12(C)	,	<	L	ç	l	ù
1	1	0	1	13(D)	-	=	M	§	m	è
1	1	1	0	14(E)	.	>	N	^	n	¨
1	1	1	1	15(F)	/	?	O	_	o	

02015-1

TABLE A-6. GERMAN CHARACTER CODES

BITS				b7 →	0	0	1	1	1	1
				b6 →	1	1	0	0	1	1
				b5 →	0	1	0	1	0	1
b4 ↓	b3 ↓	b2 ↓	b1 ↓	COLUMN → / ↓ ROW	2	3	4	5	6	7
0	0	0	0	0	SP	0	§	P	`	p
0	0	0	1	1	!	1	A	Q	a	q
0	0	1	0	2	"	2	B	R	b	r
0	0	1	1	3	#	3	C	S	c	s
0	1	0	0	4	$	4	D	T	d	t
0	1	0	1	5	%	5	E	U	e	u
0	1	1	0	6	&	6	F	V	f	v
0	1	1	1	7	'	7	G	W	g	w
1	0	0	0	8	(	8	H	X	h	x
1	0	0	1	9	)	9	I	Y	i	y
1	0	1	0	10(A)	*	:	J	Z	j	z
1	0	1	1	11(B)	+	;	K	Ä	k	ä
1	1	0	0	12(C)	,	<	L	Ö	l	ö
1	1	0	1	13(D)	-	=	M	Ü	m	ü
1	1	1	0	14(E)	.	>	N	∧	n	ß
1	1	1	1	15(F)	/	?	O	_	o	

02015-2

TABLE A-7. SWEDISH/FINNISH CHARACTER CODES

BITS b7 →					0	0	1	1	1	1
b6 →					1	1	0	0	1	1
b5 →					0	1	0	1	0	1
b4 ↓	b3 ↓	b2 ↓	b1 ↓	COLUMN → / ↓ ROW	2	3	4	5	6	7
0	0	0	0	0	SP	0	É	P	é	p
0	0	0	1	1	!	1	A	Q	a	q
0	0	1	0	2	"	2	B	R	b	r
0	0	1	1	3	#	3	C	S	c	s
0	1	0	0	4	¤	4	D	T	d	t
0	1	0	1	5	%	5	E	U	e	u
0	1	1	0	6	&	6	F	V	f	v
0	1	1	1	7	'	7	G	W	g	w
1	0	0	0	8	(	8	H	X	h	x
1	0	0	1	9	)	9	I	Y	i	y
1	0	1	0	10(A)	*	:	J	Z	j	z
1	0	1	1	11(B)	+	;	K	Ä	k	ä
1	1	0	0	12(C)	,	<	L	Ö	l	ö
1	1	0	1	13(D)	-	=	M	Å	m	å
1	1	1	0	14(E)	.	>	N	Ü	n	ü
1	1	1	1	15(F)	/	?	O	_	o	

02015-7

TABLE A-8. DANISH/NORWEGIAN CHARACTER CODES

BITS				b7 b6 b5 →	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b4	b3	b2	b1	COLUMN → / ROW ↓	2	3	4	5	6	7
0	0	0	0	0	SP	0	@	P	`	p
0	0	0	1	1	!	1	A	Q	a	q
0	0	1	0	2	"	2	B	R	b	r
0	0	1	1	3	#	3	C	S	c	s
0	1	0	0	4	$	4	D	T	d	t
0	1	0	1	5	%	5	E	U	e	u
0	1	1	0	6	&	6	F	V	f	v
0	1	1	1	7	'	7	G	W	g	w
1	0	0	0	8	(	8	H	X	h	x
1	0	0	1	9	)	9	I	Y	i	y
1	0	1	0	10(A)	*	:	J	Z	j	z
1	0	1	1	11(B)	+	;	K	Æ	k	æ
1	1	0	0	12(C)	,	<	L	Ø	l	ø
1	1	0	1	13(D)	-	=	M	Å	m	å
1	1	1	0	14(E)	.	>	N	^	n	‾
1	1	1	1	15(F)	/	?	O	_	o	

02015-10

TABLE A-9. SPANISH CHARACTER CODES

Bits				b7 / b6 / b5 →	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b4 ↓	b3 ↓	b2 ↓	b1 ↓	COLUMN → / ↓ ROW	2	3	4	5	6	7
0	0	0	0	0	SP	0	§	P	`	p
0	0	0	1	1	!	1	A	Q	a	q
0	0	1	0	2	"	2	B	R	b	r
0	0	1	1	3	£	3	C	S	c	s
0	1	0	0	4	$	4	D	T	d	t
0	1	0	1	5	%	5	E	U	e	u
0	1	1	0	6	&	6	F	V	f	v
0	1	1	1	7	'	7	G	W	g	w
1	0	0	0	8	(	8	H	X	h	x
1	0	0	1	9	)	9	I	Y	i	y
1	0	1	0	10(A)	*	:	J	Z	j	z
1	0	1	1	11(B)	+	;	K	¡	k	°
1	1	0	0	12(C)	,	<	L	Ñ	l	ñ
1	1	0	1	13(D)	-	=	M	¿	m	ç
1	1	1	0	14(E)	.	>	N	^	n	~
1	1	1	1	15(F)	/	?	O	_	o	

02015-9

This appendix shows the various keycap configurations of the Type 3 terminal keyboard and lists the keyboard codes.

TYPE 3 TERMINAL KEYCAP CONFIGURATIONS

Figures B-1 through B-7 show the variety of keycap configurations that may be used with the terminal. These include the:

- Keyboard with standard keycaps (figure B-1)
- Keyboard with United Kingdom keycap option (figure B-2)
- Keyboard with French keycap option (figure B-3)
- Keyboard with German keycap option (figure B-4)
- Keyboard with Swedish/Finnish keycap option (figure B-5)
- Keyboard with Danish/Norwegian keycap option (figure B-6)
- Keyboard with Spanish keycap option (figure B-7)

Each keycap option consists of a user-installed keycap kit. The corresponding characters are displayed by changing the factory-set language in the terminal installation parameters. This is covered in the 110 Installation Guide.

NOTE

The] [key of the French, German, Swedish/Finnish, Danish/Norwegian, and Spanish options generates a unique code sequence that is only for PLATO applications. This unique code sequence does not correspond with the symbols on the key.

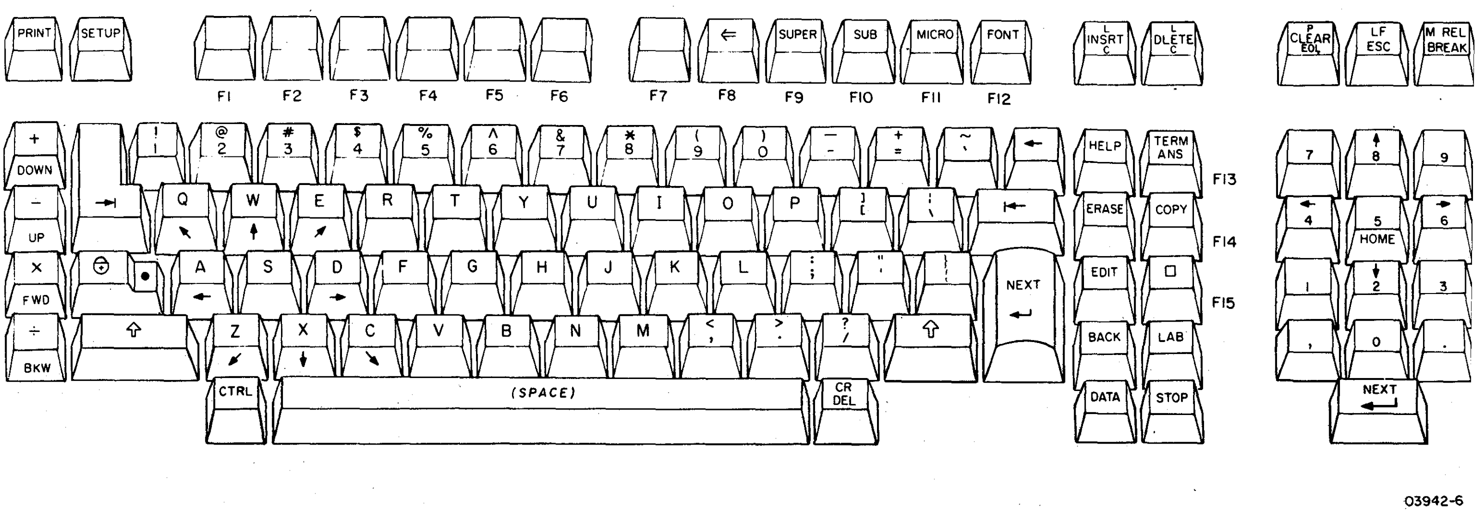
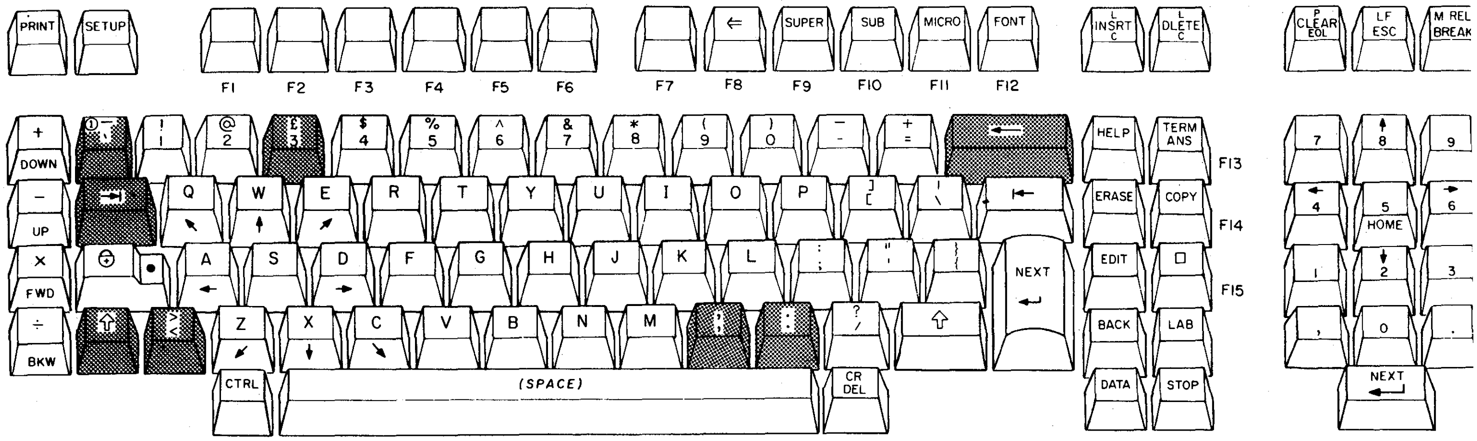


Figure B-1. Type 3 Terminal Keyboard with Standard Keycaps

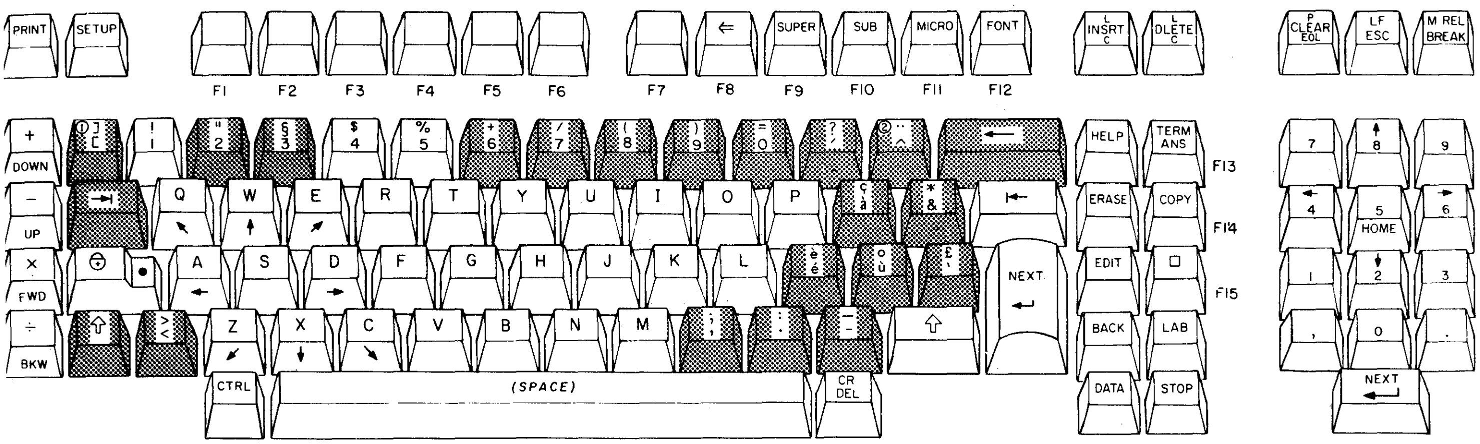


NOTES:
OPTIONAL KEYCAPS ARE SHADED FOR ILLUSTRATING PURPOSES

03942-1

①UPPER-OVER BAR
LOWER-GRAVE ACCENT

Figure B-2. Type 3 Terminal Keyboard with United Kingdom Keycap Option



NOTES:
OPTIONAL KEYCAPS ARE SHADED FOR ILLUSTRATING PURPOSES

03942-18

①PROVIDED FOR PLATO USAGE.
②UPPER-UMLAUT

Figure B-3. Type 3 Terminal Keyboard with French Keycap Option

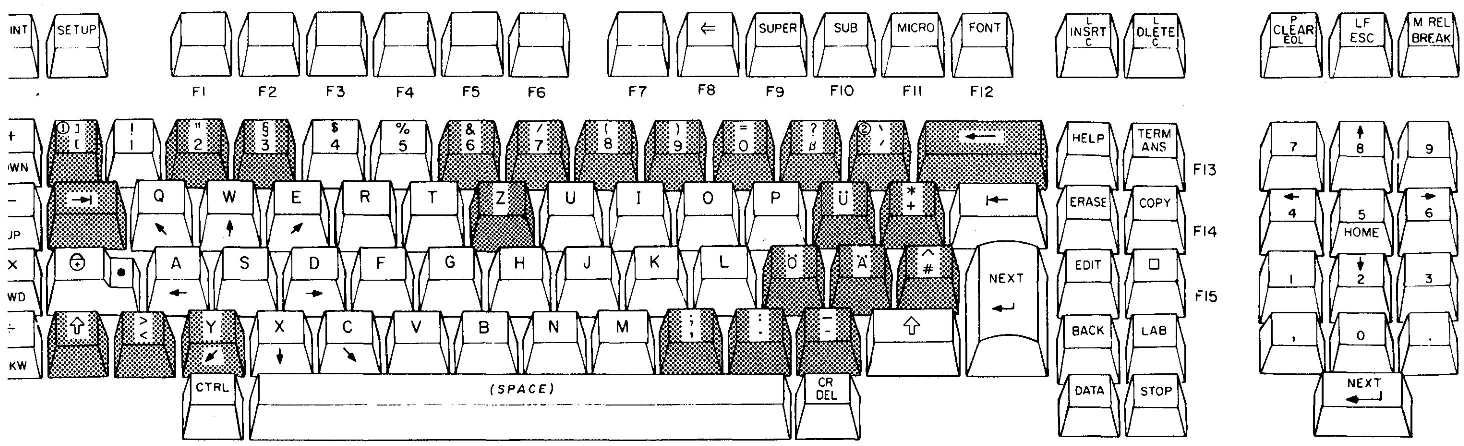


Figure B-4. Type 3 Terminal Keyboard with German Keycap Option

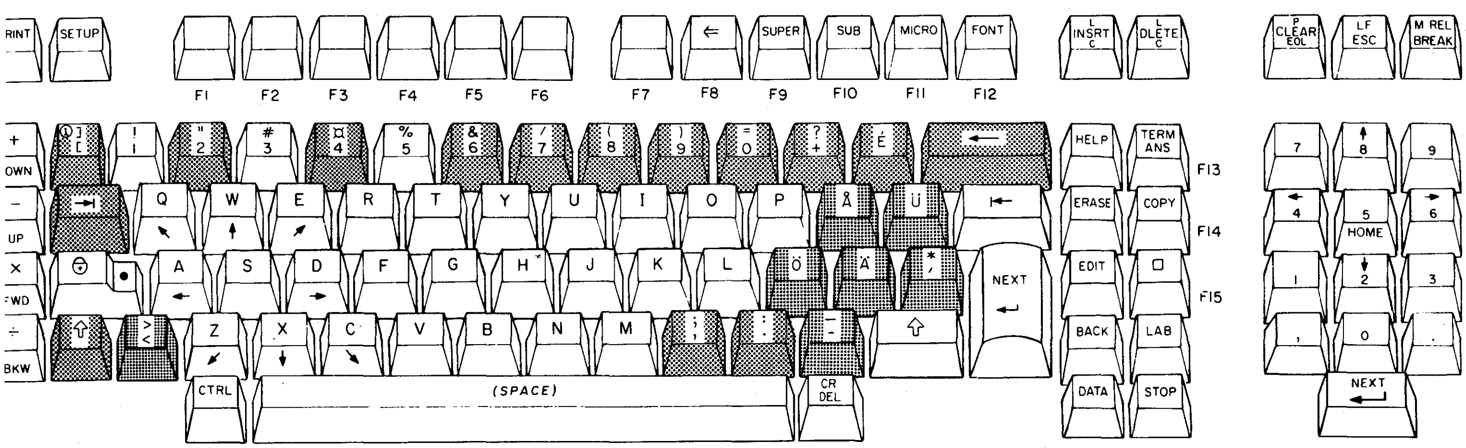
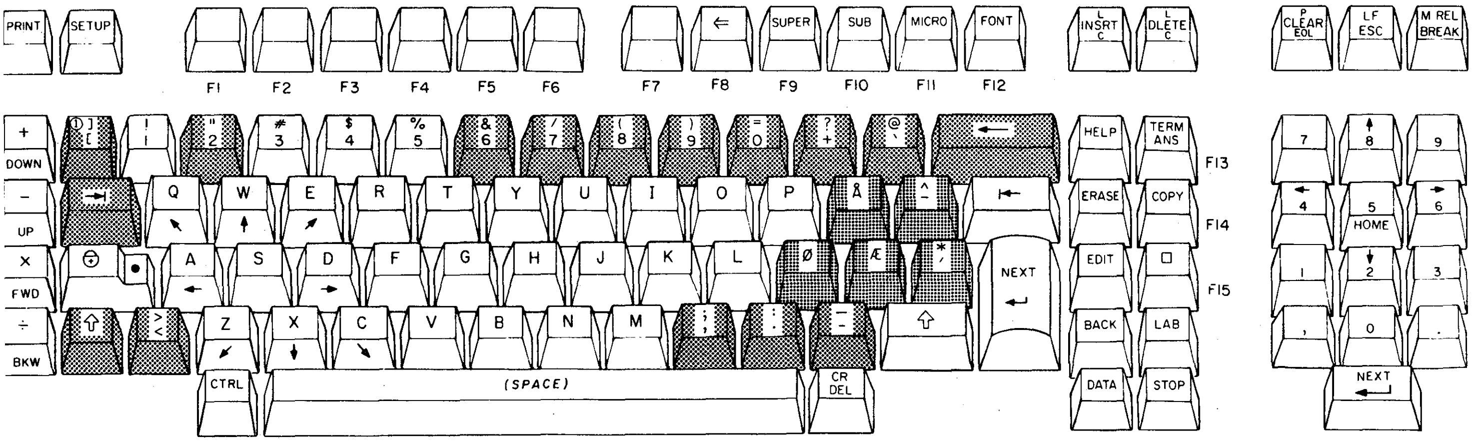


