PC

Technical Reference Manual

REFLECTION® 1/7

Walker Richer & Quinn, Inc.

REFLECTION® 1/7

Walker Richer & Quinn, Inc.

Fransen/King, Ltd.
Office Extend is a trademark.

Genoa Systems Corporation
Genoa is a trademark.

GTE Telenet Communications Corporation
Telenet is a public network.

Hayes Microcomputer Products, Inc.
Smartmodem is a registered trademark and Hayes is a trademark.

Hercules Computer Technology
Hercules is a trademark.

Hewlett-Packard Company
AdvanceNet, DSG/3000, HP 3000, HP Draw, HP EasyChart, HP Desk, HP-Telnet, LaserJet, PaintJet, Vectra, and Vectra ES are trademarks.

Intel Corporation
Intel is a registered trademark and AboveBoard is a trademark.

International Business Machines Corporation
IBM and PS/2 are registered trademarks and PC-XT, PC-AT, IBM-LANACS and Token-Ring are trademarks.

Kabushiki Kaisha Toshiba
Toshiba is a registered trademark.

Lotus Development Corporation
DIF, Lotus and 1-2-3 are registered trademarks.

McDonnell Douglas Network Systems Company
TYMNET is a public network and a registered trademark.

Microcom, Inc.
Microcom Network Protocol and MNP are trademarks.

MicroPro International Corporation
WordStar is a registered trademark.

Microsoft Corporation
Microsoft is a trademark and MS-DOS, Multiplan, XENIX, and Windows are trademarks.

Network Products
ACS^2 is a trademark.

Novell, Inc.
NetWare for VMS, NACS, and Nasi are trademarks.

Okidata America Coporation
Okidata is a registered trademark.

Robelle Consulting, Ltd.
QEDIT is a trademark.

Paradise Systems, Inc.
Paradise VGA Plus Card is a trademark.

Quarterdeck Office Systems
DESQview is a trademark.

Racal-Interlan, Inc.
Racal-Interlan is a trademark.

STB, Inc.
Chauffeur HT is a trademark.

Tektronix, Inc.
TEKTRONIX is a registered trademark.

Tseng Labs
EVA and ULTRAPAK are trademarks.

Ungermann-Bass, Inc.
Net/One and NIU are registered trademarks.

WordPerfect Corporation
WordPerfect is a trademark.

Xerox Corporation
Ethernet is a registered trademark.

Walker Richer & Quinn, Inc.
P.O. Box 31876
Seattle, WA 98103-1876
USA

Telephone: (206) 324-0407
TELEX: 311 743 WRQ UR
FAX: (206) 329-7565
Technical Support: (206) 325-4357
Bulletin Board: (206) 322-8047

In Europe
Telephone: +31-70-356-09-63
FAX: +31-70-356-12-44

Reflection 1/7 *Technical Reference Manual*
Version 4.0
August 1990

This manual is printed on recycled paper.

Contents

Figures

Tables

Preface

Welcome to Reflection. The Reflection Series was created by Walker Richer & Quinn to allow you to make the most use of your IBM PC and compatibles. Now all the stand-alone features of your PC are complemented by the ability to communicate with your HP 3000, HP 9000, DEC VAX, UNIX system (including ULTRIX), or other asynchronous host. Reflection 1 and 7 are both documented in this manual. See page 3 for a list of the terminals that are emulated.

Version 4.00 includes two new important features: *

- You can use wildcards when sending or receiving files. *
- A new *state save* feature lets you exit Reflection temporarily without *
  leaving the program resident in memory. Information about your *
  Reflection session is saved to a file. When you run Reflection again, the *
  state save file is loaded if it is found. *

Reflection's **Command Language** and the **PLUS** option are discussed in separate manuals. All Reflection products share a command language; it is an alternate means of taking advantage of Reflection's many features.

PLUS is an option that gives you network support, lets you to back up your PC disk files to the host and restore them when necessary, and provides the full set of Reflection's Application Program Interface (API) libraries.

An upgrade from Reflection 1 to Reflection 7 or from a non-PLUS version to a PLUS version can be ordered at any time.

Section 1

Overview

Chapter 1

Introducing Reflection 1 and 7

1.1 Terminal Emulation

Reflection enables the IBM Personal Computer or compatible PC to *emulate*, or operate like, the terminal of a much larger computer system. Reflection emulates Hewlett-Packard and Digital Equipment Corporation terminals, depending on which terminal class you choose, or which program you load.

- **Reflection 1 and 7 HP Emulation:**

  2392A, 2622A, 700/92, 700/94, 2624B, 2626A, 2394A
- **Reflection 7 HP Graphics Emulation:**

  2390A, 2397A, 150A, 2627A Color and 2623A, 2393A Monochrome, Tektronix emulation
- **Reflection 1 and 7 Digital Emulation:**

  VT52, VT102, VT220*

See *HP Terminal Configuration* on page 103 and *VT Terminal Configuration* on page 121.

All emulation modes allow you to communicate with a host computer as if you were using a terminal. You can transfer data between the PC and a host computer, and send data from the display memory of the PC to a printer or a disk file.

* Most of the features of the VT220 are included. The full set of UDKs is not supported, and DRCS is not supported. Reflection 2 and 4 are complete and more DEC-like emulations of the VT241 and VT320.

Reflection 7 graphics terminal emulation is documented fully in this manual. You must have the required hardware to take advantage of these features.

1.2 Reflection Features

Reflection includes a complete system for transferring files from the PC to HP 3000s, VAX/VMS, and UNIX-based systems and supports five file transfer protocols.

Reflection has its own command language, which provides a shortcut to data communications operations. This allows you to write your own programs for quick execution of commonly performed operations. It is covered in a separate manual called the Reflection *Command Language Manual*, included with the Reflection package.

Reflection has a multitasking feature that allows the program to run in the background while you use PC software in the foreground. Along with the ability to pop up Reflection, it *keeps running* when you temporarily exit to work with a PC application. If you begin a file transfer or a command file, Reflection continues to execute these until finished, while you have access to the PC for other tasks.

1.3 Requirements

- Reflection program and support files disks
- An IBM Personal Computer (or 100% compatible)
- Version 2.0 or later of the IBM Disk Operating System (PC-DOS) or Microsoft's MS-DOS (hereafter, just DOS)
- At least 256K of read/write memory for Reflection 1 (the graphics package requires more, up to 320K)
- Either an alphanumeric or color/graphics display adapter card, and a monitor capable of displaying 80 columns by 25 rows
- A supported graphics adapter card for Reflection 7
- At least one disk drive

Reflection is intended to be used in conjunction with a host computer. Communicating with a host requires the following additional equipment:

- A communications port: a serial port, an internal modem card, or a local area network card

- Another computer with asynchronous communications capabilities to serve as the host computer
- A link to the host computer: a direct RS-232 cable connection, an internal modem card, an external modem with an RS-232 cable to the PC, or appropriate LAN hardware and software
- To use VAXLINK2, Walker Richer & Quinn's proprietary host file transfer program for VT emulation, you must have VAX/VMS version 5.x or greater.

1.4 Using the Reflection Manuals

The Reflection 1 and 7 documentation set consists of the four manuals described below.

Quick Start Guide

The *Quick Start Guide* is a general overview of Reflection 1 and 7. If you are new to Reflection or have never explored beyond basic emulation, read this guide.

Technical Reference Manual

The *Technical Reference Manual* has the following sections:

- **Overview** introduces Reflection, including how to load, configure, and exit the program.
- **Host Connections** is devoted to more detailed instructions for connecting the PC to the host computer.
- **Troubleshooting** guides you through problems you may have when using Reflection, including a list of error messages and what they mean.
- **Configuration** covers the configuration values for data communications, terminal, printer, plotter, color, and user keys as well as global configuration values.
- **Keyboard** explains what PC keystrokes are used to perform HP and VT terminal functions and Reflection functions.
- **Keyboard Remapping** shows you how to reassign keyboard functions to new keystrokes.
- **Operations** describes all aspects of remote and local operations. It contains information on how to run Reflection in the background, discusses disk and printer operations, and explains how to use Reflection with other software packages.

- **Special Features** explains Reflection terminal enhancements. It also discusses memory management techniques.
- **File Transfer** details the five file transfer protocols and describes how each is used.
- **HP Graphics Emulation** applies if you are using Reflection 7. It introduces the use of Reflection with graphics host programs.
- The **Appendices** cover escape sequences (including graphics), HP block transfers and format mode, cursor positioning and sensing, status requests, and the HP character sets. Similar information is provided for VT terminal emulation: control functions, keyboard generated codes, and character sets. Another appendix lists screens and other error messages.
- The **Glossary** describes terms used throughout this manual.

Command Language Manual

The Reflection *Command Language Manual* covers all aspects of Reflection's command language. It introduces the use of individual commands as well as how to create command files. A variety of examples and detailed information on how each command works is included.

The PLUS Manual

The *PLUS Manual* documents additional Reflection features available with the PLUS option:

- Complete network support via a variety of LANs.
- The ability to back up your PC disk files to the host and restore them when necessary.
- The full set of Reflection's Application Program Interface (API) libraries.

An upgrade from a non-PLUS version of Reflection to a PLUS version can be ordered at any time.

1.5 Documentation Conventions

- The term *PC* is used throughout the manual to mean any IBM Personal Computer or compatible.

- Commands to be entered through the keyboard appear in **bold** type, as follows:

  Type **R1** and press the Return key.

- On a line by itself, a command to be entered through the keyboard appears in the following indented format:

```
SET TYPE-AHEAD YES
```

- The names of keys are framed, e.g., Return, Shift, Esc. Key combinations are shown connected by a hyphen: e.g., Alt-A. Do not include the hyphen when pressing the key combination.
- Material that has recently been added to the documentation is *flagged* to *
  highlight it. This paragraph is an example. If an existing feature has *
  been significantly clarified, it is also flagged. *
- Screen labels, escape sequences, fields, and messages are shown exactly as they appear on the screen. When such expressions appear as part of the text, they are displayed in bold type to distinguish them from other text.
- The parameter <filespec> refers to a file's drive, path, and name.
- Variable parameters are enclosed in angle brackets, (< >), like this:

```
SEND <filespec> TO <filespec>
```

  When entering values for these parameters, do not include the angle brackets.

- The eight labels across the bottom of the screen, reading from left to right, correspond to the function keys F1 through F8 on the keyboard.
- Some key names used in these manuals differ from those on the PC keyboard. For this reason, the text refers to the keys by name instead of by symbol. *Using the Keyboard* defines some of these keys and their functions as used in Reflection on page 169.
- Most functions and commands may be keyed in either upper or lowercase. Commands are shown in uppercase.

  Note: The exception to this convention is the representation of escape sequences, which can contain upper and lowercase characters that must be keyed precisely as displayed in the text.

- The graphics emulation portion of this manual covers the graphics cursor, graphics memory, and graphics display. If you are using a non-graphics emulation, only the alpha cursor, alpha memory, and alpha display are relevant to you.
- Control codes are represented by a two letter mnemonic; for example, EC indicates the escape control code and CR indicates the carriage return control code.

Chapter 2

Getting Started

The *Quick Start Guide* is designed for new users of Reflection and discusses, among other things, how to install Reflection, run it, and configure it. This chapter covers the same material in more depth.

2.1 Installing Reflection

Installation of Reflection is performed through a series of prompts. A supplied program walks you through the installation process. Install Reflection on a PC, XT, AT, IBM portable or 100% compatible as follows:

1. Put the Reflection distribution diskette in drive A or B.
2. Type **A:** (or **B:**) and press [Return].
3. Type **SETUP** and press [Return].
4. Respond to the prompts until all the files are copied.

The following support files are included on the Reflection diskettes:

- SETUP.EXE creates a working copy of Reflection. SETUP can also be run to change the configuration you chose when you first installed.
- R1.HLP or R7.HLP is the online help file for Reflection's command language.
- WHATSNEW.DOC describes changes in Reflection since the previous release.
- Each Reflection disk contains a separate README file listing all the files on that disk and their functions.

- PCLINK2.PUB is the host file transfer program for the HP 3000.

  The following files are required to upload a new version: PCLINK2.PUB, WRQUPLOA, UPLOADHP.RCL, and VERIFYPH.
- VAXLINK2 is the host file transfer program for VAX/VMS hosts.

  The following files are required to upload a new version: VAXLINK2.EXE, WRQUPLOA.MAR, and UPLOADVX.RCL.
- UNXLINK2.C is the host file transfer program for a UNIX system.

  UNXLINK2.C, UNXXFER2.C, WRQKERM.C, and UPLOADUX.RCL are required for uploading the host file transfer program to UNIX systems, including HP 9000s.
- WRQBACK.RCL works with PLUS versions of Reflection. It steps you through backup and restore operations. A key label on the File Transfer menu provides a simple means of executing the command file, or it can be run from the command line.

  Additional files associated with the backup and restore capabilities of PLUS versions of Reflection are HPDIR.LBL, UPLHPDIR.RCL, XTRACT.LBL, UPXTRACT.RCL, VAXDIR.EXE, VAXDIR.DCL, VAXDIR.RCL, UNIXDIR.C, and UNIXDIR.RCL.
- HOWMANY shows how many users are currently running a given server version of Reflection, and what the maximum number of users is. Type **HOWMANY** for documentation on how to use the program. *

  If you exceed the allowed number of users, Reflection does not run and an error message is displayed at the DOS prompt. You can increase the number of users by purchasing an "Add-On Pack" from Walker Richer & Quinn. *
- AL2WRQ.EXE is used to convert AdvanceLink command files to Reflection's command language.
- COMCHECK.COM allows you to check your serial ports.

  COMDRIVE.COM supports additional IRQ assignments to a COM port.
- FILEINFO.COM helps you determine if non-ASCII characters are contained in a PC file.
- CRYPT.COM allows you to encrypt Reflection command files.

- CHARTRAN is a utility to translate between character sets.
- Reflection command file examples have the extension .RCL.
- COBDEMO.COB is an example of a host-initiated file transfer program. It is described in detail in the *Command Language Manual.*
- DIAL is a sample modem dial-up file.
- A series of files with the extension .PIF are program information files *
  that tell Microsoft Windows how to run Reflection. COM34WIN.EXE *
  allows you to use COM3 or COM4 on a non-microchannel AT-class *
  machine. *
- Font files have the extension .FT.
- KEYMAP.EXE, KEYMON.COM, and a series of supplied keyboard files with the extension *.KBM are all part of keyboard remapping.
- Files supporting other languages have the extension .LOC.
- R<n>PRINT.CFG is the Reflection printer configuration file, where <*n*> is the Reflection product you are using.

2.2 Running Reflection

To run your working copy of Reflection, type **R1** for Reflection 1, or **R7** for Reflection 7, and press Return. You can add a switch, a configuration filename, or a command file to execute upon loading. The entire syntax follows:

```
R1 [/<switch>[/<switch>...]] [config] [,rcl]
```

Switches specify certain configuration options that are not stored in the configuration file or may only be available from the DOS command line.

[/switch /switch ...]

Any number of switches may be used when loading Reflection. They are explained below.

[?]

Type **R1 ?** or **R7 ?** to see a list of all possible switches and a short explanation for each one.

[config]

The name of a configuration file is optional. Without this parameter the default configuration file (R1.CFG or R7.CFG) is used.

[,rcl]

The name of a command file is optional. If specified, the command file is immediately executed upon entering Reflection. If a configuration file does not precede the command filename, the comma is required. The extension **.RCL** is recommended for command language files.

Startup Switches

/A—CGA or Monochrome

Forces Reflection to assume a CGA or monochrome card is being used. This switch also allows DESQview windowing.

/B—Background

Loads Reflection and immediately puts it into the background.

/C—IBM Font and 25-line mode

Loads Reflection with the IBM font, using 23 lines for text and two lines for labels. On an HP Vectra with a Multimode monitor, suppresses 27-line mode. On an EGA, this switch is identical to **/I**.

/D—Defaults

Loads Reflection with all default settings. Any switches you have set up in your AUTOEXEC.BAT file (see page 21) are ignored. A state save file, if one exists, is not loaded.

/E—Expanded Memory

Tells Reflection to use expanded memory (EMS) if a LIM expanded *
memory driver is present. Reflection loads both program and overlay *
code into expanded memory, making more memory available for *
running other PC applications. *

/F—Font File

Loads Reflection with a file called EGA7X12.FT if it's found in the current directory. This file is provided on a support disk, and loads an EGA-compatible font. Reflection's default font is not used if this switch is used.

This switch is only valid with Reflection 1, since Reflection 7 always loads the external font file R7.FT.

/G<n>—Graphics Adapter

Explicitly defines the graphics adapter type for Reflection 7.

When loading, Reflection 7 attempts to determine your graphics hardware. Depending on your graphics adapter, Reflection is configured as a 2623A (monochrome) graphics terminal or as a 2627A (color) graphics terminal. The terminal ID is checked by Reflection after determining the adapter type to ensure that the capabilities of the terminal indicated are available. Some clones and/or display adapters may occasionally require you to explicitly define your adapter type by using one of the switches listed below when loading Reflection.

While no special configuration is required to run Reflection 7 in graphics mode, running without graphics emulation preserves approximately 100K of memory in the PC. If you are not using the graphics portion, use **/G0** when loading Reflection to force a non-graphics emulation. **/G0** emulates a 2392A terminal.

Suppress Graphics

/G0—Suppress graphics
Suppresses graphics (uses less memory).

Monochrome Graphics

/G1—CGA, EGA, VGA with standard monitor
Assumes a CGA, EGA, or VGA with standard color monitor.

/G2—EGA, VGA monochrome adapter
Assumes an EGA or VGA monochrome emulation, with either a monochrome monitor or an enhanced color monitor.

/G3—Hercules
Assumes a Hercules monochrome graphics card.

/G4—AT&T 6300
Assumes an AT&T 6300 adapter.

/G5—HP Multimode
Assumes an HP Multimode adapter.

/G6—Sigma Design
Assumes a Sigma Design Color 400 card.

/G8— MCGA
Assumes an MCGA adapter.

Color Graphics

/G7—Full-color graphics support
Assumes an EGA with 128K and enhanced color monitor or VGA and VGA monitor.

/G9—Suppress Reflection's graphics mode set
Lets the PC's own BIOS set the graphics mode on the graphics adapter card for EGAs and VGAs instead of having Reflection set it. Use this switch if garbled text appears on the screen in graphics mode. The Reflection SET command XOFF-ON-MODE-SWITCH should be set to *YES* when using this switch. See the Reflection *Command Language Manual*. *

/H—HP Emulation
Loads HP emulation, regardless of the configured default.

/I—IBM Font
Inhibits loading Reflection fonts into the video adapter when Reflection is running in text mode. This limits the characters that can be displayed to those normally available on the IBM PC and reduces Reflection's RAM requirements.

If you're using an HP Vectra with a Multimode adapter, this switch displays a font that is larger than HP's Roman 8 character set. The 27-line display is maintained, but only a subset of the Roman 8 character set is available. This switch suppresses 27-line mode on an EGA or VGA. If you are using an EGA or VGA, this switch is identical to /C.

/J—Do Not Rotate Interrupt Priorities
By default, Reflection rotates interrupt priorities such that datacomm is the highest. To prevent the rotation of priorities, use /J when starting Reflection.

* The XOFF-ON-MODE-SWITCH defaults to *YES* for R1V and R7V.

/K0*—pre-Enhanced keyboard

Some PCs and memory-resident programs are incompatible with the use of Enhanced keyboard interrupt 16 calls, and this can make Reflection hang when it is started. **/K0** forces Reflection to use its default mapping and assume that your PC is equipped with a pre-Enhanced keyboard BIOS. Your keyboard is treated as though it were an older PC-style keyboard.

/K1

Forces Reflection to handle keyboard interrupts via interrupt 9 and do PC/XT-style keyboard enable (this is how Reflection handled the keyboard in versions prior to 4.0).

/K2

Forces Reflection to handle keyboard interrupts via interrupt 9 and do AT-style scan code locking if the machine is an AT or compatible. Use this switch if both of the following are true:

- You are using an AT or compatible machine that does *not* have interrupt 15 support for handling keyboard interrupts (e.g., an older Vectra).
- You are a DECnet/PCSA user and you find that the cursor pad arrow keys on your Enhanced keyboard occasionally generate numbers instead of cursor movement (as if the $\boxed{\text{Shift}}$ key were being held down).

If you are using an older TSR, **/K2** may hang your machine.

/K3

Note: By default, Reflection operates as if **/K3** were in effect. The explanation for the switch is provided here in case you have an earlier AUTOEXEC.BAT file or command file that references it.

Forces Reflection to use interrupt 15 if available (on PS/2s and most ATs). Use this switch if both of the following are true:

- You are using an AT or compatible machine that *does* have interrupt 15 support for handling keyboard interrupts.

---

* /K0 is equivalent to /K for versions of Reflection prior to 3.47.

– You are a DECnet/PCSA user and you find that the cursor pad arrow keys on your Enhanced keyboard occasionally generate numbers instead of cursor movement (as if the Shift key were being held down). *

/L—French Canadian Code Page 863

Loads the French Canadian character set (Code Page 863). This can be used if Reflection does not correctly translate between Roman 8 and Code Page 863.

/M—Maximum Scrolling

Loads Reflection with *MAXIMUM* scrolling speed set. This is the default; its use is limited to situations where scrolling speed has been reconfigured.

/MEM:{S | M | L}

Specifies the amount of memory Reflection uses for overlays. **S** is small, **M** is medium (the default), and **L** is large. *

Use **/MEM:S** to minimize the amount of memory that Reflection uses. Use **/MEM:L** to let Reflection use more memory so command files can run faster. See page 327 for more information on memory management techniques. *

/N—Norwegian Language Support

Loads Reflection with the Danish or Norwegian language support (Code Page 865).

/O—Loading Two Copies of Reflection

Allows you to run two copies of Reflection at the same time. If a copy of Reflection 3.3 or higher is in memory and you start Reflection from DOS, the existing copy in memory is brought to the foreground as if the hot-key had been pressed. Using **/O** overrides this default.

/P<n>—Mouse Driver

Prevents extraneous characters from appearing on the screen when Microsoft Windows built-in mouse driver for bus and serial mice is active. This driver keeps control of the mouse even if Reflection is in the foreground. If the **/P<n>** switch is not used, the mouse cursor tries to appear in Reflection's display: the result is "garbage" on the screen.

/P<n> allows you to identify the IRQ (interrupt request) line that the mouse is using so that Reflection's screen is not corrupted by the mouse.

When a bus mouse is installed, the jumper block on the bus interface card is used to set the mouse IRQ. How that IRQ is set depends on the hardware you are using. In **/P<n>**, the value of **<n>** should match the IRQ value selected when the mouse was installed. An asterisk marks the most likely setting:

Bus Mice	Serial Mice
PC with no hard disk: 2*, 3, 4, 5, 7	COM1: 4
PC with a hard disk: 2*, 3, 4, 7	COM2: 3
XT: 2*, 3, 4, 7	
AT: 5*, 3, 4, 7	
PS/2: 12	

If, for example, you have an AT and your bus mouse is configured to use IRQ5, load Reflection as follows:

```
C:>R1 /P5
```

/Q—Suppressing the Copyright Screen

Used to suppress the display of the copyright screen when Reflection is started.

/R—Rotate Priorities

This rotates interrupt priorities such that datacomm is the highest. If the Error Recap screen (Alt-E) indicates that you are getting receiver overruns (lost data) during data communications, try using this switch. If you are using an international keyboard driver and getting overruns, this switch may solve the problem.

Note: By default, Reflection operates as if **/R** were in effect. The explanation for the switch is provided here in case you have an older AUTOEXEC.BAT file or command file that references **/R**.

/S—Save Graphics Images

Enables Reflection to save and restore EGA or VGA graphics images (such as those in Microsoft Word or Windows) when you toggle in and out of Reflection with the hot-key. The saving of fonts, palettes, and some greater-than-25-line modes are also enabled.

If you are using an EGA or VGA clone card with auto-switch capability, video problems may occur when you hot-key to DOS. To avoid this, try disabling the autoswitch capability on your display adapter.

The /S switch saves the foreground screen without using any additional memory by caching it in an unused part of the EGA or VGA's memory. It reduces Reflection's memory requirements by at least 1.6K.

Note: This switch works with Reflection 7, but only when used in *
combination with **/G0** (suppress graphics). This applies to *
EGA and VGA displays. *

/SD—Don't Load State Save File

Use the **/SD** switch to run Reflection without loading a state save file *
(R<n>STATE.STA), if such a file exists. Reflection still deletes the *
state save file unless **/SK** is also used. See *State Save* on page 329 for *
more information. *

/SK—Keep State Save File

Normally, Reflection deletes an existing state save file when it is run. *
This can be inhibited with the **/SK** switch. [Alt]-[B] will still overwrite *
an existing state save file. *

/SP—Pass States Between Terminal Classes

Causes Reflection to use state save files when switching between HP *
and DEC terminal classes, preserving user key definitions and other *
terminal modes. This is useful when running host applications such as *
Oracle. See *Using Reflection with Other Software* on page 312. *

/T1—Disable Plotter Emulation

Prevents loading of Reflection 7 plotter code in order to save memory.

/U—Underline

Loads Reflection 7 with the underline option. This is only valid in color graphics with SET IGNORE-COLOR-PAIRS NO.

/V—VT Emulation

Loads VT emulation, regardless of the configured default. Directly running R1V.EXE or R7V.EXE also loads VT emulation.

/W—API Interface Support

Installs Reflection's Application Program Interface. API is available in all Reflection products, while its libraries are included only with the PLUS option.

/X—3270 File Exchange

Used with the 3270 FileExchange program (a Reflection Complement available in versions 3.2 and higher).

/Y—DOS Command Line Editors

Disables interrupt 21 function ØAH handling. This switch provides compatibility with certain DOS command line editors (e.g., CED, DOSEDIT, and NDE).

/Z— Save VGA Palette/Video

VGA only. Saves the VGA palette/video configuration when hot-keying from an application to Reflection. The palette, the blink state, the font size (but not the font itself), and the vertical resolution (EGA or VGA mode) are saved. Used in combination with **/S**, **/Z** can prevent blink/bright irregularities that occur when using the hot-key to switch back to an application like Microsoft Word from Reflection. **/Z** can be used in cases when **/S** cannot be used. *

Examples

- Type **R1** alone to load Reflection 1. R1.CFG is used as the configuration file automatically if it exists.
- Type **R1 /D** to load Reflection with all of its original default values. The values in R1.CFG are ignored.
- Type **R1 /C R1TEST.CFG** to load Reflection with the IBM font and force the display to show twenty-five lines. CFG is the recommended extension for configuration filenames. Any name you saved configuration values under can replace **R1TEST.CFG** in the above example.
- Type **R1 ,<filename>** to immediately execute a command file you have created with Reflection's command language. Note that the comma after **R1** is required if no configuration file is specified before the command file. RCL is the recommended extension for command file names.
- Type **R1 <configuration filename>.CFG <command filename>.RCL** to load Reflection with a configuration other than the default and execute a command file.

2.3 Changing from HP to VT Emulation

The Global Configuration menu allows you to toggle between HP and DEC terminal emulation. As long as you have not put Reflection into background mode, you may switch at any time.

To run the VT emulation at the DOS prompt, use **R1/V** or **R7/V**. To load the HP emulation regardless of the defaults stored in the configuration file, use **R1/H** or **R7/H**.

2.4 Setting the DOS Environment

You can modify your AUTOEXEC.BAT file to tell Reflection the following:

- The directory for Reflection-related files
- The switches to include when loading Reflection
- The directory and name for the state file

These commands can be issued at the DOS prompt or included in your AUTOEXEC.BAT file. They are explained in detail below.

You can also set up some of these options as part of the Reflection installation procedure.

Setting the Directory for Reflection Files

You can modify your AUTOEXEC.BAT file so that you can run Reflection regardless or where you are in DOS. Modify the PATH command by adding the name of the Reflection directory. For example, if the Reflection files are in the directory **C:\R1**, add it as follows to your path:

```
PATH=C:\DOS;C:\TXT;\R1
```

The DOS PATH command tells DOS which directories to look in to find a program that you are trying to run.

The DOS PATH command lets you indicate the path to search when the *
program you are trying to run is not in the current DOS directory or drive. In *
DOS 2.0, however, it does not provide a means for finding related support *
files, even if they are in the same directory as the program. Reflection needs to *
be able to find certain support files and have immediate access to the overlay *
structure. Use one of the following DOS SET commands, depending on which *
product you are using: *

```
SET REFLECT1=[drive:][path]
SET REFLECT7=[drive:][path]
```

The **[drive:][path]** indicates the possible location of these support files if they are not in the current directory:

- R1.CFG and R7.CFG
- R1.HLP and R7.HLP
- Localization files: R1?.LOC and R7?.LOC
- R7.FT
- An invoked or chained command file

When a Reflection file cannot be found in the current subdirectory, Reflection *
tries the drive/path specified by the DOS SET command. This command is *
not allowed on all PCs; HP 150s and Vectras running under PAM, for *
instance, do not support it. *

Adding Load Switches

Switch information is not stored in Reflection's configuration files. If you want to ensure that a particular switch value is always used when you load Reflection, use the following DOS SET command (you may add also it to your AUTOEXEC.BAT file):

```
SET R<n>SW=<switches>
```

where **<n>** is 1 or 7, depending on the product you are using. For example, if you are running Reflection 7 and want to suppress Reflection's copyright screen and always take advantage of Reflection's expanded memory support, include the line below in your AUTOEXEC.BAT file:

```
SET R7SW= /Q /E
```

Any number of switches can be used in this context.

Setting the Directory and Name of the State File

Reflection's state save feature ([Alt]-[B]) lets you quit Reflection and then *
later return to the same point by saving the state of Reflection to *
R<n>STATE.STA in the directory in which Reflection resides. *

You can specify a different filename and directory to use by issuing the *
following DOS SET command (add it to your AUTOEXEC.BAT file if you *
want it to be in effect whenever you start up): *

```
SET R<n>STATE=<filespec>
```

Specifying where your state file should be stored is especially important if you *
run Reflection on a file server. Without this command in your *
AUTOEXEC.BAT file, a state file is stored on the file server, in the same *
directory as the Reflection executable program. This means that other users *
could end up with your file. Use this SET command to store the file in a *
directory on your PC. *

State save is useful if you want to exit Reflection temporarily without leaving *
it resident in memory. State save is covered in detail on page 329. *

Modifying CONFIG.SYS

In the root directory of the boot drive on your PC (the same location as your AUTOEXEC.BAT file), there should be a file called CONFIG.SYS. Include the the following commands in the file:

```
FILES=20
BUFFERS=20
```

The recommended minimum for files and buffers is *20*. Performance may improve with buffers set higher than *20*.

2.5 Function Key Labels

After you have established basic configuration information, the first key labels displayed each time you run Reflection are the *system keys*. These keys give you access to most of the basic functions provided by Reflection. You can return to the system keys by pressing [F10] at any time. These labels differ according to the emulation you select:

device control	margins/ tabs/col	basic config	modes keys	COMMAND LINE	file transfer	HELP	EXIT

Figure 1: HP System Keys

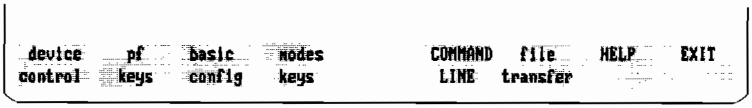


Figure 2: VT System Keys

- The **basic config** label on the system keys leads to the Basic Configuration menu. [Alt]-[C] also displays the configuration key labels.
- The **modes keys** labels indicate the status of the major operating modes and allow you to switch modes. [Alt]-[M] displays these keys.
- Use [F9] to display user keys. These keys can be configured by you or by HP host programs. You can also define special functions in user keys in either HP or VT mode. See *User Key Configuration* on page 161.
- The Help Screen is available by pressing [F7] from the system keys menu, or by using [Alt]-[H].
- The **pf keys** label (VT only) leads to the PF key and cursor control key functions of the VT series of terminals. [Alt]-[P] also displays these keys in VT mode.

The following figures summarize Reflection function key labels.

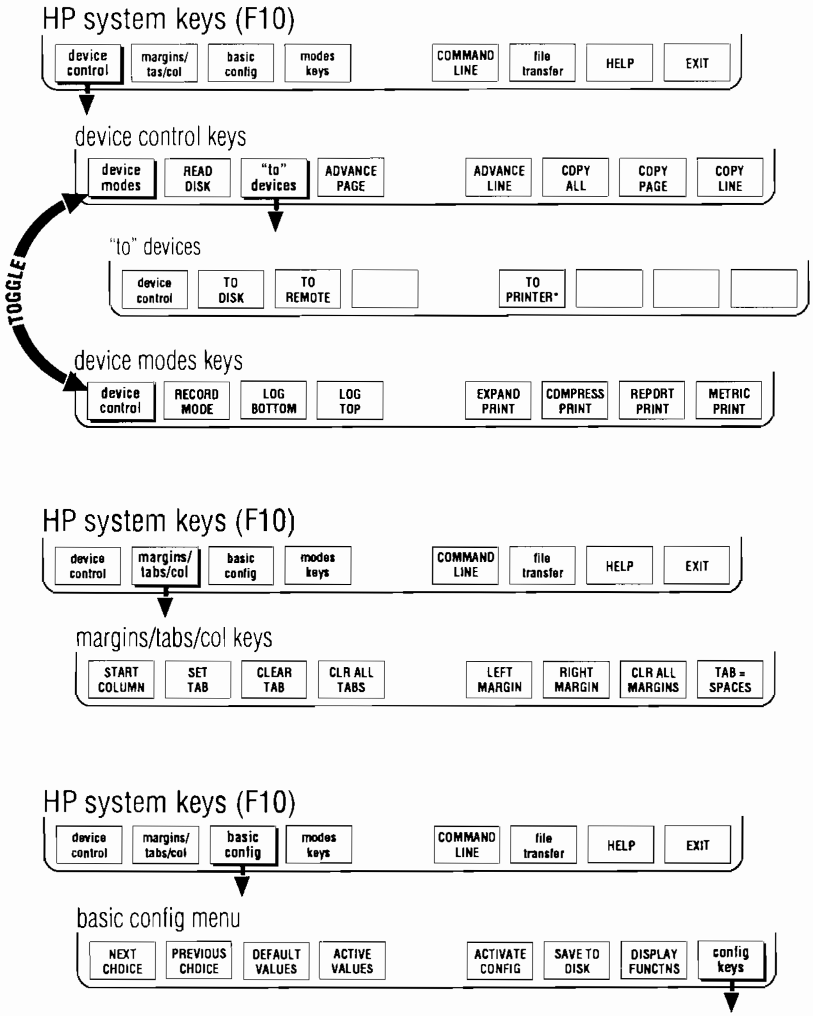
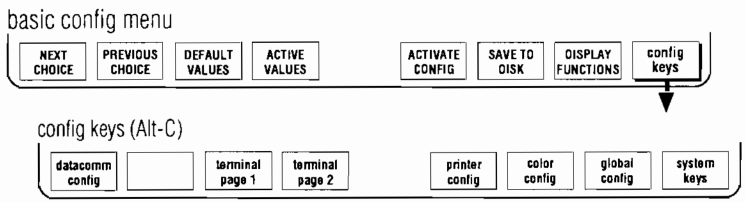
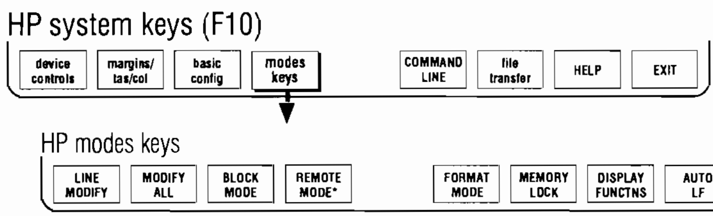
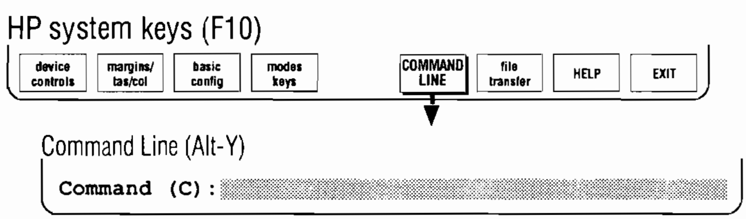
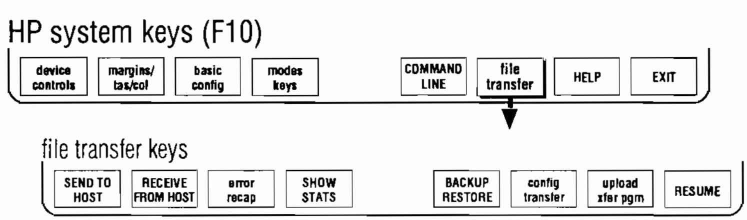


Figure 3: HP Function Key Label Tree


basic config menu
NEXT CHOICE
PREVIOUS CHOICE
DEFAULT VALUES
ACTIVE VALUES
ACTIVATE CONFIG
SAVE TO DISK
DISPLAY FUNCTIONS
config keys
config keys (Alt-C)
datacomm config
terminal page 1
terminal page 2
printer config
color config
global config
system keys


HP system keys (F10)
device controls
margins/ tabs/col
basic config
modes keys
COMMAND LINE
file transfer
HELP
EXIT
HP modes keys
LINE MODIFY
MODIFY ALL
BLOCK MODE
REMOTE MODE*
FORMAT MODE
MEMORY LOCK
DISPLAY FUNCTNS
AUTO LF


HP system keys (F10)
device controls
margins/ tabs/col
basic config
modes keys
COMMAND LINE
file transfer
HELP
EXIT
Command Line (Alt-Y)
Command (C) :


HP system keys (F10)
device controls
margins/ tabs/col
basic config
modes keys
COMMAND LINE
file transfer
HELP
EXIT
file transfer keys
SEND TO HOST
RECEIVE FROM HOST
error recap
SHOW STATS
BACKUP RESTORE
config transfer
upload xfer pgm
RESUME

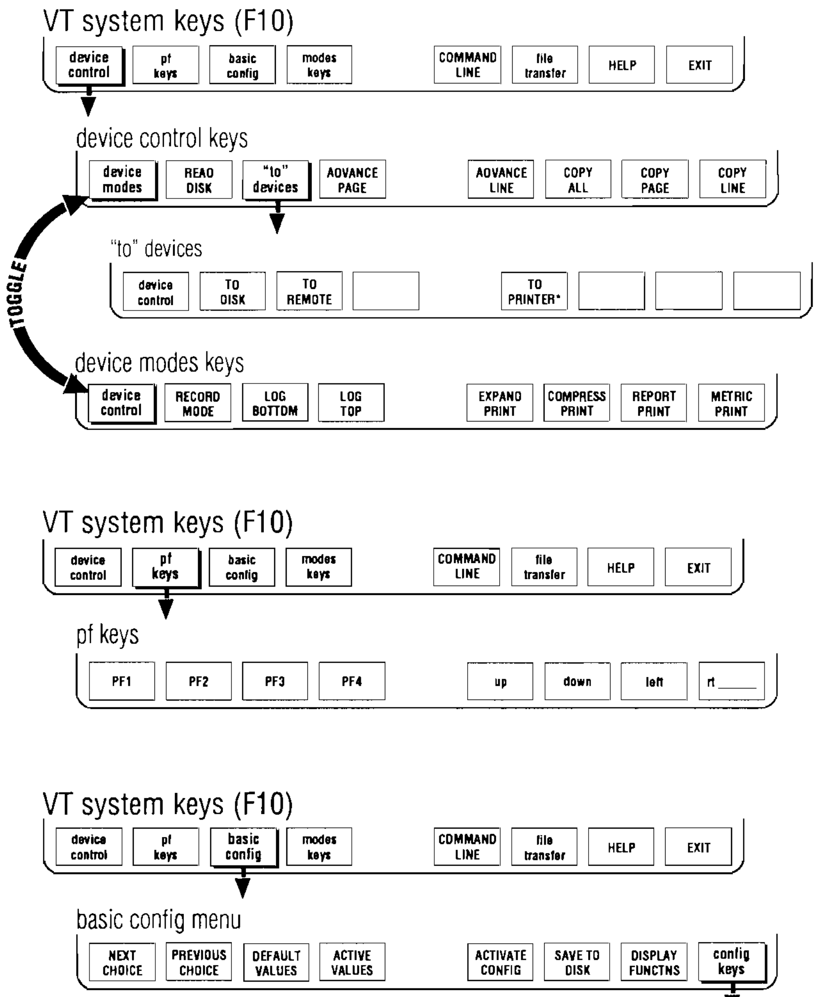
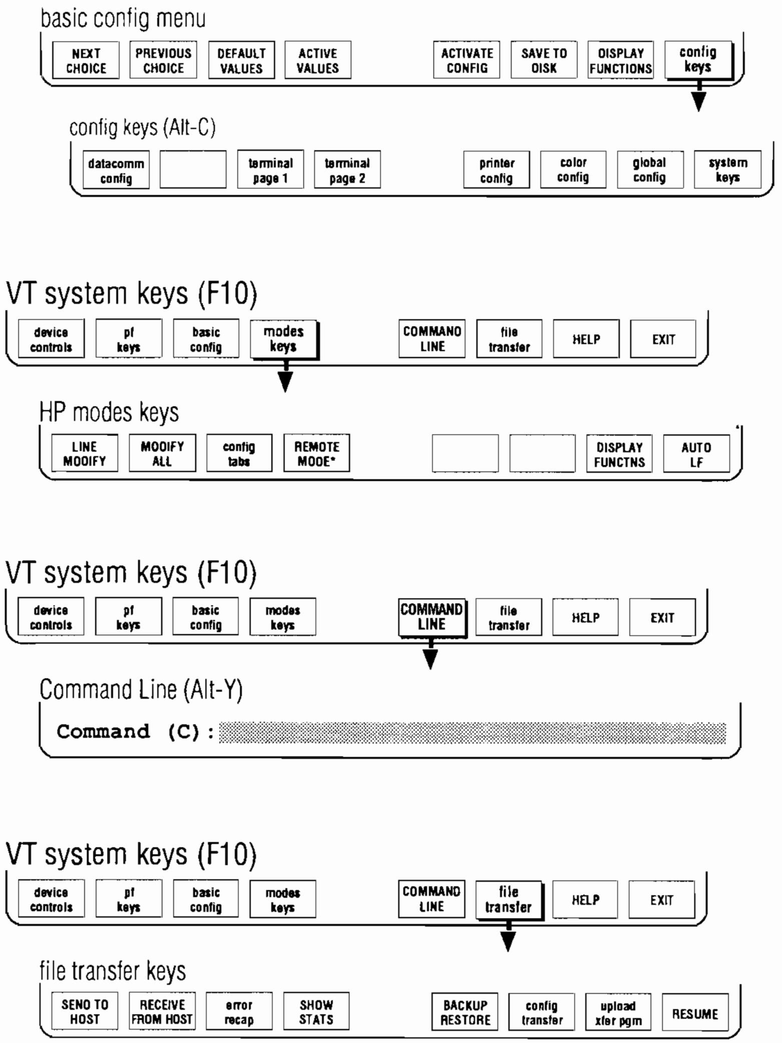


Figure 4: VT Function Key Label Tree



basic config menu
NEXT CHOICE
PREVIOUS CHOICE
DEFAULT VALUES
ACTIVE VALUES
ACTIVATE CONFIG
SAVE TO DISK
DISPLAY FUNCTIONS
config keys
config keys (Alt-C)
datacomm config
terminal page 1
terminal page 2
printer config
color config
global config
system keys
VT system keys (F10)
device controls
pf keys
basic config
modes keys
COMMAND LINE
file transfer
HELP
EXIT
HP modes keys
LINE MODIFY
MODIFY ALL
config tabs
REMOTE MODE*
DISPLAY FUNCTNS
AUTO LF
VT system keys (F10)
device controls
pf keys
basic config
modes keys
COMMAND LINE
file transfer
HELP
EXIT
Command Line (Alt-Y)
Command (C) :
VT system keys (F10)
device controls
pf keys
basic config
modes keys
COMMAND LINE
file transfer
HELP
EXIT
file transfer keys
SEND TO HOST
RECEIVE FROM HOST
error recap
SHOW STATS
BACKUP RESTORE
config transfer
upload xfer pgm
RESUME

Understanding Indicators

Letters are used between the fourth and fifth function key label at the bottom of the screen to indicate a current state, such as CapsLock mode or insert mode. Below is a list of each indicator and its meaning:

B Indicates Reflection has been installed as a memory-resident program (also called the *background* indicator).

C Indicates CapsLock mode is on.

G Indicates Ctrl - - (keypad) has toggled the keypad into graphics mode. This only appears in HP mode in Reflection 7.

H Indicates the display has been stopped. Pressing Alt - ScrollLock in HP mode or ScrollLock in VT mode restarts the display.

I Indicates insert mode is on.

K Indicates the keyboard is locked. Some applications lock and then unlock the keyboard; when they do so, the **K** appears temporarily. If the keyboard becomes locked for any other reason, you can unlock the keyboard by pressing Alt - S, soft reset or Alt - R, hard reset.

N Indicates NumLock is on. The keypad generates numbers or VT keypad functions in this state depending on your emulation.

S Indicates scrolling has been enabled. Pressing Alt - ScrollLock in VT mode or ScrollLock in HP mode allows you to scroll display memory with the cursor keys, without holding down Ctrl.

* Indicates that the selected status line to monitor for a modem (identified on the Datacomm Configuration menu) is active.

The two numbers in the area between the fourth and fifth function key labels indicate the cursor's location in display memory by row and column. Each time the cursor moves, the row or column numbers change to indicate its new position.

2.6 The Help Screen

Press [Alt]-[H] from anywhere within Reflection or [F7] from the system keys to bring up the Help screen. The Help screen lists the most commonly used keystrokes in Reflection. You can also refer to the small template or keystroke summary card included in your package.

Reflection 1 PLUS   Version 4.0   Serial # 101+400L123456
Copyright (C) 1983, 1985-1990 Walker Richer & Quinn, Inc.

FUNCTION	DEFAULT KEY	FUNCTION	DEFAULT KEY
Break	Ctrl Break	Modes keys	Alt M
Clear display/line	Alt J/Alt K	Move cursor	Arrow keys
Clear print buffer	Ctrl F1	Next/prev page	PgDn/PgUp
Command line	Alt Y (type help)	Print screen	Alt PrtSc
Config keys	Alt C	Print screen (DOS)	Shift PrtSc
Define user keys	Ctrl F9	Reset typeahead	Alt T
Delete char/line	Del/Alt D	Roll screen	Ctrl arrow key
Disconnect	Ctrl F10	Soft reset	Alt S
Error recap	Alt E	State save	Alt B
Exit to DOS	Alt X	Stop key	Alt Scroll Lk
Function key recall	Alt F	System keys	F10 or Alt A
Hard reset	Alt R	To bottom of screen	Ctrl PgDn
Home up/down	Ctrl Home/End	To start/end of line	Home/End
Help (this screen)	Alt H	To top of screen	Ctrl PgUp
Insert char/line	Ins/Alt I	Toggle user keys	F9
Hot key	Alt Rsh	User keys	F9 or Alt U

RESUME

Figure 5: HP Help Screen

Reflection 7 PLUS   Version 4.0   Serial # 107+400L123456
Copyright (C) 1983, 1985-1990 Walker Richer & Quinn, Inc.

FUNCTION	DEFAULT KEY	FUNCTION	DEFAULT KEY
Answerback	Ctrl F2	Hot Key	Alt Rsh
Break	Ctrl Break	Insert char/line	Ins/Alt I
Clear display	Alt J	Keypad	NumLock-keypad
Clear line	Alt K	Modes keys	Alt M
Clear print buffer	Ctrl F1	Move cursor	Arrow key or
Command line	Alt Y (type help)		Alt P & F5-F8
Config keys	Alt C	Next/Prev page	Ctrl PgDn/PgUp
Define user keys	Ctrl F9	PF1 - PF4	Alt P & F1-F4
Delete char/line	Del/Alt D	Print screen	Alt PrtSc
Disconnect	Ctrl F10	Print screen w/labels	Shift PrtSc
Error recap	Alt E	Roll screen	Ctrl arrow key
Exit to DOS	Alt X	Soft reset	Alt S
Hard reset	Alt R	State save	Alt B
Help (this screen)	Alt H	System keys	F10 or Alt A
Hold screen	Scroll Lock	Toggle print logging	Ctrl PrtSc
Home up/down	Ctrl Home/End	User keys	F9 or Alt U

RESUME

Figure 6: VT Help Screen

2.7 Exiting Reflection

There are a number of ways to exit Reflection:

Exit ([Alt]-[X])

You can exit by pressing [Alt]-[X] from anywhere in Reflection. Using this keystroke uninstalls Reflection from memory and returns to the DOS prompt or to an open application.

Hot-Key ([Alt]-(right) [Shift])

Use the hot-key to keep Reflection installed in memory when you toggle to DOS. You can then run other applications while Reflection continues to run. Press the hot-key again to get back to Reflection without reloading.

State save ([Alt]-[B])

If you do not have enough memory to run both Reflection and another program, you can exit Reflection using state save. *
*

State save uninstalls Reflection from memory, yet saves all information *
about your current session to a file. The default filename is *
R<n>STATE.STA, stored in the same directory as R1.EXE (or *
R7.EXE). You can change the name or path of the state save file by *
using SET R<n>STATE=<filespec>. See page 21. The next time you *
run Reflection, your session is restored from the saved file. *

All of these methods leave the PC logged on to the host computer if the connection to the host is still active.

Chapter 3

Configuration Files

At start-up, Reflection looks for a configuration file. If you type **R1** (or **R7**) at the DOS prompt with no filename, Reflection looks on the default drive for R1.CFG (or R7.CFG). This file is the default configuration file.

Once Reflection has been reconfigured, the new configuration values can be saved to a disk file for future use. Save commonly used values to the default configuration file.

Any number of other configuration files can be saved for different circumstances, under different filenames. To use a configuration file you've previously saved, follow the **R1** (or **R7**) command at the DOS prompt with the name of the configuration file you want to use. For example, to use the configuration file MINE.CFG, type **R1 MINE.CFG** at the DOS prompt and press [Return].

Once you've successfully connected the PC to the host, you are ready to use your PC as a terminal.

3.1 Basic Configuration

You can configure a variety of items in Reflection, but typically only a few must be changed from the defaults. To simplify the configuration process, a single Basic Configuration menu is presented when you first load Reflection or when you load Reflection with the **/D** (use default) switch. Once you save new settings to the default configuration file, Reflection comes up with those values rather than the defaults.

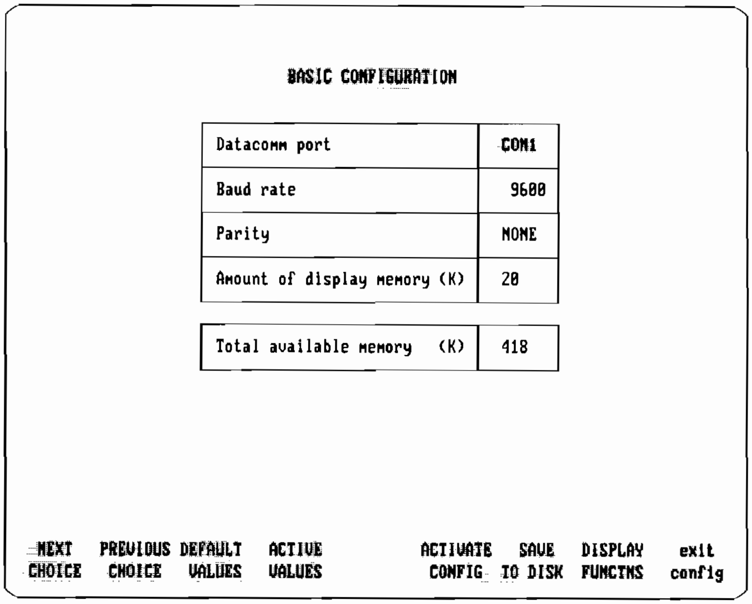


Figure 7: Basic Configuration Menu

Datacomm port	Indicates whether you are connected through a serial port or a LAN.
Baud rate	The baud rate matches that of your host if you are directly connected; if you are using a modem, it matches the modem's baud rate.
Parity	HP 3000s typically use no parity.
Amount of display memory (K)	This field allows you to set the size of Reflection's display memory. 20K is the default display memory size.
Total available memory (K)	The difference between the display memory size and the total memory available reflects the amount of memory available for running other

programs. You cannot change this field; it is calculated by Reflection. The calculation does not apply after you have entered background mode. See page 327 for memory management techniques.

Read the *Quick Start Guide* and the *Host Connections* section, which begins on page 39, for more information on configuring your connection.

The *Configuration* section starting on page 86 includes chapters on each of the configuration menus that allow you to modify your configuration.

3.2

Contents of a Configuration File

The configuration file contains separate sets of parameters for HP and VT emulation. When switching from one emulation to another while running Reflection, the configuration values change accordingly.

A configuration file contains:

- All configuration menu values
- The current definitions of the user keys (see page 161)
- The initial settings for remote mode, auto linefeed, block mode, and modify all
- Most values set with a SET command (see the Reflection *Command Language Manual*)
- Keyboard remapping information
- Video information *

Every time configuration parameters are changed and saved to a configuration file, the current values of all the other parameters listed above are re-saved. Type **VERIFY** at the command line to view current SET command values.

3.3

Configuration Menus

Reflection has several configuration menus that allow you to specify data communications parameters, terminal operating modes, printer characteristics, color configuration, and global parameters. Alt-C brings up the configuration keys anywhere in Reflection.

* Since video enhancements and colors are saved to the configuration file, you are not always able to share configuration files among different video environments (e.g., between color and monochrome monitors).

Configuration Files

datacomm config		terminal page 1	terminal page 2	printer config	color config	global config	system keys

Figure 8: Config Keys (Reflection 1)

datacomm config	plotter config	terminal page 1	terminal page 2	printer config	color pairs	global config	system keys

Figure 9: Config Keys (Reflection 7)

These keys and their associated menus may vary according to the product or emulation you're using:

- The **datacomm config** key brings up data communications parameters. VT and HP data communications menus are identical, but the default settings vary. Separate values are maintained for each in the configuration file.
- The **plotter config** key is available in Reflection 7 only.
- There are two separate terminal configuration menu pages for both HP VT modes.
- Reflection 7 defaults to a *color pairs* menu for color configuration options. You can force Reflection 7 to use the Reflection 1 style of color configuration; see page 145.

Configuration Values

A configuration menu field can have three values: the one currently displayed, the default value, and the active value.

Most of the configuration fields allow you to choose from a predefined set of values simply by pressing $\boxed{\text{F1}}$ for next choice or $\boxed{\text{F2}}$ for previous choice. A few require you to type in a value.

Using Function Keys in Configuration

The following table lists configuration menu labels and the actions performed by the corresponding function keys.

Table 1
Configuration Menu Labels

Label	Key	Action
NEXT CHOICE	F1	Changes value of the field to the next choice, unless a value must be typed in.
PREVIOUS CHOICE	F2	Changes value of the field to the previous choice, unless a value must be typed in.
DEFAULT VALUES	F3	Changes all fields on the screen to their default values.
ACTIVE VALUES	F4	Changes all fields on the screen to the currently active values, i.e., the values that were most recently saved or activated.
ACTIVATE CONFIG	F5	Activates values currently shown on the screen, clears the menu and displays the configuration keys.
SAVE TO DISK	F6	Asks for a filename for the configuration file. Saves all config values to that file.
DISPLAY FUNCTNS	F7	Where available, toggles display functions mode on and off so that ASCII control codes can be entered into menu fields that require keyboard input.
config keys	F8	Clears the configuration menu from the screen and displays the configuration keys.

Configuration Files

3.4

Saving a Configuration

Every configuration menu contains the label **SAVE TO DISK**. When this key is pressed, Reflection asks for a filename for the configuration file, as follows:

```
Enter filename: C:\REFLECT\<filename>.CFG
```

R1.CFG and R7.CFG are the default configuration filenames. If either of these names is shown and you want the current values to be automatically set the next time you load Reflection, press Return. If the filename presented happens to be other than R1.CFG (or R7.CFG) and you want to have the current values be the defaults, type in **R1.CFG** or **R7.CFG** and press Return.

Multiple Configuration Files

To supply a different configuration filename, type in a different name directly over the name shown, then press Return. The filename extension .CFG should be used for all configuration files.

For example, you might want to save values for occasional use only, such as dialing a modem. Type in a filename such as MODEM.CFG and press Return. You can invoke this configuration file when you load Reflection by typing **R1 MODEM.CFG** or **R7 MODEM.CFG** at the DOS prompt. You can also load the file from within Reflection by typing **LOAD MODEM.CFG** at the command line. See the LOAD command in the Reflection *Command Language Manual.*

Reflection configuration files always include both HP and VT values. If you create a new configuration file and expect to use both emulations, follow the directions on page 63.

Connections

Section 2

Host Connections

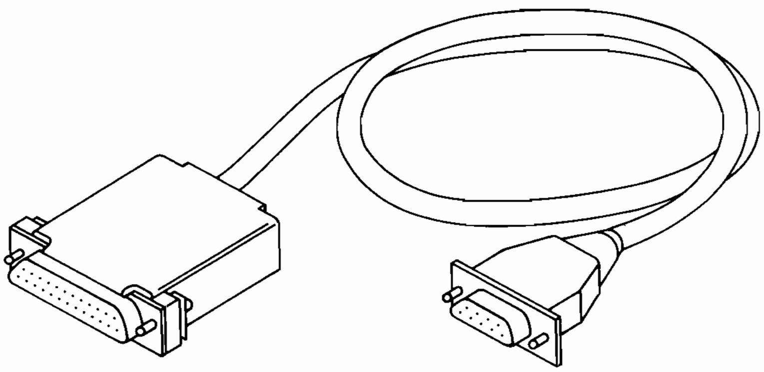


Figure 11: RS-232 25-Pin to 9-Pin Serial Interface

The adapter cable should have the following pin configuration:

Table 2
RS-232 AT Cable Wiring Scheme

25-Pin Pin #		9-Pin Pin #	Signal Name
8	---------	1	Data Carrier Detect (DCD)
3	---------	2	Receive Data
2	---------	3	Transmit Data
20	---------	4	Data Terminal Ready (DTR)
7	---------	5	Signal Ground
6	---------	6	Data Set Ready (DSR)
4	---------	7	Request to Send (RTS)
5	---------	8	Clear to Send (CTS)
22	---------	9	Ring Indicator

When the connection to the host is made via direct-connect cable and Reflection is loaded, the PC is ready to be used as a terminal.

If you are configuring Reflection to communicate with a VM/370 or MVS/TSO, which is described on page 57, you may require modem control in your cabling (DTR, DSR, RTS, and CTS).

Host Connections

4.2

Locating the Serial Ports

The IBM PC's serial port is a male 25-pin D-shell connector at the back of the computer. (Don't confuse it with the parallel printer interface, which is a female 25-pin D-shell connector.)

The IBM PC and most compatibles use a 9-pin or 25-pin male port for connection to the host computer. The IBM PC-AT uses a male 9-pin D-shell connector on its serial port. Both are illustrated below:

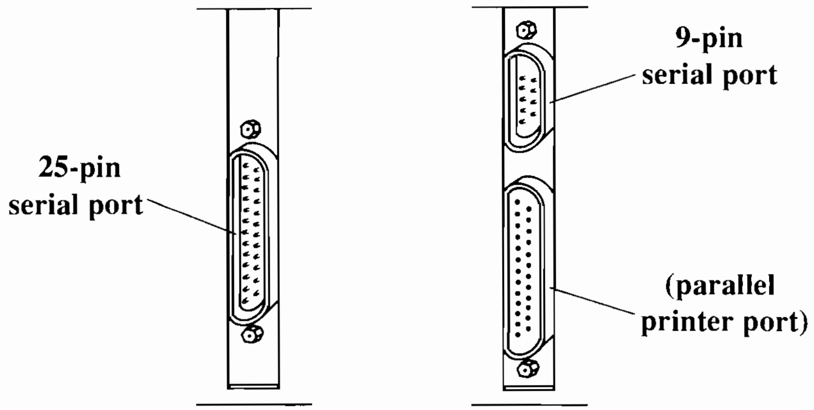


Figure 12: IBM 25-Pin and 9-Pin Serial Ports

Selecting the Serial Port

ATs may have up to four serial communications ports (called COM1, COM2, COM3, and COM4). PS/2s have up to eight communications ports. The other serial ports can be used to drive serial devices, such as a printer.

4.3

HP 3000 Configuration

Reflection's default configuration values are correctly set for an HP 3000. See page 89 for all datacomm configuration options. Check the fields listed below.

Datacomm Configuration Menu

Datacomm port

The **Datacomm port** field must be configured for the serial port you've selected for data communications or for a LAN connection. The default value is *COM1*.

Baud rate

The **Baud rate** you specify must match the baud rate of your modem or of your direct connection.

Receive pacing

The **Receive pacing** field must be set to *XON/XOFF* to enable the hold (or STOP) key function, which allows you to stop and start the display of data in display memory. In HP mode use [Shift] or [Alt] with [ScrollLock]; in VT mode use [ScrollLock].

Terminal Configuration Menus

Type ahead

Type ahead (Page 1), described on page 103, lets you type ahead of the HP 3000. [Alt]-[T], reset typeahead, may need to be used occasionally to obtain a host prompt after logging off or if handshaking must be reestablished. [Alt]-[V] clears the keyboard buffer.

Character xmit delay

Some models of the HP 3000, notably the Series 30 and 33, can't receive data blocks at 9600 baud (the default value) without danger of lost data. Reflection allows you to specify a time delay between character transmissions, to prevent overrunning the HP 3000. For a Series 30 or 33, specify a **Character xmit delay** of *3* milliseconds on the Terminal Configuration menu (Page 2). The PC can then receive data at a full 9600 baud, while cutting the effective transmission rate to the host to about 240 characters per second.

4.4 VAX/VMS Configuration

Reflection's default configuration values for VT emulation are correctly set for VAX/VMS. Only a few fields must be checked on the Basic Configuration Menu:

Datacomm port

The **Datacomm port** field must be configured for the serial port or LAN you've selected for data communications. The default value is *COM1*.

Baud rate

The **Baud rate** you specify must match the baud rate of your modem or of your direct connection.

Host Connections

4.5 UNIX System Configuration

Reflection's default configuration values *are not* set for a UNIX system. Besides the Basic Configuration menu settings, you must change the fields described below if you are using HP emulation.

Datacomm Configuration Menu

Pacing

Receive pacing and **Transmit pacing** must both be set to *XON/XOFF*. See page 89.

Terminal Configuration Menus

Inhibit handshake and **Inhibit DC2**

Inhibit handshake and **Inhibit DC2** (Page 1) should both be set to *YES*.

Use host prompt

Use host prompt (Page 2) should be set to *NO*.

Walker Richer & Quinn provides the host file transfer program for UNIX systems. See page 352 to configure for UNIX file transfers for both HP and VT emulation.

4.6 HP 1000 Configuration

Cabling

HP 1000s usually do not have pins 2 and 3 crossed at the host. Use a null modem cable (where pins 2 and 3 are crossed) rather than that recommended for HP 3000s.

Basic Configuration Menu

Check the baud rate and parity as discussed for HP 3000 and VAX hosts.

The **Parity** field is often *EVEN* on HP 1000s. Check to see what parity your system uses.

Reflection supports both XMODEM and KERMIT for file transfers to an HP 1000.

4.7

Logging On to the Host

Logon procedures are dictated by the host computer with which you're communicating, not by Reflection, so you'll need to refer to the host computer's user manual.

After establishing a connection to the host computer, press [Return] until a prompt appears on the screen. Then log on and begin using the PC as a terminal.

Logging Off

As with logging on, the logoff procedure is dictated by the host, not by Reflection.

It's possible to log off the host computer without exiting Reflection. If the host is connected via modem, determine whether the host computer breaks the phone connection automatically when logged off (the HP 3000 and VAX operating systems usually do). If it doesn't, enter the modem's *hangup* command (usually **ATH**) after logging off.

You can exit Reflection and still be logged on to the host system. See page 30 for ways to exit Reflection.

Chapter 5

Connecting via Modem

Reflection must be properly configured and in remote mode to communicate via modem with the host. You must set the modem's baud rate* and assign a datacomm port on the Basic Configuration menu or the Data Communications Configuration menu. Modem operations vary; be sure to read your modem's documentation before using Reflection.

There are two types of modems you can use to connect the PC to a host computer:

- An external modem—either a direct-connect or an acoustic coupler style.
- An internal modem (an interface card inside the PC).

5.1 External Modem Wiring Connections

Connect an external modem to the PC serial port using an RS-232 modem cable. The following table shows the proper wiring scheme for this cable.

* The Hayes Smartmodem 1200 automatically detects the baud rate when you type, so you do not need to reconfigure this even when it's different from the default setting.

Host Connections

Table 3
RS-232 External Modem Cable Wiring Scheme

Serial Port Pin #		Modem Pin #	Signal Name
1	------------------------	1	Protective Ground
2	------------------------	2	Transmit Data
3	------------------------	3	Receive Data
4	------------------------	4	Request to Send (RTS)
5	------------------------	5	Clear to Send (CTS)
6	------------------------	6	Data Set Ready (DSR)
7	------------------------	7	Signal Ground
8	------------------------	8	Data Carrier Detect (DCD)
20	------------------------	20	Data Terminal Ready (DTR)

5.2 Auto-Dial Modems

An auto-dial modem is always in one of two functional states:

Local Command State
The modem expects to receive its commands locally from the PC, such as *dial the phone* or *hang up*. This state is invoked when you first power on the modem.

Online State
The modem transmits whatever it receives from the host over the phone connection, and passes this data from the phone line to the terminal (i.e., the PC running Reflection).

You can switch between the two modem states using a special sequence of characters, known as an *escape code*. The default escape code for the Hayes Smartmodem is +++; a pause of one second is required before and after the escape code. When you type in +++, use the [+] in the upper keyboard row, not the keypad [+] key.

Establishing a Connection

If you have a Hayes Smartmodem (or similar product), the following steps get you started using Reflection.

1. Turn the modem on.

2. Run Reflection, using a configuration file that contains the proper baud rate and datacomm port selection, etc. Verify that Reflection is configured correctly for your equipment.

3. From the system keys, press F4 to display the modes key labels.

4. If there is not an asterisk in the remote mode label, press F4.

5. A command file called *DIAL* is included on your Reflection disks. This file automates the procedure that checks your configuration and dials the modem for you.

   Press Alt-Y to get to the Reflection command line, then invoke the command file by typing **DIAL** followed by a phone number. Use a comma between prefix numbers to allow for any necessary pauses while dialing. For example, to dial the WRQ bulletin board, type the following:

   ```
   DIAL 1,206,3228047
   ```

6. If the modem connection is valid, **ATDT<phone#>** appears on the screen.

7. You'll hear the phone being dialed, the ring, the answer, and finally a carrier tone. When **CONNECT** appears on the screen, press Return to get a host prompt and log on as you would normally.

8. When you finished the host session, log off.

   Some host systems drop the carrier signal when you log off; the modem responds with **OK**. If your system doesn't automatically drop the signal, you can hang up by pressing Ctrl-F10 or by typing **ATH** followed by Return.

5.3 Resetting the Modem

A 1200 or 2400 baud rate modem, and the Hayes Smartmodem 1200B internal modem card, may occasionally need to be reset. If you're experiencing problems, it often helps to reset the modem.

Put your modem in command mode. Execute one of the following commands to reset it:

Host Connections

Table 4
Modem Reset Methods

Modem Setup	What to Type
1200 baud	ATZ*
2400 baud	AT&F
Internal	Type **RESETCOMM** on the Reflection command line or press [Ctrl]-[F8] (connection reset)

RESETCOMM resets the internal card by setting the *out* bit of the modem control register to 1 for about 50 milliseconds.

5.4 Function Keys for Modem Commands

Since Reflection saves a set of user-defined function keys in its configuration files, it is convenient to load the modem dialing commands and logon commands into the user keys. Then when you run Reflection, you can dial the phone or log on with a single keystroke. See page 161, *User Key Configuration*.

When defining modem and logon commands, observe the following guidelines:

- The attribute field must be set to **N** (normal keyboard input).
- Commands to the Hayes Smartmodem must be in uppercase.
- Type a [Return] at the end of the key definition, so that you don't have to press [Return] after you press the user key. Enable **DISPLAY FUNCTNS** before you press [Return].
- Only one command, terminated by a carriage return, can be entered for any one user key, unless **Type ahead** on the Terminal Configuration, Page 1 menu, is configured to *YES*. When **Type ahead** is enabled, several command lines, each terminated by a carriage return character, can be included in a single user-key definition. One exception to this statement is the modem dialing command, which must be the only command in its user-key definition.

* Turning off a 1200 baud modem also resets it.

Host Connections

5.5 Hanging up the Modem

Determine whether the host computer breaks the phone connection automatically when you log off. If the connection is not broken, use one of the procedures below.

Table 5
Modem Hangup Methods

Modem Type	Procedure
Hayes External or Internal	Pause 2 seconds and type +++ Pause for OK and type **ATH**
External or Internal	[Ctrl]-[F10]
Hayes Internal	[Ctrl]-[F8]
External	Power off
Internal	Unplug jack

If your modem is configured to disconnect when the data terminal ready (DTR) signal is dropped, you can disconnect your modem by typing [Alt]-[X]. DISCONNECT-ON-EXIT must be set to *YES*.

5.6 Microcom's Network Protocol

Walker Richer & Quinn has developed a serial port driver which essentially turns a non-error-checking modem, such as the Hayes 1200 Smartmodem, into an error-checking modem that uses Microcom's Networking Protocol to achieve error-free communications. The MNP protocol ensures that all data which passes over a communications line is transferred error-free, regardless of line noise.

To take advantage of a modem which uses MNP, you must provide either a second MNP modem, or use the Reflection Complement, MNP® SoftModem, on the communications port of the PC using a non-MNP modem. The MNP® SoftModem is an option which must be purchased separately. It is significantly less expensive than purchasing a second MNP modem.

Note: MNP support includes classes 2 and 4. Class 3 requires synchronous communications. Other classes are negotiated during the initial handshaking.

Host Connections

Support for High Speed Modems

New control standards for high-speed modems (some 2400s and all 9600s) require flow control to ensure data is not lost during transfer sessions between computers. For example, the Microcom MNP modems default to XON/XOFF flow control with the terminal (PC) and the host computer. Because XON/XOFF causes problems with some hosts and many file transfer protocols, there is another flow control standard that has emerged and is supported by Microcom and others.

This alternative form of hardware flow control uses the RTS and CTS pins to control data flow. If you are using a high speed modem that supports RTS/CTS flow control, make the following changes on the Datacomm Configuration menu:

Receive pacing	*HARDWARE*
Transmit pacing	*HARDWARE*
CTS required	*NO*
DSR required	*NO*

You must also configure the modem to use this protocol.

5.7 Troubleshooting

If you have an external modem and cannot establish a connection, use the modem's indicator lights to help diagnose the problem. If you have an indicator marked **SD** (send data) or **TD** (transmit data), this light should flash each time you press a key.

If an indicator marked **RD** (receive data), flashes after you press a key, this means that an echo is coming back from the modem. If the **SD** light flashes but the **RD** does not, then the problem probably lies with the modem. However, this could also mean that the host computer is not echoing, either because it is not accepting input or it's operating in half-duplex mode.

The Transmit Status Indicator

The **Xmit indicator** (*) field in the Datacomm Configuration menu allows you to monitor your modem status line. If the selected line (*pin*) is in a *true* state, an asterisk is displayed at the bottom of the screen between function key labels 4 and 5. The default for the indicator is *OFF*. See *Modem-Related Fields* on page 97. You can monitor your modem status line with the **Xmit indicator** (*) set to *ON*, but be aware that some modems can be configured to force the line to a *true* state.

Chapter 6

Network Options

6.1 Using AdvanceNet (OfficeShare)

To run Reflection over OfficeShare, install the OfficeShare hardware and software as explained in the HP OfficeShare Network Installation and Configuration manual. Run the USRLOAD program as described in the HP OfficeShare Network User's Guide. After running Reflection, bring up the Basic Configuration menu. Select and activate *ADV.NET* in the **Datacomm port** field.

Walker Richer & Quinn's Command Interpreter prompt for AdvanceNet appears in display memory as the > character. At the prompt enter the command **CONNECT <nodename>**, where <nodename> is the name assigned by the HP 3000 network manager to the HP 3000 you wish to communicate with. The Command Interpreter returns a *Connected* message for a successful connection, or an error message for an unsuccessful connection. Once the connection is established, proceed as usual. There are two ways to break a connection:

- Ctrl - F10 forces a disconnect from the PC side
- **:BYE** forces a disconnect from the HP 3000 side

When a connection is broken, Reflection displays *Disconnected* and returns to the Command Interpreter.

By default, **Packet size** is set to the maximum (1280 for HP emulation). This is also the recommended packet size for AdvanceNet.

6.2

Office Extend

Support for Fransen/King Ltd.'s Office Extend Network is included in both PLUS and non-PLUS versions of Reflection 1 and 7. *

To run Reflection with Office Extend Network, make sure you have version C.Ø of the Office Extend software program and that it is installed and activated as explained in your Office Extend documentation. After you load Reflection, bring up the Datacomm Configuration menu and select *EXTEND* in the **Datacomm port** field. Activate or save this change to your configuration file. *

Once the *EXTEND* selection is made, other datacomm parameters— such as baud rate, parity, and pacing—are controlled by Office Extend. To use Reflection and Office Extend at the same time, Reflection *must* be configured with *EXTEND* as the **Datacomm port** selection. *

Chapter 7

Set-Up for IBM Mainframes and Other Hosts

7.1 Communicating with an IBM Mainframe

Many protocol converters provide a 3270 terminal emulation mode, allowing an ASCII terminal to emulate a 3270 display station for communication with an IBM mainframe. Typically, the protocol converter displays a terminal type menu where you select the terminal type you will be using to make the connection to the mainframe. A common default menu choice is a VT100. The protocol converter may be set up to accept other ASCII terminals such as an HP 2392A.

Because Reflection is able to emulate a number of terminals, it is possible, working with a protocol converter, to make a PC running Reflection appear to be an IBM 3270-style terminal. Reflection is then able to communicate with the mainframe.* The fields that you may have to change to establish communications are on the Datacomm Configuration menu:

- Baud rate
- Parity
- Receive and transmit pacing

Make changes according to your protocol converter, and save the changes to a configuration file.

* The Reflection Complement, 3270 FileExchange, goes a step further and allows file transfer between the PC and the host.

Remapping the PC Keyboard

There are a number of 3270 functions, represented by keys on the 3270 keyboard, that a PC does not have. For each function, the protocol converter requires a set keystroke sequence on the asynchronous terminal so it can translate that sequence into one the mainframe understands.

If you are emulating a VT100 terminal for instance, the IBM 3174 system control unit requires that the sequence Esc-H be entered in order for the mainframe to understand that the HOME function is intended. You can use Reflection's keyboard remapping feature, described in detail starting on page 221, to simplify sending this sequence. For example, to map the HOME function to the Home key on the PC, use these steps:

1. Put the following lines in a keyboard remapping file, which can be created with any text editor. The first line identifies what sort of keyboard you are using; the second the emulation you are using.

```
keyboard-id = enhanced
term = VT

home = "^[H"
```

2. Give the remapping file a name that indicates its contents, such as 3174.KBM.

3. Compile the remapping file using KEYMAP.EXE and incorporate it into your configuration file, which in this example is called IBM.CFG:

```
C:> KEYMAP 3174.KBM IBM.CFG
```

4. Run Reflection using the IBM.CFG configuration file:

```
C:> R1 IBM.CFG
C:> R7 IBM.CFG
```

5. Once you are connected to the mainframe via the 3174 protocol converter, pressing Home sends the HOME function to the host.

Note: Some protocol converters have been reconfigured such that a sequence other than the default is expected in order to emulate a 3270 function. Try to communicate using the defaults; if a problem arises, check with the systems operator to see if the protocol converter has been reconfigured.

7.2 Miscellaneous Host Configurations

The table below gives configuration values for the following:

- Dow Jones Service or THE SOURCE
- VM/370
- MVS/TSO

Note: **Receive pacing** and **Transmit pacing** on the Datacomm Configuration menu should be set to *NONE* for all of these hosts.

Table 6
Configurationfor Other Hosts

Host	Field	Value	Config Menu
Dow Jones or THE SOURCE	Parity	Even	Datacomm
	Local echo	NO	Terminal Page 1
VM/370	Parity	1's	Datacomm
	Local echo	YES	Terminal Page 1
	Transmit functions	YES	Terminal Page 1 (HP)
MVS/TSO	Parity	1's	Datacomm
	Local echo	YES	Terminal Page 1
	Transmit functions	YES	Terminal Page 1 (HP)

A value of *1's* in the **Parity** field is sometimes referred to as *mark* parity.

If you are emulating an HP terminal (such as 2392A), check the **Use host prompt** setting on the Terminal Configuration, Page 2 menu. When configuring for Dow Jones, THE SOURCE, or MVS/TSO, set **Use host prompt** to *NO*. For VM/370, **Use host prompt** should be set to *YES*:

DC1 is the host prompt character for VM/370. Set it by typing the following on the Reflection command line ([Alt]-[Y]):

```
SET HOST-PROMPT "^Q"
```

To restore the default host prompt (none) for VT emulation, type:

```
SET HOST-PROMPT "^@"
```

Section 3

Troubleshooting

Chapter 8

Troubleshooting Installation

8.1 Not Enough Memory to Run Reflection

If you get an error message about memory limitations, either your computer does not have enough memory or DOS has allocated some memory to another program. See page 327 for tips on conserving memory.

If you have at least 320K and are running on an IBM-compatible PC, try renaming the AUTOEXEC.BAT and CONFIG.SYS files and rebooting your PC using Ctrl-Alt-Del.

If Reflection runs after booting without these files, you may have to eliminate one or more memory-resident programs or device drivers.

8.2 Configuration Problems

It is sometimes helpful to restart Reflection with all default values. If you need to completely reset Reflection, use **/D** when loading (for example, **R1 /D**). Once reset, save these values to a configuration file and begin troubleshooting from this point.

Configuration Files with both HP and DEC Settings

The configuration file (named R1.CFG or R7.CFG by default) contains both HP and DEC configuration parameters. When switching between DEC and HP emulation, some potentially confusing situations may occur:

- If you activate particular HP mode configuration values and then switch to DEC emulation, the original HP configuration is no longer active when you return to the HP terminal class (and vice versa). You must save the desired configuration to disk prior to switching modes. Similarly, to save both HP and VT values to a single configuration file, you must press **SAVE TO DISK** from each mode.