Figure B-5. Type 3 Terminal Keyboard with Swedish/Finnish Keycap Option

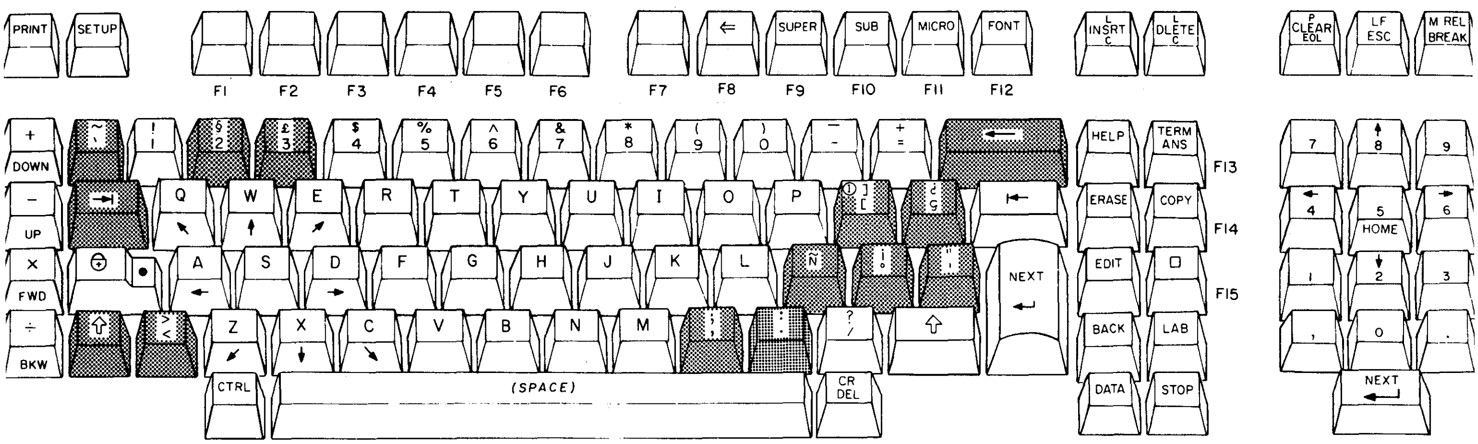


NOTES:
OPTIONAL KEYCAPS ARE SHADED FOR ILLUSTRATING PURPOSES

①PROVIDED FOR PLATO USAGE

03942-29

Figure B-6. Type 3 Terminal Keyboard with Danish/Norwegian Keycap Option



NOTES:
OPTIONAL KEYCAPS ARE SHADED FOR ILLUSTRATING PURPOSES

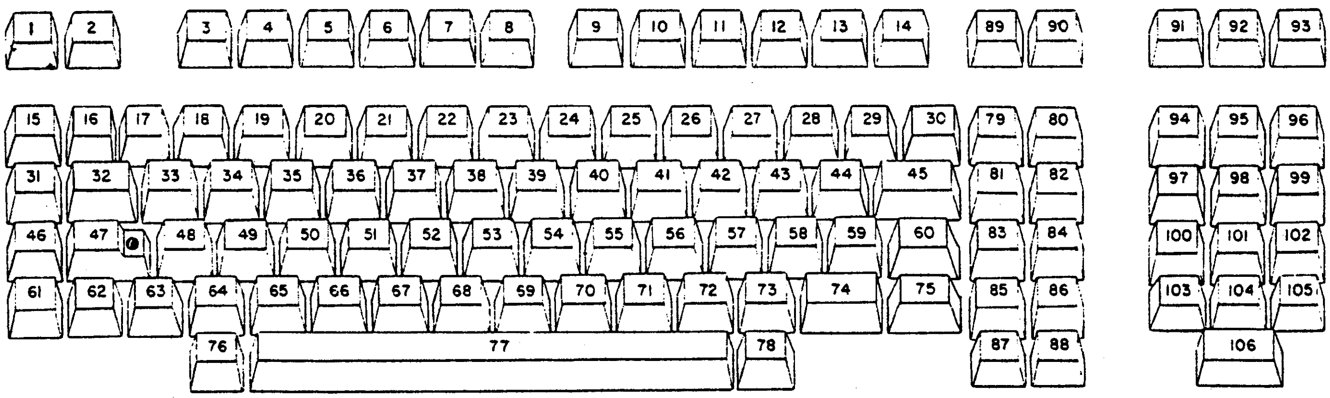
①PROVIDED FOR PLATO USAGE

03942-30

Figure B-7. Type 3 Terminal Keyboard with Spanish Keycap Option

TYPE 3 TERMINAL KEYBOARD CODES

The keystation numbers assigned to the keys of the terminal keyboard are shown in figure B-8. These keystation numbers are used in tables B-1 and B-2 to indicate the code or function of each key. Table B-1 covers the keycodes generated when in CYBER mode and table B-2 covers the keycodes generated when in compatibility mode. Each of these tables have addendums for the keycap options.



03942-11

Figure B-8. Keyboard Keystation Assignments for Type 3 Terminal

TABLE B-1. CYBER MODE KEYCODES

KEY NO.	NOTES*	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
1	DS, L		PRINT					
		Small CYBER Mode			1E,11	1E,01	1E,11	1E,01
		Large CYBER Mode			1E,02	1E,01	1E,02	1E,01
2			SETUP		-	-	-	-
3	D, L		(F1)		1E,71	1E,61	1E,31	1E,21
4	D, L		(F2)		1E,72	1E,62	1E,32	1E,22
5	D, L		(F3)		1E,73	1E,63	1E,33	1E,23
6	D, L		(F4)		1E,74	1E,64	1E,34	1E,24
7	D, L		(F5)		1E,75	1E,65	1E,35	1E,25
8	D, L		(F6)		1E,76	1E,66	1E,36	1E,26
9	D, L		(F7)		1E,77	1E,67	1E,37	1E,27

TABLE B-1. CYBER MODE KEYCODES (CONTD)

KEY NO.	NOTES*	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ ·CTRL
10	D, L		⇐ (F8)		1E,78	1E,68	1E,38	1E,28
11	D, L		SUPER (F9)		1E,79	1E,69	1E,39	1E,29
12	D, L		SUB (F10)		1E,7A	1E,6A	1E,3A	1E,2A
13	D, L		MICRO (F11)		1E,7B	1E,6B	1E,3B	1E,2B
14	D, L		FONT (F12)		1E,7C	1E,6C	1E,3C	1E,2C
15	L,D		+	DOWN **	1E,12,20	1E,12,21	1E,12,22	1E,12,23
16	R, L				-	-	-	-
17	R	1		!	31	21	31	21
18	R	2		@	32	40	00	00
19	R	3		#	33	23	33	23
20	R	4		$	34	24	34	24
21	R	5		%	35	25	35	25
22	R	6		^	36	5E	36	5E
23	R	7		&	37	26	37	26
24	R	8		*	38	2A	38	2A
25	R	9		(	39	28	39	28
26	R	0		)	30	29	30	29
27	R	-		—	2D	5F	1F	1F
28	R	=		+	3D	2B	1E	1E
29	R	`		~	60	7E	60	7E
30	R		←					
		Small CYBER Mode			19	19	19	19
		Large CYBER Mode			08	08	08	08
31	L,D		—	UP **	1E,12,24	1E,12,25	1E,12,26	1E,12,27
32	R,L,DS		→|		09	09	1E,12,57	1E,12,57

TABLE B-1. CYBER MODE KEYCODES (CONTD)

KEY NO.	NOTES*	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
33	R		Q	↖ **	71	51	11	11
34	R		W	↑ **	77	57	17	17
35	R		E	↗ **	65	45	05	05
36	R		R		72	52	12	12
37	R		T		74	54	14	14
38	R		Y		79	59	19	19
39	R		U		75	55	15	15
40	R		I		69	49	09	09
41	R		O		6F	4F	0F	0F
42	R		P		70	50	10	10
43	R	[		]	5B	5D	1D	1D
44	R	\		¦	5C	7C	1C	1C
45	R,L,DS		⇤		1E,0B	1E,0B	1E,12,58	1E,12,58
46	D,L		X	FWD **	1E,12,28	1E,12,29	1E,12,2A	1E,12,2B
47					-	-	-	-
48	R		A	← **	61	41	01	01
49	R		S		73	53	13	13
50	R		D	→ **	64	44	04	04
51	R		F		66	46	06	06
52	R		G		67	47	07	07
53	R		H		68	48	08	08
54	R		J		6A	4A	0A	0A
55	R		K		6B	4B	0B	0B
56	R		L		6C	4C	0C	0C
57	R	;		:	3B	3A	3B	3A
58	R	'		"	27	22	27	22
59	R	{		}	7B	7D	7B	7D
60	See Key 75				-	-	-	-

TABLE B-1. CYBER MODE KEYCODES (CONTD)

KEY NO.	NOTES*	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
61	D, L		÷	BKW **	1E,12,2C	1E,12,2D	1E,12,2E	1E,12,2F
62			⇧		-	-	-	-
63	R				-	-	-	-
64	R		Z	↙ **	7A	5A	1A	1A
65	R		X	↓ **	78	58	18	18
66	R		C	↘ **	63	43	03	03
67	R		V		76	56	16	16
68	R		B		62	42	02	02
69	R		N		6E	4E	0E	0E
70	R		M		6D	4D	0D	0D
71	R	,		<	2C	3C	2C	3C
72	R	.		>	2E	3E	2E	3E
73	R	/		?	2F	3F	2F	3F
74			⇧		-	-	-	-
75	L	↵		NEXT				
		Small CYBER Mode			0A	0A	0A	0A
		Large CYBER Mode			0D	0D	0D	0D
76			CTRL		-	-	-	-
77	R		(Space)		20	20	20	20
78		DEL		CR	7F	0D	7F	0D
79	D, L		HELP		1E,5C	1E,58	1E,5C	1E,58
80	D, L	ANS	(F13)	TERM	1E,7D	1E,6D	1E,3D	1E,2D
81	DS, L		ERASE					
		Small CYBER Mode			1E,5D	1E,59	1E,5D	1E,59
		Large CYBER Mode			1F	1E,5D	1E,5D	1E,59
82	D, L		COPY (F14)		1E,7E	1E,6E	1E,3E	1E,2E
83	D, L		EDIT		1E,5E	1E,5A	1E,5E	1E,5A

TABLE B-1. CYBER MODE KEYCODES (CONTD)

KEY NO.	NOTES*	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
84	D, L		□ (F15)		1E,70	1E,60	1E,30	1E,20
85	D, L		BACK		1E,5F	1E,5B	1E,5F	1E,5B
86	D, L		LAB		1E,12,31	1E,12,32	1E,12,33	1E,12,33
87	D, L		DATA		1E,12,35	1E,12,36		
88	D, L		STOP		1E,49	1E,4A	1E,49	1E,4A
89	DS, L, R	C	INSRT	L	1E,4F	1E,52	1E,4F	1E,52
90	DS, L, R	C	DLETE	L	1E,4E	1E,51	1E,4E	1E,51
91	L	EOL	CLEAR	P	0B	0C	0B	0C
92		ESC		LF	1B	0A	1B	0A
93		BREAK		M REL	BREAK	11	BREAK	11
94	R, L,N		7		37	-	37	-
95	R, L,N	8		↑	38	17	38	17
96	R, L,N		9		39	-	39	-
97	R,L,N, D	4		←				
		Small CYBER Mode			34	19	19	1E,19
		Large CYBER Mode			34	08	34	08
98	R,L,N, D	5		HOME**				
		Small CYBER Mode			35	08	08	1E,08
		Large CYBER Mode			35	19	35	19
99	R,L,N, D	6		→				
		Small CYBER Mode			36	18	18	1E,18
		Large CYBER Mode			36	18	36	18
100	R, L,N	1			31	-	31	-
101	R,L,N, D	2		↓				
		Small CYBER Mode			32	1A	1A	1E,1A
		Large CYBER Mode			32	1A	32	1A

TABLE B-1. CYBER MODE KEYCODES (CONTD)

KEY NO.	NOTES*	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
102	R, L,N	3			33	-	33	-
103	R, L,N	,			2C	-	2C	-
104	R, L,N	0			30	-	30	-
105	R, L,N	.			2E	-	2E	-
106	L,N	↵		NEXT				
		Small CYBER Mode			0A	0A	0A	0A
		Large CYBER Mode			0D	0D	0D	0D

*Key to Notes:

- N - Modified if the Numeric Pad parameter is set to SHIFT.
- R - Auto repeat if TYPAMATIC is on.
- L - Host loadable.
- D - Delimiter. CR sent when enabled by host.
- DS - Delimiter. CR sent when enabled by host in small CYBER mode.
- \- - No code generated.
- ** - Labeled on skirt of keycap.

TABLE B-1.1. CYBER MODE KEYCODE ADDENDUM FOR UNITED KINGDOM KEYCAP OPTION

KEY NO.	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
16	`		-	60	7E	60	7E
19	3		£	33	23	33	23
29				-	-	-	-
63	<		>	3C	3E	3C	3E
71	,		,	2C	2C	2C	2C
72	.		.	2E	2E	2E	2E

TABLE B-1.2. CYBER MODE KEYCODE ADDENDUM FOR FRENCH KEYCAP OPTION

KEY	KEY LEGENDS			PRESSED WITH KEY ... GENERATE			
NO.	LOWER	CENTER	UPPER		⇧	CTRL	⇧ •CTRL
16	[		]	1E, 4B	1E, 4D	1E, 4B	1E, 4D
18	2		"	32	22	00	00
19	3		§	33	5D	33	5D
22	6		+	36	2B	36	2B
23	7		/	37	2F	37	2F
24	8		(	38	28	38	28
25	9		)	39	29	39	29
26	0		=	30	3D	30	3D
27	'		?	27	3F	1F	1F
28	^		¨	5E	7E	1E	1E
29				-	-	-	-
43	à		ç	40	5C	1D	1D
44	&		*	26	2A	1C	1C
57	é		è	7B	7D	7B	7D
58	ù		°	7C	5B	7C	5B
59	`		£	60	23	60	23
63	<		>	3C	3E	3C	3E
71	,		;	2C	3B	2C	3B
72	.		:	2E	3A	2E	3A
73	-		_	2D	5F	2D	5F

TABLE B-1.3. CYBER MODE KEYCODE ADDENDUM FOR GERMAN KEYCAP OPTION

KEY NO.	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
16	[		]	1E, 4B	1E, 4D	1E, 4B	1E, 4D
18	2		"	32	22	00	00
19	3		§	33	40	33	40
22	6		&	36	26	36	26
23	7		/	37	2F	37	3F
24	8		(	38	28	38	28
25	9		)	39	29	39	29
26	0		=	30	3D	30	3D
27	β		?	7E	3F	1F	1F
28	´		`	27	60	1E	1E
29				-	-	-	-
38		Z		7A	5A	19	19
43		Ü		7D	5D	1D	1D
44	+		*	2B	2A	1C	1C
57		Ö		7C	5C	7C	5C
58		Ä		7B	5B	7B	5B
59	#		^	23	5E	23	5E
63	<		>	3C	3E	3C	3E
64		Y		79	59	1A	1A
71	,		;	2C	3B	2C	3B
72	.		:	2E	3A	2E	3A
73	-		_	2D	5F	2D	5F