- You may save a new HP configuration to another filename (such as ABC.CFG). However, only the HP configuration is saved; default values are used for the VT portion of the ABC.CFG configuration file. If you use the values for both the HP and the VT emulation, it's better to make a copy of ABC.CFG and then load this file to change selected values.
- You cannot change between DEC and HP modes once you place Reflection into background mode. When using the command SET TERMINAL-CLASS in a command file, the terminal class changes, but the execution of the command file ceases.

Background Mode

Any configuration item that allocates PC memory cannot be changed once Reflection has been placed into background mode (a **B** between the function key labels indicates that Reflection is in background mode):

- The default for **Amount on SHELL memory (K)** is *0*. If you increase SHELL memory, you must do so before Reflection has been placed in the background.
- The default for **Amount of display memory** on the Global Configuration menu is *20K*. If you need to bring large files into display memory, raise this value before placing Reflection into background mode.
- If using a print buffer, raise the **Buffer size** on the Printer Configuration menu before putting Reflection into the background.

Reflection Freezes When Loading

When Reflection starts up, it tries to determine what kind of keyboard your system has and what interrupt is used to trap keyboard interrupts. If there are software incompatibilities at this point, your PC may hang.

- Some clone PCs or may be incompatible with the use of Enhanced keyboard support. This may cause Reflection to hang when it is started up. Try running Reflection with the /K0 switch: type **R1 /K0** or **R7 /K0** and press [Return].

- If running Reflection with **/K0** does not solve the problem, you may try running Reflection with both **/K0** and **/K1**: type **R1 /K0 /K1** or **R7 /K0 /K1** and press [Return].
- You may also want to upgrade any other software that hooks Interrupt 16 to a version that supports the Enhanced keyboard.
- If you are using an IRMA card, IRMA may pop up during Reflection's initialization. If this occurs, use the IRMA hot-key to force it to the background so that Reflection can continue to load. **/K0** or an IRMA upgrade should solve this problem.
- You can create a short remapping file to specify the keyboard you are using. For example, an old IBM AT keyboard with a new AT or AT clone system may not have the [Enter] key may mapped correctly to the keypad [+] key. The BIOS for newer machines is Enhanced and Reflection assumes the keyboard is also Enhanced. The solution is to use [Shift]-[F10] or to identify your keyboard to Reflection so that it does not attempt to figure out which keyboard you are using according to the BIOS.

  If you choose to remap, a short file with a .KBM extension is sufficient. Identify your keyboard-id as AT and indicate which terminal you are emulating:

  ```
  keyboard-id = AT
  term = HP
  ```

  Then compile this program as described in *Keyboard Remapping* beginning on page 221.

Troubleshooting

Chapter 9

Problems with the Host Connection

9.1 No Initial Response

If nothing is echoed on the screen when you type characters while connected to a host computer, check for the following possible causes:

- Is Reflection in remote mode? Press Alt-M to display the modes labels. F4, the **REMOTE MODE** label, must contain an asterisk; otherwise, Reflection does not transmit anything to the host.
- Is the cable correct? If possible, verify the connection by connecting a terminal with the same cable you use to connect the PC. If the terminal works, then the cable is probably correct. If not, see page 42 for cable diagrams.
- Check the **Datacomm port** field in the Datacomm Configuration menu. You may be trying to use the wrong COM port. Even if you think you are using COM1, try COM2.
- Check the **Baud rate** field; it should match your host or modem baud rate. If the host computer is an HP 3000, use the default values for all other datacomm fields.
- If your PC *locks up* and does not respond to pressing F8, EXIT, this usually indicates that the PC hardware is incorrectly configured. Make sure that you do not have two serial cards both set up as a single COM port and check your hardware IRQ settings. See *COMCHECK* below.

- Are you using an external modem? Read about troubleshooting modems on page 54.
- Have you enabled the **Type ahead** field of the HP Terminal Configuration, Page 1 menu? If so, Reflection may be in a state of waiting for a host prompt character. Press [Alt]-[T] to release Reflection from the wait state.

9.2 Troubleshooting Serial Ports

If Reflection appears to be running but you are unable to get a response from your host computer, check your serial ports.

COMCHECK

A utility program called COMCHECK, provided with Reflection, lets you check your serial ports. It is a diagnostic tool that tells you how your COM ports are configured. An AT or compatible machine can have up to 4 COM ports; a PS/2 can have up to 8. The default IRQ (interrupt request) line each COM port uses is listed in a table on page 92. The command syntax is as follows:

```
COMCHECK [?][/1][/2][/3][/4][/5][/6][/7][/8][/Q1][/Q2][/Q3][/Q4]
```

Type **COMCHECK** without parameters to check all of your PC's serial ports. In some cases you may not want to check all ports simultaneously. If you are using a 3270 coaxial board such as *Attachmate*, check your serial ports one by one.

The switches described below check individual ports.

[?] Type **COMCHECK ?** to get a short description of each parameter.

[/1] Checks for COM 1 with IRQ4. This switch can be used on all hardware. To check COM1 alone, use /1 as follows:

```
COMCHECK /1
```

[/2] Checks for COM 2 with IRQ3. This switch can be used on all hardware.

[/3] Checks for COM 3 with IRQ10. This switch is for AT or compatible machines only.

[/4] Checks for COM 4 with IRQ11. This switch is for AT or compatible machines only.

[/5] through [/8]

Switches /5 through /8 are valid only for PS/2 machines with microchannel architecture.

COMCHECK lets you find out whether COM3 or COM4 has been set up to use either *IRQ2* or *IRQ5* via the Datacomm Configuration menu.

[/Q1] Checks for COM 3 with IRQ2.

[/Q2] Checks for COM 3 with IRQ5.

[/Q3] Checks for COM 4 with IRQ2.

[/Q4] Checks for COM 4 with IRQ5.

If you have more than one serial port in your PC and the COMCHECK program only recognizes one COM port (or none), find out if the boards are set up according to the documentation supplied with them. You may have to open your PC and look at how they are set up. There are usually two jumper wires on these boards that can be adjusted to make the serial port on the board function as COM1-COM4. (PS/2 serial boards are not configurable.) Problems occur if boards have conflicting configurations. Check the following:

The IRQ line

Check the setting on the boards for the jumper that sets the interrupt request line. See page 92 for a list of the default IRQ line each COM port uses and how to configure for a different line.

The I/O port

Check the setting on the boards for the jumper that sets the I/O port address.

See the documentation for the boards to determine how to configure them correctly.

COMCHECK does not support the datacomm selections set via the COMDRIVE utility, described on page 92.

Note: Simple communications programs that use *polling* may work even if the IRQ line is incorrectly configured. However, Reflection is interrupt driven and does not work unless the IRQ and I/O jumpers are correctly configured.

Hardware Testing

The paperclip test described in the Reflection *Quick Start Guide* is a diagnostic tool for checking on your PC's serial ports (COM ports). Run the COMCHECK program before using this test; a common problem is a serial port that is not generating interrupts, or having two serial ports trying to be the same COM port. If COMCHECK reports that your COM port is properly installed, and Reflection still can't communicate with your host, then it's time for the paperclip test.

Troubleshooting

9.3 Strange Characters Displayed

Unexpected characters on the screen may be caused by parity problems or framing errors. Reflection uses the ASCII delete character (Δ, decimal 127) as an on-screen indication of a datacomm error.

- Look at the Summary of Datacomm Errors screen (press [Alt]-[E]); this may help indicate the source of the errors. See page 71.
- Try changing **Parity** on the Basic Configuration or the Datacomm Configuration menu. Try *NONE, 0's*, or *EVEN* (in that order). If changing parity doesn't help, disconnect and reconnect; it's possible that you had a bad line.

9.4 Troubleshooting Clear Text File Transfer

Clear text file transfer is the method used to upload host file transfer programs. No error checking is performed with this approach, but the Summary of Datacomm Errors screen can indicate the possible sources of the problem. See *Analyzing Datacomm Errors* below.

The host prompt character is a possible source of failure with clear text file transfer. Sometimes Reflection sends the first line of display memory or a file to the host, and then stops and does nothing. The problem may be the host prompt character. If the **Use host prompt** field of the Terminal Configuration, Page 2 menu has a value of *YES*, Reflection waits for the specified character to be received after each line before it sends the next line. Type **VERIFY HOST*** on the command line to see what the HOST-PROMPT character is. Most HP 3000s use the $^{DC}1$ (^Q) character. If you are not connected to an HP 3000 and your host computer does not use this standard character, Reflection still waits for one. In this case, set **Use host prompt** to *NO*.

9.5 Analyzing Datacomm Errors

Press [Alt]-[E] to bring up a summary of datacomm errors.

SUMMARY OF DATACOMM ERRORS

DESCRIPTION OF ERROR	
Receiver overrun	0
Parity error	0
Framing error	0
Break detect	0
Receive buffer overflow	0

RESUME

Figure 13: Summary of Datacomm Errors

Are all the fields *0*? Any fields that are non-zero indicate some problem in data reception. Whenever the datacomm hardware (the UART) detects a data communications error, Reflection displays a DEL character (Δ) on the screen and increments the appropriate error counter in the Summary of Datacomm Errors screen. The column of counters indicates errors on the current datacomm port. The types of errors are explained below.

Receiver overrun

Each time Reflection receives a character from the host computer, the UART interrupts the processor, saying, in effect, *I just received a character*. Reflection must read that character before another character is received; if it doesn't, a *receiver overrun* error occurs. Check for the following:

- Software spoolers
- Clock card drivers
- Multitasking operating system
- The IBM 3270 control program (lowering the baud rate fixes this error)

If you have some of these programs resident (or are uncertain whether you do or not), try renaming your AUTOEXEC.BAT and CONFIG.SYS files and rebooting the PC. Then reload Reflection and try again. If you are successful, preface any suspicious lines in your AUTOEXEC.BAT files with REM, making them remarks so that they do not execute. (To change your CONFIG.SYS file, you must remove the lines entirely. Save a copy of your original CONFIG.SYS file under another name.) Reboot and try again. By a process of elimination, see what can and cannot be loaded into memory with Reflection.

Parity error

The parity error count is incremented on this screen only if **Check parity** on the Datacomm Configuration menu is set to *YES*. Parity errors are much more likely to occur with a modem connection than with a direct connection. Match the parity of the host or intermediary system. Normally, **Check parity** should be set to *NO*.

Framing error

Framing errors tend to occur when you change the baud rate (e.g., with a SPEED command on the HP 3000). A bad cable or one that is too long can also cause framing errors. A framing error (½ symbols on the screen) may also indicate that the UART received a character with the wrong number of stop bits. If you're consistently getting these errors, you may have specified the wrong number of stop bits in the configuration of the remote port. Most systems use one start bit and one stop bit, but two stop bits are often used at 110 baud.

Break detect

Whenever the UART detects a break (200 millisecond signal), it increments the number in the **Break detect** field by one. Your communication line is probably noisy and the signal is being interrupted.

Receive buffer overflow

Reflection has a buffer where characters received from the host computer are held until they can be processed. If Reflection can't keep up with the incoming data, the buffer fills up and an overflow occurs. If this happens, confirm that **Receive pacing** on the Datacomm Configuration menu is set to *XON/XOFF* (the default). If this setting is not the problem, find out what kind of flow control your system uses.

Chapter 10

Troubleshooting File Transfer

This chapter covers problems with two aspects of Reflection's file transfer:

- Problems uploading the host file transfer program (PCLINK2, VAXLINK2, UNXLINK2)
- File transfer problems with intermediary systems

A complete list of error messages starts on page 583.

10.1 Problems Uploading the Host Program

A built-in mechanism is provided for uploading PCLINK2 to an HP 3000 *, and VAXLINK2 to VAX/VMS (only the VMS operating system, version 5.0 or above, is supported by VAXLINK2). Select the appropriate emulation (HP or DEC) before uploading. From the File Transfer menu, press F7, **upload xfer pgm**. See *Uploading the Host Program* on page 347.

Note: You must have a direct-connect line or a direct modem line for the initial upload of PCLINK2. You cannot upload the PCLINK2 in an environment that requires a 7-bit protocol.

Troubleshooting the PCLINK2 Upload

The error messages that may occur during an upload of PCLINK2 follow, along with suggested solutions.

* PCLINK2 cannot be uploaded to an HP 1000.

Can't find PCLINK2.PUB

If Reflection cannot find the file PCLINK2.PUB in your Reflection directory, recopy PCLINK2.PUB from the original program diskette onto the default PC directory and try again.

Could not invoke FCOPY utility

Reflection uses the FCOPY utility to upload the PCLINK2 program by issuing the command RUN FCOPY.PUB.SYS. If for some reason the HP 3000 was not able to run FCOPY, an error results.

Make sure some other program is not running on your port by pressing Ctrl - Break, Esc and typing a colon (:). Then type **abort** at the colon prompt and press Return. If you still have problems, see your system manager.

Transmit error

Check your datacomm configuration; try setting the **CTS required** and **DSR required** fields to *NO*.

Check your connection by entering **VERIFY CONNECTION** on the command line. Make sure the setting is *DIRECT*, even if you are connected via modem.

Timeout (receive)

- Check the Datacomm Configuration menu: the **Parity** field should be set to *0's* or *NONE*.
- Check the Terminal Configuration, Page 2 menu: **Use host prompt** should be set to *YES*.
- Check the HOST-PROMPT character by typing **VERIFY HOST-PROMPT** on the command line. It should be *^Q*.
- If you're connected via modem over a phone line, you may have a poor connection. Try redialing or dial during off hours.
- Are you connected over a local area network (LAN)? A public network (X.25)? Or over a phone system that is a private branch exchange (PBX), such as Rolm or Dimension? Is there a multiplexer on the port of the host computer you are using? If so, one of these devices is probably interpreting the data you are sending. Find a way to bypass the device, e.g., by using a different port into the host computer or using an outside phone line.

Could not save program file

This is an HP 3000 upload error message. You do not have write access or save access to the group and account you specified. If you choose to save the file to different name or account, you must also change **Host startup sequence** on the File Transfer Configuration menu to match the name you assigned. Press [F6] on the File Transfer screen to display this menu.

The HP 3000 BUILD command can fail when Reflection tries to create the PCLINK2 program file while PCLINK2 is being uploaded to the HP 3000. Attempt the upload again after checking to see that you are at the MPE prompt; no other program should be running at your port.

Integrity test failed - start over

When Reflection uploads the host software to a supported host, it sends the file twice and then compares the two files. If they are not identical, this message is displayed. Try uploading again. If you're using a modem, try redialing to get a better connection.

Troubleshooting VAXLINK2 and UNXLINK2 Upload

VAX

You must have VAX/VMS 5.x or greater to upload and run VAXLINK2. If the version of the VAXLINK2 program as named in the **Host startup sequence** field of the File Transfer Configuration menu is not compatible with the version of Reflection you're running, then you receive the message **Incompatible host software.** Upload the current version of VAXLINK2.EXE from the support files diskette.

If you have problems using the automatic upload of the host program using [F7], **upload xfer pgm** on the File Transfer Configuration menu, you can upload VAXLINK2 via KERMIT using these steps: * * *

1. Make sure that you have VMS KERMIT-32 on your host. It must be version 3.3.011 or newer (the version number is displayed when you start the program).
2. Start up the host program—this is done at most sites by typing **KERMIT**.
3. At the KERMIT-32 prompt on the host, type the following:

```
SET FILE TYPE FIXED
```

4. Type **RECEIVE** at the KERMIT-32 prompt.
5. Press [Alt]-[Y] to get to the Reflection command line. Then type the following:

```
KSEND VAXLINK2.EXE BINARY
```

6. After the transfer, exit the KERMIT-32 program by typing **QUIT**.

UNIX

Before attempting a UNIX upload, make sure Reflection is configured correctly for a UNIX system; see page 351. UNXLINK2 is uploaded and then compiled via a command file on the command line (type **UPLOADUX.RCL**).

10.2 Intermediary Systems

File transfer problems often occur when you have intermediary communications hardware and/or software systems between the PC and a host computer. Typical examples of these intermediaries are public data networks (TYMNET/Telenet) and multiplexers.

These intermediaries can cause transfer problems because Reflection's default file transfer method sends binary characters during a file transfer (even with an ASCII transfer)—these characters may be interpreted by third-party systems.

If you have been having trouble with file transfers using WRQ protocol with an intermediary system between your PC and the host computer, the following changes may help:

- Change the setting of **File transfer link** on the File Transfer Configuration menu
- Lower **Packet size** on the File Transfer Configuration menu
- Raise **Character xmit delay** on the Terminal Configuration, Page 2 menu

Some of these changes slow down the speed of transfer. Once you are able to successfully transfer a file, experiment with the settings one at a time and determine which ones are relevant to your set-up.

Changing File Transfer Link

The steps below assume you are using WRQ protocol. See page 387 for *OLD-WRQ* equivalents.

Using 7-bit file transfer, each data character (including binary data) is modified to ensure that it is a non-control (printable) ASCII character. This method works with many configured line protocols and can be used with most intermediaries.

The 7-bit method does slow down transmission. With ASCII files, you can expect this technique to be 10-15% slower. With binary files, transmission may take as much as twice as long.

For the HP 3000, you may be able to change only the presentation and transfer at a higher rate. Press [Alt]-[Y] to bring up the command line and type the following:

```
SET TRANSFER-PRESENTATION "B"
```

From the system keys, press [F6] to get to the File Transfer screen, then press [F6] again for the File Transfer Configuration menu. **File transfer link** should now read *USER-DEF*. *

This setting handles some of the problems of intermediary systems without the slower speed associated with the 7-bit transfer method. If this method is successful, type **SAVE** on the command line to make this setting part of your configuration file. *

If you still experience data errors, change the setting of **File transfer link** to *7-BIT*. *

Lowering packet size

An additional adjustment for intermediary systems is to lower the *packet size*, the number of data bytes that are transmitted in each packet. On the File Transfer Configuration menu, change the **Packet size** field to a value lower than that currently displayed. Activate or save this setting and try the transfer again. Lowering packet size slows the speed of transfer.

Changing Character Transmit Delay

If you are transferring files to an HP 3000 at 9600 baud or higher, raising the **Character xmit delay** setting to between 5 and 8 on the Terminal Configuration, Page 2 menu may help eliminate data errors.

10.3 Configuring Network Servers

It is generally best to configure an asynchronous communications server to do flow control with the host. Reflection will do flow control with the network (and host), but best results are obtained by having the server do flow control as well.

Some servers allow transmit pacing and receive pacing to be configured independently.

- Flow control from an asynchronous server to the host is *transmit pacing*—the host sends an XOFF to the server when the host is not ready to receive data.
- Flow control from the host to an asynchronous server is *receive pacing*—the server sends an XOFF to the host when the server is not ready to receive data.

"Classic" HP 3000

If the host is a "classic" HP 3000, the asynchronous server should be configured to do ENQ/ACK flow control if the server supports ENQ/ACK (most do not). The server can be configured to do XON/XOFF receive pacing (the server sends an XOFF to the HP when its buffer is full), though this is usually not necessary: the HP will only send 80 characters at a time, then send an ENQ, and wait for an ACK.

If the asynchronous server does not do ENQ/ACK, then make these changes:

- Configure Reflection for ENQ/ACK pacing (the default). At the Datacomm Configuration screen, set **Enq/Ack pacing** to *YES*. The throughput will be less, because each ENQ from the host will have to be sent across the network to Reflection.
- Transmit pacing on the server should be none. It should not do XON/XOFF transmit pacing (the host sends an XOFF to the server when the host buffer is full), because the XON character is a DC1, which is the HP 3000 read trigger (host prompt). If the asynchronous server interprets the DC1 as an XON, it considers the character a flow control character, and does not pass it on to Reflection.
- If the server does not allow independent configuration of transmit and receive pacing, its pacing should be set to none.

MPE/XL HP 3000

MPE/XL hosts do not do ENQ/ACK pacing. The server must therefore be configured to do XON/XOFF receive pacing. Transmit pacing should be set to none.

If the server does not allow independent configuration of transmit and receive pacing, then the read trigger (host prompt) must be re-configured in both Reflection and on the host. The read trigger is a DC1 ($^{D}1$) by default. Changing it to DC4 ($^{D}4$) is recommended.

- To re-configure the host prompt in Reflection, type the following on the command line:

```
SET HOST-PROMPT "^T"
```

- To re-configure on the host, run the TRIGGER program on the HP 3000. This program is available via the WRQ bulletin board (see the front of this manual for the number).

VAX and UNIX

For a VAX or UNIX host, the server should be configured to do XON/XOFF in both directions (receive and transmit). Depending on the network configuration, it may be possible to configure the server to do no pacing and 8-bit file transfers.

10.4 Errors During File Transfers

Below are the most common error messages seen during file transfers. A more complete set of error messages starts on page 583.

No Host Response or Timeout

These messages are displayed at the start of a file transfer if Reflection does not receive the expected data character from the host system. This can occur if Reflection experiences consecutive timeout periods without receiving any data response from the host or Reflection can't get the normal $^{D}C_1$ prompt from the host. In either case, the transfer is terminated. There are a number of possible causes:

- There may be no connection with the host.
- You may not be logged on to the host.
- You may have another program running; the host file transfer program cannot be executed.

- The host program may not have been uploaded.
- The **Host startup sequence** is incorrect. See page 365.
- The serial port on the HP 3000 may not be configured as an HP block mode terminal. Log on to the system again, specifying **;TERM=10** at the end of your logon sequence (**HELLO...**).

Troubleshooting

Incompatible Host Software

The current release of Reflection is distributed with host file transfer programs that require the WRQ protocol. To use the WRQ protocol, you must upload the current version of PCLINK2.PUB, VAXLINK2.EXE, or UNXLINK2.C from the support files diskettes (see page 348). Previous Reflection file transfer programs can still be used by choosing the OLD-WRQ protocol. The host startup sequence varies based on the protocol selected. See page 387. *

File Transfer Link Failed

Under the default host startup sequence, an 8-bit data path (no parity) is used. First, try changing **File transfer link** as described on page 79. Next, ensure that the parity setting for Reflection matches the host parity setting. *

File transfer link failed can be a result of several conditions: *

- During a file transfer, packets of data are checked for errors as they are transferred. If an error occurs, the packet is *not acknowledged* or NAKed. More than 20 consecutive NAKs cause the file transfer to abort, and this message is displayed.
- During file transfer, Reflection received consecutive strings of unsolicited or indecipherable data. The last packet of data is then retransmitted. **Error retry limit** on the File Transfer Configuration menu specifies the number of consecutive occurrences of this condition before the message is displayed and the transfer is terminated.
- If the host receives data that it cannot read, this error message may be displayed. An error occurred during file transfer from which Reflection could not recover.
- During the file transfer, Reflection experiences consecutive timeout periods without receiving any data response from the host.

Could Not Create DOS File

This error can occur at the start of a transfer when receiving a file from the host to the PC. The file could not be created on the PC's disk. Is there space left on the disk or in the directory? Is the write-protect tab on?

Disk Space Exhausted

This message is displayed when a disk file fills up the disk during a file transfer from the host. The function key labels are changed, giving you the option of continuing the file transfer on a new disk or cancelling the file transfer. If this error occurs while transferring a file to a hard disk, cancel the transfer and erase the incomplete file from your hard disk. Make room on your disk or use another drive, then try again.

Aborted by Host (0)
END OF FILE (HP)

When a file being transferred already exists on the host, the file characteristics of the existing file are kept, even if you specify *D* (delete) in the **If file exists** field. You must purge the existing file if you want the file characteristics to be determined by the file being transferred. Use the purge option by ending the host filename with **;P**, e.g., **TEXT.DOC;P**.

Aborting File Transfers

If you want to abort a file transfer, press the **STOP** key and wait for Reflection to abort the transfer. The **ABORT** key is only there in case the transfer cannot be aborted by Reflection in a reasonable time. If you do press **ABORT**, the host file transfer program continues to run on the host. Exit the screen and press [Ctrl]-[Y] twice to stop the program.

10.5 Transmit Pacing

If Reflection were to transmit data as fast as it could, the host computer might not be able to keep pace. If that happens, use one of the following methods to control the speed of transmission:

- If the host uses XON/XOFF pacing to control transmission speed, set **Transmit pacing** on the Datacomm Configuration menu to *XON/XOFF*. **Transmit pacing** has a default value of *NONE* in HP mode, but in VT mode the default is *XON/XOFF*.
- Enter a value in the **Character xmit delay** field of the Terminal Configuration, Page 2 menu. This causes Reflection to wait the specified time between each character it transmits. A 1 to 3 millisecond delay should be sufficient for most systems.

  If you are experiencing trouble with file transfer, set the **Character** *
  xmit delay to a high number (for example, 10) and then test the *

appropriate level by lowering the number until the same problem reoccurs. Determine the lowest number you can use without incurring file transfer errors.

- If the host computer sends a prompt character when it's ready to receive a line from the terminal, enter that character with SET HOST-PROMPT "<char>" on Reflection's command line. To check the character, turn on display functions on the modes keys and press [Return] at the visible host prompt (:). The HP 3000 sends down either a $^{D}C_{1}$ (^Q) character, or no special prompt is used. It is rare for another character to be used.

10.6 Log Files

To troubleshoot file transfer, it is helpful to have a trace file that logs the interaction between the PC and the host.

To capture all data coming from the host or going to the host (regardless of the transfer protocol you are using), use the command **SET TRACE YES**. The *Command Language Manual* provides examples of this technique.

VAX/VMS File Transfer Log

When you are using VAXLINK (the host program that corresponds to the protocol *OLD-WRQ*) or VAXLINK2 (protocol *WRQ*), you can have a file transfer log file created on the host by adding a parameter to the host startup sequence. Make VAXLINK or VAXLINK2 a foreign command, then add the **L** parameter. File transfer activity is logged to a file called **WRQLOG.** (the period is required). The log file resides on the host in the current VAX directory .

UNIX File Transfer Log

When you are using UNXLINK2 (the host program that corresponds to the *WRQ* protocol), you can have a file transfer log file created on the host by adding a parameter to the host startup sequence. Add **-l** to the host startup sequence as follows:

```
unxlink2 -l
```

File transfer activity is logged to a file called WRQLOG, which resides on the host.

Chapter 11

Troubleshooting Operations

11.1 Printer Error Messages

If you do not have a printer, it is best to have *NONE* as the printer interface value. An error results if you attempt to direct output to a printer when the Printer Configuration menu has a value of *NONE* in the **Printer interface** field (see page 137) or if the printer is not available.

No Configured Printer

The **Printer interface** field on the Printer Configuration menu must match the type of printer port that you actually have installed in your system, either *SERIAL* or *PARALLEL*. If a driver is specified, it must be named in the **Driver name** field. If *FILE* is specified, a name must appear in the **File name** field.

Printer Timeout

If you direct output to the printer when the printer is offline, out of paper, or otherwise inoperable, a timeout occurs after several minutes.

Print Buffer is not Empty

Whenever you change the size of the **Printer buffer** on the Printer Configuration menu, the buffer is completely rebuilt and the old buffer is deleted. When this message is displayed, you must press either the *YES* or the *NO* function key. If you press *NO*, the buffer is not changed. If you press *YES*, the buffer is rebuilt, and its old contents are destroyed (see page 141).

11.2 Background Operations

The background option allows you to use your PC while Reflection runs in the background. Some common problems are discussed below.

Not Enough Free Memory

If your PC does not have enough memory to run Reflection and a particular application program, an error message results. Use a smaller memory option when starting Reflection by using the **/MEM:** switch. See *Memory Management* on page 327.

Need xxK of Memory for Hot-Key

Enough memory to use the hot-key must be available to put Reflection into the background and run another program. The minimum requirement is enough memory for COMMAND.COM and for storing the video image. On a monochrome monitor, 27K is usually required. You can also use state save as an alternative to the hot key; see page 329.

Section 4

Configuration

Chapter 12

Data Communications Configuration

The default values on the data communications menu are set to allow Reflection to communicate directly with an HP 3000 (in HP mode) or VAX/VMS (in VT mode) from COM1.

Auto-Dial Modems on page 50 contains helpful information about configuring for data communications with a modem. Some parameters may need to be reconfigured before you can communicate successfully.

Read *Host Connections* starting on page 39 and *Miscellaneous Host Configurations* on page 59 if you are working with hosts other than the HP 3000 or VAX/VMS. *Set-Up for IBM Mainframes and Other Hosts* starts on page 57.

12.1 Setting Datacomm Parameters

The following data communications parameters apply to both VT and HP emulation modes. Reflection command language equivalents are provided where applicable. Selecting the values for these parameters is similar to setting other Reflection configuration parameters; see page 37 for details.

To bring up the Datacomm Configuration menu:

1. Press [Alt]-[C] from anywhere in Reflection to display the configuration keys.

Configuration

Data Communications Configuration

2. From the config keys, press F1. The Datacomm Configuration menu appears.

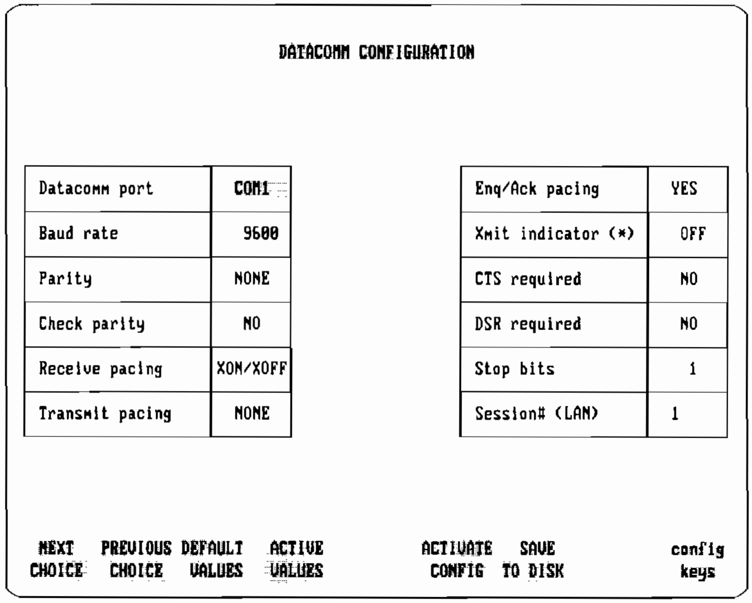


Figure 14: Datacomm Configuration Menu

Reflection allows you to connect the PC to the host computer via serial port or through one of the supported LAN connections. See page 55 for AdvanceNet (OfficeShare) and Office Extend information. See the *PLUS Manual* for information on other network configurations. Only one serial port at a time can be used for host communications. Other ports can be used to drive a serial printer or other serial device. COM1 to COM4 are only available on ATs; PS/2s can use up to COM8.

Datacomm Port

Select the option that matches your hardware connection. Use F1 to view all the choices.

Values: COM1 (the default) through COM8 *

COM3-IR2	Selects IRQ2 for COM3
COM3-IR5	Selects IRQ5 for COM3
COM4-IR2	Selects IRQ2 for COM4
COM4-IR5	Selects IRQ5 for COM4
ADV.NET	AdvanceNet
U.B.	Ungermann-Bass's Network
BRIDGE	Bridge-3COM's Network
HP-TELNT	HP-Telnet (OfficeShare)
INT-14	Generalized environment using Interrupt 14
MNP-COM1 through COM8	Microcom's Networking Protocol and other external drivers (see page 53)
LAT	Digital's Network
AT&T	AT&T Starlan
IBM-ACS	IBM's LAN Asynchronous Connection Server
RAF	Remote Access Facility from Datability
TEL-MGR	TelnetManager, a Reflection Complement
EICON	X.25 Server, Eicon Technology Corporation
NASI	Novell, Inc.
NET-VMS	Novell's NetWare for VMS
MOBIUS	Fel Computing, Inc.
CTERM	Digital's Command Terminal Access Interface
EXTEND	Fransen/King's Office Extend

Reflection command language equivalent:

```
SET DATACOMM-PORT <value>
```

On AT class machines, the Reflection defaults for COM3 and COM4 are IRQ10 and IRQ11 respectively. These defaults work with the HP24541B card; if you have this card, you should use the COM3 and COM4 selections.

If you have a COM3 or COM4 serial card on which you can select either IRQ2 or IRQ5, use the appropriate datacomm port option: COM3-IR2, COM3-IR5, COM4-IR2, or COM4-IR5. Make sure that no two devices are using the same IRQ line.

If you require an IRQ setting that is not included in the options listed above, *
COMDRIVE is a terminate-and-stay-resident utility that allows alternate *
assignment of IRQ interrupts for COM ports. *

---

* *COM5* through *COM8* are only displayed if you have a PS/2.

As indicated in the table below, COM2 through COM8 on the PS/2 share IRQ3, and these IRQs are not configurable in Reflection.

Use the COMCHECK utility to see what communications ports and IRQ lines are being used. It is described on page 68.

Table 7
IRQ Line Options

COM Port		IRQ
AT/compatible:		
	COM1	IRQ4 (the default)
	COM2	IRQ3 (the default)
	COM3	IRQ10 (the default)
		IRQ2
		IRQ5
	COM4	IRQ11 (the default)
		IRQ2
		IRQ5
	COMDRIVE options (use MNP1)	(see below)
PS/2:		
	COM1	IRQ4 (not configurable)
	COM2-8	IRQ3 (not configurable)

Configurable IRQs: COMDRIVE.COM

This terminate and stay resident program allows alternate assignment of IRQ interrupts for COM ports that are not supported as standard Reflection datacomm choices. For example, COMDRIVE allows you to assign COM3 with an IRQ of 3, 7, etc. COMDRIVE must be installed before Reflection is started; it uses less than 3000 bytes of memory.

To use COMDRIVE, change the **Datacomm port** value to *MNP-COM1*. This value allows access to the COM and IRQ settings assigned by COMDRIVE.

The syntax for COMDRIVE follows:

```
COMDRIVE/<n> [/<m>] [/u]
```

/<n> refers to a datacomm device name. For example, COMDRIVE/2 refers to COM2.

[/<m>] refers to the IRQ line. You must include this parameter with COM3 and COM4 on XT class machines.

/U uninstalls the driver.

For example, to use COM3 with an IRQ of 4, type **COMDRIVE /3/4**. You can add the COMDRIVE command to your AUTOEXEC.BAT file or a batch file that loads Reflection. COMDRIVE must be loaded before Reflection in order to operate correctly.

Baud Rate

Specifies the speed at which Reflection transmits and receives data through the serial port.*

Values:	110	1800	38400
	150	2400	57600
	300	4800	115200
	600	9600 (the default)	
	1200	19200	

Reflection command language equivalent:

```
SET BAUD <value>
```

12.2 Selecting the Correct Parity

Use the fields described below to change your parity settings and track parity problems.

Parity

Parity is used by communications equipment to detect some kinds of errors in data transmission. When *ODD* or *EVEN* parity is used, one of the bits in each character is a parity bit. The transmitter sets this bit to *on* (1) or *off* (0) so that the entire character code, including the parity bit, contains an odd (odd parity) or even (even parity) number of 1 bits.

* Baud rates above *19200* may not be appropriate for your PC or host hardware.

If **Check parity** (see below) is set to *YES*, the receiver tests each character for an odd or even number of 1 bits, and sets an error indicator whenever a character fails the test. Reflection always transmits, and expects to receive, a total of eight bits per character, not counting start and stop bits. The entire eight bits may be used to represent data, or one of the bits may be used as a parity bit, leaving seven bits for data.

Values: NONE (the default)

Each character consists of eight data bits with no parity bit. If the high-order bit is used in terminal mode, the character is interpreted as an extended Roman 8 character.

0's	Each character consists of seven data bits, with the eighth bit always 0. This is often called *space* parity.
ODD	Odd parity.
1's	Each character consists of seven data bits, with the eighth bit always 1. This is often called *mark* parity.
EVEN	Even parity.

Reflection command language equivalent:

```
SET PARITY <value>
```

Escape sequence equivalent:

See page 467

Check Parity

When **Check parity** is *YES*, Reflection checks parity on all received characters. When an error is found, the character in error is replaced by an ASCII DEL (Δ) character. When the field is *NO*, then parity on received characters is not checked. Parity is always generated for transmitted characters based on the **Parity** field. This setting is typically used for troubleshooting.

Values: NO (the default)
YES

Reflection command language equivalent:

```
SET CHECK-PARITY YES/NO
```

Escape sequence equivalent:

See page 465

12.3 Pacing Data Transfer

Use the fields below to adjust the pacing of data transfer.

Receive Pacing

It is possible for a host computer to transmit data to Reflection faster than Reflection can process it. Should this continue for too long, Reflection's *receive buffer* overflows and data is lost. If the host computer recognizes the XON/XOFF handshake, you can prevent this by enabling **Receive pacing.**

XON/XOFF receive pacing works as follows:

- When Reflection's receive buffer has a limited amount of space left, it sends an XOFF (ASCII $^{D}C_3$) character as a signal to stop transmitting.
- When Reflection has processed most of the backlog of characters in its receive buffer, it sends an XON (ASCII $^{D}C_1$) character to resume transmission from the host.

When **Receive pacing** is set to *XON/XOFF* (the default), the hold screen key is activated. ([ScrollLock] is used in VT mode and [Alt]- or [Shift]-[ScrollLock] is used in HP mode).

To determine if the host computer recognizes the XON/XOFF handshake, enter a [Ctrl]-[S] (the $^{D}C_3$ character) while Reflection is receiving data from the host system. If the host system uses XON/XOFF, it immediately stops transmitting, so that the display is frozen. When you enter [Ctrl]-[Q] (the $^{D}C_1$ character), the host system should start transmitting again.

See page 54 for information on hardware flow control.

Values: XON/XOFF (the default)
HARDWARE
NONE

Reflection command language equivalent:

```
SET RECEIVE-PACING <value>
```

Escape sequence equivalent:
See page 476

Transmit Pacing

It is possible for Reflection to send data to the host computer faster than the host can process it. With this field, you can tell Reflection to stop transmitting whenever it receives an XOFF (DC3) character from the host system, and to resume transmitting only after receiving an XON (DC1) character. This should not be used with most HP 3000s as the DC3 character is echoed, hanging the terminal.

If *NONE* is selected for **Transmit pacing**, problems may arise during file transfer for some systems. If Reflection were to transmit data as fast as it could, the host computer might not be able to keep pace. Set the **Character xmit delay** field on the Terminal Configuration, Page 2 menu to *3*. This slightly reduces the speed at which information is sent to the host while allowing you to receive data at a higher baud rate.

See page 54 for information on hardware flow control. See page 83 for troubleshooting information on transmit pacing.

Values:	NONE	(HP mode default)
	XON/XOFF	(VT mode default)
	HARDWARE	

Reflection command language equivalent:

```
SET TRANSMIT-PACING <value>
```

Escape sequence equivalent:

See page 476

Enq/Ack Pacing

The HP 3000 and HP 1000 host computers use a form of handshaking called *Enq/Ack* (ENQuire/ACKnowledge) to prevent the terminal (in this case, Reflection) from falling too far behind the host system and losing data.

With Enq/Ack pacing, the host system sends 80 characters followed by an ASCII ENQ character and stops transmitting. When Reflection has processed all of the characters preceding the ENQ, it sends an ASCII ACK character, telling the host it is ready for more data.

Values:	YES	(HP mode default)
	NO	(VT mode default)

Reflection command language equivalent:

```
SET ENQ-ACK YES/NO
```

12.4

Modem-Related Fields

The fields below are used only if you are connected to the host via a modem.

Xmit Indicator(*)

If you are using a modem to connect to the host, you may want to have Reflection display an indicator of the state of one of the RS-232 lines. Some modems can be configured to force these lines to a *true* state; in this case, the indicator has little meaning. The factory settings of a Hayes Smartmodem cause it to always set the Data Carrier Detect (DCD) signal to a *true* state.

This field allows you to specify which modem status line, if any, you wish to have monitored. If the selected line (*pin*) is in a *true* state, an asterisk is displayed between the fourth and fifth function key labels.

Values:	OFF (the default)	No indicator
	CTS	Clear To Send
	DSR	Data Set Ready
	DCD	Data Carrier Detect

CTS Required

When this field is set to *YES*, Reflection does not transmit unless the RS-232 Clear To Send signal (pin 5) is active *(true)*; otherwise, the program ignores the signal. Set this field to *NO* if *HARDWARE* is selected in the **Receive pacing** or **Transmit pacing** fields.

Values: NO (the default)
YES

Reflection command language equivalent:

```
SET  CTS-REQUIRED YES/NO
```

DSR Required

When this field is set to *YES*, Reflection does not transmit unless the RS-232 Data Set Ready signal (pin 6) is active *(true)*; otherwise, the program ignores the signal. Set this field to *NO* if *HARDWARE* is selected in the **Receive pacing** or **Transmit pacing** fields.

Values: NO (the default)
YES

Reflection command language equivalent:

```
SET DSR-REQUIRED YES/NO
```

Stop Bits

Stop bits are transmitted after each data character to signal the end of the character. Specify one stop bit for all baud rates other than 110 baud, and two stop bits for 110 baud (unless you are positive that your host or serial device requires otherwise).

Values: 1 (the default)
2

Reflection command language equivalent:

`SET STOPBITS <value>`

12.5

Session Number

If you are using a LAN that supports multiple sessions (e.g., Ungermann-Bass), Reflection allows you to identify which session you would like as the current one. Enter the session number (1-255) and activate the configuration to select a session. If you are not using a network, the value of this item has no effect.

Values: 1 - 255
Default: 1

Reflection command language equivalent:

`SET SESSION# <value>`

See the *PLUS Manual* for more information on LAN configurations.

Chapter 13

Plotter Configuration

Configuration

Plotter support is provided in Reflection 7 only. Use the Plotter Configuration menu to set the correct connection, parity, and baud rate for a plotter connected directly to the PC.

Plotters may be attached via eavesdrop cable or direct serial connection. Although an eavesdrop connection only requires one serial port, it forces you to physically switch cables if you are sometimes plotting with PC applications and sometimes with host applications. For either type of connection, pacing (e.g., XON/XOFF) is handled during plotter operations by the plotter itself, rather than by Reflection. You will probably have no need to set pacing; it is normally set by an escape sequence from the host.

13.1 Eavesdrop Mode

Eavesdrop mode requires a special eavesdrop cable. Both the host and the PC are connected to the plotter via this cable; the cable connections are clearly labeled.

The plotter *eavesdrops* on all transmissions from the host. Any escape sequences that are not for plotting are sent on to the PC. Any plotting sequences are executed.

Set the plotter before powering it on. The baud rate and eavesdrop mode must also be set correctly.

13.2 Pass Through Mode

Pass through mode prevents the need to reset the plotter and change cables when you are using PC graphics applications as well as host applications with a locally attached plotter. Pass through mode requires two serial ports on the PC. Configure the plotter serial port on the Plotter Configuration menu.

The baud rate and direct connection must be set on the plotter before powering it on. Use a null modem cable to connect the plotter to the PC. (Reflection does not currently support parallel plotters.)

Control codes from the host to the plotter are passed through the PC to the plotter. Control codes from the plotter to the host are passed from the PC to the host.

13.3 Configuring for a Plotter

To bring up the Plotter Configuration menu in Reflection 7, use [Alt]-[C] to display the configuration keys; then press [F2], **plotter config**.

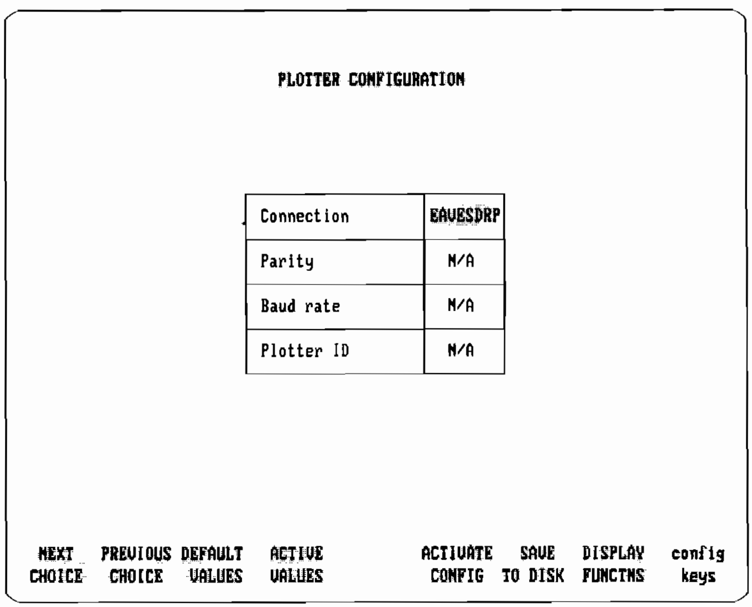


Figure 15: Plotter Configuration Menu

Connection

This field specifies the type of plotter interface, if any, through which the plotter is attached to the PC. *EAVESDRP* implies that the plotter is connected between the PC and the host via a special eavesdrop cable. In this case, parity and baud rate on the Datacomm Configuration menu still apply. If you do not have a plotter, the default setting (*EAVESDRP*) is correct.

Values:	EAVESDRP (the default)	COM5 *
	COM1	COM6
	COM2	COM7
	COM3	COM8
	COM4	NONE

If you are directly connecting your plotter to a separate serial port, you must indicate which serial port to use. On PC/XTs only COM1 and COM2 are available. PS/2s can connect plotters directly to COM1 through COM8. You must also configure the proper plotter baud rate and parity (described next).

If you are plotting to a disk file only, set **Connection** to *NONE*; plotter data only goes to the file specified by the LOGPLOT command. In this case, Reflection emulates the plotter's responses to the host based on the plotter type you select in the **Plotter ID** field.

Parity

The settings for **Parity** and **Baud rate** on this menu are used only if the plotter **Connection** (above) is other than *EAVESDRP* (the default). Otherwise, the fields read *N/A* and you cannot change them.

Check your plotter documentation for the correct parity to use with your plotter. See page 93 for a detailed description of parity.

Values:	NONE (the default)	Indicates all 8 data bits are sent to the plotter.
	0's	Only seven data bits are used, with the eighth bit always 0. This is often called *space* parity.
	ODD	Odd parity.
	1's	Only seven data bits are used, with the eighth bit always 1. This is often called *mark* parity.
	EVEN	Even parity.

* COM5 through COM8 are only available on PS/2 machines.

Plotter Configuration

Baud Rate

Specifies the speed at which Reflection transmits data to the plotter through a serial port. This setting applies only if **Connection** is set to a value other than *EAVESDRP*.

Values:

110	1800	38400
150	2400	57600
300	4800	115200
600	9600 (the default)	
1200	19200	

Plotter ID

This field only applies if you are plotting to a disk file and *NONE* is selected in the **Connection** field; otherwise, *N/A* is displayed. Reflection emulates the selected plotter when no plotter is available to communicate with the host program. Reflection provides partial emulation of the 7550A plotter and full emulation of the 7440A plotter. See page 406 for more information about plotting to a disk file.

Values: 7440A (the default)
7550A

Chapter 14

HP Terminal Configuration

This chapter examines each HP terminal configuration field in the order it appears on the Terminal Configuration, Page 1 and Page 2 menus. Reflection command language equivalents are provided where applicable.

14.1 Terminal, Page 1

To bring up the Terminal Configuration, Page 1 menu:

1. Press Alt-C from any Reflection screen to display the configuration keys.
2. From the config keys, press F3 for **terminal page 1**.

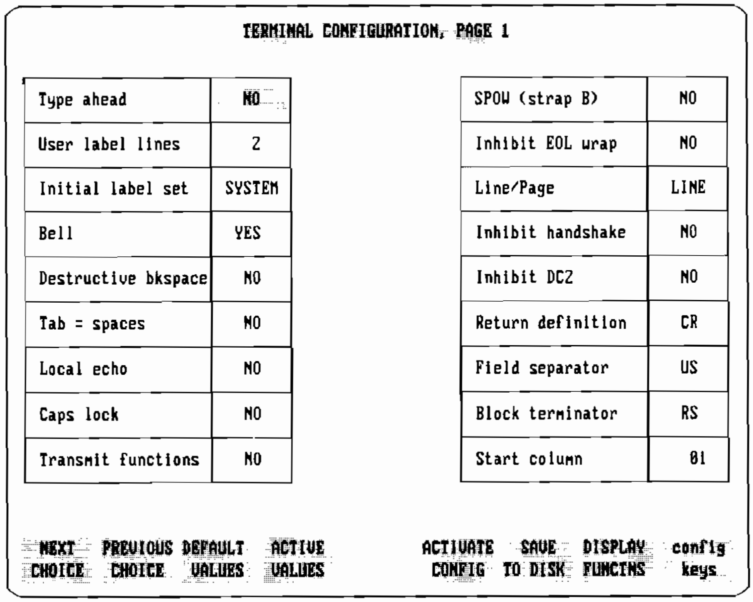


Figure 16: HP Terminal Configuration, Page 1

Type ahead

When using a terminal with an HP 3000, you must wait for a prompt from the host before entering new data. When the host has finished processing this input, it transmits a prompt, signaling that it's ready to receive new data. The host ignores characters that are received before it has issued this prompt.

The *typeahead* feature allows you to type continuously. Each time the host sends a prompt character (see the setting for **HOST-PROMPT** with the VERIFY command), Reflection sends the next line of keyboard input from its keyboard buffer. Typeahead is also supported in block mode. See page 317 for information on resetting the typeahead buffer.

Values: NO (the default)
YES

Reflection command language equivalent:

```
SET TYPE-AHEAD YES/NO
```

User Label Lines

Specifies the number of screen rows to be used for user-key labels. The default value is 2, which leaves only 23 lines of the PC's 25-line display available for text. If this field is set to a value of *1* or *0*, Reflection uses 24 lines for text whenever the user keys are displayed.

This field can only be set to *0* or *2* on Vectras and PCs with EGAs or VGAs, since they provide 24 lines of text, two label lines, and a status line. If you select the IBM font by using a /I switch, 24 text lines are reduced to 23 on EGAs and VGAs. To force a complete clone of the IBM PC on the Vectra, use /C, described on page 12. This ensures 25 lines as well as the IBM font. Use this if interference with other programs occurs due to the 27-line mode, e.g., Sidekick.

When the user-key definition menu is enabled by an escape sequence, the value in this field determines the subsequent size of the labels. If you then change the number of lines by pressing [F9], that change remains in effect until another enabling escape sequence is received or until you press [Alt]-[U].

Note: Only the user-key labels can be reduced in size; other labels always use two lines.

Values: 2 (the default)
1
0

Reflection command language equivalent:

```
SET LABEL-LINES <value>
```

Initial Label Set

Determines which set of screen labels is displayed when Reflection is loaded.

Values: SYSTEM (the default)
MODES
USERS

Bell

When this field contains a value of *YES*, Reflection emits a beep under the following conditions:

- When a character is typed eight columns left of the right margin

- In format mode, when the last character of an unprotected field is entered from the keyboard
- In file transfer mode, when file transmission is complete or terminated in error

When *NO* is specified, the bell sounds only when Reflection detects an input error, or when an ASCII BELL character (^G) is received from the host or entered from the keyboard. SET BELL-ENABLED affects all bells, regardless of the setting of this field.

Values: YES (the default)
NO

Escape sequence equivalent:
See pages 465 and 474

Destructive Bkspace

When this field contains a value of *YES*, Reflection erases the character to the left of the cursor each time an ASCII backspace character is entered. When the value is *NO*, [Backspace] moves the cursor to the left without erasing.

Values: NO (the default)
YES

Reflection command language equivalent:

```
SET DESTRUCTIVE-BACKSPACE YES/NO
```

Tab = Spaces

Under the default, pressing [Tab] transmits a tab character to the host. Many HP 3000 applications do not handle tab characters correctly.

When this field contains a value of *YES*, pressing [Tab] generates the number of space characters needed to move the cursor forward to the next tab stop. In character/remote mode, the spaces are transmitted to the host computer. If there are no tab stops to the right of the cursor position when [Tab] is pressed, the bell sounds (if enabled), the cursor does not move, and no spaces are generated.

A backtab functions in a similar manner. When [Shift]-[Tab] is pressed, Reflection generates the number of ASCII backspace characters needed to move the cursor to the previous tab stop, or to the left margin if there are no tab stops present.

Values: NO (the default)
YES

Escape sequence equivalent:
See page 477

Local Echo

When **Local echo** has a value of *YES*, each character typed at the keyboard is immediately displayed on the screen.

In remote mode, each character typed at the keyboard is transmitted to the host computer. Most host systems (e.g., the HP 3000 and VAX/VMS) *echo* the character. The character is not displayed on the screen until it is received back from the host. This is sometimes called *full duplex*.

When local echo and remote mode are both selected, each character is sent two places: directly to display memory (the screen) and to the host computer. If you are communicating with a host computer that echoes and **Local echo** has a value of *YES*, each character you type appears twice on the screen.

Set this value to *YES* only when communicating with host systems that do not echo each typed character (for example, some public networks).

When Reflection is in local mode or block mode, local echo is automatically enabled.

Values:	NO (the default)	full duplex
	YES	half duplex

Reflection command language equivalent:
```
SET LOCAL-ECHO YES/NO
```

Escape sequence equivalent:
See page 466

Caps Lock

When this field is set to *YES*, Reflection limits the set of characters that can be produced from the keyboard to Teletype-compatible codes.

- No lowercase alphabetic characters are generated. All alphabetic keys are automatically upshifted. (This can be reversed with the Shift key.)

- [,], and | are disabled.
- The ~ and ' symbols are disabled.

Values: YES
NO (the default)

Escape sequence equivalent:
See pages 465 and 474

Transmit Functions

Most of the keys on the keyboard have an associated ASCII character. Several keys, however, perform functions for which there is no character defined in the ASCII standard set, for example, Home, PgUp, Alt-J (clear display), and backtab. Certain host software programs, such as *HP Slate*, need to be informed when you press one of these non-ASCII keys. The **Transmit functions** feature provides a way for Reflection to inform the host whenever you press one of these keys.

When **Transmit functions** is enabled and Reflection is operating in character/remote mode, each time you press a non-ASCII key, the associated escape sequence is transmitted to the host. Most software that requires this feature sends the appropriate escape sequences; you probably will not need to enable it manually.

Values: NO (the default)
YES

Escape sequence equivalent:
See page 467

SPOW (Strap B)

Ordinarily, spaces entered from the keyboard overwrite and erase existing characters. When **SPOW (strap B)** is configured to *YES*, spaces entered from the keyboard can move the cursor over existing characters without replacing them. You can switch this capability on and off with a keyboard sequence. If SPOW (strap B) is set to *YES*:

- The SPOW switch is turned on by a carriage return
- The SPOW switch is turned off by sending a linefeed, tab, or home up (Ctrl-Home) function

When configured to *NO*, the spacebar functions normally.

Values: NO (the default)
YES

Escape sequence equivalent:
See pages 467 and 477

Inhibit EOL Wrap

When this field has a value of *NO*, Reflection automatically returns the cursor to the left margin in the next line when the cursor reaches the right margin or the right screen edge.

When **Inhibit EOL wrap** has a value of *YES*, the cursor is not automatically advanced when it reaches the right margin. As you type additional characters, each character overwrites the character at the right margin.

Values: NO (the default)
YES

Escape sequence equivalent:
See pages 466 and 475

Line/Page

When Reflection is operating in block mode, a block of one or more characters is transmitted when you press Enter or when the host system requests a block transfer. The setting of this field determines how much data Reflection transmits on each block transfer. This setting is automatically controlled by the host program; you should not have to change it.

Values:	LINE (the default)	Transmit a line at a time in block mode, or a field at a time in format mode.
	PAGE	Transmit an entire page or more.

Reflection command language equivalent:

```
SET LINE-PAGE LINE/PAGE
```

Inhibit Handshake, Inhibit DC2

The values in these two fields, along with some other factors, determine the type of handshaking that precedes each block transfer of data from Reflection to the host.

When set to *YES*, **Inhibit handshake** suppresses the $^{D}C_{1}$ handshake for block transfers. **Inhibit DC2** suppresses the $^{D}C_{2}$ portion of the handshake. Refer to *HP Block Transfers* starting on page 495 for a complete discussion of these fields.

Values: NO (the default for both fields)
YES

Reflection command language equivalent:

```
SET INHIBIT-HANDSHAKE YES/NO
SET INHIBIT-DC2 YES/NO
```

Escape sequence equivalent:

Inhibit handshake	See pages 466 and 475
Inhibit DC2	See pages 466 and 475

Return Definition

This field allows you to specify a string of one or two characters to be generated whenever [Return] is pressed. If the second character is a space, only the first character is generated.

Note: If you want to include a carriage return, tab, or backtab character in the definition, you must press the **DISPLAY FUNCTNS** key, [F7], before pressing any of those keys. Any other ASCII character can be entered without pressing **DISPLAY FUNCTNS**. Control characters (ASCII characters with values in the range 0-31) may be entered by holding down [Ctrl] while pressing the appropriate character key. For example, to define [Return] as a combination of carriage return and a linefeed, press **DISPLAY FUNCTNS** then type [Ctrl]-[M] and [Ctrl]-[J].

Values: Any two ASCII characters
Default: ^{C}R (ASCII carriage return character followed by a space)

Escape sequence equivalent:
See pages 458 and 476

Field Separator

When Reflection is transmitting in block, page, and format modes, it sends a field separator character after each field of the formatted screen except the last one. This field allows you to specify which ASCII character is used as the field separator.

Note: If you want to include a carriage return, tab, or backtab character in the definition, see the note under *Return Definition* above.

Values: Any ASCII character
Default: US (unit separator, decimal 31)

Reflection command language equivalent:

```
SET FIELD-SEPARATOR <value>
```

Escape sequence equivalent:
See page 475

Block Terminator

As explained in *HP Block Transfers* on page 495, under certain conditions Reflection transmits a block terminator character at the end of each block data transmission. This field allows you to specify which ASCII character is used as the block terminator.

Note: If you want to include a carriage return, tab, or backtab character in the definition, see the note under *Return definition.*

Values: Any ASCII character
Default: RS (record separator, decimal 30)

Reflection command language equivalent:

```
SET BLOCK-TERMINATOR <value>
```

Escape sequence equivalent:
See page 474

Start Column

For every line in display memory, Reflection keeps track of the leftmost column that was entered from the keyboard, as opposed to that received from the host. In this way, Reflection can distinguish the host prompt portion of each line from the user-entered portion. This information is then used when you enable **LINE MODIFY** or **MODIFY ALL** to determine the leftmost column that should be transmitted to the host when you press [Return] or [Enter]. Under some circumstances, it is impossible for Reflection to tell which column was the first user-keyed column; when this happens, Reflection uses the value of **Start column** to determine the leftmost column to be transmitted.

Values: 1-80
Default: 1

14.2 Terminal, Page 2

To access the Terminal Configuration, Page 2 menu, bring up the **config keys** with [Alt]-[C]. Then press [F4], **terminal page 2**.

TERMINAL CONFIGURATION, PAGE 2

Left margin	1
Right margin	80
Cols / horizontal scroll	1
Use host prompt	YES
Character xmit delay	0
Line xmit delay	0
Autorepeat	YES

Color scrolling	MAXIMUM
Terminal ID response	2392A
Esc Xfer	NO
Display memory response	15K
Forms buffer size (x256)	0
Cursor style	LINE
Inverse video	NO

NEXT CHOICE	PREVIOUS CHOICE	DEFAULT VALUES	ACTIVE VALUES	ACTIVATE CONFIG	SAVE TO DISK	DISPLAY FUNCTNS	config keys

Figure 17: HP Terminal Configuration, Page 2

Left Margin

Set the left margin by entering a number in this field. The value must be less than the right margin. Note that the left margin can also be set using the **margins/tabs/col** set of function keys ([F2] from the system keys).

Values: 1-80
Default: 1

Right Margin

The right margin can be set to any value from 2-10000, but it must be greater than the value of the left margin. When a value greater than 80 is entered, horizontal scrolling becomes possible, allowing for entry and viewing of lines longer than 80 characters without wraparound.

Note that the right margin can be set to a value of 80 or less using the **margins/tabs/col** set of function keys ([F2] from the system keys).

WARNING: If you configure a right margin greater than column 80, make sure FORCE-80-COLUMNS is set to *YES*. Format mode fields may otherwise be displayed incorrectly.

Values: 2-10000
Default: 80

Cols/Horizontal Scroll

This field determines the number of columns, or character positions, to be rolled right or left during one scrolling operation (with [Ctrl]-[←] or [Ctrl]-[→] keystrokes). This has no effect if the right margin is set to 80 or less.

Values: 1-80 (columns)
Default: 1

Use Host Prompt

Some host computers, notably the HP 3000, send DC1 as a prompt character to the terminal when they are ready to accept a line or block of characters. This field allows you to determine whether Reflection uses a special host prompt or not. When the **Typeahead** feature is enabled, Reflection waits for this character to be received from the host before it transmits the next line from the keyboard buffer.

If your host prompt is a character other than DC1, use the SET HOST-PROMPT "<char>" command to change the character. Most hosts either use a DC1 (^Q) character (the default) or no prompt (^@) character.

If the host system does not prompt in this manner, set this field to *NO*. In this case, you may also need to enter a non-zero value in the **Line xmit delay** field.

Values: YES (the default)
NO

Character Xmit Delay

When transmitting blocks of characters at high speeds, it may be necessary to modify the effective speed of the transmission to allow the host computer to keep pace. This frequently occurs when transmitting data blocks at 9600 baud to the HP 3000 Series 30 and 33 systems. It even occurs occasionally at 9600 baud on a heavily loaded Series 44.

This field specifies the number of milliseconds of wait time that Reflection should insert after each character when transmitting blocks of characters to the host system.

When the baud rate is 9600, a value of 3 milliseconds works best with the HP 3000 Series 30 and Series 33 systems. This lowers the effective speed of data transmission to about 2400 bits per second.

Values: 0-255
Default: 0

Reflection command language equivalent:

```
SET CHARACTER-DELAY <value>
```

Line Xmit Delay

If the host computer does not send a prompt character when it is ready for the next line of input (see **Use host prompt** on page 113), you may need to specify a non-zero value in this field. This field specifies the amount of time (in 1/10th seconds) that Reflection should wait after transmitting a carriage return character (the line delimiter) before it begins transmitting the next line.

Values: 0-255
Default: 0

Reflection command language equivalent:

```
SET LINE-DELAY <value>
```

Autorepeat

Setting this field to *YES* causes the keyboard to automatically repeat any key that is held down. When this field is set to *NO*, keys do not automatically repeat. Reflection does not modify the speed of key repeat or the initial delay set by the keyboard processor.

Values: YES (the default)
NO

The [Return] and [Shift] keys do not autorepeat.

Color Scrolling

The IBM color/graphics adapter card is not capable of scrolling fast enough to keep up with a high speed communications link (faster than 2400 bits/second) without losing display quality. If a program on the PC writes to the color/graphics display memory without waiting for a *retrace* interval, interference (*snow*) is produced on the monitor.

The display routines in the ROM BIOS used by DOS avoid display interference during scrolling through the following process:

- Turn the display off
- Move the data in display memory
- Turn display back on

While this method produces the fastest possible scrolling without snow, it also causes the entire display to blink noticeably on each line scrolled.

This field lets you choose the method of vertical scrolling that you want Reflection to use. The cleanest and slowest method is *SPEED 2*. The fastest method is the method used by DOS, called *BLINK*. In between are several speed selections, *SPEED 3* through *SPEED 6*. The higher numbers provide faster scrolling, but with more display interference. If you are communicating at 2400 baud or less, *SPEED 3* should provide adequate performance.

Some color/graphics display adapter cards claim flicker-free operation. If you have one, set this field to *MAXIMUM*. This tells Reflection to ignore retrace intervals and to write to display memory as fast as it can.

Note: This field must be set to *MAXIMUM* when Reflection is run on an IBM 3270/PC or some Zenith machines.

Values:	SPEED 2	SPEED 6
	SPEED 3	BLINK
	SPEED 4	MAXIMUM (the default)
	SPEED 5	

Terminal ID Response

This field specifies Reflection's response to a *terminal ID status request* received from the host computer. With Reflection 1, you should normally use the default (*2392A*).

Reflection 7 attempts to determine the terminal ID on the basis of hardware. You can force a particular setting by using a /G# switch when the program is loaded, although you cannot choose a terminal that is not supported with your hardware; see page 13. The graphics terminals listed here are only available in Reflection 7.

Values:

Reflection 1 *	Reflection 7	
2392A (the default)	2392A	
2622A	2622A	
2624B	2624B	
2626A	2626A	
2394A	2394A	
70092	70092	
70094	70094	
	2623A	monochrome graphics
	2393A	monochrome graphics
	2627A (the default)	color graphics†
	2390A	color graphics†
	2397A	color graphics†
	150A	monochrome graphics

Reflection command language equivalent:

```
SET TERMINAL-TYPE <value>
```

* These values also apply to Reflection 7 with the /G0 switch.

† Only available for supported graphics hardware.

Note: Reflection does not completely emulate an HP 700/94, 2624B, 2393A, or 2397A terminal. Specifying any of these terminals in this field may cause some host programs to send escape sequences that Reflection cannot interpret, making the system operate incorrectly.

Terminal models 700/92 and 700/94 are HP's replacement terminals for the 2392A and 2394A terminals respectively. Reflection supports all of the HP 700 Series terminal features listed below. Downloadable character sets (DCS) will be added in a later release.

- Cursor style can be an underline or block
- Cursor style can be turned off and on
- Display can be inverse video or normal
- Key repeat can be turned on and off
- Message line operation parameters can be configured
- Insert and delete characters with wraparound are available
- Additional information is available via terminal status responses
- Return and Tab can be modified
- Configuration selections can be saved and restored
- 132-column support is configurable*
- ANSI emulation has been upgraded to near-VT220 level

Full VT320 and VT241 emulation is available in Reflection 2 and 4 respectively.

The HP 2624B and 700/94 terminals have either 16K or 32K of display memory, as compared to the HP 2622A's 4K. Some host software may use the **Terminal ID response** to determine the amount of display memory it can use. You may want to specify *2624B* or *70094* to cause some host programs to take advantage of Reflection's display memory. Also, set this field to *2624B* or *70094* if you want to use the forms cache feature with V/PLUS. The 2626A terminal provides for lines up to 160 characters in length, using horizontal scrolling. You may want to select *2626A* in order to use horizontal scrolling with some host software.

* Reflection's 132-column support presupposes a PC with one of several supported 132-column display adapters.

Esc Xfer

This parameter defines whether display memory escape sequences are sent to the printer. If **Esc Xfer** is set to *YES*, the Reflection printer configuration file (R1PRINT.CFG or R7PRINT.CFG) determines the appropriate control codes. The printer configuration file contains the escape sequences and control codes used to print Reflection's display enhancements, set compressed and expanded printing, and change character sets. This file must be available whenever you change the **Control codes** field on the Printer Configuration menu.

Default values have been assigned in this file for each printer Reflection supports. If the values are not correct for your printer, you can change this file. Examples and further explanations are in the R#PRINT.CFG file. It can be edited with most text editors; be sure to save it in ASCII format.

Display enhancements do not print unless **Esc Xfer** is configured to *YES*. Print screen operations are also affected by this setting.

Values: NO (the default)
YES

Reflection command language equivalent:

```
SET ESC-XFER YES/NO
```

Escape sequence equivalent:
See pages 465 and 475

Display Memory Response

This field specifies the amount of display memory to be reported to the host computer as part of a primary status response. Some host software may be able to use Reflection's extended display memory if you set this field to a value greater than 4K.

Values:	4K	12K
	8K	15K (the default)

Forms Buffer Size (x 256)

This field specifies the amount of memory to be allocated to the forms cache buffer. Enter the desired number of 256-byte blocks. The value must be in the range 0 through 255. If you change this value, the contents of display memory and the printer buffer are cleared. If there is not enough memory for the specified forms buffer, the size of the buffer is not changed.

Values: 0-255 blocks
Default: 0

Escape sequence equivalent:
See page 475

Cursor Style

The cursor may appear as a blinking underline, a blinking box, or simply not be visible.

Values: LINE (the default)
BLOCK
NONE

Escape sequence equivalent:
See page 478

Inverse Video

To inverse your display, set this field to *YES*.

Values: NO (the default)
YES

Escape sequence equivalent:
See page 478

14.3 Reflection 7 Graphics Terminal Configuration, Page 2

Most of the fields in this screen are shared with the HP Terminal Configuration, Page 2 (non-graphics) menu, although they are displayed in a different order. The two fields found only for the graphics terminal configuration are documented here.

HP Terminal Configuration

TERMINAL CONFIGURATION, PAGE 2

Left margin	1	Color scrolling	MAXIMUM
Right margin	80	Terminal ID response	2627A
Cols / horizontal scroll	1	Initial graphics display	OFF
Use host prompt	YES	Tek 4010 compatibility	OFF
Character xmit delay	0	Esc Xfer	NO
Line xmit delay	0	Display memory response	15K
Cursor style	LINE	Forms buffer size (x256)	0
Inverse video	NO	Autorepeat	YES

NEXT CHOICE | PREVIOUS CHOICE | DEFAULT VALUES | ACTIVE VALUES | ACTIVATE CONFIG | SAVE TO DISK | DISPLAY FUNCTNS | config keys

Figure 18: HP Graphics Terminal Configuration, Page 2

Initial Graphics Display

The graphics display can be configured to be on or off when Reflection is loaded. A hard reset from the keyboard reinitializes Reflection according to this setting. A hard reset from the host *always* sets the graphics display *ON*.

Values: OFF (the default)
ON

Tek 4010 Compatibility

This field allows you to set compatibility mode for Tektronix terminal emulation. Scaled mode divides the incoming X/Y coordinates by 2 so that the output of programs written for the 1024 x 780 Tektronix terminal can be displayed in full on the PC's smaller screen. There is some resolution loss. In unscaled mode the PC displays one-fourth of the image. Changing the origin changes the part of the image displayed.

Values: OFF (the default)
UNSCALED
SCALED

Chapter 15

VT Terminal Configuration

This chapter examines each VT terminal configuration field on the Terminal Configuration, Page 1 and Page 2 menus. Reflection command language equivalents are provided where applicable.

15.1 Terminal, Page 1

To bring up the VT Terminal Configuration, Page 1 menu:

1. Press Alt-C from any Reflection screen to get to configuration keys.
2. From the config keys, press F3 to bring up the Terminal Configuration, Page 1 menu.

VT TERMINAL CONFIGURATION, PAGE 1

Multi page	YES
Initial label set	SYSTEM
User labels lines	2
Cursor style	LINE
Screen background	DARK
End of line wrap	NO
Caps lock	NO
Margin bell	YES
Auto repeat	YES
Esc Xfer	NO

UK character set	NO
Break key enabled	YES
Screen print extent	PAGE
Local echo	NO
Backspace key	DEL
Keypad mode	NORMAL
Cursor key mode	NORMAL
User features	UNLOCKED
Terminal type	VT102
DA response	VT102

NEXT CHOICE	PREVIOUS CHOICE	DEFAULT VALUES	ACTIVE VALUES	ACTIVATE CONFIG	SAVE TO DISK	DISPLAY FUNCTNS	config keys

Figure 19: VT Terminal Configuration, Page 1

Multi Page

Reflection allows your PC to emulate a multipage terminal that dynamically stores data. Application programs designed for DEC terminals, however, require that the terminal have only a single page of memory (24 lines). If your application program allows only single-page mode, set this field to *NO*.

When single-page mode is used, the following key functions are disabled: **Next page, Prev page, Home up, Home down, Roll up**, and **Roll down**.

Values: YES (the default)
NO

Control function equivalent:
See page 555

Initial Label Set

This field determines which set of screen labels are displayed when Reflection is loaded.

Values: SYSTEM (the default)
PF KEYS
MODES
USERS

User Labels Lines

Specifies the number of screen rows to be used for user-key labels. The default value is 2, which leaves only 23 lines of the PC's 25-line display available for text. If this field is set to a value of *1* or *0*, Reflection uses 24 lines for text whenever the user keys are called up. Only the user-key labels can be reduced in size; all other labels always use two lines.

This field can only be set to *2* or *0* on EGAs and VGAs. If you select the IBM font by using a /I switch, 24 text lines are reduced to 23 on EGAs and VGAs. To force a complete clone of the IBM PC on the Vectra, use /C, described on page 12. This ensures 25 lines as well as the IBM font.

Values: 2 (the default)
0
1

Reflection command language equivalent:

```
SET LABEL-LINES <value>
```

Some software does not work properly with two label lines (e.g., *ALL-IN-1*).

Cursor Style

The cursor may appear as a blinking underline, a blinking box, or simply not be visible. If the cursor is selected as a blinking underline or box, pressing $\boxed{\text{Ins}}$ or receipt of a control function *CSI 4h* toggles the cursor style between line and block mode. *CSI ?25h* and *CSI ?25l* (cursor visible/invisible) also affect this setting.

Values: LINE (the default)
NONE
BLOCK

Screen Background

Specifies a light or dark background for the screen. *LIGHT* overrides the Color Configuration menu values.

Values: DARK (the default)
LIGHT

Control function equivalent:
See page 554

End of Line Wrap

When this field has a value of *YES*, Reflection automatically returns the cursor to the left margin in the next (lower) line when the cursor reaches the right margin or the right screen edge.

When **End of line wrap** has a value of *NO*, the cursor is not automatically advanced when it reaches the right margin. As you type additional characters, each character overwrites the character at the right margin.

Values: NO (the default)
YES

Control function equivalent:
See page 561

Caps Lock

When this field contains a value of *YES*, Reflection limits the set of characters that can be produced from the keyboard to Teletype-compatible codes.

- No lowercase alphabetic characters are generated. All alphabetic keys are automatically upshifted.
- $\boxed{\{}$, $\boxed{\}}$, and $\boxed{|}$ are disabled.
- The ~ and ‘ symbols are disabled.

Values: NO (the default)
YES

Margin Bell

When this field contains a value of *YES*, Reflection emits a beep under the following conditions:

- When a character is typed eight columns left of the right margin
- In file transfer mode, when file transmission is complete or terminated in error

When *NO* is specified, the bell sounds only when Reflection detects an input error, or when an ASCII BELL character (^G) is received from the host or entered from the keyboard.

Values: YES (the default)
NO

Reflection command language equivalent:

```
SET BELL-ENABLED YES/NO
```


Auto Repeat

Setting this field to *YES* causes the keyboard to automatically repeat any key that is held down. When this field is set to *NO*, keys do not automatically repeat. Reflection does not modify the repeat speed and initial delay in use by the keyboard processor.

The [Return] and [Enter] keys do not autorepeat.

Values: YES (the default)
NO

Control function equivalent:
See page 561

Esc Xfer

This parameter defines whether display memory escape sequences are sent to the printer. If **Esc Xfer** is set to *YES*, the Reflection printer configuration file (R1PRINT.CFG or R7PRINT.CFG) determines the appropriate control codes. The printer configuration file contains the escape sequences and control codes used to print Reflection's display enhancements, set compressed and expanded printing, and change character sets. It must be available whenever you change the **Control codes** field on the Printer Configuration menu.

Default values have been assigned in this file for each printer Reflection supports. If the values are not correct for your printer, you can change this file. Examples and further explanations are in the R#PRINT.CFG file. It can be edited with most text editors; be sure the file you save is in normal ASCII format.

Display enhancements do not print unless **Esc Xfer** has been configured to *YES*. Print screen operations are also affected by this setting.

Values: NO (the default)
YES

Reflection command language equivalent:

```
SET ESC-XFER YES/NO
```


UK Character Set

Specifies whether the default character set (G0) for the terminal is the normal ASCII or UKASCII. The only difference between the two character sets is the ASCII character #, which is replaced by the British pound symbol £.

Values: NO (the default)
YES

Break Key Enabled

Enables and disables the action of Break.

Values: YES (the default)
NO

Screen Print Extent

When a screen print operation is performed with Alt-PrtSc, the **Screen print extent** setting determines how much of the screen is printed. A value of *PAGE* causes Reflection to print the entire screen. A value of *REGION* causes Reflection to print only the scrolling region, as defined by the top and bottom margins.

Values: PAGE (the default)
REGION

Control function equivalent:
See page 563

Local Echo

When **Local echo** has a value of *YES*, each character typed at the keyboard is immediately displayed on the screen.

In remote mode, each character typed at the keyboard is transmitted to the host computer. Most host systems (e.g., the HP 3000 and VAX/VMS) *echo* the character. The character is not displayed on the screen until it is received back from the host. This is sometimes called *full duplex*.

When local echo and remote mode are both selected, each character is sent two places: directly to display memory (the screen) and to the host computer. If you are communicating with a host computer that echoes and **Local echo** has a value of *YES*, each character you type appears twice on the screen.

Set this value to *YES* only when communicating with host systems that do not echo each typed character (for example, some public networks).

When Reflection is in local mode, local echo is automatically enabled.

Values:	NO (the default)	full duplex
	YES	half duplex

Reflection command language equivalent:

```
SET LOCAL-ECHO YES/NO
```

Control function equivalent:

See page 554

Backspace Key

Selects the meaning of Backspace. On the standard PC keyboard, Backspace generates a backspace character (ASCII 8), and Ctrl-Backspace generates a delete character (ASCII 127). By specifying *DEL* in this field, the assignment can be reversed so that Backspace generates a delete and Ctrl-Backspace generates a backspace character.

The backspace character causes the cursor to back up one position on the current line, but does not erase the character at that position. It is used by some host systems to mean *back up and delete the previous character*. On many hosts, when a delete character is encountered, the previous character in the input buffer is deleted and a three-character sequence of backspace/space/backspace is echoed back to the terminal. This effectively backs up the cursor and erases the previous character from the screen.

Values: DEL (the default)
BKSP

Reflection command language equivalent:
SET DESTRUCTIVE-BACKSPACE YES/NO
(YES=DEL, NO=BKSP)

Keypad Mode

Selects the codes to be generated by the keys on the numeric keypad. Either normal numeric values or special application escape sequences are generated. The numeric keypad is generally set by the host. Changing it locally may cause problems. If the number or PF keys aren't working properly, the problem may be that this mode was left set to *APPLCTN* when a host program terminated— changing this back to *NORMAL* should fix the problem.

See *Keyboard Codes* on page 578.

Values: NORMAL (the default)
APPLCTN

Control function equivalent:
See page 562

Cursor Key Mode

Selects the codes to be generated by the cursor keys. Either the normal cursor control escape sequences or special application escape sequences are generated. This option is usually controlled by an application program running on the host computer. Changing it locally may cause problems. See *Keyboard Codes* on page 578.