TABLE B-1.4. CYBER MODE KEYCODE ADDENDUM FOR SWEDISH/FINNISH KEYCAP OPTION

KEY NO.	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
16	[		]	1E, 4B	1E, 4D	1E, 4B	1E, 4D
18	2		"	32	22	00	00
19	3		#	33	23	33	23
20	4		¤	34	24	34	24
22	6		&	36	26	36	26
23	7		/	37	2F	37	2F
24	8		(	38	28	38	28
25	9		)	39	29	39	29
26	0		=	30	3D	30	3D
27	+		?	2B	3F	1F	1F
28		É		60	40	1E	1E
29				-	-	-	-
43		Å		7D	5D	1D	1D
44		Ü		7E	5E	1C	1C
57		Ö		7C	5C	7C	5C
58		Ä		7B	5B	7B	5B
59	'		*	27	2A	27	2A
63	<		>	3C	3E	3C	3E
71	,		;	2C	3B	2C	3B
72	.		:	2E	3A	2E	3A
73	-		_	2D	5F	2D	5F

TABLE B-1.5. CYBER MODE KEYCODE ADDENDUM FOR DANISH/NORWEGIAN KEYCAP OPTION

KEY NO.	KEY LEGENDS			PRESSED WITH KEY ... GENERATE			
	LOWER	CENTER	UPPER		⇧	CTRL	⇧ •CTRL
16	[		]	1E, 4B	1E, 4D	1E, 4B	1E, 4D
18	2		"	32	22	00	00
22	6		&	36	26	36	26
23	7		/	37	2F	37	2F
24	8		(	38	28	38	28
25	9		)	39	29	39	29
26	0		=	30	3D	30	3D
27	+		?	2B	3F	1F	1F
28	`		@	60	40	1E	1E
29				-	-	-	-
43		Å		7D	5D	1D	1D
44	‾		^	7E	5E	1C	1C
57		Ø		7C	5C	7C	5C
58		Æ		7B	5B	7B	5B
59	'		*	27	2A	27	2A
63	<		>	3C	3E	3C	3E
71	,		;	2C	3B	2C	3B
72	.		:	2E	3A	2E	3A
73	-		_	2D	5F	2D	5F

TABLE B-1.6. CYBER MODE KEYCODE ADDENDUM FOR SPANISH KEYCAP OPTION

KEY NO.	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
16	\`		~	60	7E	60	7E
18	2		§	32	40	00	00
19	3		£	33	23	33	23
29				-	-	-	-
43	]		[	1E, 4B	1E, 4D	1D	1D
44	Ç		¿	7D	5D	1C	1C
57		Ñ	´	7C	5C	7C	5C
58	°		¡	7B	5B	7B	5B
59	'		"	27	22	27	22
63	<		>	3C	3E	3C	3E
71	,		;	2C	3B	2C	3B
72	.		:	2E	3A	2E	3A

TABLE B-2. COMPATIBILITY MODE KEYCODES

KEY NO.	NOTES*	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
1	1		PRINT		Print screen	Print screen	Print screen	Print screen
2			SETUP		-	-	-	-
3	1		(F1)		02	02	02	02
4	1		(F2)		06	06	06	06
5	1		(F3)		1B,42	1B,42	1B,42	1B,42
6	1		(F4)		1B,46	1B,46	1B,46	1B,46
7	1		(F5)		01	01	01	01
8	1		(F6)		05	05	05	05
9	1		(F7)		10	10	10	10

KEY NO.	NOTES*	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ ·CTRL
10	1		⇐ (F8)		0E	0E	0E	0E
11	1		SUPER (F9)		1B,56	1B,56	1B,56	1B,56
12	1		SUB (F10)		16	16	16	16
13	1		MICRO (F11)		No action	No action	No action	No action
14	1		FONT (F12)		No action	No action	No action	No action
15	1		+	DOWN **	2B	23	2B	23
16	R				-	-	-	-
17	R	1		!	31	21	31	21
18	R	2		@	32	40	00	00
19	R	3		#	33	23	33	23
20	R	4		$	34	24	34	24
21	R	5		%	35	25	35	25
22	R	6		∧	36	5E	36	5E
23	R	7		&	37	26	37	26
24	R	8		*	38	2A	38	2A
25	R	9		(	39	28	39	28
26	R	0		)	30	29	30	29
27	R	-		—	2D	5F	1F	1F
28	R	=		+	3D	2B	1E	1E
29	R	`		~	60	7E	60	7E
30	R		◄—		08	08	08	08
31	1		—	UP **	2D	26	2D	26
32			→|		09	09	1E,12,57	1E,12,57
33	R		Q	↖ **	71	51	11	11
34	R		W	↑ **	77	57	17	17
35	R		E	↗ **	65	45	05	05

KEY NO.	NOTES*	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
36	R		R		72	52	12	12
37	R		T		74	54	14	14
38	R		Y		79	59	19	19
39	R		U		75	55	15	15
40	R		I		69	49	09	09
41	R		O		6F	4F	0F	0F
42	R		P		70	50	10	10
43	R	[		]	5B	5D	1D	1D
44	R	\		¦	5C	7C	1C	1C
45	1		⇤		08	08	08	08
46	1		X	FWD **	40	60	40	60
47			⊕		-	-	-	-
48	R		A	← **	61	41	01	01
49	R		S		73	53	13	13
50	R		D	→ **	64	44	04	04
51	R		F		66	46	06	06
52	R		G		67	47	07	07
53	R		H		68	48	08	08
54	R		J		6A	4A	0A	0A
55	R		K		6B	4B	0B	0B
56	R		L		6C	4C	0C	0C
57	R	;		:	3B	3A	3B	3A
58	R	'		"	27	22	27	22
59	R	{		}	7B	7D	7B	7D
60	See Key 75				-	-	-	-
61	1		÷	BKW **	5C	7C	5C	7C
62			⇧		-	-	-	-
63	R				-	-	-	-
64	R		Z	↙ **	7A	5A	1A	1A
65	R		X	↓ **	78	58	18	18

TABLE B-2. COMPATIBILITY MODE KEY FUNCTIONS (CONTD)

KEY NO.	NOTES*	KEY LEGENDS: LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
66	R		C	↘ **	63	43	03	03
67	R		V		76	56	16	16
68	R		B		62	42	02	02
69	R		N		6E	4E	0E	0E
70	R		M		6D	4D	0D	0D
71	R	,		<	2C	3C	2C	3C
72	R	.		>	2E	3E	2E	3E
73	R	/		?	2F	3F	2F	3F
74			⇧		–	–	–	–
75		↵		NEXT	0D	0D	0D	0D
76			CTRL		–	–	–	–
77	R		(Space)		20	20	20	20
78		DEL		CR	7F	0D	7F	0D
79	1		HELP		03	03	03	03
80	1	ANS	(F13)	TERM	1B	1B	1B	1B
81	1		ERASE		1F	08	08	08
82	1		COPY (F14)		Control code follows	Print screen	Control code follows	Print screen
83	1		EDIT		0A	0A	0A	0A
84	1		□ (F15)		Clear screen	Clear screen	Clear screen	Clear screen
85	1		BACK		08	08	08	08
86	1		LAB		1E,12,31	1E,12,32	1E,12,33	1E,12,33
87	1		DATA		7F	7F	7F	7F
88	1		STOP		13	13	13	13
89	R	C	INSRT	L	1E,4F	1E,52	1E,4F	1E,52
90	R	C	DLETE	L	1E,4E	1E,51	1E,4E	1E,51
91		EOL	CLEAR	P	0B	0C	0B	0C
92		ESC		LF	1B	0A	1B	0A
93		BREAK		M REL	BREAK	11	BREAK	11
94	R		7		37	–	37	–

TABLE B-2. COMPATIBILITY MODE KEY FUNCTIONS (CONTD)

KEY NO.	NOTES*	KEY LEGENDS			PRESSED WITH KEY ... GENERATE			
		LOWER	CENTER	UPPER		⇧	CTRL	⇧ •CTRL
95	R,N	8			38	17	38	17
96	R,N		9	↑	39	-	39	-
97	R,N	4		←	34	08	34	08
98	R,N	5		HOME**	35	19	35	19
99	R,N	6		→	36	18	36	18
100	R,N	1			31	-	31	-
101	R,N	2		↓	32	1A	32	1A
102	R,N	3			33	-	33	-
103	R,N	,			2C	-	2C	-
104	R,N	0			30	-	30	-
105	R,N	.			2E	-	2E	-
106		↵		NEXT	0D	0D	0D	0D

*Key to Notes:

1 Codes listed reflect the default values for white keys. These values may be changed using the TERMSET utility or be changed via a user program (see Customizing with TERMSET or Dynamic Alteration of Terminal Parameters in section 13).

N - Modified if the Numeric Pad parameter is set to SHIFT.

R - Auto repeat if TYPAMATIC is on.

\- - No code generated.

** - Labeled on skirt of keycap.

TABLE B-2.1. COMPATIBILITY MODE KEY FUNCTION ADDENDUM FOR UNITED KINGDOM KEYCAP OPTION

KEY NO.	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
16	`		–	60	7E	60	7E
19	3		£	33	23	33	23
29				–	–	–	–
63	<		>	3C	3E	3C	3E
71	,		,	2C	2C	2C	2C
72	.		.	2E	2E	2E	2E

TABLE B-2.2. COMPATIBILITY MODE KEY FUNCTION ADDENDUM FOR FRENCH KEYCAP OPTION

KEY NO.	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
16	[		]	1E,4B	1E,4D	1E,4B	1E,4D
18	2		"	32	22	00	00
19	3		§	33	5D	33	5D
22	6		+	36	2B	36	2B
23	7		/	37	2F	37	2F
24	8		(	38	28	38	28
25	9		)	39	29	39	29
26	0		=	30	3D	30	3D
27	´		?	27	3F	1F	1F
28	^		¨	5E	7E	1E	1E
29				–	–	–	–
43	à		ç	40	5C	1D	1D
44	&		*	26	2A	1C	1C
57	é		è	7B	7D	7B	7D
58	ù		°	7C	5B	7C	5B
59	`		£	60	23	60	23

TABLE B-2.2. COMPATIBILITY MODE KEY FUNCTION ADDENDUM FOR FRENCH KEYCAP OPTION (CONTD)

KEY	KEY LEGENDS			PRESSED WITH KEY ... GENERATE			
NO.	LOWER	CENTER	UPPER		⇧	CTRL	⇧ •CTRL
63	<		>	3C	3E	3C	3E
71	,		;	2C	3B	2C	3B
72	.		:	2E	3A	2E	3A
73	-		_	2D	5F	2D	5F

TABLE B-2.3. COMPATIBILITY MODE KEY FUNCTION ADDENDUM FOR GERMAN KEYCAP OPTION

KEY	KEY LEGENDS			PRESSED WITH KEY ... GENERATE			
NO.	LOWER	CENTER	UPPER		⇧	CTRL	⇧ •CTRL
16	[		]	1E, 4B	1E, 4D	1E, 4B	1E, 4D
18	2		"	32	22	00	00
19	3		§	33	40	33	40
22	6		&	36	26	36	26
23	7		/	37	2F	37	3F
24	8		(	38	28	38	28
25	9		)	39	29	39	29
26	0		=	30	3D	30	3D
27	ß		?	7E	3F	1F	1F
28	´		`	27	60	1E	1E
29				-	-	-	-
38		Z		7A	5A	19	19
43		Ü		7D	5D	1D	1D
44	+		*	2B	2A	1C	1C
57		Ö		7C	5C	7C	5C
58		Ä		7B	5B	7B	5B
59	#		^	23	5E	23	5E
63	<		>	3C	3E	3C	3E
64		Y		79	59	1A	1A

TABLE B-2.3. COMPATIBILITY MODE KEY FUNCTION ADDENDUM FOR GERMAN KEYCAP OPTION (CONTD)

KEY NO.	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
71	,		;	2C	3B	2C	3B
72	.		:	2E	3A	2E	3A
73	-		_	2D	5F	2D	5F

TABLE B-2.4. COMPATIBILITY MODE KEY FUNCTION ADDENDUM FOR SWEDISH/FINNISH KEYCAP OPTION

KEY NO.	KEY LEGENDS LOWER	CENTER	UPPER	PRESSED WITH KEY ... GENERATE	⇧	CTRL	⇧ •CTRL
16	[		]	1E,4B	1E,4D	1E,4B	1E,4D
18	2		"	32	22	00	00
19	3		#	33	23	33	23
20	4		¤	34	24	34	24
22	6		&	36	26	36	26
23	7		/	37	2F	37	2F
24	8		(	38	28	38	28
25	9		)	39	29	39	29
26	0		=	30	3D	30	3D
27	+		?	2B	3F	1F	1F
28		É		60	40	1E	1E
29				-	-	-	-
43		Å		7D	5D	1D	1D
44		Ü		7E	5E	1C	1C
57		Ö		7C	5C	7C	5C
58		Ä		7B	5B	7B	5B
59	'		*	27	2A	27	2A
63	<		>	3C	3E	3C	3E
71	,		;	2C	3B	2C	3B
72	.		:	2E	3A	2E	3A
73	-		_	2D	5F	2D	5F

TABLE B-2.5. COMPATIBILITY MODE KEY FUNCTION ADDENDUM FOR DANISH/NORWEGIAN KEYCAP OPTION

KEY NO.	KEY LEGENDS			PRESSED WITH KEY ... GENERATE			
	LOWER	CENTER	UPPER		⇧	CTRL	⇧ ·CTRL
16	[		]	1E,4B	1E,4D	1E,4B	1E,4D
18	2		"	32	22	00	00
22	6		&	36	26	36	26
23	7		/	37	2F	37	2F
24	8		(	38	28	38	28
25	9		)	39	29	39	29
26	0		=	30	3D	30	3D
27	+		?	2B	3F	1F	1F
28	\`		@	60	40	1E	1E
29				-	-	-	-
43		Å		7D	5D	1D	1D
44	—		^	7E	5E	1C	1C
57		Ø		7C	5C	7C	5C
58		Æ		7B	5B	7B	5B
59	'		*	27	2A	27	2A
63	<		>	3C	3E	3C	3E
71	,		;	2C	3B	2C	3B
72	.		:	2E	3A	2E	3A
73	-		_	2D	5F	2D	5F

TABLE B-2.6. COMPATIBILITY MODE KEY FUNCTION ADDENDUM FOR SPANISH KEYCAP OPTION

KEY	KEY LEGENDS			PRESSED WITH KEY ... GENERATE			
NO.	LOWER	CENTER	UPPER		⇧	CTRL	⇧·CTRL
16	\`		~	60	7E	60	7E
18	2		§	32	40	00	00
19	3		£	33	23	33	23
29				-	-	-	-
43	]		[	1E,4B	1E,4D	1D	1D
44	ç		¿	7D	5D	1C	1C
57		Ñ		7C	5C	7C	5C
58	°		¡	7B	5B	7B	5B
59	'		"	27	22	27	22
63	<		>	3C	3E	3C	3E
71	,		;	2C	3B	2C	3B
72	.		:	2E	3A	2E	3A

This appendix describes how to check and set the parameters of the Type 3 terminal for running CP/M. The terminal has factory-set parameters for a typically configured terminal. The factory-set parameters have mode 3 assigned as the CP/M mode. A definition of the parameters is given in the CDC 721 Display Terminal Operator's Guide/Installation Instructions Manual or the 721-21/31 Owner's Manual (publication numbers are listed in preface).

To check the parameters, do as follows:

1. Press RESET button on terminal.

2. When self-test is complete and the mode menu display appears, press both the CTRL and SETUP keys to examine the terminal configuration setup. The following values are a minimum configuration to operate with CP/M.

F2	000010	(Flexible-disk option in) Additional bits may be set for other options
F3	000100	(Parallel-port option in) Additional bits may be set for other options such as the first bit if the internal modem is installed.
F4	000000	(Auto select disabled)
F5 and F6		Ignored by CP/M
F7	0 0000	(Standard United States characters are to be displayed). A first digit other than 0 or 1 will apply if terminal has a keycap option. The correct number for each keycap option is given in the previously mentioned operator's guide/installation instructions manual.
F8 and F9		Ignored by CP/M

   If no change to these parameters is required, go to step 3. If a change is required, press the F2 through F7 key that corresponds with the numbered block where the change is to be made. This moves the cursor within that block. Use the Space bar to go forward or the Backspace key to go backward and move the cursor to the digit of wrong value. Then key in the correct digit. When all changes have been made, press the COPY key to record them in nonvolatile memory before going to step 3.

3. Press the F10 key. Then press 3 and the NEXT keys to examine the parameters for mode 3 (CP/M mode). The following list gives the factory-set values for this mode.

F2	100110	(Mode enabled, load external, load from disk)
F3	000100	(Communication words of 7 data bits, odd-parity bit, and 1 stop bit)
F4	000000	(Typamatic on, home upper left, auto LF off)
F5	000000	(Pacing and bias disabled)
		or if the PHONE utility is to be used and the revision of the terminal firmware is below 4.0:
	010000	(Same as preceding plus cursor biasing enabled)*
F6	6C25	(6 = alert soft, margin alert on, printer selected, online) C = large CYBER, roll screen, alpha lock. 2 = cursor blink, cursor box, background dark. 5 = 30 lines, 80 char/line, full duplex.)
F7	000000	Not used
F8	000000	Not used
F9	00 6 6	Not used

If CYBER mode operations under CP/M control are to be performed, the parameters in blocks F3 and F6 for mode 3 must be changed to the following:

F3	001000	(Changes communication words to 8 data bits with the 8th bit a space and no parity)
F6	4C25	(Deselects printer so printing is not immediately enabled when CP/M mode is entered)

*The keyboard diagnostic test cannot be run with cursor biasing enabled.

If any changes to mode 3 parameters are to be made, follow the instructions in step 2 for making changes. When changes are made, be sure to press the COPY key to record them in nonvolatile memory.