Values: NORMAL (the default)
APPLCTN

Control function equivalent:
See page 562

User Features

Locks the following items so that they cannot be changed by the host computer:

- Screen background
- Auto repeat
- Tab stops
- Keyboard lock
- Scrolling speed

Values: UNLOCKED (the default)
LOCKED

Terminal Type

Specifies which terminal should be emulated: VT52, VT102, or VT220 with 7- or 8-bit controls. The VT52 and VT102 are full terminal emulations; some features of the VT220 are not emulated. These choices affect the codes generated by the numeric keypad and the function keys, the interpretation of escape sequences, and the response to terminal identification requests.

Values: VT52
VT102 (the default)
VT220-7
VT220-8

Reflection command language equivalent:

```
SET TERMINAL-TYPE <value>
```

DA Response

The value in this field determines how Reflection responds to a primary device attribute request.

Table 8
Device Attributes Request Responses

Value	Reply	Meaning
VT102 (the default)	^{E}c[?6c	VT102
VT100	^{E}c[?1;0c	VT100
VT100av	^{E}c[?1;2c	VT100 advanced video
VT100p	^{E}c[?1;11c	VT100 w/printer
VT220	^{E}c[?62;1;2;6;7;8c*	VT220 normal
VT220x	^{E}c[?62;1;2;6c	Reflection

15.2 Terminal, Page 2

VT TERMINAL CONFIGURATION, PAGE 2

Left margin	1
Right margin	80
Columns per horizontal scroll	10
Character xmit delay (x .001 seconds)	0
Line xmit delay (x .1 seconds)	0
Color scrolling method/speed	MAXIMUM
Conceal answerback message	NO

Answerback message	

NEXT CHOICE	PREVIOUS CHOICE	DEFAULT VALUES	ACTIVE VALUES	ACTIVATE CONFIG	SAVE TO DISK	DISPLAY FUNCTNS	config keys

Figure 20: VT Terminal Configuration, Page 2

* The VT220 response means: VT200 family terminal (62) with 132 columns (1), printer port (2), selective erase (6), downloadable character sets (7), and user-defined keys (8).

Left Margin

Set the left margin by entering a number in this field. The value must not be greater than the right margin.

Values: 1-80
Default: 1

Right Margin

The right margin can be set to any value from 2 through 10000, but it must be greater than the value of the left margin. When a value greater than 80 is entered, horizontal scrolling becomes possible, allowing for entry and viewing of lines longer than 80 characters without wraparound.

WARNING: If you configure a right margin greater than column 80, Reflection does not work properly with host computer software that is programmed to wrap at column 80.

Values: 2-10000
Default: 80

Columns per Horizontal Scroll

This field determines the number of columns, or character positions, to be rolled right or left during one horizontal scrolling operation with [Ctrl]-[←] or [Ctrl]-[→] keystrokes. The parameter has no effect if the right margin is set to 80 or less.

Values: 1-80
Default: 10

Character Xmit Delay (x .001 Seconds)

When transmitting blocks of characters at high speeds, it may be necessary to modify the effective speed of the transmission to allow the host computer to keep pace. This field specifies the number of milliseconds of wait time that Reflection should insert after each character when transmitting blocks of characters to the host system.

Values: 0-255
Default: 0

Reflection command language equivalent:

```
SET CHARACTER-DELAY <value>
```

Line Xmit Delay (x .1 Seconds)

This field specifies the amount of time (in tenths of a second) that Reflection should wait after transmitting a carriage return character (the line delimiter) before it begins transmitting the next line.

Values: 0-255
Default: 0

Reflection command language equivalent:

```
SET LINE-DELAY <value>
```

Color Scrolling Method/Speed

The IBM color graphics adapter card is not capable of scrolling fast enough to keep up with a high speed communications link (faster than 2400 bits/second) without losing display quality. If a program on the PC writes to the color graphics display memory without waiting for a *retrace* interval, interference (*snow*) is produced on the monitor.

The display routines in the ROM BIOS, which are used by DOS, avoid display interference during scrolling by turning the display off, moving the data in display memory, and then turning the display back on. While this method produces the fastest possible scrolling without snow, it also causes the entire display to blink noticeably on each line scrolled.

This field lets you to choose the method of vertical scrolling that you want Reflection to use. The cleanest and slowest method is *SPEED 2*. The fastest method is the method used by DOS, called *BLINK*. In between are several speed selections, *SPEED 3* through *SPEED 6*. The higher numbers provide faster scrolling, but with more display interference. If you are communicating at 2400 baud or less, the default value, *SPEED 3*, should provide adequate performance.

Some color/graphics display adapter cards claim flicker-free operation. If you have one, make sure this field is set to *MAXIMUM*. This tells Reflection to ignore retrace intervals and to write to display memory as fast as it can.

Note: This field must be set to *MAXIMUM* when Reflection is run on an IBM 3270/PC or some Zenith machines.

Values: SPEED 2 SPEED 3 SPEED 4 SPEED 5 SPEED 6 BLINK MAXIMUM (the default)

Conceal Answerback Message

When configured to *YES*, the current answerback message is concealed. Once the string has been concealed, there is no way to retrieve it— you have to replace the text.

Values: NO (the default)
YES

Answerback Message

This 30-character message is transmitted to the host when the answerback key (Ctrl-F2) is pressed or upon the receipt of an ENQ character (decimal 5) from the host.

Values: Any message up to 30 characters *
Default: NULL

Configuration

Chapter 16

Printer Configuration

Complete the Printer Configuration menu, even if you don't have a printer attached to the PC. Setting the **Printer interface** field to *NONE* indicates no printer is available and streamlines disk operations.

This chapter describes each printer configuration field and provides Reflection command language equivalents where applicable.

To get to the Printer Configuration menu:

- Press [Alt]-[C] from any Reflection screen to display the configuration keys
- From the config keys, press [F5], **printer config.**

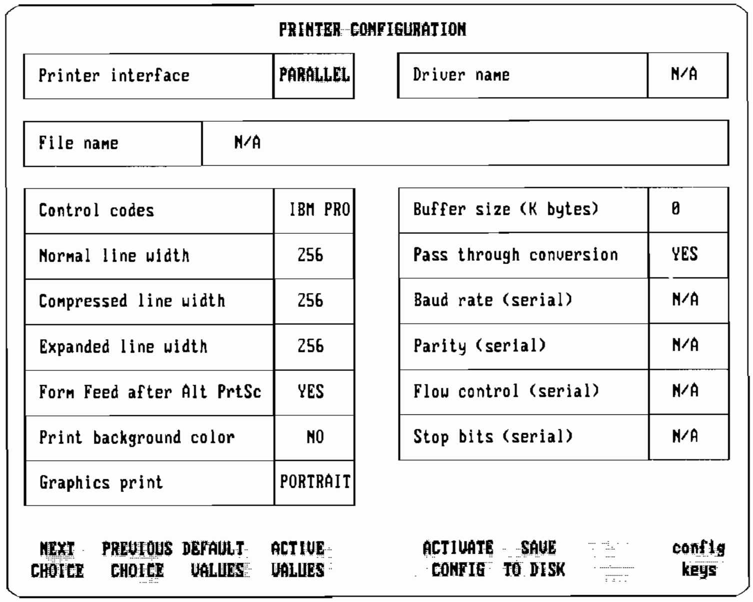


Figure 21: Printer Configuration Menu (Reflection 7, HP)

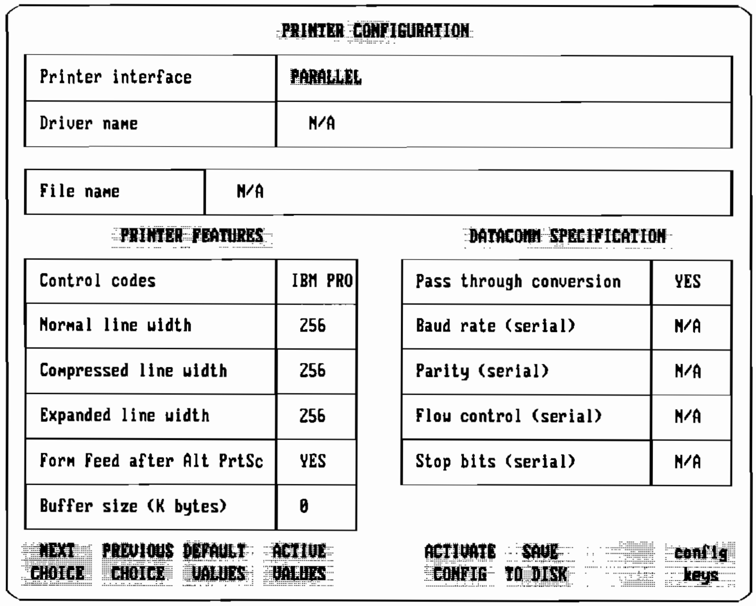


Figure 22: Printer Configuration Menu (Reflection 1 and VT)

16.1 Printer Options

The Reflection 1 and VT emulations do not have some of the fields that appear on the Reflection 7 menu. Even if you don't have a printer attached to the PC, you can use this menu to print to a disk file.

Printer Interface

This field specifies the type of printer interface, if any, through which the printer is attached to the PC. *PARALLEL* implies that the printer uses the parallel printer adapter on the IBM PC. Reflection allows you to define parallel ports 1, 2, 3, or 4 as the printer interface. If you specify *SERIAL*, you must configure the proper baud rate, flow control, and parity. Reflection allows you to define serial ports 1 and 2 on PCs and XTs, 1 to 4 on ATs, and 1 to 8 on PS/2s as the printer interface.

Values:			
Values:	PARALLEL (the default)	SERIAL-2	SERIAL-7
	PARALL-2	SERIAL-3	SERIAL-8
	PARALL-3	SERIAL-4	DRIVER
	PARALL-4	SERIAL-5	FILE
	SERIAL-1	SERIAL-6	NONE

Reflection command language equivalent:

```
SET PRINTER-INTERFACE <value>
```

Driver Name

If you specify *DRIVER* in the **Printer interface** field, you must enter the proper driver name here. You can default to whatever mode settings you already have defined by typing in a DOS device name: PRN, LPT1, LPT2, or LPT3.

Value: PRN (the default)

File Name

If you specify *FILE* in the **Printer interface** field, you must enter a filename here. See page 289 for more information on printing to a disk file.

Reflection command language equivalent:

```
SET PRINTER-FILE <filespec>
```

Control Codes

This field specifies the control codes, if any, that the printer uses to enable and disable expanded and compressed print, to print display enhancements when **Esc Xfer** is set to *YES*, and to do graphics dumps. If the printer does not have these features, select the value *OTHER*.

Values:	IBM PRO (the default)	TOSHIBA
	OKIDATA	OTHER
	ANSI	PAINTJET
	HP	IBM/EPSN

Reflection command language equivalent:

```
SET PRINTER-CONTROL <value>
```

Reflection attempts to translate characters to the character set of your attached printer based on the selection you make in this field. *

For example, selecting *IBM PRO* helps indicate how to control the printer; it *
also indicates that the printer's font matches the Code Page active in the PC. *

Table 9
Printer Control Codes

Printer	Expanded	Compressed	Normal
Proprinter	DC2 EC W SOH	EC W NUL SI	EC W NUL DC2
Okidata	US	CAN	GS RS
Ansi (Tally)	EC [0w	EC [6w	EC [4w
HP	EC &k1S	EC &k2S	EC &k0S
Toshiba	EC !	EC [	EC] EC "
PaintJet	N/A	EC &k2S	EC &k0S
IBM / Epson	EC ! SP	EC ! EOT	EC ! NUL
Other	(no codes)	(no codes)	(no codes)

The ASCII Code equivalents for these display control characters are as follows:

Display Control Character	ASCII Code
CAN	24
DC2	18
EOT	4
EC	27
GS	29
NUL	0
RS	30
SI	15
SOH	1
SP	32
US	31

Printing Line Drawing Characters

Reflection prints line drawing characters in two situations: *

- When **Control codes** is set to *IBM PRO*, Reflection sends the IBM 8 *
  line drawing characters. IBM Proprinters, some Epsons, and several *
  other printers support this when in IBM emulation mode. Settings for *
  Pass through conversion and **Esc Xfer** have no effect. *

- When **Control codes** is set to *OTHER* and **Esc Xfer** (Terminal Configuration, Page 2) is set to *YES*, line drawing characters are printed. This is supported by HP printers (including some LaserJets) that can switch to line drawing when a [Ctrl]-[N] is received. * * * *

In all other cases, Reflection sends a period (.) in place of each line drawing character.

Normal, Compressed, and Expanded Line Width

These fields specify a maximum line width for each of the print sizes. Reflection limits the maximum number of characters it prints on a line to these values.

Values: 40-9999

Defaults:	256	Normal
	256	Compressed
	256	Expanded

Reflection command language equivalent:

```
SET LINE-WIDTH-NORMAL <value>
SET LINE-WIDTH-COMPRESSED <value>
SET LINE-WIDTH-EXPANDED <value>
```


Form Feed after Alt PrtSc

When configured to *YES*, Reflection automatically advances paper to the top of the form after a print-screen operation ([Alt]-[PrtSc]) or graphics dump ([Alt]-[PgDn]).

Values: YES (the default)
NO

Print Background Color

This field applies to Reflection 7 HP emulation only.

The setting in this field determines if you print the background color when sending a graphics image to a PaintJet. It is only available when **Control codes** is set to *PAINTJET*. When **Print background color** is set to *YES*, all the colors on the screen are printed. When set to *NO*, all but the background color is printed; the background is left white and the white areas are printed as black.

Values: NO (the default)
YES

Graphics Print

This field applies to Reflection 7 HP emulation only.

This field controls the orientation of a graphics dump on a page. *LNDSCAPE* *
produces a larger, more readable printout on some printers. *

Values: PORTRAIT (the default)
LNDSCAPE

Reflection command language equivalent:

```
SET GRAPHICS-PRINT <value>
```

Buffer size (K Bytes)

You can set aside a portion of memory to be used for printer buffering if your printer interface is other than *DRIVER*. Having a printer buffer frees the keyboard for use during printing.

The first time you specify a printer buffer, the amount of memory requested is taken from whatever is free in display memory. An error message informs you if less memory is allocated than you requested. If you save this value to a configuration file, you will not get an error message the next time you load Reflection.

Values: 0-254 K bytes
Default: 0

Pass Through Conversion

Reflection normally doesn't translate characters sent from the host to a PC printer. Pass through printing can be performed using record mode, data transfer escape sequences (^{E}c&pW series) or VT printer controller mode. Data is sent to the printer without first appearing on the display. **Pass through conversion** controls whether or not Reflection should attempt to translate data when a pass through method of printing is being used (record mode or ^{E}c&p data transfer sequences). If your host software understands PC printers, **Pass through conversion** should be set to *NO*.

When **Pass through conversion** is configured to *YES*, **Control codes** determines the following:

HP, PaintJet	No effect. Data is passed straight to the printer.
all others	Eight bit characters are converted to IBM 8.

Note: If the **Parity** field on the Basic Configuration Menu is set to a value other than *NONE* and **Pass through conversion** is set to *YES*, data is in Roman 8 when directed to an HP printer. The data is IBM 8 if directed to an IBM printer. *
*
*
*

IBM 8 is either Code Page 437 or 850, depending on the current code page loaded. See pages 180 and 186.

Values: YES (the default)
NO

Reflection command language equivalent:

```
SET PRINTER-PASSTHRU-CONV YES/NO
```

Configuration

Baud Rate (Serial)

Specifies the speed at which Reflection transmits data to the printer through a serial port. (Applies only if a *SERIAL* option is selected for **Printer interface.**)

Values:

110	1800	38400
150	2400	57600
300	4800	115200
600	9600 (the default)	
1200	19200	

Reflection command language equivalent:

```
SET PRINTER-BAUD <value>
```

Parity (Serial)

Check your printer documentation for the correct parity to use with your printer. See page 93 for more information on this field. (Applies only if a *SERIAL* option is selected for **Printer interface.**)

Values: NONE (the default) — Each character consists of eight data bits with no parity bit. If the high-order bit is used and the printer has the Roman 8 character set available, the

		character is interpreted as an extended Roman 8 character.
	0's	Each character consists of seven data bits, with the eighth bit always 0. This is often called *space* parity.
	ODD	Odd parity.
	1's	Each character consists of seven data bits, with the eighth bit always 1. This is often called *mark* parity.
	EVEN	Even parity.

Reflection command language equivalent:

```
SET PRINTER-PARITY <value>
```

Flow control (Serial) It specifies how the flow of data from Reflection to the serial printer is to be controlled. Refer to the printer documentation to determine the correct value. (Applies only if a *SERIAL* option is selected for **Printer interface.**)

Values:	NONE	The serial printer has no way to stop and start the flow of data from the computer.
	XON/XOFF (the default)	When the printer's buffer is nearly full, it sends an ASCII $^{DC}3$ (XOFF) character to the computer. When the printer is ready for more data, it sends an ASCII $^{DC}1$ (XON) to the computer.
	ETX/ACK	After sending each 250-character block of data to the printer, the computer sends an ASCII ETX character. The printer sends an ASCII ACK character to the computer when it is ready for another block.
	DSR	The computer sends characters to the printer as long as the RS-232 Data Set Ready signal (pin 6) is active, *true*. The printer sets the line *false* when it is not ready for more data.
	CTS	The computer sends characters to the printer as long as the RS-232 Clear To Send signal (pin 5) is active, *true*. The printer sets the line *false* when it is not ready for more data.

Reflection command language equivalent:

```
SET PRINTER-FLOW-CONTROL <value>
```

Stop bits (Serial)

Every character is followed by either one or two stop bits. Specify one stop bit for all baud rates other than 110 baud, and two stop bits for 110 baud, unless you are positive that your serial device requires otherwise. (Applies only if a *SERIAL* option is selected for **Printer interface.**)

Values: 1 (the default)
2

Reflection command language equivalent:

```
SET PRINTER-STOPBITS <value>
```

Chapter 17

Color Configuration

The steps to change the color configuration apply to both HP and VT emulations.

There are two methods for configuring color with Reflection 7:

- Reflection 7 by default uses *color pairs* configuration, similar to an HP color terminal. See *Configuring Color Pairs* on page 150.
- If you want a different way to configure color, bring up the Reflection command line (Alt-Y) and type **SET IGNORE-COLOR-PAIRS YES**. You can then use the configuration instructions below.

Color configuration and other display enhancements can be set using an escape sequence. See page 470 for display enhancement escape sequences or page 490 for color pair settings (Reflection 7 only).

17.1 The Color Configuration Menu

To bring up the Color Configuration menu:

1. Press Alt-C to display the configuration keys.
2. From the config keys, press F6. The Color Configuration screen appears.

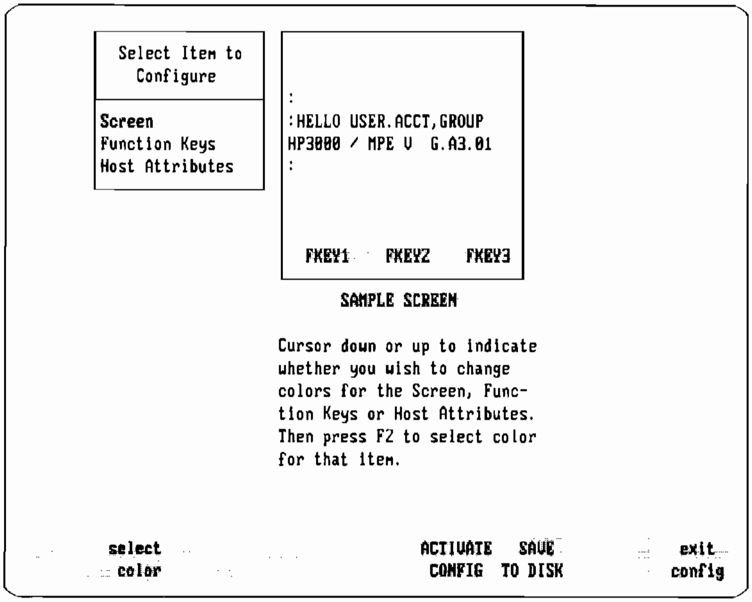


Figure 23: Color Configuration Screen

There are three items you can configure:

Screen

Foreground (text) and background (screen) colors

Function Keys

Foreground and background function key label colors

Host Attributes

How data sent from the host is displayed

After selecting an item to configure, [F7] reads **ENABLE BLINK**. You can *
choose between having *blink* as a character attribute, or disable it and double *
the number of available colors. Press [F7] to remove the asterisk and disable *
blink.* Keep in mind that this setting affects all enhancements: you cannot *
enable blink for host attributes and then disable it for the function key labels. *

* If you load Reflection with /A, [F7] is blank.

Changing Screen Color

To change the screen colors, confirm that *Screen* is highlighted and press [F2] to see the color choices.

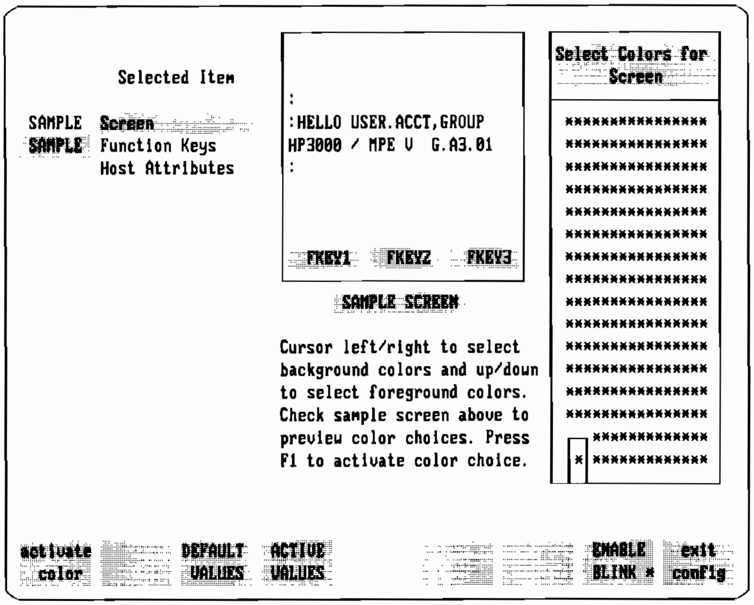


Figure 24: Selecting Colors

Use [←] and [→] to select the background color, and [↑] and [↓] for the foreground color. As you press any of the arrow keys, the colors on the *SAMPLE SCREEN* and *SAMPLE* change. Press [F1], **activate color**, to activate your selection. When the Color Configuration menu returns, activate ([F5]) or save ([F6]) the color configuration.

Changing Function Key Color

To change the colors of the function key labels, make sure that *Function Keys* is highlighted and press [F2] to see the color choices. Use the same steps as above to select a background and foreground color.

Host Attributes

No single PC adapter is capable of displaying all 16 terminal enhancement combinations. Instead, Reflection lets you choose a color combination that represents a given combination of display enhancements.

From the Color Configuration menu, press ⇩ until *Host Attributes* is highlighted. The display changes to show a list of 14 host attributes. To see the default for each attribute, use ⇩ or ⇧ to select the attribute and watch the sample screen. If you would like to change the default, press F2 to see the color choices.

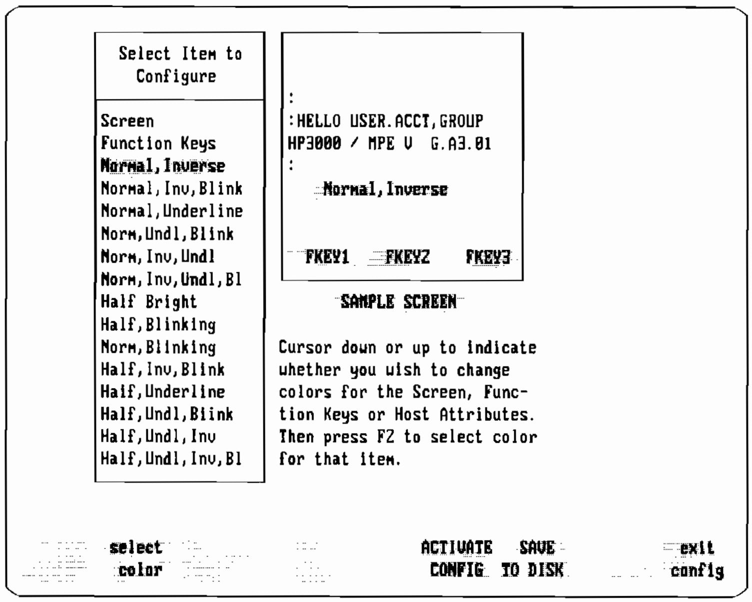


Figure 25: Selecting Colors for Host Attributes

17.2 Adapters

Monochrome Adapters

A monochrome adapter supports all four of the basic enhancements, but not all of the combinations. For example, it cannot combine inverse video and underlining. Also, bold-intensity background and the blinking enhancement for the foreground (i.e., characters) cannot be selected at the same time.

Graphics Adapters

Underlining is available if Reflection 7 is loaded with the /U switch for an EGA or VGA monitor. If you are in graphics mode, inverse and underline are supported, but blinking is not.

Below is a summary of adapters and switches that support specific display enhancements.

Reflection 1

Color adapter, color display (CGA, MCGA, EGA, VGA)

- Choice of blink or background intensity *
- Inverse supported *
- Underline not supported *
- Unsupported enhancements represented with different colors *

Monochrome

- Choice of blink or background intensity *
- Inverse supported *
- Underline supported except when combined with inverse *

Reflection 7

Graphics Mode

EGA/VGA	Blink unsupported	*
	Half intensity represented as color pair 3	*
	(same as terminal)	*
all others	Blink and half intensity unsupported	*

Alpha Mode

EGA/VGA	Underline unsupported	*
/U switch	Everything, including underline, supported	*
IGNORE COLOR-PAIRS	Identical to Reflection 1	*
all others	Identical to Reflection 1	*

17.3

Configuring Color Pairs

When Reflection 7 is run as a color graphics terminal, color is configured by selecting color pairs. Each pair consists of a foreground and a background color.

To configure colors in the same way as it is done in Reflection 1, type **SET IGNORE-COLOR-PAIRS YES** at the command line.

Note: Turning the background color to anything other than black causes the graphics display to be completely blocked if the alpha display memory is on. Graphics are not visible when the alpha display is on and other than black. To create a background color for a graphics screen, select a background pen color with a graphics escape sequence.

To bring up the color pairs configuration screen press [Alt]-[C] for the config keys, then press [F6].

ALPHA COLOR PAIR DEFAULT CONFIGURATION

COLOR PAIR #	FOREGROUND COLOR	BACKGROUND COLOR	SPECIAL USES
0	WHITE	BLACK	Normal text
1	RED	BLACK	
2	GREEN	BLACK	
3	YELLOW	BLACK	Half intensity
4	BLUE	BLACK	
5	MAGENTA	BLACK	
6	CYAN	BLACK	
7	BLACK	YELLOW	Function key labels

NEXT CHOICE | PREVIOUS CHOICE | DEFAULT VALUES | ACTIVE VALUES | ACTIVATE CONFIG | SAVE TO DISK | config keys

Figure 26: Alpha Color Pair Default Configuration

Only eight color pairs may be displayed at any one time. The default screen pair is 0, a black background with white letters. The default function key values are yellow labels with black letters. The color pair options follow:

Table 10
Color Pair Defaults

Color Pair	Foreground	Background
0	White	Black
1	Red	Black
2	Green	Black
3	Yellow	Black
4	Blue	Black
5	Magenta	Black
6	Cyan	Black
7	Black	Yellow

Chapter 18

Global Configuration

Generally, Reflection allows you to have separate configuration settings for HP and VT terminal types. For instance, your HP **Baud rate** can be set to *9600* while your VT **Baud rate** can be set at *19200*; both values are saved in the same configuration file. The fields on the Global Configuration menu are exceptions; these values are shared by both emulations.

18.1 Global Configuration Parameters

To bring up the Global Configuration menu:

1. Press [Alt]-[C] from any Reflection screen to display the configuration keys.
2. From the config keys, press [F7] **global config**.

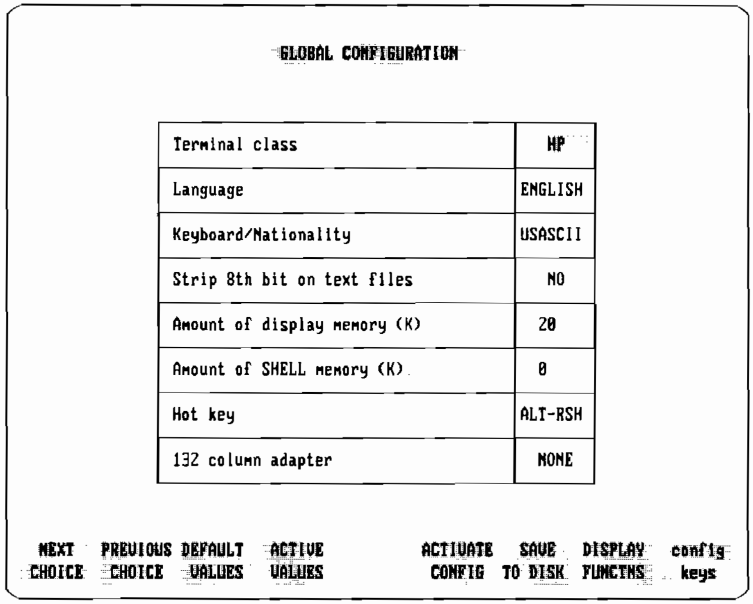


Figure 27: Global Configuration Menu

Terminal Class

Terminal class lets you select which class of terminals you wish to emulate, the Hewlett-Packard series or the Digital Equipment Corporation VT 200 series. The two emulations can be configured separately; a complete set of parameters can be stored in a single configuration file for each emulation.

The **Terminal class** field lets you toggle between the two sets of configuration values. For example, if your PC has two serial ports, you can configure the HP emulation to use one value in the **Datacomm port** field on the Datacomm Configuration menu, and select a different value for VT emulation. Then you can simultaneously log on to an HP 3000 and a VAX. Switch between host computers through the Global Configuration menu.

You cannot change this field once you have put Reflection into background mode. As an alternative, you can load two separate copies of Reflection, one in background and the other as a PC application. Then use the hot-key to toggle between the two.

Switching terminal class is restricted for Oracle applications. See page 312.

Values: HP (the default)
DEC

Reflection command language equivalent:

```
SET TERMINAL-CLASS <value>
```

Escape sequence equivalent:
See page 459

If SET TERMINAL-CLASS is used in a command file, the class is switched, but the command file aborts.

Language

The **Language** field lets you select the language that the labels, help screen, prompts, and error messages use. Files with the extension .LOC (localization) on the Reflection diskettes contain the text for the various languages. English text is contained in the file R1U.LOC. In order to select a language, the appropriate file must be available.

Values:	ENGLISH (the default)	(R1U.LOC)
	FRANCAIS	(R1F.LOC)
	DEUTSCH	(R1G.LOC)
	ITALIANO	(R1I.LOC)

Keyboard/ Nationality

Some PCs have keyboards and keyboard drivers available for languages other than English. If you have a non-English keyboard, change the **Keyboard/Nationality** field to match your keyboard.

In 8-bit operation (parity set to *NONE*), the **Keyboard/Nationality** field has no effect. In 7-bit operation it limits the characters that can be entered to those that are defined for the configured keyboard and determines the replacement characters that are used during data communications. See the *ISO Replacement for 7-Bit Operations* chart on page 183.

Values: USASCII (the default)	FRANCAIS	ESPAÑOL
UK	VLAAMS	SVENSK
NEDERL.	DEUTSCH	SCHWEIZ
SUOMI	ITALIANA	SUISSE
CANADIEN	NORSK	DANSK
CANADIAN	ESP. LAT	

Note: Reflection uses the DOS country code to determine the date and time formats. When the **Keyboard/Nationality** field is anything other than *USASCII*, this choice overrides the DOS country code.

Strip 8th Bit on Text Files

When this field is set to *YES*, the 8th bit of all data read in the following operations is set to zero:

- All disk input operations including ASCII file transfer, the TYPE, TRANSMIT, and PRINT commands
- All **READ DISK** operations

With programs that use the 8th bit for special functions, you can ensure correct translation by setting **Strip 8th bit on text files** to *YES*. You must be operating in English and able to operate without using Roman 8 characters.

For example, *WordStar* files use the 8th bit for control sequences unique to *WordStar*. In order to view these files or transfer them to a mainframe system in a usable format, the 8th bit must be stripped.

If you are using the Roman 8 character set, the 8th bit must be kept. Set this field to *NO*.

See pages 177 through 186 for more information on national characters.

Values: NO (the default)
YES

Reflection command language equivalent:

```
SET STRIP-TEXT YES/NO
```

Amount of Display Memory (K)

Reflection allows you to configure the amount of memory to use for display memory. The default value is relatively small in order to preserve the maximum amount of your PC's memory for background and SHELL operations. See *Reflection As a Pop-Up Terminal* on page 265. Note that if printer and forms buffers are configured, those amounts are subtracted from your available memory. The graphics work area in Reflection 7 requires a certain amount of memory that cannot be released without disabling graphics capability.

Values: 8-700
Default: 20

Amount of SHELL Memory (K)

SHELL is a means of executing DOS commands and batch files from within Reflection. You can use SHELL to get to the DOS prompt even after another application and Reflection are loaded. On the Reflection command line, type **SHELL** to exit to the DOS prompt, or type **SHELL** followed by the name of the DOS program or utility. When the application is exited, Reflection returns without reloading the program. To return to Reflection from the DOS prompt, type **EXIT**.

You can use the SHELL command even after Reflection has been placed into background mode if memory has been allocated for SHELL before entering background mode. The default is *0*. SHELL memory is calculated independent of print buffer, display memory, forms buffer, and program memory used by Reflection. See page 265 for more information about background operations. See page 327 for information on state save and other memory management techniques.

Values: 0–<n> (memory available after loading Reflection)
Default: 0

Reflection command language equivalent:

```
SET SHELL-MEMORY <value>
```

Hot Key

The hot-key allows you to place Reflection in background mode in order to access DOS or another PC application. See page 265, *Reflection As a Pop-Up Terminal*, for more on this feature. The hot-key is initially set as [Alt]-(right)[Shift]. When these keys are pressed, Reflection is immediately placed into background mode; i.e., it is installed as a pop-up program. Pressing these keys again brings Reflection back to the foreground.

A value of *NONE* prevents Reflection from being placed into background mode.

Values: ALT-RSH (the default)
CTL-LSH
ALT-LSH
ALT-CTL
NONE
LSH-RSH
CTL-RSH

*

Reflection command language equivalent:
`SET HOT-KEY <value>`

Configuration

132 Column Adapter This field allows you to indicate that you are using one of the supported enhanced video adapters with a 132-column display. When set to *NONE*, Reflection uses horizontal scrolling if the right margin is greater than 80. The adapter is put 132-column mode only when Reflection's **Right margin** (Terminal Configuration, Page 2) is set to a value greater than 80. An escape sequence for switching between 80 and 132 columns is discussed on page 467 for HP and page 556 for VT emulation.

Values:	Description:
NONE	(Default) No adapter present
EVA	Tseng Labs EVA or EVA 480 or Orchid designer VGA
ULTRAPAK	Tseng Labs Ultrapak
STB-HT	STB Chauffeur-HT
STB-MR2	STB Multires-II
AHEAD	Ahead Systems 2001 Ahead Systems VGA Enhancer
ATI-WNDR	ATI Wonder Card
MODE 23H	software
PARADISE	Paradise EGA 480 or VGA Plus
EVEREX	Everex EV657 or EV659
GENOA	Genoa SuperEGA HiRes
MODE 24H	software
AST 3G+2	AST 3G Plus-II

AST VGA	AST VGA
VIDEO 7	VEGA VIDEO 7

MODE 23H and *MODE 24H* let you try using a 132-column adapter that is not specifically listed as an option for this field. Both require a TSR (terminate and stay resident) program that chains to interrupt 10. MODE 24H differs from 23H in that Reflection does not attempt to load a font.

If you select one of the following values for this field, your monitor must be in *mono mode* in order for 132 columns to display:

ULTRAPAK
STB-HT

For the following choices, your monitor must be in *color mode* for 132 columns to display:

GENOA
AST 3G+2
AST VGA
VIDEO 7

Chapter 19

User Key Configuration

This chapter explains how to define the user set of function keys to make working with Reflection faster and easier in both HP and VT emulation modes. By redefining the meanings of function keys [F1] through [F8], you can create one-stroke short cuts to perform repetitive, multi-stroke operations. For more technical information on escape sequences related to user keys, see page 461. Also see *Defining the User Keys Programmatically* on page 462.

A user key definition consists of the following:

- A 16-character label (two 8-character fields)
- A string of up to 80 characters to be generated when the key is pressed
- An attribute that determines exactly how Reflection processes the string when the key is pressed

19.1 Displaying User Keys

Press [F9] (or [Alt]-[U]) to display the currently-defined labels for the user keys. When you first load Reflection, the user keys are labeled f1 through f8. These keys can be defined by escape sequences ^{E}*Cp* through ^{E}*Cw*.

Note: When an HP 3000 program takes over the user keys, any definitions you have made may be erased. To restore the configured values for user keys, press [Alt]-[F]. Only saved or activated user key definitions are restored.

If the host program you are running does not define user keys (for example, *MCBA*), any definitions you make may interfere with running the host application; the host program expects the function keys to contain their default values. Press [Shift] with the user key to send the default definition.

Changing User Key Labels

The height of the user key labels may be set to two, one, or no screen rows. There are two ways to do this:

- Configure the number of user label lines on the Terminal Configuration, Page 1 menu. There are hardware restrictions on the number of user label lines settings; see page 105 or 123 for details. When the user keys are enabled, either from the keyboard or by the host, they occupy the configured number of rows.
- Display the user keys with [F9], and then press [F9] again to toggle the height of the labels. With each subsequent stroke of [F9], the height of the label rows changes to two, one, and then no rows. If you have 27 display lines, labels can only be two lines high or invisible. Toggling the height with [F9] is effective only for the current activation of the user keys. When the host system re-enables them, or when [F9] (or [Alt]-[U]) is pressed, the number of rows reverts to the configured value.

Default User Key Definitions

The default user key definitions are as follows:

- The attributes are all **T** (transmit-only)
- The labels are **f1** through **f8**
- Each function key has a unique two-character escape sequence

19.2 Defining User Keys

To create user key definitions, press [Ctrl]-[F9] to bring up the definition menu.

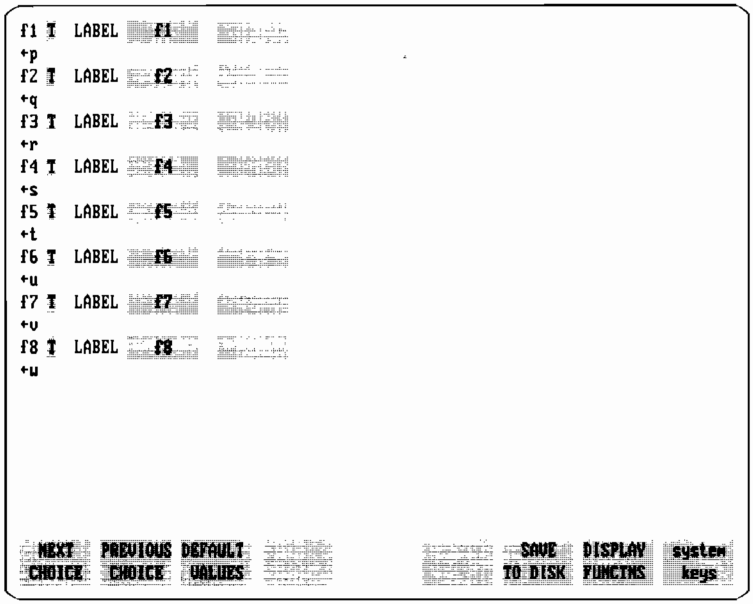


Figure 28: User Key Definition Menu

The screen displays a user key definition menu with two screen rows for each definition. The first row contains the function key label identification (f1, f2, etc.), a one-letter attribute field displayed in inverse video, the word *LABEL*, and two eight-character inverse video fields that make up the key's label. The second row contains space for a string of up to 80 characters that is generated when you press the key.

Move the cursor from field to field with Tab and Shift-Tab (Backtab) or ↑/↓. The current field contains a blinking cursor.

One of three attributes must be assigned to each user key definition. Change attribute fields by pressing F1 or F2, **NEXT CHOICE** and **PREVIOUS CHOICE.**

T Transmit-only. In remote mode, Reflection transmits the user key string to the host computer after completing a block transfer handshake, automatically transmitting a carriage return. In local mode, the key has no effect.

N Normal keyboard input. The string is treated exactly as if it had been entered from the keyboard. A carriage return is *not* automatically transmitted. If Reflection is in local mode, the string is displayed on the screen, and embedded escape sequences are executed locally. In remote mode, with local echo off, the string is transmitted to the host computer, and is executed and displayed only if the host system echoes.

L Local-only. The user key string is executed locally but is not transmitted to the host system.

In the two label fields, type in the user key label exactly as it is to appear when the user keys are activated. The first field contains the upper half of the label, and the second field contains the lower half. For single-line user key labels, make the first part of the label as descriptive as possible, since only the upper half is displayed.

In the second line of each user key definition, enter the exact character string the user key is to generate. Include escape sequences and ASCII control codes within the string by pressing [F7], the **DISPLAY FUNCTNS** key, before entering the control codes. Disable display functions mode immediately after entering the desired control code by pressing [F7] again. When entering data into the label and definition fields, you can use all the editing keys ([Ins], [Del], clear line, etc.) as if format mode were active. The only exception is the clear display ([Alt]-[J]) function, which is disabled.

Reflection Commands in User Keys

User keys can be loaded with sequences that include command language keywords. Select *L* (local) and enter the following sequence:

$^{E}{}_{C}$&oC<Reflection command>$^{C}{}_{R}$

Display functions must be on when you press [Esc] and [Return]. ^{E}C appears either as a left arrow or ^{E}C; the carriage return looks like a musical note or ^{C}R.

You could, for example, label a user key as DIR A: and include the following sequence:

$^{E}{}_{C}$&oCDIR A:$^{C}{}_{R}$

Pressing this user key would list the directory of a diskette in the A: drive.

User Key Applications

Transmit-only user keys can be used to store commonly used strings such as program run commands. Normal user keys can be used to store commands that have changing parameters or that dial modems. Local user keys are useful for Reflection commands and escape sequences. Some examples follow:

- Store the sequence that invokes a host program you use frequently in a *transmit* function key. When you press the key at the host prompt, the program is invoked.
- Store a sequence where the parameters change occasionally in a *normal* function key. The command is not immediately executed when you press the user key. It waits for you to add any information, such as a filename, and then press [Return].
- The *local* user key definition of $^{E}_{C}$&oCSHELL BASIC$^{C}_{R}$ invokes Reflection's SHELL command and start the BASIC interpreter. When you leave BASIC, you return directly to Reflection.
- Assume the file LOGON.RCL is a Reflection command file that dials your host computer and goes through the logon and password sequence. (See the *Command Language Manual* for examples.) Then, by storing $^{E}_{C}$&oCLOGON.RCL$^{C}_{R}$ in a *local* user key, you could invoke this command file and, in one keystroke, log on to your system.
- You can build function labels using escape sequences. See page 462.

Activating and Saving User Key Definitions

If you make any changes to the user key definitions, they are automatically activated when you leave the definition menu. These definitions can be saved in any configuration file you choose. After completing the user key definition menu, press [F6] **(SAVE TO DISK)**, enter the configuration filename, then press [Return].

Storing and Recalling User Key Definitions

You can store Reflection's user key labels for later recall (similar to [Alt]-[F]) with the STORE and RECALL commands. Because host application programs typically take over the user key definitions, Reflection allows you to temporarily store values that can later be reset.

Section 5

Keyboard

Chapter 20

Using the Keyboard

Reflection uses your PC keyboard to emulate an HP or VAX keyboard. This chapter explains what PC keystrokes are used to perform terminal functions. It also introduces Reflection keyboard functions.

20.1 Special Keys on the IBM PC

Certain keys on the PC keyboard are assigned to perform specific terminal functions:

Return

The Enter key, often marked by a bent left arrow located to the right of the character keys on the PC keyboard, is referred to as Return in this manual. It executes commands and transmits a carriage return.

Enter

Reflection uses the keypad + key as the HP Enter key. Enhanced keyboards use their own keypad Enter key. When the manual says to press Enter, use this key, not Return. Shift-F10 may also be used. You can configure Return to be Enter with the following SET command:

```
SET RETURN=ENTER YES
```

You can also remap Enter to another key. *Keyboard Remapping* begins on page 221.

CapsLock

When you press CapsLock, C appears between the function key labels. Letters can be forced to lowercase by holding down Shift.

NumLock

When you press NumLock, an **N** appears between the function key labels. NumLock affects the meaning of the keys on the numeric keypad. When **N** is visible, these keys generate numerals. When **N** is absent, pressing the keypad keys causes the cursor to move or the display to scroll.

You can temporarily reverse the effect of NumLock by holding down Shift while pressing a keypad key.

20.2 The Cursor

In Reflection, the cursor usually takes the form of a blinking underline. In insert character mode, it is displayed as a blinking block. The cursor style is configurable on the HP Terminal, Page 2 menu and the VT Terminal, Page 1 menu.

Using Cursor Keys

The cursor keys are the arrow keys on the numeric keypad. When used alone, the arrow keys move the cursor one position up, down, left, or right on the screen. Pressing Ctrl with a cursor key causes the display to *scroll* in the specified direction, while the cursor remains in place on the screen.

HP Mode Cursor Movement

- Home moves the cursor to the beginning of the current line.
- Ctrl-Home moves the cursor to the beginning of display memory.
- ↑ moves the cursor up one line. If the cursor is already on the first screen line, ↑ moves it to the bottom line.
- Ctrl-↑ scrolls the window up one line as it rolls the display down.
- PgUp scrolls the window up 24 lines. PgUp also sends the cursor to the first column, first row of the screen.
- Ctrl-PgUp moves the cursor to the first row of the screen.
- ← moves the cursor one column to the left. If the cursor is in the first column of the line, ← moves it to the last column of the previous line.
- Ctrl-← scrolls the window 1 column to the left if the right margin is set beyond 80 (or beyond 132 in 132-column mode) and the left margin

Keyboard Functions

is already scrolled horizontally off the screen. You can change the number of columns for each horizontal scroll with the **Cols/horizontal scroll** field of the Terminal Configuration, Page 2 menu.

- [→] moves the cursor one column to the right. If the cursor is in the last column of the row, [→] moves it to the first column of the next line.
- [Ctrl]-[→] scrolls the window 1 column to the right if the right margin is set beyond 80 (or beyond 132 in 132-column mode). You can change the number of columns for each horizontal scroll with the **Cols/horizontal scroll** field of the Terminal Configuration, Page 2 menu.
- [End] moves the cursor to the end of the current line.
- [Ctrl]-[End] moves the cursor to the end of display memory.
- [↓] moves the cursor down one line. If the cursor is on the last line of the screen, [↓] moves it to the first line of the screen.
- [Ctrl]-[↓] scrolls the window down one line as it rolls the display up.
- [PgDn] scrolls the window down 24 lines. [PgDn] also sends the cursor to the first column, first row of the screen.
- [Ctrl]-[PgDn] moves the cursor to the bottom line of the screen.

VT Mode Cursor Movement

- [Ctrl]-[Home] moves the cursor to the beginning of display memory.
- [Home] has no effect.
- [↑] moves the cursor up one line.
- [Ctrl]-[↑] scrolls the window up one line as it scrolls the display down.
- [Ctrl]-[PgUp] scrolls the window up 23 or 24 lines, depending on how many lines are visible on the screen. [Ctrl]-[PgUp] also sends the cursor to the first column, first row of the screen.
- [PgUp] has no effect.
- [←] moves the cursor one column to the left.
- [Ctrl]-[←] scrolls the window 1 column to the left if the right margin is set beyond 80 (or beyond 132 if 132-column mode) and the left margin is already scrolled horizontally off the screen. You can change the number of columns for each horizontal scroll with the **Columns per horizontal scroll** field of the Terminal Configuration, Page 2 menu.

- [→] moves the cursor one column to the right.
- [Ctrl]-[→] scrolls the window 1 column to the right, if the right margin is set beyond 80 (or beyond 132 in 132-column mode). **Columns per horizontal scroll** on of the Terminal Configuration, Page 2 menu changes the number of columns to be scrolled.
- [Ctrl]-[End] moves the cursor to the end of display memory.
- [End] has no effect.
- [↓] moves the cursor down one line.
- [Ctrl]-[↓] scrolls the window down one line as it scrolls the display up.
- [Ctrl]-[PgDn] scrolls the window down 23 or 24 lines, depending on how many lines are visible on the screen. It also sends the cursor to the first column, last row of the screen.
- [PgDn] has no effect.

In VT mode, the function keys [F5] through [F8] act as cursor movement keys when the **pf key** labels are visible. The **pf key** labels are useful when Reflection is emulating a VT terminal; see page 174.

20.3 Edit Keys

[Del] Deletes the character at the cursor position.

[Ins] Toggles Reflection into and out of insert mode. In insert mode, new characters move existing characters to the right of the cursor.

When you press [Ins], an **I** appears between the function key labels. The cursor shape also changes.

[Backspace] Moves the cursor backwards one character at a time.

In VT mode, **Backspace key** on the Terminal Configuration, Page 1 menu can be set to *BKSP* or *DEL*. When set to *BKSP*, [Backspace] generates an ASCII backspace character (ASCII 8); when configured to *DEL* (the default), [Backspace] generates a delete character (ASCII 127). When communicating with a VAX/VMS host, this field should be set to *DEL*. See page 127 for information on configuring backspace.

20.4 Operations Keys

Most operations keys don't generate characters, at least not by themselves. Instead, they either alter the meaning of characters typed with them, or cause some operation to be executed on the computer.

Enter In HP mode only, press + or Enter on the keypad. Shift-F10 can also be used.

Esc Pressing Esc generates an invisible character that tells Reflection that you are going to enter an *escape sequence.* Unlike a shift key, Esc must be released before entering the rest of the escape sequence.

An escape sequence is a string of characters, beginning with the escape character, that make up a command. Usually this is a command from the host to the terminal (in this case, Reflection) to perform some action such as moving the cursor to a particular row and column. Escape sequences can be entered at the keyboard and sent to the host using the TRANSMIT command, or sent to Reflection using the DISPLAY command.

ScrollLock When you press ScrollLock, an **S** appears between the function key labels.

HP Press ScrollLock to scroll display memory without holding down Ctrl with the arrow keys. Use Shift- or Alt-ScrollLock to start and stop the display of data from the host. **Receive pacing** on the Datacomm Configuration menu must be *XON/XOFF*.

When you press Alt or Shift-ScrollLock, an **H** appears between the function key labels indicating that the *hold screen* function has been activated. To continue scrolling, press Alt- or Shift-ScrollLock again. This function is equivalent to the HP Stop key. Use Pause on some keyboards.

VT Use Alt-ScrollLock to scroll display memory without holding down the Ctrl key along with the arrow keys. ScrollLock alone allows you to start and stop the scrolling of display memory. When you press ScrollLock, an **H** appears between the function key labels indicating that the *hold* function has been activated.

Break — To activate the Break key, press Ctrl-ScrollLock. Break is used in remote mode to interrupt the flow of data from the PC to the host computer. See SET BREAK-LENGTH in the *Command Language Manual* for information on setting the length of the break.

PrtSc — The Print Screen key, as its name indicates, prints the visible portion of display memory. In Reflection, PrtSc can be used alone or in combination with Ctrl, Alt, or Shift.

These three options are discussed on page 288.

Unshifted, PrtSc generates an asterisk. In VT mode an unshifted PrtSc produces a comma.

20.5 VT PF Keys

VT PF keys are available through the F2 key on the VT system keys. The PF keys are also accessible from any Reflection screen by pressing Alt-P.

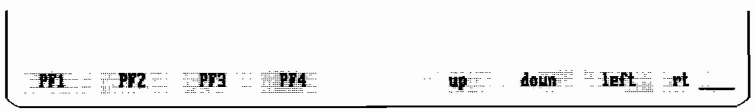


Figure 29: VT PF Keys

Follow these guidelines when using pf keys:

- Many host programs use the VT100 terminal PF1 through PF4 keys for special commands. If you want to issue these commands, the pf key labels must be displayed on the screen.
- Some host applications use the numeric keypad keys for program commands. In Reflection, NumLock must be active in order to use the keypad in this way. However, the numeric keypad is also used for cursor movement in Reflection. The pf key labels offer an alternative to the numeric keypad arrow keys, with function keys F5 through F8 operating as alternate cursor control keys.

- The pf key labels can be configured to be the initial label set on the Terminal Configuration, Page 1 menu.
- The four underscores following *rt* on [F8] correspond to the VT100 LEDs (Light Emitting Diodes). These can be used by application programs to indicate various conditions. When the host sends the LED control function, an asterisk appears in the corresponding position, 5 through 8, of the [F8] label. See page 568 for control function information.

[illegible]

Chapter 21

National Keyboards and Characters

This chapter explains how to enter national characters, how Reflection handles national characters, and the **Parity** and **Keyboard** configuration fields.

The entry of a national character depends on three factors:

- The character
- Your keyboard
- Whether you are operating in 7- or 8-bit mode

Some PCs have keyboards and keyboard drivers available for languages other than English. If the character you wish to type is on your keyboard, simply striking the key produces the character. Reflection does nothing to alter your keyboard's characters no matter what keyboard or language you have selected on the Global Configuration menu.

21.1 Typing Non-English Characters

If a character is not on the keyboard you are using, there are three ways to produce it:

- Use an *extend* sequence, then type a diacritical mark (e.g., an ¨ or ^) followed by a character.
- Use the Alt key in combination with the numeric keypad to generate characters from the IBM extended character set.

Extend sequences

If the character you wish to type appears on the extended keyboard (see the following keyboard diagram entitled *Mapping of Extended Characters*), you can use a two-stroke *extend* sequence. Type [Alt]-[Z] followed by the appropriate key from the extended keyboard. (Unlike on the HP terminal, you do not keep the [Alt]-[Z] keys depressed when you enter the extended character.) If the next character is not a valid extended character, the [Alt]-[Z] sequence is ignored.

If your keyboard does not have diacritical marks on the keycaps, and the character you wish to type is an accented character that does not appear on the extended keyboard, you must use a three-stroke extend sequence. Type [Alt]-[Z] followed by the appropriate diacritical mark from the extended keyboard, and then the character you wish to accent. For example, to produce Â, type [Alt]-[Z], then type ^ from the extended keyboard ([Y]), then type A.

Extended sequences are independent of the keyboard and DOS keyboard drivers installed on your PC; the position of the characters shown on the extended keyboard layout is the same for all keyboards. The extended keyboard shows how Roman 8 characters have been mapped to your keyboard.

If you are operating in 7-bit mode, many characters are not available. See *ISO Replacement for 7-bit Operations* on page 183.

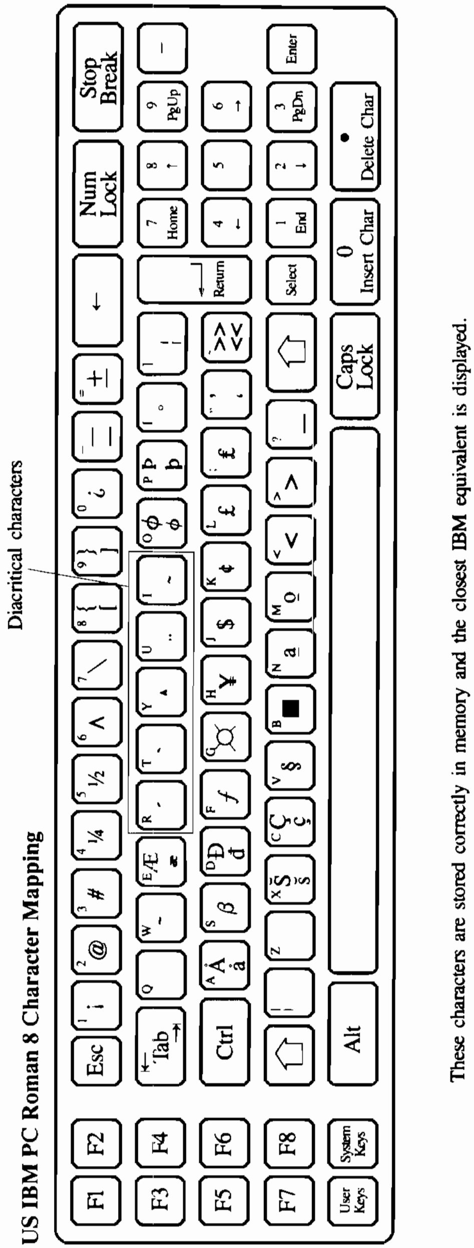


Figure 30: Mapping of Extended Characters

Generating Characters from the IBM Extended Set

The [Alt] key can be used with the numeric keypad to generate characters from the IBM extended character set. For instance, if you press [Alt] and hold it down while pressing [1], [4], and [2] in succession from the numeric keypad, you produce an A with an umlaut: Ä.

When you are running Reflection 7, the numeric keypad is used for emulating graphics functions. In order to generate characters from the IBM extended character set, press both [Alt] and [Shift] while typing the number sequence.

21.2 Reflection and IBM Code Pages

Because Reflection is designed to be used as an HP 3000 terminal, it uses the ASCII character set for most common characters and the Roman 8 extended character set for national and special characters. The PC does not use the Roman 8 character set. In the U.S, it uses Code Page 437, which is a different collection of national and special characters. Code Page 437 and Roman 8 contain many of the same characters, but there are many differences. Even when similar characters exist in both character sets, they are often represented by different numbers. For example, in Roman 8 the uppercase cedilla (Ç) is represented by the number 180, while in Code Page 437 it is represented by the number 128.

If you rarely use national characters, you do not need to make any changes to Reflection or the PC code page. Reflection will function as it always has. If you do need national characters, set up your PC to allow code page switching and use a code page, such as Page 850, which has most of the Roman 8 characters in it. Reflection uses internal tables to translate between the PC and the Roman 8 character sets, so using characters that exist in both sets preserves the most characters.

For example, when the character Ã is translated from Reflection's own font to Code Page 437, it becomes an A. When it's sent back to the host or to a printer that supports Roman 8 characters, there is no way to know that the character was originally something other than an A, so an A is sent. Even when Code Page 850 is used, the translation is not completely accurate.

21.3 Translation of Characters

There are times when translation should not occur at all in order to completely preserve what is being read from disk. DISABLE-TRANSLATION should be set to *YES* in these cases (the default setting is *NO*). If DISABLE-TRANSLATION is set to *YES*, Reflection assumes that the characters being read from disk are already in Roman 8 nothing is translated. It is up to you to determine whether the file should be translated or not.

There are two basic issues for deciding if Reflection should translate between Roman 8 and the PC's character set:

- The source of the characters
- The destination for the characters

The following table lists the source and destinations for text in display memory and disk files. *NO* means that translation should occur (translation is *not* disabled); *YES* means that translation should not occur (translation is disabled).

Table 11
SET DISABLE-TRANSLATION

Source of file read from disk	**Destination or Use for File**		
	HP 3000	PC Application	Display Memory
HP 3000 file	YES	NO	YES
Saved from display memory	N/A	NO	YES
PC Application file	NO	YES	NO

DISABLE-TRANSLATION affects file transfers before the point of reading from disk. The method and actual transfer of the file is configured separately.

21.4 File Transfer Considerations

Strip 8th bit on text files on the Global Configuration menu affects ASCII file transfers to the host. If you are using national characters in 7- or 8-bit mode, set this field to *NO*.

If you are operating in ISO-7 mode (i.e., in 7-bit mode with the **Keyboard/Nationality** field configured to a value other than *USASCII*), you may want to modify the default file transfer configuration fields. The ASCII translation fields **Change Roman 8 to ISO-7** and **Change ISO-7 to Roman 8** should probably be set to *YES*.

21.5

8-Bit and 7-Bit Operation

Transmission of Roman 8 characters depends on the setting for **Parity** on the Datacomm Configuration menu. If it is set to *NONE*, Reflection is operating in 8-bit mode. All 8 bits of each byte are used for data. This allows all Roman 8 characters to be sent and received so that every character on the extended keyboard can be used.

When **Parity** is set to a value other than *NONE*, Reflection is operating in 7-bit mode. This means that the 8th bit of each byte is used for parity and is not available for use as data. Since the Roman extension characters all use the 8th bit, they cannot be directly sent or received in 7-bit operations. In 7-bit operations some characters in the USASCII character set are replaced by certain characters from the Roman extension as shown in the table on page 183. This happens only when the **Keyboard/Nationality** field on the Global Configuration menu is configured to a value other than *USASCII*. This is often called *ISO-7 mode* because the International Standards Organization has defined which USASCII (7-bit) characters are to be replaced.

CONFIGURED KEYBOARD	CHARACTERS													
USASCII	#	’	<	>	@	[	\	]	^	‘	{	|	}	~
Danish	§	´	<	>	@	Æ	Ø	Å	^	`	æ	ø	å	¨
Dutch	#	’	<	>	@	ç	\	§	^	`	ƒ	¦	´	¨
Finnish	#	’	<	>	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
Flemish and French	£	’	<	>	à	°	ç	§	^ *	‘	é	ù	è	¨ *
English Canadian and French Canadian	#	’	<	>	@	[	ç	]	^	`	é	Ç	É	¨
French Swiss and German Swiss	£	´	é	è	à	°	ç	§	^	`	ä	ö	ü	¨
German	£	’	<	>	§	Ä	Ö	Ü	^	‘	ä	ö	ü	ß
Italian	£	’	<	>	§	°	ç	é	^	ù	à	ò	è	ì
Norwegian	#	´	<	>	@	Æ	Ø	Å	^	`	æ	ø	å	¨
European Spanish	#	´ *	<	>	@	¡	Ñ	¿	°	` *	’	ñ	ç	¨ *
Latin Spanish	#	´ *	<	>	@	¡	Ñ	¿	^	` *	’	ñ	ç	¨ *
Swedish	#	’	<	>	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
U.K.	£	’	<	>	@	[	\	]	^	‘	{	|	}	~
Line Drawing Elements	┳	╤	╫	╪	╞	╡	¨	╨	·	╞	╡	¨	╨	·

Note: In 7-bit operations, characters marked by an "*" are diacritical marks.

Figure 31: ISO Replacement for 7-bit Operations

21.6 Character Set Options

The following instructions assume that a PC with an EGA or equivalent video adapter is being used. The monochrome graphics adapter and CGA do not allow alternate character sets to be used. If your PC does not recognize the Norwegian or French Canadian character set code pages, use the /N and /L switches respectively (see page 16) to force Reflection to recognize them.

Note: To use code page settings, parity must be set to *NONE.*

Preparing Code Page 850

Refer to the *IBM Disk Operating System Reference Manual* for DOS 3.3 for information about setting up a code page. The following reviews some basic information:

- Before a code page can be selected in DOS, it must be prepared.
- Every device that uses the prepared code page must be defined with a DEVICE command in CONFIG.SYS.
- If ANSI.SYS is used, it must be defined before DISPLAY.SYS.
- The COUNTRY command is still used by DOS. It has an added parameter that allows you to select a code page along with the country time and date format you prefer.
- The AUTOEXEC.BAT file is typically used to prepare the code page. CODEPAGE (CP) is a new parameter of the DOS MODE command for preparing code pages.
- CHCP is used to change code pages. It can be used within the AUTOEXEC.BAT file, but is more typically used at the DOS prompt. You can use CHCP to switch between code pages as needed.
- The NLSFUNC command must be issued before you can use CHCP to switch between code pages dynamically.

The following example sets up the display as an EGA with a hardware Code Page of 437 and a prepared Code Page of 850. The printer is set up as an IBM Proprinter with a default hardware code page of 437.

CONFIG.SYS

```
COUNTRY=001,437,C:\DOS\COUNTRY.SYS
DEVICE=C:\DOS\ANSI.SYS
DEVICE=C:\DOS\DISPLAY.SYS CON=(EGA,437,1)
DEVICE=C:\DOS\PRINTER.SYS PRN=(4201,437,1)
```

AUTOEXEC.BAT

```
NLSFUNC C:\DOS\COUNTRY.SYS
MODE CON CP PREPARE=((437 850) C:\DOS\EGA.CPI)
MODE PRN CP PREPARE=((437 850) C:\DOS\4201.CPI)
CHCP 850
KEYB US,437,C:\DOS\KEYBOARD.SYS
```

21.7 Character Set Conversion Utility

CHARTRAN is a translation utility program that converts the characters in a file from the Roman 8 character set to an IBM code page and vice versa. It has a number of pre-defined translation tables. The type of table is specified by a switch added to the CHARTRAN command. Type **CHARTRAN** at the DOS prompt with no parameters to see a summary of the options.

```
CHARTRAN /<switch> <source file> <destination file>
```

The possible values of **/<switch>** are as follows:

/1 Translates Roman 8 to IBM (Code Page 437)

/2 Translates IBM (Code Page 437) to Roman 8

/5 Translates Roman 8 to IBM (Code Page 850)

/6 Translates IBM (Code Page 850) to Roman 8*

Other CHARTRAN parameters:

/H

General help information.

* The other values for **<switch>** (/3, /4, /7, and /8) apply only to translations from or to the DEC Multinational character set.

/F=<filename>

CHARTRAN can also accept a user-defined translation table. The table file must contain 256 entries. Each entry must be in the form **<nnn>**, where **<nnn>** is a right-justified, zero-filled number. The numbers may run together or they may be separated by any non-numeric character (for example, **001,012,345**). The entries may be placed on multiple lines, but each entry must reside entirely on one line. Comments may be placed in the file, but must be enclosed in single or double quotes.

The data in the **<source file>** is converted from one character set to another and then stored in the **<destination file>**.

21.8 Printer Support for Roman 8

Reflection attempts to translate characters to the character set of your attached printer. It uses the selection you have made in the **Control codes** field on the Printer Configuration menu to determine what character set is available on the printer.

The **Control codes** selection also defines what codes the printer recognizes as instructions to perform actions such as beginning compressed printing. For example, selecting *IBM PRO* indicates that the printer font matches the code page being used by the PC. See page 138.

Sources of Characters

There are three possible sources of characters for printing:

- Reflection's display memory, where characters are stored in Roman 8.
- Directly from the host via record mode. Presumably these characters are in Roman 8, unless host-based software pre-translates the data for the PC printer.
- The PC itself—the characters can come from a disk file or the keyboard. When the character source is the PC, it is usually coded in the IBM 8-bit character set that is currently active.

Set DISABLE-TRANSLATION to *YES* if the file to be printed was saved to disk as a Roman 8 file (with Roman 8 characters left untranslated when it was saved to disk).

Guidelines for Character Translation

To determine what character is received and printed by the printer you must know the source of the characters to be printed.

Due to incompatibility between character sets, it may not be possible for Reflection to perform a particular translation. For example, the symbol A acute (Á) exists in the Roman 8 character set, but not in IBM's Code Page 437. If Reflection receives this character from an HP 3000 host and the PC is using Code Page 437, there would be no way to print the character. In this situation, Reflection maps the A character to the closest character available in Code Page 437—a capital A—so that an unaccented capital A appears in the printout.

Roman 8 and HP Printers

Selecting *HP* or *PaintJet* indicates that HP's Roman 8 character set is active on the printer. In this case Reflection does not attempt to translate Roman 8 characters, but passes them directly to the printer.

When printing a file from disk, however, the character set is translated by default. If the source of the file is already in Roman 8, use the following SET command to keep this translation from occurring:

```
SET DISABLE-TRANSLATION YES
```

In any other case, SET DISABLE-TRANSLATION should be set to *NO*. Printer control codes determine whether or not Roman 8 characters are sent directly or translated after the characters are read from disk. Refer to your printer's reference manual to see if Roman 8 support is available.

Code Pages and the IBM Proprinter

The IBM Proprinter can print 8-bit characters and is supported by the DOS 3.3 National Language Support. Proprinters can be configured to use either the standard IBM8 font (Code Page 437) or the new Code Page 850.

Configuring a Proprinter to use a particular code page is handled in DOS 3.3 via the National Language Support facilities. The necessary device driver for the Proprinter is specified in CONFIG.SYS:

```
DEVICE=C:\DOS\PRINTER.SYS PRN=(4201,437,1)
```

The MODE command necessary to prepare the proper font to the Proprinter is typically coded in the AUTOEXEC.BAT file:

```
MODE PRN CP PREPARE=((850) C:\DOS\4201.CPI)
```

Type **CHCP 850** at the DOS prompt to select Code Page 850 for all devices. When the PC is booted, the device driver for the (Model 4201) Proprinter is loaded. When the CHCP command is issued, the font for Code Page 850 is downloaded to the printer.

Other Printers

HP printers can be configured to use a Roman 8 font and IBM Proprinters can be configured to use the currently active code page. There is no standard for national characters on other printers. Reflection handles other printer selections in the same manner as the *IBM PRO* selection.

Chapter 22

HP Keyboard Functions

Many of these functions are listed on the Reflection Help screen. The following information assumes Reflection has not been remapped. To change the default keystrokes, see *Keyboard Remapping* on page 221.

22.1 Quick Reference

Table 12
HP Keyboard Functions

Function	Keystroke
Break	[Ctrl]-[ScrollLock]
Clear alpha display	[Alt]-[J]
Clear keyboard buffer	[Alt]-[V]
Clear line	[Alt]-[K]
Clear print buffer	[Ctrl]-[F1]
Command line	[Alt]-[Y]
Config keys	[Alt]-[C]
Connection reset	[Ctrl]-[F8]
Define user keys	[Ctrl]-[F9]
Delete character	[Del]
Delete line	[Alt]-[D]
Delete wraparound	[Ctrl]-[Del]
Disconnect	[Ctrl]-[F10]
Enter*	[+] (keypad) or [Shift]-[F10]
Error recap	[Alt]-[E]

* Use keypad [Enter] on an Enhanced keyboard.

*

HP Keyboard Functions

Exit to DOS	Alt-X	
Extend key	Alt-Z or Alt-Shift-<key>	
Function key recall	Alt-F	
Hard reset	Alt-R	
Help	Alt-H	
Home down	Ctrl-End	
Home up	Ctrl-Home	
Hot key	Alt-(right) Shift	
Insert character	Ins	
Insert line	Alt-I	
Insert wraparound	Ctrl-Ins	
Interrupt command	Ctrl-Y	
Keypad	NumLock	
Modes keys	Alt-M	
Move cursor down/up	↓/↑	
Move cursor left/right	←/→	
Move to bottom row	Ctrl-PgDn	
Move to line end/start	End/Home	
Move to top row	Ctrl-PgUp	
Next page	PgDn	
Previous page	PgUp	
Print screen	Alt-PrtSc	
Print screen with labels (DOS)	Shift-PrtSc	
Reset handshake	Alt-Q	
Reset typeahead	Alt-T	
Reset video	Alt-=	*
Roll down	Ctrl-↑	
Roll left	Ctrl-→	
Roll right	Ctrl-←	
Roll up	Ctrl-↓	
Soft reset	Alt-S	
State save	Alt-B	*
Stop key*	Shift-ScrollLock or Alt-ScrollLock	
System keys	F10 or Alt-A	
Toggle print logging	Ctrl-PrtSc	
Toggle user labels	F9	
User keys	F9 or Alt-U	

* This key works only when **Receive pacing** is *XON/XOFF*. The H indicator shows when the Stop (or Hold) function is activated; S appears when ScrollLock is pressed, indicating that cursor scrolling has been activated.

22.2

Keyboard Functions

This reference describes local keyboard functions in HP mode. (VT keyboard functions are described on page 209.) Many of these functions are also listed on the HP Help screen ([Alt]-[H]).

Break ([Ctrl]-[Break])

Esc sequence: None
Availability: Always

Transmits a break signal of approximately 200 milliseconds to the host computer, assuming that the PC is connected to a host. The break signal is sent even if Reflection is in local mode.

The Reflection command language equivalent is BREAK. The length of the break signal can be configured; see SET BREAK-LENGTH in the Reflection *Command Language Manual.*

This use of [Ctrl]-[Break] differs from some other programs' break function. Many programs use this key combination to exit to the DOS command prompt. Compare with *Exit to DOS*, ([Alt]-[X]). Note that [ScrollLock] on the PC and older AT keyboards is [Break] without [Ctrl]. Enhanced keyboards call this the *Pause* key.

Clear alpha display ([Alt]-[J])

Esc sequence: $^{E}{}_{C}J$
Availability: Always, except at a menu

Erases all of alpha display memory from the alpha cursor position through the end of display memory. In format mode, only the characters within unprotected fields are erased, beginning with the alpha cursor location.

Clear keyboard buffer ([Alt]-[V])

Esc sequence: None *
Availability: Always *

When **Typeahead** is set to *YES* (Terminal Configuration, Page 1), *
Reflection buffers all keystrokes until it receives a DC1 from the HP *
3000, indicating that the host is ready to receive more input. If you type *
ahead of the HP 3000 and then decide not to transmit the buffered *
keystrokes, clear the keyboard buffer by pressing [Alt]-[V]. After *
clearing the buffer, you can continue to type ahead of the host. *

Clear line ([Alt]-[K])

Esc sequence: ^{E}c*K*
Availability: Local and remote mode (at the user key definition menu, file transfer screen, and the command line)

Erases all characters from the alpha cursor through the end of the line. In format mode, this key works only if the alpha cursor is within an unprotected field. All characters from the alpha cursor position through the end of the field are erased.

Clear print buffer ([Ctrl]-[F1])

Esc sequence: None
Availability: Always

Clears the contents of Reflection's internal printer buffer. The printer continues to print if it has a built-in buffer. To stop sending data to the printer buffer, use the soft reset ([Alt]-[S]) function.

Command line ([Alt]-[Y])

Esc sequence: None
Availability: Always, except at a menu

Causes Reflection to display the command line. Exit the command line by pressing [Esc] or [F10], or by leaving the command line blank and pressing [Return].

Config keys ([Alt]-[C])

Esc sequence: None
Availability: Always, except when disabled by the ^{E}c*&jS* escape sequence, or when the configuration is locked

Displays the configuration keys. If the selection of system, modes, and user keys has been disabled, pressing this key causes a beep and the function is not performed.

Connection reset ([Ctrl]-[F8])

Esc sequence: ^{E}c*&bR*
Availability: Always

- If using a Hayes Smartmodem 1200B, resets to its power-on state. Sets the *out 1* bit of the modem control register to 1 for at least 50 milliseconds as described in the Hayes manual.

- This is also called CI mode. If using a LAN, this keystroke suspends the current LAN session.

Define user keys ([Ctrl]-[F9])

Esc sequence: ^{E}cj
Availability: In remote mode

Brings up the user-key menu for you to define user keys. See page 162.

Delete character ([Del])

Esc sequence: ^{E}cP
Availability: Local and remote mode (the user-key definition menu, file transfer screen and the command line)

With format mode off, [Del] deletes the character at the alpha cursor position and moves all following characters on the line one position to the left.

When the alpha cursor is to the left of the right margin, only the characters up to the right margin are moved left, and a space is inserted at the right margin.

When the alpha cursor is to the right of the right margin, all characters to the right of the alpha cursor are moved left, and the line is shortened by one column. This also applies to characters that cannot be displayed, such as display enhancements and unprotected field delimiters. When the alpha cursor is at the first column of an unprotected field, [Del] changes the field from unprotected to protected. When the alpha cursor is at the first column of a display enhancement, however, the display enhancement is not changed.

With format mode on, [Del] operates only within an unprotected field. Further, only characters that can be displayed within the unprotected field are affected. The size of the field remains unchanged, and any display enhancements or character set changes within the field are unchanged.

Delete line ([Alt]-[D])

Esc sequence: ^{E}cM
Availability: In local and remote mode, except format mode

Deletes the alpha cursor line and moves all subsequent lines in alpha display memory up one row. The alpha cursor then moves to the left margin.

Within a locked area of the screen, the size of the locked area is not changed; instead, the alpha cursor line is deleted and the line immediately following the locked area is moved up into the locked area.

Delete wraparound (Ctrl - Del)

Esc sequence: ^{E}cO
Availability: In remote mode, except format mode

When you use delete character with the wraparound function, the cursor remains stationary, the character at the cursor position is deleted, and the rest of the line is moved left one column. A blank line is considered to be 80 blank characters; all 80 characters would have to be deleted before the blank line is deleted.

Only the line containing the cursor and the next line are affected by this function. Unlike a terminal, this feature is not limited to 80 columns in Reflection.

Disconnect (Ctrl - F10)

Esc sequence: ^{E}cf
Availability: During datacomm, when the PC and the host are connected

Disconnect functions differently depending on the type of host connection. If you are connected via a serial port, Reflection lowers the Data Terminal Ready signal (DTR) for 50 milliseconds. This function disconnects a properly configured modem. Note that some modems can be configured to not respond to the disconnect signal. See *Auto-Dial Modems* on page 50.

If you are connected via a LAN, the appropriate LAN disconnect signal is transmitted with this keystroke. See the *PLUS Manual* for more information on network configurations. Also see DISCONNECT and SET DISCONNECT-ON-EXIT in the Reflection *Command Language Manual.*

Enter (+ (keypad) or Shift - F10) *

Esc sequence: None
Availability: Always

---

* Enhanced keyboards use Enter located on the numeric keypad.

In remote mode, [Enter] transfers a block of data to the host. The specifications of this block depend on *block mode*, *format mode*, and *line/page* mode settings:

- If you are operating in character mode, the current line is transferred.
- If you are operating in block mode, either the current line or all pages in memory are transferred, depending on the setting of **Line/Page** on the Terminal Configuration, Page 1 menu.
- In local HP mode, [Enter] homes the cursor and sends all of display memory to a local printer.

The Reflection command language equivalent is ENTER.

Error recap ([Alt]-[E])

Esc sequence: None
Availability: In remote mode

Displays a menu of errors that can occur in data communication. A count is displayed for each type of error on the current datacomm port. See page 71 for instructions on using this screen to diagnose and correct datacomm problems.

Exit to DOS ([Alt]-[X])

Esc sequence: None
Availability: Always

- Uninstalls Reflection from memory and returns to the DOS prompt.
- Erases everything in display memory.
- Leaves the PC logged in to the host computer if the connection to the host was active.
- Erases any state save file (R<n>STATE.STA) that is in the Reflection directory, or by the directory specified by SET REFLECT<n>=<filespec>.
- Terminates a currently executing command language program.