4. Press F1 key to return to mode menu display. The parameters are set up to load CP/M from the flexible disk drive when the F3 key is pressed with mode menu on the screen.

This appendix gives some notes and cautions on the operation of the Control Data 110 CP/M 2.2 system.

TECHNICAL NOTES

- SUBMIT with XSUB
  - The control P character is not allowed in a file of XSUB commands, thus one cannot echo terminal output to the printer while using SUBMIT.
  - Commands cannot be submitted to PIP. Commands submitted to ED must be terminated with a control Z character. Apparently these two programs use BDOS function 10 for input, and the CP/M 2.2 User's Guide states that programs using function 10 cannot use XSUB as a source of input.
- DDT
  - The Control Data 110 system does not have the capability to execute an RST 7.
  - Assembly language programs can terminate with a RET statement, which causes them to return to the CCP. If these programs are executed under DDT, the last RET statement can cause the system to hang. This problem can be alleviated by replacing the RET with a JMP 0 or by terminating DDT before it reaches the final RET.
- CP/M 2.2 Interface Notes
  - The filename in an FCB is blank filled.
  - On initiation, CP/M 2.2 loads the checksums of the flexible disk in the Type 1 and Type 2 drives into memory. Each time a BDOS call is made, a comparison between the disk checksum and the checksum for that drive stored in memory is done. If the checksums do not match, the drive is defined to be read-only and disk interface functions that alter data on the disk are disallowed. Thus any time a disk in a drive is removed and a new disk is inserted, the new disk is defined to be read-only. The status of the new disk can be changed to read/write by rebooting the system with a control C or by calling the reset drive BDOS function.

- The Search For First and Search For Next BDOS functions place directory information into the DMA. Before the file desired can be accessed, the directory information must be loaded into the FCB for that file. Note that the drive code for the file is not contained in the directory information.

- The Search For Next function requires the original FCB with the imbedded question marks in the filename to operate correctly.

- Hazeltine Escape Sequences - The first tilde (~) is not printed out on the screen because it is a preface to an escape sequence and is used in the same manner as the ESC key.

- Use direct BIOS call to send to terminal values which are (or may be) null.

CAUTION

If the control P character locks the Type 1 printer in the print mode, it can be switched into the feed mode by pressing the PRINT and FEED buttons simultaneously.

This appendix contains a description of the commands used to direct the Type 3 terminal in CYBER mode and the resulting operations. However, preceding that are paragraphs that describe the CYBER submodes and the means of transferring between compatibility mode and CYBER mode.

CYBER SUBMODES

CYBER mode has two selectable submodes called large CYBER mode and small CYBER mode, which control the way the terminal translates commands. In the operator parameters of the terminal, large CYBER mode is always to be selected. When operating under CP/M, this is necessary to make terminal translation compatible with CP/M (this is described further in section 13 under Changing Screen Characteristics.

When CYBER mode is entered, the operator parameter selection of large CYBER causes the terminal to respond only to commands applicable in large CYBER mode (commands which apply only to large CYBER mode or small CYBER mode are indicated in tables E-1 and E-2 later in this appendix). However, small CYBER operations can be performed in CYBER mode by the user program issuing:

RS,DC2,'A'

As defined in table E-1, this code sequence commands the terminal to enter small CYBER mode. This overrides the large CYBER parameter selection and when small CYBER mode operations are complete, the program must issue the following command sequence to return to large CYBER mode before exiting to CP/M.

RS,DC2,'B'

TRANSFERRING FROM COMPATIBILITY MODE

Compatibility mode is in effect when the terminal is under CP/M control (the terminal characteristics in compatibility mode are discussed under Terminal Characteristics in section 13). To transfer from compatibility mode to CYBER mode requires that the user program issue the terminal an ESCape sequence of:

ESC,ESC,'b','b'

TRANSFERRING FROM CYBER MODE

The following ESCape sequence transfers the Type 3 terminal from CYBER mode back to compatibility mode.

ESC,ESC,'c,'c'

CYBER MODE COMMANDS AND RESPONSES

The remainder of this appendix describes the commands and responses of the Type 3 terminal while operating in CYBER mode. Some terms in the following descriptions may be unfamiliar. For definitions, refer to the CDC 721 Enhanced Display Terminal Hardware Reference Manual if your terminal has revision 4.0 firmware or to the CDC 721 Display Terminal Hardware Reference Manual if your terminal has earlier firmware than revision 4.0 (publication numbers are listed in preface). These manuals cover the characteristics of the terminal while operating in resident CYBER mode. Those characteristics also apply while operating in CYBER mode under CP/M control.

The CYBER mode commands are covered in tables E-1 and E-2 which follow. Table E-1 lists the commands by function in alphabetical order. When you find the desired command in table E-1, note its hexadecimal (hex) code and refer to table E-2. Table E-2 lists the commands in hex numerical order and describes the terminal responses.

TABLE E-1. LIST OF CYBER MODE COMMANDS BY FUNCTION

FUNCTION	HEX CODE
Alarm	
Sound	07
Attributes	
Attribute	
Clear to end of line and extend attribute*	1E,2A
Clear to end of page and extend attribute*	1E,2B
Read attribute	1E,0E
Use old attribute disable*	1E,2D
Use old attribute enable*	1E,2C
Blank	
Start	1E,12,5B
End	1E,12,5C
Blink	
Disable	04
Enable	03
Start	0E
Stop	0F
Code Bias	
Enable*	1E,30
Disable*	1E,31
Dim	
End	1D
Start	1C
Host Loaded Code	
Enable	1E,32
Disable	1E,33
Intensity	
Clear high intensity field	1E,40
Clear low intensity field	1E,3F
Inverse	
End	1E,45
Start	1E,44

*These commands only apply to terminals having revision 4.0 firmware or above.

TABLE E-1. LIST OF CYBER MODE COMMANDS BY FUNCTION (CONTD)

FUNCTION	HEX CODE
Protect	
Clear	1E,12,4A
Disable	1E,12,4C
Enable	1E,12,4B
Set all protect bits	1E,47
Start	1E,12,49
Underscore	
End	15
Start (large CYBER mode)	06
Start (small CYBER mode)	14
Validation	
End	1E,12,6E
Start	1E,12,6D
Communications control	
Bidirectional port	
Select bidirectional port N	1E,12,55,(N)
Block mode	
Disable Clear key from exiting block mode*	1E,25
Disable send to current position*	1E,21
Enable Clear key to exit from block mode*	1E,24
Enable send to current position*	1E,20
Enter block mode	1E,12,61
Exit block mode	1E,12,62
Start block mode send	1E,12,44
Carriage return (CR) delimiter	
Disable sending CR delimiter	1E,12,5A
Enable sending CR delimiter	1E,05
X-off/X-on	
X-off (large CYBER mode)	13
X-off (large CYBER mode)	1E,13
X-on (large CYBER mode)	11
X-on (large CYBER mode)	1E,11

*These commands only apply to terminals having revision 4.0 firmware or above.

TABLE E-1. LIST OF CYBER MODE COMMANDS BY FUNCTION (CONTD)

FUNCTION	HEX CODE
Cursor control	
Backspace	
Backspace (large CYBER mode)	1F
Backspace	1E,19
Carriage return (CR)	
Carriage return	0D
Disable automatic carriage return*	1E,27
Enable automatic carriage return*	1E,26
Cursor address	
Read cursor address	05
Write cursor address (large CYBER mode)	02
Write cursor address (small CYBER mode)	10
Down	
Cursor down (large CYBER mode)	0A
Cursor down	1A
Cursor down	1E,1A
Home	
Cursor home (large CYBER mode)	19
Cursor home (small CYBER mode)	08
Cursor home	1E,08
Left	
Cursor left (large CYBER mode)	08
Cursor left (small CYBER mode)	19
New line (small CYBER mode)	0A
Tab	
Back tab	1E,0B
Clear all tabs	1E,12,59
Clear column tab	1E,12,58
Disable automatic tabbing*	1E,23
Enable automatic tabbing*	1E,22
Tab (large CYBER mode)	09
Tab (small CYBER mode)	1E,04
Set column tab	1E,12,57
Skip	
Cursor skip	18
Cursor skip	1E,18

*These commands only apply to terminals having revision 4.0 firmware or above.

TABLE E-1. LIST OF CYBER MODE COMMANDS BY FUNCTION (CONTD)

FUNCTION	HEX CODE
Up	
Cursor up	17
Data	
Read data	1E,10
Display characters	
Basic characters	1E,1D
Extended characters	
Enter RAM extended characters*	1E,28,(X)
Extended characters	1E,54,(X)
Exit RAM extended characters*	1E,29
Load RAM extended character generator	1E,53,(W)
Line drawing symbols	1E,1C
PLATO characters	1E,12,54,(X)
Display control	
Clear	
Clear all data	1E,50
Clear high intensity fields	1E,40
Clear low intensity fields	1E,3F
Clear to end of page and extend attributes*	1E,2A
Clear to end of page and extend attributes*	1E,2B
Delete	
Delete character	1E,4E
Delete line	1E,51
Disable display	1E,12,4F
Display format	
Select 24 lines	1E,12,5D
Select 30 lines	1E,12,5E
Set 80-character lines	1E,12,48
Set 132-character lines	1E,12,47
Enable display	1E,12,50

*These commands only apply to terminals having revision 4.0 firmware or above.

TABLE E-1. LIST OF CYBER MODE COMMANDS BY FUNCTION (CONTD)

FUNCTION	HEX CODE
Erase	
Erase	1E,59
Erase	1E,5D
Erase page	0C
Erase to end of line	0B
Insert	
Insert character	1E,4F
Insert line	1E,52
Scroll	
Field scroll down	1E,56
Field scroll up	1E,55
Roll disable (large CYBER mode)	16
Roll disable (small CYBER mode)	13
Roll enable	12
Set scroll field	1E,57,(U),(L)
Indicators	
Turn off indicator	1E,12,66,(N)
Turn on indicator	1E,12,65,(N)
Keyboard control	
Clear all host loaded codes*	1E,2E
Define function or action key code sequence or controlware sequence	1E,09,(V),(W),(X),(Y...),(Z)
Disable host loaded code*	1E,33
Disable keyboard	1E,12,4D
Disable typamatic	1E,12,6A
Enable host loaded code*	1E,32
Enable keyboard	1E,12,4E
Enable typamatic	1E,12,69
Normal numeric pad	1E,12,6C
Shift numeric pad	1E,12,6B
Mode changing	
Clear all host loaded codes*	1E,2E
Disable host loaded code*	1E,33
Enable host loaded code*	1E,32
Enter large CYBER mode	1E,12,42
Enter small CYBER mode	1E,12,41
Execute host loaded controlware	1E,12,70 thru 7F

*These commands only apply to terminals having revision 4.0 firmware or above.

TABLE E-1. LIST OF CYBER MODE COMMANDS BY FUNCTION (CONTD)

FUNCTION	HEX CODE
Exit host down-line load	1E,42
Initiate host down-line load	1E,41
Model report request (large CYBER mode)	1E,43,(n)
Parameters	
Read parameters	1E,0F
Read parameters (small CYBER mode)	1E,13
Store mode parameters in NVM	1E,12,6F
Write new mode parameters	1E,12,56,(Y),(Z)
Printing	
Blind printer	1E,7F
Page print (small CYBER mode)	11
Page print (small CYBER mode)	1E,11
Print form	01
Print form	1E,01
Print I/O	1E,46
Reserved	1E,3C thru 1E,3E
Status	
Read status	1E,14
Test	
Initiate test	1E,16
Touchpanel	
Disable touchpanel	1E,12,51
Enable touchpanel	1E,12,52

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES

COMMAND NAME	ASCII MNEMONIC (SEE NOTES AT END OF TABLE)	HEX CODE (SEE NOTES AT END OF TABLE)	TERMINAL RESPONSE
Print Form	SOH	01	Transfers all nondimmed displayed data to printer from beginning of current line to end of page. Dimmed data is sent as space code (20). Keyboard locks, communications data is received but ignored until end of operation (not lost). Printing may be aborted by actuation of Shift/M REL. Print completion is signaled by terminal transmitting an 06 (ACK) or, if the operation is aborted by actuating SHIFT/M REL, by transmission of an 1E, 15 (RS, NAK) sequence. If there is no printer DTR when the Print Form command is received, an RS, NAK is sent in small CYBER mode. There is no completion response in large CYBER mode.
NOOP Small CYBER Mode	STX	02	No operation.
Write Cursor Address - Large CYBER Mode	STX	02	See Write Cursor Address command, hex code 10.
Enable Blink	ETX	03	Blinks characters whose blink bit is set to 1 (refer to Start Blink command, hex code 0E). Following power up or page erase, blink is automatically enabled.
Disable Blink	EOT	04	Disables character blinking on display page.
Read Cursor Address	ENQ	05	Causes terminal to send cursor address header code (1F) followed by codes containing column and row address. Column position transfers first and is numbered from left to right (00 through 4F) for 80-column mode. In 132-column mode, a 7E code precedes the column position address producing a code sequence of 7E, 00, 00 through 4F for

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Read Cursor Address (Contd)			the first 80 columns 01, 00 through 33 for columns 81 through 132. The next code is line-position numbering from top to bottom (00 through 1D). Row/column addresses may be biased to avoid codes 00 through 1F by enabling address bias parameter selection [refer to the CDC 721-21/31 Owners Manual or the 721 Operator's Guide/Installation Instructions Manual to set parameter (preface lists publication numbers)]. When address biasing is enabled, cursor position 00 equals 20. Addressing continues in normal binary progression through 6F for 80-column mode. The 132-column mode sequence is 7E, 20, 20 through 7E, 21, 53 for columns 0 through 132, respectively. The line position address is 20 through 3D for both 80- and 132-column modes.
NOOP Small CYBER Mode	ACK	06	No operation.
Start Underline Large CYBER Mode	ACK	06	Sets the underline attribute bit to 1.
Alarm	BEL	07	Sounds audible alarm for 250 milliseconds.
Home Small CYBER Mode	BS	08	Moves cursor to the home position selected in parameters.
Cursor Left Large CYBER Mode	BS	08	Moves cursor left one character position.
NOOP Small CYBER Mode	HT	09	No operation.
TAB Large CYBER Mode	HT	09	If protect is not enabled, the cursor advances to the first position following the next low-intensity field or next column tab (whichever comes

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
TAB (Contd) Large CYBER Mode			first). If protect is enabled, cursor advances to the beginning of the next unprotected field or next unprotected column tab (whichever comes first). If neither are present, the cursor moves to the top of page. If top of page is protected, the cursor tabs again. No response is sent.†
New Line Small CYBER Mode	LF	0A	Moves cursor to first character position in next line.
Cursor Down Large CYBER Mode	LF	0A	Moves cursor down one line while remaining in the same position. If on the last line, screen will scroll and cursor will move to first column if roll is enabled; or cursor will move to top line if page is enabled.
EOL (Erase to End of Line)	VT	0B	Erases all unprotected characters from, and including current cursor position to end of current unprotected field or the end of that line. Enters 20 in affected positions. Modified attribute bits for all cleared character positions are cleared in character mode, set in block mode if keyboard input.
EP (Erase Page)	FF	0C	Erases all unprotected characters on screen. Cursor moves to home position. Enters 20 in affected positions. Clears background memory and enables blink if previously disabled. Return to enter normal data (clears enter blink, underscore, reduced intensity, dim, and blank). Modified attribute bits are cleared in character mode, set in block mode.

†Terminals having firmware earlier than revision 4.0 respond slightly different than described. The difference is that if protect is enabled, the cursor advances to the beginning of the next unprotected field or to the next protected/unprotected column tab. If neither are present, the cursor moves to the top of the page regardless of whether that position is protected or not.

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC (SEE NOTES AT END OF TABLE)	HEX CODE (SEE NOTES AT END OF TABLE)	TERMINAL RESPONSE
Carriage Return	CR	0D	Moves cursor to first character position in line that it is on. If the Auto-Line-Feed parameter is selected, a LF is performed.
Start Blink	SO	0E	Sets blink bit to 1. Each succeeding displayed character received is shown blinking on the screen.
Stop Blink	SI	0F	Sets blink bit to 0. Each succeeding displayed character received is steadily illuminated.
Write Cursor Address Small CYBER Mode	DLE	10	Interprets next characters as cursor column and row address. Cursor moves to position defined by addresses. Column address is numbered from left to right (00 through 4F) for 80-column mode. In 132-column mode, a 7E code precedes the column position address, producing a code sequence of 7E, 20, 00 through 4F for columns 0 through 80, and 7E, 21, 00 through 34 for columns 81 through 132. Line position is numbered from top to bottom (00 through 1D). If column position code is greater than 4F in 80-column mode or 01, 34 in 132-column mode, cursor control logic wraps around. Line position operates in a similar manner (e.g., 1F equals 01). Row and column addresses may be biased in same manner as described for Read Cursor Address, hex code 05.
NOOP Large CYBER Mode	DLE	10	No operation.
Page Print Small CYBER Mode	DC1	11	Transfers to printer all displayed data from current line to end of page. Keyboard is locked and received data ignored until end of operation (not lost). Printing may be aborted by pressing Shift/M REL. Print completion is signaled by terminal transmitting

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Page Print (Contd) Small CYBER Mode			an 06 or, if the operation was aborted, by transmitting a RS, NAK (1E, 15) sequence. If printer is not ready when the Page Print command is received, an RS, NAK is returned.
X-On Large CYBER Mode	DC1	11	Enables transmission to the host or initiates continuation of suspended transmission from the host (refer to X-Off command, hex code 13). For further information, refer to X-Off/X-On in section 4 of the appropriate Hardware Reference Manual for your terminal (preface lists publication numbers).
Roll Enable	DC2	12	Enables roll mode; screen scrolls up one line each time cursor overflows bottom line or, if a new line code is received when cursor is on bottom line, cursor moves to first character position on bottom line. Bottom line clears; top line is lost. Powering on terminal enables scroll feature.
Roll Disable Small CYBER Mode	DC3	13	Enables page mode; moves cursor to home position when new line code is received and cursor is on bottom line.
X-Off Large CYBER Mode	DC3	13	Causes the terminal to temporarily halt transmission to the host until the X-On command (hex code 11) is received. When sent to the host, means data cannot be acted upon. For further information, refer to X-Off/X-On in section 4 of the appropriate Hardware Reference Manual for your terminal (preface lists publication numbers).
Start Underscore Small CYBER Mode	DC4	14	Sets underscore bit to 1. Each succeeding displayed character received is underlined on the screen.
NOOP Large CYBER Mode	DC4	14	No operation.