The Reflection command language equivalent is HARDEXIT.

When the PC is online over a modem, Reflection does not automatically hang up the phone. To hang up before exiting Reflection, press [Ctrl]-[F10], Disconnect. See *Auto-Dial Modems* on page 50.

It is recommended that you load Reflection before most PC applications; when you unload Reflection, you free up the maximum amount of memory.

Extend key ([Alt]-[Z])

Esc sequence: None
Availability: Whenever data can be entered

The character typed after the extend sequence matches the keyboard mapping shown on page 178 and gives access to the Roman 8 character set as shown on page 524.

Function key recall ([Alt]-[F])

Esc sequence: None
Availability: In remote mode

Restores the configuration file values of user-key definitions if they have been cleared by the host software or a hard reset. The Reflection command language equivalent is RECALL.

Hard reset ([Alt]-[R])

Esc sequence: $^{E}{}_{C}E$
Availability: Always, except during file transfer

A hard reset does the following:

- Emits a beep.
- Unlocks the keyboard if it was locked.
- Turns memory lock off if it was on.
- Clears alpha display memory and homes the cursor.
- Sets the margins to columns 1 and 80.
- Clears all tab stops.
- In Reflection 7, the graphics display is cleared and turned off, along with the graphics cursor.
- Sets all terminal configuration parameters to their last activated values.

- Disables the following:

  Format mode
  Display functions
  Insert character mode
  Caps lock
  Report mode
  Metric mode
  Record mode
  Log top and log bottom

  It does not change remote, block, auto linefeed, and modify all modes settings.

- Initializes serial communications ports to their last activated values and clears the receive buffer.
- Clears the printer buffer, and sends a reset command to the printer.
- Sets user keys to their last saved values. (^{E}*CE* does not parallel the keystroke in this case; the user keys are set to defaults.)
- Sends the alpha cursor home.
- Resets typeahead and clears the keyboard buffer.
- Transmits an XON if receive pacing is enabled.
- Displays modes key labels and enters terminal mode.
- Terminates a currently executing command language program.

Help ([Alt]-[H])

Esc sequence:	None
Availability:	In remote mode

Displays the Help screen. Return to the point at which you entered the Help screen with **RESUME** ([F8]).

Home down ([Ctrl]-[End])

Esc sequence:	^{E}*CF*
Availability:	In remote mode

Causes alpha display memory to roll up, so the last line in display memory occupies the next-to-last row of the screen. The alpha cursor moves to the left margin of the last row of the screen.

If the last row in alpha display memory is already on the screen in a row other than the last row, the display is not rolled up; the alpha cursor moves to the left margin of the row below the last line.

When memory lock is enabled, the alpha cursor moves to the left margin of the first row below the locked area. If the cursor is within the locked area, it moves to the first row below the locked area.

Home up ([Ctrl]-[Home])

Esc sequence: ^{E}CH
Availability: In remote mode, user-key definition menu, and file transfer

With format mode off, alpha display memory rolls down so that the first line of display memory occupies the first row of the screen. The alpha cursor moves to the left margin of the top row.

With memory lock enabled, the alpha cursor moves to the left margin of the first row below the locked area. If the cursor is within the locked area, it moves to the first row below the locked area.

With format mode on, the text rolls down as described above, but the alpha cursor moves to the first column of the first unprotected field on the screen. If there are no unprotected fields, the alpha cursor moves to the upper left corner. If memory lock is enabled, the cursor moves to the first field, even if that field is within the locked area.

Hot key ([Alt]-(right) [Shift])

Esc sequence: None
Availability: Always, if **Hot Key** (Global Configuration) is *not* configured to *NONE*.

Puts Reflection into the background, allowing access to DOS applications. Host operations can continue in the background. This keystroke can be changed; see page 157.

Insert character ([Ins])

Esc sequence: ^{E}CQ enables insert mode, ^{E}CR disables insert mode
Availability: Always

Toggles insert character mode on and off. When on, the letter **I** appears between the function key labels. The alpha cursor becomes a blinking block when insert character mode is on.

When format mode is off, characters are inserted at the alpha cursor location, and all characters to the right of the cursor move right one column. The character at the right margin is deleted as other characters shift right. If the alpha cursor is beyond the right margin when the inserted character is typed, the character at the right margin is deleted.

Any nondisplaying characters to the right of the alpha cursor position are shifted along with the displaying characters, so that the range of unprotected fields and display enhancements may be extended.

When format mode is on, insert character mode works only within the field that contains the alpha cursor. If the cursor is within a protected area, it moves to the first column of the next field before the character is inserted. Only displaying characters are moved; nondisplaying characters such as display enhancements and field definitions remain fixed.

Insert line (Alt-I)

Esc sequence: $^{E}{}_{C}L$
Availability: In local and in remote mode, except format mode

Inserts a blank line in the screen row that contains the alpha cursor. The old contents of that row and all the following rows in alpha display memory are pushed down one row. The cursor then moves to the left margin of the new (blank) row.

If alpha display memory is full, then either the first row or the last row of display memory must be deleted to make room for the new row. If the first row is not currently displayed on the screen, it is deleted; otherwise, the last row of display memory is deleted.

If memory lock is on and the alpha cursor is within the locked area when you press Alt-I, then the last line of the locked area is pushed down and becomes unlocked.

Insert wraparound (Ctrl-Ins)

Esc sequence: $^{E}{}_{C}N$ enables and $^{E}{}_{C}R$ disables
Availability: In remote mode, except format mode

When on, characters are wrapped to the next line. Unlike the terminal, Reflection does not restrict this function to 80 columns if the right margin has been configured to a number greater than 80.

Keypad (NumLock)

Esc sequence: None
Availability: Always

When NumLock is on, the keypad generates numbers. The letter N is displayed between the function key labels. When it's off, the keypad keys are the alpha cursor movement keys.

If you have both a keypad and a cursor pad, Reflection may handle these keys differently. See *Keyboard Remapping* on page 221. When the graphics keypad is enabled, the letter G appears, and the cursor keys move the graphics cursor.

Modes keys (Alt-M)

Esc sequence: ^{E}C*&jA*
Availability: Always, except when disabled by ^{E}C*&jS*

Brings up the modes keys. If the system, modes, and user keys are disabled, pressing this key causes a beep, and the function is not performed.

Move cursor down (↓, Shift-↓ with NumLock on)

Esc sequence: ^{E}C*B*
Availability: Always

Moves the alpha cursor down one row on the screen without changing the contents of the screen. If the cursor is already in the bottom row of the screen, it moves to the same column in the top row. In menu mode, the cursor moves to the next field.

Move cursor left (←, Shift-← with NumLock on)

Esc sequence: ^{E}C*D*
Availability: When the alpha cursor is visible

Moves the alpha cursor one column to the left, without changing the contents of the screen. If the cursor is already at the left column of the screen, it moves to the rightmost column of the next higher row. If the cursor is at the upper left corner of the screen, it moves to the lower right corner.

Move cursor right (→, Shift-→ with NumLock on)

Esc sequence: ^{E}C*C*

Availability: When the alpha cursor is visible

Moves the alpha cursor one column to the right, without changing the contents of the screen. If the cursor is already at the right column of the screen, it moves to the leftmost column of the next row. If the cursor is at the lower right corner of the screen, it moves to the upper left corner.

Move cursor up (↑, Shift-↑ with NumLock on)

Esc sequence: ${}^{E}CA$
Availability: Always

Moves the alpha cursor up one row on the screen, without changing the contents of the screen. If the cursor is already in the top row of the screen, it moves to the same column in the bottom row. In menu mode the cursor moves to the previous field.

Move to bottom row (Ctrl-PgDn)

Esc sequence: One occurrence of the ${}^{E}CB$ (cursor-down) sequence for each row that the alpha cursor is moved
Availability: In remote mode

Moves the alpha cursor to the bottom row of the screen, without changing its column.

This function is implemented as a series of cursor-down functions. When Ctrl-PgDn is entered and **Transmit functions** is configured to *YES*, Reflection transmits one cursor-down escape sequence for each row the alpha cursor is moved.

Move to line end (End)

Esc sequence: One occurrence of the cursor-right or cursor-left sequence (${}^{E}CC$ or ${}^{E}CD$) for each column the alpha cursor is moved
Availability: In remote mode

Moves the alpha cursor one position to the right of the last column containing data in the current row. If the right margin is greater than column 80, and if the last column that contains data is the right margin, then the cursor is moved only as far as the right margin.

If the new cursor position is either to the left or right of the current window, the alpha display is scrolled left or right so the alpha cursor is on the screen.

Move to line start ([Home])

Esc sequence: One occurrence of the cursor-left sequence (^{E}CD) for each column the alpha cursor is moved
Availability: In remote mode

Moves the alpha cursor to column 1 on the screen. If necessary, the text is rolled right as far as possible, so that column 1 appears on the screen.

Move to top row ([Ctrl]-[PgUp])

Esc sequence: One occurrence of the cursor-up (^{E}CA) sequence for each row the alpha cursor is moved
Availability: In remote mode

Moves the alpha cursor to the top row of the screen, without changing its column.

This function is implemented as a series of cursor-up functions. When [Ctrl]-[PgUp] is entered with the **Transmit functions** set to *YES*, Reflection transmits one cursor-up escape sequence for each row the alpha cursor is moved.

Next page ([PgDn])

Esc sequence: ^{E}CU
Availability: In remote mode

Displays the next page of alpha display memory. A page consists of 24 lines of display memory, unless memory lock is enabled. When memory lock is enabled, a page is 24 lines minus the number of locked screen rows.

If the last line of alpha display memory is already on the screen when you press [PgDn], the display is rolled up until the last line of display memory appears in the first unlocked screen row.

Previous page ([PgUp])

Esc sequence: ^{E}CV
Availability: In remote mode

Displays the previous page of alpha display memory. A page consists of 24 lines of display memory, unless memory lock is enabled. When memory lock is enabled, a page is 24 lines minus the number of locked screen rows. The Previous Page function leaves the locked rows unchanged and rolls the unlocked rows down. If the first line of alpha display memory is already on the screen when you press [PgUp], the operation has no effect.

Print screen ([Alt]-[PrtSc])

Esc sequence: None
Availability: Always

Prints a copy of the currently displayed screen (alpha only), not including the function key labels.

Print screen with labels ([Shift]-[PrtSc])

Esc sequence: None
Availability: Always

Prints a copy of the currently displayed screen (alpha only), including the function key labels. Only 25 lines are printed. This method is unreliable with the graphics display on. This is a DOS function, not a Reflection function.

Reset handshake ([Alt]-[Q])

Esc sequence: None
Availability: Always

If configured for a host prompt character other than NONE (null), Reflection can wait until it receives that character from the host computer before it transmits the next line or block of data. (On a LAN the result varies; see the *PLUS Manual* for more information.) If this character is not received or is garbled on reception, Reflection goes into an endless wait. Reset handshake tells Reflection to transmit anyway.

Reset typeahead ([Alt]-[T])

Esc sequence: None
Availability: Always

Typeahead allows you to type ahead of the HP 3000. When typeahead is enabled, Reflection buffers each line of keystrokes until it sees a *DC1*, indicating the HP 3000 is ready to receive. If you have logged off the system, the host does not send a *DC1*, and thus Reflection goes into an endless wait. The *DC1* can also be lost or garbled in transmission.

The Reset typeahead function temporarily disables typeahead and forces Reflection to transmit the contents of the typeahead buffer immediately. Whatever is in the typeahead buffer is sent all at once: it may not be interpreted correctly by the host. Typeahead resumes when the next *DC1* is seen, indicating communication with the HP 3000 has begun again.

Reset video ([Alt]-[=])

Esc sequence: None *

Availability: Always, except when the command line or *
configuration screen is displayed. *

This keystroke is available in Reflection 1 only. Some PC applications *
can take Reflection out of 27-line mode or alter other video effects such *
as font and palette. Use [Alt]-[=] to reinitialize the video card and *
redisplay the screen. *

Roll down ([Ctrl]-[↑])

Esc sequence: $^{E}{}_{C}T$ or $^{E}{}_{C}\&r<x>D$
Availability: In remote mode

Rolls the text in alpha display memory down one line and brings the previous line of text onto the top row of the screen. If the first line of display memory is already in the top row of the screen, then roll down has no effect.

When memory lock is enabled, only the unlocked portion of the alpha display rolls down; the locked portion remains fixed in the upper part of the screen. If the first line of display memory is already on the screen, roll down has no effect.

Roll left ([Ctrl]-[→])

Esc sequence: $^{E}{}_{C}\&r<x>L$
Availability: When the right margin is beyond 80

If the right margin is set beyond column 80, Roll left brings text beyond the right border of the screen into view. Each repetition shifts a specified number of columns off the left side of the screen and the same number of columns onto the right side. The alpha cursor remains in the same position on the screen. The number of columns scrolled is specified in the **Cols/horizontal scroll** field of the Terminal Configuration, Page 2 menu.

If the right margin column is already on the screen, this function has no effect.

Roll right (Ctrl-←)

Esc sequence: *ᴱc&r<x>R*

Availability: In remote mode, when the right margin is beyond column 80

When the right margin is set beyond column 80, roll right brings text beyond the left border of the screen into view. Each repetition shifts a specified number of columns off the right side of the screen and the same number of columns onto the left. The alpha cursor remains in the same position on the screen. The number of columns scrolled is specified in the **Cols/horizontal scroll** field of the Terminal Configuration, Page 2 menu.

When column 1 is already on the screen, this function has no effect.

Roll up (Ctrl-↓)

Esc sequence: *ᴱcS* or *ᴱc&r<x>U*
Availability: In remote mode

Rolls the text in alpha display memory up one line and brings the next line of text onto the bottom row of the screen. If the last line of display memory is already in the top row of the screen, roll up has no effect.

When memory lock is enabled, only the unlocked portion of the alpha display rolls up; the locked portion remains fixed in the upper part of the screen. If the last line of display memory is already in the screen row immediately below the locked portion, roll up has no effect.

Select key (no default keystroke)

Esc sequence: *ᴱc&P*
Availability: In remote mode

Transmits *ᴱc&P* over the datacomm. Select is used to select fields in forms or menus in HP applications. This function is typically enabled by an escape sequence (*ᴱc&k 1]*) sent from the host. The interpretation of this sequence depends on the application program. See page 495 for handshaking considerations.

Note: This function can only be activated from the keyboard if you use keyboard mapping to map it to a specific keystroke.

Soft reset ([Alt]-[S])

Esc sequence: $^{E}_{C}g$
Availability: Always, except in file transfer

Soft reset does the following:

- Emits a beep.
- Unlocks the keyboard if it was locked.
- Turns display functions off if it was on.
- Initializes the serial communications ports to the last activated values, and clears the receive buffer.
- Resets typeahead and clears the keyboard buffer.
- Sends an XON if **Receive pacing** is set to *XON/XOFF* (the default).

State save ([Alt]-[B])

Esc sequence: None

Availability: Always, except at a menu or during the execution of a command

State save allows you to exit Reflection and save all settings and parameters related to a host session, a Telnet session, and any host application used within Reflection. These parameters are saved in a PC file that is automatically loaded when you reinitiate Reflection. Using state save rather than a hot-key to return to DOS provides more PC memory for other DOS applications. See page 329 for more information.

Stop key ([Alt]- or [Shift]-[ScrollLock])

Esc sequence: None
Availability: Only if **Receive pacing** has a value of *XON/XOFF*

The Stop (or *Hold*) key alternately stops and starts the process of taking data from the receive buffer and displaying it on the screen. It differs from the [Ctrl]-[Q]/[Ctrl]-[S] combination in that it does not immediately send the XON and XOFF characters to the host. Instead, it depends on the receive pacing mechanism to send the XON/XOFF to keep the receive buffer from overflowing. An **H** appears between the [F4] and [F5] labels when the Hold key is active.

System keys ([F10])

Esc sequence: None
Availability: Always, except when disabled by *EC&jS*

Brings up system keys. [F10] also clears any menu and displays the most recent page of alpha display memory.

When the selection of system, modes, and user keys is disabled, pressing this key causes a beep and the function is not performed.

Toggle print logging ([Ctrl]-[PrtSc])

Esc sequence: None
Availability: In remote mode

Toggles **LOG BOTTOM** on and off and **TO PRINTER** on. Before using this function, confirm that **AUTO LF** is on: press [Alt]-[M] and then press [F8] to display an asterisk.

Press [Ctrl]-[PrtSc]: any subsequent data entered at the keyboard and received from datacomm is routed immediately to the printer. Press [Ctrl]-[PrtSc] again to toggle print logging off.

Toggle user labels ([F9])

Esc sequence: None
Availability: When user keys are enabled

Changes the number of screen label rows to one, zero, or two. If Reflection's 27-line mode is available, using this function toggles the labels on and off instead of changing their height. The effect of this function is temporary; the next time the user keys are displayed, the labels revert to their configured height.

See page 103 for information on configuring the initial height of the user-key labels and presenting them as the startup labels.

User keys ([F9] or [Alt]-[U])

Esc sequence: *EC&jB*
Availability: Always, except when disabled by EC&jS

Brings user keys to the screen. [F9] may then be used to toggle the height of the user-key labels. When the selection of system, modes, and user keys is disabled with *EC&jS*, pressing this key causes a beep and the function is not performed. Use [Ctrl]-[F9] to configure user keys as explained on page 161.

Chapter 23

VT Keyboard Functions

23.1 Quick Reference

Also see *Keyboard Codes* starting on page 578.

Table 13
VT Keyboard Functions

Function	Keystroke
Clear display	Alt - J
Clear line	Alt - K
Clear print buffer	Ctrl - F1
Command line	Alt - Y
Config keys	Alt - C
Connection reset	Ctrl - F8
Define user keys	Ctrl - F9
Delete character	Del
Delete line	Alt - D
Disconnect	Ctrl - F10
Error recap	Alt - E
Exit to DOS	Alt - X
Extend key	Alt - Z
Hard reset	Alt - R
Help	Alt - H

VT Keyboard Functions

Function	Keystroke
Hold screen	ScrollLock (or with Shift)
Home down	Ctrl - End
Home up	Ctrl - Home
Hot key	Alt - (right) Shift
Insert character	Ins
Insert line	Alt - I
Keypad	NumLock
Move cursor down/up	↓/↑
Move cursor left/right	←/→
Modes keys	Alt - M
Next page	Ctrl - PgDn
Next session	Alt - N
PF keys	Alt - P
Previous page	Ctrl - PgUp
Print screen	Alt - PrtSc
Print screen with labels (DOS)	Shift - PrtSc
Reset video	Alt - = *
Roll down	Ctrl - ↑
Roll left	Ctrl - →
Roll right	Ctrl - ←
Roll up	Ctrl - ↓
Scroll display	Alt - ScrollLock
Soft reset	Alt - S
State save	Alt - B *
System keys	F10 or Alt - A
Toggle print logging	Ctrl - PrtSc
User keys	F9 or Alt - U

23.2

Keyboard Functions

This reference describes local keyboard functions in VT mode. Many of these functions are also listed on the VT Help screen (Alt - H). None of the functions described here transmit characters to the host computer; for a description of keystrokes that transmit characters to a host, see page 578.

Clear display (Alt-J)

Availability: Always, except at a menu

Erases display memory from the cursor position through the end of display memory.

Clear line (Alt-K)

Availability: Local and remote mode (at the user key definition menu, File Transfer screen, and the command line)

Erases all characters from the cursor through the end of the line.

Clear print buffer (Ctrl-F1)

Availability: Always

Clears the contents of Reflection's internal printer buffer. The printer continues to print if it has a built-in buffer.

Command line (Alt-Y)

Availability: Always, except at a menu

Causes Reflection to display the command line. Exit the command line by pressing Esc, F10, or by leaving the command line empty and pressing Return.

Config keys (Alt-C)

Availability: Always

Displays the configuration keys.

Connection reset (Ctrl-F8)

Availability: In remote mode

- If using a Hayes Smartmodem 1200B, resets to its power-on state. Sets the *out 1* bit of the modem control register to 1 for at least 50 milliseconds as described in the Hayes manual.
- This is also called CI mode. If using a LAN, suspends the current LAN session.

Define user keys (Ctrl-F9)

Availability: In remote mode

Brings up the user-key menu for you to define user keys. See page 162.

Delete character (Del)

Availability: In local and remote mode (the user-key definition menu, file transfer screen and the command line)

Deletes the character at the cursor position and moves all following characters on the line one position to the left.

When the cursor is to the left of the right margin, only the characters up to the right margin are moved left, and a space is inserted at the right margin.

When the cursor is to the right of the right margin, all characters to the right of the cursor are moved left, and the line is shortened by one column.

Delete line (Alt-D)

Availability: In local mode

Deletes the cursor line and moves all subsequent lines in display memory up one row. The cursor then moves to the left margin.

Disconnect (Ctrl-F10)

Availability: During datacomm, when the PC and the host are connected

Disconnect functions differently depending on the type of host connection you have. If you are connected via a serial port, Reflection lowers the Data Terminal Ready signal (DTR) for 50 milliseconds. This function disconnects a properly configured modem. Note that some modems can be configured to not respond to the disconnect signal. See *Auto-Dial Modems* on page 50.

If you are connected via a LAN, the appropriate LAN disconnect signal is transmitted with this keystroke. See the *PLUS Manual* for more information on network configurations. Also see DISCONNECT and SET DISCONNECT-ON-EXIT in the *Command Language Manual.*

Error recap (Alt-E)

Availability: In remote mode

Displays a menu of errors that can occur in data communication. A count is displayed for each type of error on the current datacomm port. See page 71 for information on error diagnosis.

Exit to DOS (Alt-X)

Availability: Always

- Uninstalls Reflection from memory and returns to the DOS prompt.
- Erases everything in display memory.
- Leaves the PC logged in to the host computer if the connection to the host was active.
- Erases any state save file (R<n>STATE.STA) that is in the Reflection directory or the one specified by SET REFLECT<n>=<filespec>.
- Terminates a currently executing command language program.

The Reflection command language equivalent is HARDEXIT.

When the PC is online over a modem, Reflection does not automatically disconnect. To hang up before exiting Reflection, press Ctrl-F10, Disconnect. See *Auto-Dial Modems* on page 50.

It is recommended that you load Reflection before most PC applications; when you unload Reflection, you free up the maximum amount of memory.

Extend key (Alt-Z)

Availability: Whenever data can be entered

The character typed after the extend sequence matches the keyboard mapping shown on page 178. It also provides access to the Roman 8 character set as shown on page 524.

Hard reset (Alt-R)

Availability: Always, except during file transfer

A hard reset does the following:

- Emits a beep.
- Terminates a currently executing command language program.
- Clears display memory and homes the cursor.
- Sets all terminal configuration parameters to the last activated values.
- Initializes the serial communications ports and clears the receive buffer.
- Clears the keyboard buffer.
- Switches off display functions, insert, and caps lock.
- Sets character sets, color configurations, and selective erase to their default values.
- Sends an XON to the host when the **Receive pacing** parameter has a value of *XON/XOFF*.
- Displays the **pf keys** labels and enters remote mode.

Help (Alt-H)

Availability: In remote mode

Displays the Help screen. Return to the point at which you entered the Help screen with **RESUME** (F8).

Hold screen (ScrollLock)

Availability: In remote mode

Hold screen alternately stops and starts the process of taking data from the receive buffer and displaying it on the screen. It differs from the Ctrl-Q/Ctrl-S combination in that it does not immediately send the XON and XOFF characters to the host. Instead, it depends on the receive pacing mechanism to send the XON/XOFF as they are needed so that the receive buffer does not overflow. An **H** appears between the F4 and F5 labels when hold screen is active. Shift may also be used with ScrollLock to activate this function.

Home up/down (Ctrl-Home and Ctrl-End)

Availability: In remote mode

Home up rolls display memory down so that the first line of display memory occupies the first row of the screen. The cursor moves to the left margin of the top row.

Home down causes display memory to roll up, so the last line in display memory is visible on the screen. The cursor moves to the last position in display memory. When top and bottom margins are specified, or **Multi page** mode is *NO*, Home up and Home down are disabled.

Hot key (Alt- (right) Shift)

Availability: Always, if **Hot key** (Global Configuration) is *not* configured to *NONE*.

Puts Reflection into the background, allowing access to DOS applications. Host operations can continue in the background. This keystroke can be changed; see page 157.

Insert character (Ins)

Availability: Always

Toggles insert character mode on and off. When on, an **I** appears between the F4 and F5 labels. The cursor becomes a blinking block when insert mode is on.

Characters are inserted at the cursor location, and all characters to the right of the cursor move right one column. The character at the right margin is deleted as other characters shift right.

If the cursor is beyond the right margin when the inserted character is typed, the character in column 80 is deleted.

Insert line (Alt-I)

Availability: Always

Inserts a blank line in the screen row that contains the cursor. The old contents of that row and all the following rows in display memory are pushed down one row. The cursor then moves to the left margin of the new (blank) row.

If **Multi page** mode is *YES*, and top and bottom margins are not defined, lines rolled off the screen are pushed down into display memory. Otherwise, lines are deleted.

Keypad (NumLock)

Availability: Always

When NumLock is on, the keypad generates numbers or VT keypad functions. When it's off, the keypad keys control cursor movement.

Modes keys (Alt-M)

Availability: Always

Brings up the modes keys.

Move cursor down (↓, Shift-↓ with NumLock on)

Availability: Always

Moves the alpha cursor down one row on the screen without changing the contents of the screen. If the cursor is already in the bottom row of the screen, it moves to the same column in the top row.

Move cursor left (←, Shift-← with NumLock on)

Availability: When the alpha cursor is visible

Moves the alpha cursor one column to the left, without changing the contents of the screen. If the cursor is already at the left column of the screen, it moves to the rightmost column of the next higher row. If the cursor is at the upper left corner of the screen, it moves to the lower right corner.

Move cursor right (→, Shift-→ with NumLock on)

Availability: When the alpha cursor is visible

Moves the alpha cursor one column to the right, without changing the contents of the screen. If the cursor is already at the right column of the screen, it moves to the leftmost column of the next row. If the cursor is at the lower right corner of the screen, it moves to the upper left corner.

Move cursor up ([↑], [Shift]-[↑] with [NumLock] on)

Availability: Always

Moves the alpha cursor up one row on the screen, without changing the contents of the screen. If the cursor is already in the top row of the screen, it moves to the same column in the bottom row.

Next/Prev page ([Ctrl]-[PgDn], [Ctrl]-[PgUp])

Availability: In remote mode

Next page displays the next page of display memory. A page consists of 24 lines of display memory. If the last line of display memory is on the screen, the operation has no effect.

Prev page displays the previous page of display memory. If the first line of display memory is on the screen, the operation has no effect.

When **Multi page** mode is *NO*, or top and bottom margins are specified, Next page and Prev page are disabled.

Next session

Availability: CI Mode

For the multi-session LAT interface, moves between active sessions.

PF keys ([Alt]-[P])

Availability: Always

Brings up the PF key labels.

Print screen ([Alt]-[PrtSc])

Availability: Always

Prints a copy of the currently displayed screen, not including the function key labels, and sends a form feed to the printer.

Print screen with labels ([Shift]-[PrtSc])

Availability: Always

Prints a copy of the currently displayed screen (alpha only), including the function key labels. Only 25 lines are printed. This is a DOS function, not a Reflection function.

Reset video (Alt-=)

Availability: Always, except when the command line or configuration screen is displayed.

This keystroke is available in Reflection 1 only. Some PC applications can alter video effects such as font and palette. Use Alt-= to reinitialize the video card and redisplay the screen. * * *

Roll screen left/right (Ctrl-→/←)

Availability: In remote mode

When the right margin is set beyond column 80, rolling left brings text from beyond the right border of the screen into view. Each instance shifts a specified number of columns off the left side of the screen, and the same number of columns onto the right side. The number of columns scrolled is specified in the **Columns per horizontal scroll** field of the Terminal Configuration, Page 2 menu.

When the right margin column is visible on the screen, rolling left has no effect.

When the right margin is set beyond column 80, rolling right brings text from beyond the left border of the screen into view. Each instance shifts a specified number of columns off the right side of the screen and the same number of columns onto the left. When column 1 is on the screen, rolling right has no effect.

Roll screen up/down (Ctrl-↑/↓)

Rolling down moves the text in display memory down one line and brings the previous line of text onto the top row of the screen. When the first line of display memory is in the top row of the screen, rolling down has no effect.

Rolling up moves the text in display memory up one line and brings the next line of text onto the bottom row of the screen. When the last line of display memory is in the top row of the screen, rolling up has no effect.

Rolling up/down is disabled if **Multi page** mode is *NO* or top or bottom margins have been set.

Scroll display (Alt-ScrollLock)

Availability: Always

Reflection allows you to scroll display memory without using Ctrl with the arrow keys. When you press Alt-ScrollLock, an S appears between the function key labels. When you press the cursor keys in this mode, display memory lines are scrolled while your cursor position on the screen remains unchanged. If you are using VAX/VMS applications, this function may create problems as applications typically control movement of the cursor.

Soft reset (Alt-S)

Availability: Always, except in file transfer

A soft reset, also called *clear communications,* does the following:

- Emits a beep.
- Initializes the serial communications ports to their last activated values and clears the receive buffer.
- Clears the keyboard buffer.
- Sends an XON to the host if the **Receive pacing** configuration parameter is set to *XON/XOFF*.

State save (Alt-B)

Availability: Always, except at a menu or during the execution of a command

State save allows you to exit Reflection while saving all settings and parameters related to a host session, a Telnet session, and any host application used within Reflection. These parameters are saved in a PC file that is automatically loaded when you reinitiate Reflection. Using state save rather than a hot-key to return to DOS provides more PC memory for other DOS applications. See page 329 for more information.

System keys (F10)

Availability: Always

Brings up the system keys. F10 also clears any menu and displays the most recent page of display memory.

Toggle print logging (Ctrl-PrtSc)

Availability: In remote mode

Toggles **LOG BOTTOM** on and off and **TO PRINTER** on. Before using this function, confirm that **AUTO LF** is on: press Alt-M and then press F8 to display an asterisk. *

Press Ctrl-PrtSc: any subsequent data entered at the keyboard and received from datacomm is routed immediately to the printer. Press Ctrl-PrtSc again to toggle print logging off.

User keys (F9 or Alt-U)

Availability: Always

Brings user keys to the screen. F9 may then be used to toggle the height of user-key labels. The effect of this function is temporary; the next time the user keys are displayed, the labels revert to the configured number of lines.

The initial height of the user key labels and the ability to present them as the startup labels can be configured. See the VT Terminal Configuration, Page 1 menu on page 121.

Section 6

Keyboard Remapping

Each Reflection or terminal keyboard function—such as "bring up the system keys", or "go to the command line"—has a default keystroke. Remapping the keyboard so that these functions are associated with different keystrokes is entirely optional. Remapping lets you re-assign keyboard functions to new keystrokes. It also lets you assign a combination of keyboard functions, strings of characters, escape sequences, and ASCII control codes to a given keystroke. Up to 127 characters or functions may be assigned to a single keystroke. Remapping offers a great deal of flexibility:

- Take advantage of keyboards with non-standard layouts and features
- Attach commonly-used strings of text and control codes to a keystroke
- Move a Reflection function to another keystroke
- Disable a Reflection function, such as hard reset or exit

Keyboard mapping information is stored in Reflection configuration files. You can store different keyboard maps in various configuration files, then load the appropriate configuration when you change host computers or applications.

This section is organized as follows:

Introduction

This introduction is an overview of how to remap your keyboard, including a simplified diagram.

Quick Reference

The quick reference chapter is for users familiar with keyboard remapping. It shows the rules for creating a mapping definition file and the complete syntax for keystroke definitions, along with page references. Beginners can skip this chapter.

Default Mapping

Your program disks includes mapping files — the most important one is DEFAULT.KBM. This chapter explains how to use this file as a model for your own mapping file and how to compile the mapping.

Keyboard-ID and Initial State

This chapter lists the possible values for the first two equations in a keyboard mapping file (**keyboard-ID** and **terminal emulation**) and describes how to set the initial state of NumLock and CapsLock.

Keystroke Definition Syntax

An explanation for each element in a keystroke definition is presented here.

Compiling the Mapping

The final step in remapping your keyboard is compiling the new keystroke definitions into a configuration file. This chapter also explains how to reset Reflection to its default mapping.

Keyboard Layouts

This chapter contains the keyboard layout for each of the keyboard ID choices. Each key has a name: this is the name that must be used in the keystroke definitions.

Alternate Methods of Defining Keystrokes

If you have an unusual keyboard, you may need to define a keystroke in terms of the scan code it generates. This chapter explains how to do so.

Chapter 24

Introduction

24.1 Introduction to Keyboard Remapping

You can use remapping to assign strings or functions to any keystroke, or use it to move Reflection functions from the default keystroke to one of your choosing. For VT emulation, you may want to use a supplied remapping file to make your keyboard operate more like a VT terminal. The following example and figure give an overview of how to assign a Reflection function to a keystroke.

In Reflection, you can press either [F10] or [Alt]-[A] to get to the system keys. In this example, [Alt]-[A] is remapped so that it displays the File Transfer screen instead. (There is no default keystroke for bringing up this screen.)

Create a keyboard mapping file with a text editor.

1. Identify the kind of keyboard you are using
2. Identify the terminal emulation (HP or VT)
3. Enter one or more keystroke definitions
4. Compile the information into a Reflection configuration file with KEYMAP.EXE

The figure below summarizes this process.

1

Using an ordinary text editor, create a new file called DEMO.KBM

2

The first line of the file is your keyboard-ID. For example:

```
keyboard-id = enhanced
```

The second line of your file is your terminal emulation. For example:

```
term = HP
```

3

Enter any number of keystroke definitions. The simplest keystroke definitions follow this pattern:

Keystroke = String

```
alt shift g = "Good morning"
```

Keystroke = Function

```
alt lshift a = clear-display
```

A keystroke definition can combine strings and functions:

```
alt tab = command-line "dir^M"
```

For example, let's define the keystroke "alt a" to bring up the Reflection File Transfer screen:

```
keyboard-id = enhanced
term = HP
alt a = file-xfer-screen
;"file-xfer-screen" is a
;Reflection function name.
;Comments (like this one)
;are preceded by a semicolon
```

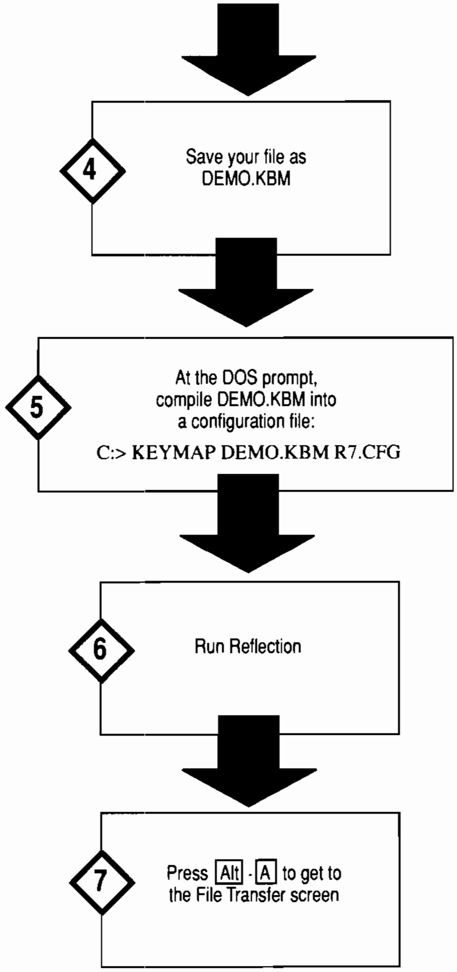


Figure 32: The Keyboard Mapping Process

Chapter 25

Quick Reference

Source files for keyboard mapping definitions consist of these basic elements:

Keyboard-ID (required)	You must define the keyboard you will be using when you run Reflection. See page 233.
Terminal emulation (required)	You must indicate the terminal emulation to be used— HP or VT.
Initial state of CapsLock and NumLock	You can opt to set the initial state of these keys. See page 235.
Keystroke definition syntax	See page 237.
Comments	Comments can be at the end of a line or on a separate line. (This is different from Reflection's command language.)

The complete syntax for each entry in a mapping definition file follows:

Keyboard-ID

```
KEYBOARD-ID = {PC | AT | ENHANCED | HP-VECTRA | NONENH}
```

This is the first entry in the mapping file; it is required.

Terminal emulation

```
TERM = {HP | VT}
```

Setting the terminal emulation is required. It must immediately follow the keyboard-ID.

Initial state

```
SET {NUM-LOCK | CAPS-LOCK} {ON | OFF}
```

Setting the initial state of NumLock and/or CapsLock is optional. If it is done, the set statements must immediately follow the terminal emulation.

Keystroke definition

```
<input specification> = <output specification> [;<comments>]
```

<input specification>

```
{<keyspec> | <int16spec> | <int9spec>}

  <keyspec>    : [<shiftname>] <keyname>
  [<shiftname>]: {SHIFT | ALT | CTRL | LSHIFT | RSHIFT | IGNORE_SHIFTS}
  <keyname>    : {F1 | F2 | ... A | B | ....HOME | NUM-LOCK | ...}
  <int16spec>  : [<shiftname>] <number> <number>
  <number>     : {0 | 1 | 2 | 3 ... 255}
  <int9spec>   : [<shiftname>] <number> [HIDE] [E0-PREFIXED] [E1-CTRL-PREFIXED]
```

<output specification>

```
{NULL | [<string> | <keybd function> | <number>]}

  <string>    : "<charspec>"
  "<charspec>"    : "{<charname> | <ctrl charname>}"
  <charname>    : {a | b | ... Y | Z | ... 1 | 2 | 3 | ... [ | \ | ] | ^ | _}
  <ctrl charname> : ^<charname>

  <keybd function>: {HOST-BREAK | SOFT-RESET | VT-PF1 | ...}

  <number>    : {0 | 1 | 2 | 3 ... 255}

  [;<comments>]   : ;<can be any text preceded by a semicolon>
```

25.1 Sample Keystroke Definitions

In the following examples for an Enhanced keyboard, *shift states* are described by **alt, ctrl, alt lshift,** and **shift ctrl.** The *key names* are **cp-right, home, h, x,** and **end,** and **z.** The desired result of the key mapping appears after the equal sign.

```
keyboard id = enhanced

term = hp

alt cp-right = command-line "display '^[&j@' ^M^M"
alt home = disconnect
ctrl h = "HELLO <logon> ^M"
alt x = null
shift ctrl end = reset-typeahead
alt lshift z = 72 105 33 13
```

Once compiled into a configuration file, which is then loaded with Reflection, these keystrokes would have the following results:

- The keystroke Alt-→ (cursor key) goes to the Reflection command line and uses an escape sequence to activate the user key labels.
- Alt-Home sends a disconnect signal.
- Ctrl-H does a logon, followed by return.
- The keystroke to exit Reflection—Alt-X—is disabled: pressing it has no effect.
- Shift-Ctrl-End resets typeahead as described for HP emulation.
- Pressing Alt-(left) Shift-z displays **Hi!** on the screen and sends a carriage return.

Chapter 26

Default Mapping

If you would like to change the way your keyboard is mapped, you can use the supplied mapping files as a model.

26.1 Sample Files

DEFAULT.KBM (on one of the support disks) lists what the default keystrokes and function names are for Reflection. It is not meant to be compiled: it is for documentation only. It shows how your keyboard behaves if you do not make use of the keyboard mapping feature. EXAMPLES.KBM shows examples for an Enhanced keyboard.

You can copy portions of files to produce new mapping files. Be sure to define your keyboard and terminal type before compiling. These files can be viewed in any PC text editor. The .KBM extension identifies keyboard files. If you create your own files, this extension is recommended.

For example, you may want to remap a new Enhanced keyboard so the Enter function is available with the keypad [+] key, the same location as your previous PC. The example below assigns the Enter function to the keypad [+] key.

Default Mapping

Table 14
Sample Mapping File

```
keyboard-id = enhanced
term = hp

kp-plus  = enter
```

The Enhanced keyboard default for the Enter function, Enter, will still be in effect— after compiling this file, you will have *two* keystrokes that perform the Enter function. Your keyboard mapping is a combination of the default mapping plus any additional assignments in your own keyboard remapping file.

26.2 VT Remapping Files

You can remap the PC keyboard to operate more like a VT terminal. The following is from the Enhanced keyboard mapping file, VT_ENH1.KBM:

```
keyboard-id = ENHANCED
term = VT

num-lock = vt-pf1
kp-slash = vt-pf2
kp-star = vt-pf3
kp-minus = vt-pf4

kp-plus = vt-minus
shift kp-plus = vt-comma
```

The VT's PF1 function key is mapped to the NumLock key. PF2 is mapped to / (slash on the keypad), and so on.

Additional keyboard remapping files (VT_EN700.KBM, VT_AT700.KBM, VT_ENH1.KBM, and VT_AT1.KBM) are supplied for users of Oracle applications. See page 313 for more information.

Chapter 27

Keyboard-ID and Initial State

The first statement of a keyboard mapping definition source file must be the keyboard-ID statement. This statement defines the keyboard on which the mapping will be used.

Each of the following is a valid keyboard-ID statement:

```
Keyboard-ID = PC          ; IBM PC or PC/XT-type keyboard
Keyboard-ID = AT          ; older IBM AT
Keyboard-ID = Enhanced    ; new IBM AT, PS/2, XT-286, Vectra ES
Keyboard-ID = HP-VECTRA   ; older Hewlett-Packard Vectra (non-Enhanced)
Keyboard-ID = NONENH      ; Enhanced keyboard with non-Enhanced BIOS
```

If your keyboard is not among these, choose the option that your keyboard most closely resembles.

27.1 PC

This is the original PC keyboard, supplied with the IBM PC and PC/XT. The name for each key is shown in the figure on page 254.

27.2 AT

This keyboard is nearly identical in appearance and function to the PC keyboard. A few keys are moved to different positions, including [Esc], [`], and [PrtSc] (kp-star). [SysReq] was added. The name for each key is shown in the figure on page 254.

27.3 ENHANCED

The Enhanced keyboard differs quite a bit from the keyboards that preceded it. New keys include F11 and F12, Enter, / (kp-slash), and a set of ten cursor control keys. These new keys can be distinguished from the keys on the keypad by PC applications. The name for each key is shown in the figure on page 255.

27.4 HP-VECTRA

The keyboard-ID *HP-VECTRA* refers to early HP Vectra keyboards. If you are using the newer HP-Vectra keyboard (similar to IBM's Enhanced keyboard), use **Keyboard-ID = Enhanced.** See the figure of the Enhanced keyboard on page 255 for the name of each key.

This keyboard is very similar to the IBM AT keyboard. HP added a duplicate set of the keys F1 through F8 across the top of the keyboard for users accustomed to HP terminals. An auxiliary set of eleven cursor control keys was also added. Intelligence built into the keyboard attempts to ensure that the cursor and control keys are never interpreted by the PC to generate numeric characters, regardless of the state of NumLock and Shift. The name for each key is shown in the figure on page 255.

27.5 NONENH—Non-Enhanced Keyboard

Use this keyboard-ID if your keyboard is Enhanced, but your PC is equipped with a non-Enhanced keyboard BIOS. Reflection then interprets keystrokes at the hardware interrupt level. Use KEYMON to find out about the BIOS you're using. KEYMON is a program included on your diskettes for diagnosing keyboard problems and investigating the function of your PC's keyboard and BIOS. Type **KEYMON** at the DOS prompt. If you have a non-Enhanced BIOS, you should see the following:

```
PC keyboard BIOS is standard
```

Use **Keyboard-ID = NONENH** in your mapping definition file. For key names, refer to the figure of the Enhanced keyboard on page 255.

27.6 Terminal Type

You must indicate in the remapping file the terminal emulation to be used: HP or VT. Use *one* of the following lines to indicate the emulation:

```
term = VT
```

or

```
term = HP
```

When the keyboard information is compiled into a Reflection configuration file, only the emulation specified is affected. If you want to store special mapping information for both emulations in a single configuration file, you must compile two different keyboard mapping files —one with **term = HP**, and one with **term = VT**—into the same configuration file.

27.7 Initial State of NumLock and CapsLock

The initial state of NumLock and CapsLock can be defined within a keyboard mapping definition file via set statements. It is often convenient to have NumLock on when doing VT emulation, since in this mode the numeric keypad keys emulate the VT200 series application keypad keys. Users may then hold down a Shift key while striking the arrow keys on the keypad to position the cursor (holding down a shift key cancels the effect of NumLock or CapsLock).

These optional set statements must follow the keyboard-ID statement and precede any keystroke definitions. For example, if you are a PC keyboard user and you wish to have NumLock forced on and CapsLock forced off, you would put the following in your keyboard mapping definition file:*

```
keyboard-ID = PC

term = HP

set num-lock  on
set caps-lock off
```

* These set statements are only valid in a mapping definition file; they are not part of Reflection command language.

Keyboard-ID and Initial State

Keystroke definitions may or may not follow these statements—they too are optional. Compile the mapping definition into a configuration file (as explained on page 251). Initialization of NumLock and CapsLock occurs whenever that particular configuration file is loaded.

Chapter 28

Keystroke Definition Syntax

A keystroke definition equates a keystroke with a task. The left side of the equation is the keystroke, and the right side is the action to be taken in response to that keystroke. Each keystroke definition has these possible elements:

- The keystroke being defined, including shift keys
- A required equals sign (=)
- The characters, functions, or control codes that should result when the keystroke is performed
- Any comments regarding the mapping (optional)

The syntax for a keystroke definition statement follows:

```
<keystroke specification> = <output specification>
```

Keystroke specifications are described below; output specifications are covered on page 241.

28.1 <Keystroke Specification>

The keystroke specification includes both the key name for the key that is pressed and the shift specification for any shift keys that are held down while the key is pressed.

The key name for Home, for example, is **home**. Using **home** alone specifies only a key name, while **alt home** specifies both a shift specification (**alt**) and a key name.

Keystroke Definition Syntax

Keystrokes are defined in terms of names given to each of the physical keys on the keyboard. In most cases these names are identical to the symbols found on the keys of U.S. versions of the keyboards. In some cases the name varies slightly from the symbol, or describes the function of the key. Examples:

- The name of the key with the symbol **F1** is *f1*.
- The name of the space bar (on which no symbol appears) is *space*.
- The name of the key that has both **5** and **%** is *five*.

The keyboard mapping definition compiler is only case-sensitive in the case of alphabetic key names and within a defined string. Keyboard diagrams beginning on page 253 show the names of the keys on the supported keyboard types.

Shift Keys

A keystroke may include one or more *shift* keys, held down at the time the key is struck. A shift key is any key that may be used in conjunction with another key to alter the key's output. [Alt] and [Ctrl] are commonly used shift keys in addition to [Shift].

In order to take full advantage of your keyboard and preserve as many existing Reflection keystrokes as possible, use various shift states to distinguish keystrokes. *Shift state* is defined by these key names:

Table 15
Shift State Options

shift	[Shift] (left, right, or both)	ctrl	[Ctrl]
lshift	[Shift] (left)	alt	[Alt]
rshift	[Shift] (right)		

Note that the left and right [Shift] can be distinguished from one another. If **shift** is used alone, either [Shift] key key works. If both are specified, both must be pressed with the defined key.

Almost any combination of the above shift key names can be used when defining a keystroke. The only restriction is that a particular shift state cannot be used twice: **alt alt** is not valid. Unless a shift state is defined, an unshifted state is assumed. Some examples follow:

```
alt rshift t = command-line ;press [Alt]-(right)[Shift] and [t] to
                            ;bring up Reflection's command line
ctrl shift g = hard-reset   ;press [Ctrl]-[Shift] (either left or
                            ;right) and [g] to do a hard reset
```

Keystroke Restrictions

Alt and **ctrl** are valid shift-state options for any key on the keyboard. **Lshift, rshift,** and **shift**, on the other hand, are not valid shift state options for the eleven keys of the numeric keypad, the cursor keys of the Enhanced keyboard, or the alphabetic keys. Different rules apply for these keys because of the way [NumLock] and [CapsLock] affect their shift state.

[CapsLock]

You must use the key name that correctly describes the key in either its shifted or unshifted state. For example, the name used to describe the letter G key when it is not shifted is **g**. When using the [G] key, use an uppercase **G** to indicate it is shifted in some way (either [CapsLock] is on or one or both of the [Shift] keys is down).

[NumLock]

Each of the eleven keys of the numeric keypad has also been assigned two names because the state of [NumLock] as well as [Shift] must be considered when deciding whether these keys are numeric or cursor/display control functions. Depending on the state of [NumLock], holding down [Shift] and striking [Home] may result in either of two quite different outputs. Use **kp-7** to describe the keystroke that generates the number 7, and use **home** to describe the keystroke that homes the cursor.

For keypad numerics use the following key names:

kp-1	kp-2	kp-3	kp-4
kp-5	kp-6	kp-7	kp-period
kp-8	kp-9	kp-0	

Use **home, up, pgup** and so on for these keys if the keystroke you are mapping includes the **alt** or **ctrl** shift keys. If you have an Enhanced keyboard, the cursor keypad keys may be mapped separately. The names for those keys follow:

cp-ins	cp-del	cp-up	cp-down
cp-home	cp-end	cp-left	cp-right
cp-pgup	cp-pgdn		

Because of the way the Enhanced keyboard handles the auxiliary keypad, the above restriction concerning the shift keys also applies to the cursor keypad. Intelligence built into the keyboard itself ensures that these keys never appear to be shifted, even if you hold one or both Shift keys down. **Shift, lshift,** and **rshift** may not be used with these key names. The same applies to the **kp-slash** key on the Enhanced keyboard.

28.2 Keyboard Remapping in Reflection 7

Some keystrokes cannot be remapped when running Reflection 7 in graphics mode.

- Graphics functions cannot be overridden or assigned to other keystrokes under keyboard remapping.
- Because Alt and keypad keys are used for special graphics functions, you cannot use **alt** as a shift key name with the following keys:

  Table 16
  Restricted Keypad Remapping

home	end
down	up
pgdn	pgup
left	right
center	

- The keystroke used to toggle on the graphics cursor, Ctrl-- (keypad) cannot be overridden or remapped.
- When the graphics keypad is active, the keypad keys can only be used to move the graphics cursor.

  The following keynames, when used without a shift key, are ignored when the graphics keypad has been toggled on (G visible):

Key name	Alternate Key name	Key name	Alternate Key name
kp-0	ins	kp-5	center
kp-1	end	kp-6	right

kp-2	down	kp-7	home
kp-3	pgdn	kp-8	up
kp-4	left	kp-9	pgup

28.3

<Output Specification>

The output specification defines the characters to be generated and/or the functions to be performed in response to a given keystroke. It can either be **NULL**, or a combination of the following:

```
{<string> | <number> | <keyboard function>}
```

The following is a brief description of each of these parameters, along with page references for more detailed information:

NULL — Disables a keystroke. See below.

<String> — A string is enclosed in quotes and can include control characters and/or control functions: "{a, ... Z, ... 0, ... ^m, ...}". See page 242.

<Number> — Defines a character in terms of its numeric value: {0, 1, 2, ... 254, 255}. See page 243.

<Keyboard function> — Each Reflection and terminal keyboard function has a name. For example, "command-line" is a Reflection function name.

See page 243 for a description, and page 244 for a table of function names and the keystrokes they are mapped to by default.

28.4

NULL

If a keystroke is defined as null, it is completely disabled. For instance, Reflection can be exited with the keystroke [Alt]-[X]. If [Alt]-[X] is remapped to null, the keystroke has no effect:

```
alt x = NULL
```

28.5

<String>

A string is a series of characters enclosed in quotes, as in the following:

```
"This is a string enclosed in quotes."
```

A string can also contain two-character sequences that represent ASCII control characters in the range 0 through 31.* See *The ASCII Character Set* on page 519 for the list of two-character sequences. The following keystroke definition means that pressing Alt-F2 will be equivalent to typing **SHOWME** and pressing Return:

```
alt f2 = "SHOWME ^M"
```

In the following example an escape sequence is assigned to a keystroke via a string in a Reflection command:

```
alt lshift s = command-line "display '^[&r5U' ^M^M";roll display up 5 lines
alt lshift j = command-line "display '^[&r5D' ^M^M";roll display down 5 lines
```

String Restrictions

Three ASCII characters have special meaning when defining strings.

Character	ASCII Code	Description
Quotes (" ")	34	String delimiter
Caret (^)	94	First character of an ASCII control code
Backslash (\)	92	Default literal escape character

When a backslash, caret, or quote mark is preceded by a backslash— the literal escape character—its ASCII value rather than its special meaning is indicated.

To make the keystroke Ctrl-4 display **This is a caret: ^**, enter the following keystroke definition:

```
ctrl four = "This is a caret: \^"
```

* The delete character cannot be expressed in terms of a two-character sequence; it must be expressed as a <number>, 127.

Concatenating a String

When you are defining a long string in your keyboard mapping file (.KBM), you may concatenate it so that the entire string is visible on the screen at one time. Example:

```
alt f1 = "This string appears on your " &
         "screen as one line. To dis" &
         "play it in YOUR.KBM in this" &
         " format, use ampersands (&).^M"
```

There are a few rules for concatenating strings:

- No more than 3 ampersands can be used for one string.
- A comment line (a semicolon followed by some text) cannot appear on a separate line within the concatenated string. Comments at the end of a line are legal.

28.6

<Number>

Defines a character in terms of its numeric value. Values in the range 0 through 127 conform to the ASCII character set. Values in the range 128 through 255 represent characters in the IBM extended font. For example, the mapping of the keystroke Alt-R to the string "Reflection" could be accomplished in either of these ways:

```
alt r = "Reflection"
alt r = 82 101 102 108 101 99 116 105 111 110
```

ASCII control codes in the range 0–31, including carriage return, horizontal tab, form feed, etc., can be specified either by their control code equivalents (as explained on page 242, under **<String>**), or by their decimal values. To append a carriage return to the example above, for instance, add **13** to the output specification.

28.7

<Keyboard Functions>

Keyboard functions include both Reflection keyboard functions and HP/VT terminal functions. See page 244 for a listing of the functions available for mapping, along with the keystrokes to which they are mapped by default.

An output specification can include any combination of up to 127 function names and characters. For example, the following is a valid mapping:

```
alt comma = command-line "set remote-mode no^M^M^" home-up clear-display
```

Once this mapping is compiled into a configuration file, pressing Alt-, brings up the Reflection command line, puts the PC into local mode, does two carriage returns (one to execute the command, and one to exit the command line), homes the cursor, and clears the display.

The Reflection keyboard functions can be mapped to alternate keystrokes. See the list of default mappings on page 244 to find the appropriate name and spelling for all functions. Keyboard function names of two or more words (such as *clear-display*) must be hyphenated as shown.

Special Function Restrictions

Reset Functions

Reset functions can only be mapped to function keys F1 to F10 or alphabetic keys shifted with Ctrl, Alt, or both. This includes **soft-reset, hard-reset, communications-reset,** and **exit-to-dos.**

NumLock & CapsLock

These keys are mappable keys; their functions, however, cannot be mapped to other keystrokes.

Function Names and Default Mapping

The following tables should be consulted whenever you plan to remap keystrokes. The first column is a description of the function. The second column shows the keystroke definition as it applies to all keyboards. Where there is a difference—for instance, on an Enhanced keyboard—the definition is listed in the third column. Note that mapping a function to a predefined key overrides the original function for that keystroke, and there may be no other means of accomplishing the same task. By default, for instance, the keystroke Ctrl-F8 does a connection reset. If you assign the function main-menu to it **(ctrl f8 = main-menu)**, you no longer have a keystroke that resets your connection.

Table 17
Default Reflection Keystroke Definitions

Terminal Function	Default Keystroke Definition	
Break	ctrl scroll-lock = host-break	
Clear display	alt j = clear-display	
Clear line	alt k = clear-line	
Clear print buffer	ctrl f1 = clear-print-buffer	
Command line	alt y = command-line	
Config keys	alt c = config-keys	
Connection reset	ctrl f8 = connection-reset	
Delete line	alt d = delete-line	
Disconnect	ctrl f10 = disconnect	
Error recap menu	alt e = error-recap	
Exit to DOS	alt x = exit-to-dos	
Extend key	alt z = extend-key	
File transfer screen	(no default keystroke) = file-xfer-screen	
Hard reset	alt r = hard-reset	
Help screen	alt h = help-screen	
Insert line	alt i = insert-line	
Modes keys	alt m = modes-keys	
Print screen	alt kp-star = print-screen	
Reset video	alt equals = reset-video *	*
Return	return = return	
Soft reset	alt s = soft-reset	
State save	alt b = exit-with-save	*
System keys	alt a = system-keys	
System keys	f10 = system-keys	
Toggle print logging	ctrl kp-star = toggle-print-logging	
Toggle user keys	f9 = toggle-user-keys	
User-key labels	alt u = user-keys	
User-key labels	f9 = user-keys	

* Available for Reflection 1 only

Keystroke Definition Syntax

Terminal Function	Default Keystroke Definition	
		keyboard-id = enhanced
Delete character	del = delete-char	cp-del = delete-char
Enter	kp-plus = enter	kp-enter = enter
Home down	ctrl end = home-down	cp-end = home-down
Home up	ctrl home = home-up	cp-home = home-up
Insert char	ins = insert-char	cp-ins = insert-char
Move down	down = move-down	cp-down = move-down
Move left	left = move-left	cp-left = move-left
Move right	right = move-right	cp-right = move-right
Move up	up = move-up	cp-up = move-up
Roll down	ctrl up = roll-down	cp-up = roll-down
Roll left	ctrl right = roll-left	cp-right = roll-left
Roll right	ctrl left = roll-right	cp-left = roll-right
Roll up	ctrl down = roll-up	cp-down = roll-up

Table 18
HP-Only Default Mapping - IBM PC

Terminal Function	Default Keystroke Definition	
Beginning of line	home = home-left	
Clear keyboard buffer	alt v = clear-keyboard-buffer	*
Delete wrap	ctrl delete = delete-wrap	
Delete wrap	cp-ctrl delete = delete-wrap	
Function key recall	alt f1 = function-key-recall	
Insert wrap	ctrl insert = insert-wrap	
Insert wrap	cp-ctrl insert = insert-wrap	
Move to top	ctrl pgdn = move-to-top	
Move to bottom	ctrl pgup = move-to-bottom	
Reset handshake	alt q = reset-handshake	
Reset typeahead	alt t = reset-typeahead	
Scroll display	scroll-lock = toggle-arrow-functions	
Select key	(no default keystroke) = select-key	
Start of line	home = home-left	
Stop *	alt scroll-lock = hold-screen	
Stop	shift scroll-lock = hold-screen	
Tab key (configurable)	(no default keystroke) = tab-key	
		keyboard-id = enhanced
Backtab	shift tab = backtab	cp-home = home-left
End of line	end = home-right	cp-end = home-right
Next page	pgdn = next-page	cp-pgdn = next-page
Prev page	pgup= prev-page	cp-pgup = prev-page

* Receive pacing must be set to XON/XOFF for this to function as the HP Stop key. See page 173.

Table 19
VT102-Only Default Mapping - IBM PC

Terminal Function	Default Keystroke Definition
Backspace	ctrl backspace = "^h"
CI mode*	ctrl f8 = ci-mode
Delete character	backspace = 127
Hold screen	scroll-lock = hold-screen
Hold screen	shift scroll-lock = hold-screen
Next session†	no default keystroke = next-session
PF key labels‡	alt p = pf-key-labels
Scroll display	alt scroll-lock = toggle-arrow-functions
Send answerback	ctrl f2 = send-answerback
VT 0	kp-0 = vt-0
VT 1	kp-1 = vt-1
VT 2	kp-2 = vt-2
VT 3	kp-3 = vt-3
VT 4	kp-4 = vt-4
VT 5	kp-5 = vt-5
VT 6	kp-6 = vt-6
VT 7	kp-7 = vt-7
VT 8	kp-8 = vt-8
VT 9	kp-9 = vt-9
VT comma	kp-star = vt-comma
VT enter	kp-plus = vt-enter
VT minus	kp-minus = vt-minus
VT period	kp-period = vt-period
VT PF1 (GOLD)‡	f1 = vt-pf1
VT PF2 ‡	f2 = vt-pf2
VT PF3 ‡	f3 = vt-pf3
VT PF4 ‡	f4 = vt-pf4

* LAT and CTERM Command Interpreter only.

† Only has meaning when Walker Richer & Quinn's TelnetManager, LAT, or CTERM Command Interpreter is in use.

‡ The PF key labels must be active for f1 through f4 to equal vt-pf1 through vt-pf4.

The following keystrokes are only active when VT220 is selected as the terminal ID.

Table 20
VT220 Keypad

Terminal Function	Default Keystroke Definition
Find	`alt one = find`
Insert here	`alt two = insert-here`
Next screen	`alt six = next-screen`
Previous screen	`alt five = prev-screen`
Remove	`alt three = remove`
Select	`alt four = select`
VT F6	`shift f6 = vt-f6`
VT F7	`shift f7 = vt-f7`
VT F8	`shift f8 = vt-f8`
VT F9	`shift f9 = vt-f9`
VT F10	`shift f10 = vt-f10`
VT F11	`alt f1 = vt-f11`
VT F12	`alt f2 = vt-f12`
VT F13	`alt f3 = vt-f13`
VT F14	`alt f4 = vt-f14`
VT F15 (help)	`alt f5 = vt-f15`
VT F16 (do)	`alt f6 = vt-f16`
VT F17	`alt f7 = vt-f17`
VT F18	`alt f8 = vt-f18`
VT F19	`alt f9 = vt-f19`
VT F20	`alt f10 = vt-f20`

[illegible]
[illegible]

Chapter 29

Compiling the Mapping

Once you have created (or selected) a mapping file, you have to compile the information it contains into a Reflection configuration file.

First select a configuration file in which to store the keyboard information.

R1.CFG and R7.CFG are reserved as Reflection's default configuration filenames. Since it is loaded automatically if no other configuration file is named, use another name initially.

29.1 KEYMAP Syntax

KEYMAP is a stand-alone utility program that compiles keyboard definition statements into a format that Reflection can interpret. It then inserts this information into the named configuration file. The program is run at the DOS prompt. The syntax is as follows:

```
C:> KEYMAP <keyboard mapping filename> <config filename>
```

Example:

```
C:> KEYMAP MYKEYS.KBM MYNEW.CFG
```

Once compiled, run Reflection using this configuration file. For example, to load Reflection 1 with **MYNEW.CFG**, enter the following at the DOS prompt:

```
C:> R1 MYNEW.CFG
```

Any keys not specifically remapped retain their default Reflection values. Reflection picks up keyboard mapping information from the configuration file whenever Reflection is started or a configuration file is loaded (using the Reflection LOAD command).

29.2 Resetting Default Values

If you used R1.CFG or R7.CFG as your configuration filename and want to reset all of Reflection's original values, including keyboard mapping defaults, you must reload Reflection with the **/D** switch. Alternatively, you may delete R1.CFG (or R7.CFG) altogether and then reload Reflection. Resave the set-up under the default filename.

If you run KEYMAP and give it the name of a new configuration file—a file that contains no configuration information—KEYMAP creates a file that contains mapping information only. You have two choices:

- Run Reflection from DOS with the compiled file. This gives you the default configuration, plus whatever mapping you compiled into your file.

  ```
  C:> R1 REMAP.CFG
  ```

- Load the compiled file from within Reflection. This gives you the *current* configuration of Reflection, plus the mapping in REMAP.CFG. Save the configuration before you leave Reflection. Both of these commands are typed at the Reflection command line, one at a time.

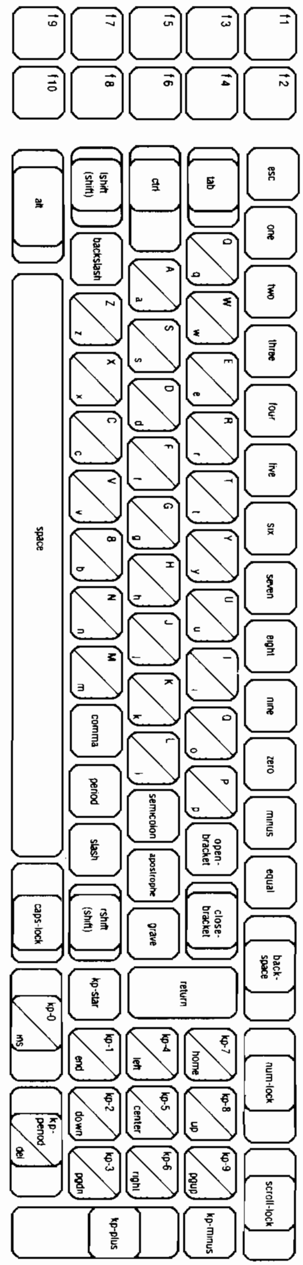
  ```
  LOAD REMAP.CFG
  SAVE <current config file>
  ```

  When you save a configuration file, the configuration for the alternate emulation (HP or VT) is saved with default values. If you want to store special mapping information for both emulations in a single configuration file, you must compile two different keyboard mapping files —one with **term** = **HP**, and one with **term** = **VT**—into the same configuration file and save the configuration file under both emulations.

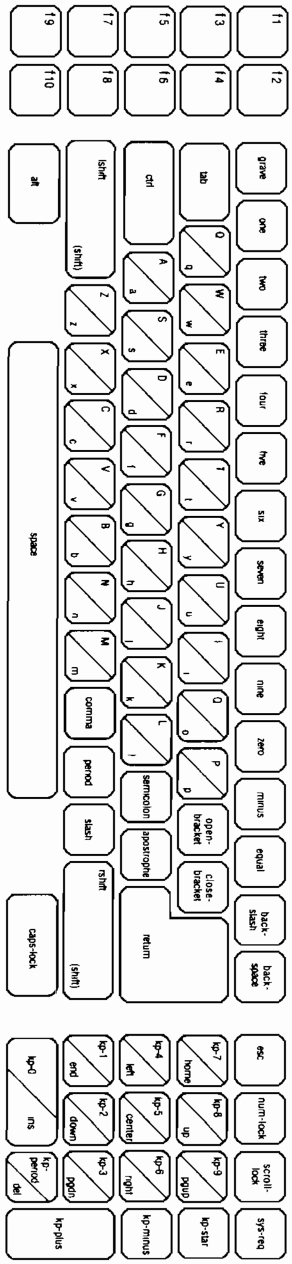
Chapter 30

Keyboard Layouts

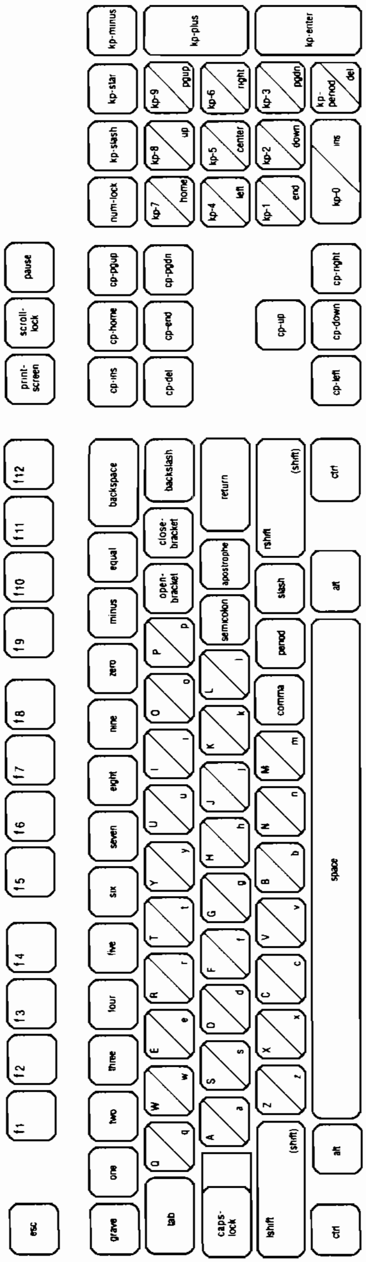
IBM PC or XT Key Names



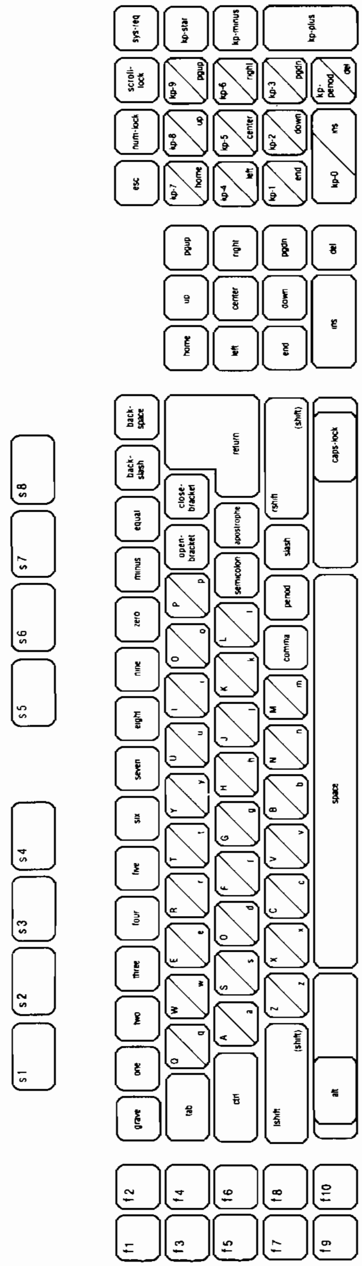
IBM PC/AT Key Names



IBM Enhanced 101/102 Key Names



HP Vectra (non-Enhanced) Key Names



Chapter 31

Alternate Methods of Defining Keystrokes

The standard method of defining a keystroke is by means of a key name. The keyboard layouts, which begin on page 253, show the names of the keys on the supported keyboards. However, a keystroke specification may also be described in terms of its appearance at software interface levels interrupt 16 and interrupt 9. The material in this chapter assumes that you are familiar with these interface levels.

These alternate forms are included to allow limited support of non-standard keyboards. If your keyboard is supported by the KEYMAP.EXE utility program, you should not have to concern yourself with doing keystroke mappings via interrupt 16 or interrupt 9 specifications; specifying the key name is sufficient. See page 233 for a list of supported keyboards.

31.1 Keyboard Diagnostics

A program called KEYMON.COM is provided on one of your diskettes to help you determine how your keyboard and BIOS are interacting to process keystrokes. KEYMON is a tool for diagnosing keyboard problems and investigating the function of your PC's keyboard and BIOS. Keystrokes should normally be defined by name, as indicated on the keyboard diagram that corresponds to your keyboard type.

Type **KEYMON** at the DOS prompt to determine the following:

- Whether the PC BIOS is standard or Enhanced
- What the value of the hardware flag is (*FC* indicates an AT, *FF* is a PC, *FE* is an XT, *F8* is a PS/2 Model 80, and *FA* is a PS/2 Model 30)
- Whether support for the interrupt 15 Function 4F for hooking keyboard interrupts is available

The output looks something like this:

```
KEYMON Version 1.40 -- Keyboard Diagnostic Program
Copyright (C) 1987-90 Walker Richer & Quinn, Inc.
   Type "e" to exit.

PC keyboard BIOS is Enhanced
Hardware type is PC
Interrupt 15 Function 4F is supported
```

To exit KEYMON, type **e**.

31.2 Determining Interrupt 9 and 16 Values

KEYMON monitors both the hardware and software keyboard interrupts. KEYMON reports the scan code of any keyboard interrupts it sees, and reports any characters that become available at interrupt 16 (generated by the BIOS in response to keyboard interrupts).

By running KEYMON and striking keys on the keyboard, you can see both how the keyboard is acting and how the BIOS is reacting. For example, here's what KEYMON would report if you hit an unshifted m on your keyboard:

```
Interrupt 9 scan code: 32h, 50d
Interrupt 16 returns:  AH=32h, 50d   AL=6Dh,109d  <m>
Interrupt 9 scan code: B2h,178d
```

The first line reports the keyboard interrupt that occurred when m was depressed; its scan code is 50. The second line reports the output that became available via interrupt 16 because the BIOS generated a character in response to seeing m go down. It consists of a high byte (AH), which in this case is the key's scan code, and a low byte (AL) that contains the ASCII code for the character **m**. The third line reports the scan code generated when the key was released.

31.3 Using Interrupt 9

The keystroke specification in a keystroke definition can be given in terms of the way the keystroke appears at the interrupt 9 interface level. The syntax for such an input specification is as follows (see page 228 for the full keystroke definition syntax):

```
[<shiftname>] <number> [HIDE] [E0-PREFIXED] [E1-CTRL-PREFIXED]
```

[<Shiftname>]

```
[<shiftname>]: {SHIFT | ALT | CTRL | LSHIFT | RSHIFT | IGNORE_SHIFTS}
```

See *Shift Keys* on page 238 for a complete description.

When you specify **IGNORE-SHIFTS**, Reflection ignores whether left shift, right shift, or neither is used. In the example below, **sys-req** is remapped to **Tab** no matter how it is shifted.

```
IGNORE-SHIFTS 84 = tab-key ;maps 84 (sys-req) to Tab
```

<Number>

<Number> is the decimal number returned by KEYMON in the range 0–255. When you press PgUp on the cursor pad of an Enhanced keyboard, for instance, KEYMON returns something similar to the following:

```
KEYMON Version 1.40 -- Keyboard Diagnostic Program
Copyright (C) 1987-90 Walker Richer & Quinn, Inc.
   Type "e" to exit.

PC keyboard BIOS is Enhanced
Hardware type is PC
Interrupt 15 Function 4F is supported

Interrupt 9 scan code:49h, 73d
Interrupt 16 returns: AH=49h, 73d  AL=00h,  0d  < >
Interrupt 9 scan code:C9h,201d
```

The hex value of the key is *49* (shown as *49h* by KEYMON), and its decimal value is *73* (shown as *73d*). A keystroke specification must be given in terms of its decimal value. To map the VT F6 function to PgUp, use the following:

```
73 = vt-f6   ;maps 73 (pgup) to VT F6
```

HIDE

The interrupt 9 HIDE option indicates that the keystroke should not be passed on to any other routines in the keyboard interrupt chain. For instance, you can map the m key at the interrupt 9 interface level to the Reflection function **user-keys**. The definition would look like this in your keyboard mapping file:

```
50 = user-keys
```

Each time that m is pressed, the user-key labels are displayed and, if you are in remote mode, an **m** is transmitted. The m is interpreted at the interrupt 9 level as the **user-keys** function and at the interrupt 16 level as the character **m**, thus producing two unrelated results.

To restrict the definition of m to the Reflection user-keys function, use the following definition:

```
50 HIDE = user-keys
```

Prefix Options

The interrupt 9 E0-PREFIXED and E1-CTRL-PREFIXED options refer to prefix scan codes issued by Enhanced 101/102 type keyboards that use an Enhanced BIOS. They allow you to distinguish between duplicate keys on the key and cursor pads.

31.4 Using Interrupt 16

A keystroke specification can be a keystroke as it appears at the interrupt 16 interface level. The syntax for such a keystroke definition is as follows (see page 228 for the full keystroke definition syntax):

```
[<shiftname>] <number> <number>
```

[<Shiftname>]

This parameter is discussed on page 259.

<Number>

<Number> <number> are the decimal AH and AL values returned by KEYMON for interrupt 16. Each number can be in the range 0–255.

For example, to make an unshifted m bring up the system keys, first run KEYMON and press m. You should see output similar to the following:

```
Interrupt 9 scan code:32h, 50d
Interrupt 16 returns: AH=32h, 50d   AL=6Dh,109d  <m>
Interrupt 9 scan code:B2h,178d
```

To map the Reflection system-keys function to Alt-m, use the following keystroke definition:

```
alt 50 109 = system-keys
```

To make (right) Shift-F10 home the cursor, run KEYMON and press that particular keystroke. You should see something similar to this:

```
Interrupt 9 scan code:36h, 54d
Interrupt 9 scan code:44h, 68d
Interrupt 16 returns: AH=5Dh, 93d   AL=00h,  0d  < >
Interrupt 9 scan code:B6h,182d
Interrupt 9 scan code:C4h,196d
```

To map the Reflection home-up function to (right) Shift-F10, use the following keystroke definition:

```
rshift 93 0 = home-up
```

See the *IBM Technical Reference* manual for a description of the interrupt 9 and interrupt 16 keyboard interfaces. In general, use interrupt 16 if you can. This way BIOS replacement routines, such as the IBM supplied KEYBGR.EXE, still function.

Section 7

Operations

Chapter 32

Reflection As a Pop-Up Terminal

Terminal emulation means that your IBM PC or compatible can do any operation a terminal can. By using Reflection as a pop-up terminal, you simultaneously have access to all the power of your PC as well as a connection to the host computer.

After loading Reflection into memory, hold down Alt and press Shift on the right side of the keyboard.* This *hot-key* puts Reflection into the *background*. When the DOS prompt appears, you can run other PC software. Whenever you need to have access to your host computer, use the hot-key to pop up Reflection. The letter **B** appears between the key labels once Reflection has been placed in background mode.

32.1 Advantages of Background

Reflection's background mode allows it to continue running while you are using other programs on your PC. You can do word processing, spreadsheet work, or any other PC application while Reflection runs as a terminal, performs file transfers, or executes command files you created with Reflection's command language. Background mode provides you with the following advantages:

- An instant terminal and host connection

* The hot-key keystroke is configurable; see page 157.

- Background file transfers
- Background printing
- Background terminal operations
- Background command file execution

Configuration and Background

You cannot change display memory size, SHELL memory size, printer buffer size or terminal class once you have placed Reflection into background mode. If you need to increase any of these values, do so before placing Reflection in the background, or unload Reflection from memory using [Alt]-[X] or [Alt]-[B] and then reload Reflection.

Using Reflection as a Pop-Up Terminal

Once Reflection is loaded and placed into background, you can press Reflection's hot-key whenever you want to bring Reflection back and have access to the host. The program you are currently using is temporarily interrupted, and Reflection appears on your screen. Operate as a terminal for as long as you like and then use the hot-key again to return to your previous application, exactly where you left off.

Note: Once you go into the background, you cannot change between DEC and HP terminal emulations, although you can run two copies of Reflection, one HP and one DEC (see page 270).

Background File Transfers

File transfers can sometimes take a long time. Once you begin a transfer with Reflection, you can hot-key into the background and have immediate access to your PC. Except for the flicker of drive lights, the transfer continues in the background unnoticed. A message is displayed at the top of the foreground screen when a transfer completes (or an error occurs).

If the backup capability (PLUS) is available, you can back up your PC to an HP 3000, UNIX or VAX/VMS host with Reflection while you use your PC locally.

When you use Reflection for background file transfer or backup/restore operations, you should not try to gain access to these files as they are being transferred.

Background Printing

Instead of tying up your PC while you print a file, use Reflection's command language to begin printing and then toggle Reflection to background mode.

1. Press [Alt]-[Y] to bring up the command line.
2. Type **PRINT <filespec>** where <filespec> is the file you wish to print.
3. Press [Alt]- (right) [Shift] to go to a PC application while Reflection prints your file.

Note: Do not try to print something with a foreground application after Reflection has started printing in the background. When Reflection is finished printing, your foreground PC software again has access to the printer.

Background Terminal

Typically, your PC cannot multitask when you use it as a stand-alone computer. Reflection allows you to maintain a host connection even while you are using PC-only programs. If you are logged on to the host, it can take over the operation of your PC if necessary. Reflection's display memory retains four screens of text sent by the host (the default), so you can switch into Reflection at any time and view text sent from the host.

Reflection also allows the host to control the PC. The host can force Reflection into foreground mode through escape sequences. It can also perform file transfers without your intervention.

Background Command Files

Reflection includes a powerful command language that allows you to program your PC. You can set up repetitive sequences, such as logging on, in a command file. Command files can be executed in the background to free up your PC.

Because command files can perform functions at specific times, you can set up a complete session with your host to begin and end at specific times, all in the background. See the Reflection *Command Language Manual* for a complete explanation.

32.2 SHELL vs. Background

The SHELL command gives you quick access to DOS and allows you to run other PC programs from within Reflection; you can even run a third program or utility when you toggle between Reflection and another application. There may be occasions when SHELL is more appropriate than background (another alternative is the state save keystroke (Alt-B), described on page 329).

From the Global Configuration menu, define the amount of memory SHELL needs *before* Reflection is put in background (you cannot modify the field when the **B** indicator is visible). SHELL memory can also be changed with the command SET SHELL-MEMORY. The default is *0*. Calculate this memory requirement separately from print buffer, display memory, forms buffer, and program memory used by Reflection. For information on switches that limit the amount of memory required when loading Reflection, see *Memory Management* on page 327.

32.3 Background Commands

The following commands are specific to using Reflection's pop-up feature. Refer to the *Command Language Manual* for an explanation of their uses.

ALERT	SET GLOBAL-NLOCK
BACKGROUND	SET HOT-KEY
FGD	SET POPUP-ONLY
FOREGROUND	SET SHELL-MEMORY
HARDEXIT	SET VIDEO-BUFFER
SET ALERT	TSR
SET GLOBAL-CAPS-SCROLL	

32.4 Background Notes

The following notes are on various aspects of background operation. Some are hints for easier use of background and some are cautions about potential problems or errors.

Loading Reflection into Background

From the keyboard

The first use of the hot-key puts Reflection into the background. After background operation is initiated, and you return to Reflection, there are three methods for toggling between the DOS environment and Reflection:

- Alt-(right) Shift (the hot-key)
- F8 from the system keys
- Issuing the EXIT command

The hot-key returns Reflection to the foreground.

From the DOS prompt

You can load Reflection immediately into background mode with the **/B** switch:

```
R1 /B
```

In addition you can include a configuration file and/or a command file. For example:

```
R1 /B <config file> <command file>
```

See page 11 for more information.

From an AUTOEXEC.BAT file

Reflection can also be loaded (into the background or foreground) from an AUTOEXEC.BAT file. Use the syntax described above.

Removing Reflection from Memory

If you need more memory in your PC, you can unload Reflection with Alt-X or use the state save option (Alt-B) described on page 329. Once unloaded, pressing the hot-key has no effect since Reflection is no longer available in memory. To use Reflection again, simply reload it.

Directories

Reflection keeps track of which drive and directory it is in. Initially, this is the drive and directory where you typed **R1** or **R7** to load Reflection. You can change the current directory (just as you can in DOS) by using the CD command on the Reflection command line. Once it is installed into background, DOS and Reflection keep track of their directories separately. Your foreground application and Reflection in the background can be working in separate directories.

To confirm the current path, type **CD** on the command line. The current drive is always displayed to the left of the command line. If you are in the root directory, nothing is displayed when you execute the CD command alone.*

Other Pop-Ups with Reflection

Other memory-resident programs may not be compatible with Reflection in background mode. Popping up other programs within Reflection (such as Sidekick) once it is installed as a pop-up program may not work. Removing Reflection from the background while other pop-ups are present may force you to reboot your PC.

Suppressing Multitasking

If you do not want Reflection to continue to function when it is in the background, type **SET POPUP-ONLY YES** on the command line. You can toggle back and forth between Reflection and DOS, even though you have disabled Reflection's ability to process data in the background. This value can be saved to a configuration file.

Running Two Copies of Reflection

When running two copies of Reflection, you must load the second copy of Reflection at the DOS prompt with the **/O** switch.

Only one copy of Reflection can multitask. However the other copy can be a foreground application that you can use just as you would any other PC application. The copy of Reflection loaded first multitasks; the other freezes whenever it is not the foreground application.

If you have both HP and DEC host computers, you may need to switch between emulation types. To move quickly, the recommended approach is to load Reflection as either an HP or VT terminal first. Then, toggle to background using the hot-key. Now at the DOS prompt, load the other emulation (use **R1V**or **R7V** to load VT emulation). You may then switch easily between terminal classes via your configured hot-key.

If you run two copies of Reflection 7, the first one loaded is the one that determines the video-buffer size for toggling. If, for instance, the first copy has a video buffer size of 8K and the second copy has 16K, you will probably be unable to toggle between the two copies of Reflection.

---

* Removing Reflection's current directory prevents changing from that directory once you return to Reflection; you must unload Reflection from memory with Alt-X and reload it.

32.5

Bringing up Reflection Programmatically

It is possible to bring Reflection to the foreground programmatically. The following assembler code forces Reflection to the foreground:

```
MOV AH, 0DEH
MOV AL, "Q"
MOV DX, 0
INT 21H
```

WR is returned in AX.

If Reflection's hot-key is set to *NONE*, the above code would be the only means of bringing Reflection to the foreground, other than an API application. See the *PLUS Manual* for more information. The Reflection BACKGROUND command can always be used regardless of the setting of the hot-key.

Chapter 33

Reflection Disk Operations

This chapter explains local disk operations (including the various methods for moving data between alpha display memory, a host computer, and a disk file) and summarizes disk function keys.

Instructions are given for the following:

- Reading the contents of a disk file into alpha display memory
- Saving the contents of alpha display memory to a disk file
- Saving the output of a host computer to a disk file

33.1 Adding Text to Display Memory

You can save all or part of the contents of alpha display memory to a disk file on the PC. This procedure is useful for saving data you generate during a host session.

Using READ DISK

You can use the READ DISK or TYPE command to put PC text files into alpha display memory.

To read a text file into display memory:

1. Place the cursor on the row where you want the first line of the file displayed.
2. From the system keys, press [F1] to display the device control keys .
3. Press the [F2], **READ DISK.** A prompt appears:

```
Enter file name:
```

4. Enter the filename. Include path information if it is different from Reflection's current drive and directory. If a filename was entered previously in this session, it automatically displayed.
5. After you specify the filename, press [Return].

As the file is being read from disk, the message *Reading <filespec>* is displayed on the screen row just above the labels.

If the file is larger than the total display memory, Reflection reads only as much of the file as will fit into display memory. When the memory is full, Reflection displays this message:

```
The file would not fit in memory - the first part has been loaded.
Press RETURN to clear.
```

Using the TYPE Command

Another method of viewing a file is to use the TYPE command:

1. Enter the command **TYPE <filespec>** on the command line.

   The entire file is scrolled into alpha display memory one line at a time.
2. To stop the screen's scrolling action, press [Ctrl]-[S]; [Ctrl]-[Q] resumes scrolling.
3. To terminate the TYPE command while lines are being displayed on the screen, press [Ctrl]-[Y].

Note: In VT mode, [ScrollLock] can be used to start and stop the flow of data. In HP mode, use [Alt]- or [Shift]-[ScrollLock] to start and stop the flow of data. **Receive pacing** on the Datacomm Configuration menu must be set to *XON/XOFF* when you are in remote mode.

If the disk file is larger than display memory, Reflection deletes lines from the *top* of memory as it fills up (no warning message is issued). Only that portion of the file that fits into display memory is present at any one time. The configured display memory size determines the amount of the disk file that is retained.

33.2 Saving All of Display Memory to Disk

The MSAVE command saves all of alpha display memory. Text in display memory can be any of the following:

- Data you typed at the keyboard
- An existing file that was edited in alpha display memory
- Data received from the host

To save display memory text to a disk file:

- Type **MSAVE <filespec>** on the command line.

All of alpha display memory is saved to the specified file on the PC, regardless of cursor position or "to" devices. If the file exists, you must choose if you want to overwrite it or append the current display memory to the existing file. If you want to overwrite a file that already exists without being prompted, type **MSAVE <filespec> DELETE.**

33.3 Sending Display Memory to the Host

You can transfer data directly from display memory to the host. The data may originate from a PC disk file or may be typed in from the keyboard. You must use a host program that can accept text.

To transmit the entire contents of display memory to the host, type **MSAVE REMOTE.**

33.4 System Keys Disk Functions

Reflection's disk operations can also be performed using function keys, as explained earlier with **READ DISK.** Three associated sets of screen labels provide disk function operations.

Device Control Keys Functions

Access: [F1] from system keys or device modes keys

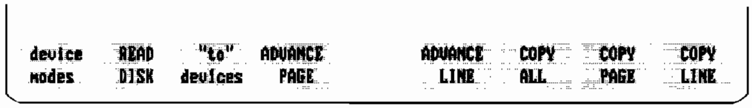


Figure 37: Device Control Keys

Table 21
Device Control Functions

Label	Key	Function
device modes	[F1]	Displays device modes keys.
READ DISK	[F2]	Reads a disk file into alpha display memory.
"to" devices	[F3]	Displays "to" devices keys.
ADVANCE PAGE	[F4]	Rolls the paper in the printer up one page (form feed).
ADVANCE LINE	[F5]	Rolls the paper up one line (linefeed).
COPY ALL	[F6]	Copies all of display memory, starting with the cursor position, to the "to" devices.
COPY PAGE	[F7]	Copies display memory, from the cursor to the last line of the screen, to the "to" devices.
COPY LINE	[F8]	Copies the line containing the cursor to the "to" devices.

Device Modes Functions

Access: [F1] from the device control keys

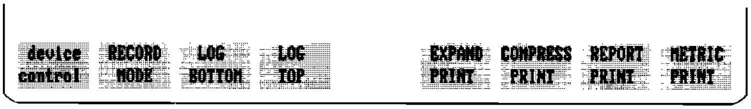


Figure 38: Device Modes Keys

Table 22
Device Modes Functions

Label	Key	Function
device control	[F1]	Displays the device control keys.
RECORD MODE	[F2]	Copies entire contents of display memory to disk file in local mode. In remote mode, sends incoming data to the "to" devices.
LOG BOTTOM	[F3]	Sends data to the "to devices" one line at a time as it is received in display memory.
LOG TOP	[F4]	Sends data to the "to devices" one line at a time as it is forced off the top of the screen during scrolling.
EXPAND PRINT	[F5]	Prints expanded characters (dot-matrix printer).
COMPRESS PRINT	[F6]	Prints compressed characters (dot-matrix printer).
REPORT PRINT	[F7]	Feeds the paper up in the printer after 60 lines have been typed.
METRIC PRINT	[F8]	Feeds the paper up in the printer after 64 lines have been typed.

"To" Devices Functions

Access: F3 from the device control keys

Figure 39: "To" Device Keys

Table 23
"To" Devices Functions

Label	Key	Function
device control	F1	Displays the device control keys.
TO DISK	F2	Prompts for a PC disk filename as a destination device for the copy functions. Toggle **TO DISK** (F2) off to close the disk file.
TO REMOTE	F3	Selects the host computer as a destination device for the copy functions.
TO PRINTER	F5	Selects the printer as a destination device for the copy functions. This label appears only if a printer is specified in the Printer Configuration menu. To send data only to disk, toggle **TO PRINTER** off.

Note: Set **Printer interface** on the Printer Configuration menu to *NONE* if you do not have a printer. This eliminates having to remove it as a "to" device.

33.5 Using the COPY Function

COPY saves all of alpha display memory from the current cursor line to the end of display memory. (MSAVE saves all of display memory regardless of cursor position.) COPY is useful for:

- Copying only from the cursor down
- Saving a specific portion of alpha display memory
- Copying excerpts from various places in display memory

To use the COPY function:

1. Press [F10] then [F1] to display the **device control** keys.
2. Press [F3] to display the **"to" devices** keys.
3. Toggle the **TO PRINTER** label off (no asterisk) by pressing [F5].
4. Press [F2], **TO DISK.**
5. Type in a filename to which you want to save the text lines. Press [Return] when finished. An asterisk is displayed in the **TO DISK** key label.
6. Press [F1] to bring back the device control keys.
7. Press [F6], **COPY ALL** to save all alpha display memory from the cursor to the end of display memory to a disk file.
8. Press [F3], **"to" devices** and press [F2] to close the disk file.

33.6 Saving to a Default Disk File

Reflection remembers the last filename entered during the current session. If a filename has been entered since Reflection was loaded, the prompt that appears when you press **TO DISK** contains that filename.

1. If you want to save to the file named in the prompt, simply press [Return].
2. The screen labels change and a prompt appears:

```
File "D:YOURFILE.TXT" already exists — press one of the
labeled function keys
```

You have the following options:

Table 24
Existing File Storage Options

Label	Key	Action
APPEND TO FILE	F1	Appends the data to the end of the file, preserving existing text.
OVER-WRITE	F2	Replaces the old contents of the file with new data.
RE-ENTER NAME	F8	Leaves the original file unchanged so that a different filename can be entered.

33.7

Saving Information

You can save a single line, a screen, all of display memory, or host output with function keys.

Saving a Screen or Line

Two device control function keys allow you to save only a portion of alpha display memory: **COPY PAGE,** F7, and **COPY LINE,** F8.

1. Press F10, then F1, to display the **device control** keys.
2. Press F3 to display the **"to" devices** keys.
3. Toggle the **TO PRINTER** label off (no asterisk) by pressing F5.
4. Press F2, **TO DISK.**
5. Type in the filename and press Return. An asterisk appears in the **TO DISK** key.
6. Press F1 to bring back the device control keys.
7. Use ↑ and ↓ to move the cursor to the first line of data to be saved.
8. Press **COPY PAGE,** F7, or **COPY LINE,** F8. **COPY PAGE** saves the lines visible on the screen from the cursor line down; **COPY LINE** saves the cursor line.

Operations

9. Press [F3] to get to the "to" devices keys. Press [F2], **TO DISK**, to close the file.

Saving All of Display Memory

You can save the contents of alpha display memory to a disk file by activating record mode while you are in local mode. This method is only effective in HP local mode.

Open a disk file using the **TO DISK** function described above.

1. Press [F1] to display the **device control** keys.
2. Press [F1] again to display the **device modes** keys.
3. Press [F2] to turn on **RECORD MODE**. An asterisk is displayed.
4. When the asterisk disappears from the **RECORD MODE** label, return to the "to" device labels to close the disk file.

When record mode is enabled while in local mode, whatever is in alpha display memory at the time is copied immediately to the selected "to" devices. Record mode is automatically turned off when the copy operation is completed, and also closes the disk file. See the record mode example on page 282 for more information.

33.8 Saving Host Output to a Disk File

The following material assumes that the PC is in remote mode and logged on to a host computer. There are several methods for saving host output to a PC disk file:

- Log bottom saves the output of the host as the data is being read, a line at a time, into display memory.
- Log top allows you to save lines scrolled off the top of display memory to the current "to" devices.
- Record mode saves the output of the host without displaying the data on the screen. Record mode locks the keyboard; follow the procedure in Method 3 (below) carefully.
- The Reflection LOG command copies data arriving from the host through display memory to disk; it functions in the same manner as log bottom.

Operations

- The SET CAPTURE command allows you to capture all data coming down from the host.

Note: Normally, form feeds from the host are only passed on to the printer if record mode is used. Reflection allows you to configure whether or not form feeds are stored in display memory. Data that is logged or copied from alpha display memory to either the printer or a disk file can contain form feeds if DO-FORM-FEEDS is set to *YES* (the default). Also see the SET CAPTURE command in the Reflection *Command Language Manual.*

Method 1: Log Bottom

When **LOG BOTTOM,** [F3], is enabled, data received into alpha display memory, either from the keyboard or from datacomm, is sent to the printer and/or to a disk file a line at a time. Lines must be delimited by linefeeds. When one form of logging is enabled, the other form is automatically disabled.

Escape sequences can be sent from a host computer to enable and disable logging in HP and VT modes. Refer to pages 455 and 539.

Method 2: Log Top

When **LOG TOP,** [F4], is enabled, lines of data are sent to the "to" devices only as they are forced off the top of display memory. When one form of logging is enabled, the other form is automatically disabled.

Method 3: Record Mode

When **RECORD MODE,** [F2], is enabled, you can capture the output of a program running on the host computer. Data received from the host is immediately passed to the specified "to" devices without appearing on the screen. The keyboard is locked except for [Return], [Ctrl]-[Break], soft reset ([Alt]-[S]), and hard reset ([Alt]-[R]).

To save host output using record mode, follow the steps below:

1. From the system keys, press [F1], **device control.** Then press [F3], **"to devices"** and type in or confirm the name of the disk file you want to use.
2. With the device modes keys visible, type the command you want to send to the host system, but don't press [Return] yet.
3. Press [F2], **RECORD MODE.** An asterisk is displayed.

4. Press [Return] to complete the host command entry. The data received from the host is directed to the active "to" devices (in this case, a file).
5. Press [F2] to disable record mode when the operation is finished.

By default, the disk file is closed when record mode is turned off; see SET REC-MODE-CLOSES-DISK in the Reflection *Command Language Manual.* HP record mode and VT record mode can also be initiated by the host with escape sequences; refer to pages 468 and 563 for more information.

Method 4: The LOG Command

1. Press [Alt]-[Y] to bring up the command line.
2. Type **OPEN <filespec>**, and press [Return]. If you want to add data to an existing file, type **OPEN <filespec> APPEND**.
3. Type **CLOSE PRINTER** if you have a printer configured and the output is to be saved to a disk file only.
4. Now type **LOG** and press [Return]. All data subsequently received from the host is saved to the specified file.
5. Type **LOG OFF** to disable logging, and then type **CLOSE DISK**.

Method 5: Capturing Text Files

To capture a file to disk from the host computer use the following steps. The current "to" devices are ignored.

1. Prepare to list the file on the host system. For example, if you are using an HP host, type the following at the host prompt:

   ```
   FCOPY FROM=<hostfile>;TO=
   ```

   Do not complete the command with a carriage return at this time.
2. Bring up the Reflection command line ([Alt]-[Y]) and enter the following commands:

   ```
   OPEN <filespec>
   SET CAPTURE YES
   ```

3. Press [Esc] to clear the command line.
4. Press [Return] to complete the list command on the host.

5. When the file is completely listed, bring up the Reflection command line and enter the following commands to close the file:

```
CLOSE DISK
SET CAPTURE NO
```

The following command file automates the process described above:

```
DISPLAY 'What is the host filename? "
ACCEPT V1
DISPLAY "^M^J"
DISPLAY 'What is the PC filename?"
ACCEPT V2
DISPLAY "^M^J"
TRANSMIT "FCOPY FROM=$1;TO="
OPEN $2
SET CAPTURE YES
TRANSMIT "^M"
WAIT FOR "^Q"
CLOSE DISK
SET CAPTURE NO
```

Note that the $1 and $2 are required. **FROM=V1** would look for a file named *V1*; **OPEN V2** would open a file named *V2*. The SET CAPTURE command allows you to capture all data, including control codes coming down from the host; see the Reflection *Command Language Manual* for more information.

33.9 Using Tabs in Output Disk Files

By default, files are copied to disk without changing spaces to tab characters. This causes the file to take up more disk space, but may make it more acceptable to your PC application software.

If you would like a series of spaces to be replaced with ASCII tab characters when copying lines from alpha display memory to a disk file, you must change the value of **Change spaces to tabs** on the File Transfer Configuration menu to *YES*. Tab stops are set as in the *EDLIN* editor, i.e., column nine and every eighth column thereafter. This produces a smaller file, but the resulting file may not be compatible with your application software.

You can also use the command SET SPACES-TO-TABS. Refer to the Reflection *Command Language Manual*.

33.10

PC File Information

FILEINFO.COM is a program that provides information regarding your PC ASCII files. If the file contains any characters outside of the ASCII set (higher than 127), the ASC/BIN field reads *BINARY*. The program also indicates if the file has been backed up.

At the DOS prompt, type the following:

```
FILEINFO <filename>
```

The parameter **<filename>** may contain wildcards. The program lists the following:

RECSIZE
Specifies the size of the largest record.

ASC/BIN
Indicates whether a file is in ASCII or binary format. Record statistics (**#RECORDS**) are not shown for binary files.

TYPE
Displays *V* if records are of variable length; *F* is displayed if records are fixed-length.

#RECORDS
Shows the total number of records in an ASCII file.

EOF (^Z)
Shows the record where the ^Z end-of-file marker was found. If no ^Z is found, *NONE* is displayed in this column.

BYTES
Shows the number of bytes in the file.

ARCV'D
Indicates whether the file has been backed up by a backup utility (e.g., DOS, or Reflection PLUS backup).

TABS
Indicates whether the file contains TAB (09 decimal) characters.

For example, typing the command **FILEINFO DIAL** at the DOS prompt yields results similar to the following:

```
FILE EXT -----------LOGICAL RECORD---------------
       RECSIZE  ASC/BIN TYPE #RECORDS  EOF(^Z) BYTES ARCV'D TABS

DIAL        78  ASCII  V     101     NONE     2413  YES   YES
```

Chapter 34

Printing

This chapter describes printing in Reflection, providing detailed instructions for the following:

- Printing the contents of alpha display memory
- Printing to a disk file
- Printing a disk file
- Printing data received from a host computer

34.1 Selecting the Printer as the Destination

Before proceeding, make sure that Reflection is configured for a printer. The default configuration assumes a parallel printer with IBM PRO codes. Refer to the *Quick Start Guide* and the *Printer Configuration* chapter on page 135.

If Reflection has been properly configured for a printer, the printer is automatically selected as a destination device when Reflection is loaded. To confirm that **TO PRINTER** is active, use the following steps:

1. Press [F10] to display the system keys.
2. Press [F1], **device control.**
3. Press [F3], **"to" devices.**
4. Look for the asterisk in the **TO PRINTER** label. If it's not there, press [F5] to select the printer as a destination device.

Operations

Printer operations for HP and VT modes are nearly identical. Differences are indicated where necessary.

34.2 Printing Display Memory

There are five simple methods for printing data that is in alpha display memory:

- [PrtSc] prints only a portion of alpha display memory–what's visible on the screen.
- **MSAVE PRINTER** homes the alpha cursor and sends all of alpha display memory to the printer.
- Record mode prints all of alpha display memory.
- The COPY functions print a line, a screen, or all of alpha display memory by means of the device control keys.
- [Enter] can be used in local mode (HP emulation) to send data to the printer.

Method 1: [PrtSc]

There are three [PrtSc] options to print what is visible on the screen:

- Press [Shift]-[PrtSc]. This copies the current screen, including the function key labels, to the printer. The "to" device selections are ignored. [Shift]-[PrtSc] is a DOS function.
- Press [Alt]-[PrtSc]. This copies the current screen, not including the function key labels, to the configured printer, and sends a form feed. The "to" device selections are ignored.
- Press [Ctrl]-[PrtSc]. This is sometimes called *echo printing*: whatever is displayed or typed on the screen is simultaneously sent to the printer. If you are in local mode, make sure that automatic linefeed is on (toggle **AUTO LF**, [F8] from the modes keys); linefeeds are used as terminators. To stop echo printing press [Ctrl]-[PrtSc] again.

Method 2: MSAVE

1. Bring up the command line with [Alt]-[Y].
2. Type **MSAVE PRINTER** and press [Return].
3. The alpha cursor is moved to the top of alpha display memory and all data in display memory is sent to the printer.

Method 3: Record Mode

This method is effective only in local mode.

1. Press [F10] to display the system keys.
2. Press [F1], **device control.**
3. Press [F1] to display **device modes.**
4. Press [F2] to turn on **RECORD MODE.** The entire contents of alpha display memory is sent to the printer.

Method 4: COPY

You may use this method in both local and remote modes.

1. Press [F10] to display the system keys.
2. Press [F1], **device control.**
3. Place the alpha cursor on the first screen line you want to print.
4. Press [F6], [F7], or [F8]:

   [F6], **COPY ALL,** prints all of alpha display memory, beginning with the cursor line.

   [F7], **COPY PAGE,** prints whatever is visible on the screen, beginning with the cursor line.

   [F8], **COPY LINE,** prints the cursor line only.

Method 5: [Enter]

This method applies only to HP emulation. For Enhanced keyboards, use the [Enter] key, for other keyboards, use the keypad [+].

1. Make sure you are in local mode.
2. With the alpha cursor on the screen, press the keypad [+] (or [Enter]) key. Everything on the screen is printed. The current "to" device labels are ignored.

34.3 Printing to Disk

You can configure Reflection to route all printing to disk. The instructions below use the Printer Configuration menu to set this up. You can also use Reflection's command language to print to a file; see SET PRINTER-INTERFACE in the *Command Language Manual.*

The following steps send a screen dump to an ASCII text file:

1. Press [Alt]-[C] to display the configuration keys. Press [F5], **printer config.**
2. Change the **Printer interface** field to *FILE*.
3. Enter a name in the **File name** field. Use SCREEN.DSK for this example.
4. Press [F5] to activate the changes. The file SCREEN.DSK is now open. If you save a configuration file at this point, SCREEN.DSK is closed when you exit Reflection and re-opened when you load it. Each print job is appended to the file.
5. Select a screen you want to print to disk. To print the Help screen, for instance, press [Alt]-[H], then press [Alt]-[PrtSc].

   The screen is saved to SCREEN.DSK. By default, a form feed is appended to the end of the file, unless **Form Feed after Alt PrtSc** has been set to *NO* on the Printer Configuration menu.

If you print to disk using Reflection's other printing functions, like COPY, you need to close the file before you can view or edit it. Close the file using one of these methods:

- Go to the "to" devices and toggle **TO PRINTER** off (an asterisk in the label indicates that it is on).
- Enter a different name in the **File name** field.
- Set the **Printer interface** to something other than *FILE*.
- Exit Reflection ([Alt]-[X]).
- Type **CLOSE PRINTER** on the command line ([Alt]-[Y]).

34.4 Printing a Disk File

Any of the methods described on page 288 will print the contents of a file that is in display memory, provided the file fits. It is sometimes easier to print directly from disk than to read the file into display memory first.

To print a PC disk file larger than display memory, use the Reflection PRINT command. The **TO PRINTER** label need not be active.

1. Press [Alt]-[Y] to bring up the command line.

2. Enter the following command:

```
PRINT <filespec>
```

The entire file is sent directly to the printer, bypassing display memory. You may use DOS path names with the PRINT command.

34.5

Printing Output from a Host Computer

Printing the output of a host computer is similar to printing in local mode, from a disk file, or display memory. The major difference is that Reflection must be in remote mode. Press [Alt]-[M]; an asterisk must be in the **REMOTE MODE** label.

The following procedures to print host output are performed by means of the device modes keys.

Printing with LOG Options

From the system keys, press [F1], **device control**. Press [F1] again . Notice the **LOG BOTTOM** and **LOG TOP** labels. When one form of logging is enabled, the other form is automatically disabled.

Enabling either of these modes directs data to the printer and/or a disk file (depending on the "to" devices).

When **LOG BOTTOM** ([F3]) is enabled, data received into display memory is sent to the printer and/or to a disk file *one line at a time*. Lines must be delimited by linefeeds in order to be recognized as lines.

When **LOG TOP** ([F4]) is enabled, lines of data are sent to the "to" devices *only as they are forced off the top of display memory.*

Pressing [Ctrl]-[PrtSc] begins logging to the printer regardless of the setting of the "to" devices. If a disk file is currently selected on the "to" devices, it is closed. If the **TO REMOTE** label contains an asterisk, it is cleared. To stop printer logging, press [Ctrl]-[PrtSc] again.

Printing in Record Mode

Record mode is used in remote mode to capture the output of a program running on the host computer. When record mode is enabled, data received from the host is immediately passed to the specified "to" devices without appearing on the screen. In HP mode, the keyboard is locked except for [Return], [Ctrl]-[Break], soft reset ([Alt]-[S]), and hard reset ([Alt]-[R]).

Operations

1. Type the command you want to send to the host system, but don't press Return.
2. Press F2, **RECORD MODE.**
3. Press Return to complete the host command entry.

   Data received from the host is sent to all current "to" devices.
4. Press F2, **RECORD MODE,** to disable record mode when the operation is complete.

Printing Host Output Using Commands

Reflection command language provides another simple method for printing host output.

1. Bring up the command line with Alt-Y.
2. Type **OPEN PRINTER** and press Return.
3. Type **LOG** and press Return to send all of the data received from the host to the printer as it is displayed on the screen. Data is sent to all current "to" devices.
4. On the command line, type **LOG OFF** to disable logging, and type **CLOSE PRINTER** to stop sending data to the printer. This disables the printer as an active "to" device.

You can use **MSAVE PRINTER** from the command line to send the current contents of alpha display memory to the printer, regardless of the "to" devices or cursor position.

Saving HP Printer Files to Disk

Files that are normally sent to an HP 3000 printer can be received on the PC as disk files. For example, if you want information in the same format as reports created for the host printer, generate the printer output and then transfer the output to the PC. A priority of 1 on a print job holds the file in the print spooler. When the file has been prepared for the printer, use the following steps:

1. Run SPOOK (documented in the HP *Utilities* manual).
2. Use the SPOOK SHOW command to see what files are yet to be printed.
3. A file suspended with a low priority can be copied into an MPE file with the following SPOOK command:

```
COPY #<file#>;ALL,<filespec>
```

4. To maintain the control characters but transform them into ASCII control characters that your PC printer understands, use a file equation to set Reflection as the destination for the FCOPY command:

   ```
   FILE WINDOW;DEV=ldn;REC=-132;CCTL
   ```

   FCOPY converts control characters to ASCII when copying to the terminal (in this case, Reflection). SHOWME at the MPE prompt shows your <ldn> number. The REC size is for files with 132 columns.

5. Type the following FCOPY command at the MPE prompt without pressing Return:

   ```
   FCOPY FROM=<filespec>;TO=*WINDOW
   ```

6. At the Reflection command line, type **OPEN <filespec>** and press Return. (The setting for DO-FORM-FEEDS must be *YES*, the default.)
7. Enter **SET CAPTURE YES** to turn on capture mode. The "to" devices are ignored.
8. Press Esc to exit the command line, and press Return after the FCOPY command entered in step 5 above.
9. When finished, turn off capture mode (**SET CAPTURE NO**) and close the file (**CLOSE DISK**).
10. Now you can edit and print the file. Any control characters in the host file are correctly interpreted by the PC printer.

34.6 Printer Control Functions

Reflection supports several printer control functions that allow you to customize the printing operation to your immediate printing needs and preferences without reconfiguring the program. These functions can be enabled from the keyboard through the device modes keys.

Expand Print and Compress Print

If one of the selected destination devices is a dot-matrix printer capable of printing expanded and compressed characters, you can use **EXPAND PRINT** (F5) and **COMPRESS PRINT** (F6). Select the correct printer in the **Control codes** field on the Printer Configuration menu.

Report Print and Metric Print

When **REPORT PRINT** (F7) is enabled, Reflection automatically sends an ASCII form feed character to the printer after each group of 60 lines to provide a margin at the top and bottom of each page. If form feeds exist in the file being printed, they are acted on separately.

METRIC PRINT (F8) has the same effect, except that the number of lines per page is 64. In either mode, the line counter is restarted every time an ASCII form feed character is sent to the printer.

When one of these modes is enabled, Reflection automatically disables the other.

Printing Wide Reports

Reflection is capable of printing lines as wide as your printer can handle. If you want to print lines wider than 80 columns, you must configure the line width fields of the Printer Configuration menu to match the capabilities of your printer.

When Reflection receives a line of data that extends beyond column 80 or the right margin (whichever is greater), it automatically wraps around to the next line on the screen and marks the new line as a *continuation* line. This happens regardless of the data source (keyboard, host, or disk file). The continuation stays with the line in alpha display memory until a line is inserted or deleted immediately above it. When lines are printed from display memory, continuation lines are sent to the printer as if they were part of the previous line, i.e., without a carriage return or linefeed.

Your host computer may be configured to limit the length of lines that it sends to Reflection to 80 characters. To override this, configure Reflection (Terminal Configuration, Page 2 menu) for a larger right margin (for example, 133). Enter the following file equation at the MPE prompt:

```
FILE <filespec>=$STDLIST;REC=-133
```

You can then send data to that file using FCOPY:

```
FCOPY FROM=<printfile>;TO=*<filespec>
```

If you are using log bottom to print the file, see the information below about capturing form feeds.

You may find a difference between the way your host computer and Reflection handle line spacing and page ejects on the printer. The data going to the printer from Reflection generally must use only the ASCII carriage return, form feed, and linefeed characters to control vertical spacing.

Capturing Form Feeds

Any form feeds that enter Reflection's display memory from the host are discarded. Therefore, logging to the printer or to disk does not include form feed characters. There are several options:

- You can use CAPTURE to print as described on page 292 in this chapter.
- You can use record mode to print. Form feeds are sent directly to the printer in record mode.
- You can set DO-FORM-FEEDS to *YES* on the command line. See the Reflection *Command Language Manual.*
- You can enter form feeds directly into display memory wherever you would like them. From the modes keys (F4 from the system keys), toggle **REMOTE MODE** off and move the cursor to the correct location. Toggle **DISPLAY FUNCTNS** on and press Ctrl-L to place a form feed control character in the text. When the file is printed or saved to disk from display memory, the form feed characters are kept.

Advance Page and Advance Line

ADVANCE PAGE and **ADVANCE LINE** on the device control keys give you immediate control of the printer from the keyboard.

- Press F4, **ADVANCE PAGE**, to send an ASCII form feed character to the printer.
- Press F5, **ADVANCE LINE**, to send an ASCII linefeed character to the printer and advance the paper one line.

34.7 Using and Clearing the Print Buffer

Print buffers allow you to print without tying up your keyboard. You can define any amount for a print buffer; 32K is recommended for printing graphics display. When you perform a printing operation, data fills Reflection's print buffer, leaving your keyboard free. If you are printing large files, you may want to temporarily increase the size of the buffer.

Use Ctrl-F1 to clear the contents of Reflection's internal printer buffer. A soft reset (Alt-S) also stops transmission of data to the print buffer. (The printer continues to print if it has its own built-in buffer.)

34.8

Sending Escape Sequences to the Printer

If your printer can be controlled by escape sequences, use the PRINT command followed by the escape sequence enclosed in quotes. Below is the HP LaserJet command to switch to the Courier font:

```
PRINT "^[&l0O^[(8U^[(s0p10h12v0s0b3T"
```

Courier is recommended if you are sending text files not previously prepared for printing on the LaserJet. This font avoids proportional spacing problems.

Reflection fully supports HP's LaserJet printer, either directly connected to the PC or connected to the HP 3000. If the LaserJet is connected to the HP 3000 and defined as device LASERJET, use a file equation such as the following to define the device:

```
FILE LJ;DEV=LASERJET;CCTL
```

Then, use LJ as the host filename in the SEND command.

If you are sending graphics or PC printer files that include escape sequences, you must use a program on the host to prevent these sequences from being read by the host. RSVP (*Reflection's Spooled Virtual Printer*), a Reflection Complement, ensures that the file is transmitted to the host printer intact. The PC portion captures to a disk file everything that would have been sent to a directly attached printer. This gives your PC application program full control over how the file is printed.

Chapter 35

Using Reflection with Other Software

35.1

Running Reflection with Windows

There are two ways you can use Reflection with Windows:

- If you have an 80386 computer (or higher) with at least 2MB of memory (640 of conventional memory and 1024K of extended memory) you can run Reflection under Windows *386 enhanced mode*. This gives you the benefit of Windows memory management capabilities.
- If you don't have the disk and memory resources to run in 386 enhanced mode, you can run Reflection under Windows *standard* or *real* mode. See your Windows documentation for explanations of these modes.

Cut and paste can be used to move text to and from the Reflection window, and Reflection multitasks even when it is just an icon on the screen.

The PIF File

Four program information files (.PIF) are included on the Reflection diskettes:

- R1386.PIF and R7386.PIF are for running Reflection 1 and Reflection 7 in enhanced 386 mode
- R1.PIF and R7.PIF are for running Reflection 1 and Reflection 7 in standard or real mode

If you want to run VT emulation Windows, copy the appropriate .PIF file and rename it:

Copy...	to...	to run
R1.PIF	R1V.PIF	Reflection 1 (VT) in standard or real mode
R7.PIF	R7V.PIF	Reflection 7 (VT) in standard or real mode
R1386.PIF	R1V386.PIF	Reflection 1 (VT) in 386 enhanced mode
R7386.PIF	R7V386.PIF	Reflection 7 (VT) in 386 enhanced mode

See the guidelines on page 300 for information on settings you must change in the .PIF file if you are running VT emulation under Windows.

The Reflection .PIF files are in Windows 3.0 format, but they can also be read by earlier versions of Windows. Copy the appropriate PIF file to your Windows directory. To start Reflection, double-click on the PIF file from within Windows.

Most of the information necessary for running Reflection is in the provided PIF file. You may need to change some fields, like the drive and subdirectory of the Reflection program.

If you are using a pre-3.0 version of Reflection, add /I to the PIF run parameters.

Windows 3.0

The following figure shows the parameters set by R7386.PIF. The defaults for other Reflection products may vary slightly.

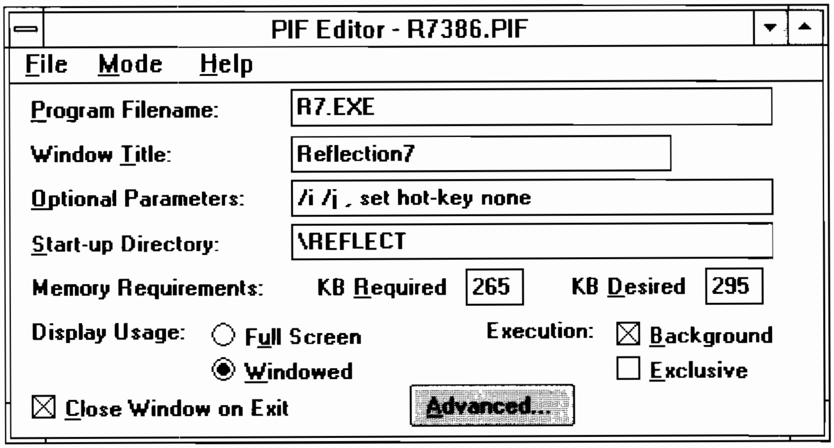


Figure 40: R7386.PIF for Windows 3.0 (Page 1)

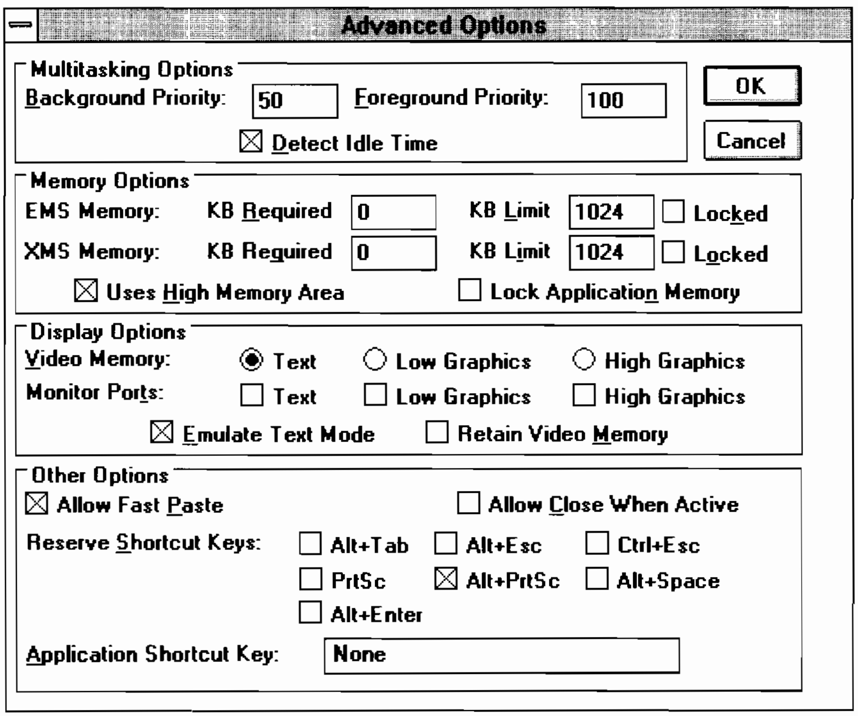


Figure 41: R7386.PIF for Windows 3.0 (Page 2)

The following guidelines are for Windows 3.0:

- For Windows 3.0, the hot-key (Alt-Tab) only works if Windows is running in 386 enhanced mode.
- If you have a PIF file from a previous version of Reflection, replace it with a new Reflection 4.0 PIF file.
- The **Display Usage** options let you specify how Reflection starts in Windows. (These options are not available if you're running in real or standard mode.) Select **Full Screen** or **Windowed**. Select **Full Screen** if you are configuring Reflection 7 to start in graphics mode.
- The **Execution** options let you specify how Reflection runs with other applications. These options are not available if you are running in real or standard mode. Select **Background** if you want Reflection to continue to run when other applications are in the foreground. Select **Exclusive** if you want to suspend all other DOS (non-Windows) applications when Reflection is running. Windows applications (that is, applications that don't require .PIF files) continue to run.
- If one of the .PIF files distributed with this version of Reflection is edited with a pre-3.0 PIF editor, the Windows 3.0-specific fields will return to their default values.
- To modify a PIF file for VT emulation, set **KB Required** to 130 and set **KB Desired** to 160.
- Under Windows 3.0, you will probably have receiver overrun problems (lost data that shows up as a triangle symbol on your screen) if you use COM3 or COM4 on a non-microchannel AT-class machine. Windows is unaware of the IRQ and address that each of these COM cards is using. Use the steps below to solve this:

    1. Run COM34WIN.EXE (provided with Reflection) before running Windows. Putting it in your AUTOEXEC.BAT file is the simplest way.

    2. Add the following lines to the [386 Enh] section of your Windows configuration file, SYSTEM.INI. This file should be in the Windows directory.

```
COM3IRQ=10
COM3BASE=3E8H
COM4IRQ=11
COM4BASE=2E8H
```

The COM3BASE and COM4BASE lines are correct for all COM3 and COM4 cards. COM3IRQ=10 and COM4IRQ=11 match the default IRQs that Reflection uses for these COM ports. However, many COM3 and COM4 cards use other IRQs. If your COM card requires a different IRQ assignment, change the **Datacomm port** on the Datacomm Configuration menu and the values entered in SYSTEM.INI to match those of the COM card. See page 92 for IRQ configuration information.

Note: On microchannel PS/2 machines, neither of these steps is necessary.

Copying to the Clipboard

In Windows, the Alt-PrtSc keystroke is used to copy the screen image to the clipboard. This is also the default keystroke for Reflection's print screen operation. You have two options for resolving this conflict: disable this keystroke in Windows, or remap it in Reflection.

- To disable the Alt-PrtSc keystroke in Windows, mark "Alt+PrtSc" on page 2 (Advanced Options) of the .PIF editor. (This is done for you if you used the provided .PIF file.)
- To remap the Alt-PrtSc keystroke in Reflection, follow these steps:

  1. Create a keyboard remapping file called ALTPRTSC.KBM.

```
keyboard-id = enhanced
                        ;the keyboard is Enhanced
term = HP
alt period = print-screen
                       ;map Alt-. (or another keystroke of
                       ;your choice) to perform the Reflection
                       ;print-screen function
```

  2. Use the keyboard remapping compiler to add this remapping to a configuration file:

```
C:> KEYMAP ALTPRTSC.KBM <configuration filename>
```

The new mapping will take effect when you load Reflection using <configuration filename>. For further information on keyboard remapping, see page 221.

35.2

DESQview

DESQview is a PC program that lets you run a number of applications simultaneously, each in its own window. Reflection can also be run inside a DESQview window, provided graphics are disabled.

In order for DESQview to install Reflection correctly, the following information needs to be added.

1. From the first menu, go to *Open a Window* and select *Add a Program.*
2. In the **Memory size (in K)** field, enter *180* for Reflection 1 and *200* for Reflection 7.
3. In **Parameters**, enter */A* (ignore the graphics adapter).
4. The following options should all be set to *NO*:

   Writes directly to screen
   Displays graphics information
   Can be swapped out of memory

35.3

Precision Architecture (Spectrum) Users

If you are connected to a Precision Architecture host, no Reflection configuration changes are required.

- The default settings for **Receive pacing** and **Enq/Ack pacing** are appropriate for for Precision Architecture environments.

  Reflection matches the HP 700 series terminals for the default value of **Receive pacing**—*XON/XOFF*. This is significant for Precision Architecture users: Precision Architecture (MPE/XL) uses XON/XOFF pacing instead of the former HP 3000 standard of ENQ/ACK. This means that, should the host send characters to Reflection at a rate faster than they can be displayed, it sends an XOFF (ASCII 17) character as a request for the host to pause.

  XON/XOFF pacing is supported by the older HP terminals and by all versions of Reflection. In the HP 700 series of terminals this is the default for **Receive pacing**. The HP 700/92 and HP 700/94 terminals (and Reflection) continue to provide ENQ/ACK pacing for MPE compatibility.

If you are connected to an MPE/XL machine through a terminal controller device and experience flow control problems, there may be a problem with the interpretation of XON with the intermediary device. See page 80.

- When transferring files to spooled devices, use **;O (;Output)** as part of the host filename. This prevents extra form feeds being issued on the printer.

35.4 Using TDP/3000 with Reflection

Under most circumstances, the TDP text editor limits you to editing only a line at a time. By using TDP's SCREEN command, you can take advantage of Reflection's extended display memory and issue a series of commands that give you the full screen editing capabilities of an HP terminal.

To use this feature:

1. Call up the TDP program on the host.
2. From the TDP prompt (/), specify the file you want to edit by entering **T <filespec>**.
3. At the next TDP prompt, enter the command **set screenmax=250**.
4. At the next TDP prompt, enter the command **screen** <x/y>, where <x> is the first numbered line and <y> is the last numbered line you want to edit.

   You can also enter the command **screen all** if you want to edit the entire file.
5. When TDP asks for your terminal type, type **2622**.
6. To end screen mode, press [Enter] (keypad [+] or [Enter]).

This procedure allows you to screen up to 250 lines of a file at a time and to use all the editing functions of an HP terminal.

35.5 Using EDT with VAX/VMS

By default, Reflection VT emulation maps the functions used within EDT to the following PC keys:

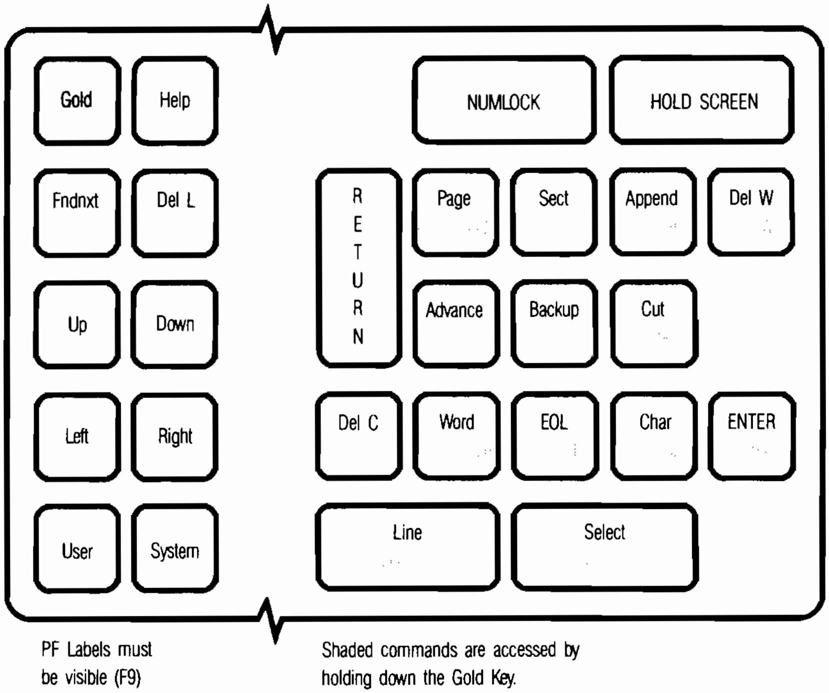


Figure 42: EDT Keyboard Layout for the IBM PC

Del	Rubout character
Linefeed	Rubout word
Backspace	Beginning of line
Ctrl-**A**	Compute tab level
Ctrl-**D**	Decrease tab level
Ctrl-**E**	Increase tab level
Ctrl-**K**	Define key
Ctrl-**T**	Adjust tabs
Ctrl-**W**	Refresh screen
Ctrl-**Z**	Return to line mode

35.6 Using DOS Command Line Editors

Certain resident command line editors are deactivated when Reflection loads, and must be enabled again.

To avoid this, start Reflection with the **/Y** switch. Type **R1/Y** or **R7/Y** at the DOS prompt to make Reflection compatible with such editors as CED, DOSEDIT, and NDE.

35.7 Transferring QEDIT Workfiles

Walker Richer & Quinn's PCLINK2 program by default transfers files in QEDIT format to the PC as straight ASCII text files. (KEEP is no longer necessary before file transfers.) Use **,NQ** as part of the host startup sequence if you want to override this default.

When sending files with a QEDIT format back to the host, add **;Q** to the host filename. If the file already exists on the host in QEDIT format, this switch is not necessary.

Retaining File Attributes

QEDIT files can have a *language* attribute: FORTRAN, Pascal, COBOL, SPL, 80-byte job, or text. The example below starts a QEDIT session for a new COBOL file:

```
:RUN QEDIT.PUB.SYS
SET LANGUAGE COBOL
NEW COBFILE
```

If you transfer a QEDIT file to your PC, Reflection writes a label to the PC file that describes the QEDIT file attributes.

```
RECSIZE=80; LANGUAGE=3
```

QEDIT LANGUAGE codes are listed below.

1 SPL (72-byte records)
2 FORTRAN (72-byte records)
3 COBOL without comments (66-byte records)
4 RPG (80-byte records)
5 Job (80-byte records, KEEP UNN)
6 Text (SET LENGTH sets record size up to 256 bytes)
7 Pascal (72-byte records)
8 COBOLX with comments (74-byte records)

LANGUAGE tells QEDIT the record format and which compiler to use with the source file. For *Transact* or *Powerhouse*, you can use SPL or JOB, and then invoke the interpreter via a user-defined command (UDC).

When you use Reflection to send files from the PC to the HP 3000 with the QEDIT option, it uses the label in the PC file, if any, to create a QEDIT file with the proper attributes.

QEDIT files are received on the PC without sequence numbers, since they are seldom used by PC tools. When a PC file is uploaded into QEDIT, new sequence numbers are assigned to the lines.

If you are receiving a QEDIT file that contains PC data (such as a Reflection command file or an Assembler program), use SET LANGUAGE JOB in QEDIT for PC files. A JOB file has 80-byte records and is saved without sequence numbers. When you download a QEDIT JOB file, Reflection does *not* add a label to it; JOB is the default.

QEDIT's Full-Screen Mode

With an ordinary HP terminal, QEDIT's VISUAL mode is limited to 80 columns of data per line; scrolling left and right is done by QEDIT and is slow and cumbersome. In the past, even if Reflection were configured to have 256 columns per line, QEDIT's screen mode would only read and write 80 columns per line.

QEDIT, as of version 3.6, takes advantage of Reflection's ability to handle long lines. It detects that you are using Reflection (via the ^{E}c*s12346^ status request) and automatically increases the width of display memory whenever needed. To scroll the screen image left and right, use [Ctrl]-[←] and [Ctrl]-[→].

Reflection Configuration Changes in QEDIT

To make QEDIT as fail-safe as possible, the following settings are automatically changed in Reflection:

- Completion codes are enabled.
- To keep the function key labels from blinking as Reflection executes commands, QUIET COMMAND ON disables display of the command line.
- To avoid minor problems with typeahead in block mode, typeahead is disabled while in VISUAL mode.

- Reflection's variable V9 is used to save and reset typeahead for the user (i.e., if typeahead was on before entering VISUAL mode, it will be on again when you come out). Because QEDIT uses the V9 variable, it can't be used in any of your own Reflection command files.

To accomplish these changes, QEDIT performs the following Reflection commands:

```
SET DISABLE-COMP-CODES NO
QUIET COMMAND ON
LET V9=VALUE(TYPE-AHEAD)
SET TYPE-AHEAD NO
SET TYPE-AHEAD $9
```

In addition to character mode and screen mode, QEDIT has a third edit mode called *VEMODIFY*, a single-line, interactive, visual mode. VEMODIFY is notified of Reflection's wider margins and is able to make use of them.

File Transfers Within QEDIT

QEDIT accepts MPE commands with or without a colon. If QEDIT does not recognize a command as one of its own, it tries to interpret it as an MPE command. QEDIT accepts the same RUN command that MPE does, eliminating the need of constantly reconfiguring Reflection's **Host startup sequence** to add a colon before the RUN statement. Another benefit is that you can put MPE commands into your function keys and use them with both QEDIT and MPE.

Reflection Commands in QEDIT

QEDIT has a :REFLECT command that allows you to execute any Reflection command. You can include Reflection commands in UDCs.

```
:REFLECT receive report.dta from reptdata delete
:REFLECT chdir \ lotus
:REFLECT shell 123
```

Having Reflection's command language available as part of QEDIT's command language allows more PC-HP 3000 integration. For example, using a QEDIT UDC, you could do a :REPORT into a file, then download the file to a PC and update a PC database that monitors use of HP 3000 system resources.

35.8

Using Host Data with LOTUS 1-2-3

Files to be imported by Lotus 1-2-3 must follow certain format rules:

- The filename extension must be .PRN, as in *PRINTOUT.PRN*.
- Labels must be delimited by single or double quotes (' or ").
- Numbers must be separated by any character other than a number or an embedded period. This means that Lotus 1-2-3 treats 1,239.99 as two numbers: 1 and 239.99.
- Non-numeric characters to be displayed must be delimited by quote marks ("). Lotus 1-2-3 does not display anything placed outside quotes, except for numbers and embedded periods.

A sample import file follows:

```
""""col 1  """  col 2  """ col3  "
"row 1"11.11,222.22,333
"row 2 "111.22 222.33    344
```

Note that the *columns* in the above file need not be aligned vertically.

Producing Files in Lotus 1-2-3 Format

Reports generated by application programs, while they may not be entirely compatible with Lotus, can be transferred via Reflection to a PC and then edited locally to make them readable by Lotus 1-2-3. More useful is output generated specifically for Lotus 1-2-3. HP users have written programs in QUIZ, COBOL, and QUERY, which produce Lotus-compatible files. These files can then be transferred to a PC and imported by Lotus 1-2-3.

The following is an excerpt from a QUERY program to produce a Lotus-compatible file. It was run from Reflection, and its output was copied from display memory to the PC's disk under a .PRN filename. Once on the PC's disk, the file was imported by Lotus 1-2-3, as described below in *Importing Files*.

```
>REPORT
>>H1,""""PRICE"""",13;D,ITEM-PRICE1,13,E2;E2,"ZZZZZZ.99"
>>H1,""""ON HAND"""",25;D,QTY-ON-HAND,25;END
```

The QUERY output is:

```
"PRICE"        "ON HAND"
295.00         89
395.00         17
```

Importing Files

The following table shows the commands necessary to import a .PRN file to Lotus 1-2-3.

Table 25
File Import Commands for LOTUS 1-2-3

Command	Meaning
/	Enter command mode
F	Access files
I	Import a file
N	Allow numbers as well as labels
<filespec>	Enter the name of the file you want to import (the .PRN extension is not necessary)

35.9

File Transfers from HP Desk

In order to perform file transfers from HP Desk, you must have MPE access.* If you are always working within HP Desk, place a colon before the *RUN PCLINK2* statement in the **Host startup sequence** on the File Transfer Configuration menu. Then save the change to the default configuration file.

If you perform file transfers both from within and outside HP Desk, it is best to set and then reset the startup sequence.

To simplify implementation, a system administrator can set up the two UDCs described below. Both can be called by the HP Desk UDC already set up on your system. The first UDC might be called *SETLINK* and contain the following lines:

* DeskDirect is a Reflection Complement that allows you to transfer files via HP Desk whether you have MPE access or not.

Operations

```
OPTION LIST
COMMENT EcC&oFSET HOST-STARTUP ":run pclink2.pub.sys"CR
```

The LIST option directs the UDC to echo any following lines to Reflection (MPE normally does not echo). The comment line changes the host startup sequence so that a colon precedes it. Note that EC and CR represent keystrokes entered with display functions turned on. The EC&oF sequence is one of three Reflection escape sequence that invokes Reflection's command interpreter.* It does not return a completion code.

To reset the startup sequence upon exiting HP Desk, a second UDC containing the following lines could be used:

```
OPTION LIST
COMMENT EcC&oFSET HOST-STARTUP "run pclink2.pub.sys"CR
```

This UDC, perhaps called *RESETLINK*, would return the startup sequence to its default setting (without the colon).

Several lines of the HP Desk UDC follow:

```
SETLINK
CONTINUE
RUN HPMAIL.HPMAIL.SYS;LIB=g;PARM=:tray;INFO="!user"
RESETLINK
```

The CONTINUE indicates that if an error occurs while trying to run HP Desk, the rest of the UDC should be executed anyway. This ensures that the startup sequence is reset by the RESETLINK UDC, regardless of whether HP Desk runs successfully or not.

35.10 X.25 Software with HP INP

If you're using TYMNET/Telenet and have an HP INP with X.25 software, configure Reflection as follows:

Datacomm Configuration menu:

Check parity	NO (the default)
Receive pacing	XON/XOFF (the default)
Transmit pacing	XON/XOFF

* Note that pre-3.4 versions do not recognize this sequence. If you have an old version, see DISABLE-COMP-CODES in the *Command Language Manual.*

Terminal Configuration menus:

Type ahead	NO
Use host prompt	NO
Inhibit handshake	YES
Inhibit DC2	YES

File Transfer Configuration menu:

Host startup sequence	RUN PCLINK2.<grp>.<acct>
File transfer link	8-BIT

Then, when signing on to the host, use *;TERM 24*.

If you find that you are experiencing data errors, change the presentation by typing the following on the Reflection command line:

```
SET TRANSFER-PRESENTATION "B"
```

This changes **File transfer link** on the File Transfer Configuration menu to *USER-DEF*. If this does not solve the problem, change the value of **File transfer link** to *7-BIT*. The selection of *7-BIT* may slow down transfers considerably, but files transfer successfully.

X.25 Software without HP INP

If you don't have an HP INP running X.25 software, you must also disable echo at the local PAD and use the local echo feature of Reflection. To disable echo at the PAD, consult the documentation for your particular network. In Reflection, set **Local echo** to *YES* on the Terminal Configuration, Page 1 menu.

Check to see if DC1 (^Q) characters pass from the host computer to the PC. (This is determined by host PAD configuration and the terminal type setting of the host port.) If DC1 characters are being passed to the host, you should be able to produce them in Reflection.

From the modes keys, press F7 to turn on display functions, and type carriage returns at the host prompt. If a $^{D}C_{1}$ exists, the following sequence appears:

$^{L}{}_{F}$:$^{D}{}_{1}$$^{C}{}_{R}$
$^{L}{}_{F}$:$^{D}{}_{1}$$^{C}{}_{R}$
$^{L}{}_{F}$:$^{D}{}_{1}$

If you see this sequence, no special configuration changes (other than local echo) must be made; use the defaults.

If $^{D}C_{1}$ characters do not appear, configure Reflection as noted above for X.25 software with HP INP.

35.11 Oracle: Switching Terminal Modes

Oracle applications can take advantage of Reflection's ability to switch *
between HP and DEC terminal using the guidelines below. *

Use of the hot-key is prohibited for this support; once the hot-key has been *
used in Reflection you can no longer switch between HP and DEC terminal *
classes. If you have used the hot-key, you must exit (Alt-X) and restart *
before running Oracle. This is true for both HP 3000 and HP 9000 hosts. *

Note: If you use (Alt-B) to exit Reflection while using Oracle, you must reload Reflection with the same terminal class. If you were in the *DEC* terminal class when you pressed Alt-B, type **R1V** or **R7V** when you load Reflection again. If you are operating under the HP emulation when you press Alt-B, type **R1** or **R7** when you load Reflection again. See page 329 for general state save information.

HP 9000

Oracle on the HP 9000 can be started from Reflection's DEC or HP emulation. The **/SP** switch mentioned below for the HP is not required, but no error results if it is added.

HP 3000

After making the configuration changes described below, use the **/SP** switch when loading Reflection for Oracle applications. This switch tells Reflection to use a state file to pass terminal state information between HP and DEC emulations. Using **/SP** allows you to hold network sessions open when switching terminal class.

To automate the procedure, you can add this switch to your AUTOEXEC.BAT file in a set environment command.

```
SET R1SW= /SP
```

You can add other switches to this command. Any switches specified are automatically used each time you load Reflection.

Configuring for Oracle

Make the following configuration changes in Reflection to support Oracle:

1. Activate the DEC emulation by bringing up the config keys ([Alt]-[C]) and pressing [F7], global config. Change **Terminal class** to *DEC*. Activate the configuration (press [F5]).
2. After you have confirmed that your datacomm choices are set correctly, press [Alt]-[Y] to bring up the command line. Verify that SET ENABLE-SWITCHING is set to *YES* (the default); type the following on the command line:

   ```
   VERIFY ENABLE-SWITCHING
   ```

3. Type the following to change the host prompt. This allows Oracle to load the [F8] user key properly.

   ```
   SET HOST-PROMPT "^Q"
   ```

4. Finally, change the terminal type (the default is VT102).

   ```
   SET TERMINAL-TYPE VT220-7
   ```

5. Save these settings to disk by typing SAVE on the command line. This saves all configuration settings to your current configuration file, for example, *R1.CFG*.

Reflection supplies files that allow you remap the numeric keypad and adjacent keys to operate like the terminal keyboard you prefer:

Terminal Keyboard	PC Keyboard	Reflection file
HP700/92	Enhanced 101/102	VT_EN700.KBM
	AT	VT_AT700.KBM
DEC	Enhanced 101/102	VT_ENH1.KBM
	AT	VT_AT1.KBM

Operations

To remap these keys, type the following at the DOS prompt:

```
KEYMAP <remap file> <config file>
```

For example:

```
KEYMAP VT_EN700.KBM R1.CFG
```

Section 8

Special Features

Chapter 36

Reflection Terminal Features

36.1 Typeahead with an HP 3000

Typeahead is a terminal enhancement that allows you to continue typing while the HP 3000 processes the last command. Configure the **Type ahead** field in the Terminal Configuration, Page 1 menu to *YES*. The keyboard is never locked when **Type ahead** is enabled, so if you know what input is required next, you can go ahead and type it.

When **Type ahead** is set to *YES*, you may have to press the [Alt]-[T] (reset typeahead) key combination occasionally. This is necessary when Reflection is waiting for a $^{D}C_{1}$ host prompt character from the host, and the $^{D}C_{1}$ is not forthcoming. This can occur under the following conditions:

- When initially logging on, if the HP 3000 does not respond with a colon (:) prompt after the first time you press [Return]
- When an MPE SPEED command is entered to change the baud rate
- When you log off the HP 3000, and then try to log on again without restarting Reflection

[Alt]-[V] allows you to clear the keyboard buffer if you decide not to transmit the keystrokes you have just entered. After clearing the buffer, you can continue to type ahead. *

36.2

132-Column Adapters

Reflection supports the 132-column mode of a variety of adapter boards. See page 158 for more information. Using this feature requires that the right margin be set to more than 80 on the Terminal Configuration, Page 2 menu.

36.3

Display Memory

Reflection saves the data it receives from datacomm in its display memory. You can configure the size of display memory on the Global Configuration menu before Reflection is installed as a background program. The size depends on the amount of memory available and what other programs you want to run; see page 156. The characters visible on the screen at any one time may be only a small part of the total display memory.

Text Window Size

Either 23 or 24 text lines may be available on the screen at one time. When the function key labels occupy two lines, the window can display 23 lines of text (unless 27-line mode is available to you).

Enhanced graphics adapters and Vectras come up in Reflection with a 27-line display. Some programs do not correctly reset the 27 lines available when toggling in and out of Reflection in background mode. Reflection allows you to suppress 27-line mode by using the /C or /I switch. See page 12. Reflection uses the IBM font when operating with these switches: only a subset of the Roman 8 character set is available with the IBM font. See page 524.

36.4

Character Mode vs. Block Mode

In character mode, each letter, numeral, and special character typed at the keyboard transmits the ASCII code for that key to the host computer. Reflection then waits for the host computer to *echo* the character back over the communications link before it displays the character on the screen.

HP mode has one more option, *block* or *remote* mode (F3 on the modes keys). When **BLOCK MODE** contains an asterisk, Reflection is in block mode; otherwise, it's in character mode.

Block mode can be activated in three ways:

- Typically, block mode is initiated by software running on the host computer.
- You can toggle the **BLOCK MODE** label on.

- You can enter the block mode escape sequence from the keyboard (see page 465).

In block mode, Reflection waits until [Enter] is pressed before it transmits characters to the host. If you're working on an IBM PC or workalike, [Enter] is the gray keypad [+] key. IBM Enhanced keyboards have their own keypad [Enter] key. The block of data transmitted may vary in size from a few characters to several screen pages. (For more detailed information about block mode operations, see page 495.)

36.5 Display Functions

The ASCII character set includes 33 control characters with the decimal values 0 through 31 and 127. (See page 519 for a table of ASCII control characters.) Reflection doesn't normally display these characters on the screen; instead, it performs the function (if any) that each character represents.

For example, the carriage return control character (decimal value 13) moves the cursor back to the left margin of the current screen row.

When **DISPLAY FUNCTNS** ([F7] on the modes keys) is enabled, Reflection displays most of these characters on the screen and does not execute them. You can see exactly what characters are received from the host computer, and what control characters are generated from the keyboard. (If you are using ENQ/ACK pacing, ^{E}NQ does not show up.)

Displaying functions is useful for debugging communications problems and defining user keys.

36.6 Using HP Memory Lock

Memory lock fixes specified lines of data on the screen and allows you to move blocks of data from location to location within display memory. Memory lock is available in HP emulation from the modes keys ([F6]) and can also be enabled and disabled by an escape sequence from the host computer (see page 458).

The two types of memory lock, display lock and overflow protect, are related to the cursor's location when memory lock is activated.

HP Display Lock

If the cursor is located in any row except the top screen row when memory lock is activated, the rows above the cursor become fixed on the screen. When the screen fills and new lines are added to the bottom, the lines immediately below the locked area roll past the locked rows and off the top of the screen.

This feature is useful for keeping column headings or instructions on the screen during scrolled data entry. It can also be used to move blocks of data (fewer than 24 lines) within display memory:

1. Using [Ctrl]-[↑] and [Ctrl]-[↓] to scroll, position the first line you want to move at the top of the screen.
2. Move the cursor down, one line below the block to be moved.
3. Press [Alt]-[M] to display the modes key labels.
4. Press [F6], **MEMORY LOCK**. The block is now *locked* in position on the screen.
5. Scroll the remaining lines up or down around the locked block until the text is in the desired order.
6. Press [F6] again to disable memory lock.

HP Overflow Protect

If the cursor is in the top row of the screen when memory lock is enabled, Reflection protects the contents of display memory so that no more lines can be added when display memory becomes full. If you try to add more lines, Reflection displays an error message and emits a beep.

36.7

Wraparound

HP In local mode (with the asterisk removed from the **REMOTE MODE** key), type three or four lines of text, but do not press [Return] when you reach the right margin. As you approach the end of a line, the PC automatically *wraps* subsequent characters around to the first column of the next row. You can either enter a *carriage return* (with AUTO LF on) by pressing [Return] or let wraparound take care of linefeeds.

VT To activate the wraparound feature in VT mode, set **End of line wrap** to *YES* on the Terminal Configuration, Page 1 menu.

36.8

Setting Margins and Tabs in HP Emulation

Margins set using the procedure described below are not saved as part of the configuration file. They remain active only until they're changed or until you exit Reflection. Margins can be specified and saved using the **Left margin** and **Right margin** settings on the Terminal Configuration, Page 2 menu.

Setting Margins

1. From the system keys, press [F2], **margins/tabs/col.**
2. To clear previously set margins, press [F7], **CLR ALL MARGINS.**
3. To set the left margin to a point other than column 1, move the cursor to the desired column (using the row/column screen indicator as a guide). Press [F5], **LEFT MARGIN.**

To set the right margin, follow the same procedure, but press [F6], **RIGHT MARGIN.**

Setting Tabs

1. From the system keys, press [F2], **margins/tabs/col.**
2. Move the cursor to the desired column and press [F2], **SET TAB.** Repeat the procedure until all desired tabs are set.

To clear a single tab stop, press [Tab] until the cursor arrives at the tab you want to clear. Then press [F3], **CLEAR TAB.** To clear all tabs, press [F4], **CLR ALL TABS.**

36.9

Setting Tabs in VT Emulation

1. From the VT system keys, press [F4], **modes keys.**
2. Press [F3], **config tabs.**

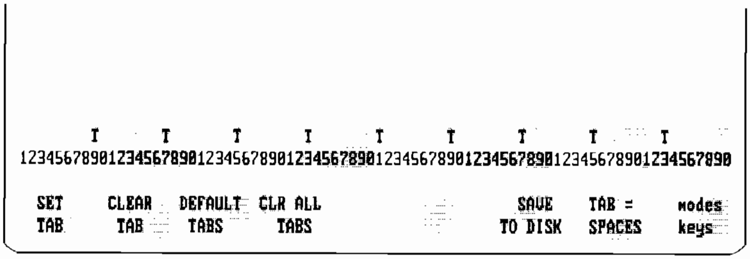


Figure 43: VT Tabs Keys

3. Move the cursor to the desired column and press [F1], **SET TAB.**

In VT mode, margins can be reset only through the modes keys. See page 131.

36.10 Keyboard Locking

The keyboard automatically locks during a block transfer from Reflection to the host. The letter **K** appears between the [F4] and [F5] key labels.

While the keyboard is locked, pressing most keys causes the PC to beep and the keystroke is ignored. Only the following key combinations remain active when the keyboard is locked:

- [Alt]-[R] forces a hard reset.
- [Alt]-[S] forces a soft reset.
- [Alt]-[X] exits Reflection.
- [Ctrl]-[Break] transmits a break signal. (See SET BREAK-LENGTH in the *Command Language Manual* for how to configure the length of the break.)

Refer to pages 189 (HP) and 209 (VT) for explanations of these and other keyboard functions.

36.11

Line Modify and Modify All

When Reflection is in character/remote mode, modify modes allow you to change a line on the screen and retransmit the modified line to the host, saving laborious retyping. These two modes are identical, except that **LINE MODIFY** is disabled after a [Return] or [Enter]; **MODIFY ALL** remains active until disabled ([F2] from the modes key labels).

For example, if you enter a long command line and have the command rejected by the host computer because of a typing mistake, use the following procedure:

1. Press [Alt]-[M] to get to the modes keys.

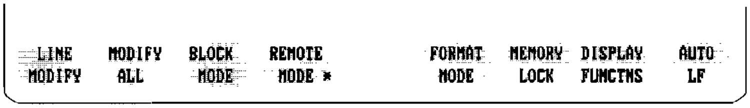


Figure 44: Modes Keys

2. Press [F1], **LINE MODIFY**.
3. Position the cursor to the portion of the line that needs to be corrected.
4. Correct the line by typing over, inserting, and/or deleting characters.
5. Press either [Return] or [Enter]. Reflection retransmits the corrected version of the entire line.

36.12

Using Editing Keystrokes Remotely

Certain keys have no standard ASCII codes to represent their functions, so they're generally not transmitted to the host computer, even though they have an effect on the screen.

Table 26
Non-ASCII Keys and Key Combinations

Single keys	Key combinations
NumLock	Alt - J (clear display)
ScrollLock	Alt - K (clear line)
Home (HP only)	Alt - D (delete line)
↑ (HP only)	Alt - I (insert line)
PgUp (HP only)	Alt - Y (Command mode)
← (HP only)	Ctrl - End
→ (HP only)	Ctrl - ↑
End (HP only)	Ctrl - ↓
↓ (HP only)	Ctrl - ←
PgDn (HP only)	Ctrl - →
Ins	Ctrl - PgUp
Del	Ctrl - PgDn
CapsLock	Ctrl - Home

In HP mode, when the **Transmit functions** field is configured to *YES* on the Terminal Configuration, Page 1 menu, an escape sequence (such as ^{E}*CA*) is automatically sent to the host computer each time a non-ASCII key is pressed. (See page 108 for information on configuring **Transmit functions.**) Some software, such as *HP Slate*, uses this **Transmit functions** setting for full-screen editing in character/remote mode.

Backspace in Remote Mode

Backspace generates a special backspace character (ASCII ^{B}S, decimal 8), which is used by the HP 3000 to edit line input.

Ctrl - Backspace generates a *rubout* character (ASCII DEL, decimal 127). Several host systems, notably VAX/VMS, use DEL rather than ^{B}S to edit line input. Turn on **DISPLAY FUNCTNS** from the modes keys to see which character is generated with Ctrl - Backspace and Backspace. See page 172 for detailed information on the VT Backspace function.

Tabs in Remote Mode

In remote mode, a Tab transmits an ASCII horizontal tab character to the host. Some host programs cannot correctly interpret the tab character. To transmit the appropriate number of spaces to represent a Tab, enable **TAB=SPACES**.

HP Emulation:

1. From the system keys, press F2, **margin/tabs/col.**
2. Press F8 to set tabs equal to spaces (an asterisk appears when enabled).

VT Emulation:

1. Press Alt-M to reach the modes keys.
2. Press F3, **config tabs.**
3. Press F7 to set tabs equal to spaces (an asterisk appears when enabled).

Special Features

Chapter 37

Memory Management

37.1 Memory Allocation

When Reflection is loaded, it determines the amount of memory available in your PC. Included in the memory count required for Reflection are display memory, SHELL memory, printer buffers, graphics requirements, and so on. The rest of your PC's memory is available for running other programs.

Memory Allocation after Background

When you use Reflection's background mode to keep Reflection in memory while running other programs, you are using some of your PC's available memory. If you have insufficient memory to use background mode, pressing the hot-key displays an error message. You may also find that you cannot load your PC application due to memory limitations.

You have several options for controlling how Reflection allocates PC memory:

- Use memory saving switches when initiating Reflection
- Limit your display memory, shell memory, and buffer requirements

If you find that these options do not provide enough memory, use state save (described on page 329) as an alternative to background mode.

Once Reflection has been placed in background mode, you cannot increase or decrease the size of display memory, forms buffers, printer buffers, or SHELL memory. The memory requirements for Reflection are based on the settings specified at the time you first put Reflection into background mode.

37.2

Saving Memory: Startup Switches

Use any combination of the following switches when loading Reflection to make Reflection use less memory. (See page 12 for a summary of all Reflection startup switches.)

Assume CGA or Monochrome: /A

Forces Reflection to assume a CGA or monochrome card is being used. Applies to both Reflection 1 and Reflection 7.

Use Expanded Memory: /E

If your PC has an expanded memory card and a LIM 3.20 compatible expanded memory driver, you can tell Reflection to load program and overlay code into expanded memory using the **/E** switch. Overlays are retrieved from expanded memory (rather than from disk), which is faster.

Suppress Graphics: /G0

If you do not intend to run a graphics application with Reflection 7, you can use **/G0** to save about 100K of memory.

Use IBM font: /I

/I loads Reflection with the default IBM font, saving 4K.

Determine Memory Usage: /MEM

Starting with version 4.0, not all of Reflection's code has to be present in memory at one time. This allows Reflection to offer increased functionality while using less memory. Segments of Reflection—such as file transfer and command language—are loaded on demand. While using less memory, some operations may be slower because Reflection has to read these segments (called *overlays*) off of disk. *

The **/MEM:<x>** switch loads Reflection with a predetermined amount of memory for overlays. The choices are *S* (small), *M* (Medium), or *L* (Large). The default is *M*. There is a trade off between performance and memory usage. The more memory you use, for example, the better command language performance will be. *

/MEM:S

The **/MEM:S** option can affect Reflection's speed. For example, when Reflection is placed in background mode to run a command file, some commands are noticeably slower. The **/MEM:S** switch should save approximately 6K of memory over the medium option.

/MEM:M

This is the default choice.

/MEM:L

The **/MEM:L** option loads all program overlays into memory: this offers the fastest possible performance. If you are using Reflection from a floppy disk, **/MEM:L** is automatically used, since loading overlays from a floppy is too slow.

If you are using a disk caching program, you may want to use the **/MEM:L** switch. Disk cache programs offer big improvements in PC performance, but some may cause problems unless the **/MEM:L** switch is used. Poorly written disk cache programs may disable interrupts for long periods of time, causing receiver overruns (lost characters) to occur. If Reflection is loading overlays off of disk, the disk cache program will be active and you are more likely to lose incoming characters. If this happens, Reflection will print a special character on the screen (Δ or ■) to let you know that an error has occurred. Use the **/MEM:L** switch to prevent this from happening.

Save/Restore Graphics Screen: /S

On an EGA or VGA adapter, this switch saves the foreground screen by caching it in an unused part of the EGA/VGA memory. This reduces Reflection's memory requirements by at least 1.6K.

Suppress Plot to Disk: /T1

/T1 loads Reflection 7 without plot-to-disk capability, saving 4.5K.

37.3 Saving Memory: Configuration Options

Limiting the memory configured for the following items makes more memory available for other applications.

Display Memory

By default, 20K is set aside for Reflection's display memory. Reducing display memory to 8K, the minimum, saves 12K. The size of display memory is configured on the Basic or the Global Configuration menu.