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
End Underscore	NAK	15	Sets underscore bit to 0. Each succeeding displayed character received is shown without underlining.
NOOP Small CYBER Mode	SYN	16	No operation.
Roll Disable Large CYBER Mode	SYN	16	Enables page mode; moves cursor to home position when new line code is received and cursor is on bottom line.
Cursor Up	ETB	17	Moves cursor up one line while remaining in same column (character) position. Stored data is not affected.
Skip	CAN	18	Moves cursor right one character position without affecting stored data. On terminals with revision 4.0 firmware, a tab is performed if the new position is protected and autotab is enabled (refer to Enable Automatic Tabbing, hex code 1E,22).
Cursor Left Small CYBER Mode	EM	19	Moves cursor left one character position without affecting stored data. On terminals with revision 4.0 firmware, the cursor moves backward to the first unprotected position if the new position is protected and autotab is enabled (refer to Enable Automatic Tabbing, hex code 1E,22).
Home Large CYBER Mode	EM	19	Moves cursor to the home position selected in parameters.
Cursor Down	SUB	1A	Moves cursor down one line while remaining in same column (character) position. If cursor is on the last line, it will wrap around to the top. Stored data is not affected.
NOOP	ESC	1B	No operation.

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Start Dim	FS	1C	Sets dim bit to 1. Each succeeding displayed character received is dimmed on the screen.
End Dim	GS	1D	Sets dim bit to 0. Each succeeding displayed character received is displayed at full intensity on the screen.
NOOP Small CYBER Mode	US	1F	No operation.
Backspace Large CYBER Mode	US	1F	Moves cursor left one position and clears the data. Protected data is not cleared. On terminals with revision 4.0 firmware, the cursor moves backward to the first unprotected position if the new position is protected and auto-tab is enabled (refer to Enable Automatic Tabbing, hex code 1E,22).
Print Form	RS, SOH	1E, 01	See Print Form command, hex code 01.
Tab Small CYBER Mode	RS, EOT	1E, 04	If protect is not active, the cursor advances to the first position following the next low-intensity field or next column tab whichever comes first). If neither are present, the cursor moves to top of page. If protect is active, the cursor advances to the next unprotected field or unprotected column tab (whichever comes first). If neither are present, the cursor moves to top of page. If top of page is protected, the cursor tabs again. Completion response is identical to Read Status response (refer to hex code 1E,14).†

†Terminals having firmware earlier than revision 4.0 respond differently than described. The differences are that if protect is active, the cursor advances to the beginning of the next unprotected field or to the next protected/unprotected column tab. If neither are present, the cursor moves to the top of page regardless of whether that position is protected or not.

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC (SEE NOTES AT END OF TABLE)	HEX CODE (SEE NOTES AT END OF TABLE)	TERMINAL RESPONSE
NOOP Large CYBER Mode	RS, EOT	1E, 04	No operation.
Enable CR Delimiter	RS, ENQ	1E, 05	Causes a CR delimiter (0D) to be added to certain responses from the terminal. Refer to the note at the end of this table.
Home	RS, BS	1E, 08	See Home commands, hex codes 08 or 19.
Define Function or Action Key Code Sequence or Controlware Sequence	RS, HT, (V), (W) (X), (Y...), (Z)	1E, 09, (V), (W), (X) (Y...), (Z)	Allows special code sequences or controlware sequences to be generated by certain keys. For further information, refer to Host-Specified Code Sequence/ Controlware in section 4 of the appropriate Hardware Reference Manual for your terminal (preface lists publication numbers).
Back Tab	RS, VT	1E, 0B	Moves cursor backward as follows: • With protect not enabled, to the first position following a low-intensity field or column tab, whichever is encountered first. If neither is encountered, the cursor moves to the upper-left position. • With protect enabled, to the beginning of the current unprotected field or to the preceding unprotected column tab, whichever is encountered first. If neither is encountered, the cursor moves to the upper-left position. If the upper-left position is protected, the cursor does another backtab.†

†Terminals having firmware earlier than revision 4.0 respond differently than described. The differences are that if protect is enabled, the cursor moves to the beginning of the current unprotected field or to the preceding protected/ unprotected column tab. If neither are present, the cursor moves to the upper-left position regardless of whether that position is protected or not.

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Back Tab (Contd)			After cursor movement is completed, a response identical to the Read Status response is sent to the host in small CYBER mode (refer to Read Status command, hex code 1E, 14). No response is sent in large CYBER mode.
Read Attribute	RS, SO (X), (Y)	1E, 0E (X) (Y)	Causes terminal to respond with two characters that indicate the attributes of character at cursor position. Cursor is not advanced; stored data is not affected. Data word one bit significance is 2^0 - internal program use; 2^1 - underscore bit; 2^2 - blink bit; 2^3 - reduced-intensity bit; 2^4 and 2^5 are set to a 1. If bit 2^6 is a 1, the character displayed at the current cursor position is not an alphanumeric or control character, but is a line-drawing symbol, a PLATO symbol, or a user-loaded extended character [line-drawing and PLATO symbols are shown in tables A-2 and A-3; extended characters are described under Load RAM Extended Character Generator command, hex code 1E, 53, (W), (X), (Y), (Z)]. Which type of special symbol is being displayed is determined by issuing a Read Data command (hex code 1E, 10) to read the code of the symbol. A code of 20 through 3F indicates a line-drawing symbol; a code of 40 through 7F indicates a PLATO symbol or an extended character, whichever had been in use (PLATO symbols and extended characters cannot be shown on the screen simultaneously). Data word two bit significance is: 2^0- modified bit; 2^1 - protect bit; 2^2 - blank bit; 2^3 - inverse bit; 2^4 and 2^5 are set to a 1; 2^6 is set to a 0.

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Read Parameter	RS, SI	1E, 0F	Causes terminal to transmit settings of terminal operating parameters. Settings are sent out in data words preceded by sequence 02, 06, 25 and, if in small CYBER mode, terminated with a Read Status response (see Read Status command, hex code 1E, 14). No response is sent in large CYBER mode. For the format of the data words sent, refer to Read Parameter Data Word Format table in section 4 of the appropriate Hardware Reference Manual for your terminal (preface lists publication numbers).
Read Data	RS, DLE	1E, 10	Causes data word stored in memory at cursor position to be transmitted. Cursor is not advanced. Code contains seven data bits. Determining if the code represents an alphanumeric character, line drawing, extended character, or control code requires that the attribute character be read. Refer to Read Attribute command, hex code 1E, 0E.
Page Print Small CYBER Mode	RS, DC1	1E, 11	See Page Print command, hex code 11.
X-ON Large CYBER Mode	RS, DC1	1E, 11	This is not a normal sequence. With DC1 following RS, the X-On function will be performed (see X-On command, hex code 11), but the next code received will be acted upon as if an RS (1E) preceded it.
Read Parameter Small CYBER Mode	RS, DC3	1E, 13	See Read Parameter command, hex code 1E, 0F.

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
X-Off Large CYBER Mode	RS, DC3	1E, 13	This is not a normal sequence. With DC3 following RS, the X-Off function will be performed (see X-Off Command, hex code 13), but the next code received will be acted upon as if an RS (1E) preceded it.
Read Status	RS, DC4	1E, 14	Causes terminal to transmit 02, 06, 06 (STX, ACK, ACK) if all preceding self-test operations were completed successfully. The response 02, 06, 15 (STX, ACK, NAK) is transmitted if any self-test failed.
Initiate Test	RS, SYN	1E, 16	Causes terminal to perform a self-test. This command is not to be used because it will cause CP/M to lose control of the terminal.
Skip	RS, CAN	1E, 18	See Skip command, hex code 18.
Backspace	RS, EM	1E, 19	See Cursor-Left command, hex code 19.
Cursor Down	RS, SUB	1E, 1A	See Cursor-Down command, hex code 1A.
Line Drawing	RS, FS	1E, 1C	Causes terminal to interpret any following data words received from 20 to 3F as line-drawing characters. Refer to table A-2 for codes.
Basic Character	RS, GS	1E, 1D	Causes terminal to interpret received data as normal characters.
Enable Send to Current Position - for Terminals with 4.0 Firmware only	RS, SP	1E, 20	Causes the terminal to send all unprotected data from the top of screen to, but not including, the current cursor position when in block mode.
Disable Send to Current Position - for Terminals with 4.0 Firmware only	RS, !	1E, 21	Causes the terminal to send all unprotected data on the screen when in block mode.

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Enable Automatic Tabbing - for Terminals with 4.0 Firmware only	RS, "	1E, 22	Causes the cursor to automatically tab out of protected fields, except for cursor-up and cursor-down functions.
Disable Automatic Tabbing - for Terminals with 4.0 Firmware only	RS, #	1E, 23	Allows the cursor to remain in protected fields. This is the default condition.
Enable Clear Key to Exit Block Mode - For Terminals with 4.0 Firmware only	RS, $	1E, 24	Causes the clear page function (Shift and P/CLEAR/EOL keys) to disable protect, clear the screen, and exit from block mode.
Disable Clear Key - for Terminals with 4.0 Firmware only	RS, %	1E, 25	Causes the clear page function to clear just unprotected data from the screen. This is the default condition.
Enable Automatic Carriage Return - for Terminals with 4.0 Firmware only	RS, &	1E, 26	Causes the cursor to automatically move to the next line after the last position of a line is filled. This is the default condition.
Disable Automatic Carriage Return - for Terminals with 4.0 Firmware only	RS, '	1E, 27	Causes the cursor to remain in the last position of a line until a control code is received that moves it to the next line.
Enter RAM Extended Character Mode - for Terminals with 4.0 Firmware only	RS, (, (X)...	1E, 28 (X)	Causes the terminal to display all codes between 40 through 7F from the RAM character generator. Codes outside that range perform the normal operation.

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Exit RAM Extended Character Mode - for Terminals with 4.0 Firmware only	RS,)	1E, 29	Causes the terminal to display normal characters for all codes between 40 through 7F. This is the default condition.
Clear to EOL Extend Attribute - for Terminals with 4.0 Firmware only	RS, *	1E, 2A	Clears to the end of line just like the EOL command (hex code 0B) except the background memory is set to the current attributes.
Clear to EP Extend Attribute - for Terminals with 4.0 Firmware only	RS, +	1E, 2B	Clears to end of page just like the Clear All Data command (hex code 1E,50) except the background memory is set to the current attributes.
Use Old Attribute Enable - for Terminals with 4.0 Firmware only	RS, ,	1E, 2C	Enables the reusing of the old attribute during host and keyboard entries and during all clear operations.
Use Old Attribute Disabled - for Terminals with 4.0 Firmware only	RS, -	1E, 2D	Disables the reusing of the old attribute. As data is entered from the host and keyboard, the new attribute is stored. During clear operations, the attribute is cleared. This is the default condition.
Clear All Host Loaded Codes - for Terminals with 4.0 Firmware only	RS, .	1E, 2E	Clears all previously loaded codes/controlware. This includes host loaded subroutines, host specified codes for keys, host loaded controlware for loaded keys, validation controlware, prologue loaded codes and host functions.

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Enable Code Bias - for Terminals with 4.0 Firmware only	RS, 0	1E, 30	Causes terminal to add and subtract 20 hex when sending and receiving X/Y-positioning information or set-scroll field information. The default condition is set in mode parameter F5, position 2.
Disable Code Bias - for Terminals with 4.0 Firmware only	RS, 1	1E, 31	Causes terminal to accept X/Y-positioning information and set-scroll field information without doing the 20-hex biasing.
Enable Host Loaded Code - for Terminals with 4.0 Firmware only	RS, 2	1E, 32	Causes all host loaded codes/controlware to be used.
Disable Host Loaded Code - for Terminals with 4.0 Firmware only	RS, 3	1E, 33	Causes all host loaded codes/controlware to be ignored. All host loaded keys, subroutines, and controlware will remain loaded, but not used.
Reserved	RS, <	1E, 3C	Hebrew usage.
Reserved	RS, =	1E, 3D	Hebrew usage.
Reserved	RS, 7	1E, 3E	Hebrew usage.
Clear Fields: • Low Intensity • High Intensity • All Data	 RS, ? RS, @ RS, P	 1E, 3F 1E, 40 1E, 50	Causes terminal to clear all unprotected data from cursor position to end of page or unprotected high- or low-intensity data as selected. No responses to I/O commands are made during the operation. Modified attribute bits for all cleared character positions are cleared. A Read-Status response is transmitted to indicate the operation is complete (see Read-Status command, hex code 1E, 14). No response is sent in large CYBER mode.

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Initiate Host DLL	RS, A	1E, 41	Initiates a host-specified downline load. This command is not to be used because it will cause CP/M to lose control of the terminal.
Exit Host DLL	RS, B	1E, 42	This command and the Initiate Host DLL command (hex code 1E, 41) are not applicable for CYBER mode operations under CP/M control.
Model Report Request Large CYBER Mode	RS, C,(n)	1E, 43, (n)	Terminal transmits a report message containing model and configuration information. In addition, if the n code of the command indicates a request for parameter data, the terminal includes the requested data in the message. That parameter data may be the operator parameters stored in RAM or the installation parameters stored in nonvolatile memory. For further details, refer to Model Report Request in section 4 of the appropriate Hardware Reference Manual for your terminal (preface lists publication numbers).
NOOP Small CYBER Mode	RS, C	1E, 43	No operation.
Start Inverse	RS, D	1E, 44	Sets inverse bit to 1. Each succeeding received character is displayed in inverse video (dark characters on light background).
End Inverse	RS, E	1E, 45	Clears inverse bit to 0. Each succeeding received character is displayed in normal video (light characters on dark background).
Print I/O	RS, F	1E, 46	Causes terminal to direct all data to printer interface. Completion response is identical to the Read-Status response (see Read-Status command, hex code 1E, 14). No response is sent in large CYBER mode.

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Set All Protect Bits	RS, G	1E, 47	Disables character protect feature and sets the protect bit in the attribute code for every character position. The protect bits being set have no affect on operations until a Protect Enable command (hex code 1E, 12, 4B) is received. Note: If the Protect Enable command is issued with all protect bits set, the entire screen will be protected and no keyboard input will be possible.
Delete Character	RS, N	1E, 4E	Deletes the character at the current cursor position. All characters right of the cursor are shifted left one position. If a Protect Enable command (hex code 1E, 12, 4B) is in effect, the shift occurs only up to the protected data and the old attribute is reused for the new position.
Insert Character	RS, O	1E, 4F	Inserts a space character at the current cursor position. The character at the cursor position and the characters right of the cursor are shifted right one position. If a Protect Enable command (hex code 1E, 12, 4B) is in effect, the shift occurs only up to the protected data and the old attribute is reused for the new position.
Clear All Data	RS, P	1E, 50	See preceding description in this table for Clear Fields - all data.
Delete Line	RS, Q	1E, 51	Causes all unprotected lines below cursor and within the logical page or unprotected area limits to move up one position; the current line is lost and the bottom line is cleared. The terminal will not respond to I/O commands during the operation. Completion response is identical to the Read-Status response (see Read-Status command, hex code 1E, 14). No response is sent in large CYBER mode. Modified attribute bits for all cleared character positions are cleared.

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC (SEE NOTES AT END OF TABLE)	HEX CODE (SEE NOTES AT END OF TABLE)	TERMINAL RESPONSE
Insert Line	RS, R	1E, 52	Causes unprotected data on current line to move one line down; bottom line within logical page or unprotected area is lost and the current line is cleared. The terminal will not respond to I/O commands during the operation. The response sent after completion is identical to the Read Status response (see Read-Status command, hex code 1E, 14). Modified attribute bits for all cleared character positions are cleared.
Load RAM Extended Character Generator†	RS,S,(W), (X), (Y), (Z)	1E, 53,(W) (X), (Y), (Z)	Causes the terminal to interpret the characters following the RS, S, command [(W) (X), (Y), (Z)] as symbol data for the RAM character generator. This character generator generates a user-loaded symbol on the screen when directed by an Extended-Character command, hex code 1E, 54, (X), or if the terminal has 4.0 revision firmware, by an Enter RAM-Extended-Character Mode command, hex code 1E, 28. Each symbol loaded requires an identifying code in word 1 (W), a start-scan count in word 2 (X), thirty two codes that define the dot pattern in a group of word 3s (Y), and a termination code in word 4 (Z). These words are formatted as follows: • Word 1 (W) - Character Code. This identifying code must be between 40 through 7F. Codes outside this range cause an RS NAK to be sent to the host when the termination code is received.

†Resequence per dual hexadecimal code.

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Load RAM Extended Character Generator† (Contd)			• Word 2 (X) - Start Scan Count. Symbols are formed in a 8-dot wide by 16-dot high matrix. Bits 2^0 through 2^3 of this word specify the number of dot rows down in which the first illuminated dot is to appear within the matrix. The top row in the matrix is row 0 and the bottom row is row F. The other bits in word 2 must have bit 2^4 set to 0, bit 2^5 set to 1, and bit 2^6 set to 0. • Word 3 (Y) - Dot Pattern. The thirty two words of dot patterns are grouped in pairs. Each pair corresponds with one of the 16 rows in the matrix. The pair for row 0 is sent first and the pair for row F is sent last. The four least significant bits (bits 2^0 through 2^3) of the first word of each pair represent the dots in the right half of the row and the four least significant bits of the second word represent the dots in the left half. Within each half, bit 2^0 represents the leftmost of the four dots and bit 2^3 represents the rightmost of the four dots. Further details on forming the dot pattern are in section 4 of the 721 Enhanced Display Terminal Hardware Reference Manual under Load RAM Extended Character Generator (preface lists publication number). If a dot is to be lit, the corresponding bit must be a 1. Bit 2^4 in the first word of each pair must be a 0, bit 2^5 must be a 1, and bit 2^6 must be a 0. Bit 2^6 in the second word of each pair must be a 1.

†Resequence per dual hexadecimal code.

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Load RAM Extended Character Generator† (Contd)			• Word 4 (Z) - Termination Code CR (0D). If no errors were received, the terminal responds with an ACK (06); otherwise, an RS, NAK (1E, 15). No response is sent in large CYBER mode.
Extended Character†	RS, T,(X)	1E, 54,(X)	Causes terminal to interpret (X) as a character to be displayed from the RAM character generator [see Load Extended Character Generator command, hex code 1E, 53, (W), (X), (Y), (Z)]. X must be in the range of 40 through 7F. Codes outside of this range cause the parity-error symbol (■) to be displayed. Restriction: Extended characters cannot be simultaneously displayed with PLATO symbols.
Field Scroll Up	RS, U	1E, 55	Causes each display line in scroll field to move up one position [see Set Scroll Field command, hex code 1E, 57, (U), (L)]. The uppermost line in the scroll field is lost and the bottom line is cleared. There is no response to I/O commands during the operation. The response at completion is identical to the Read Status response (see Read Status command, hex code 1E, 14). No response is sent in large CYBER mode.
Field Scroll Down	RS, V	1E, 56	Causes each line in scroll field to move down one position [see Set Scroll Field command, hex code 1E, 57, (U), (L)]. The lowest line in the scroll field is lost and the uppermost line is cleared. There is no response to I/O commands during the operation. The response at completion is identical to the Read-Status response (see Read-Status command, hex code 1E, 14). No response is sent in large CYBER mode.

†Resequence per dual hexadecimal code.

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC (SEE NOTES AT END OF TABLE)	HEX CODE (SEE NOTES AT END OF TABLE)	TERMINAL RESPONSE
Set Scroll Field	RS,W,(U), (L)	1E, 57, (U), (L)	Establishes a scroll field on the screen. The word U in the command specifies the upper display line of the field and the word L specifies the lower line of the field. Line numbers outside of 1 through 30 ($1E_{16}$) automatically cause lines 1 and 30 to be selected. If address biasing is in effect (not selected in factory-set parameters), the acceptable line numbers are 33 (21_{16}) through 62 ($3E_{16}$). Note: This works in conjunction with the Field Scroll Up and Down commands, (hex codes 1E, 55 and 1E, 56).
Erase	RS, Y	1E, 59	See Erase command, hex code 1E, 5D.
Erase	RS,]	1E, 5D	All character locations in the current unprotected field are cleared to spaces and the cursor is moved to the beginning of the unprotected field.
Blind Printer	RS, DEL	1E, 7F	Causes terminal to stop transferring received and transmitted data to printer. This is also governed by a mode installation parameter. RS (1E), DEL (7F) is transmitted to printer. The completion response is identical to the Read-Status response (see Read-Status command, hex code 1E, 14). No response is sent in large CYBER mode.
Enter Small CYBER Mode	RS, DC2, A	1E, 12, 41	Enter small CYBER mode of operation.
Enter Large CYBER Mode	RS, DC2, B	1E,12, 42	Enter large CYBER mode of operation.

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Start Block Mode Send	RS, DC2, D	1E, 12, 44	If the terminal has revision 4.0 firmware and an Enable Send-to-Current Position command (hex code 1E,20) has been received, the terminal transmits the unprotected data characters from the top of page up to, but not including, the current cursor position. If the terminal has earlier firmware than revision 4.0 or an Enable Send-to-Current Position command has not been received, the terminal instead transmits all the unprotected data characters on the screen. A CR (0D) delimiter indicates the end of the operation. For the format of the transmission, refer to Block Mode Operation in section 4 of the appropriate Hardware Reference Manual for your terminal (preface lists publication numbers).
Set 132 Character Line	RS, DC2, G	1E, 12, 47	Terminal displays 132 characters/line. If the initial line length is 80 characters per line, the display is cleared and the cursor is moved to home position.
Set 80 Character Line	RS, DC2, H	1E, 12, 48	Terminal displays 80 characters/line. If the initial line length is 132 characters per line, the display is cleared and the cursor is moved to home position.
Start Protect	RS, DC2, I	1E, 12, 49	Sets protect bit in attribute code for each succeeding character received.
Clear Protect	RS, DC2, J	1E, 12, 4A	Clears protect bit in attribute code for each succeeding character received.
Enable Protect	RS, DC2, K	1E, 12, 4B	Characters with the protect bit set in their attribute codes are protected from operator action. Only the host can make changes to the protected data.

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC (SEE NOTES AT END OF TABLE)	HEX CODE (SEE NOTES AT END OF TABLE)	TERMINAL RESPONSE
Disable Protect	RS, DC2, L	1E, 12, 4C	Disables protection of data. Operator input in previously protected character positions is allowed. Protect bits for characters are unaffected.
Disable Keyboard	RS, DC2, M	1E, 12, 4D	Disables keyboard entries until an Enable Keyboard command (hex code 1E, 12, 4E) is received or RESET switch is pressed. (Pressing RESET switch will cause CP/M to lose control of the terminal.)
Enable Keyboard	RS, DC2, N	1E, 12, 4E	Enables keyboard entries.
Disable Display	RS, DC2, O	1E, 12, 4F	Disables changes to the display memory. All incoming data is ignored until a Enable Display command (hex code 1E, 12, 50) is received.
Enable Display	RS, DC2, P	1E, 12, 50	Enables normal display operation.
Disable Touchpanel	RS, DC2, Q	1E, 12, 51	Disables input from the touchpanel until an Enable Touchpanel command (hex code 1E, 12, 52) is received or RESET switch is pressed. (Pressing RESET switch will cause CP/M to lose control of the terminal.)
Enable Touchpanel	RS, DC2, R	1E, 12, 52	Enables input from the touchpanel. For further information, refer to Touchpanel Operation in section 4 of the appropriate Hardware Reference Manual for your terminal (preface lists publication numbers).
Mode Select	RS, DC2, S, (n)	1E, 12, 53, (n)	This command is not applicable for CYBER mode operations under CP/M control.

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC (SEE NOTES AT END OF TABLE)	HEX CODE (SEE NOTES AT END OF TABLE)	TERMINAL RESPONSE
PLATO Character	RS, DC2, T (X)	1E, 12, 54, (X)	Causes terminal to interpret (X) as PLATO character to be displayed. X must be in the range of 40 through 7F (see table A-3). Codes outside this range will cause the display of a parity-error symbol (█). Restriction: PLATO characters and extended characters [see Extended Character command, hex code 1E, 54, (X)] cannot be displayed simultaneously.
Select Bi-directional Port N	RS, DC2, U, (N)	1E, 12, 55, (N)	This command is only applicable if an optional dual asynchronous-interface board is installed in the terminal. The command causes data from the host to be transferred to the specified port and data from the port to be transferred to the host. The word N a 0 specifies port A (connector J1); N a 1 specifies port B (connector J2). All transferred data is ignored by the terminal and the keyboard is locked to prevent entries. A RS (1E), DC2 (12) code sequence returns the terminal to normal operation. For further information, refer to Host Select Bidirectional Port in section 4 of the appropriate Hardware Reference Manual for your terminal (preface lists publication numbers).
Write New Mode Parameters	RS, DC2, V, (Y), (Z)	1E, 12, 56, (Y), (Z)	Writes parameter changes into RAM. This command is not to be used because it will cause CP/M to lose control of the terminal.
Set Column Tab	RS,DC2,W,	1E, 12, 57	Causes the terminal to set a column tab for the current column
Clear Column Tab	RS,DC2,X,	1E, 12, 58	Causes the terminal to clear the column tab position of current column.
Clear All Tabs	RS,DC2,Y	1E, 12, 59	Clears all column tabs.

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Disable CR Delimiter	RS, DC2,Z	1E, 12, 5A	Disables transmission of the CR (0D) delimiter in multiple code and controlware sequences.
Start Blank	RS,DC2,[	1E, 12, 5B	Sets the blank attribute bit. Each succeeding received character is stored but not displayed.
End Blank	RS,DC2,\	1E, 12, 5C	Clears the blank attribute bit for each succeeding character received.
Select 24 lines	RS,DC2,]	1E, 12, 5D	Terminal uses 24 lines for displaying characters. If 30 lines were being used, the display is cleared and the cursor is moved to home position.
Select 30 lines	RS,DC2,^	1E, 12, 5E	Terminal uses 30 lines for displaying characters. If 24 lines were being used, the display is cleared and the cursor is moved to home position.
Enter Block Mode	RS, DC2,a	1E, 12, 61	Terminal stops character-by-character transmission of keyboard entries. Instead, entries are stored for a block transmission. The only exceptions to this are the uppercase use of the L/INSRT/C key and the L/DLETE/C key. The codes for these keys are separately transmitted and their insert/delete line function is not performed until the transmitted code is echoed or sent back by the host. Terminals with revision 4.0 firmware can do a block transmission of the unprotected data from the top of the page up to, but not including, the current cursor position. This is enabled by an Enable Send-to-Current Position command (hex code 1E,20). If this command is not received or is cancelled by a Disable Send-to-Current Position command (hex code 1E,21), the block transmissions include all unprotected data on the screen. That type of block transmission is the only type available on terminals that have firmware earlier than revision 4.0.

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC (SEE NOTES AT END OF TABLE)	HEX CODE (SEE NOTES AT END OF TABLE)	TERMINAL RESPONSE
Enter Block Mode (Contd)			A block transmission is started by pressing any one of the F1 through F15 function keys or one of the special keys HELP, EDIT, BACK, DATA, +/DOWN, -/UP, X/FWD, or -/BKW (using the -/BKW key to start a block transmission only applies to terminals having earlier firmware than revision 4.0). A Start Block Mode Send command (hex code 1E, 12, 44) also starts a block transmission. For the format of the transmission, refer to Block Mode Operation in section 4 of the appropriate Hardware Reference Manual for your terminal (preface lists publication numbers).
Exit Block Mode	RS, DC2, b	1E, 12, 62	Terminal returns to transmitting keyboard entries character by character.
Turn On Indicator	RS, DC2, e, (N)	1E, 12, 65, (N)	Turns on the indicator specified by (N). N = 30: ALERT indicator N = 31: PROGRAM indicator 1 N = 32: PROGRAM indicator 2 N = 33: PROGRAM indicator 3 N = 34: MESSAGE indicator
Turn Off Indicator	RS, DC2, f, (N)	1E, 12, 66, (N)	Turns off the indicator specified by (N). N = 30: ALERT indicator N = 31: PROGRAM indicator 1 N = 32: PROGRAM indicator 2 N = 33: PROGRAM indicator 3 N = 34: MESSAGE indicator
Driver Request	RS, DC2, h	1E, 12, 68	This command is not applicable for CYBER mode operations under CP/M control.
Enable Typamatic	RS, DC2, i	1E, 12, 69	Data-entry and control keys begin repeating after being pressed for longer than a second (refer to table B-1 for affected keys).

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		
Disable Typamatic	RS, DC2, j	1E, 12, 6A	Disables data-entry and control keys from repeating (refer to table B-1 for affected keys). This command ignores Margin + Bell at the end of line.
Shift Numeric Pad	RS, DC2, k	1E, 12, 6B	Causes the numeric keypad to operate as if the shift key were always in use.
Normal Numeric Pad	RS, DC2, l	1E, 12, 6C	Returns the numeric keypad to normal operation.
Start Validation	RS, DC2, m	1E, 12, 6D	Sets validation attribute bit to 1 for each succeeding character received from the host. For further information, refer to Validation Bits and Host Specified Code Sequence/Controlware in section 4 of the appropriate Hardware Reference Manual for your terminal (preface lists publication numbers).
End Validation	RS, DC2, n	1E, 12, 6E	Sets validation attribute bit to 0 for each succeeding character received from the host. For further information, refer to Validation Bits and Host Specified Code Sequence/Controlware in section 4 of the appropriate Hardware Reference Manual for your terminal (preface lists publication numbers).
Store Mode Parameters in NVM	RS, DC2, o	1E, 12, 6F	Writes parameters stored in RAM into nonvolatile memory. This command is not to be used because it will cause CP/M to lose control of the terminal.
Host Execute Loaded Controlware	RS, DC2, p thru DEL	1E, 12, 70 thru 7F	If the host has loaded controlware for the 70 through 7F code in the command, the terminal begins executing the instruction stored at the assigned starting memory address. If no controlware has been loaded for the code, no operation is performed. For further information, refer to Host-Specified Code Sequence/Controlware in section 4 of the appropriate Hardware Reference Manual for your terminal (preface lists publication numbers).

TABLE E-2. CYBER MODE COMMANDS AND RESPONSES (CONTD)

COMMAND NAME	ASCII MNEMONIC	HEX CODE	TERMINAL RESPONSE
	SEE NOTES AT END OF TABLE		

Notes:

Codes are listed in hexadecimal order. Unlisted codes cause no operation except codes 20 through 7F. Those codes display the characters or symbols of the code set which is in effect. The normal code set is the language character set established in block F7 of terminal installation parameters (appendix C outlines the parameters, and tables A-1 and A-4 through A-9 in appendix A contain the code sets for the various languages). Line-drawing symbols, PLATO symbols, or extended characters become the effective code set through the Line-Drawing command (hex code 1E, 1C); the PLATO-Character command [hex code 1E, 12, 54, (X)]; or the Extended-Character command [hex code 1E, 54, (X)]. Those commands are defined in this table and the code sets for the line-drawing and PLATO symbols are in tables A-2 and A-3 in appendix A. The code set and symbols for extended characters are user-defined.

All RS, ACK and RS, NAK responses from the terminal are followed by a CR(0D) delimiter if an Enable CR Delimiter command (hex code 1E,05) is in effect.

RIGID-DISK INTERFACE COMMANDS F

This appendix describes the interface commands issued from the terminal to the rigid-disk adapter. There are nine commands:

- Read from disk - Read n, h, track, sector + m
- Write to disk - Write n, h, track, sector + m
- Format disk - Format n, h, track, sector
- Select disk - Select, t, n, f, dev. code
- Execute from loc a in RAM - Xcute loc a
- Execute interface diagnostic - Xcute diag
- Read RAM/PROM - R RAM k, + words
- Write RAM - W RAM k, + words
- Abort present activity - Abort, retry

READ FROM DISK

The Read-from-disk command is a two-word command with the following format:

Read n, h, track, sector + m

0	0	n	0	0	0	0	Track
0	1	2 3	4	5	6	7	8 15

0	0	n	m	h	Sector
0	1	2 3	4 7	8 10	11 15

Where:

n indicates drive number (should be 00 for CP/M operation)

h indicates head number

m indicates number of following sectors (- 1) to be transferred from the disk (not applicable in this operation)

sector defines first sector to be read

At the end of an operation the rigid-disk adapter sends a status word to the terminal. The format of the status word is:

0	WP	BS	F	N	R	W	S	Step	
0	1	2	3	4	5	6	7	8	15

Where:

S if set denotes fatal seek during operation.

W if set denotes bad write operation (not used in this operation).

R if set denotes bad read operation.

WP if set denotes write-protect flag set (not applicable in this operation).

N if set denotes there is no device installed.

BS denotes bad-sector mark detection.

F if set denotes error during format operation (not used in this operation).

If an error occurs during the read operation, the step field is set to m + 1. If there are no errors, the step field is cleared to all zeros.

WRITE TO DISK

The Write-to-disk command is a two-word command with the following format:

Write n, h, track sector + m

0	0	n	0	0	0	1	Track
0	1	2 3	4	5	6	7	8 15

0	0	n	m	h	Sector
0	1	2 3	4 7	8 10	11 15

Where:

n indicates drive number (should be 00 for this operation)

h indicates head number

m indicates how many following sectors will be transferred to the disk (should be 00 for this operation)

track and sector define initial track and sector

During a multiple-sector-write operation, the rigid-disk adapter issues a seek-to-next-sector whenever the command requires that data be written to sectors on the next track. A maximum of 16 sectors can be written to during one operation. At the end of the write operation, the rigid-disk adapter sends a status word to the terminal. The terminal driver changes mode from output to input in order to receive the status word. The format of the status word is shown below:

0	WP	BS	F	N	R	W	S	Step
0	1	2	3	4	5	6	7	8 15

Where:

S if set denotes fatal seek during operation

W if set denotes bad write operation

WP if set indicates write-protect flag set

BS if set denotes bad-sector mark set during a format of this sector

R if set denotes bad read operation (not applicable during this operation)

N if set denotes there is no device installed

F if set denotes error during format operation (not applicable during this operation)

If an error occurs during the write operation, the step field is set to m + 1. If an error occurs during multiple-sector write, data transfer to the disk is halted, but the rigid-disk adapter continues to retrieve data from the terminal until all data sectors are transferred from the terminal to the rigid-disk adapter.