37.4 State Save

State save (Alt-B) lets you exit Reflection temporarily without leaving it resident in memory. If you find that your memory requirements do not allow you to use the hot-key, you can use state save instead. *
*
*

Pressing [Alt]-[B] to exit Reflection saves information about your Reflection session to a file. When you load Reflection again, the state save file is loaded if it is found. For example, you can work on a complex graphics image on the host, use [Alt]-[B] to exit, and run a PC application that has extensive memory requirements. The graphics image is restored when you load Reflection again. * * * * * *

State save has a few restrictions:

Keyboard remapping
: The state save file does not contain all keyboard remapping information. Information that has been saved to a configuration file is available, but temporary keyboard remapping that has not been saved to disk is not saved in the state file.

Open files
: If you are writing to a file when you press [Alt]-[B] (for instance, you are using TRACE, CAPTURE, or LOG), those files are closed. If you want to use the same trace file again, use the APPEND parameter when you re-open it.

When State Save Cannot be Used
: You cannot use the state save keystroke if a menu is displayed or if a command file is running.

Startup Switches
: Most Reflection startup switches are not saved with [Alt]-[B], allowing you to override specific switches when you load Reflection again. Normally, you should load Reflection with the same switches that were used when you last loaded Reflection and exited with [Alt]-[B].

The exceptions to this rule are switches that specify hardware settings:

- /A (CGA or monochrome monitor)
- /G (graphics adapter)
- /I and /C (IBM font)

If these switches are saved to a state file, they cannot be overridden—when you run Reflection again, the switches are in effect. Furthermore, if you use a state save file that does not contain these settings, you cannot use them at startup. Instead, delete the state file or run Reflection using **/SD** (see below).

DISCONNECT-ON-EXIT

If DISCONNECT-ON-EXIT is set to *YES*, it is disregarded when you use state save to exit Reflection: your session is not disconnected.

State save keeps all settings and parameters relating to your host session, a Telnet session, and any application you may be using from within Reflection. If you plan to use both the LAT command interpreter and Reflection's state save feature, the LAT CI must be invoked at the DOS prompt— not installed automatically.

State Save Options

- The default name for the state save file depends on which product you are using. R1.EXE uses R1STATE.STA, and R7.EXE uses R7STATE.STA. You can specify a different filename with a DOS SET command. See page 22. * * *
- By default, the state save file is deleted after running Reflection. Use the **/SK** switch to *keep* a state save file after it has been used. You can use this switch as part of a logon procedure that includes loading a state save file. To avoid having the state save file overwritten by a new one, exit Reflection with Alt-X, not Alt-B.
- If a state save file has been created, it is automatically used the next time Reflection is run. If you do *not* want to use the state save file, start Reflection with **/SD**.

  The state save file is deleted (unless **/SK** is used). For example, if the system manager discontinues all sessions overnight, a state file with session information is not useful.

  If you find that you have inadvertently loaded an old state file, exit Reflection with Alt-X and start again.
- An additional state save switch, **/SP**, is used with Oracle applications. See page 312 for more information.
- You can configure Reflection so that it transmits an XOFF ($^{D}3$) character to the host when state save is invoked with the Alt-B keystroke. Enter **SET XOFF-ON-STATESAVE YES** on the command line. This prevents the host from transmitting any data to the PC while Reflection is not running.

STSAVE and STLOAD are Reflection commands that allow you to load and save a state save file. See the Reflection *Command Language Manual* for more information. * * *

Section 9

File Transfer

You can transfer files in Reflection using the File Transfer screen or command language. Five different protocols are available within Reflection.

The chapters in this section are organized as follows:

- *Using File Transfer* —sending and receiving files using the File Transfer screen
- *Protocols*—Specific information for each protocol is found in a separate chapter:

Protocol	Page Reference
WRQ	See page 347
KERMIT	See page 379
XMODEM	See page 383
FTP	See page 385
OLD-WRQ	See page 387

You may also perform clear text transfers to any host using the TRANSMIT command; however a program such as an editor must be running on the host to receive data. No error-checking is performed with clear text file transfers.

Chapter 38

Using File Transfer

38.1 File Transfer Overview

You can transfer files in Reflection using the file transfer screen or command language.

To get to the File Transfer screen, press [F6], **file transfer**, from the system keys ([F10]).

Using File Transfer

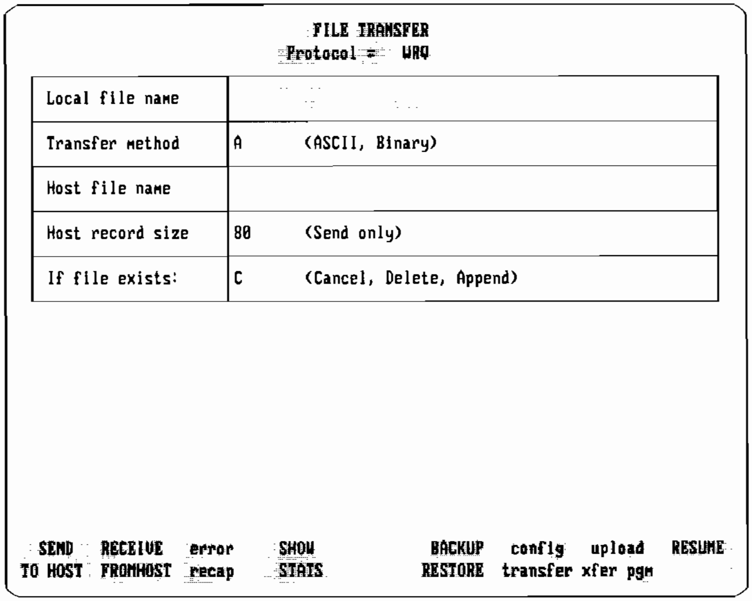


Figure 45: File Transfer Screen

Reflection supports five different file transfer protocols. The one that is currently selected is shown on the second line of the FIle Transfer screen.

File Transfer Host Programs

To perform a file transfer, you must have the appropriate host program available. See the chapter on each protocol for specific information.

Note: Before you can use Reflection's *WRQ* protocol, you must upload one of the following:

To upload PCLINK2.PUB, see page 348.
To upload VAXLINK2.EXE, see page 350.
To create UNXLINK2 on the host, see page 351.

File Transfer Function Keys

The function key labels displayed at the bottom of the File Transfer screen may not apply, depending on the protocol displayed at the top of the screen or your network environment.

Table 27
File Transfer Function Key Labels

Label	Key	Action
SEND TO HOST	F1	Begins transfer to the host. When the protocol is *WRQ* or *OLD-WRQ*, the **Host startup sequence** (defined on the File Transfer Configuration menu) is transmitted to start the host side of the transfer operation.
RECEIVE FROM HOST	F2	Begins transfer to the PC. When the protocol is *WRQ* or *OLD-WRQ*, the **Host startup sequence** is transmitted to start the host side of the transfer operation.
error recap	F3	Displays the Summary of Datacomm Errors screen. This applies to *WRQ* and *OLD-WRQ* in non-network environments.
SHOW STATS	F4	Displays information on the transfer, including errors.
BACKUP RESTORE	F5	Steps through a backup or restore operation for the PLUS version of Reflection (*WRQ* and *OLD-WRQ* protocols only). In non-PLUS versions, the label is blank.
config transfer	F6	Displays the File Transfer Configuration menu where you can select and configure a protocol.
upload xfer pgm	F7	Uploads the host program from the PC to the host (*WRQ* protocol). Does not apply to UNIX systems.
RESUME	F8	Exits the File Transfer screen and redisplays the system keys.

38.2 Reflection Protocol Options

Use the following table to confirm which fields on the File Transfer screen apply to each protocol. A description of each field follows.

Table 28
Protocol Summary

	WRQ	KERMIT	XMODEM	FTP	OLD-WRQ
Local filename	yes	yes	yes	yes	yes
Transfer method	A, B	A, B	B	A, B	A, B
Host filename	yes	receive only	receive only	yes	yes
Host record size	yes	no	no	no	yes
If file exists					
Cancel	yes	receive only	receive only	receive only	yes
Delete	yes	receive only	receive only	receive only	yes
Append	yes	no	no	no	receive only

38.3 File Transfer Screen Fields

Before starting a file transfer, confirm that the protocol displayed at the top of the screen is correct. *WRQ* is Reflection's file transfer protocol, and is the default. Press F6, **config transfer**, to see the configuration information associated with the displayed protocol or to change the protocol.

Local File Name

The local PC filename, whether *source* or *destination*, must satisfy DOS syntax rules. You may include a drive specifier, path name, and extension.The local filename is optional if you are receiving a file or files from the host.

Using the *WRQ* protocol, wildcards can specify a group of PC files to be sent to the host. See page 359.

Transfer Method

This field has a default value of *ASCII*, represented by an **A**. Type a **B** to change the value to *Binary*.

ASCII

If you want to simply transfer a text file from a PC to a host computer, or vice versa, then the ASCII method is correct. The file is readable on the system to which it has been moved.*

Binary

If your goal is to transmit a file between two PCs via a host computer, use the binary method. Binary is also often appropriate for transfer between two hosts via a PC.

Host File Name

The host filename, whether source or destination, must satisfy host syntax rules. Using the *WRQ* protocol, host wildcard characters can specify a group of host files to be received. The host filename is optional if you are sending a file or files to the host.

With both the *WRQ* and *OLD-WRQ* protocol, you can add file specifications to the host filename. See page 353 (HP) or pages 357 and 391 (VAX).

Host Record Size (Send only)

This field applies to the *WRQ* and *OLD-WRQ* protocols only. Its default value *
changes according to the file transfer protocol being used. *

If File Exists

The *Cancel* and *Delete* options are available for all protocols when receiving a file on the PC.

C (Cancel)

If this field is set to *C* (the default), the file transfer is cancelled when a file with the specified name already exists. If the file is part of a wildcard set, no other files are transferred after a duplicate is found.

When sending to a VAX, a file transfer is cancelled only if an existing file version number is also specified. Otherwise, a new file with the same name is created with a higher version number.

---

* Some PC word processing packages contain ASCII control characters. If you are sending such a file to another PC, use the binary method. If you want to use the file on the host, see *Strip 8th bit on text files* on page 156.

D (Delete) Set this field to *D* to replace the existing file with the file being transferred.

When sending a file to the VAX, all existing versions of the file on the host are purged; the new file has a version number one higher than the last version of the file.

A (Append) To append a file to an existing file, set this field to *A*. If no such file exists, it is created. This option is available for transfers to the host using the *WRQ* protocol only. You can use the *A* (Append) option with both *WRQ* and *OLD-WRQ* when receiving a file on the PC.

When sending a file to the VAX, the PC file is appended to the VAX file with the highest version number.

38.4 Transferring a File

The file transfer process is similar regardless of the protocol used or the direction of the transfer:

1. Log on to the host computer before performing a file transfer. Be sure to log on to an environment that allows you to read, write, and save files. All Reflection file transfers can be initiated from the File Transfer screen.
2. After confirming that the protocol and the associated configuration information are correct, complete the File Transfer screen.
3. Press F1 to send a file or F2 to receive a file. You can monitor the file transfer progress by pressing F4, **SHOW STATS** (explained on page 342). When the transfer ends, you can select F3, **error recap**, to review any errors that may have occurred.

If you experience difficulties, refer to *Troubleshooting File Transfer* beginning on page 75.

Starting the File Transfer

Check all the fields on the screen before starting the transfer; some settings from a previous file transfer may still be displayed. Press F1, **SEND TO HOST**, or F2, **RECEIVE FROM HOST**. With *WRQ* and *OLD-WRQ* protocols, pressing one of these keys transmits the host startup sequence to the host computer, initiating the program on the host that performs the file transfer. For other protocols, you must initiate the host program before starting the transfer.

When Reflection and the host program start interacting successfully, Reflection displays the message *Transfer in progress.* If the file transfer cannot be done for some reason, the message may say *No host response* or something about the file being transferred (e.g., *Local file does not exist*).

Monitoring the Transfer

When the transfer begins, the screen displays additional information about the transfer in progress.

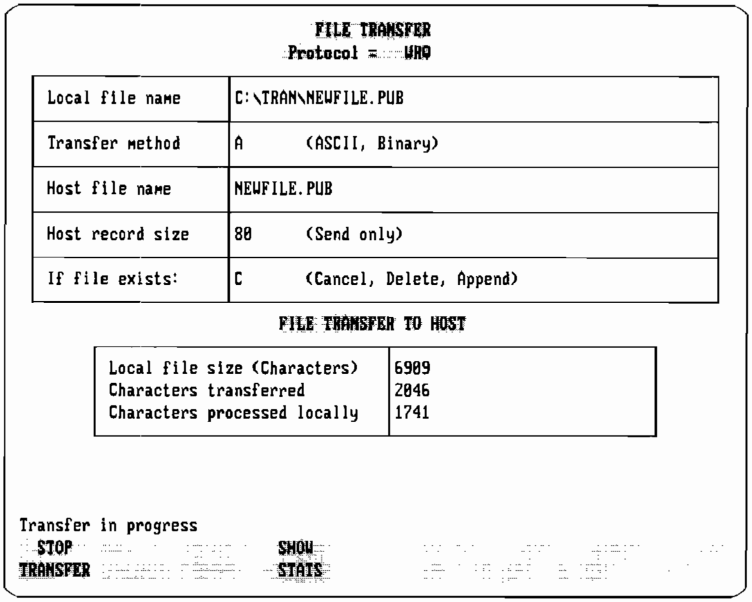


Figure 46: Monitoring the File Transfer

Sending

Local file size (Characters)

When sending a file to the host, the local file size (in characters) is displayed by Reflection when the transfer begins.

Characters transferred

This field shows the number of characters transferred to the host. The number is updated by Reflection as the transfer progresses. Due to encoding and compression, this number does not equal the number of characters processed locally.

Characters processed locally

This field shows how many characters have been read from the local file and successfully sent.

Receiving

Host file size (Records/Blocks)

This field displays the host file size (in characters) when receiving a file from the host.

Characters received

This field shows the number of characters transferred from the host. It is updated by Reflection as the transfer progresses.

Records/Blocks received

This field indicates the number of characters actually written to the local file from the host file.

At the end of a successful transfer, the **Host file size** and **Records/Blocks received** numbers match.

Stopping the File Transfer

You can stop a transfer by pressing F1, **STOP TRANSFER**. The transfer is stopped and the message *Terminated by user* is displayed. Wait for the transfer to time out. The F5 key label may read **ABORT TRANSFER** after a few seconds. The abort function is provided only for instances when you do not regain control within a few seconds. The transfer program on the host may continue running after the abort: exit the File Transfer screen and press Ctrl-Y twice to stop it.

38.5 File Transfer Statistics

Press F4, **SHOW STATS**, before or during a transfer to see file transfer statistics.

```
                          FILE TRANSFER
                         Protocol = WRQ
```

Local file name	C:\TRAN\NEWFILE.PUB
Transfer method	A (ASCII, Binary)
Host file name	NEWFILE.PUB
Host record size	80 (Send only)
If file exists:	C (Cancel, Delete, Append)

```
Blocks Received 0   Receive Naks 0   Blocks Xmitted 0   Xmit Naks 0
       Timeouts 0    Retransmits 0        Line Hits 0   00:00:00.00

  SEND    RECEIVE   error   SHOW        BACKUP   config   upload  RESUME
TO HOST  FROMHOST   recap   STATS*      RESTORE  transfer xfer pgm
```

Figure 47: File Transfer with Stats

Table 29
File Transfer Statistics

Blocks Received	The number of good data blocks received by the PC since the transfer began.
Timeouts	The number of times the transfer program was unable to obtain a host response within the limits set by the **Receive timeout (seconds)** field on the File Transfer Configuration menu.
Receive Naks	The number of received data blocks that have been rejected (Nak stands for *Not Acknowledged*)
Retransmits	The number of blocks that have been retransmitted because of naks or timeouts.

Blocks Xmitted	The number of data blocks transmitted by the PC.
Line Hits	The number of times that framing or receiver overrun errors occurred.
Xmit Naks	The number of blocks transmitted by the PC that have been rejected by the host.
00:00:00.00	The elapsed time since the transfer began expressed in hours, minutes, seconds, and hundredths of a second.

If errors are detected, you can get additional information from the Summary of Datacomm Errors screen: press F3, **error recap**, at the File Transfer screen. This error recap applies to WRQ and OLD-WRQ protocols when used in a non-network environment. See the discussion on page 71.

38.6 The BACKUP and RESTORE Option

If you have the PLUS version of Reflection, F5 on the File Transfer screen reads **BACKUP/RESTORE**. This function key invokes a command file called WRQBACK.RCL, which walks you through backup and restore operations. This option is supported by the *WRQ* and *OLD-WRQ* protocols only. See the *PLUS Manual* for details.

38.7 Transferring Files to a PC Printer

The PC printer's device name, usually PRN, can be used as the **Local file name** when receiving a file. As the file is transferred it is immediately directed to the printer.

38.8 Command Language

Another method for transferring files is to use Reflection's command language. These commands allow you to provide additional detail about the file transfer. For example, the SEND and RECEIVE commands used with the WRQ protocol let you transfer a set of files created before a certain date. See the Reflection *Command Language Manual* for information on the commands listed on the following page.

Protocol	Commands
WRQ	SEND, RECEIVE, SERVER, FINISH
OLD-WRQ	SEND, RECEIVE *
KERMIT	KSEND, KRECEIVE, KBYE, KFINISH, KGET
XMODEM	XSEND, XRECEIVE
FTP	FSEND, FRECEIVE, FCD, FDIR, FGET
	FLS, FMGET
(n/a)	ISEND, IRECEIVE (IBM protocol)

* The Reflection *Command Language Manual* documents the syntax for SEND and RECEIVE separately according to protocol.

Chapter 39

WRQ Protocol

Walker Richer & Quinn provides host programs for Reflection's *WRQ* protocol on the HP 3000, VAX/VMS, and UNIX systems. A single copy of a host program can support all users of Reflection.

The protocol is supported by the following host file transfer programs:

- PCLINK2 for HP 3000s
- VAXLINK2 for VAX/VMS
- UNXLINK2 for UNIX-based systems

A general description of the File Transfer screen is found on page 335. If you are upgrading from a pre-4.0 version of Reflection, also read *OLD-WRQ Protocol* on page 387.

The *WRQ* protocol is the default protocol for file transfer, and should be used in most circumstances. It has the following advantages:

- CRC-CCITT error checking is used to ensure error-free transmission. Using this method, blocks are retransmitted, if necessary, to ensure accurate transmission.
- Wildcard transfers can be performed in either direction.
- Files can contain information other than ASCII characters.
- When you transfer a file from the host, all the file parameters of the host can be kept on the PC copy. This information can later be used to create an exact copy of the original file on the same or another host.

- *WRQ* protocol works in a similar fashion for the HP 3000, DEC VAX, and UNIX-based systems.
- Data compression techniques produce faster transfers.
- Files can be sent from the host to a PC printer and from a PC to a host printer.
- Using the *WRQ* protocol with Reflection's command language offers even more options than using it from the menu. For example, you can limit selection for wildcard file transfers (BEFORE, SINCE, EXCLUDE).

See the Reflection *Command Language Manual* for the full syntax of the SEND and RECEIVE commands.

39.1 Uploading the Host Program

Before you can use the *WRQ* protocol for file transfer, you must upload the appropriate host program to your host system. The process is automated using Reflection's command language.

Depending on the type of host you are transferring files to, the process for getting the host program on the host varies. See the page references below for uploading each type of host:

Uploading PCLINK2 to an HP 3000	See below
Uploading VAXLINK2 to a VAX	See page 350
Uploading UNXLINK2 to a UNIX System	See page 351

Uploading PCLINK2 to an HP 3000

One of the Reflection program diskettes contains a program file called PCLINK2.PUB. This file must be copied to the HP 3000 before performing file transfers using the *WRQ* protocol. You must have write access on the PUB.SYS account in order to upload PCLINK2 under the default program name. The Reflection diskettes include four files that upload PCLINK2 for you. You only have to do this once for each HP 3000.

To upload PCLINK2, do the following:

1. Make sure that the following files are available in the current directory or on a diskette in the current drive. *
*

- PCLINK2.PUB *
- WRQUPLOA *
- UPLOADHP.RCL *
- VERIFYPH *

2. Start Reflection by typing **R1** or **R7**.
3. Log on to the HP 3000. Make sure that you are signed on with a term-type equal to *10* (the normal value) by adding the following to your logon procedure:

```
HELLO <logon>;TERM=10
```

4. Confirm that **Parity** on the Basic Configuration menu is set to *NONE* or *0's*.
5. From the system keys, press [F6], **file transfer**.
6. At the File Transfer screen, press [F7], **upload xfer pgm**. The following screen is displayed:

```
            START UPLOAD HOST FILE TRANSFER PROGRAM TO HP

The following will help ensure the success of the of the UPLOAD procedure:
1.   Configure your communications port with parity set to NONE or
     0's.
2.   If your host is an HP3000, make sure that you are signed on with
     with a term-type equal to '10' (normal value). You can force this
     to be true by signing on with 'HELLO <logon>;TERM=10'.
3.   If UPLOAD is unsuccessful, then before re-trying ensure that no
     host program is still executing. This can be done by typing :
                    <break> <esc> ':' 'ABORT' <cr>

 Enter target program name [PCLINK2.PUB.SYS]:
```

Figure 48: Upload Host File Transfer Screen

7. You have the option of using the default name, PCLINK2.PUB.SYS, or you can use another name. You are notified if a file by that name already exists. Do *not* use the name *PCLINK*; see page 393. You can back out of the upload process if you do not want to continue.

   Note: If you try to upload to a group that does not have PH (process handling) capability, you are notified that PCLINK2 will not support wildcards when receiving files. It is best to upload PCLINK2 to a group that has PH capability. If you do not have access to such a group, ask your system manager to perform the upload. *

8. The message *Uploading 1st copy of WRQUPLOA* appears and a record count shows the progress of the upload. The same procedure is repeated when *Uploading 2nd copy of WRQUPLOA* appears. Next a message indicates that the two copies are being compared. (This process ensures that no errors occur in the upload.)
9. The File Transfer screen appears and the upload begins. You are notified when the transfer is complete.

If you have problems, see *Troubleshooting the PCLINK2 Upload* on page 75.

Uploading VAXLINK2 to a VAX

The Reflection diskettes contain three files that upload VAXLINK2 for you. You only have to do this once for each VAX/VMS if you make VAXLINK2 a foreign command—see page 366 for instructions on how to do so. You must have VMS 5.x or greater to use the VAXLINK2 host program.* *

To upload VAXLINK2, do the following:

1. Make sure the following files are available in the current directory, or insert the support diskette in the current drive.
   - VAXLINK2.EXE
   - WRQUPLOA.MAR
   - UPLOADVX.RCL
2. Start Reflection's VT emulation by running R1V.EXE or R7V.EXE. You can also change to VT emulation by changing the terminal type to *DEC* on the Global Configuration menu.

---

* If you are using a pre-5.x version of VAX/VMS, you may need a copy of Walker Richer & Quinn's earlier host file transfer program (VAXLINK). Contact our technical support.

3. Log in to the VAX.
4. Press [F6] from the system keys to display the file transfer keys.
5. Press [F7], **upload xfer pgm**.

The command file UPLOADVX.RCL starts the upload. If a copy of VAXLINK2 is already installed on the host, you are given an opportunity to stop the upload. Uploading takes about eleven minutes at 9600 baud. With a modem transfer you must have a clear connection, i.e., without a lot of noise on the line. After it is finished, you should see *Successful Transfer* displayed on the screen. You will then find a file called VAXLINK2.EXE in your current directory on the VAX.

Uploading UNXLINK2 to a UNIX System

The upload for UNIX systems is controlled by a command file called UPLOADUX.RCL. This file describes the upload process, sets some configuration values, uploads, and then compiles the host files.

To upload UNXLINK2, do the following:

1. Make sure the following files are available in the current directory or insert the support diskette in the current drive.
   - UNXLINK2.C
   - UNXXFER2.C
   - UPLOADUX.RCL
   - WRQKERM.C
2. Load Reflection; if you are using the VT emulation, type **R1V** or **R7V**. Otherwise, type **R1** or **R7**.
3. Bring up the command line ([Alt]-[Y]) and type **UPLOADUX.RCL**.
4. A series of prompts walks you through the upload process. Some of the configuration values for UNIX connections are set at this time.
5. After the upload, verify that the following Reflection values are set:

   Datacomm Configuration menu

Datacomm port	COM1 to COM8 (5-8 PS/2s only)
Receive pacing	XON/XOFF
Transmit pacing	XON/XOFF

File Transfer Configuration menu

Host startup sequence*	unxlink2 (lowercase!)	
Packet size	64-256	
Expand embedded tabs	NO	
Change spaces to tabs	NO	

Terminal Configuration menus

Inhibit handshake (Page 1)	YES	HP only
Inhibit DC2 (Page 1)	YES	HP only
Use host prompt (Page 2)	NO	(Use SET command in VT emulation)
Character xmit delay (Page 2)	0-3	

Additional UNIX Settings

With many UNIX systems you may experience difficulties transferring files to the host unless you set some additional parameters. In general, UNIX systems cannot always process large blocks of data when they are received at higher speeds.

For some systems, changing the packet size helps. For example, if you are connected directly to the host at 9600 baud, bring up the command line (Alt-Y) and type the following:

```
SET TRANSFER-WINDOW-SIZE 2
SET TRANSFER-FRAME-SIZE 128
```

It might also be necessary to add a character transmit delay if the speed of the transfer is too high for the host to handle (9600 baud or more). Change this on the Terminal Configuration, Page 2 menu.

Experiment until you find the most efficient settings for file transfers with your particular UNIX host.

If a UNIX system is your primary host, you may want to save these values to the default configuration file (R1.CFG or R7.CFG). Otherwise, save them to a configuration file such as UNIX.CFG; it can then be loaded with Reflection (type **R1 UNIX.CFG** at the DOS prompt), or from within Reflection (type **LOAD UNIX.CFG** at the Reflection command line).

---

* The host startup sequence is entered for you during the upload process.

Note: Some systems with job control capabilities use ^Z as a *suspend* character. This is handled appropriately under the Reflection default for UNIX systems. If you or your system manager has redefined this keystroke, you may need to make corresponding changes to the transfer procedure. See page 372, *UNIX Transfer*.

39.2 File Transfer with WRQ Protocol

In order to use the *WRQ* protocol for file transfer, **Protocol = WRQ** should be displayed at the top of the File Transfer screen.

If any other protocol is displayed, press [F6], **config transfer**. Change the **File transfer protocol** field to *WRQ*, then activate or save the change. The basic file transfer process and the settings for the fields on the File Transfer screen are described on page 348.

Some of the fields on the File Transfer screen apply to the *WRQ* and *OLD-WRQ* protocols only, and HP emulation and VT emulation defaults also differ. The File Transfer screen defaults specific to the *WRQ* protocol are summarized below for both the VT and HP emulation. See the Reflection *Command Language Manual* for a description of the UNIX attributes that are supported with the *WRQ* protocol. See page 364 for a complete description of all of the File Transfer Configuration menu fields.

HP File Transfer

When transferring to or from an HP 3000, use the guidelines below to fill in the fields on the File Transfer screen.

Transfer Method

When performing an ASCII transfer to an HP, the host defaults to an ASCII MPE file with fixed-length records. The specified record size is used as the actual record size.

With a binary transfer to an HP, the resulting host file is a binary MPE file with variable-length records (unless **;F**, the fixed format switch, is used— see page 354). The specified record size is used as a maximum size for the records.

Host File Name

You can append the switches listed below to the host filename. (For information on switches specific to QEDIT and Precision Architecture transfers, see pages 305 and 302.)

Purge Existing HP 3000 Files (;P)
When you select *D* (Delete) in the **If file exists** field, the characteristics of the existing file are inherited; an END OF FILE error message occurs if the file being transferred is too large for the original file limit.

To avoid this problem, purge the host file as part of the transfer to the host. Append **;P** to the end of the host filename; e.g., **TEXT.DOC;P**. If you still have problems, see the file equation information on page 356.

Note: The **;P** switch overrides the setting of the **If file exists** field.

Include Label Information (;L)
You can include a **;L** switch at the end of the host filename. When receiving a host file, this switch instructs the host to include HP 3000-specific file information in the data transmitted to the PC. The HP file information is encoded in an ASCII format in the PC file. If you want to transfer a file between two HP 3000 systems via Reflection, use the binary method and append **;L** to the host filename.

When sending a file to the host, this switch instructs the host program to expect HP 3000-specific file system information in the data transmitted from the PC. This option can only be used if the file was first received on the PC using **;L**. This switch forces the inclusion of *user* labels on HP 3000 files. Files such as message catalogs can be transferred with no information loss.

If the file already exists on the host and is smaller than the file being transferred, the **;P** switch must also be included.

The number displayed in the **Host record size** field is ignored; the file attributes portion of the data contains record size information.

You can also use the **;L** switch to transfer COBOL files to your PC. Make sure that **Change spaces to tabs** is set to *NO* on the File Transfer Configuration menu so that columns are maintained. When transferring the file back to the host, include the **;P** and **;L** switches at the end of the host filename.

Fixed-Length File Transfers (;F)

Reflection allows you to specify fixed-length file transfers when using the binary file transfer method. Add **;F** to the host filename to specify this option. (The fixed-length method allows you to send graphics data files to the HP 3000 from files which should be fixed-length after transferring, such as *Charting Gallery* files.)*

This switch sends *Memo Maker* files, for example, to the HP 3000 with a record length of 80 bytes.

Transferring variable-length files to the HP 3000 means that PC files transferred to the host and back to a PC are identical. Fixed-length transfers are padded with nulls when returned to the PC.

Lockwords

If a file on the host has a lockword, you must add it to the host filename before performing a transfer to the PC:

```
<filename>/<lockword>
```

If the PC file is sent back to the host using the original filename and the *Delete* option is selected in the **If file exists** field, the lockword must be specified in the host filename.

You can create a lockword file when transferring a PC file to the host by adding the lockword to the host filename as described above.

Host Record Size

When transferring to an HP 3000, this field identifies the record size expected by the host program that will use the file.

Values: 1–32767
Default: 80

(If you are in doubt as to the size, check other host files used by the same program.) The number is used as an upper limit for the amount of data to be placed in records of an existing file. The host record size is ignored if the ;L switch is used (see page 354).

* AdvanceLink *always* creates fixed-length files on the HP 3000. Reflection defaults to variable length files, but gives you the option of either fixed or variable formats in binary transfers.

If File Exists

When a host file is first created, an estimated file size is assigned. A generous estimate for the file size is initially given: when the transfer is complete, unused disc space is released. There are two cases where you may need to use a file equation to ensure that the file will be large enough:

- When creating a new host file, a PC source file with many blank lines or short lines may exceed the estimate, and the transfer aborts. If this occurs, purge the created file on the host and then use a file equation at the MPE prompt to define a larger file. For example:

```
:FILE SAMPLE;DISC=10000;NEW
```

- A file transfer to an HP 3000 using the *A* (Append) option in the **If file exists** field assumes the existing host file limit is large enough to handle the resulting increase.

  If you find that you cannot append to an existing host file, create a larger file with another name using a file equation similar to the example above. You can then include both files into the larger file. (Using the same filename requires a purge before the new file limit can take effect.) A file that is appended inherits the file specifications of the first file.

VAX/VMS File Transfer

When transferring to or from a VAX host, use the guidelines below to fill in the fields on the File Transfer screen.

Transfer Method

When sending to a VAX using ASCII transfer, the host creates an ASCII file with variable-length records.

If you use the binary method when sending files to the host, the resulting host file is a binary file with fixed-length records of the size specified in the **Host record size** field. (This differs from binary transfers to the VAX using *OLD-WRQ*; see page 391.) If the default record size of *512* is not changed, the record size is ignored. If the default is changed (to either a lower or higher number), the file transfer program uses the specified size.

Host File Name

The host filename must satisfy VAX file system rules—not DOS syntax. The host filename can include additional file disposition and creation information (called host file *qualifiers*):*

/C — DCL Batch Files

ASCII file transfer only. The file on the VAX is submitted as a batch DCL file upon completion of the transfer. This is equivalent to the SUBMIT parameter of the SEND command. Note that files are deleted after they have been submitted; see /K below.

/D — DECDx Format Files

Applies to binary file transfer only. /D transfers files in DECDx format between the VAX and PC. This is equivalent to the COUNTED parameter of the SEND command.

/I — Image Transfer

This qualifier applies to binary file transfer only. Image transfer ensures that the file will be usable by VAX applications when moving a file from the VAX to a PC and back again. Because all existing attributes are already set in an image file, this qualifier cannot be combined with other qualifiers. An indexed file, an .OBJ file, and an .MAI file, for example, are all recreated correctly by using the /I qualifier to append file directory information to the file on the PC and then using this information when recreating the file on the VAX. This is equivalent to the IMAGE parameter of the SEND command.

/K — Keep Submitted Files

This qualifier is valid only in conjunction with the /C and /S switches, which submit files to a queue on the host. Once the files are in the queue, they are deleted. To keep a copy of the file on the host, use the /K (*Keep*) switch. The SEND command parameter equivalent for this qualifier is KEEP. * * * * *

/S — Spooler Files

ASCII file transfer only. The file or files on the VAX are submitted to the spooler for printing upon completion of the

---

* Previous versions of Reflection supported /F (fixed) and /W (WordPerfect) qualifiers. These qualifiers are no longer necessary for the *WRQ* protocol; the default for binary files is now fixed format and this is appropriate for WordPerfect. See page 388 for additional qualifiers used with *OLD-WRQ* protocol.

transfer. This is equivalent to the PRINT parameter of the SEND command. Note that files are deleted after they have been submitted; see /K above.

/V — Variable Host Record Size

Binary file transfer only. Transfers files using a variable host record size. The SEND command equivalent for this qualifier is USING "/FORMAT=VARIABLE".

Host Record Size

When transferring to a VAX/VMS, this field specifies a binary file's maximum record length.

Values: 1–32767
Default: 512

An uploaded binary file is a sequential file with fixed-length records. An uploaded ASCII file has variable-length records.

If the default record size of *512* for ASCII transfers is not changed, the record size is assumed to be long enough for the longest record in the file. If the default is changed (to either a lower or higher number), the file transfer program uses the specified size.

If File Exists

If you select *C* (Cancel), a file transfer to the VAX is cancelled only if an existing file version number is also specified. Otherwise, a new file with the same name is created, but with a version number one greater than the highest version number of the existing file. For example, if you are transferring *FILE.DAT* to the VAX, and FILE.DAT;1 and FILE.DAT;2 currently exist, the new file is named *FILE.DAT;3*. This is true only of transfers *to* the VAX, never of transfers *from* the VAX to the PC.

When transferring to the VAX using *D* (Delete), all other versions of the file on the host are purged.

When transferring to the VAX using *A* (Append), the PC file is appended to the VAX file with the highest version number.

39.3

Wildcard File Transfer

The *WRQ* protocol supports wildcard transfers to and from the host. The source files can be identified with wildcards; however, the target files will automatically have the same name as the source files—no filename is allowed. Directory or group information can be specified for target files.

After you have filled in the fields on the File Transfer screen and start the transfer, the **Local file name** and **Host file name** fields display the name of each file as it is being transferred. If you have the *C* (Cancel) option selected or there is some other error that stops the transfer, the remaining files that match the wildcard specification are not sent.

Note that all files transferred using wildcards use the specified values for **Transfer method, Host record size,** and **If file exists.** ASCII and binary files should be transferred using separate operations.

Sending Files with Wildcards

Use the standard DOS wildcard specification to identify a file. Both * and ? wildcards are supported. In the following example, all files in the EDIT subdirectory with the extension .DOC are sent to the host.

```
                         FILE TRANSFER
                        Protocol =  WRQ

 Local file name      \EDIT\*.DOC
 Transfer method      A       (ASCII, Binary)
 Host file name
 Host record size     80      (Send only)
 If file exists:      C       (Cancel, Delete, Append)

 SEND     RECEIVE   error   SHOW        BACKUP   config   upload   RESUME
 TO HOST  FROMHOST  recap   STATS       RESTORE  transfer xfer pgm
```

Figure 49: Sending Files with Wildcards

The host filename should be left blank or specify HP group (or VAX directory) information only; the PC filenames will be copied as host filenames. If the host filename field is left blank, the current group on an HP or the current directory on a VAX is assumed.

Sending Files to an HP

File naming conventions in DOS and on the HP 3000 differ. In particular, DOS allows files with multiple extensions to be located in the same subdirectory. With wildcard transfer to the host, the DOS *extension* is interpreted as an HP *group* name.

When sending files to the HP 3000, it's a good idea to preview the files that match the wildcard specification—use the DIR command at the Reflection command line.

The examples below summarize these differences and provide some hints on how to perform wildcard transfer to the HP 3000:

Example 1: Using a Wildcard Filename

The PC files you want to transfer:

```
NEWFILE.DOC
OLDFILE.DOC
ARCFILE.DOC
```

Specify ***.DOC** in the **Local file name** field and **.PUB** in the **Host file name** field.

The resulting HP files are:

```
NEWFILE.PUB
OLDFILE.PUB
ARCFILE.PUB
```

Note that all files matching the extension .DOC in the specified subdirectory will be transferred.

Example 2: Using a Wildcard Extension

Don't use wildcard extensions when sending to an HP 3000 unless you have access to a corresponding group on the HP for every DOS extension.

The PC files you want to transfer:

```
NEWFILE.DOC
NEWFILE.ASM
NEWFILE.SAV
```

If you enter **NEWFILE.*** in the **Local file name** field and leave the host filename blank, you must have access to three different groups on the HP—DOC, ASM, and SAV.

However, if you enter **.PUB** in the **Host file name** field (and you have access to that group), you will receive only one file— NEWFILE.PUB, a copy of whatever file was transferred last.

In such cases, you may want to rename or copy your files so that the extension matches a valid HP group name.

Example 3: Using the Default Group

If your files have no extension, the current group name is assumed as the destination.

The files you want to transfer:

```
MYNEW
MYOLD
MYARC
```

If you use the wildcard string **MY*.** in the **Local file name** field, leave the **Host file name** field blank and you are in the ACT group, the resulting files are:

```
MYNEW.ACT
MYOLD.ACT
MYARC.ACT
```

Sending Files to a VAX

Basic VAX wildcard syntax is similar to DOS wildcard syntax. Assume you want to transfer the following files:

```
NEWFILE.DOC
NEWFILE.ASM
NEWFILE.SAV
```

You can type **NEWFILE.*** in the **Local file name** field. The three files transfer to the VAX with unique extensions into the default directory. See page 358 for information on the **If file exists** field.

Receiving Files with Wildcards

To receive a set of files from the host, you may specify a PC subdirectory in the **Local file name** field. The PC filenames will be the same as the host filenames. The most recent version of the VAX file is transferred.

Wildcards provide easy manipulation of multiple files; however it's a good idea to preview the files you want to receive before starting the transfer. Before receiving the files, use the LISTF utility on the HP or the DIR command on the VAX to confirm the files that you will be receiving. There may be many files that meet your wildcard specification that you may not want to receive. (You can also use the RECEIVE command with wildcards and exclude given files; see your *Command Language Manual.*)

In the example below, a set of HP host files matching the wildcard string TR@.PUB are received as PC files. The **Local file name** field contains drive (C:) and directory designation (\TRAN\ ).

For VT emulation, use standard VAX/VMS wildcard syntax.

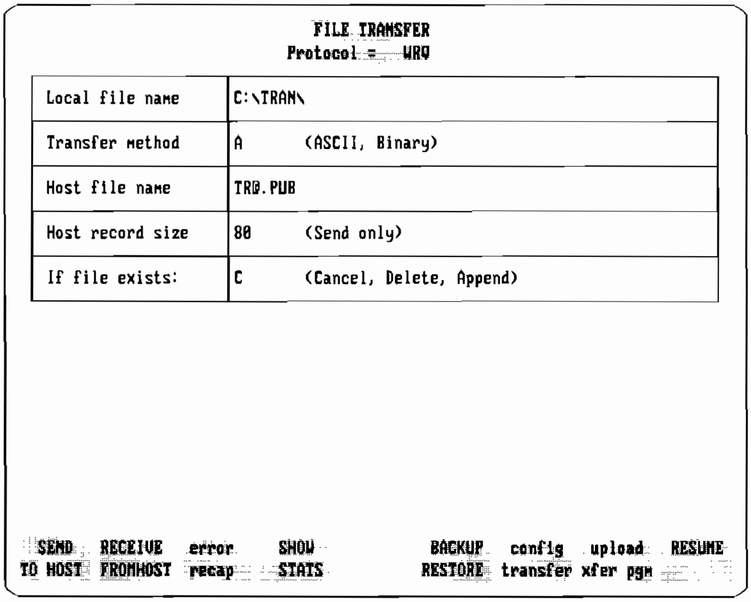


Figure 50: Receiving Files with Wildcards

When the **Local file name** field is left blank, the files are transferred to the current directory. If you do not specify a group name in the **Host file name** field, no the PC files will have no extension. If you specify the group in the **Host file name** field (for example, .PUB), the resulting PC files will have the extension .PUB.

If the group on the HP or the extension on the VAX is longer than three characters, only the first three characters are used to produce a DOS extension. For example, if the HP group is PROD, the PC files will have the extension .PRO.

Note: If you have used the hot-key to put Reflection into background mode, the current directory as defined by Reflection can differ from the current DOS directory. Type **CD** from the command line to verify the current Reflection directory. See page 269 for more information.

39.4

Configuring WRQ Protocol

The File Transfer Configuration menu settings for the *WRQ* protocol are described below.

Depending on your situation, you may configure file transfer only once or each time you transfer files. The menu is the same for HP and VT emulations, but the defaults differ slightly. To get to the File Transfer screen, press F6, **config transfer**. The screen below displays file transfer configuration defaults for the *WRQ* protocol on an HP 3000.

FILE TRANSFER CONFIGURATION

File transfer protocol		WRQ	
Host startup sequence		RUN PCLINK2.PUB.SYS	
Receive timeout (seconds)	15	Error retry limit	10
Packet size	1280	File transfer link	8-BIT

ASCII TRANSLATION TO HOST

CR/LF = Record separator	YES
Expand embedded tabs	YES
Use ctrl-Z as EOF	YES
Change Roman-8 to ISO-7	NO

ASCII TRANSLATION FROM HOST

Record separator = CR/LF	YES
Change spaces to tabs	NO
Write ctrl-Z EOF	YES
Delete trailing spaces	YES
Change ISO-7 to Roman-8	NO

NEXT CHOICE | PREVIOUS CHOICE | DEFAULT VALUES | ACTIVE VALUES | ACTIVATE CONFIG | SAVE TO DISK | DISPLAY FUNCTNS | exit config

Figure 51: File Transfer Configuration Menu

File Transfer Protocol

When you select *WRQ*, the current values associated with that protocol are displayed. The other protocol choices are discussed in later chapters. * *

Reflection command language equivalent:

```
SET TRANSFER-PROTOCOL <value>
```

Host Startup Sequence

Each time you start a file transfer using the *WRQ* protocol, Reflection transmits a host startup sequence, as entered in this field, to the host computer. This is a command to the host telling it to run the host transfer program, using the name you assigned when you uploaded the program.

Value: A string of up to 31 characters
Default: RUN PCLINK2.PUB.SYS (HP mode)
RUN VAXLINK2 (VT mode)
unxlink2 (UNIX)

Reflection command language equivalent:

```
SET HOST-STARTUP <value>
SET HOST-START-WRQ <value>
```

You can save separate host startup sequences for the *WRQ* and *OLD-WRQ* protocols in the same configuration file. See page 387 for more information.

PCLINK2

Some of the default values that govern *WRQ* protocol file transfers to an HP 3000 can be altered by appending parameters to the **Host startup sequence** (described below) or by changing the **File transfer link** field (described on page 369).

Optional parameters for the startup sequence include:

- **QEDIT:** Files with label information similar to QEDIT's are received as if they are QEDIT files. Add **,NQ** to the host startup sequence if you do not want to retain this label information when receiving files. **;Q** should be added to the host filename to maintain QEDIT information when a file is sent to the host. See page 305 for more information.
- **Timeout:** Use **PARM=<value>** at the end of the host startup sequence to change the default timeout value for PCLINK2 (30 seconds). Bits 0-12 (these bits are counted left to right) are used to indicate the timeout interval in seconds. Bits 13-15 are reserved. For example, to set a

timeout of 60 seconds use **PARM=480** (480 is equal to 8 * 60). If the parm value of these bits is zero, the default timeout is used.

See page 387 for information on modifying PCLINK defaults to support the *WRQ* protocol.

VAXLINK2

To pass parameters to VAXLINK2 as part of the host startup sequence (for VAX/VMS hosts), you must install VAXLINK2 as a foreign command. Add the following to your LOGIN.COM file:

```
$VAXLINK2 :== $[dev:][directory]VAXLINK2.EXE
```

If you make VAXLINK2 a foreign command, simply use VAXLINK2 as the host startup sequence. VAXLINK2 can then accept the following parameters:

```
VAXLINK2 [T<seconds>] [N<n>] [L] [V] [X<n>]
```

[T<seconds>] Sets the number of seconds that elapse without a response from the PC. The default timeout is *30* seconds. The VAXLINK2 timeout parameter should always be set to a value at least five seconds greater than **Receive timeout** (see below).

[N<n>] When there is no response from the PC, the file transfer is *
attempted a default of *10* times before giving up. This *
qualifier sets the retry limit to <n> consecutive times. <n> *
must be an integer. *

[L] Creates a log file of the file transfer called WRQLOG— *
see page 84 in *Troubleshooting File Transfer* for more *
information. *

[V] Displays the VAXLINK2 version number and exits the file *
transfer program. *

[X<n>] During file transfer, blocks may need to be retransmitted. *
[**X<n>**] sets the number (<n>) of retransmits to be *
performed before quitting. *

UNXLINK2

The UNXLINK2 default startup sequence is entered during the upload process. The startup sequence must be lowercase and read *unxlink2*.

UNXLINK2 supports a number of host startup sequence parameters:

```
unxlink2 - [t<seconds>] [n<n>] [l] [v] [x<n>] [c]
```

[t <seconds>] Sets the number of seconds that elapse without a response from the PC. The default timeout is *30* seconds. The UNXLINK2 timeout parameter should always be set to a value that is different from **Receive timeout** (see below).

[n<n>] When there is no response from the PC, the file transfer is attempted a default of *10* times before giving up. This qualifier sets the retry limit to <n> consecutive times. <n> must be an integer.

[l] Creates a log file. See *Troubleshooting File Transfer* on page 84.

[v] Displays the version number and exits the file transfer program.

[x<n>] During file transfer, blocks may need to be retransmitted. **[x<n>]** sets the number (<n>) of retransmits to be performed before quitting.

[c] Wildcard transfers to UNIX hosts result in lowercase filenames on the host. To have them result in uppercase filenames, append **-c** to the host startup sequence. (In file transfer without wildcards, the local filename is reproduced on the host with upper and lowercase as entered: it is not changed during transfer.)

Receive Timeout (Seconds)

This field determines the maximum number of seconds the file transfer waits for a response from the host or PC. If nothing is received within the period specified, the program attempts to resynchronize. The error retry limit determines when the transfer program no longer attempts to retransmit the block.

Values: 1-9999
Default: 15 (seconds)

Reflection command language equivalent:

```
SET RECEIVE-TIMEOUT <value>
```

Packet Size

Packet size tells the file transfer program how many bytes of data to transfer at a time.

The transfer program breaks a file into packets that are transmitted one at a time until the entire file has been transmitted. You specify the number of data bytes each packet contains. There is no standard correlation between this field and the size of the records or blocks on the host computer.

A large packet size is more efficient when there are few transmission errors. Therefore, the value you assign to this field should depend on the likelihood of transmission errors. This is especially important if you're communicating over a modem, which has a greater chance of error. Use the default packet size in most cases. The size of the packet affects the speed of transfers:

- Too small a packet results in long transmission times because of the overhead that occurs for each packet.
- Too large a packet may increase the elapsed time because the chance of error increases with packet size.

Use the default packet size in most situations.

Values:	32-1280	HP (*WRQ* protocol only)	*
	32-1024	VT (*WRQ* protocol only)	*
Default:	1280	HP	*
	512	VT	*

Reflection command language equivalent (*WRQ* protocol only): *

```
SET TRANSFER-FRAME-SIZE <value>
```

For information on setting **Packet size** for UNIX transfers, see page 352.

Error Retry Limit

The error retry limit tells Reflection how many consecutive times to attempt to correct an error before abandoning the transfer.

If an error condition occurs many times consecutively, you may have an insurmountable communications problem. With this parameter, you define the point at which an error state is declared unsolvable.

Values: 1-9999
Default: 10 (retries)

Reflection command language equivalent:

```
SET RETRY-LIMIT <value>
```

File Transfer Link

This field is relevant only for transfers using the *WRQ* protocol. *

The value for this field (*8-BIT*, *7-BIT*, or *USER-DEF*) indicates the extent to which non-printing ASCII characters are automatically converted to printable ASCII characters. These conversion options are related to file transfer only; they do not alter the data sent. The field values are listed here and discussed in detail below. *

Values: 8-BIT (the default)
7-BIT
USER-DEF

Reflection command language equivalent:

8-BIT (the default)

```
SET TRANSFER-PRESENTATION "A"
SET TRANSFER-START-CHAR "^B"
SET TRANSFER-END-CHAR "^C"
```

7-BIT

```
SET TRANSFER-PRESENTATION "C"
SET TRANSFER-START-CHAR "("
SET TRANSFER-END-CHAR ")"
```

USER-DEF

If the values for transfer presentation and start and end of text characters do not correspond to *8-BIT* or *7-BIT*, *USER-DEF* is automatically used.

8-BIT

This is the default value. It provides the fastest file transfer, converting only the following characters:

WRQ Protocol

Description	String	
Carriage Return	^M	
DC_1 (XON)	^Q	
DC_3 (XOFF)	^S	
End of Medium	^Y	
Pound sign	#	
Ampersand	&	
DC_2	^R	HP only
ENQ	^E	HP only
ACK	^F	HP only
DEL	7FH	HP only
Null	^@	HP and UNIX only
Ctrl-Z	^Z	UNIX only
Linefeed	^J	UNIX only
^Q, high bit	P_1	UNIX only
^S, high bit	S_F	UNIX only
Decimal 128	Hex 0	UNIX only
Decimal 255	Hex FF	UNIX only

In addition, the start of frame and end of frame characters are converted. The default start of frame character is ^B (STX). The default end of frame character is ^C (ETX).

The *8-BIT* setting requires an 8-bit data path. Other configuration parameters in Reflection can affect your data path, overriding the displayed *8-BIT* setting .

- If you change *Parity* to a value other than *NONE* on the Datacomm Configuration menu, you no longer have an 8-bit data path. All characters except for the start of frame and end of frame characters are converted.
- For VT emulation, change **Terminal type** to one of the *VT220* options on the VT Terminal Configuration, Page 1 menu for an 8-bit data path.

7-BIT

When *7-BIT* is chosen, each data character, including binary data, is examined and (if necessary) modified to ensure that it is a non-control (printable) ASCII character. This data modification is reversed by the recipient (host computer or PC), so that the data being transferred is not affected. Each packet starts with an opening parenthesis—"(" and ends with a closing parenthesis— ")" .

Selecting this value slows down the speed of transfer.

USER-DEF

USER-DEF (user-defined) is displayed if the set of characters that are converted to printable ASCII are other than the defaults used by *8-BIT* or *7-BIT* values. This approach applies to certain host applications and network environments. The value of **File transfer link** is automatically changed to *USER-DEF* if the values for transfer presentation or start and end of frame characters do not correspond to the values for *8-BIT* or *7-BIT*.

Changing Start and End of Text Characters

The symbol ^B (start of frame) is normally used to begin data packets, and ^C (end of frame) is normally used to end them. Alternate characters are required if the default characters do not remain intact when passed through intermediary devices such as multiplexers and error-correcting modems.

You can change the start and end characters from the command line, for example:

```
SET TRANSFER-START-CHAR "("
SET TRANSFER-END-CHAR ")"
```

The value for either of these SET parameters must be a control character (e.g., ^L, ^J) or a parenthesis (as in the above example). Replacing these values automatically changes **File transfer link** to *USER-DEF*. This is faster than the *7-BIT* option.

Changing the Presentation

On the HP 3000, you can change the value of SET TRANSFER-PRESENTATION to take advantage of flow control without slowing down to 7-bit transfer. Changing the presentation also allows you to use SET TRANSFER-EXTRA-CHARS: you can specify additional character translation beyond the base set of characters listed under the *8-BIT* option.

Flow control from the host is automatically turned off with the *8-BIT* option. For a network environment requiring flow control, the recommended setting for fastest transfer (highest speed while still using flow control) is as follows:

```
SET TRANSFER-PRESENTATION "B"
SET TRANSFER-EXTRA-CHARS ""
```

After making these changes, **File transfer link** has a value of *USER-DEF*.

UNIX Transfer

UNIX transfer converts ^Z so that systems using this sequence as a job control *suspend* keystroke can function correctly.

If you or your system manager has redefined a suspend as *^U*, for example, you may need to convert that character accordingly. Make the following changes on the command line:

```
SET TRANSFER-PRESENTATION "B"
SET TRANSFER-EXTRA-CHARS "^U"
```

TelnetManager Users

There are some configuration issues to remember if you are using Walker Richer & Quinn's TelnetManager and the *WRQ* file transfer protocol.

In order for you to do 8-bit file transfer, TelnetManager must be in 8-bit (binary) mode. Otherwise Reflection will do a 7-bit transfer, regardless of the value of **File transfer link**. See the TelnetManager documentation for information on how to set it up for an 8-bit data path.

39.5 ASCII Translation to Host Fields

The settings for these fields apply when you send files using the ASCII transfer method, via the File Transfer screen or Reflection's command language. Refer to page 339 for a discussion of binary vs. ASCII file transfer methods.

CR/LF = Record Separator

Setting this field to *YES* tells the host program to generate a new record every time it receives a carriage return/linefeed sequence.

Values: YES (the default)
NO

Reflection command language equivalent:

```
SET CR/LF=SEPARATOR YES/NO
```

Expand Embedded Tabs

This field should be *YES* if you want the PC file's tab characters to be expanded to spaces in the host file. The system then replaces each tab character with the number of spaces necessary to fill out to the next tab stop, where tab stops are assumed in column 9 and every eighth column thereafter.

Values: YES (the default)
NO

Reflection command language equivalent:

```
SET TABS-TO-SPACES YES/NO
```

Use Ctrl-Z as EOF

On the PC, ASCII text files normally end with a Ctrl-Z (^Z) character. When this field is set to *YES* (the default), file transfer uses Ctrl-Z as the end-of-file marker and strips it from the file being sent. If you do *not* want the file transfer to use Ctrl-Z as an end-of-file marker, set this field to *NO*; the character count in the file directory is used to determine the file length.

Values: YES (the default)
NO

Reflection command language equivalent:

```
SET CTRL-Z=EOF YES/NO
```

Change Roman 8 to ISO-7

If your host does not support the Roman 8 character set, set this field to *YES*. Roman 8 files are translated to an ISO-compatible format during the transfer. The **Keyboard/Nationality** setting on the Global Configuration menu determines the conversion. See the *ISO Replacement for 7-Bit Operations* chart on page 183.

Values: NO (the default)
YES

Reflection command language equivalent:

```
SET ROMAN8-TO-ISO7 YES/NO
```

39.6

ASCII Translation from Host Fields

The settings for these fields apply when you select [F2], **RECEIVE FROM HOST**, from the File Transfer screen or when you use Reflection's RECEIVE command.

Record Separator = CR/LF

If you want lines received from the host to be separated by a carriage return/linefeed on the PC, set this field to *YES*.

Values: YES (the default)
NO

Reflection command language equivalent:

```
SET SEPARATOR=CR/LF YES/NO
```

Change Spaces to Tabs

If you want tab characters to replace consecutive spaces, set this field to *YES*. Many software products on the PC interpret embedded tabs as standard 8-column tab stops. Selecting *YES* can save considerable disk space on the PC.

Note: This field is also used to control replacement of tabs with spaces when logging to disk or copying from display memory.

Values: NO (the default)
YES

Reflection command language equivalent:

```
SET SPACES-TO-TABS  YES/NO
```

Write Ctrl-Z EOF

On the PC, ASCII text files normally end with a Ctrl-Z (^Z) character. If you want a ^Z character to be added to the file when it is received, set this field to *YES*. Many PC programs require the ^Z end-of-file marker.

Values: YES (the default)
NO

Reflection command language equivalent:

```
SET WRITE-CTRLZ YES/NO
```

Delete Trailing Spaces

Some host text files use fixed-length records to delimit lines and pad the end of each record with blanks. Most PC text processing programs use CR/LF (carriage return, linefeed) to delimit lines and paragraphs, and thus do not need blanks preceding a delimiter. You can save a great deal of disk space by setting this field to *YES*.

Note: This field also applies when you are logging or copying to a disk file.

Values: YES (the default)
NO

Reflection command language equivalent:

```
SET DELETE-TRAILING-SPACES YES/NO
```

Change ISO-7 to Roman 8

If you want ISO files from the host to be converted to Roman 8, set this field to *YES*. ISO files are translated to Roman 8. The **Keyboard/Nationality** field on the Global Configuration menu determines the character conversion. See the *ISO Replacement for 7-Bit Operations* chart on page 183.

Reflection ignores this field when performing a binary file transfer.

Values: NO (the default)
YES

Reflection command language equivalent:

```
SET ISO7-TO-ROMAN8 YES/NO
```

39.7 Setting Up Remote Transfers

It is possible to force PCLINK2 to use a communications port other than the one assigned to your session. For instance, a host program running as part of a batch job could schedule and perform backups or file transfers to and from any number of PCs running Reflection.

When PCLINK2 opens the communications port, it opens the formal file **HPPCPORT**. This enables PCLINK2 to do file transfers to the PC assigned as the $STDLIST/$STDIN device, as well as any other PC not opened by any process.

When trying to open a non-$STDLIST/$STDIN port, note the following limitations:

- The port must not be currently opened by another process, including the command interpreter (CI). The CI is awakened by a carriage return, and eventually times out and releases the port if it detects no activity.
- For PCLINK2 to communicate with a PC designated via HPPCPORT, Reflection must be running and the baud rate must be configured to match that of the port's default baud rate. PCLINK2 does not do any speed sensing to adjust the port's baud rate.

  PCLINK2 is *slave* oriented. It expects Reflection to tell it what files go in which direction. If the host is to control a file transfer, it must force Reflection to initiate the file transfer. The Reflection command escape sequence ^{E}*C&oC* is the tool for host-initiated file transfers.

  For example, to tell Reflection to send the PC file PCFILE.PC to the HP 3000, a host program would go through the following steps:

  1. FOPEN the PC's port. This gives you a file handle for subsequent FWRITES and FREADS. The port should be opened with the following:

     foptions = %600 (Undefined length records)
     aoptions = %404 (Read/Write and NOBUF access)

  2. FWRITE the following escape sequence:

     ```
     EC&oCSEND pcfile.pc TO hpfile ASCII CR
     ```

     (^{E}C and ^{C}R are the ASCII characters for escape and carriage return.)

  3. FREAD the host startup sequence that Reflection sends to start the host file transfer software. (The default is *RUN PCLINK2.PUB.SYS.*)
  4. Create and activate PCLINK2.
  5. FREAD the completion code that Reflection sends indicating the success or failure of the host-initiated command.

  If the host program is only concerned with the PC that is the $STDLIST/$STDIN port, the first step is unnecessary and subsequent FREADs and FWRITEs could be replaced with standard program statements that read and write to the standard display device (i.e., DISPLAY and ACCEPT in COBOL).

The file COBDEMO.COB on the support files diskettes is an example of this type of remote operation. COBDEMO.COB is covered in *Issuing Commands from the Host* in the Reflection *Command Language Manual.*

Reflection supports separate startup procedures for *WRQ* and *OLD-WRQ* protocols, and remote transfers initiated within a host procedure should verify the host startup sequence as part of the transfer. COBDEMO illustrates this process; see page 392 for more information.

Note: If the transfer process you are attempting involves restoring files from a backup file to a PC, you must repeat steps two through five. RESTORE runs the PCLINK2 program twice, so the completion code must be captured twice.

Chapter 40

KERMIT Protocol

KERMIT is a file transfer protocol developed by Columbia University. The advantage of KERMIT is its wide availability. KERMIT software is available for a variety of machines, including the HP 1000. It can often be obtained through user groups, or you can contact:

KERMIT Distribution
Columbia University Center for Computing Activities
612 West 115th Street
New York, New York 10025

40.1 Using KERMIT

File transfers can be performed from the Reflection File Transfer screen or by using the KSEND and KRECEIVE commands. You can transfer files directly between PCs or between a PC and host running KERMIT software.

To use the File Transfer screen to send a file from your PC to a host with KERMIT, follow these steps:

1. Log on to the host using Reflection as a terminal and invoke the host KERMIT program by typing **KERMIT** (or the string used by your particular version of KERMIT).
2. If the file is being transferred from the PC to the host, put the remote KERMIT in receive mode by typing **RECEIVE**.

   If the file transfer is from the host to the PC, type **SEND <filespec>** (wildcards are permissible) at the host prompt.

3. Go to the File Transfer screen (F6 from the system keys).
4. If the protocol displayed at the top of the screen is not KERMIT, select F6, **config transfer**.
5. Use F1, **NEXT CHOICE** from the File Transfer Configuration menu, until *KERMIT* is displayed. The default transfer method for KERMIT is binary. The **ASCII TRANSLATION TO HOST** and **ASCII TRANSLATION FROM HOST** areas of the File Transfer Configuration menu affect ASCII KERMIT file transfers. Normal ASCII translation capability (such as expanding embedded tabs, etc.) can therefore be performed.

   Be sure to check these values before starting your transfer; refer to page 364. Activate or save the configuration.
6. Fill in the applicable information for the local and host filenames.
7. *Using File Transfer* starting on page 335 describes the basic file transfer process. Refer to page 338 for a summary of all protocol options available from the File Transfer screen.

When sending a single file, you must issue the SEND or RECEIVE command to the host KERMIT before transferring each file. If you are using wildcards to send multiple files, you must not include any parameters with the RECEIVE command to the host KERMIT.

Using KERMIT's Server Mode

If the KERMIT on your host has a server mode, you can use the following file transfer process:

1. After invoking KERMIT on your host computer, put it in server mode by typing **SERVER**.
2. To send a file, type **KSEND <filespec>** from the Reflection command line (Alt-Y).
3. To receive a file, type **KGET <filespec>** from the Reflection command line.
4. Use **KFINISH** or **KBYE** to terminate the server and log off the host system respectively.

A list of available KERMIT commands is on page 382. A full description of the syntax is in the Reflection *Command Language Manual.*

40.2

Using KERMIT Between PCs

You can use KERMIT to send files between PCs. One PC acts as the *host* PC and the other acts as the *terminal* PC.

1. Connect the two PCs using a null modem cable (pins 2 and 3 crossed), and load Reflection on both PCs.*
2. Make sure that the baud rate and parity are the same on both PCs.
3. You can verify that steps 1 and 2 are correct by typing on one PC. The letters you type should appear on the other PC.
4. On the receiving PC type **KRECEIVE** on the command line.
5. On the sending PC type **KSEND <filespec>** on the command line.

Aborting PC to PC Transfer

To abort the file transfer, press [F1], **STOP**, on the terminal PC and wait for an orderly termination, which should take a few minutes at most. *Do not press* [F5], **ABORT TRANSFER**, unless the host is another PC or the host program does not time out. Usually the host terminates the program for you once [F1] is pressed. If KERMIT is still running on the *host*, abort it by typing [Ctrl]-[C] on the terminal PC. If the host is a PC, you can type [Ctrl]-[Break] on the host PC to terminate the program.

40.3

Parity Settings

The parity to be used during file transfers is determined by both machines. If both are able to operate with **Parity** set to *NONE* (8-bit mode), Reflection transfers in 8-bit mode. If one parity is set at other than 8-bit, 7-bit mode is used. Problems occur if both machines operate at 8-bit, but an intermediary, connecting system can only operate in 7-bit mode. You must set Reflection's parity to *0's* in this case to force KERMIT to operate in 7-bit mode.

---

* Or the PC and Macintosh.

40.4 Command Language

In addition to KSEND and KRECEIVE, there are a number of KERMIT-related Reflection commands:

KBYE
KFINISH
KGET
SET KERMIT-RECEIVE-START
SET KERMIT-SEND-START

See the Reflection *Command Language Manual* for descriptions.

Chapter 41

XMODEM Protocol

XMODEM is public domain software available for a variety of machines. File transfers can be performed from the Reflection file transfer screen or by using the XSEND and XRECEIVE commands. You can transfer files directly between PCs or between a PC and a host running XMODEM software. Hosts for which no Reflection protocol exists may require this method of file transfer.

Some versions of XMODEM are able to perform CRC-error checking (Cyclic Redundancy Checking). When sending, Reflection responds to either a CRC or checksum request. When receiving, Reflection first requests CRC-error detection; if this fails, Reflection uses checksum error detection. SET XMODEM-CRC NO forces checksum.

41.1 Using XMODEM

To send or receive files between computers using XMODEM, follow these steps:

1. Load Reflection (or other software with XMODEM commands) on the sending PC. (When transferring between two PCs, load communications software such as Reflection on *both* PCs.) Reflection must be in remote mode.
2. Be sure that the baud rates are the same (if two PCs are used, the appropriate datacomm port on each machine should be the same).

3. Connect the PC to the host or another PC with a null modem cable. Pins two and three should be crossed once, unless the host is an HP 3000. You may also connect via a modem.
4. On the host, invoke the XMODEM program.
5. Go the File Transfer screen ($\boxed{\text{F6}}$ from the system keys).
6. If the protocol displayed at the top of the screen is not XMODEM, select $\boxed{\text{F6}}$, **config transfer.**
7. Use $\boxed{\text{F1}}$, **NEXT CHOICE** from the File Transfer Configuration menu, until *XMODEM* is displayed. Activate or save the configuration.
8. *Using File Transfer*, starting on page 335, describes the basic file transfer process. **Transfer method** does not apply to XMODEM; all transfers are binary. Refer to page 338 for a summary of the XMODEM options available from the File Transfer screen.

If you prefer, you can use XSEND and XRECEIVE commands from the command line. See the Reflection *Command Language Manual.*

Timeouts During Transfers

During an XMODEM file transfer, the sending or receiving PC might *time out* and cancel the transfer. If this happens, check your cable connections, or simply reestablish communications and try the transfer again.

Aborting PC to PC Transfer

To abort the file transfer, press $\boxed{\text{F1}}$, **STOP**, on the PC and wait for an orderly termination. *Do not press* $\boxed{\text{F5}}$, **ABORT TRANSFER**, unless you are transferring between PCs or the host program does not time out. Usually the host terminates the program for you once $\boxed{\text{F1}}$ is pressed. If XMODEM is still running on the host PC, abort it by typing $\boxed{\text{Ctrl}}$-$\boxed{\text{C}}$ on the terminal PC. If the host is also a PC, you may type $\boxed{\text{Ctrl}}$-$\boxed{\text{Break}}$ to abort the transfer.

Chapter 42

FTP Protocol

The FTP (*File Transfer Protocol*) choice on the File Transfer Configuration menu takes advantage of the speed of the widely used TCP/IP network.

The following is required in order to use the FTP protocol:

- A PLUS version of Reflection (LAN support)
- TelnetManager, a Reflection Complement, installed on the PC
- An FTP server program on the host
- The PC and the host must be on a TCP/IP network

To transfer files using the FTP protocol, check the following:

- Your **Datacomm port** on the Basic or Datacomm Configuration menu should be *TEL-MGR*.
- Your protocol choice on the File Transfer screen should be *FTP*. If it is not, go to the File Transfer Configuration menu (F6) and press F1 in the **File transfer protocol** field until *FTP* is displayed. Activate or save the setting to a configuration file.

Using the File Transfer screen you can send and receive files. If you use Reflection's command language, you have additional options:

- Wildcard transfer—FMGET, FMPUT
- Host directory listings: short (FLS) and long (FDIR)
- The ability to change host directories—FCD (use **CD** on the Reflection command line to change your local directory)

- You can use TENEX as a method for transfers from hosts that use a 9-bit byte to convert to an 8-bit byte format (such as the DECsystem 20)

See the Reflection *Command Language Manual* for the full syntax of these commands.

42.1 File Transfer Screen

For the most part, the fields on the File Transfer screen work as they do when you are using the WRQ protocol (described on page 335). The differences that apply when you use the FTP protocol are described below.

- Wildcards are not allowed at the File Transfer screen (they can be used with command language). On the screen, a wildcard character is treated like an ordinary character.
- If no filename is entered, the source file specification is assumed to be the same as the destination file specification.
- The only field on the File Transfer Configuration menu that applies to the FTP protocol is **File transfer protocol.**

Chapter 43

OLD-WRQ Protocol

In almost all cases, the new *WRQ* default protocol should be your standard method of transferring files. There are a few instances where you might need to temporarily switch to the *OLD-WRQ* protocol:

- You may need to dial to another system that does not have a copy of PCLINK2, VAXLINK2, or UNXLINK2 installed
- You may be using existing command files written to support both PCs and Macintosh computers *

DeskDirect, a Reflection Complement, automatically picks up your new startup sequence and protocol. Version 3.0 of RSVP, another Reflection Complement, uses the WRQ protocol exclusively.

If you have existing command files or procedures with a literal reference to these earlier file transfer programs, change all references to PCLINK2, VAXLINK2, or UNXLINK2. Additional changes that may be required are covered in this chapter.

The *OLD-WRQ* protocol can be used only if you have a copy of one of Reflection's pre-4.0 host file transfer programs, PCLINK, VAXLINK, or UNIXLINK already installed. In most cases, you can change to the *WRQ* protocol as soon as PCLINK2, VAXLINK2, or UNXLINK2 is installed on the host. You can then take advantage of the new wildcard transfer as well as extended options of the SEND and RECEIVE commands.

* Reflection for the Macintosh does not yet support the new WRQ protocol.

OLD-WRQ Protocol

43.1 Configuring for OLD-WRQ Protocol

The File Transfer screen supports a variety of protocols, but some fields apply to the WRQ protocol only. See the table on page 338 for a comparison of the protocol-specific fields of the File Transfer menu.

When you change from *WRQ* to *OLD-WRQ*, some fields on the File Transfer screen and the File Transfer Configuration menu are affected.

Function Key Labels

F7 on the File Transfer screen, **upload xfer pgm**, uploads PCLINK2 or VAXLINK2 (the host file transfer program). It does not upload VAXLINK or PCLINK.

Transfer Method

ASCII

When sending to a host using ASCII transfer, the host defaults to an ASCII file with variable-length records. The specified host record size is used as the maximum record size.

Binary

When sending to a VAX host using binary transfer, the host defaults to a binary file with variable-length records. The specified host record size is used as the maximum record size. Since binary data does not contain any end of record characters, the host file will have records that are all of the specified length, except for the last one, which will contain the remaining bytes of the file.

Host Filename

For HP emulation, this field on the File Transfer screen supports the same switches, (**;F**, **;L**, etc.) for both PCLINK and PCLINK2.

For VAXLINK, qualifiers **/C** , **/D**, and **/S** are supported as documented on page 357. Additional qualifiers specific to VAXLINK follow.

/F The default format for binary transfer using *OLD-WRQ* is variable; the default for *WRQ* is fixed. Use the **/F** qualifier to specify fixed length binary files when using *OLD-WRQ*.

/W Specifying WordPerfect format with **/W** can be used with VAXLINK (*OLD-WRQ*) only; this qualifier is unnecessary if the protocol is *WRQ*.

/I Image transfer is slightly different under the two protocols. To maintain an exact copy of your file, use the same protocol (*OLD-WRQ* or *WRQ*) both in sending and receiving the file—don't send it with one protocol and then receive it with another.

Host Record Size (Send only)

The field determines the maximum record size for ASCII and binary files sent to the HP.

When transferring to a VAX/VMS host, this field specifies a binary file's maximum record length:

- An uploaded binary file is a sequential file with variable-length records.
- An uploaded ASCII file has variable-length records; the maximum record size is specified by **Host record size.**

Values:	1–32767	
Default:	80	HP
	255	VT

If File Exists

When receiving a file from the host with the *OLD-WRQ* protocol, this field allows all three options—*C* (Cancel), *D* (Delete), and *A* (Append). When sending to the host, you cannot append a file. (With HP emulation, you receive an error message. With VT emulation, a new file is created.)

Host Startup Sequence

You can save two separate startup sequences on the File Transfer Configuration menu—one for *WRQ* and one for *OLD-WRQ*. See page 392 for information on using two host startup sequences and a comparison of the parameters that can be added to each.

Value: A string of up to 31 characters
Default: RUN PCLINK.PUB.SYS (HP mode)
RUN VAXLINK (VT mode)
unixlink (UNIX)

Reflection command language equivalent:

```
SET HOST-STARTUP <value>
SET HOST-START-OLD <value>
```

OLD-WRQ Protocol

Packet Size

When *OLD-WRQ* is selected, **Packet size** refers to a data block size; the default and the range of values differ from the packet size used in *WRQ* protocol.

Values:	64-512 (HP)	OLD-WRQ only
	64-478 (VT)	OLD-WRQ only
Default:	512 (HP)	OLD-WRQ only
	478 (VT)	OLD-WRQ only

Reflection command language equivalent (OLD-WRQ only):

```
SET BLOCK-SIZE <value>
```

The command language equivalent for this field using *WRQ* protocol is SET-TRANSFER-FRAME-SIZE <value>. Different packet size defaults for these two protocols can be saved independently.

The remaining fields on the File Transfer Configuration menu operate as documented on pages 369 through 393.

43.2 Switching to the WRQ Protocol (HP)

Some PCLINK settings are set differently with PCLINK2. 7-bit file transfer is still the most reliable method to ensure data integrity. With PCLINK, this was set by adding PARM=1 to the host startup sequence. If you experience transfer errors with PCLINK2's default 8-bit transfer, try the following:

- Type **SET TRANSFER-PRESENTATION "B"** on the Reflection command line.* If this solves the transfer errors, save the setting to your configuration file.
- If you still have problems, change the value of **File transfer link** to *7-BIT*. This is equivalent to using PCLINK's PARM=1 setting.

Compression techniques in PCLINK2 differ from PCLINK, and are not set by PARM= additions to the host file transfer program. See SET COMPRESSION in the Reflection *Command Language Manual* for more information.

If a transfer lost a start-of-header character, PCLINK provided the parameter PARM=5 to avoid this. PCLINK2 uses the commands SET TRANSFER-START-CHAR and SET TRANSFER-END-CHAR. See page 371.

* This automatically changes the field **File transfer link** to *USER-DEF*.

Table 30
Comparing WRQ and OLD-WRQ (HP)

PCLINK Parameter	PCLINK2 Equivalent
PARM=1	Set **File transfer link** to *7-BIT*.
PARM=2, PARM=3	No equivalent. PCLINK2 handles compression independently.
PARM=5	SET TRANSFER START-CHAR <value> SET TRANSFER END-CHAR <value>
Timeout	Still used, any existing setting still applies. Refer to page 365 if changes are required.

43.3 Switching to the WRQ Protocol (VT)

The default transfer link for VAXLINK is 7-bit, while for VAXLINK2 it is 8-bit. If you used the faster 8-bit method with VAXLINK (specified by the **e** parameter), changing to VAXLINK2 automatically defaults to the faster method.

If you find that the change to VAXLINK2 produces transfer errors, change the value of **File transfer link** to *7-BIT*.*

* If you can identify which characters are not being processed correctly, you may be able to use a faster user-defined method by changing the presentation and assigning extra characters for translation. See page 371 for more information.

Table 31
Comparing WRQ and OLD-WRQ (VAX)

VAXLINK	VAXLINK2
[e] 8-bit transfer	VAXLINK2 default— **File transfer link** is *8-BIT*.
[t <n>] timeout	Timeout is unchanged.
[n] retry limit	Retry limit is unchanged.
[r] readsync	Readsync mode not necessary. VAXLINK2 sets readsync automatically.
[o] old (PC 2622)	Not supported with VAXLINK2.
Default Binary Format	
Variable	Fixed

43.4 Maintaining Two Host Startup Sequences

Existing command files and procedures that specify PCLINK or VAXLINK by name can be changed to use the new file transfer program. In most cases, changing the host startup sequence to PCLINK2 (or VAXLINK2) and making any of the modifications covered above are the the only changes required.

However, some host-based procedures may need to be upgraded so that the host file transfer program's name is based on the current setting defined by Reflection.

COBDEMO.COB, distributed with Reflection, is a sample COBOL program that illustrates how a host-based transfer procedure can query the Reflection startup sequence. Both COBDEMO.COB and a VAX example, DEMO1.DCL, are listed in the Reflection *Command Language Manual.*

If you need to support startup sequences for both protocols, any existing command files can be updated so that SET HOST-STARTUP is updated from the value saved in SET HOST-START-OLD or SET HOST-START-WRQ. See the Reflection *Command Language Manual* for more information.

Note: Do *not* rename your new host file transfer programs (PCLINK2, VAXLINK2, UNXLINK2) to the old filenames (PCLINK, VAXLINK, UNIXLINK). This could cause confusion for users with pre-4.0 versions of Reflection. In addition, you will receive the message *Host software requires OLD-WRQ transfer protocol* if your startup sequence specifies PCLINK, VAXLINK, or UNIXLINK and your selected protocol is *WRQ*.

Section 10

HP Graphics Emulation

HP Graphics

Chapter 44

Emulating a Graphics Terminal

Reflection 7 allows your PC to emulate the HP 2623A monochrome graphics terminal and the HP 2627A color graphics terminal. It also includes emulation of a Tektronix 4010 terminal. See page 435 for more on the technical aspects of color graphics.

44.1 Graphics Terminal Emulation

You can use Reflection's graphics terminal emulation in a number of ways. Its main purpose is to allow you to have access to graphics programs running on the host computer with your PC. *HP Draw*, *HP EasyChart*, and *DSG3000* are examples of such programs. Because each application program is different, review the program's documentation for specific information.

The host controls all of the graphics features discussed here and in the *Programming Graphics Functions* beginning on page 407. You do not have to enter escape sequences or even turn on the graphics keypad; the host activates the keypad and graphics display as you use graphics software on the host computer.

Note: Reflection allows you to use graphics data that contains decimal points. The fractional portion is ignored.

Alphanumeric and Graphics Memory

When Reflection is emulating a graphics terminal, the PC maintains two distinct memories:

Alphanumeric memory
Contains the data you normally see displayed on the screen. This memory stores the ASCII characters you type at the keyboard, send to the display from disk, or receive from the host computer. The capacity of alphanumeric memory varies depending on the machine you use and your configuration, but it is usually many pages long.

Graphics memory
Stores the graphics text, lines, and patterns unique to graphics programs. The capacity of graphics memory is one screen.

To simplify the notion of graphics memory, visualize it as a grid composed of hundreds of individual pixels (dots) arranged closely together in columns and rows on a plane. When graphics memory is empty, each pixel contains a value of zero, indicating that it is turned *off* and is invisible. When the graphics display is active, giving a pixel a value of 1 (one) turns it *on*, creating a visible image on the screen.

Color graphics uses three planes: red, green, and blue. Placed one on top of the other, these planes allow a variety of colors to be displayed by mixing which plane is on or which pixels in any plane are on. By turning on or off all or part of a plane, the color displayed varies. See page 435.

Alpha and Graphics Displays

The screen windows associated with the alphanumeric and graphics memories are also distinct, with very different characteristics. The graphics cursor, for example, looks like a large crosshair, while the alphanumeric cursor is a small blinking underline or rectangle.

Note: When the graphics display is on, the image of the screen may appear to shrink, and Reflection's scrolling action slows down because the amount of data on the screen is several times the amount contained in alphanumeric display.

Display Control

You can control the graphics and alphanumeric memories independent of their associated displays. When graphics memory is being addressed, its display is normally *on*, or visible on the screen. However, graphics memory may also remain active after you turn its display off, allowing you to write to memory *invisibly*. The same is true of alpha memory.

Turning alpha or graphics display off does not erase the contents of its memory; it simply makes it temporarily invisible.

You can have both alpha and graphics displays on the screen simultaneously. However, the background color should be black: if the SET command IGNORE-COLOR-PAIRS is set to *NO* (the default) when background color has been changed from black, the graphics display is blocked out when the alpha display is on.

Graphics Text vs. Alpha Text

The text generated in graphics text mode is different from the characters you see on the screen during normal operations. Graphics text can be generated in eight different sizes, in upright or slanted faces, and can be oriented at any of four angles. You select the graphics text image you want by means of escape sequences typed at the keyboard; they are typically set by escape sequences received from the host computer.

44.2 Configuring for Graphics

No special configuration is required to run Reflection 7 in graphics mode. Depending on your graphics adapter, you are configured as a 2627A (color) graphics terminal, a 2623A (monochrome) graphics terminal, or, if no graphics adapter is found, a 2392A alphanumeric terminal. The **Terminal ID response** is checked by Reflection after determining the adapter type to ensure that the capabilities of the terminal indicated are available. See *HP Terminal Configuration* on page 103 for more information.

Note: Running without graphics emulation preserves around 100K of memory in the PC. If you are not using the graphics portion, use **/G0** when loading Reflection 7 to force a 2392A non-graphics emulation.

Defining the Graphics Adapter at Startup

When loading, Reflection 7 attempts to determine your graphics hardware. You may occasionally need to explicitly define your adapter type by using a /G switch when loading; see page 13.

You can use a DOS SET environment command to ensure that Reflection 7 is run with the correct graphics switch each time you load the program. See page 21.

44.3

Controlling the Display and Cursor

If **Terminal ID response** on the Terminal Configuration, Page 2 menu is set to *2623A* or *2627A*, the alpha display and alpha cursor are on, and the graphics display, graphics cursor, and graphics keypad are off.

The Graphics Keypad

The following figure shows how your PC numeric keypad is mapped to simulate the graphics display control and cursor movement keys.

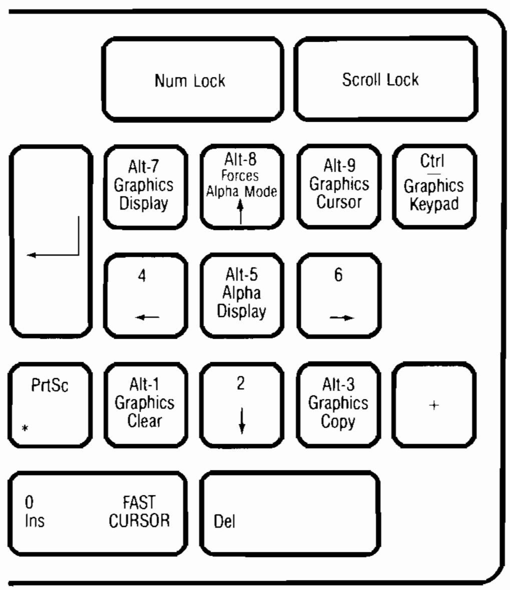


Figure 52: Graphics Keypad

All of the graphics keypad functions (Alt-<n>) are available regardless of whether the graphics keypad is toggled on. When the keypad has been selected for graphics cursor movement, the letter **G** appears between the key labels on the screen. You can press Ctrl-- (keypad) to make the PC's arrow keys control the graphics cursor if the host program does not toggle the keypad on for you with an escape sequence.

The numeral keys referenced in the table below are the *keypad* numeral keys —not the ones on the first row of your keyboard above the letter keys.

Table 32
Graphics Keypad Functions

Keystroke	Key Name	Function
Alt - 1	GRAPHICS CLEAR	Clears the contents of graphics memory.
Alt - 3	GRAPHICS COPY	Copies the contents of graphics memory to the printer (even if the graphics display is off).
Alt - 5	ALPHA DISPLAY	Toggles alphanumeric display on and off. When the alpha display is off, the screen labels and the alpha cursor disappear, along with any alphanumeric characters that may be on the screen. Toggling alpha display off does not affect the contents of the alpha display memory.
Alt - 7	GRAPHICS DISPLAY	Toggles graphics display on and off. The graphics cursor is invisible when graphics display is off. Toggling graphics display off does not affect the contents of graphics memory.
Alt - 8	FORCE ALPHA	Returns the PC to alpha mode once graphics mode has been turned on.
Alt - 9	GRAPHICS CURSOR	Toggles the graphics cursor on and off.
Ins	FAST CURSOR	Increases the speed of the graphics cursor when used with arrow keys.

Alpha and Graphics Modes

Unlike the terminal, the PC has two separate modes: graphics and alpha. In alpha mode the throughput of text on the screen is much faster. The graphics display should be off if you are displaying a lot of text on the screen.

Because switching from graphics to alpha mode causes a noticeable flicker, you may want to get rid of the flicker by using the following command on Reflection's command line:

```
SET AUTO-ALPHA NO
```

Then use Alt-8 to force the display back into alpha mode whenever necessary. Alpha mode should be used once you are finished using graphics and want to run non-graphics programs. The setting for AUTO-ALPHA can be saved to a configuration file.

You can configure the initial mode of the graphics display to be on by changing the default setting on the Terminal Configuration, Page 2 menu for **Initial graphics display**.

44.4

Controlling the Graphics Cursor

The arrow keys on the numeric keypad become graphics cursor movement keys when the keypad has been toggled into graphics mode either by you or the host application. For continuous movement, hold down the arrow keys. For faster movement, hold down Ins with the arrow key. To move diagonally, hold down two arrow keys simultaneously; e.g., holding ↑ and → moves the cursor toward the upper right corner of the screen diagonally.

Using a Mouse to Control the Graphics Cursor

Reflection 7 supports the Microsoft mouse for graphics cursor movement. Most mice on the market have drivers that are compatible with the Microsoft mouse interface.

In order to use a mouse with Reflection 7, you must first install the mouse driver software; see the manual that came with your mouse for instructions on installing the software. Also see **/P<n>** on page 17. For mice with two buttons, the left button acts as Enter, and the right button acts as Return. In response to the Esc*s4^, *read graphics cursor position with wait* sequence, the left button returns a code of 128.

Sample Graphics Session

Perform the following steps to learn how to use the graphics keypad:

1. Load Reflection 7 by typing **R7** at the DOS prompt. The display is blank, except for the alpha cursor and the screen labels. These indicate that alpha display is on. Logon to the host as you would normally do.
2. Press [Alt]-[Y] to bring up the Reflection command line.
3. Type **DIR BAR.RCL** to see if this file is available in the current directory. If it is not, copy it into the current directory from the product diskette.
4. Type **BAR.RCL** and press [Return].

   The graphics display is turned on and a graphics chart is displayed on the screen. Notice how the display changes in appearance when in graphics mode. BAR.RCL is a command file written in Reflection's command language. The DISPLAY command allows you to send graphics sequences directly to Reflection, just as a host program would.
5. Press [Return]. The graphics display is removed.
6. Press [Return] or [Esc] to exit the command line. Press [Alt]-[7] (using the 7 on the keypad) to bring up the graphics display again.
7. Press [Alt]-[5] (keypad) to turn off the alphanumeric display. Reflection clears the screen labels and the alpha cursor, leaving only the graphics memory visible.
8. Press [Alt]-[9] (keypad) to toggle the graphics cursor on. You see a fine hairline in the lower left corner of the screen. This is a portion of the graphics cursor.
9. Press [Ctrl]-[-] (keypad) to toggle the keypad to graphics mode. Graphics keypad cursor keys can now be used to move the graphics cursor.
10. Hold down [8] (keypad) and then [6] (keypad). The *crosshair* graphics cursor moves toward the center of the screen.
11. Practice moving the graphics cursor about the screen, using the arrow keys. Position the graphics cursor at the left edge of the screen, near the vertical center.
12. Press [Alt]-[5] (keypad). The alphanumeric display is toggled back on. Notice that the graphics cursor remains on the screen. Now, both the graphics and alpha displays are active, and the graphics and alpha

cursors are active at the same time. The letter **G** between the function key labels indicates that the keypad moves the graphics cursor only.

13. Notice that when you type characters, they appear at the alpha cursor position, but the graphics cursor moves when you press the arrow keys.
14. Press [Ctrl]-[-] (keypad) to turn the graphics keypad off. Notice that you can now move the alpha cursor with the arrow keys.
15. Press [Alt]-[7] to clear graphics.
16. Type out the contents of the file BAR.RCL from Reflection's command line with TYPE BAR.RCL. Notice that all the escape sequences are sent with the DISPLAY command.*
17. Press [Alt]-[5] (keypad) to toggle off the alpha display, then press [Alt]-[7] to bring back the graphics display. Now try printing the graphics screen as described below.

44.5 Printing the Graphics Screen

Press [Alt]-[3] (keypad) to send the contents of the graphics display to the printer. If there is no print buffer configured, the printer should immediately begin printing, unless it is a laser printer. If there is a buffer, expect a 3- to 20-second delay (depending on the speed of your computer) while Reflection fills it.

To avoid having your keyboard locked while a graphics screen is being printed, set up a print buffer. A 32K to 64K buffer holds an entire graphics screen. A form feed occurs if **Form Feed after Alt PrtSc** on the Printer Configuration menu is set to *YES* (the default).

If you are having problems with graphics printing, confirm that Reflection has been configured for your printer, and that your printer supports graphics printing. Reflection does not support graphics printing for the printer selections *TOSHIBA* and *OTHER*.

Some printers are available with other than their native graphics language. For instance, some Okidata Microline 182 printers use Okidata control codes and some use IBM/Epson. Always configure Reflection for the control code your printer uses.

---

* If you typed the escape sequences at the keyboard, remote mode would have to be off so that the escape sequences would not be sent to the host.

44.6

Plotting with Reflection

Escape sequences to draw graphics data can be sent to either the screen or a plotter. Host software requires that you define where sequences are to be sent before plotting. Do this according to the software you are using. Reflection allows you to plot to a disk file if a plotter is not available; see below.

If you are using PC graphics software as well as host software and you have a plotter connected directly to your PC, use *pass through* mode to prevent having to change cables and the settings of your plotter between PC application plotting and plotting from host software. See page 100.

Plotting to Disk

LOGPLOT and PLOT are Reflection commands for plotting graphics to and from disk. The following command allows you to send plotting information to disk, with or without a plotter present.

```
LOGPLOT {<filespec> [APPEND|DELETE] |OFF}
```

If **Connection** on the Plotter Configuration menu is set to a COM value, plotter data goes to the configured plotter as well as to the file specified by LOGPLOT.

If **Connection** is set to *NONE*, plotter data only goes to the file specified by LOGPLOT. In this case, Reflection emulates the plotter's responses to the host. You can choose to have Reflection emulate the HP 7440A or the HP 7550A by setting the **Plotter ID** field. When the plot is complete, type **LOGPLOT OFF** at the command line to close the plot file.

Once stored on disk, the PLOT command allows you to plot the data to the configured plotter.

Chapter 45

Programming Graphics Functions

Reflection 7 allows you to emulate the 2623A monochrome graphics, and 2627A and 2390A color graphics terminals. Most of the features of the 2393A and 2397A terminals are also included. The following list describes the features that are *not* part of Reflection:

Table 33
Unsupported Terminal Features

HP 2393A/2397A Feature	Reflection Support
Mouse/Tablet escape sequences	Only partial support
High resolution mode (640 x 400)	Not supported
Color palettes in 2397A (select 8 of 64)	Not supported
F9 to F12 support	Not supported
Split function key enhancements	Not Supported
ANSI/EM52 emulation	VT102 emulation

45.1 Graphics Operations

All graphics operations consist of specifying pixel locations on a screen to create specific images. These locations may be used to express the end points of a line, the corners of a rectangular area to be filled in by a pattern, or the location for the graphics cursor.

Graphics operations allow you (or a graphics program running on the host) to specify these points by means of predefined codes. Escape sequences are the codes that drive graphics operations. They may be entered from the PC keyboard, sent from the host computer, or embedded in a command file. Sequences may also be loaded into user keys. The following pages outline graphics escape sequences. Also see the graphics escape sequence appendix starting on page 479.

45.2 Graphics Control Functions

Reflection 7's graphic control functions include the following:

- Display and memory control from the keyboard and by escape sequences issued from the host computer
- Graphics cursor control
- Graphics text control
- Several drawing modes
- User-definable and predefined line and area patterns
- Absolute and relocatable area fill
- ASCII and binary plotting control
- A *compatibility mode* that emulates a Tek 4010 graphics terminal
- A facility that allows you to print a *snapshot* of a graphics image on selected printers
- Rubberband lines that allow you to preview lines before making them permanent
- Polygon area fill
- Support for the Microsoft mouse

45.3 Graphics Escape Sequences

Graphics functions are controlled with escape sequences that begin with ^{E}C* . The character following the asterisk, always lowercase, specifies the type of sequence. For example, ^{E}C**p* begins a plotting sequence.

Table 34
Graphics Sequence Types

$^{E}{}_{C}$*d	Display Control	$^{E}{}_{C}$*p	Line Plotting
$^{E}{}_{C}$*l	Labeling	$^{E}{}_{C}$*s	Graphics Status
$^{E}{}_{C}$*m	Drawing Mode	$^{E}{}_{C}$*t	Compatibility Mode

Within escape sequences, control codes such as $^{C}{}_{R}$ and $^{L}{}_{F}$ are generally ignored. $^{E}{}_{C}$ cancels any previous sequence that was not terminated correctly and begins a new escape sequence.

Graphics Commands

Graphics commands are composed of both upper and lowercase letters (A-Z and the ^ character). Upper and lowercase letters execute the same function; an uppercase letter terminates an escape sequence. To simplify input, you can give several commands in one sequence by using lowercase letters for all but the ending command. Sequences can be any length. For example:

```
Ec*m4a7b100,100,100,200J
```

The a, b, and J are all commands under the $^{E}{}_{C}$*m sequence (drawing mode). Parameter values, all numbers here, come before the commands they modify. Omitted commands default to their currently active value. The X and Y coordinate is separated by a comma; coordinate pairs are separated by spaces.

```
100,100 100,200
```

The following sequence plots a line between the given coordinates by selecting a line plotting sequence ($^{E}{}_{C}$*p), lifting the pen and moving it to 100,300, and selecting ASCII absolute format to specify the coordinates. The capital Z ends the sequence.

```
Ec*paf100,300 300,300Z
```

Note: Any capital letter terminates an escape sequence. Lowercase characters that are not commands are ignored.

Parameters

Parameters come from columns 2 and 3 of the ASCII table (SPACE through ?). They are typically numbers that indicate coordinates or select a particular setting. Both ASCII and binary formats are possible; they are discussed later in this section. Note that in binary formats spaces are treated as data. Parameters precede the command letter; for example:

```
EC*m2a7b256,195J
```

- ^{E}C*m defines the type of sequence being used
- *2a* selects drawing mode 2
- *7b* selects the line type associated with number 7
- *256,195J* sets the relocatable origin and ends the sequence

45.4

Restoring Graphics Defaults

^{E}C*mR restores all graphics defaults. When programming escape sequences, you may want to ensure initial default conditions for all values. All of the defaults listed below can be reset at any time with ^{E}C*mR .

Restoring Selected Graphics Defaults

^{E}C*m<default>R

If a 1 is used as the <*default*>, a subset of default values is set. Otherwise, all defaults are set. The members of this subset are marked by a double asterisk below.

Table 35
Graphics Defaults

	Parameter	Default
**	Pen condition	Down
**	Line type	1 (solid)
**	Drawing mode	2 (SET)
**	User-defined line pattern	255,1 (solid line)
**	Area fill type	2 (user-defined pattern)
**	User-defined area fill	255, 255,..., (Solid)
**	User-defined dither pattern	255, 255,..., (Solid)
**	Current dither pattern	255, 255,..., (Solid)
**	Background pen	0 (black)
**	Primary pen	7 (white)
**	Secondary pen	0 (black)

**	Boundary pen	off
**	Graphics text	off
**	Text color	primary pen
**	Text direction	1
**	Text origin	1 (left,bottom)
**	Text size	1
**	Text slant	Off
	Relocatable origin	0,0
	Alpha video	On
	Alpha cursor	On
	Graphics video*	On
	Graphics cursor	Off
	Graphics cursor address	0,0
	Rubberband line	Off
	Compatibility mode	Off
	Page full straps	0 (out)
	GIN strap	0 (CR only)

To change any of these defaults, use the graphics escape sequences that control the particular setting to be changed. To obtain information regarding the current status of various settings, use the graphics status request sequence outlined below.

Graphics Hard Reset

Initial values for Reflection match the defaults for the HP 2627A terminal, except that the graphics keypad is initially off.

$^{E}{}_{C}$*wR performs a graphics hard reset. This changes all graphics parameters to their default values. It also clears raster memory buffer and places the primary drawing pen at 0,0. It is as if a hard reset were performed for graphics only. This reset can only be performed with an escape sequence.

45.5

Graphics Status Request

To request graphics status, use $^{E}{}_{C}$**s*<*x*>^ where *x* is one of the following:

* Reflection's default for the graphics display is OFF. This can be changed on the Terminal Configuration, Page 2 menu.

HP Graphics

Table 36
Graphics Status

x	STATUS	x	STATUS
1	Terminal ID	7	Graphics text status
2	Pen position	8	Read zoom status
3	Graphics cursor position	9	Relocatable origin
4	Read cursor position and wait for key	10	Reset status
5	Display size	11	Area shading
6	Graphics capabilities	12	Dynamics

45.6 Graphics Display and Cursor Control

The graphics display on the IBM PC depends on the graphics adapter installed. The resolution of each adapter varies; see below for specific values.

The structure of the graphics display differs from alpha display. The alpha display is composed of 80 column positions and 25 rows. The HP 2627A terminal's graphics display is composed of 512 columns and 390 rows of pixels.

Your PC graphics adapter probably uses a different number of pixel rows and columns. Reflection accommodates this difference by calculating the pixel on your adapter that is closest to that on the HP 2627A. An IBM color graphics display adapter can accommodate 640 columns of pixels, but only 200 rows. An Enhanced Graphics Adapter (EGA) may give the screen 640 x 350 pixels, depending on the kind of monitor you have. Hercules provides 720 x 348.

Each pixel can be represented by a set of two numeric values that specify the row/column position of that pixel on the screen. Through the keyboard or a series of escape sequences issued by a host program, you can send the PC instructions to give a value of 1 (one) to specified pixels on the screen, making these pixels visible and forming the image you want.

It is not necessary to specify each individual pixel to create an image. Character keys and escape sequences do this for you.

Controlling the Display

To turn the graphics display ON and OFF, use the following sequences:

^{E}C*dC (ON)
^{E}C*dD (OFF)

The graphics display is initially *OFF* when Reflection is loaded. **Initial graphics display** on Terminal Configuration, Page 2 allows you to configure this setting.

PCs have two modes, graphics and alpha. Reflection initiates graphics mode whenever the graphics display is turned on. Graphics mode slows the scrolling speed of the alpha display. ^{E}C*d<z> controls the graphics display or cursor where <z> is replaced by one of the following:

Table 37
Display Control Functions

A	Clears graphics memory
B	Sets graphics memory
C	Turns on graphics display
D	Turns off graphics display
E	Turns on alphanumeric display
F	Turns off alphanumeric display
K	Turns on graphics cursor
L	Turns off graphics cursor
Q	Turns on alphanumeric cursor
R	Turns off alphanumeric cursor
S	Turns on graphic text mode
T	Turns off graphic text mode
Z	No operation is performed
<X,Y>O	Moves graphics cursor to horizontal position **X** and vertical position **Y** (relative to the bottom left corner)
<X,Y>P	Moves graphics cursor to horizontal position **X** and vertical position **Y** (relative to its present location)

Graphics Keypad Control

The graphics keypad can be turned on or off by the following escape sequences:

Graphics keypad OFF: ^{E}*c&k0O*
Graphics keypad ON: ^{E}*c&k1O*

Graphics Cursor Control

A separate cursor is used to locate points in the graphics display. The graphics cursor is used to input position data or to interact with graphics application software. It is *OFF* when Reflection is first loaded.

Graphics cursor ON: ^{E}*c*dK*
Graphics cursor OFF: ^{E}*c*dL (default)*

The cursor is initially at position 0,0 after Reflection is loaded or after a full reset. The cursor can be positioned even while it is off. You can position the cursor relative to its current position or with absolute coordinates.

Absolute coordinates: ^{E}*c*d<X,Y>O*
Relative coordinates: ^{E}*c*d<X,Y>P*

For example, ^{E}*c*d10,10O* places the cursor at the screen coordinates 10,10. The capital O ends the sequence. ^{E}*c*d10,10P* places the cursor plus 10x and 10y in relation to its current position.

Mouse Control of the Graphics Cursor

Reflection 7 supports the Microsoft mouse for movement of the graphics cursor. Most mice and graphics tablets on the market have drivers that are compatible with the Microsoft mouse interface.

Reflection 7 supports the mouse escape sequence ^{E}*c*j<n>A* where *<n>* is as follows:

0 Mouse set off line

1 Mouse online with default values (Default)

Cursor Status with Wait

The graphics keypad is automatically turned on when ^{E}*c*s4^* or ^{E}*c*s33^* is received. These escape sequences request the user to position the graphics cursor and then press a key or a mouse button, at which time Reflection transmits the cursor position. In Tektronics emulation, the **GIN** mode acts in a similar fashion. At all other times, the graphics keypad must be manually activated before the keypad arrow keys will move the graphics cursor.

Graphics Memory Control

The graphics display can be turned on or off without affecting the data in graphics memory. Set the display to all ones by setting graphics memory (a white screen) or all zeros by clearing memory (black screen).

Clear graphics memory: ^{E}C*dA
Set graphics memory: ^{E}C*dB

Note: You can print graphics memory with ^{E}C&p7sF. See page 405 for information on printing the graphics screen.

45.7 Graphics Text Escape Sequences

Typically, text is sent to the alphanumeric memory where it can be easily scrolled or altered without affecting graphics memory. In order to send text to graphics memory, you must turn graphics text mode on.

Begin graphics text mode: ^{E}C*dS
End graphics text mode: ^{E}C*dT

The graphics cursor marks where the next character appears when text is entered in graphics memory. Other sequences can be sent to scale, rotate, or otherwise alter text as it is entered in memory.

Selecting Graphics Text Styles

Size

Graphics text may be any of eight sizes. Graphics text size is determined by the escape sequence, ^{E}C*m<n>M, where <n> is a value, 1 through 8.

Slant

Graphics text may be upright or slanted to the right. The default is upright. To turn on slanted characters, type the escape sequence, ^{E}C*mO. To turn it off, type ^{E}C*mP.

Orientation

Graphics text can run from left to right, the normal orientation, or it can be upside down, running from right to left. It can also run from bottom to top or top to bottom. Orientation is set by the escape sequence, ${}^{E}C$**m<n>N*, where *<n>* is a value, 1 through 4.

Justification

Characters can be justified in relation to the current position of either the pen or the graphics cursor. The position is used as a baseline. Left bottom justification (1) is the default. Justification is changed with ${}^{E}C$**m<origin>Q*, where *<origin>* is replaced with a value between 0 and 9.

Table 38
Graphics Text Escape Sequences

Parameter	Sequence	Action
Size:	${}^{E}C$*m1M	Smallest characters
	${}^{E}C$*m2M	
	${}^{E}C$*m3M	
	${}^{E}C$*m4M	
	${}^{E}C$*m5M	
	${}^{E}C$*m6M	
	${}^{E}C$*m7M	
	${}^{E}C$*m8M	The largest characters, about one and one-half times the height of the graphics cursor
Slant:	${}^{E}C$*mO	Slanted characters
	${}^{E}C$*mP	Upright characters
Orientation:	${}^{E}C$*m1N	Normal orientation, default
	${}^{E}C$*m2N	Rotated 90 degrees counter-clockwise, running from the bottom to the top of the screen
	${}^{E}C$*m3N	Rotated 180 degrees counterclockwise, upside down
	${}^{E}C$*m4N	Rotated 270 degrees counterclockwise, from top of the screen to the bottom
Justification:	${}^{E}C$*m1Q	Text justified bottom left
	${}^{E}C$*m2Q	Text justified middle left

$^{E}{}_{C}$*m3Q	Text justified top left
$^{E}{}_{C}$*m4Q	Text justified bottom center
$^{E}{}_{C}$*m5Q	Text justified middle center
$^{E}{}_{C}$*m6Q	Text justified top center
$^{E}{}_{C}$*m7Q	Text justified bottom right
$^{E}{}_{C}$*m8Q	Text justified middle right
$^{E}{}_{C}$*m9Q	Text justified top right
$^{E}{}_{C}$*m0Q	Text justified cursor position

Note: When center justification or right justification is used (sequences $^{E}{}_{C}$*m4Q through $^{E}{}_{C}$*m9Q), no characters appear until a carriage return or a linefeed is received.

Labels

A single line of graphics text can be sent to the graphics display memory with the following sequence. The current text size, angle, slant, justification and color are all used.

*$^{E}{}_{C}$*l<text string>$^{C}{}_{R}$$^{L}{}_{F}$*

For example, *$^{E}{}_{C}$*lSales Data for 1990$^{C}{}_{R}$$^{L}{}_{F}$* would send this label to the graphics memory at the current pen position.

Graphics Text Colors

In Reflection 7 use the following escape sequence to select a color for the graphics text. If no color is selected, the primary pen color is used:

*$^{E}{}_{C}$*n<pen#>X*

The color selected is used for all subsequent text and labels until it is changed by another escape sequence or the defaults are set.

HP Graphics

Chapter 46

Graphics Drawing Modes

Drawing commands with various parameters can specify the following:

- Whether data is stored in graphics memory as 1s or 0s
- Line or area patterns for drawing vectors
- The relocatable origin position
- Graphics text settings

Graphics drawing escape sequences begin with ^{E}C**m*, followed by one or more commands and parameters.

46.1 Selecting Modes

You can draw vectors by setting, clearing, or complementing the data in graphics memory. Normally, vectors are drawn as white lines on a dark screen (bits turned off). To draw black lines on a white screen, set the graphics memory (turn bits on), select a clear or complement line type, and draw dark vectors.

Note: Color graphics provides a wider variety of drawing modes. See the table on page 443.

^{E}C**m<x>A* Selects the drawing mode.

Table 39
Drawing Modes - Monochrome Graphics

Mode	Name	Pattern	Effect
0	NOP	0,1	NOP
1	CLEAR	0	NOP
		1	Sets bit to 0
2	SET	0	NOP
		1	Sets bit to 1
3	COMP	0	NOP
		1	Complements bit
4	JAM	0	Sets bit to 0
		1	Sets bit to 1

Clear Mode

When clear mode is selected, area fill or solid lines are created by turning bits in the graphics display off. If the display is already off, the lines are invisible. Unless the line pattern is solid, all bits in the pattern are turned off and bits that are not part of the pattern are left as they were: dashed lines turn the bits that create the dashes off and leave the bits that normally create the spaces between dashes as they are, either on or off. No bits are actively turned on in clear mode.

Set Mode

Set mode turns on all the bits in the selected line or area pattern. A dashed line turns on bits to create dashes and leaves bits between dashes as they are. If bits between dashes are already on, they are left on. No bits are actively turned off. Lines drawn in set mode are invisible when the graphics display is in inverse video (turned on).

Complement Mode

Complement mode turns bits both on and off, depending on their current state. Only the bits that are on in the line pattern are affected. Dashed lines turn bits on or off during the solid pattern, but the spaces between dashes are left unaffected, whether the bits there are on or off.

Note: Complement mode is useful for selective line erasing. To prevent gaps that occur when the erased line intersects with other lines, first draw the line in complement mode. Then redraw the line in complement mode to erase it. This preserves the original graphics display. Use this technique to draw and erase temporary figures.

Jam Mode

Jam mode affects both the on and off bits in the pattern. Bits are turned off or on depending on the pattern, regardless of their initial state. Unlike the other modes, all of the bits in the pattern, whether on or off, affect the display; i.e., if the display bits are initially off, the dashes are turned on and the space between dashes is turned off.

46.2 Plotting Lines

Lines can be plotted in relation to the graphics cursor position or a pen position. Depending on the escape sequence given, the first X,Y position is located at the cursor or the absolute coordinates of 0,0 (unless the pen has been moved). To plot a vector, issue the plot command ^{E}C*p followed by coordinates. For long sequences, when plotting several lines, use Z (*no-op*) to ensure synchronization.

Initially, the pen and cursor are both in the same position: 0,0. The pen is initially in the *up* position; when it is moved to new coordinates, no line is drawn. After the first move, the pen is in the *down* position unless otherwise directed. The new point becomes the current pen position. When new coordinates are given, a line is drawn from the initial position to the indicated point. You can specify an absolute position on the screen, or a relative one, by the parameters given in the plotting sequence.

You can position the pen in relation to absolute coordinates on the screen or to relative coordinates in relation to its current position. You can specify coordinates in either ASCII or binary.

Rubberband Line

The *rubberband line* allows you to temporarily establish a line and move it to various places in graphics memory before plotting it to a particular location.

1. The current pen position is used as the origin point. Turn on rubberband line mode using ^{E}C*dM.

2. Move the rubberband line by moving the graphics cursor, either directly or through escape sequences.

3. Plot the line by establishing the current cursor location as a point. The escape sequence ^{E}c*pC makes the line between the original point and the current cursor position permanent.

ASCII Formats

ASCII formats can be of three types:

- Absolute
- Incremental
- Relocatable

In ASCII formats, the numbers 0 to 9 are used to specify X and Y coordinates. The first number given is considered the X coordinate and the second the Y coordinate. All values must be whole integers separated by commas or spaces. Any numbers, extra spaces, or commas following a decimal are ignored.

ASCII Absolute: ^{E}c*pF (default)

This is the default plotting format. Values from -16384 through 16383 may be given, but only values within the graphics display range are visible on the screen.

^{E}c*pa15,15 65,15 65,70 15,70 15,15Z draws a rectangle with the bottom left corner at 15,15 and the top right corner at 65,70. The **a** raises the pen; since no format is defined, ASCII absolute is assumed and the **F** is not required. The pen is lowered after the first coordinate pair is received. The coordinates are determined from a 0,0 starting point.

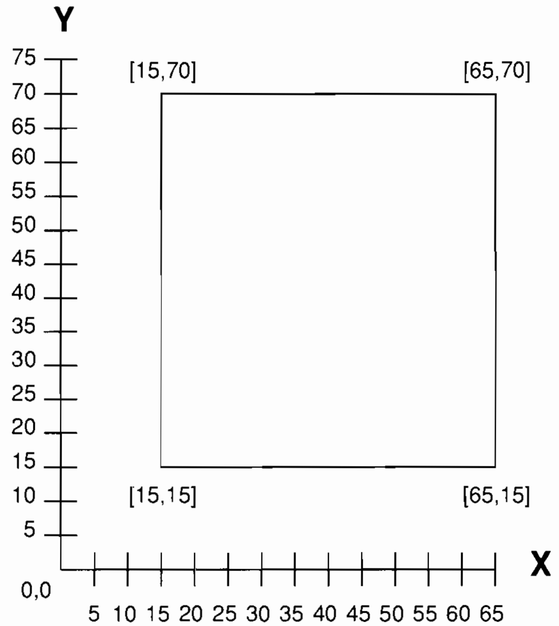


Figure 53: ASCII Absolute Plotting Format

ASCII Incremental: ^{E}C*pG

All values are added to the current pen position. This allows you to use the same escape sequence to draw a figure anywhere on the screen merely by starting at a different pen position.

^{E}C*pg10,0 0,50 -50,0 0,-50

The first set of coordinates moves the pen 10 units along the X-axis and 0 units along the Y-axis. The second moves the pen 0 units further on the X-axis and 50 units on the Y-axis, etc. All vectors are relative to the current position rather than absolute positions on the screen.

ASCII Relocatable: ^{E}C*pH

When relocatable coordinates are being used, the X and Y coordinates of the *relocatable origin* are added to all input coordinates. This allows you to draw the same figure with the same escape sequences by changing only the relocatable origin.

The box defined by the absolute coordinates 25,50 50,50 50,100 25,100 25,50 can be repeated at 75,100 100,100 100,150 75,150 75,100 by setting a relocatable origin at 50,50, and sending the same drawing sequence again.

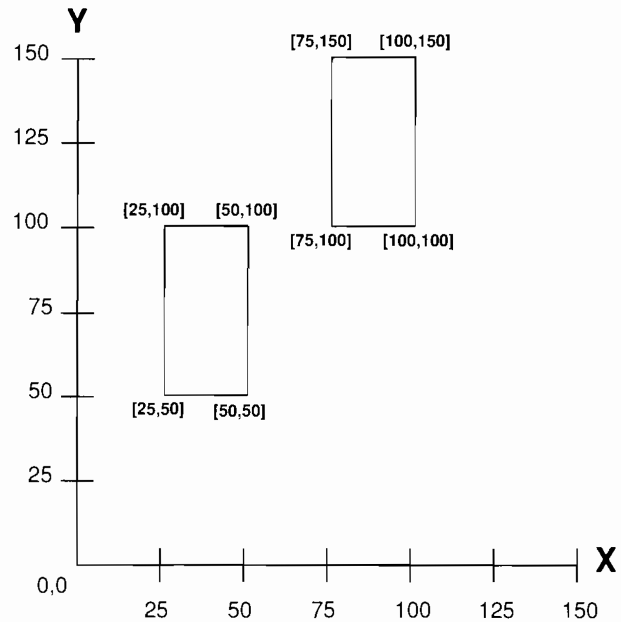


Figure 54: ASCII Relocatable Plotting Format

Binary Formats

Binary formats can be of four types:

- Absolute
- Short incremental
- Incremental
- Relocatable

In binary formats the bit patterns of ASCII characters are used to define the X and Y values. Depending on the format type, 4 to 6 bytes (characters) are required to provide each coordinate. Values can range from -16384 to 16383.

Binary Absolute: ^{E}C*pl

As in ASCII absolute, 0,0 is considered the initial origin. The given coordinates are plotted in relation to 0,0. Four bytes are required to define each point. Only points within the graphics display area are visible on the screen.

Table 40
Binary Absolute Format

BIT	7	6	5	4	3	2	1	
BYTE 1	0	1	X9	X8	X7	X6	X5	HI X
BYTE 2	0	1	X4	X3	X2	X1	X0	LOW X
BYTE 3	0	1	X9	X8	X7	X6	X5	HI Y
BYTE 4	0	1	X4	X3	X2	X1	X0	LOW Y

Table 41
Computation of Data Bytes

X = 0 =	00000	00000	Y = 0 =	00000	00000
	HI X	LOW X		HI Y	LOW Y

BYTE 1 = 01	00000 = <SPACE> HI X
BYTE 2 = 01	00000 = <SPACE> LOW X
BYTE 3 = 01	00000 = <SPACE> HI Y
BYTE 4 = 01	00000 = <SPACE> LOW Y

X = 360 =	01011	01000	Y = 180 =	00101	10100
	HI X	LOW X		HI Y	LOW Y

BYTE 1 = 01	01011 = + HI X
BYTE 2 = 01	01000 = (LOW X
BYTE 3 = 01	00101 = % HI Y
BYTE 4 = 01	10100 = 4 LOW Y

Binary Short Incremental: ^{E}C*pJ

This format allows you to alter the current pen position by incrementing the current X/Y coordinates. Under this format, only two bytes are required to give the change in position, but the range is only from -16 to

+15 from the present coordinates. The five least significant bits are added to the current pen position to obtain the new end point.

Table 42
Incremental (Short) Vector Bytes

0	1	2	3	4	5	6	7
	!	"	#	$	%	&	'
8	9	10	11	12	13	14	15
(	)	*	+	,	-	.	/
-16	-15	-14	-13	-12	-11	-10	-9
0	1	2	3	4	5	6	7
-8	-7	-6	-5	-4	-3	-2	-1
8	9	:	;	<	=	>	?

Binary Incremental: ^{E}c*pK

Using the incremental format allows you to specify a much larger range, but requires that you use six bytes of data. Delta X and delta Y can range from -16384 to +16383.

Table 43
Binary Incremental Format

BYTE1	0	1	DX14	DX13	DX12	DX11	DX10	HI DELTA X
BYTE2	0	1	DX9	DX8	DX7	DX6	DX5	MID DELTA X
BYTE3	0	1	DX4	DX3	DX2	DX1	DX0	LOW DELTA X
BYTE4	0	1	DX14	DX13	DX12	DX11	DX10	HI DELTA Y
BYTE5	0	1	DX9	DX8	DX7	DX6	DX5	MID DELTA Y
BYTE6	0	1	DX4	DX3	DX2	DX1	DX0	LOW DELTA Y

Table 44
Characters Used in Binary Data Formats

ASCII Char	BIT Pattern	ASCII Char	BIT Pattern
SP	01 0 0000	0	01 1 0000
!	01 0 0001	1	01 1 0001
"	01 0 0010	2	01 1 0010
#	01 0 0011	3	01 1 0011
$	01 0 0100	4	01 1 0100
%	01 0 0101	5	01 1 0101
&	01 0 0110	6	01 1 0110
'	01 0 0111	7	01 1 0111
(	01 0 1000	8	01 1 1000
)	01 0 1001	9	01 1 1001
*	01 0 1010	:	01 1 1010
+	01 0 1011	;	01 1 1011
,	01 0 1100	<	01 1 1100
-	01 0 1101	=	01 1 1101
.	01 0 1110	>	01 1 1110
/	01 0 1111	?	01 1 1111

Binary Relocatable: ^{E}C*pL

Under the relocatable format, the origin can be set to any position in the -16384 to +16383 range. The X and Y coordinates are then added to this origin. In this way the same figure can be drawn at different locations using the same data.

Table 45
Binary Relocatable Format

BIT	7	6	5	4	3	2	1	
BYTE 1	0	1	X14	X13	X12	X11	X10	HI X
BYTE 3	0	1	X9	X8	X7	X6	X5	MID X
BYTE 2	0	1	X4	X3	X2	X1	X0	LOW X
BYTE 4	0	1	X14	X13	X12	X11	X10	HI Y
BYTE 5	0	1	X9	X8	X7	X6	X5	MID Y
BYTE 6	0	1	X4	X3	X2	X1	X0	LOW Y

BYTE 1 = 01	11111 = ?	HI X
BYTE 2 = 01	01101 = -	LOW X
BYTE 3 = 01	01000 = (	HI Y
BYTE 4 = 01	00000 = <SP>	HI X
BYTE 5 = 01	00110 = &	LOW X
BYTE 6 = 01	01000 = (	HI Y

46.3 Drawing Patterns

Select the type of line to be used to draw vectors or to fill rectangular areas. Nine predefined line patterns can be selected as well as a user-defined line or area pattern.

Eleven line types are available. Once the pattern has been selected, all vectors are drawn with that type.

Table 46
Line Types

x	Pattern	
1	Solid line (default)	
2	User-defined line pattern	
3	Current area pattern	
4	Line #1	
5	Line #2	
6	Line #3	
7	Line #4	
8	Line #5	
9	Line #6	
10	Line #7	
11	Point plot	. (point plot) .

Point plot causes a single point to be plotted at the coordinates specified. This is useful for *scattergram* graphs.

If the area fill pattern is selected (3), the line patterns used are selected from the eight lines making up the pattern. The graphics display is divided into groups of eight rows and eight columns. Horizontal and vertical lines are drawn using the appropriate row or column from the area pattern. Diagonal lines are always drawn using a solid vector.

Defining Line Patterns

Besides the pre-defined line patterns, you can define a pattern using the bit pattern of any byte from 0 to 255. For example, 170 has the following pattern:

10101010 or ■_■_■_■_

Besides the basic pattern, you can define a scale that allows you to stretch the pattern. Each bit in the pattern is repeated X times, where X is the scaling factor. Scales range from 1 to 16. For example, 170, the example used above, could be given a scale of 2 to produce the following pattern:

■■_■■_■■_■■_

To draw with the user-defined line type, select it as the line type before plotting any lines. Vectors are drawn with dots according to the pattern. A series of horizontal or vertical lines using this line type can be used to create more complicated patterns than those possible using the area fill pattern.

EC*m<x><y>C

Defines an 8-bit segment of line pattern and a scale according to the following:

x is a number from 0 to 255 that, when converted to its binary form, illustrates the segment of line pattern.

y is a number from 1 to 16 that indicates the number of times each dot of the line pattern should be repeated, i.e., the scale.

Plotting Lines

Table 47
Plotting Functions

$^{E}{}_{C}$**p<x>*

Performs the action indicated by **x**.

A Lifts the pen.

B Lowers the pen.

C Uses the graphics cursor position as a new point.

D Draws a point at the current pen position and lifts the pen.

E Sets the relocatable origin to the current pen position.

F Data is ASCII absolute.

G Data is ASCII incremental.

H Data is ASCII relocatable.

I Data is binary absolute.

J Data is binary short incremental.

K Data is binary incremental.

L Data is binary relocatable.

Z No operation is performed.

46.4 Filling Rectangle and Polygon Areas

Both rectangle and polygon areas can be filled, although the sequences vary. See page 435 for more information on filling with color graphics.

Filling Rectangles

To fill a rectangular area with a pattern, give the lower left and upper right coordinates of the rectangle. Use either an absolute or relocatable format when giving coordinates.

Area Fill Absolute

Absolute area filling uses absolute coordinates for the area. Use the following sequence:

$^{E}{}_{C}$**m<XLL,YLL> <XUR,YUR>E*

where *<XLL,YLL>* and *<XUR,YUR>* are the absolute coordinates of the lower left and upper right corners.

Area Fill Relocatable

Relocatable area filling uses area coordinates in ASCII format. Use the following sequence:

*$^{E}{}_{C}$*m<XLL,YLL> <XUR,YUR>F*

where *<XLL,YLL>* and *<XUR,YUR>* are the relocatable coordinates of the lower left and upper right corners.

Filling Polygonal Areas

Any polygon with up to 148 sides can be filled with the current area fill pattern. Polygon filling can be affected by any of the following if they are active:

- Current drawing mode
- Current area pattern
- Current boundary pen
- Current pen selections

Use the following steps:

1. Begin the polygon area fill with *$^{E}{}_{C}$*ps*.

   All subsequent coordinates are read as vertices of the polygon.

2. Specify the vertices of the polygon using the following sequence:

   *$^{E}{}_{C}$*pas <X,Y X,Y X,Y ...>T*

   If the letter a occurs within the string of coordinates, the current polygon is closed, the pen is lifted, and a new polygon is begun.

3. Any capital letter or the close polygon area fill command (t) closes off the polygon and fills the area.

See page 435 for more information about color and polygon filling, especially *dither patterns.*

Creating Area Patterns

Area patterns are created by defining an 8 x 8 pattern. The bit pattern of each parameter given in the definition is used as the pattern for one of the 8 rows.

The area pattern is typically used to fill rectangular areas. It can also be used as the line drawing pattern. If you are drawing horizontal or vertical lines, the corresponding row or column of the area fill pattern is used as the line pattern. Diagonal vectors are always drawn with a solid line. You can also create irregular shapes by selecting the area pattern as the line type and using successive lines.

Each of the 8 parameters in the area pattern can be different. A single row of the design is defined by each.

To define an area pattern, use the following sequence:

*^{E}C*m<row 0> <row 1>...<row 7>D*

Row 7 is the top of the area and row 0 is the bottom of the area. Enter a space or comma between each of the eight numbers. The following area pattern is created by the sequence:

*^{E}C*m170,181,170,181,170,181,170,181d*

```
■_■■_■_■
■■_■_■■_
■_■■_■_■
■■_■_■■_
■_■■_■_■
■■_■_■■_
■_■■_■_■
■■_■_■■_
```

Relocatable Origins

Relocatable origins can be used for either of the following:

- Displaying the same figure at various locations
- Displaying different portions of a figure too large to fit on the PC's display

The figure data is left unaltered; Reflection adjusts where or what portion of the figure is displayed.

Relocatable origin absolute

*EC*m<X,Y>J*

Defines absolute coordinates as the new origin where X and Y are ASCII numbers between -16348 and 16383.

Relocatable origin at pen position

*EC*mK*

Defines the current pen position as the new origin.

Relocatable origin at graphics cursor

*EC*mL*

Defines the graphics cursor position as the new origin.

Chapter 47

Color Graphics

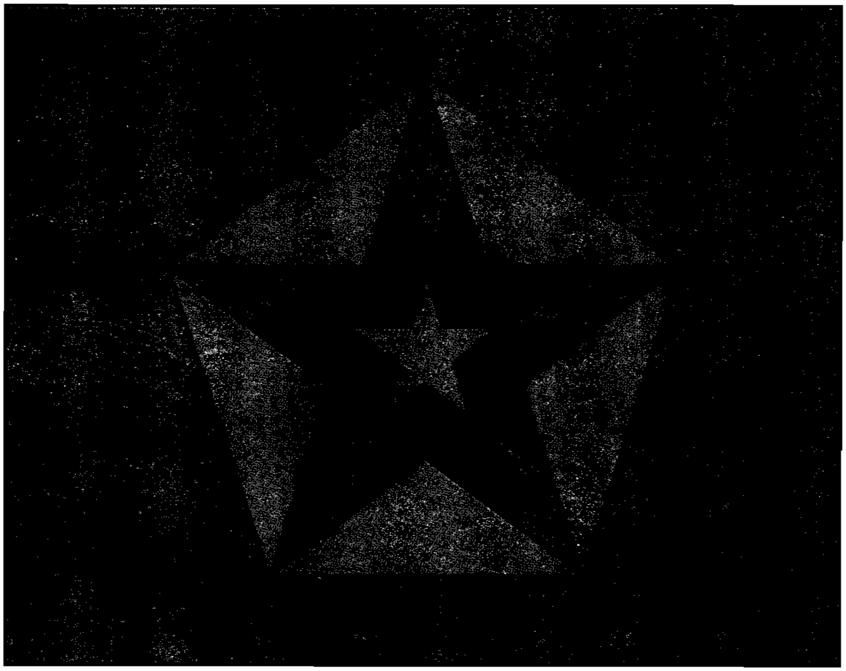


Figure 55: Color Graphics Sample

Color graphics capability provides a variety of ways to highlight graphics material. Reflection 7 allows you to emulate the HP 2627A terminal in all aspects on an IBM PC or compatible with a Color Display and an EGA or VGA card. Color graphics are only possible when you are in Reflection 7.

47.1 Basic Concepts

Graphics terminals produce color by overlaying three planes. These are either defined as the colors red, green, and blue (RGB color method), or hue, saturation, and luminosity (HSL color method). The RGB method is used as the basis for the examples below.

Because the graphics display is composed of three planes, what planes are on or off at any one time affects the color. The pixels within the plane can also be turned on and off. Varying the percentages of pixels that are on or off creates even more colors, called *dither patterns*.

For example, if all pixels of the blue plane are on, and the other planes are completely off, a blue display is presented.

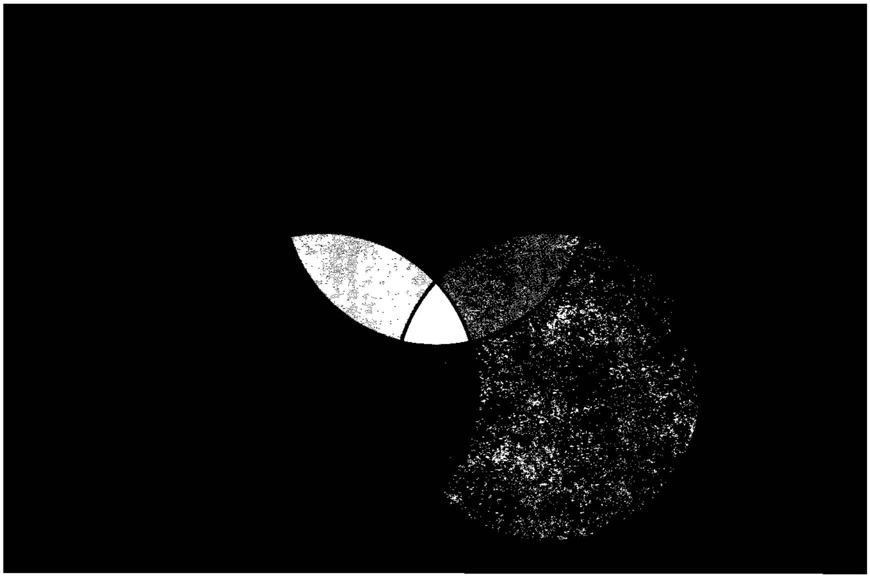


Figure 56: The Additive Color System

47.2 Pen Colors

When plotting graphics, *pens* (or colors) are selected through escape sequences. The colors selected to draw lines or fill in rectangle or polygon areas are considered pens. You can define primary and secondary pen colors as well as a boundary pen color. The primary pen color is used for most lines and area fills. The secondary pen is used in a drawing mode that needs a second color. The boundary pen is used to outline rectangles or polygons. You may also set a background color for the entire graphics display. Numbers associated with pen colors follow.

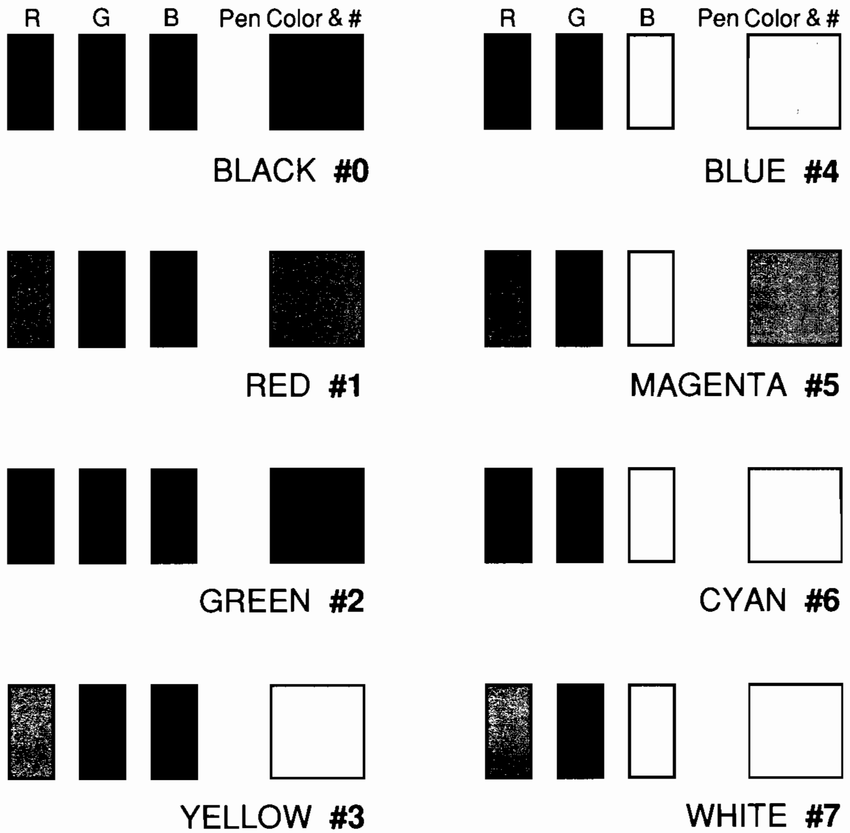


Figure 57: Basic Pen Colors

Selecting Colors for Plotting

- Select primary pen color with ^{E}c*m<pen#>X.
- Select secondary pen color with ^{E}c*m<pen#>Y. This is used for special line types and area fills according to the selected drawing mode.
- Select area boundary pen with ^{E}c*m<pen#>H.
- Select background color with ^{E}c*e<pen#>B. Black is the default. When a background pen is selected, the entire graphics memory is filled with the selected color.

- Select text color with EC*n<pen#>X.

Once selected, these pen color assignments are changed with another escape sequence or when the defaults are reset.

Selecting a Color for the Display

The entire display can be set or cleared to a color.

Clear graphics memory with EC*d<pen#>A
(0 is the default)
Set graphics memory with EC*d<pen#>B
(7 is the default)

To clear the graphics data and turn all pixels in graphics display to red, use EC*d1A. (When you use the keypad **Graph Clear** key ([Alt]-[F1]), the screen is left black.)

Area Fill with Color

Areas can be filled with either an area fill pattern or a dither pattern. Only one method can be used at any one time. Patterns use colors described under *Drawing Modes* on page 442. Dithering manipulates the densities of the color planes regardless of the pen selections.

To select dithering as the area fill type, use the area pattern escape sequence with 0 as the selection:

EC*m0G

Whichever pattern has been selected as the current dither pattern is used.

As some resolution may be lost when using dither patterns, double width lines and text (or larger) are suggested for plotting over a dither pattern.

Selecting Dither Patterns

To select a pattern type, use the following sequence:

^{E}C**m*<*dither pattern*>*W*

where <*dither pattern*> is replaced by one of the following:

#1 User-defined dither pattern

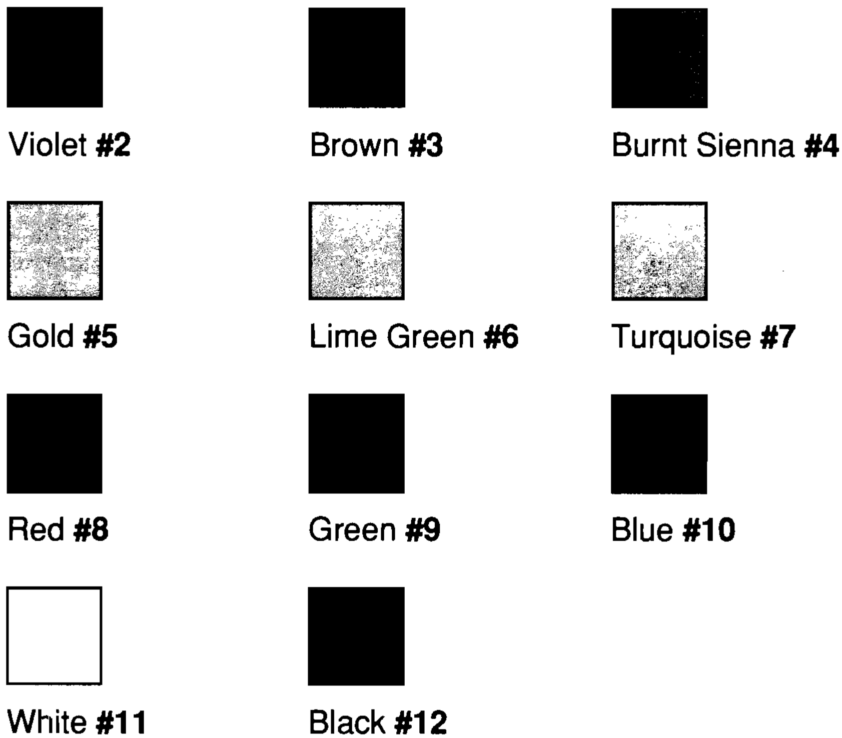


Figure 58: Predefined Dither Patterns

Creating a Dither Pattern

Additional colors can be created by mixing the amount of each plane used (or the amount of hue, saturation, and luminosity). These colors are called ***dither patterns.*** A defined number of pixels, in a specified order, are turned on and off in each plane to simulate a variety of colors.

1	13	4	16
9	5	12	8
3	15	2	14
11	7	10	6

How pixel percentage is mapped.

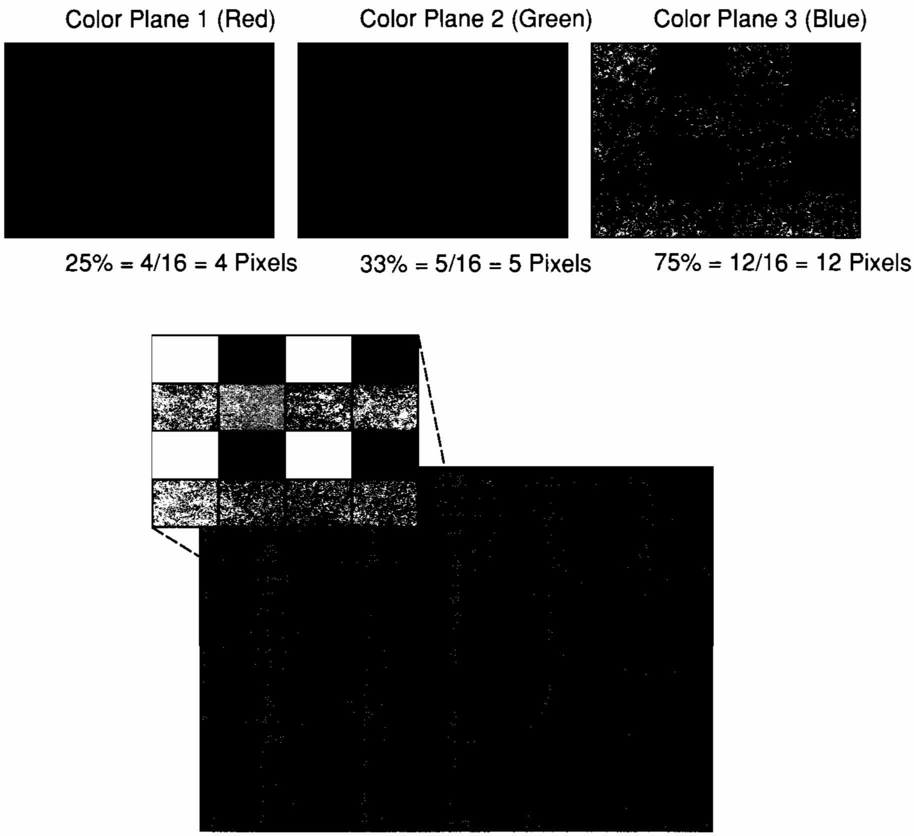


Figure 59: Dithering

To define your own dither pattern use the following steps:

1. Select the current dither pattern as the area fill type with ^{E}C**m0G*.
2. Select the user-defined dither pattern as the pattern to use with ^{E}C**m1W*.
3. Define the dither pattern by using decimal fractions that represent the density value for each color plane.

   ^{E}C**m<d1,d2,d3>V*

   In this escape sequence, *d1-d3* define the percent of each plane to include. An example might be 25% red, 50% green, and 75% blue: the escape sequence would be ^{E}C**m.25,.5,.75V*. Note that these percentages represent how many pixels are on in each plane.

Using an Area Boundary Pen

Both rectangle and polygon shapes can be outlined with a boundary pen color if desired. The boundary pen must be selected before drawing the area. Use ^{E}C**m<pen#>H* where one of the eight possible pen colors is selected.

Once the boundary pen is selected it is used to outline the area drawn. To lift the pen, use ^{E}C**pU*. The pen can then be lowered with ^{E}C**pV* at any point in the plotting.

To disable the boundary pen altogether, use ^{E}C**mH*.

47.3 Drawing Modes

The drawing modes for the 2627A terminal are expansions of the 2623A drawing modes, taking advantage of the color options available.

The escape sequence ^{E}C**m<mode>A* selects the mode. These modes are explained in the following table. See the explanation and definitions of mode types on page 419 for more information. In most cases if a bit in the pattern is turned on, the primary color is used; if it is turned off, the pixel is not affected.

Table 48
Drawing Modes - Color Graphics

Mode	Name	Pattern	Effect
0	NOP	0,1	NOP
1	CLEAR1	0	NOP
		1	Pixel = Bpen
2	JAM1	0	NOP
		1	Pixel = Cpen
3	COMP1	0	NOP
		1	Pixel = NOT Pixel
4	JAM2	0	Pixel = Spen
		1	Pixel = Cpen
5	OR	0	NOP
		1	Pixel = Pixel OR Cpen
6	COMP2	0	NOP
		1	Pixel = Pixel XOR Cpen XOR Bpen
7	CLEAR2	0	NOP
		1	Pixel = Pixel AND NOT Cpen

NOP No effect

Cpen Current pen (primary drawing color or text color) or dither pixel, in which case the color of the pixel for that dither pattern is the current color

Spen Secondary drawing pen (or color)

Bpen Background pen (or color)

Overstriking Graphics Text

In most drawing modes, overstriking a graphics text character produces an unreadable result, since only the foreground bits of the character cell are changed. To avoid this problem, use the JAM2 write mode. In JAM2 mode, all bits in the character cell are written. See page 419 for more information on selecting graphics modes.

Underlining in Reflection 7

To enable underlining, start up Reflection by typing **R7 /U** instead of the usual **R7**. With some EGA cards, this causes *snow* to appear on the screen while scrolling. See the table on page 149 for more information.

47.4

Setting Compatibility Mode

Reflection allows you to use graphics programs intended for Tektronix 4010 terminals. *Compatibility* mode presents either part (one-quarter) or all of a graphics image created for terminals with 1024 x 780 displays. To configure compatibility mode, set the **Tek 4010 compatibility** field of the Terminal Configuration, Page 2 menu to either *SCALED* or *UNSCALED*. The default is *OFF*.

Scaled mode scales the larger image down so that the entire image is displayed on the PC. Unscaled mode presents as much of the image as possible without altering the image size. By changing the origin in unscaled mode, you can view all of the image in four parts.

The following sequences set these modes programmatically:

^{E}*C&s1p0Q* - Scaled compatibility mode on
^{E}*C&s0p1Q* - Unscaled compatibility mode on
^{E}*C&s0p0Q* - Compatibility mode off (default)

P and Q settings in the above sequences determine Reflection's mode when initialized or after a hard reset.

Scaled Mode

In scaled mode the X and Y coordinates are divided by 2. The program maps the 1024 x 780 display as 512 by 390. Programs intended for the 4010 run unchanged and display the entire image with some loss in resolution.

In scaled mode text is sent to graphics memory unless redirected. The text size, origin, and direction are set to their default values, and cannot be changed. The initial origin is the bottom left portion of the display.

Unscaled Mode

In unscaled mode it is necessary to change the relocatable origin in order to view another part of the image. The relocatable origin is subtracted from the graphic data coordinates. The default is 0,0 which displays the lower left part of the image. Text in unscaled mode is sent to alphanumeric memory, unless graphics text is turned on.

Setting Terminal Straps

Along with compatibility in image size, Reflection allows you to set other straps available on the 4010 through escape sequences.

Graphic input terminator

The following straps allow you to configure what character acts as a terminator for cursor address information. You can select CR, CR and EOT, or select no terminator. The following escape sequences set this strap:

EC*t0A - Carriage return terminator (default)
EC*t1A - Carriage return and EOT terminator
EC*t2A - No terminator

Page full break

This strap determines whether a 200 millisecond break signal is sent to the host on a full page condition.

EC*t0B - No break
EC*t1B - Send break signal to host on full page

Page full busy

This determines whether the keyboard locks after a full page of text is received. The **Graphics Clear** function unlocks the keyboard. In unscaled mode this strap is ignored. The following escape sequences set this strap:

EC*t0C - Normal, no keyboard locking
EC*t1C - Keyboard locked on full page

The following escape sequences turn on scaled compatibility mode, define no terminator for input, and specify that the keyboard should be locked and a break sent after a full page:

EC&s1p0Q
EC*t2a1c1B

Graphics Data in Compatibility Mode

Each coordinate requires two bytes in compatibility mode. A 10-bit coordinate from (0-1023) is determined by the lower five bits of the two bytes. When sending data to the terminal, the Y coordinate must be sent first: <Upper Y>, <Lower Y>, <Upper X>, <Lower X>. Data is returned with the X value first: <Upper X>, <Lower X>, <Upper Y>, <Lower Y>. Bits 6 and 7 of each byte are used to indicate whether the coordinate is upper or lower.

BIT:	7	6	Value:
	0	1	Upper X or Y
	1	0	Lower X
	1	1	Lower Y

It is not always necessary to send 4 bytes for each X,Y coordinate pair. However, the following must always be sent:

- Lower X byte
- Any changed byte
- Lower Y if Upper X changes

Appendix A

Summary of HP Escape Sequences

HP Escape sequence		Function	Page
$^{E}{}_{C}0$		Home up and copy	458
$^{E}{}_{C}1$	$^{E}{}_{C}2$	Set/Clear tab	460
$^{E}{}_{C}3$		Clear all tabs	460
$^{E}{}_{C}4$	$^{E}{}_{C}5$	Set left/right margin	460
$^{E}{}_{C}9$		Reset default margins	460
$^{E}{}_{C}A$	$^{E}{}_{C}B$	Move cursor up/down	464
$^{E}{}_{C}C$	$^{E}{}_{C}D$	Move cursor right/left	464
$^{E}{}_{C}E$		Hard reset	196
$^{E}{}_{C}F$	$^{E}{}_{C}H$	Home down/up	464
$^{E}{}_{C}G$		Cursor to left margin	464
$^{E}{}_{C}I$		Horizontal tab	464
$^{E}{}_{C}J$	$^{E}{}_{C}K$	Clear display/line	460
$^{E}{}_{C}L$	$^{E}{}_{C}M$	Insert/Delete line	460
$^{E}{}_{C}N$	$^{E}{}_{C}O$	Insert/Delete with wraparound	460
$^{E}{}_{C}P$		Delete character	460
$^{E}{}_{C}Q$	$^{E}{}_{C}R$	Insert mode on/off	460
$^{E}{}_{C}S$	$^{E}{}_{C}T$	Roll display up/down	471
$^{E}{}_{C}U$	$^{E}{}_{C}V$	Next/Previous page	471
$^{E}{}_{C}W$	$^{E}{}_{C}X$	Format mode on/off	458
$^{E}{}_{C}Y$	$^{E}{}_{C}Z$	Display functions on/off	458

Summary of HP Escape Sequences

HP Escape sequence		Function	Page
^{E}C*a*		Sense cursor position, absolute	463
^{E}C*b*	^{E}C*c*	Unlock/Lock keyboard	458
^{E}C*d*		Block transfer request	458
^{E}C*f*		Disconnect	458
^{E}C*g*		Soft reset	459
^{E}C*h*		Home cursor	463
^{E}C*i*		Backtab	463
^{E}C*j*	^{E}C*k*	Display/Remove user-key menu	461
^{E}C*l*	^{E}C*m*	Memory lock mode on/off	458
^{E}C*p to* ^{E}C*w*		Default user-key values	461
^{E}C*z*		Terminal self-test	459
^{E}C*[*	^{E}C*]*	Start/End unprotected field	459
^{E}C*@*		Delay one second	458
^{E}C*^*		Primary status request	458
^{E}C_		Write nondisplaying terminator	477
^{E}C`		Sense cursor position, relative	463
^{E}C{		Start transmit-only field	459, 477
^{E}C/		Erase nondisplaying terminator	475
^{E}C~		Secondary status request	459
^{E}C*!*	^{E}C*"*	132/80 column mode, (Zentec)	467
^{E}C*)@*	^{E}C*)B*	Select base set/line drawing	471
^{E}C*)A*	^{E}C*)C*	Select space set	472
^{E}C*&a+/-<col>c+/-<row>R*		Position cursor, display relative	464
^{E}C*&a+/-<col>x+/-<row>Y*		Position cursor, screen relative	464
^{E}C*&a<col>c<row>R*		Position cursor, display	464
^{E}C*&a<col>x<row>Y*		Position cursor, screen	464
^{E}C*&bN*		Next session (LAT)	456
^{E}C*&bR*		Connection reset	456
^{E}C*&d<x>*	^{E}C*&ds<x>*	Display enhance./security	469
^{E}C*&f0B*	^{E}C*&f1B*	Store/Restore configuration	478
^{E}C*&f1m149P<n>*		Return definition	458
^{E}C*&f211P<n>*		Modify Tab	459
^{E}C*&f<parameters>*		User key control	461
^{E}C*&f<color parameters>*		User key control, Reflection 7	491
^{E}C*&f<color pair>c<video>v0L*		User key control, Reflection 7	491
^{E}C*&fR*		Reset Tab/Return	459

HP Escape sequence		Function	Page
^{E}C&f<parms>	^{E}C&f<x>E	Define/Trigger user keys	462, 461
^{E}C&j<n>D		Set bell, CR, labels	472
^{E}C&j<x>L<string>		Replace labels with message	473
^{E}C&j@		Enable user keys, no labels	461
^{E}C&jA		Display modes labels	472
^{E}C&jB		Display and enable user keys	461
^{E}C&jC		Remove message, replace labels	472
^{E}C&jR	^{E}C&jS	Enable/Disable sys, modes, user	472
^{E}C&k0A	^{E}C&k1A	Auto linefeed off/on	465
^{E}C&k0B	^{E}C&k1B	Block mode off/on	465
^{E}C&k0C	^{E}C&k1C	CapsLock off/on	465
^{E}C&k0D	^{E}C&k1D	Bell off/on	465
^{E}C&k0I	^{E}C&k1I	Previous/No parity	467
^{E}C&k0J	^{E}C&k1J	Frame rate 60/50*	466
^{E}C&k0K	^{E}C&k1K	Auto keyboard lock off/on	466
^{E}C&k0L	^{E}C&k1L	Local echo off/on	466
^{E}C&k0M	^{E}C&k1M	Modify all mode off/on	466
^{E}C&k0N	^{E}C&k1N	SPOW latch off/on	467
^{E}C&k0O	^{E}C&k1O	Graphics keypad off/on	483
^{E}C&k0P	^{E}C&k1P	Caps mode off/on	465
^{E}C&k0R	^{E}C&k1R	Remote mode off/on	467
^{E}C&k0Z		Set TRANSMIT to ALL	476
^{E}C&k1Z		Set TRANSMIT to MODIFIED	476
^{E}C&k0\	^{E}C&k1\	Terminal class HP/VT	459
^{E}C&oA		5-digit absolute cursor position	456
^{E}C&oB<command>^{C}R		Reflection command with comp. code	456
^{E}C&oC<command>^{C}R		Reflection command	456
^{E}C&oD		Refresh screen	456
^{E}C&oF<command>^{C}R		Reflection command without comp. code	456
^{E}C&oX		Clear typeahead buffer	455
^{E}C&p4^ or ^{E}C&p6^		Printer status	468
^{E}C&p7sF		Print graphics memory	484
^{E}C&p9^		Forms buffer status request	473
^{E}C&p9u<f#>p<<con>>L		Store form, length unknown	473
^{E}C&p9u<f#>p<len>L<con>		Store form	473

* This sequence is ignored by Reflection.

Summary of HP Escape Sequences

HP Escape sequence	Function	Page
$^E{}_C$&p9u<form#>pF	Display form	473
$^E{}_C$&p9u<form#>pL	Purge form	473
$^E{}_C$&p<a>d<Y>	Copy data to dest.device	467
$^E{}_C$&p<a>d<b>d<c>d<x>W<data string>	Transfer binary string	469
$^E{}_C$&p<a>d<b>d<c>dW<data string>	Transfer ASCII data string	469
*$^E{}_C$&p<form#>p9^*	Forms buffer status request (number)	473
$^E{}_C$&p<num>p20C	Record mode	468
$^E{}_C$&p<x>p<y>u<z>C	Printer control action	468
$^E{}_C$&q0L *$^E{}_C$&q1L*	Unlock/Lock configuration	466
$^E{}_C$&q1te1{0G	Set transmit pacing NONE	476
$^E{}_C$&q1te1{0H	Set receive pacing NONE	476
$^E{}_C$&q1te1{1G	Set transmit pacing XON/XOFF	476
$^E{}_C$&q1te1{1H	Set receive pacing XON/XOFF	476
$^E{}_C$&q3t0{<x>W	Page width	470
$^E{}_C$&q8te0{<0>D	Turn bell off	474
$^E{}_C$&q8te0{<1>D	Turn bell on	474
$^E{}_C$&q8te1{0R	Set RETURN=ENTER NO	476
$^E{}_C$&q8te1{1R	Set RETURN=ENTER YES	476
$^E{}_C$&q8te1{0T	Set TAB=SPACES NO	477
$^E{}_C$&q8te1{1T	Set TAB=SPACES YES	477
$^E{}_C$&q8te1{<x>A	Set $\boxed{\text{Return}}$ definition 1st char	476
$^E{}_C$&q8te1{<x>B	Set $\boxed{\text{Return}}$ definition 2nd char	476
$^E{}_C$&q<m>te0{<0/1>A	Set transmit functions NO/YES	477
$^E{}_C$&q<m>te0{<0/1>B	Set SPOW NO/YES	477
$^E{}_C$&q<m>te0{<0/1>C	Set Inhibit EOL Wrap NO/YES	475
$^E{}_C$&q<m>te0{<0/1>D	Set Line/Page mode	476
$^E{}_C$&q<m>te0{<0/1>G	Set Inhibit Handshake NO/YES	475
$^E{}_C$&q<m>te0{<0/1>H	Set Inhibit DC2 NO/YES	475
$^E{}_C$&q<m>te0{<0/1>J	Turn autoterm off/on	474
$^E{}_C$&q<m>te0{<0/1>K	Turn clearterm off/on	474
$^E{}_C$&q<m>te0{<0/1>N	Esc xfer to printer	475 *
$^E{}_C$&q<m>te1{<0/1>A	Turn auto linefeed off/on	474
$^E{}_C$&q<m>te1{<0/1>B	Turn block mode off/on	474
$^E{}_C$&q<m>te1{<0/1>C	Turn CapsLock off/on	474

HP Escape sequence		Function	Page
^{E}C&q<m>te1{<0/1>L		Turn local echo off/on	476
^{E}C&q<m>te1{<0/1>M		Turn modify all mode off/on	476
^{E}C&q<m>te1{<0/1>R		Turn remote mode off/on	476
^{E}C&q<m>te2{<x>D		Set alternate character set	473
^{E}C&q<m>te2{<x>F		Field separator character	475
^{E}C&q<m>te2{<x>L		Set forms buffer size	475
^{E}C&q<m>te2{<x>R		Set block terminator character	474
^{E}C&q<m>te2{<0/1>Z		Set transmit ALL/MODIFIED	477
^{E}C&r<n>L	^{E}C&r<n>R	Roll left/right <n> columns	471
^{E}C&r<n>U	^{E}C&r<n>D	Roll up/down <n> lines	471
^{E}C&s0A	^{E}C&s1A	Transmit functions NO/YES	467
^{E}C&s0B	^{E}C&s1B	SPOW NO/YES	467
^{E}C&s0C	^{E}C&s1C	Inhibit EOL wrap NO/YES	466
^{E}C&s0D	^{E}C&s1D	Line/Page mode	466
^{E}C&s0G	^{E}C&s1G	Inhibit handshake NO/YES	466
^{E}C&s0H	^{E}C&s1H	Inhibit DC2 NO/YES	466
^{E}C&s0J	^{E}C&s1J	Set Autoterm off/on (active)	474
^{E}C&s0K	^{E}C&s1K	Set Clearterm off/on (active)	474
^{E}C&s0N	^{E}C&s1N	Esc xfer to printer NO/YES	465
^{E}C&s0Z	^{E}C&s1Z	Check parity NO/YES	465
^{E}C&s0p0Q		Compatibility mode off	488
^{E}C&s0p1Q		Unscaled compatibility mode on	488
^{E}C&s1p0Q		Scaled compatibility mode on	488
^{E}C&v0m	^{E}C&v1m	Select RGB/HSL method	490
^{E}C&v<0-7>^		Color pair definition status	491
^{E}C&v<0-7>i		Color pair to be initialized	491
^{E}C&v<0-7>s		Color pair to be selected	491
^{E}C&v<decimal><a,b, or c>		Foreground: Red(a), Green(b), Blue(c), or Hue(a), Saturation(b), Luminosity(c)	490
^{E}C&v<decimal><x,y, or z>		Background: Red(x), Green(y), Blue(z), or Hue(x), Saturation(y), Luminosity(z)	491
^{E}C&w12F	^{E}C&w13F	Display on/off	470
^{E}C&w5f<n>W		Page width (active value only)	470
^{E}C&w6f<n>X		Assign 80/132 *	471
^{E}C&x<x>C		Cursor position mode	463
^{E}C*d0E	^{E}C*d1E	Normal/inverse display	478
^{E}C*d0Q	^{E}C*d1Q	Underline/block cursor	478

* This sequence only works with a 132 column adapter installed.