FORMAT DISK

The Format-disk command is a two-word command with the following format:

Format n, h, track, sector,

0	0	n	0	0	1	0	Track
0	1	2 3	4	5	6	7	8 15

0	0	n	0	0	BS	S	H	Sector
0	1	2 3	4	5	6	7	8 10	11 15

Where:

n indicates drive number (should be 00 for CP/M operation)

h indicates head number

s if set denotes that whole surface must be formatted

BS if set denotes that bad-sector mark will be set in the formatted sector. Marking the whole surface with the bad-sector flag is not allowed and results in setting illegal curl bit to one.

The rigid-disk adapter sends a status word to the terminal. The format of the status word is:

0	WP	IC	F	N	R	W	S	Not Used
0	1	2	3	4	5	6	7	8 — 15

Where:

S if set denotes fatal seek during operation

W if set denotes bad write operation (not applicable in this operation)

R if set denotes bad read operation (not applicable in this operation)

N if set denotes there is no device installed

F if set denotes error during format operation

IC denotes illegal command parameter--attempt to mark whole surface with bad-sector flag

WP denotes write-protect switch set

SELECT DISK

The Select-disk command is a one-word command with the following format:

Select t, n, f, device code

0	0	n	1	0	1	0	0	0	Dev. Code
0	1	2 3	4	5	6	7	8	9	10 — 15

Where:

n indicates drive number (should be 00 for this operation)

This command is a multipurpose command which does the following:

1. Selects the disk controller associated with the device code parameter
2. Sets the type of disk in the type field
3. Checks if the device is ready to receive commands and data
4. Checks if the write/protect flag is set

If the device code is specified and does not correspond to 26_8, 36_8, 46_8 or 56_8, the previous device code selection remains unchanged.

At the end of an operation the rigid-disk adapter sends a status word to the terminal. The format of the status word is:

0	0	n	WP	R	t	0	0	Dev. Code
0	1	2 3	4	5	6 7	8	9	10 15

Where:

t describes type of disk

t = 10 12.5-Mb storage
t = 11 25-Mb storage

WP indicates write/protect flag is set

R when set indicates disk drive is ready for operation

This command must be used at the beginning of the session to describe the type of disk, mode, and device code. During a session, this command can select a different device, change the mode, and also orient the system software. For example, when the command Select o, n, o, o is issued and the device code is 26_8 and n = o for a 25 mB rigid disk, the status word is as follows:

0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

0	0	0	0	0	0	1	1	0	0	0	1	0	1	1	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

EXECUTE FROM LOCATION A IN RAM

The Execute-from-location-a-in-RAM command is a two-word command with the following format:

Xcute loc a

0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

0	0	0	0	0	0	Address	
0	1	2	3	4	5	6	15

This command causes the rigid-disk adapter to jump to location $A_6 \ldots A_{15}$ of the interface RAM. The code to be executed must be NEED CORRECT TERM MP100-type object code. This command is useful for writing patches and executing some diagnostic routines.

EXECUTE INTERFACE DIAGNOSTICS

The Execute-interface-diagnostics command is a one-word command with the format:

Xcute diag.

0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

This command invokes the interface diagnostic test stored in the firmware. The self-test includes a RAM test by ROM checksum. Test C checks the I/O devices. After the interface diagnostics routine the rigid-disk adapter presents a status word to the terminal. The format of the status word is:

0	0	N1	N2	N3	N4	P	R	0	0	b1	b2	b3	b4	0	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Where:

R if set indicates RAM error - during Read Write test

P if set indicates some problem with PROM revision number (check sum error detected)

N4 if set indicates no disk controller at device 56_8

b4 if set indicates that there is I/O IC associated with device code 56_8

N3 if set indicates no disk controller at device 46_8

b3 if set indicates that there is I/O IC associated with device code 46_8

N2 if set indicates no disk controller at device 36_8

b2 if set indicates that there is I/O IC associated with device code 36_8

N1 if set indicates no disk controller at device 26_8

b1 if set indicates that there is I/O IC associated with device code 26_8

READ RAM/PROM

The Read-RAM/PROM instruction is a two-word command with the following format:

R RAM k, + words

0	0	0	0	0	1	1	0	Words
0	1	2	3	4	5	6	7	8 15

0	k	0	0	0	Address
0	1	2	3	4	5 15

Where:

5 thru 15 specifies beginning address

k when set allows the user to read ROM memory

when k = 0, RAM is being read

This command allows the system to read a block of RAM memory up to 256 words, starting at the address specified by the second word.

WRITE RAM

The Write-RAM command is a two-word command with the format:

W RAM k, + words

0	0	0	0	1	0	0	0	Words
0	1	2	3	4	5	6	7	8 15

0	0	0	0	0	0	Address
0	1	2	3	4	5	6 15

Where:

6 thru 15 specifies beginning address

The Write-RAM command allows the system to write into the interface RAM. The words field defines how many words are written in the block. This command must be used with caution because writing in block 0 might destroy the interrupt handler, and only a reset from the terminal will bring the I/F under control.

ABORT, RETRY

The Abort, Retry command is a one-word command with the format:

Abort, retry

0	1	2	3	4	5	6	7	8–15
0	0	0	0	1	0	0	1	Retry

The abort instruction causes the rigid-disk adapter to abandon the present task. The rigid-disk adapter issues a universal clear on the I/O channel. If retry is specified and different from 0, the retry counter is set to the value specified in the retry field. At power up, the firmware automatically sets the retry counter to 10.

This appendix contains ASCII conversion information for the graphics option. Table G-1 lists the five character sizes which can be displayed in alpha mode. The table also indicates the displayable screen area for the different size characters.

TABLE G-1. CHARACTER SIZES

SELECTION SEQUENCE	SIZE	TEKTRONIX 4014		CDC					
				401X AREA		TOTAL AREA		CHARACTER SIZE	
		WIDTH	LINES	WIDTH	LINES	WIDTH	LINES	TOTAL	BASE
ESC 7	0	-	-	64	24	64	32	8 x 16	7 x 9
ESC 8	1	74	35	73	39	73	51	7 x 10	5 x 7
ESC 9	2	81	38	85	43	85	56	6 x 9	5 x 7
ESC :	3	121	58	128	55	128	73	4 x 7	3 x 5
ESC ;	4	133	64	128	65	128	85	4 x 6	3 x 5

Note that while there are five character size presentations, there are only three basic character sizes used.

a. Size 0 is the standard PLATO size 0.
b. Sizes 1 and 2, 7 x 10 and 6 x 9 respectively, use the same 5 x 7 base characters.
c. Sizes 3 and 4, 4 x 7 and 4 x 6 respectively, use the same 3 x 5 base characters.

The point-plot, line, and block modes all require sets of coordinates. To specify coordinate positions, the 10-bit X coordinate and the 10-bit Y coordinate must be converted to ASCII characters as indicated in table G-2.

The terminal retains the last High Y, Low Y, and High X addresses when switched to operations that do not require coordinate information. When returned to an operation requiring coordinate information, only Low X must be received by the terminal to reset to its previous coordinates. It is not necessary that all four ASCII characters describing the coordinate be transmitted. The table G-3 shows the coordinate byte transmission requirements. Table G-4 is a coordinate conversion chart.

TABLE G-2. COORDINATE ASCII CODING

BYTE	BIT 7	BIT 6	BITS 5 THROUGH 1*
High Y	0	1	5 MSB of Y Coordinate**
Low Y	1	1	5 LSB of Y Coordinate
High X	0	1	5 MSB of X Coordinate**
Low X	1	0	5 LSB of X Coordinate

*MSB is most significant bit; LSB is least significant bit.
**Since the resolution of this terminal is 512 by 512, the MSB (bit 10) of each coordinate is outside the screen display area. Coordinates exceeding 511 yield unpredictable results. If either scale mode is selected, all X and Y coordinates are scaled down bya factor of two, and the Y coordinate is then biased upward by 122 dots if scaling with Y bias is selected.

TABLE G-2. COORDINATE BYTE TRANSMISSION REQUIREMENTS

BYTES WHICH CHANGE				BYTE TRANSMISSION REQUIRED			
HIGH Y	LOW Y	HIGH X	LOW X	HIGH Y	LOW Y	HIGH X	LOW X
			X				X
		X			X	X	X
	X				X		X
X				X			X
		X	X		X	X	X
	X		X		X		X
X			X	X			X
	X	X			X	X	X
X		X		X	X	X	X
X	X			X	X		X
	X	X	X		X	X	X
X		X	X	X	X	X	X
X	X		X	X	X		X
X	X	X		X	X	X	X
X	X	X	X	X	X	X	X
Sending to Initial Address				X	X	X	X
Returning to Remembered Address							X

TABLE G-4. COORDINATE CONVERSION CHART

LOW ORDER X			X or Y COORDINATE								LOW ORDER Y		
ASCII	DEC.	HEX									HEX	DEC.	ASCII
@	64	40	0	32	64	96	128	160	192	224	60	96	`
A	65	41	1	33	65	97	129	161	193	225	61	97	a
B	66	42	2	34	66	98	130	162	194	226	62	98	b
C	67	43	3	35	67	99	131	163	195	227	63	99	c
D	68	44	4	36	68	100	132	164	196	228	64	100	d
E	69	45	5	37	69	101	133	165	197	229	65	101	e
F	70	46	6	38	70	102	134	166	198	230	66	102	f
G	71	47	7	39	71	103	135	167	199	231	67	103	g
H	72	48	8	40	72	104	136	168	200	232	68	104	h
I	73	49	9	41	73	105	137	169	201	233	69	105	i
J	74	4A	10	42	74	106	138	170	202	234	6A	106	j
K	75	4B	11	43	75	107	139	171	203	235	6B	107	k
L	76	4C	12	44	76	108	140	172	204	236	6C	108	l
M	77	4D	13	45	77	109	141	173	205	237	6D	109	m
N	78	4E	14	46	78	110	142	174	206	238	6E	110	n
O	79	4F	15	47	79	111	143	175	207	239	6F	111	o
P	80	50	16	48	80	112	144	176	208	240	70	112	p
Q	81	51	17	49	81	113	145	177	209	241	71	113	q
R	82	52	18	50	82	114	146	178	210	242	72	114	r
S	83	53	19	51	83	115	147	179	211	243	73	115	s
T	84	54	20	52	84	116	148	180	212	244	74	116	t
U	85	55	21	53	85	117	149	181	213	245	75	117	u
V	86	56	22	54	86	118	150	182	214	246	76	118	v
W	87	57	23	55	87	119	151	183	215	247	77	119	w
X	88	58	24	56	88	120	152	184	216	248	78	120	x
Y	89	59	25	57	89	121	153	185	217	249	79	121	y
Z	90	5A	26	58	90	122	154	186	218	250	7A	122	z
[	91	5B	27	59	91	123	155	187	219	251	7B	123	{
\	92	5C	28	60	92	124	156	188	220	252	7C	124	|
]	93	5D	29	61	93	125	157	189	221	253	7D	125	}
^	94	5E	30	62	94	126	158	190	222	254	7E	126	~
_	95	5F	31	63	95	127	159	191	223	255	7F	127	RUBOUT (DEL)
DECIMAL			32	33	34	35	36	37	38	39			
ASCII			SP	!	"	#	$	%	&	'			
HEXADECIMAL			20	21	22	23	24	25	26	27			
			High Order X and Y										

INSTRUCTIONS: Find coordinate value in body of chart; follow that column to bottom of chart to find decimal value, hex value, or ASCII character which represents the High Y or High X byte; go to the right in the row containing the coordinate value to find the Low Y byte or go to the left to find the Low X byte. EXAMPLE: 200Y, 48X equals 38 104 33 80 in decimal character equals & h ! P in ASCII character equals 26 68 21 50 in hexadecimal character.

TABLE G-4. COORDINATE CONVERSION CHART (CONTD)

LOW ORDER X			X or Y COORDINATE								LOW ORDER Y		
ASCII	DEC.	HEX									HEX	DEC.	ASCII
@	64	40	256	288	320	352	384	416	448	480	60	96	`
A	65	41	257	289	321	353	385	417	449	481	61	97	a
B	66	42	258	290	322	354	386	418	450	482	62	98	b
C	67	43	259	291	323	355	387	419	451	483	63	99	c
D	68	44	260	292	324	356	388	420	452	484	64	100	d
E	69	45	261	293	325	357	389	421	453	485	65	101	e
F	70	46	262	294	326	358	390	422	454	486	66	102	f
G	71	47	263	295	327	359	391	423	455	487	67	103	g
H	72	48	264	296	328	360	392	424	456	488	68	104	h
I	73	49	265	297	329	361	393	425	457	489	69	105	i
J	74	4A	266	298	330	362	394	426	458	490	6A	106	j
K	75	4B	267	299	331	363	395	427	459	491	6B	107	k
L	76	4C	268	300	332	364	396	428	460	492	6C	108	l
M	77	4D	269	301	333	365	397	429	461	493	6D	109	m
N	78	4E	270	302	334	366	398	430	462	494	6E	110	n
O	79	4F	271	303	335	367	399	431	463	495	6F	111	o
P	80	50	272	304	336	368	400	432	464	496	70	112	p
Q	81	51	273	305	337	369	401	433	465	497	71	113	q
R	82	52	274	306	338	370	402	434	466	498	72	114	r
S	83	53	275	307	339	371	403	435	467	499	73	115	s
T	84	54	276	308	340	372	404	436	468	500	74	116	t
U	85	55	277	309	341	373	405	437	469	501	75	117	u
V	86	56	278	310	342	374	406	438	470	502	76	118	v
W	87	57	279	311	343	375	407	439	471	503	77	119	w
X	88	58	280	312	344	376	408	440	472	504	78	120	x
Y	89	59	281	313	345	377	409	441	473	505	79	121	y
Z	90	5A	282	314	346	378	410	442	474	506	7A	122	z
[	91	5B	283	315	347	379	411	443	475	507	7B	123	{
\	92	5C	284	316	348	380	412	444	476	508	7C	124	|
]	93	5D	285	317	349	381	413	445	477	509	7D	125	}
^	94	5E	286	318	350	382	414	446	478	510	7E	126	~
_	95	5F	287	319	351	383	415	447	479	511	7F	127	RUBOUT (DEL)
DECIMAL			40	41	42	43	44	45	46	47			
ASCII			(	)	*	+	,	-	.	/			
HEXADECIMAL			28	29	2A	2B	2C	2D	2E	2F			
			High Order X and Y										

TABLE G-4. COORDINATE CONVERSION CHART (CONTD)

LOW ORDER X			X or Y COORDINATE								LOW ORDER Y		
ASCII	DEC.	HEX									HEX	DEC.	ASCII
@	64	40	512	544	576	608	640	672	704	736	60	96	`
A	65	41	513	545	577	609	641	673	705	737	61	97	a
B	66	42	514	546	578	610	642	674	706	738	62	98	b
C	67	43	515	547	579	611	643	675	707	739	63	99	c
D	68	44	516	548	580	612	644	676	708	740	64	100	d
E	69	45	517	549	581	613	645	677	709	741	65	101	e
F	70	46	518	550	582	614	646	678	710	742	66	102	f
G	71	47	519	551	583	615	647	679	711	743	67	103	g
H	72	48	520	552	584	616	648	680	712	744	68	104	h
I	73	49	521	553	585	617	649	681	713	745	69	105	i
J	74	4A	522	554	586	618	650	682	714	746	6A	106	j
K	75	4B	523	555	587	619	651	683	715	747	6B	107	k
L	76	4C	524	556	588	620	652	684	716	748	6C	108	l
M	77	4D	525	557	589	621	653	685	717	749	6D	109	m
N	78	4E	526	558	590	622	654	686	718	750	6E	110	n
O	79	4F	527	559	591	623	655	687	719	751	6F	111	o
P	80	50	528	560	592	624	656	688	720	752	70	112	p
Q	81	51	529	561	593	625	657	689	721	753	71	113	q
R	82	52	530	562	594	626	658	690	722	754	72	114	r
S	83	53	531	563	595	627	659	691	723	755	73	115	s
T	84	54	532	564	596	628	660	692	724	756	74	116	t
U	85	55	533	565	597	629	661	693	725	757	75	117	u
V	86	56	534	566	598	630	662	694	726	758	76	118	v
W	87	57	535	567	599	631	663	695	727	759	77	119	w
X	88	58	536	568	600	632	664	696	728	760	78	120	x
Y	89	59	537	569	601	633	665	697	729	761	79	121	y
Z	90	5A	538	570	602	634	666	698	730	762	7A	122	z
[	91	5B	539	571	603	635	667	699	731	763	7B	123	{
\	92	5C	540	572	604	636	668	700	732	764	7C	124	|
]	93	5D	541	573	605	637	669	701	733	765	7D	125	}
^	94	5E	542	574	606	638	670	702	734	766	7E	126	~
_	95	5F	543	575	607	639	671	703	735	767	7F	127	RUBOUT (DEL)
DECIMAL			48	49	50	51	52	53	54	55			
ASCII			0	1	2	3	4	5	6	7			
HEXADECIMAL			30	31	32	33	34	35	36	37			
			High Order X and Y										

TABLE G-4. COORDINATE CONVERSION CHART (CONTD)

LOW ORDER X			X or Y COORDINATE								LOW ORDER Y		
ASCII	DEC.	HEX									HEX	DEC.	ASCII
@	64	40	768	800	832	864	896	928	960	992	60	96	`
A	65	41	769	801	833	865	897	929	961	993	61	97	a
B	66	42	770	802	834	866	898	930	962	994	62	98	b
C	67	43	771	803	835	867	899	931	963	995	63	99	c
D	68	44	772	804	836	868	900	932	964	996	64	100	d
E	69	45	773	805	837	869	901	933	965	997	65	101	e
F	70	46	774	806	838	870	902	934	966	998	66	102	f
G	71	47	775	807	839	871	903	935	967	999	67	103	g
H	72	48	776	808	840	872	904	936	968	1000	68	104	h
I	73	49	777	809	841	873	905	937	969	1001	69	105	i
J	74	4A	778	810	842	874	906	938	970	1002	6A	106	j
K	75	4B	779	811	843	875	907	939	971	1003	6B	107	k
L	76	4C	780	812	844	876	908	940	972	1004	6C	108	l
M	77	4D	781	813	845	877	909	941	973	1005	6D	109	m
N	78	4E	782	814	846	878	910	942	974	1006	6E	110	n
O	79	4F	783	815	847	879	911	943	975	1007	6F	111	o
P	80	50	784	816	848	880	912	944	976	1008	70	112	p
Q	81	51	785	817	849	881	913	945	977	1009	71	113	q
R	82	52	786	818	850	882	914	946	978	1010	72	114	r
S	83	53	787	819	851	883	915	947	979	1011	73	115	s
T	84	54	788	820	852	884	916	948	980	1012	74	116	t
U	85	55	789	821	853	885	917	949	981	1013	75	117	u
V	86	56	790	822	854	886	918	950	982	1014	76	118	v
W	87	57	791	823	855	887	919	951	983	1015	77	119	w
X	88	58	792	824	856	888	920	952	984	1016	78	120	x
Y	89	59	793	825	857	889	921	953	985	1017	79	121	y
Z	90	5A	794	826	858	890	922	954	986	1018	7A	122	z
[	91	5B	795	827	859	891	923	955	987	1019	7B	123	{
\	92	5C	796	828	860	892	924	956	988	1020	7C	124	|
]	93	5D	797	829	861	893	925	957	989	1021	7D	125	}
^	94	5E	798	830	862	894	926	958	990	1022	7E	126	~
_	95	5F	799	831	863	895	927	959	991	1023	7F	127	RUBOUT (DEL)
DECIMAL			56	57	58	59	60	61	62	63			
ASCII			8	9	:	;	<	=	>	?			
HEXADECIMAL			38	39	3A	3B	3C	3D	3E	3F			
			High Order X and Y										

Table H-1 summarizes the escape sequences used with the graphics option.

TABLE H-1. GRAPHICS OPERATION CONTROL CODES

ASCII CHAR.	HEX. CODE	KEY-STROKES	EFFECT WITH ESC CONDITION CLEARED*	EFFECT WITH ESC CONDITION SET*
NUL	00	CTRL @	No effect.	Leaves ESC condition set.
SOH	01	CTRL a	No effect.	Selects graphics operation.
STX	02	CTRL b	No effect.	No effect. (Reserved.)
ETX	03	CTRL c	No effect.	Selects CYBER mode operation.
EOT	04	CTRL d	No effect.	No effect.
ENQ	05	CTRL e	No effect.	Alpha mode - Causes terminal to transmit status and alpha cursor position, and does not change display bypass. GIN mode - Causes terminal to transmit GIN cursor position, selects alpha mode, and does not change display bypass. Other modes - Causes terminal to transmit status and X,Y position, and does not change display bypass.
ACK	06	CTRL f	No effect.	No effect.
BEL	07	CTRL g	Graph mode - Following GS, causes the first vector to be written, clears display bypass, and sounds audible alarm.	Graph mode - Following GS, causes the first vector to be written, clears display bypass, and sounds audible alarm.

TABLE H-1. GRAPHICS OPERATION CONTROL CODES (CONTD)

ASCII CHAR.	HEX. CODE	KEY-STROKES	EFFECT WITH ESC CONDITION CLEARED*	EFFECT WITH ESC CONDITION SET*
			<u>Other modes</u> - Clears display bypass and sounds audible alarm.	<u>Other modes</u> - Clears display bypass and sounds audible alarm.
BS	08	ERASE or CTRL h or ← or ⇤	Moves cursor one position to the left and clears display bypass. If at margin position on a line, cursor moves to rightmost position on line above. If at margin position of topmost line, cursor moves to last position of top line. ERASE or ⇤ additionally erases character if off-line or half duplex is selected.	Moves cursor one position to the left and clears display bypass. If at margin position on a line, cursor moves to margin position on line above. If at margin postion of topmost line, cursor moves to last position of top line. ERASE or ⇤ additionally erases character if off-line or half duplex is selected.
HT	09	⇥ or CTRL i	Moves cursor one position to the right and clears display bypass. If at rightmost position on a line, cursor moves to margin position on next lower line. If at last position of bottom line, cursor moves to new margin position on top line.	Moves cursor one position to the right and clears display bypass. If at margin position on a line, cursor moves to margin position on next lower line. If at last position of bottom line, cursor moves to new margin position on top line.
LF	OA	LF or CTRL j	Moves cursor down one line and clears display bypass. If at bottom line, cursor moves to top line. No effect if not in alpha mode. May cause page-full condition to occur.	No effect. Leaves ESC condition set.
VT	OB	CTRL k	Moves cursor up one line and clears display bypass. If at top line, cursor does move.	Moves cursor up one line and clears display bypass. If at top line, cursor does not move.

TABLE H-1. GRAPHICS OPERATION CONTROL CODES (CONTD)

ASCII CHAR.	HEX. CODE	KEY-STROKES	EFFECT WITH ESC CONDITION CLEARED*	EFFECT WITH ESC CONDITION SET*
FF	0C	CTRL l	No effect.	Clears screen, selects alpha mode, moves position to upper left corner of display, and clears display bypass.
CR	0D	NEXT or CTRL m or CR	Moves cursor to margin position of current line, resets terminal to alpha mode, and clears display bypass.	Leaves ESC condition set.
SO	0E	CTRL n	No effect.	No effect.
SI	0F	CTRL o	No effect.	No effect.
DLE	10	CTRL p	No effect.	No effect.
DC1	11	CTRL q	No effect.	<u>Alpha mode</u> - Selects inverse video write. <u>Other modes</u> - Selects erase.
DC2	12	CTRL r	No effect.	<u>Alpha mode</u> - Selects overstrike write. <u>Other modes</u> - Selects write.
DC3	13	CTRL s	No effect.	<u>Alpha mode</u> - Selects overstrike erase. <u>Other modes</u> - Selects erase.
DC4	14	CTRL t	No effect.	<u>Alpha mode</u> - Selects clear write. <u>Other modes</u> - Selects write.
NAK	15	CTRL u	No effect.	No effect.
SYN	16	CTRL v	No effect.	No effect.
ETB	17	CTRL w	No effect.	Makes screen copy.

TABLE H-1. GRAPHICS OPERATION CONTROL CODES (CONTD)

ASCII CHAR.	HEX. CODE	KEY-STROKES	EFFECT WITH ESC CONDITION CLEARED*	EFFECT WITH ESC CONDITION SET*
CAN	18	CTRL x	No effect.	Sets display bypass.
EM	18	CTRL y	Selects block mode and clears display bypass.	Undefined.
SUB	1A	CTRL z	No effect.	Selects GIN mode and sets display bypass.
ESC	1B	ESC	Sets ESC condition (beginning of an ESC sequence).	Leaves ESC condition set.
FS	1C	CTRL \	Selects point plot mode and clears display bypass.	Selects special point plot mode and clears display bypass.
GS	1D	CTRL]	Selects graph mode and clears display bypass.	Selects graph mode and clears display bypass.
RS	1E	CTRL =	Selects incremental plot mode and clears display bypass.	Undefined.
US	1F	CTRL _	GIN mode - No effect. Other modes - Selects alpha mode and clears display bypass.	GIN mode - No effect. Other modes - Selects alpha mode and clears display bypass.
!	21	!	**	No effect if graphics tablet is deselected (character following ! is discarded). Character following ! is used for tablet control if tablet is selected.
7	37	7	**	Selects size 0 alpha characters (largest).
8	38	8	**	Selects size 1 alpha characters.
9	39	9	**	Selects size 2 alpha characters.

TABLE H-1. GRAPHICS OPERATION CONTROL CODES (CONTD)

ASCII CHAR.	HEX. CODE	KEY-STROKES	EFFECT WITH ESC CONDITION CLEARED*	EFFECT WITH ESC CONDITION SET*
:	3A	:	**	Selects size 3 alpha characters.
;	3B	;	**	Selects size 4 alpha characters (smallest).
<	3C	<	**	Selects scaled coordinates with Y bias.
=	3D	=	**	Selects scaled coordinates without Y bias.
>	3E	>	**	Selects unscaled coordinates.
[	5B	[	**	Selects X-ON/X-OFF flow control.
]	5D	]	**	Deselects X-ON/X-OFF flow control.
_	5F	_	**	Character after _ controls: A. Selects tablet continuously active B. Allows tablet arming via ESC ! X C. Deselects tablet D. Causes transmission of graphics firmware revision to host system
`	60	`	**	Selects solid vectors.
a	61	a	**	Selects dotted vectors.
b	62	b	**	Selects dot-dashed vectors.
c	63	c	**	Selects short-dashed vectors.
d	64	d	**	Selects long-dashed vectors.

TABLE H-1. GRAPHICS OPERATION CONTROL CODES (CONTD)

ASCII CHAR.	HEX. CODE	KEY-STROKES	EFFECT WITH ESC CONDITION CLEARED*	EFFECT WITH ESC CONDITION SET*
h	68	h	**	Selects solid vectors.***
i	69	i	**	Selects dotted vectors. vectors.***
j	6A	j	**	Selects dot-dashed vectors.***
k	6B	k	**	Selects short-dashed vectors.***
l	6C	l	**	Selects long-dashed vectors.***
p	70	p	**	Selects solid vectors.***
q	71	q	**	Selects dotted vectors.***
r	72	r	**	Selects dot-dashed vectors.***
s	73	s	**	Selects short-dashed vectors.***
t	74	t	**	Selects long-dashed vectors.***
All Oth-ers	--		**	No effect.

*ESC condition is initiated by reception of an ESC character and terminates following reception of the following character, unless otherwise shown.

**The code is a displayable character in alpha mode, or part of a coordinate in point-plot, special point-plot, graph, and block modes.

***Provided only for compatibility with Tektronix 401X terminals. Do not use for new applications.

The following table lists the characters that vary between the character sets that the terminal can display.

CHARACTER SET	HEXADECIMAL CODE											
	23	24	40	5B	5C	5D	5E	60	7B	7C	7D	7E
Standard	#	$	@	[	\	]	^	`	{	¦	}	~
French	£	$	à	°	ç	§	^	`	é	ù	è	¨
German	#	$	§	Ä	Ö	Ü	^	`	ä	ö	ü	β
Swedish/Finnish	#	¤	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
United Kingdom	£	$	@	[	\	]	^	`	{	¦	}	‾
Spanish	£	$	§	i	Ñ	¿	^	`	°	ñ	ç	~
Danish/Norwegian	#	$	@	Æ	Ø	Å	^	`	æ	ø	å	‾

ASCII	A standard code, using a coded character set, used for information interchange among data processing systems, data communications systems, and associated equipment. The ASCII set consists of control characters and graphic characters, and corresponds to the terminal keyboard. ASCII is an acronym for American Standard Code for Information Interchange.
Application Software	Software packages that perform specific individual user functions, such as ledger accounting, word processing, or inventory control. WORDSTAR is an example of application software.
Asynchronous	Occurring without a regular or predictable time relationship to a specific event. As applied to program execution, unexpected or unpredictable in regard to instruction sequence.
Backup	To make additional copies of files, on either flexible or rigid disks, for use or for storage.
Baud	A unit of signalling speed equal to the number of discrete conditions or signal events per second. In a train of binary signals, one baud equals one bit per second.
Baud Rate	The number of discrete conditions or signal events per second.
BDOS	An acronym for Basic Disk Operating System.
BIOS	An acronym for Basic Input Output System.
Bootstrap	To load the first few instructions of a routine into storage, and to use those instructions to bring in the rest of the routine. Often involves either entering a few instructions manually or using a special key on the console. Also called deadstart.

Byte	A sequence of adjacent binary digits used as a unit and normally shorter than a word. A standard byte is eight bits in length as differentiated from a six bit character.
Byte Size	The number of bits making up a byte. Usual size is eight bits.
Checksum	A number used to check that the contents of a data block have not been altered.
Cold Boot	To activate a computer system from a completely inactive state.
Controlware	A processor program, integral to a specific product, that provides the product with a set of functional operating characteristics. Controlware is supplied as part of a product and is necessary for correct product operation. The programmed functions may be analogous to hardware logic and are thus documented, maintained, and supported.
CP/M	Control Program for Microcomputers or Control Program Monitor - a registered trademark at Digital Research Inc.
CPU	Acronym for Central Processing Unit. The portion of the hardware in a computer system that is distinct from the peripheral equipment and provides primary data storage (main memory), program execution, and input/output.
Cursor	A movable marker used as a reference point on a CRT screen.
Disk Designating Letter	A letter designating each logical disk portion of a physical disk.
Disk Drive	A mechanism used to move a disk pack or magnetic disk and to control the movement of the disk.
Disk Surface Fault	Deterioration or flaws on the surface of a rigid disk. When ten percent or more of useable data area is flawed, the system displays an error message.

Double-Density Disk	A double-sided flexible disk that holds approximately 1.2 million-bytes of usable storage.
Firmware	A physical electronic component that contains a specific program. The component is incorporated in a product to provide a programmed mode of operation that defines the functional characteristics of the product. Firmware is not self-modifiable. It can only be changed by physical modification or replacement.
Format	To place address information and data block construction on a disk.
Input/Output Buffering	Temporary storage of information during a transfer of information. Used to compensate for a difference in the rate of data flow, or the time that events occur, when data is transmitted from one device to another.
Inverse Video	A display on a CRT screen that is the reverse of the normal display. For example, when green letters are usually displayed on a black background, inverse video shows black letters on a green background. Use mainly for highlighting.
Load	To put data into internal storage or working registers.
Logical Disk	A 4- or 8-megabyte subdivision of a physical disk in the rigid-disk configuration.
Menu	The various options within an interactive application program. The system displays these options on the terminal screen, enabling the user to select the desired option.
Microcomputer	1. A computer that is constructed using a microprocessor as a basic element. 2. A computer that combines all of the CPU, memory, and peripheral functions on a chip of silicon. It may be sold in an integrated circuit package, or with additional memory and peripheral circuits packaged on a board at a console. An eight-bit computer on a chip is used as a component.

Mnemonic	A symbol chosen to assist the human memory, such as an abbreviation for a word or concept. "Mpy" for "multiply" is a mnemonic.
Modem	A contraction of "modulator-demodulator." A unit of signal-conversion equipment that modulates and demodulates signals. Modems provide interface to data processing equipment and then drive signals across the communication line to allow data to pass. A telephone line is an example of a communication line.
Multi-Processor System	A computing system having two or more, not necessarily identical, computation modules that can be executing concurrently.
Parameter	A variable that is given a constant value for a specific purpose or process.
Parity	An error detection method where a check bit is added to each byte to make the sum of all bits, including the check bit, always even or always odd.
Physical Disk	The actual rigid disk, containing 12.5- or 25-megabyte usable storage.
PLATO	An acronym for Programmed Logic for Automatic Teaching Operations and Programmed Learning for Automated Teaching Operations. PLATO is a computer-based education system of Control Data.
Read from Disk	To sense or retrieve information recorded on a disk.
Read-only Memory (ROM)	An addressable memory with a permanent data pattern. The memory cannot be written or changed. ROM patterns are established when a device is manufactured.
Sector	A physical section of storage on a disk that is the smallest addressable portion of that disk.
Single-Density Disk	A single-sided flexible disk that holds approximately 243-thousand bytes of usable storage.
Software	Programs, procedures, rules, and associated documentation pertaining to the operation of a computer system.

Surface Analysis	The process of reading each sector of a single- or double-density flexible disk and reporting any sector read errors. Surface analysis can only read flexible disks, and can be used to check the quality of any Control Data 110 formatted and written to flexible disk.
Touchpanel	A sheet of plastic covering the terminal display area that, when activated, allows positioning of the cursor in the area where the surface is touched.
Utility	A program or routine in general support of the processes of a computer. For example, an input routine or a sort program.
Verify	To determine whether transcription of data or any other operation has been accomplished accurately.
Warm Boot	To activate or reactivate software in an already active computer system.
Write to Disk	To record information on a disk.
Write-Protect Slot	A small slot on the bottom right edge of some flexible disks. The slot exposes the disk material and must be covered by a small piece of tape (supplied by the manufacturer) before the disk may be formatted.

COMMENT SHEET

MANUAL TITLE: CDC® 110 Viking System
Software User's Manual

PUBLICATION NO.: 62940076 REVISION: B

NAME: ______________________________

COMPANY: ______________________________

STREET ADDRESS: ______________________________

CITY: ____________ STATE: ____________ ZIP CODE: ____________

This form is not intended to be used as an order blank. Control Data Corporation welcomes your evaluation of this manual. Please indicate any errors, suggested additions or deletions, or general comments below (please include page number references).

NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

TAPLE STAPLE

OLD FOLD

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 8241 MINNEAPOLIS, MINN.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION

Technical Publications Department

2401 North Fairview Avenue

St. Paul, Minnesota 55113

CUT ALONG LINE

LD FOLD

03377

CORPORATE HEADQUARTERS, P.O. BOX 0, MINNEAPOLIS, MINN. 55440
SALES OFFICES AND SERVICE CENTERS IN MAJOR CITIES THROUGHOUT THE WORLD

LITHO IN U.S.A.

CONTROL DATA CORPORATION