HP Appendices

HP Escape sequence		Function	Page
*EC*d<X,Y>O*		Move graphics cursor absolute	484
*EC*d<X,Y>P*		Move graphics cursor relative	484
*EC*d<pen#>A*		Clears graphics memory	484
*EC*d<pen#>B*		Sets graphics memory	484
*EC*dC*	*EC*dD*	Graphics display on/off	484
*EC*dE*	*EC*dF*	Turn alpha display on/off	484
*EC*dK*	*EC*dL*	Turn graphics cursor on/off	483
*EC*dM*	*EC*dN*	Turn rubberband line on/off	484
*EC*dQ*	*EC*dR*	Turn on/off alpha cursor	483
*EC*dS*	*EC*dT*	Turn on/off graphics text mode	485
*EC*e<pen#>B*		Select background pen color	490
*EC*j0A*	*EC*j1A*	Mouse offline/online	483
*EC*l<text>CR*		Graphics labels	485
*EC*m0A*		No effect	486
*EC*m0G*		Dither pattern as polygon pattern	487
*EC*m11B*		Point plot	428
*EC*m1A*		Clear (turn off) graphics bits	486
*EC*m1B*		Solid line (default)	428
*EC*m1G*		Solid area pattern	487
*EC*m1N*		Rotated 0 degrees	485
*EC*m1W*		User-defined dither pattern	489
*EC*m2A*		Set (turn on) graphics bits	486
*EC*m2B*		User-defined line pattern	428
*EC*m2G*		User-defined area fill pattern	487
*EC*m2N*		Rotated 90 degrees	485
*EC*m3A*		Complement (toggle) bits	486
*EC*m3B*		Current area pattern	428
*EC*m3N*		Rotated 180 degrees	485
*EC*m4A*		Jam (according to data)	486
*EC*m4N*		Rotated 270 degrees	485
*EC*m<X,Y>J*		Relocatable origin absolute	488
*EC*m<d1,d2,d3>V*		Define dither pattern	490
*EC*m<lower> <upper>E*		Rectangle fill absolute	487
*EC*m<lower> <upper>F*		Rectangle fill relocatable	487
*EC*m<mode>A*		Select drawing mode (0-7)	485
*EC*m<n>B*		Select line type (4-10)	428
*EC*m<n>G*		Predefined area pattern (3-10)	487

HP Escape sequence	Function	Page
*EC*m<origin>Q*	Justification (0-9)	485
*EC*m<pen#>H*	Set area boundary pen (0-7)	489
*EC*m<pen#>X*	Select primary pen color	490
*EC*m<pen#>Y*	Select secondary pen color	490
*EC*m<row 0>...<row 7>D*	Define an area pattern	487
*EC*m<x><y>C*	Define a line pattern	487
*EC*m<n>M*	Character size (1-8)	485
*EC*m<n>W*	Predef. dither patterns (2-12)	489
*EC*mK*	Relocatable origin at pen	488
*EC*mL*	Reloc. origin at graphics cursor	488
*EC*mO*	Slanted characters	485
*EC*mP*	Upright characters	485
*EC*m<default flag>R*	Restore graphics defaults	479
*EC*n<pen#>X*	Graphics text colors	490
*EC*pA* *EC*pB*	Lift/Lower pen	486
*EC*pC*	Graphics cursor as point	486
*EC*pD*	Draw point at pen, lift pen	486
*EC*pE*	Set relocatable origin at pen	486
*EC*pF*	Data is ASCII absolute	486
*EC*pG*	Data is ASCII incremental	486
*EC*pH*	Data is ASCII relocatable	486
*EC*pI*	Data is binary absolute	486
*EC*pJ*	Data is binary short incremental	486
*EC*pK*	Data is binary incremental	486
*EC*pL*	Data is binary relocatable	486
*EC*pS* *EC*pT*	Start/End polygon area fill	488
*EC*pU* *EC*pV*	Lift/Lower boundary pen	489
*EC*pZ*	No operation is performed	486
*EC*s-<n>^*	Terminal capabilities	492
*EC*s1^*	Read device ID	481
*EC*s2^*	Read pen position	479
*EC*s3^*	Read graphics cursor position	482
*EC*s4^*	Read position, wait for key	482
*EC*s5^*	Read display size	481
*EC*s6^*	Read graphics capabilities	480
*EC*s7^*	Read graphics text status	482

HP Appendices

Summary of HP Escape Sequences

HP Escape sequence		Function	Page
EC*s8^		Read zoom status	483
EC*s9^		Read relocatable origin	483
EC*s10^		Read reset status	483
EC*s11^		Read area shading	479
EC*s12^		Read modifications	482
EC*s12345^	EC*s12346^	Request 9/11-char serial no.	457
EC*s12347^		Request 14-char serial no.	456
EC*s32^		Read mouse status	482
EC*s33^		Request serial no. (PC 2622)*	457
EC*s33^		Read cursor position (mouse)	482
EC*t0A		Set carriage return terminator	489
EC*t0B		Page full break, no break	489
EC*t0C		Page full busy, normal	488
EC*t1A		Set return and EOT terminator	489
EC*t1B		Page full break, break	489
EC*t1C		Page full busy, lock	488
EC*t2A		Disable all terminators	488
EC*wR		Graphics hard reset	479
EC*y^		Report DCS capability	478

* This sequence does not work when using a graphics terminal emulation.

Appendix B

HP Escape Sequences

An escape sequence is a character string beginning with the ASCII escape character (decimal 27). Such strings constitute a command, usually to the terminal, to perform some action. Escape sequences are usually sent to a terminal from a host computer, but they can also be entered through the keyboard. Graphics escape sequences are listed in a separate appendix.

47.4

A Word about Notation

In this reference, escape sequence parameters you must supply are enclosed in angled brackets. For example, the <*x*> parameter in the escape sequence $^{E}{}_{C}$*&d<x>* asks you to supply one of several codes specifying the display enhancement you want. The parameter <*n*> or <*num*> usually represents a repetition count or a screen coordinate.

When coding escape sequences, take care to distinguish between upper and lowercase characters.

The control codes are represented by $^{E}{}_{C}$, $^{C}{}_{R}$, $^{L}{}_{F}$, etc.

47.4

Reflection Escape Sequences

Reflection escape sequences are ignored if HOST-INITIATED-COMMANDS is set to *NO*.

Clear typeahead buffer

$^{E}{}_{C}$*&oX* —Clears the contents of the typeahead buffer.

Next session (LAT)

^{E}*C&bN* —Selects the next session when using LAT, CTERM, or Telnetanager.

Connection reset

^{E}*C&bR* —Performs one of the following, depending on the type of connection:

- Resets an internal Hayes Smartmodem 1200B modem card by setting the *out 1* bit of the modem control register to 1 for about 2 seconds.
- Suspends a LAN session.

Five-Digit absolute cursor position

^{E}*C&oA* — Returns a 5-digit cursor position, reflecting the absolute value of the current position.

Reflection command initiator

^{E}*C&oC<command>*^{C}*R* — Where *<command>* is any Reflection command. The *<command>* is passed to the command language processor as if it had been entered from the keyboard or a batch file. (See the *Command Language Manual* and below.)

Reflection indicates the success or failure of the command by returning a completion code of **S** or **F** (unless DISABLE-COMP-CODES is set to *YES*; see the Reflection *Command Language Manual*).

Use ^{E}*C&oB<command>*^{C}*R* to *always* have a completion code returned. The **S** or **F** is returned according to the rules of a type 3 block transfer (see pages 495 and 495).

Use ^{E}*C&oF<command>*^{C}*R* to *never* have a completion code returned. Reflection acts just like a terminal in this case, and ignores Reflection specific escape sequences.

Refresh screen

^{E}*C&oD* — Causes a refresh of the Reflection screen.

Serial Number Request

The host computer can request the serial number and version of the Reflection program running on the PC by sending the escape sequence ^{E}*C*s12347^*.* After exchanging the appropriate block transfer

---

* ^{E}*C*s12346^* and ^{E}*C*s33^* are obsolete versions of this sequence. ^{E}*C*33^* conflicts with a graphics escape sequence.

handshake, Reflection sends a 14-character serial number in the form IØ1+2ØØI123456. See $SERIAL in the Reflection *Command Language Manual.* If HOST-INITIATED-COMMANDS is set to *NO*, a serial number request is treated as a terminal ID request.

Note: If this sequence is sent to a terminal, the terminal ID response is sent in place of the Reflection escape sequence.

To return a sequence of the form WRQ-12345, use Ec*s12345^. Previous versions of Reflection may use this exclusively. A nine-character serial number is returned in the form WRQ-12345, where the *WRQ-* is fixed and the numeric portion is variable.

Type of workstation being used

- *H* HP150
- *I* IBM
- *M* Macintosh
- *W* Wang

Product number

- *01* Reflection 1
- *02* Reflection 2
- *03* Reflection 3
- *04* Reflection 4
- *07* Reflection 7

PLUS or non-PLUS

- \+ Program is a *PLUS* version: it has backup/restore capability and LAN support.
- \- Non-PLUS versions have a minus sign.

Version number

For example, *4.ØØ*.

License type

I is an individual license, *S* is a server license, and *L* is a LAN license.

47.4 Terminal Control

Block transfer request

^{E}*Cd* — Requests a block transfer from display memory.

Delay one second

^{E}*C@* — Delays one second.

Disconnect

^{E}*Cf* — Disconnects (lowers DTR line for 50 milliseconds). On a LAN connection, drops the virtual connection.

Display functions on/off

^{E}*CY* — Turns display functions mode on.

^{E}*CZ* — Turns display functions mode off.

Format mode on/off

^{E}*CW* — Turns format mode on.

^{E}*CX* — Turns format mode off.

Hard reset

^{E}*CE* — Performs a hard reset. See page 196.

Home up and copy

^{E}*C0* — Homes cursor and copies from memory to the "to" devices.

Lock/Unlock keyboard

^{E}*Cc* — Locks the keyboard.

^{E}*Cb* — Unlocks the keyboard.

Memory lock on/off

^{E}*Cl* —Turns memory lock mode on.

^{E}*Cm* — Turns memory lock mode off.

Primary terminal status

Ec ^ — Requests primary terminal status.

Return definition

^{E}*C&f1m149P<n>* — Modifies the action performed by Return. <*n*> may be one of the following:

null - Return equals Return
!149 - Return equals Return
!154 - Return equals Enter

Reset Return and Tab

^{E}*C&fR* — Return is reset to non-volatile memory selection; Tab is reset to a tab.

Modify Tab

^{E}*C&f211P<n>* — Modifies the action performed by [Tab]. <*n*> may be one of the following:

null - Numeric keypad [Tab] equals [Tab]
!149 - Numeric keypad [Tab] equals [Return]
!154 - Numeric keypad [Tab] equals [Enter]
!150 - Numeric keypad [Tab] equals [Tab]
!211 - Numeric keypad [Tab] equals [Tab]

Secondary terminal status

^{E}*C~* — Requests secondary terminal status.

Self test

^{E}*Cz* — Initiates terminal self-test (always successful).

Soft reset

^{E}*Cg* — Performs a soft reset. See page 206.

Start/End unprotected field

^{E}*C[* — Starts an unprotected field.

^{E}*C]* — Ends an unprotected field.

Terminal class

^{E}*C&k<n>* — Switches between HP and VT terminal class, as follows:

```
<n>  = 0 change to HP
<n>  = 1 change to VT
```

Changing terminal class reactivates the last saved or last activated *
configuration values for the new terminal class. ^{E}*C&k0* is disabled if *
SET ENABLE-SWITCHING is set to NO. *

Transmit-only field

^{E}*C{* — Start a *transmit only* field. This sequence is treated by Reflection as *start an unprotected field* (^{E}*C[*) unless the terminal emulation is HP 2624B, HP 2393A, HP 2397A, or HP 700/94.

47.4

Setting Margins and Tabs

Clear all tabs

$^{E}C3$ — Clears all tab stops.

Clear/Set tab at cursor

$^{E}C2$ — Clears the tab stop at the current cursor position.

$^{E}C1$ — Sets a tab stop at the current cursor position.

Reset default margins

$^{E}C9$ — Sets the left margin in column 1, and the right margin in column 80.

Set left/right margin

$^{E}C4$ — Sets the left margin at the current cursor position.

$^{E}C5$ — Sets the right margin at the current cursor position.

47.4

Editing Display Memory

Clear display/line

^{E}CJ — Clears display from cursor through end of memory.

^{E}CK — Clears display from cursor through end of line.

Delete character

^{E}CP — Deletes character.

Insert/Delete line

^{E}CL — Inserts line.

^{E}CM — Deletes line.

Insert mode on/off

^{E}CQ — Starts insert character mode.

^{E}CR — Ends insert character mode.

Insert/Delete with wraparound

^{E}CN — Insert character with wraparound.

^{E}CO — Delete character with wraparound.

47.4

User Key Control

Default [F1]

^E^Cp — Default definition of user key [F1].

Default [F2]

^E^Cq — Default definition of user key [F2].

Default [F3]

^E^Cr — Default definition of user key [F3].

Default [F4]

^E^Cs — Default definition of user key [F4].

Default [F5]

^E^Ct — Default definition of user key [F5].

Default [F6]

^E^Cu — Default definition of user key [F6].

Default [F7]

^E^Cv — Default definition of user key [F7].

Default [F8]

^E^Cw — Default definition of user key [F8].

Define user keys

^E^C&f<parms> — Defines user keys (see *Defining the User Keys Programmatically* below for detailed instructions).

Display user key labels

^E^C&jB — Displays user key labels and enables user keys.

Trigger user key

^E^C&f<x>E — Triggers function key *<x>*. When received, Reflection acts exactly as if the function key *<x>* had been pressed. Only the current user function keys are triggered.

User keys enabled, no labels

^E^C&j@ — Enables user keys, remove labels.

User key menu displayed

^E^Cj — Displays the user key definition menu.

User key menu removed

^E^Ck — Removes the user key definition menu.

47.4

Defining the User Keys Programmatically

The host system can send an escape sequence to Reflection to define a user key. This escape sequence takes the following form:

^{E}c&f<attr><key><label length><string length><label><string>

Up to 160 characters can be loaded into a user key from the host.

The following table defines the parameters in the user key definition escape sequence. The parameters *<attr>*, *<key>*, *<label length>*, and *<string length>* may appear in any order. However, the letter that identifies the last parameter must be uppercase, and all the preceding identifiers must be lowercase.

Table 49
User Key Escape Sequence Parameters

Variable	Parameters	Default
<attr>	Øa (Normal) 1a (Local only) 2a (Transmit only)	Øa
<key>	key number, 1k-8k	1k
<label length>	Ød-16d	Ød
<string length>	ØL-8ØL -1 erases the field definition	
<label>	a string of exactly <label length> characters	
<string>	a string of exactly *<string length> characters*	

If a parameter is omitted, the default value is used. If a value of zero is used for the *<label length>* or *<string length>*, then the current contents of the label or string are left unchanged. The *<label>*, if its length is not zero, must precede the *<string>*.

The following example shows two equivalent escape sequences that assign the string **HELLO MGR.DEV** to F1. The attribute is normal (**N**), the label reads **LOG ON**. The periods represent spaces, and are required so that the function begins on the function definition line, not in the label.

```
Ec&f0a1k16d13L...LOG.....ON...HELLO MGR.DEV
Ec&f1k0a13l16D...LOG.....ON...HELLO MGR.DEV
```

47.4 The User Key Menu and the Host

Once the user key menu has been displayed, the host system can read it by sending escape sequences. Enter the escape sequences to put Reflection in block/page mode, home up, and perform a block transfer.

Reflection responds by sending one ^{E}*C&f...* escape sequence for each of the eight user keys. This capability is used by some host programs to save the current user key definitions before changing them, so that they can be reloaded with their original definitions when the program terminates.

47.4 Cursor Positioning

For more information, see *HP Cursor Positioning and Sensing* on page 507. See page 455 for a 5-digit cursor position response.

Backtab

^{E}*Ci* — Performs a Backtab.

Cursor position absolute

^{E}*Ca* — Senses cursor position, absolute.

Cursor position relative

^{E}*C'* — Senses cursor position, relative to the current display.

Cursor position mode

^{E}*C&x<x>C* — *<x>* = 1 sets; *<x>* = Ø clears *send cursor position mode.* When this mode is set and [Enter] or a transmit-only user key is pressed, the absolute cursor position report escape sequence is added to the beginning of the transmitted block.

Home cursor

^{E}*Ch* — Homes the cursor. In format mode the cursor is positioned in the first unprotected field. *Transmit only* fields are ignored if the terminal emulation is HP 2624B, HP 2393A, HP 700/94, or HP 2397A.

Home down/up

^{E}*CF* — Homes down the cursor.

^{E}*CH* — Homes up the cursor. In format mode, the cursor is positioned in the first unprotected or transmit-only field.

Move cursor up/down

^{E}*CA* — Moves the cursor up one row.

^{E}*CB* — Moves the cursor down one row.

Move cursor right/left

^{E}*CC* — Moves the cursor right one column.

^{E}*CD* — Moves the cursor left one column.

Move cursor to left margin

^{E}*CG* — Moves cursor to left margin.

Tab

^{E}*CI* — Performs a horizontal tab.

Cursor Positioning by Row and Column

Row and column numbers are expressed relative to zero in the following sequences. The column and row commands can be switched, in which case the C (or X) would be in uppercase and the other lower.

Position cursor (screen)

^{E}*C&a<col>x<row>Y* — Moves the cursor to *<col>* and *<row>*, relative to the currently displayed screen.

Position cursor (display memory)

^{E}*C&a<col>c<row>R* — Moves the cursor to *<col>* and *<row>*, relative to the start of display memory.

Position cursor relative (screen)

^{E}*C&a±<col>x±<row>Y* — Moves the cursor *<col>* columns (left or right) and *<row>* rows (up or down), relative to the current cursor position. Positioning is limited to screen size. The ± may be either a + or - sign.

Position cursor relative (display memory)

^{E}*C&a±<col>c±<row>R* — Moves the cursor *<col>* columns (left or right) and *<row>* rows (up or down), relative to the current cursor position. The ± may be either a + or -.

47.4

Mouse Control

Mouse on/off line

^{E}C*j0A — Mouse set off line.

^{E}C*j1A — Mouse online with default values (Default).

47.4

Configuration Control

The escape sequences that begin with ^{E}C&k... and ^{E}C&s... can be of any length, as long as the last character is a capital letter. This allows a number of parameters to be specified in one sequence. Sequences are listed here with an ending capital letter as if they are issued alone.

Auto linefeed off/on

^{E}C&k0A — Turns auto linefeed off.

^{E}C&k1A — Turns auto linefeed on.

Bell off/on

^{E}C&k0D — Turns bell off.

^{E}C&k1D — Turns bell on.

Block mode off/on

^{E}C&k0B — Turns block mode off.

^{E}C&k1B — Turns block mode on.

CapsLock off/on

^{E}C&k0C — Turns CapsLock off.

^{E}C&k1C — Turns CapsLock on.

Caps mode off/on

^{E}C&k0P — Turns caps mode off.

^{E}C&k1P — Turns caps mode on.

Check parity NO/YES

^{E}C&s0Z — Sets **Check parity** to NO.

^{E}C&s1Z — Sets **Check parity** to YES.

Escape transfer to printer YES/NO

^{E}C&s0N — Sets **Esc Xfer** to NO.

^{E}C&s1N — Sets **Esc Xfer** to YES.

Frame rate = 60/50

Ec &k0J — Sets frame rate = 60 (ignored by Reflection).

Ec&k1J — Sets frame rate = 50 (ignored by Reflection).

Inhibit DC2 NO/YES

Ec&s0H — Sets **Inhibit DC2** to NO.

Ec&s1H — Sets **Inhibit DC2** to YES.

Inhibit EOL wrap NO/YES

Ec&s0C — Sets **Inhibit EOL wrap** to NO.

Ec&s1C — Sets **Inhibit EOL wrap** to YES.

Inhibit handshake NO/YES

Ec&s0G — Sets **Inhibit handshake** NO.

Ec&s1G — Sets **Inhibit handshake** YES.

Keyboard lock off/on

Ec&k0K — Turns auto keyboard lock off.

Ec&k1K — Turns auto keyboard lock on.

Line/Page mode

Ec&s0D — Sets line mode.

Ec&s1D — Sets page mode.

Local echo off/on

Ec&k0L — Turns local echo off.

In the case of a host that does not echo, the combination of having local *
echo off when remote mode is on means that escape sequences are not *
performed. *

Ec&k1L — Turns local echo on.

Lock/Unlock configuration

Ec&q1L — Locks the configuration menu; also locks modify all, block mode, remote mode, and auto linefeed.

Ec&q0L — Unlocks the configuration.

Modify all off/on

Ec&k0M — Turns modify all mode off.

^{E}C*&k1M* — Turns modify all mode on.

Previous/No parity

These sequences are functional only for the following terminal IDs: 2392A, 2394A, 2390A, 2397A, 2393A, 70092, 70094

^{E}C*&k0I* — Sets previous parity (7-bit).

^{E}C*&k1I* — Sets no parity (8-bit).

Remote mode off/on

^{E}C*&k0R* — Turns remote mode off.

^{E}C*&k1R* — Turns remote mode on.

SPOW (space overwrite) latch off/on

^{E}C*&k0N* — Turns the SPOW latch to off.

^{E}C*&k1N* — Turns the SPOW latch to on.

SPOW NO/YES

^{E}C*&s0B* — Sets **SPOW** to NO.

^{E}C*&s1B* — Sets **SPOW** to YES.

Transmit functions NO/YES

^{E}C*&s0A* — Sets **Transmit functions** to NO.

^{E}C*&s1A* — Sets **Transmit functions** to YES.

Zentec mode 132/80*

^{E}C*!* — Sets 132-column mode.

^{E}C*"* — Sets 80-column mode.

47.4 Printer and Data Operations

Copy to destination device

^{E}C*&p<a>d<Y>* — Copies the amount of data implied by *<Y>* to the destination device *<a>*. If no destination device is specified, the currently selected "to" devices are assumed. Only numbers 4 and 6 are recognized; both refer to a printer. *<Y>* may be one of the following:

```
B = the line containing the cursor
F = the current screen page, from the cursor
M = all of display memory, from the cursor
```

* Available only with a supported 132-column adapter.

Printer control action

${}^{E}C$&p<x>p<y>u<z>C — Performs the action specified by <z> on the device(s) specified by <y>. The device codes of 4 or 6 indicate the printer.

The <z> action codes are as follows:

0	Generates 1 form feed.
1	Skips <x> lines.
2-10	Generates 1 form feed.
11	Turns on log bottom mode.
12	Turns on log top mode.
13	Turns off either logging mode.
14	Prints normal characters.
15	Prints expanded characters.
16	Prints compressed characters.
17	Turns on report mode.
18	Turns on metric report mode.
19	Turns off any report mode.
20	Turns on record mode.

If the action is *skip line*, the <x> value can be entered to specify the number of lines to skip. If the action is record mode, the <x> value defines the character that turns off record mode. The <x> value is the decimal equivalent of the ASCII character, in the range 0-127.

Request printer status

*${}^{E}C$&p4^* or *${}^{E}C$&p6^*— Either of these escape sequences requests the status of the printer. Reflection returns its *device status* response (see page 515).

Record mode

${}^{E}C$&p<num>p20C — Record mode. In record mode, data received from the host is passed through to the specified "to" devices without appearing on the screen. The *<num>p* parameter is optional. It is the decimal equivalent of an ASCII character that terminates record mode (the terminator cannot be a NULL, but can be a control character).

Data received while in record mode should consist of lines ending in linefeeds. The terminator character, if specified, must follow a linefeed or a 256-character block without linefeeds; otherwise, the terminator is ignored by Reflection, which remains in record mode.

If a terminator is not specified, record mode can be switched off only from the terminal. When record mode is switched off (either from the terminal or upon recognizing a terminator), Reflection transmits **S** (Success) or **F** (Failure), unless DISABLE-COMP-CODES has been set to *YES*; see the Reflection *Command Language Manual.* This transmission is a type 3 block transfer (see pages 495 and 495).

Transfer ASCII string

$^{E}_{C}$*&p<a>d<b>d<c>dW<data string>* — Transfer the ASCII data string from the host computer to the destination devices *<a>*, *<b>*, and *<c>*.

Any number of destination devices can be specified. If no device is specified, the currently selected "to" devices are assumed. The data string is terminated by either the 256th character or by an ASCII linefeed character.

Transfer binary string

$^{E}_{C}$*&p<a>d<b>d<c>d<x>W<data string>* — Transfers the *<data string>*, which is *<x>* bytes in length, from the host computer to the devices specified by *<a>*, *<b>*, and *<c>*. This is a binary transfer, so the data string can contain non-ASCII characters. Only the printer may be specified; the currently selected "to" devices are assumed.

47.4 Display Enhancements

Display enhancements

$^{E}_{C}$*&d<x>* — Begins display enhancements, where *<x>* is a character from the table below.

The HP 2392A terminal supports 16 combinations of display enhancements, including blinking, inverse video, underline, and half intensity. See page 149 for information on specific adapters and the display enhancements they support.

Some of the combinations are not supported by the monochrome display adapter. The underline and inverse video attributes cannot be combined.

Any of the sixteen possible combinations can be selected by the escape sequence $^{E}_{C}$*&d<x>*, where *<x>* is one of the following:

@	No enhancement
A	Blinking
B	Inverse video
C	Blinking, inverse video
D	Underline
E	Blinking, underline
F	Inverse video, underline
G	Blinking, inverse video, underline
H	Half bright
I	Half bright, blinking
J	Half bright, inverse video
K	Half bright, blinking, inverse video
L	Half bright, underline
M	Half bright, blinking, underline
N	Half bright, inverse video, underline
O	Half bright, blinking, inverse video, underline

Security display enhancements

EC&ds<x> — Begins display enhancements, where <*x*> is a character from the table above, with the security enhancement (**S**) combined with another enhancement.

47.4 Screen Blanking

Display on/off

EC&w12F — Turns the display on.

EC&w13F — Turns the display off.

47.4 Display Window

Assign page width (non-volatile)

EC&q3t0{<x>W — Assigns <*x*> columns as the page width, where *
<*x*> is a number in the range 1-9999. This sequence changes the value *
of the **Right margin** on the Terminal Configuration Page 2 menu, and is *
maintained after a hard reset. *

Assign page width (active value only)

EC&w5f<x>W — Assigns <*x*> columns as the line length, where <*x*> *
is a number in the range 1-9999. This sequence is equivalent to pressing *

the **RIGHT MARGIN** *key on the margins/tabs/col function keys. This value is not maintained after a hard reset.*

Zentec 132/80 column mode *

EC! — Assigns 132 column mode, 700 series terminals *only.*

Assign 132/80 column mode *

EC&w6f<x>X — Assigns <x> as the column width, where <x> is either 80 or 132.

47.4 Display Control Operations

Next/Previous page

ECU — Displays next page of display memory.

ECV — Displays previous page of display memory.

Roll display up/down

ECS — Rolls display up one line.

ECT — Rolls display down one line.

Roll display down/up <n> lines

EC&r<n>D — Rolls the display down <n> lines.

EC&r<n>U — Rolls the display up <n> lines.

Roll display left/right

EC&r<x>L — Rolls the display left <x> columns, if the page width is greater than 80 columns.

EC&r<x>R — Rolls the display right <x> columns, if the page width is greater than 80 columns.

None of the following escape sequences are supported when **Terminal ID** is ...302A (the Reflection 1 default).

...set ... character set as the alternate character set.

HP Escape Sequences

Auto linefeed on/off

EC&q<m>te1{0A — Turns Auto Linefeed off (<*m*> = 4-7).

EC&q<m>te1{1A — Turns Auto Linefeed on (<*m*> = 4-7).

Autoterm on/off (active and non-volatile values)

EC&q<m>te0{0J — Turns Autoterm off (<*m*> = 4-7).

EC&q<m>te0{1J — Turns Autoterm on (<*m*> = 4-7).

Autoterm off/on (active value only)

EC&s0J — Turns off Autoterm.

EC&s1J — Turns on Autoterm.

Bell off/on

EC&q8te0{0D — Turns Bell off.

EC&q8te0{1D — Turns Bell on.

Block mode off/on

EC&q<m>te1{0B — Turns block mode off (<*m*> = 4-7).

EC&q<m>te1{1B — Turns block mode on (<*m*> = 4-7).

Block terminator character

EC&q<m>te2{<x>R — Sets block terminator character where <*m*> = 4-7 and <*x*> = 0-127.

CapsLock off/on

EC&q<m>te1{0C — Turns CapsLock off (<*m*> = 4-7).

EC&q<m>te1{1C — Turns CapsLock on (<*m*> = 4-7).

Clearterm off/on (active and non-volatile values)

EC&q<m>te0{0K — Turns Clearterm off (<*m*> = 4-7).

EC&q<m>te0{1K — Turns Clearterm on.

Clearterm off/on (active value only)

EC&s0K — Turns Clearterm o

EC&s1K — T

EC&k1Z — Turns modified data tags on.

SPOW NO/YES

EC&q<m>te0{0B — Sets **SPOW** to NO (<*m*> = 4-7).

EC&q<m>te0{1B — Sets **SPOW** to YES (<*m*> = 4-7).

Start transmit-only field

EC{ — Starts a transmit-only field.

TAB=SPACES NO/YES

EC&q8te1{0T — Sets TAB=SPACES to NO.

EC&q8te1{1T — Sets TAB=SPACES to YES.

Transmit ALL/MODIFIED

EC&q<m>te2{0Z — Sets **TRANSMIT** to ALL (<*m*>= 4-7).

EC&q<m>te2{1Z — Sets **TRANSMIT** to MODIFIED (<*m*>= 4-7).

Transmit functions NO/YES

EC&q<m>te0{0A — Sets **Transmit Functions** to NO (<*m*>= 4-7).

EC&q<m>te0{1A — Sets **Transmit Functions** to YES (<*m*> = 4-7).

Write nondisplaying terminator

*EC_* — Writes a nondisplaying terminator at the cursor position.

47.4 HP 700 Series Sequences

The following sequences require that the **Terminal ID response** field on the Terminal Configuration, Page 2 menu be set to *70092* or *70094*.

Alphanumeric cursor

*EC*dQ* — Turns on the alphanumeric cursor.

*EC*dR* — Turns off the alphanumeric cursor.

Configure cursor

^{E}C*d0Q — Displays the cursor as an underline.

^{E}C*d1Q — Displays the cursor as a block.

Report downloadable character set capabilities

^{E}C*y^ — Reflection 1 and 7 do not currently support downloadable character sets.

Normal/Inverse video

^{E}C*d0E — Displays a normal screen.

^{E}C*d1E — Displays an inverse video screen.

Store/restore configuration, user keys

^{E}C&f0B — Store configuration, function keys, user keys

^{E}C&f1B — Restore configuration, function keys, user keys

Also see page 471 for 132/80 column mode sequences for these terminals.

Appendix C

HP Graphics Escape Sequences

The escape sequences listed below only relate to Reflection 7. Sequences that refer to choosing pen colors relate to 2627A emulation only.

47.4

Graphics Defaults

Graphics hard reset

*Ec*wR* — Sets all defaults, clears raster memory buffer, and sets the pen position to Ø,Ø. See page 411.

Restoring graphics defaults

*Ec*mR* — Restores all graphics defaults.

Restoring selected graphics defaults

*Ec*m<defaultflag>R* — If a 1 is used as the *<defaultflag>*, a subset of default values are set. Otherwise all defaults are set. See page 410.

47.4

Reading Graphics Status

Read area shading capability

*Ec*s11^*— Area shading capabilities are fixed for the HP 2627A. Shading can be a polygon 8 pixels wide and 8 pixels high. The terminal always responds as follows: *2,8,8, <terminator>*.

Read current pen position

*Ec*s2^*— Both the pen position and whether it is up (Ø) or down (1) are returned as a string of ASCII characters. Value returned: *<X> <Y> <pen state> <terminator>*.

Read device capabilities

^{E}C*s6^— Returns a list of graphics and plotting features. If the capability is not available, a zero is returned. The sequences marked with an asterisk are only available if you are emulating the HP 2627A terminal.

Device Capability		Response
Clear display	b1	Ø = no clear 1 = paper advance 2 = clear (total erase) 3 = partial clear by area*
Number of pens	b2	(8*)
Color capability	b3	Ø = black or white 1 = gray levels 2 = color*
Color levels capability	b4	(2*) 2 equals the number of color levels for each plane.
Area shading	b5	Ø = no 1 = yes*
	b6 and b7 — Not used.	
Dynamic modification	b8	Ø = no 1 = yes*
Graphics character size	b9	Ø = fixed 1 = integer multiples of the basic cell size* 2 = any size

Graphics character angles	b10	Ø = fixed 1 = multiples of 90°* 2 = multiples of 45° 3 = any angle
Graphics character slant	b11	Ø = fixed 1 = 45°* 2 = any angle
Dot-Dash line patterns	b12	Ø = none 1 = predefined only 2 = user-defined and predefined*

Reflection configured as a 2627A terminal responds as follows:

`3,8,2,2,1,0,0,1,1,1,1,2,0,0,0,0,<terminator>`

Reflection configured as a 2393A terminal responds as follows:

`3,8,2,2,1,0,0,1,1,1,1,2,0,0,0,0,<terminator>`

Reflection configured as a 2623A terminal responds as follows:

`3,1,0,0,1,0,0,1,1,1,1,2,0,0,0,0,<terminator>`

Read device ID

*^{E}C*s1^*— Requests the device ID of the terminal. The response depends on the configured value of the **Terminal ID response** field. Value returned: *2627A <terminator>*

Read display size

*^{E}C*s5^*— Returns the number of pixels which can be displayed per millimeter available on the graphics display. If you are addressing graphics devices with varying display areas, this sequence allows you to scale data.

<LLX>,<URX>	The lower left and upper right X coordinates.
<LLY>,<URY>	The lower left and upper right Y coordinates.
<MMX>,<MMY>	The number of pixels per millimeter in both axes. (Five digits and a decimal point)

Value returned:
`+00000,+00000,+00511,+00389, 00002.,00002.,<terminator>`

Read graphics cursor position

^{E}C*s3^— The position of the graphics cursor is returned. Value returned: *<X> <Y> <terminator>*.

Read graphics cursor position with wait

^{E}C*s4^— The cursor can be positioned by the user before the position coordinates are returned. The arrow keys or mouse can be used to position the cursor, which is turned on if not previously on. When a character key is struck or one of the mouse buttons is pressed, the position of the cursor is given. If another escape sequence is received before a key is struck, the read sequence is ignored. Along with coordinates, the ASCII value of the key struck is given. Value returned: *<X> <Y> <key code> <terminator>*.

The keycode equals 120 for the left mouse button and 013 for the right mouse button.

Read graphics modifications

^{E}C*s12^— The terminal's ability to change selected portions of the display are read. The terminal's fixed response is selective erase and complement capabilities, expressed as follows: *1,1 <terminator>* .

Tablet/Mouse status

^{E}C*s32^— Requests input device ID.

^{E}C*s33^— Requests cursor position if device originates a data transfer.

Read graphics text status

^{E}C*s7^— The current text size, orientation, slant and type of justification are returned. Values are returned in the following order.

<X size>	The <x> dimension of the character cell. (Sign plus 5 digits)
<Y size>	The <y> dimension of the character cell. (Sign plus 5 digits)
<origin>	Relative position of the text to the graphics cursor. (1 digit)
<angle>	Text angle 0,90,180, or 270. (Five digits and a decimal point)
<slant>	00000. or 00045. degrees.

Read relocatable origin

*EC*s9^* — Reads the position of the relocatable origin. Value returned: *<X> <Y> <terminator>.*

Read reset status

*EC*s1Ø^* — Checks to see if a reset was done since the last reset status was done. 1 indicates a full reset has been done; Ø indicates no reset has been done since last check. (Bytes 1 to 7 are returned but not used.)

Read zoom status

*EC*s8^* — Reads the zoom status. When emulating an HP 2627A, the zoom status is always a constant value. Value returned: *001.,0CR.*

Note: If any other parameter is sent to the terminal which does not have an assigned value, the terminal's ID is returned.

47.4

Graphics Keypad Control

Graphics keypad off/on

EC&kØO — Turns the graphics keypad off.

EC&k1O — Turns the graphics keypad on.

47.4

Cursor Control

Also see *Controlling the Display*, page 413.

Alphanumeric cursor

*EC*dQ* — Turns on the alphanumeric cursor.

*EC*dR* — Turns off the alphanumeric cursor.

Graphics cursor

*EC*dK* — Turns on the graphics cursor.

*EC*dL* — Turns off the graphics cursor (the default).

Mouse online/offline

*EC*j1A* — (Default) Mouse set online with default values.

*EC*jØA* — Mouse set off line.

Move graphics cursor, absolute

$^{E}_{C}$*d<X,Y>O — Positions the graphics cursor to absolute screen coordinates.

Move graphics cursor, relative

$^{E}_{C}$*d<X,Y>P — Positions the graphics cursor relative to its current position.

47.4 Display Control

Alphanumeric display

$^{E}_{C}$*dE — Turns on the alpha display.

$^{E}_{C}$*dF — Turns off the alpha display.

Clear/Set graphics memory

$^{E}_{C}$*d<pen#>A — Turns all graphics memory pixels off, or to the selected pen number, if given. If no <pen#> is provided, the default pen color is Ø (black).

$^{E}_{C}$*d<pen#>B — Turns all graphics memory pixels on, or to the selected pen number. If no <pen#> is provided, the default pen color is 7 (white).

Graphics display on/off

$^{E}_{C}$*dC — Turns on the graphics display.

$^{E}_{C}$*dD — Turns off the graphics display (the default).

Print graphics memory

$^{E}_{C}$&p7sF — Prints graphics memory.

Rubberband line

$^{E}_{C}$*dM — Turns on rubberband mode.

$^{E}_{C}$*dN — Turns off rubberband mode.

47.4 Graphics Text Escape Sequences

See *Graphics Text Escape Sequences* on page 415 for details on these sequences.

Graphics labels

^{E}C**l<text string>*^{C}R (or ^{L}F) — Creates a label in graphics memory using current text settings. Must end with a ^{C}R, ^{L}F, or both. The *<text string>* may not exceed 73 characters.

Graphics text colors

^{E}C**n<pen>X* — Selects the color for graphics text. The default is the primary pen color.

Graphics text mode

^{E}C**dS* — Turns on graphics text mode.

^{E}C**dT* — Turns off graphics text mode.

^{E}C**m<n>M* —Character size (1-8)

^{E}C**mO* —Slanted characters

^{E}C**mP* —Upright characters

^{E}C**m<origin>Q* —Justification:Origin (∅-9)

^{E}C**m<n>N* —Orientation

where	1 = default (rotated 0 degrees)
	2 = rotated 90 degrees counter-clockwise
	3 = rotated 180 degrees, upside down
	4 = rotated 270 degrees counter-clockwise

47.5 Selecting Drawing Modes

See *Graphics Drawing Modes* on page 419 for more information on using these sequences. HP 2627A drawing modes are on page 442.

Select drawing mode

^{E}C**m<n>A* — The value *<n>* defines which mode is selected.

0	No effect
1	Clear (turn off graphics bits)
2	Set (turn on graphics bits)
3	Complement (toggle graphics bits)
4	Jam (turn bits on or off according to data)

47.5

Plotting Lines

Plotting Lines on page 421 provides details on using these sequences.

Lift/Lower pen
*^{E}C*pA* *^{E}C*pB*

Graphics cursor as new point
*^{E}C*pC*

Draw point at pen position and lift pen
*^{E}C*pD*

Set relocatable origin at pen position
*^{E}C*pE*

Data is ASCII absolute
*^{E}C*pF*

Data is ASCII incremental
*^{E}C*pG*

Data is ASCII relocatable
*^{E}C*pH*

Data is binary absolute
*^{E}C*pI*

Data is binary short incremental
*^{E}C*pJ*

Data is binary incremental
*^{E}C*pK*

Data is binary relocatable
*^{E}C*pL*

No operation is performed
*^{E}C*pZ*

47.5

Defining Line Patterns

See page 430 for more information on using these sequences.

Define a line pattern

*^{E}C*m<x><y>C* — Defines an 8-bit segment of line pattern and a scale.

<x>	is a number from 0 to 255 which, when converted to its binary form, illustrates the segment of line pattern.
<y>	is a number from 1 to 16 which indicates the number of times each dot of the line pattern should be repeated.

47.5

Area Patterns

See *Filling Rectangle and Polygon Areas* starting on page 430 for further information.

Define an area pattern

*^{E}C*m<row 0> <row 1>...<row 7>D* — Where each decimal number represents one row of the pattern.

Rectangle fill, absolute

*^{E}C*m<x1> <y1> <x2> <y2>E* — Values *<x1>* and *<y1>* are the lower left corner coordinates; *<x2>* and *<y2>* are the upper right corner coordinates.

Rectangle fill, relocatable

*^{E}C*m<x1> <y1> <x2> <y2>F* — Values *<x1>* and *<y1>* are the relocatable coordinates of the lower left corner; *<x2>* and *<y2>* are the relocatable coordinates of the upper right corner. These values are added to the relocatable origin coordinates to compute the actual coordinates.

Select area pattern

*^{E}C*m<area pattern>G* — where the area pattern is one of the following:

0	Current dither pattern (color graphics only)
1	Solid area fill pattern
2	User-defined area fill pattern
3-10	Predefined area pattern

Start/End polygon area fill

EC*pS — Starts filling polygon area.

EC*pT — Stops polygon area fill.

EC*pA — (Lift/lower pen.) When the pen is lifted in the middle of a polygon fill, the previous polygon is closed and a new polygon is begun.

47.5 Relocatable Origins

Relocatable origin absolute

EC*m<X,Y>J — X and Y are ASCII numbers between -16348 and 16383.

Relocatable origin at pen position

EC*mK — X and Y are the current pen position.

Relocatable origin at graphics cursor

EC*mL — The graphics cursor position is the new origin.

47.6 Setting Compatibility Mode

See *Setting Compatibility Mode* on page 444 for more information on using these sequences.

Compatibility mode scaled/unscaled

EC&s1p0Q — Sets scaled mode on.

EC&s0p1Q — Sets unscaled mode on.

EC&s0p0Q — Turns off compatibility mode (the default).

Straps

Disable all terminators

EC*t2A — Sets nothing as the terminator.

Page full busy, normal

EC*t0C — Normal indicates there is no keyboard locking.

Page full busy, lock

EC*t1C — The keyboard locks on a full page.

Page full break, no break

*Ec*t0B* — No break occurs.

Page full break, break

*Ec *t1B* — Sends a break signal to host after the 35th line.

Set carriage return terminator

*Ec *t0A* — Sets a carriage return as the terminator (the default).

Set carriage return and EOT terminator

*Ec *t1A* — Sets both the carriage return and EOT as terminators.

47.6 Color Graphics Escape Sequences

Color features are only available with EGAs and enhanced displays or VGAs with VGA displays. See page 435 for more information on color graphics usage.

Pen colors

Numbers associated with pen colors:

0	Black	**4**	Blue
1	Red	**5**	Magenta
2	Green	**6**	Cyan
3	Yellow	**7**	White

Boundary pen

*Ec*m<pen#>H* — Selects the boundary pen color, where *<pen#>* is replaced with one of the eight possible pen colors listed above. See page 442 for more information.

*Ec*pU* — Lifts boundary pen.

*Ec*pV* — Lowers boundary pen.

Dither pattern

*Ec*m<d1,d2,d3>V* — Defines the dither pattern, where *d1-d3* define the percent of each plane to include.

*Ec*m<pattern>W* — Selects the dither pattern. See page 440 for examples.

1 User defined dither pattern

2 Predefined dither pattern (violet)

3 Predefined dither pattern (brown)

4 Predefined dither pattern (burnt orange)

5 Predefined dither pattern (gold)

6 Predefined dither pattern (lime green)

7 Predefined dither pattern (turquoise)

8 Predefined dither pattern (red)

9 Predefined dither pattern (green)

10 Predefined dither pattern (blue)

11 Predefined dither pattern (white)

12 Predefined dither pattern (black)

Selecting Colors

See page 438 for more information.

Graphics text colors

*EC*n<pen#>X* — Selects the color for graphics text.

Select primary pen color

*EC*m<pen#>X* — Selects primary pen color.

Select secondary pen color

*EC*m<pen#>Y* — Selects secondary pen color.

Select background pen color

*EC*e<pen#>B* — Sets the background pen color, and set the graphics memory to that color. Black is the default.

Color Pairs

Selecting color pairs

EC&v<parameter> — where *<parameter>* can be replaced by one of the list below. The defaults (except for status) are zero. All decimal values must be in the range of 0 to 1, but are truncated to two decimal places. Other values are ignored.

Øm	Red, Green, Blue color method.
1m	Hue, Saturation, Luminosity color method.
<decimal>a	Red (or hue) color value for foreground.

<decimal>b	Green (or saturation) color value for foreground.
<decimal>c	Blue (or luminosity) color value for foreground.
<decimal>x	Red (or hue) color value for background.
<decimal>y	Green (or saturation) color value for background.
<decimal>z	Blue (or luminosity) color value for background.
<Ø-7>i	Color pair number to be initialized.
<Ø-7>s	Color pair number to be selected.
<Ø-7>^	Color pair definition status. Because this character is considered a *capital* this must be the last parameter of the escape sequence where more than one parameter is used.

Selecting Drawing Modes

See *Drawing Modes* on page 442 for a discussion of these sequences.

Select drawing mode

*Ec*m<mode>A* — Selects one of seven drawing modes. The default is JAM1.

47.6

User Keys in Color Graphics

Reflection 7 allows you to define color pairs and video enhancements for user keys. Each user-key label can have its own pair of colors as well as video enhancement combinations (blinking, underline, and inverse video). The rest of the parameters are identical to those discussed in the *User-Key Escape Sequence Parameters* table on page 462.

Note: If you are *only* specifying a color pair and/or video enhancements, you must specify a Ø length parameter with ØL, as it normally defaults to one.

```
Ec&f<color pair>c<video enh>vØL
```

User keys are displayed as color pair 7 under the default. If half-bright is selected as an enhancement, the color pair 3 is used. If a color is also selected, the half-bright enhancement is ignored.

Because the <*key*> parameter resets the color pairs and video enhancements to their defaults, you must place these values after <*key*>. The following sequence includes all possible parameters and a suggested order. Any combination of parts may be selected, but the final parameter must be capitalized to complete the sequence. The three parts shown here should appear on a single line.

```
Ec&f<attribute>a<key>k<color pair>c
      <label length>d<string length>l
      <video enh>V<label><string>
```

Table 50
Color Parameters for User Keys

Variable	Parameters	Default
<color pair>	Ø-7c	7
<video enh>	Ø-15v	Ø

#	Value	#	Half-Bright and ...
0		9	Blinking
1	Blinking	10	Inverse Video
2	Inverse Video	11	Blinking and Inverse Video
3	Blinking and Inverse Video	12	Underline
4	Underline	13	Blinking and Underline
5	Blinking and Underline	14	Inverse Video
6	Inverse Video and Underline	15	Blinking, Inverse, Underline
7	Blinking, Inverse, and Underline		
8	Half-Bright		

47.6 Terminal Capabilities

*Ec*s-<n>^*

This sequence reports terminal capabilities. Depending on the current terminal ID the responses to this escape sequence vary. The following sequences are only available when the terminal type is set to an HP 2397A or 700 series terminal. The terminal capability response varies depending on the ID:

*^{E}C*s-1^*

Alphanumeric capabilities:

2397A response: *^{S}P#^{C}R <terminal>*
2393A response: *^{S}P"^{C}R <terminal>*
700/92 response: *C ^{S}P ! # ^{C}R*
700/94 response: *C , ! # ^{C}R*

*^{E}C*s-2^*

Graphics capabilities:

2397A response: *A!^{C}R <terminal>*
2393A response:
700 Response: *A ^{S}P ^{C}R*
2393A response:

*^{E}C*s-3^*

RAM available for downloading:

2393A response:
700 Response: *B ^{S}P ^{S}P ^{C}R*

*^{E}C*s-4^*

Interface capabilities (depends on printer configuration):

2397A response: *B$^{S}P$$^{S}P$$^{C}R$* (No interfaces)
2393A response:
700/92 Response: *B " ^{S}P ^{C}R*
700/94 Response: *B * ^{S}P ^{C}R*

*^{E}C*s-5^*

HP-HIL interface

2397A response: *B!!^{C}R* (With mouse)
2393A response: *B!$^{S}P$$^{C}R$* (Without mouse)

The terminal responds with a single byte which indicates the number of status bytes in the response, followed by the bytes of status information, in which the high-order bits of each byte contain 001.

See page 516 for more information on the format of the response.

Appendix D

HP Block Transfers

47.6 Block Transfer Handshaking

Reflection uses three types of handshaking:

- No handshake: The block of data is sent immediately.
- DC1: Reflection sends the block after receiving the host prompt character.
- DC1/DC2/DC1: Reflection sends an ASCII DC2 character after receiving a host prompt character. After receiving another host prompt, it sends the block.

Following is a list of the types of block transfers, classified according to the handshaking rules that apply:

Type 1

Press Enter in block mode. In block/page mode, press a transmit-only user function key or Select (if mapped).

Type 2

Press Enter in character mode.

Type 3

- Press a transmit-only user function key in character or block/line mode, or Select (if mapped).

- Reflection responds to a primary status request, secondary status request, terminal ID request, serial number request, device status request, or cursor sense escape sequence.
- Reflection sends a block of data from display memory or from the user-key definition menu, in response to an ^{E}Cd sequence from the host.
- Reflection responds to a device control operation with a device control completion code (**S**, **F**, or **U**).

Type 4

Press [Enter] or [Return] in character mode with **LINE MODIFY** or **MODIFY ALL** enabled.

The table below summarizes the four transfer types.

Table 51
Block Transfer Handshaking Protocol

TRANSFER TYPE	INHIBIT HNDSHK	INHIBIT DC2	HANDSHAKE
1	No	No	$^{D}C_1/^{D}C_2/^{D}C_1$
1	No	Yes	None
1	Yes	No	$^{D}C_1/^{D}C_2/^{D}C_1$
1	Yes	Yes	None
2	No	No	None
2	No	Yes	None
2	Yes	No	$^{D}C_1/^{D}C_2/^{D}C_1$
2	Yes	Yes	None
3	No	No	$^{D}C_1$
3	No	Yes	$^{D}C_1$
3	Yes	No	$^{D}C_1/^{D}C_2/^{D}C_1$*
3	Yes	Yes	None
4	N/A	N/A	None*

* With an HP 2622A terminal, the Inhibit $^{D}C_2$ parameter is set to YES to avoid sending a $^{D}C_2$ character when you press [Enter] or [Return]. In contrast, Reflection assumes you always want to send the data block, not a $^{D}C_2$.

47.6 Block Transfer Terminators

Reflection appends one or more characters to the end of each line or field that is sent as part of a block transfer. It also appends one or more characters to the end of the entire block.

In character mode or block/line mode, Reflection appends a carriage return (or carriage return/linefeed if **AUTO LF** is enabled) to the end of the data. If data is not being sent, or the transmission is terminated by encountering a block terminator in display memory, Reflection sends a block terminator followed by a carriage return.

In block/page (non-format) mode, Reflection appends a carriage return (or carriage return/linefeed if **AUTO LF** is enabled) to the end of each line except the last one. If a block terminator character in display memory ends the transfer, then only the block terminator is sent after the last line. If the operation ends because of encountering the end of display memory, then Reflection appends a carriage return, followed by a block terminator to the last line.

In block/page/format mode, Reflection appends a field separator character to the end of each transmitted field except the last one. Reflection appends a block terminator to the last field. If there are no fields to be sent, Reflection just sends a block terminator.

Appendix E

HP Format Mode

Reflection has a *format mode* that allows the use of protected and unprotected fields. By using combinations of unprotected fields, color configurations, and the line drawing character set, you can create forms for *fill in the blanks* data entry using an HP 3000 forms program such as *HP VPLUS/3000.*

47.6 Format Mode Fields

In format mode, the screen is divided into protected and unprotected fields. Data can be entered only in the unprotected fields. If the cursor is in a protected area of the screen and a character is typed, the cursor advances to the next unprotected field before the character is displayed on the screen.

Enable and disable format mode through the modes keys ([Alt]-[M]) by pressing [F5], **FORMAT MODE**. Format mode can also be enabled and disabled by the following escape sequences, either from the keyboard or from the host computer:

$^{E}{}_{C}W$ enables format mode.
$^{E}{}_{C}X$ disables format mode.

When format mode is enabled, a home up function is performed; the cursor moves to the first column of the first unprotected field in display memory. All tab stops are cleared, and the margins are set to columns 1 and 80. Tab stops are ignored in format mode. If there are no unprotected fields in display memory, the cursor moves to row 1, column 1.

Note: When format mode is enabled, MSAVE starts from the top of display memory; see page 275.

Format mode affects keyboard functions as listed in the table in the next section.

47.6 Designing Forms

Forms can be drawn on the screen manually through the keyboard, or by escape sequences and text received from the host computer. With the HP VPLUS/3000 forms handling system, forms are initially designed from the keyboard; then, when the application is run, the form descriptions are sent to the terminal to draw the forms.

Table 52
Keyboard Functions in HP Format Mode

Function	Keystroke	Effect in Format Mode
Insert line	Alt-I	Disabled.
Delete line	Alt-D	Disabled.
Insert char	Ins	Works within unprotected field boundaries.
Delete char	Del	Works within unprotected field boundaries.
Clear line	Alt-K	Clears to end of unprotected field.
Clear display	Alt-J	Clears only unprotected fields from cursor to end of screen.
Tab	Tab	Advances cursor to next unprotected field.
Backtab	Shift-Tab	Moves cursor back to start of current field or to previous unprotected field.
Home up	Ctrl-Home	Moves cursor to first column of first unprotected field in display memory.

The steps involved in drawing a form, either from the keyboard or from the host computer, are as follows:

1. Make sure that format mode is not on. The ^{E}*CX* sequence turns it off.
2. Home the cursor (^{E}*CH*) and clear display memory (^{E}*CJ*).
3. For each unprotected field, move the cursor to the row and column where you want the field to start. If you want the field to be enhanced (inverse video, underlined, etc.), enter the ^{E}*C&d<x>* escape sequence to start the display enhancement.

4. Enter the $^{E}c[$ escape sequence to start the unprotected field.
5. Move the cursor one column beyond the last column that is to be part of the unprotected field. Enter the escape sequence $^{E}c]$ to terminate the field. If you want a field to extend through column 80, do not enter an *end of field* escape sequence. If two fields are to be contiguous, entering $^{E}c[$ ends the previous field and starts a new one.
6. To clear a *start of field* indicator, position the cursor at the first column of the field and press [Del].
7. To define a field that wraps around from the end of one screen row to the next, do not enter an *end of field* escape sequence. Move the cursor to the first column of the following row, and then enter an $^{E}c[$ sequence. This method may be used to define a field to span any number of screen rows.
8. To end a display enhancement, move the cursor one column beyond the last column that is to be enhanced, and enter the sequence $^{E}c\&d@$.
9. When the form has been drawn, you can enable format mode with the escape sequence ^{E}cW , or by pressing [F5], the **FORMAT MODE** key, from the modes labels.

Forms generation can be simplified by loading escape sequences into the user keys. Assign the attribute **N** (normal keyboard input) to each key's definition. See pages 161 and 462 for detailed information about user keys.

47.6 Forms Cache

Forms cache is a feature of Reflection that can be used to reduce the amount of data transmission from the host computer to Reflection when performing applications that use forms. In a typical format mode application, most of the data that is sent to the terminal merely defines the forms to be displayed on the screen. If a given form is to be displayed more than once, a lot of time is wasted by redefining the same form to the terminal each time it is needed.

With the forms cache feature, the programmer can send a number of forms definitions to Reflection for local storage and cause a given form to be displayed by sending a single short escape sequence.

In order to use the forms cache feature with VPLUS/3000 or other HP 3000 software, you must change the **Terminal ID response** field in the Terminal Configuration, Page 2 menu to one of the following:

2624B
2394A
70094

47.6

Allocating the Forms Buffer

Before any forms can be stored in the PC, a memory buffer must be allocated. This can be done manually through the Terminal Configuration, Page 2 field, **Forms buffer size**, or by the escape sequence:

^{E}*C&q<m>te2{<x>L*

where *<m>* = 4-7 and *<x>* is the number of 256-byte blocks to be allocated.

If the requested number of blocks cannot be allocated, then the size of the forms buffer is not changed. The maximum number of blocks that can be requested is 255. A value of zero deletes the forms buffer.

Any attempt to change the forms buffer size causes memory to be re-allocated, so that the contents of display memory and the printer buffer are lost. Note that the forms buffer size directly relates to the amount of free memory available for running other applications with Reflection; see page 327.

In order for a host program to determine if the requested allocation was successful, the programmer must issue a status request, as described below.

47.6

Reading the Forms Buffer Status

A program on the host computer can request Reflection to send the status of the forms buffer, which may include the following:

- Number of 256-byte blocks of memory in the forms buffer
- Number of unused 256-byte blocks remaining in the buffer

The status request escape sequence takes the form:

^{E}*C&p9^* or ^{E}*C&p<form#>p9^*

If the *<form#>* parameter is missing, or if *<form#>* is zero, then the total size of the forms buffer is returned. Otherwise, the number of unused blocks and the presence or absence of the specified form number is returned.

The form of the response is as follows:

Ec\p9<xyz>

where *<xyz>* are three ASCII characters for which the high-order four bits are 0011 and the low-order four bits are as defined below.

- The low-order four bits of the first byte, *<x>*, contain the most significant four bits of the number of blocks (either in total or unused) in the forms buffer.
- The low-order four bits of the second byte, *<y>*, contain the least significant four bits of the number of 256-byte blocks.
- If a non-zero form number is specified in the status request, the low-order four bits of the third byte, *<z>*, indicate the presence (0001) or absence (0000) of the form in the buffer.

In the examples in the following table, 50 blocks with 256 characters each have been allocated for the forms buffer, and form numbers 1 through 15 have been stored, leaving 30 blocks unused.

Table 53
Sample Requests and Responses

Status Request	Status Response
*Ec&p9^*	*Ec \p9320*
*Ec &p0p9^*	*Ec \p9320*
*Ec &p7p9^*	*Ec\p91>1*
*Ec &p19p9^*	*Ec \p91>0*

47.6 Storing a Form

There are two escape sequences that may be used to store a form in the forms buffer. In either sequence, the form number may be any number from 1 through 255. If the specified form number already exists in the buffer, it is overwritten with the new form's contents. The first escape sequence follows:

Ec&p9u<form#>p<length>L<contents>

It causes the next *<length>* characters following the *L* to be stored. Note that the < and > characters delimit variables, and are not part of the escape sequence.

^{E}C&p9u<form#>p<<contents>>L

In this escape sequence, the additional < and > characters before and after the *<contents>* are part of the sequence, and are used to delimit the *<contents>*. Contrary to what is documented in the HP 2624B manual, the *<contents>* can contain any ASCII characters except NUL, DEL, ENQ, and DC1. This escape sequence is useful if you don't know the length of the *<contents>* in advance.

47.6

Displaying a Form

To display a form that has previously been stored in the forms buffer, send the following escape sequence to Reflection:

^{E}C&p9u<form#>pF

A completion code of **S** (success) or **F** (failure) is returned to the host by Reflection after the appropriate handshaking.

47.6

Purging a Form

The following escape sequences purge a form from the forms buffer:

^{E}C&p9u<form#>pL or *^{E}C&p9u<form#>p0L*

If the form number is found, the amount of space that it used in the buffer is released.

47.6

Special Format Mode Features

Reflection must be configured as a terminal that supports these features:

2624B	2393A (Reflection 7 only)
2626A	150A (Reflection 7 only)
70094	2390A (Reflection 7 only)
2394A	2397A (Reflection 7 only)

See page 117 for a summary of Reflection terminal emulation capabilities; page 473 provides a list of restricted escape sequences.

Transmit-only Fields

Transmit-only fields are similar to unprotected fields, although data is not usually entered into transmit-only fields. ([Tab] and [Ctrl]-[Home] skip over transmit-only fields.)

^{E}C*{* is used to begin a transmit-only field.

Nondisplaying Terminators

Reflection supports nondisplaying terminators and the Autoterm and Clearterm configuration options when configured as a terminal that supports this feature. ^{E}C_ and ^{E}C|, respectively, insert and delete a nondisplaying terminator at the current cursor position. Autoterm and Clearterm can be enabled by escape sequence or by using Reflection SET commands.

Modified Data Tags

With the appropriate configuration, Reflection supports the modified data tag feature. When an HP 2624B (HP 2393A, HP 2394A, or HP 2397A) terminal is configured to Transmit Modified rather than All fields, trailing blanks are eliminated from the contents of unprotected fields and, if in Block Page mode, only modified fields are sent. Reflection supports the escape sequences to change the Transmit option and also provides a command: SET TRANSMIT ALL/MODIFIED.

HP Appendices

Appendix F

HP Cursor Positioning and Sensing

The cursor can be placed anywhere in Reflection's display memory by means of an escape sequence issued from the host computer or from the keyboard. There are three ways to specify cursor position within display memory:

- Screen-relative
- Cursor-relative
- Absolute

In the escape sequences, row and column numbers are always expressed relative to zero. In other words, the first row and column position is row 0, column 0. The row and column numbers within the escape sequences are expressed as ASCII decimal digits. If a number is preceded or followed by spaces, the spaces are ignored.

47.6 Screen-Relative Cursor Positioning

In this form of cursor positioning, the new position is specified relative to the upper left corner of the current screen. The escape sequences take one of the following forms:

- ^{E}*C&a<row>y<column>X*
- ^{E}*C&a<column>X<row>y*

- *^{E}C&a<row>Y*
- *^{E}C&a<column>X*

If only a row is specified, the cursor is moved to the current column in the specified row. If only a column is specified, the cursor is moved to the specified column in the current row. If the value of either row or column is so large that it implies moving the cursor off the screen, the maximum on-screen value is used instead of the specified value. In other words, if you try to place the cursor in column 85 using this form, Reflection places it in column 79 (relative to zero).

For example, the following sequence places the cursor in the 12th screen row and the 10th column:

^{E}C&a11y009X

Note that there must be no space between the ^{E}C and the &.

47.6 Absolute Cursor Positioning

In this form of cursor positioning, the new position is expressed relative to the first row (row 0) in display memory. If the new position is outside the bounds of the current screen window, the display is rolled in the direction required to bring the target position onto the screen.

The escape sequence takes one of the following forms:

- *^{E}C&a<row>r<column>C*
- *^{E}C&a<column>c<row>R*
- *^{E}C&a<row>R*

Note that only the *<row>* portion of the escape sequence is *absolute,* since column numbers are always relative to column 0.

The maximum value of *<column>* is either 79 or the right margin column, whichever is greater. If the specified value of *<column>* is greater than the limit, the limit is used instead of the specified value. This prevents the cursor from being placed in a column that is not accessible by horizontal scrolling.

The maximum allowable *<row>* value is 32767. If the *<row>* value is larger than Reflection's display memory can hold, then rows are deleted off the top of display memory and added to the bottom until the cursor has been moved the number of rows that are implied by the escape sequence.

For example, the following escape sequence moves the cursor to the 103rd column of the 1451st row of display memory:

^{E}c&a102c1450R

47.6

Cursor-Relative Positioning

In this form of cursor positioning, the new position is specified relative to the current cursor position. If necessary, the display is rolled in whatever direction is required to bring the new position within the screen window.

Cursor relative positioning is specified by placing a + or - sign in front of the row or column number. For example, the following escape sequence moves the cursor up 2 rows and right 7 columns from its current position:

^{E}c&a-2r+07C

The following escape sequence moves the cursor left 20 columns within the current row.

^{E}c&a-20C

47.6

Cursor Sensing

The host computer can request and receive the current cursor position. The position can be expressed relative to the current screen or relative to the first row in display memory.

The cursor sensing response is a block transfer, so the rules of block transfers apply (see page 495, *HP Block Transfers*).

47.6

Screen-Relative Sensing

The host computer sends the escape sequence ^{E}c' . Reflection responds with a cursor positioning escape sequence of the form *^{E}c&a<column>c<row>Y*, where *<column>* and *<row>* are both three-digit ASCII decimal numbers. The maximum value of *<column>* is 79; the maximum value of *<row>* is 23.

47.6

Absolute Sensing

The host computer sends the escape sequence ^{E}C*a* . Reflection responds with a cursor positioning escape sequence of the form, ^{E}C*&a<column>c<row>R* , where *<column>* and *<row>* are both three-digit ASCII decimal numbers. The maximum value of either number is 999. Note that the actual row and column numbers may be larger than 999; in that case, Reflection reports only the low-order three digits.

47.6

Five-Digit Cursor Positioning

Reflection provides an escape sequence that returns a 5-digit cursor position, reflecting the absolute value of the current position.

^{E}C*&oA*

Appendix G

HP Status Requests

A program running on the host computer can request status and terminal information from Reflection. There are five types of status requests:

- Primary status
- Secondary status
- Terminal ID
- Device status
- Serial number request

Reflection responds to each status request with a block transfer. The rules for block transfer handshaking and the terminator characters that Reflection appends to the end of any block transfer are explained in *HP Block Transfers* on page 495.

47.6

Primary Status Request

To request the primary status from the terminal, the host computer sends the escape sequence ^{E}C^. The first byte indicates the amount of display memory available, 15K for HP 700 terminals.* Reflection responds by performing a block transfer handshake and then sends the response followed by a block transfer terminator. The response consists of nine characters, as follows:

- The first two characters are ^{E}C\.
- The remaining characters are all 8-bit binary quantities, where the high-order 4 bits of each byte are 0011. This ensures that all responses can be

* Reflection previously reported the HP 2392A value of 4K.

read as ASCII characters. The low-order four bits of each byte are described below, where the bits are numbered as follows:

```
7 6 5 4  3 2 1 0 - bit numbers
| | | |  | | | |
0 0 1 1  ? ? ? ? - bit values
```

Byte 0: (indicates the amount of display memory)

Bit 3	1 = 8K bytes
Bit 2	1 = 4K bytes
Bit 1	1 = 2K bytes
Bit 0	1 = 1K bytes

Reflection responds with the value for 4K bytes, 8K, 12K, or 15K, depending on the value configured for **Display memory response**.

Byte 1: (configuration straps A-D)

Bit 3	Strap D, page/line mode	0 = line, 1 = page
Bit 2	Strap C, inhibit end-of-line wrap	0 = no, 1 = yes
Bit 1	Strap B, space overwrite	0 = no, 1 = yes
Bit 0	Strap A, transmit functions	0 = no, 1 = yes

Byte 2: (configuration straps E-H)

Bit 3	Strap H, inhibit DC2	0 = no, 1 = yes
Bit 2	Strap G, inhibit DC1 handshake	0 = no, 1 = yes
Bit 1	Always 0	
Bit 0	Always 0	

Byte 3: (latching keys)

Bit 3	Always 1, indicating this model of terminal is capable of sending secondary status	
Bit 2	Auto linefeed	0 = off, 1 = on
Bit 1	Block mode	0 = character mode, 1 = block mode
Bit 0	$\boxed{\text{CapsLock}}$	status 0 = off, 1 = on

Byte 4:	Pending transfers	
Bit 3	Secondary status pending	0 = no, 1 = yes
Bit 2	Enter pending	0 = no, 1 = yes
Bit 1	User function key pending	0 = no, 1 = yes
Bit 0	Cursor sense pending	0 = no, 1 = yes
Byte 5:	Errors	
Bit 3	Printer error	0 = no error, 1 = last printer operation failed
Bit 2	Always 0	
Bit 1	Self-test result	0 = Error or self-test not requested 1 = No error
Bit 0	Datacomm error status	0 = no error, 1 = framing, overrun or parity error has occurred since the last status request

Byte 6: Device transfer pending flags. This byte reports on the **S**, **F**, or **U** completion codes associated with the ^{E}C*&p* device control escape sequences.

Bit 3	Always 0	
Bit 2	Always 0	
Bit 1	Device operation status pending	0 = no, 1 = yes
Bit 0	Device status pending	0 = no, 1 = yes

47.6 Secondary Status Request

To request the secondary status response from the terminal, the host computer sends the escape sequence ^{E}C~ . Reflection responds by performing a block transfer handshake and then sends the response followed by a block transfer terminator. The response consists of 9 characters, as follows:

- The first two characters are ^{E}C| (the ASCII vertical slash character, decimal 124).
- The remaining characters are all 8-bit binary quantities, where the high-order 4 bits of each byte are 0011. This ensures that all responses can be read as ASCII characters. The low-order four bits of each byte are described below, where the bits are numbered as follows:

```
7 6 5 4  3 2 1 0 - bit numbers
| | | |  | | | |
0 0 1 1  ? ? ? ? - bit values
```

Byte 0: Reports the amount of available terminal memory, besides display memory, available for data buffers.

Bit 3	1 = 8K bytes
Bit 2	1 = 4K bytes
Bit 1	1 = 2K bytes
Bit 0	1 = 1K bytes

Reflection always returns a value of zero in this byte.

Byte 1: Terminal firmware configuration

Bit 3	Always 0, non-programmable terminal
Bit 2	Always 1, terminal identifies self
Bit 1	Always 0, no APL firmware
Bit 0	Always 1, I/O firmware installed

Byte 2: Configuration straps J-M. These straps do not apply to the HP 2622A terminal. Reflection returns a 0 in this byte.

Byte 3: Keyboard interface keys (N-R). Reflection returns a 0 in this byte.

Byte 4: Configuration straps S-V. These straps do not apply to the HP 2622A terminal. Reflection returns a value of 0 in this byte.

Byte 5: Configuration straps W-Z. These straps do not apply to the HP 2622A terminal. Reflection returns a value of 0 in this byte.

Byte 6: Memory lock mode status

Bit 3	Always 0
Bit 2	0 = memory is not full, 1 = memory is full
Bit 1	0 = memory lock not on, 1 = memory lock on
Bit 0	0 = locked in row 0 (i.e., overflow protect), 1 = not locked in row 0

47.6

Terminal ID

The host computer can request the terminal ID with the escape sequence ^{E}C*s^. After exchanging the appropriate block transfer handshake, Reflection sends the five-character string, for example *2392A*, *2622A*, or *70094* depending on the value configured for **Terminal ID response**.

47.6

Device Status

The host computer can request the status of the attached printer by issuing the escape sequence ^{E}C*&p* <device#>^, where *<device#>* is either 4 or 6. Both 4 and 6 indicate external printers in Reflection, so there is no difference.

Reflection responds by sending the string ^{E}C*\p<device#>* , followed by three bytes of status information, in which the high-order four bits of each byte contain 0011.

Byte 0:

Bit 3	Always 0	
Bit 2	Always 0	
Bit 1	Printer error report	0 = no error, 1 = printer error
Bit 0	Tracks **S** or **F** response of device operation	0 = last command successful (S) 1 = last command failed (F)

Byte 1:

Bit 3	Tracks **U** response of device operation	0 = last command was interrupted 1 = last command was performed
Bit 2	Always 0	
Bit 1	Always 0	
Bit 0	Printer status	0 = not busy, 1 = busy

Byte 2:

Bits 3 - 0	0001 (printer present)

HP Appendices

47.6

Terminal Capabilities

The terminal capabilities can be requested with the following escape sequence:

^{E}C*s-<*n*>^

where <*n*> selects the capability type, 1 to 5; status responses follow.

47.6

Graphics Status Responses

Graphics status responses include terminal alphanumeric characterstics, graphics characteristics, available memory for downloading code, terminal interface capabilities, and HP-HIL capabilities.

Terminal Alphanumeric Capabilities

Byte 0: (Returns 2 as the string length)

Byte 1: Forms characteristics

Bit 1	Simple edit checks	0 = no 1 = yes
Bit 2	Extended edit checks	0 = no 1 = yes
Bit 3	Modified data tags	0 = no 1 = yes
Bit 4	Forms cache	0 = no 1 = yes
Bit 5	Not used	

Byte 2: Display characteristics

Bit 1	Security enhancement supported	0 = no 1 = yes
Bit 2	Color	0 = no 1 = yes
Bit 3	Not used	
Bit 4	Not used	
Bit 5	Not used	

Terminal Graphics Capabilities

Byte 0: Returns 1 as the string length

Byte 1 — Graphics capabilities:

Bit 1	Graphics present	0 = no 1 = yes
Bit 2	Alternate display resolution support	0 = no 1 = yes
Bit 3	Not used	
Bit 4	Not used	
Bit 5	Not used	

Memory Available for Downloading Code

Byte 0: Returns 2 as the string length

Byte 1: Least significant byte

Bit 1	2^0	0 = no 1 = yes
Bit 2	2^1	0 = no 1 = yes
Bit 3	2^2	0 = no 1 = yes
Bit 4	2^3	0 = no 1 = yes
Bit 5	2^4	0 = no 1 = yes

Byte 2: Most significant byte

Bit 1	2^5	0 = no 1 = yes
Bit 2	2^6	0 = no 1 = yes
Bit 3	2^7	0 = no 1 = yes
Bit 4	2^8	0 = no 1 = yes
Bit 5	2^9	0 = no 1 = yes

To determine the memory size in K bytes, convert the binary number represented by bytes 1 and 2 to a decimal number, and then multiply by 8K.

Terminal Interface Capabilities

Byte 0: Returns 2 as the string length

Byte 1: Interfaces

Bit 1	HP-IB interface	0 = no 1 = yes
Bit 2	Serial printer port present	0 = no 1 = yes
Bit 3	Printer pass-through	0 = no 1 = yes
Bit 4	Programmable configuration (^{E}*c&q*)	0 = no 1 = yes
Bit 5	Not used	

Byte 2: 8-bit parallel port:

Bit 1	Not used	
Bit 2	Not used	
Bit 3	8-bit parallel (Centronics) port	0 = no 1 = yes
Bit 4	Not used	
Bit 5	Not used	

HP-HIL Capabilities

Byte 0: Returns 2 as the string length

Byte 1: Alpha devices

Bit 1	Input data code is keycode	0 = no 1 = yes
Bit 2	Input data code is ASCII	0 = no 1 = yes
Bit 3	Touchscreen present	0 = no 1 = yes
Bit 4	Not used	
Bit 5	Unsupported/unrecognized device	0 = no 1 = yes

Byte 2: Positioning device

Bit 1	Relative positioning device
Bit 2	Absolute positioning device
Bit 3	Not used
Bit 4	Not used
Bit 5	Not used

Appendix H

HP Character Sets

47.6

The ASCII Character Set

Decimal Value	→	0	16	32	48	64	80	96	112
↓	Hexa-decimal Value	0	1	2	3	4	5	6	7
0	0	$^{N}_{U}$	$^{D}_{L}$		0	@	P	`	p
1	1	$^{S}_{H}$	$^{D}_{1}$	!	1	A	Q	a	q
2	2	$^{S}_{X}$	$^{D}_{2}$	"	2	B	R	b	r
3	3	$^{E}_{X}$	$^{D}_{3}$	#	3	C	S	c	s
4	4	$^{E}_{T}$	$^{D}_{4}$	$\$$	4	D	T	d	t
5	5	$^{E}_{Q}$	$^{N}_{K}$	%	5	E	U	e	u
6	6	$^{A}_{K}$	$^{S}_{Y}$	&	6	F	V	f	v
7	7	$^{B}_{L}$	$^{E}_{B}$	'	7	G	W	g	w
8	8	$^{B}_{S}$	$^{C}_{N}$	(	8	H	X	h	x
9	9	$^{H}_{T}$	$^{E}_{M}$	)	9	I	Y	i	y
10	A	$^{L}_{F}$	$^{S}_{B}$	*	:	J	Z	j	z
11	B	$^{V}_{T}$	$^{E}_{C}$	+	;	K	[	k	{
12	C	$^{F}_{F}$	$^{F}_{S}$	,	<	L	\	l	|
13	D	$^{C}_{R}$	$^{G}_{S}$	-	=	M	]	m	}
14	E	$^{S}_{O}$	$^{R}_{S}$	.	>	N	^	n	~
15	F	$^{S}_{I}$	$^{U}_{S}$	/	?	O	_	o	$^{D}_{T}$

Decimal Value	Keystroke	Mono Screen Display	Definition
0	^@		Null
1	^a	☺	Start of heading
2	^b	☻	Start of text
3	^c	♥	End of text
4	^d	♦	End of transmission
5	^e	♣	Enquiry
6	^f	♠	Acknowledge
7	^g	•	Bell
8	^h	◘	Backspace
9	^i	○	Horizontal tabulation
10	^j	◙	Line feed
11	^k	♂	Vertical tabulation
12	^l	♀	Form feed
13	^m	♪	Carriage return
14	^n	♫	Shift out
15	^o	☼	Shift in
16	^p	►	Data link escape
17	^q	◄	Device control 1 (XON)
18	^r	↕	Device control 2
19	^s	‼	Device control 3 (XOFF)
20	^t	¶	Device control 4
21	^u	§	Negative acknowledge
22	^v	▬	Synchronous idle
23	^w	↨	End of transmission block
24	^x	↑	Cancel
25	^y	↓	End of medium
26	^z	→	Substitute
27	^[	←	Escape
28	^\	∟	File Separator
29	^]	↔	Group separator
30	^^	▲	Record separator
31	^_	▼	Unit Separator

Decimal Value	Screen Display	Definition
32		Space (blank)
33	!	Exclamation point
34	"	Quotation mark
35	#	Number sign
36	$	Dollar sign
37	%	Percent sign
38	&	Ampersand
39	'	Apostrophe
40	(	Opening parenthesis
41	)	Closing parenthesis
42	*	Asterisk
43	+	Plus
44	,	Comma
45	-	Hyphen and minus sign
46	.	Period and decimal point
47	/	Slant
48	0	Zero
49	1	One
50	2	Two
51	3	Three
52	4	Four
53	5	Five
54	6	Six
55	7	Seven
56	8	Eight
57	9	Nine
58	:	Colon
59	;	Semicolon
60	<	Less than
61	=	Equals
62	>	Greater than
63	?	Question mark

Decimal Value	Screen Display	Definition
64	@	Commercial "at"
65	A	Uppercase A
66	B	Uppercase B
67	C	Uppercase C
68	D	Uppercase D
69	E	Uppercase E
70	F	Uppercase F
71	G	Uppercase G
72	H	Uppercase H
73	I	Uppercase I
74	J	Uppercase J
75	K	Uppercase K
76	L	Uppercase L
77	M	Uppercase M
78	N	Uppercase N
79	O	Uppercase O
80	P	Uppercase P
81	Q	Uppercase Q
82	R	Uppercase R
83	S	Uppercase S
84	T	Uppercase T
85	U	Uppercase U
86	V	Uppercase V
87	W	Uppercase W
88	X	Uppercase X
89	Y	Uppercase Y
90	Z	Uppercase Z
91	[	Opening bracket
92	\	Reverse slant
93	]	Closing bracket
94	^	Circumflex
95	_	Underscore

Decimal Value	Screen Display	Definition
96	`	Grave accent
97	a	Lowercase a
98	b	Lowercase b
99	c	Lowercase c
100	d	Lowercase d
101	e	Lowercase e
102	f	Lowercase f
103	g	Lowercase g
104	h	Lowercase h
105	i	Lowercase i
106	j	Lowercase j
107	k	Lowercase k
108	l	Lowercase l
109	m	Lowercase m
110	n	Lowercase n
111	o	Lowercase o
112	p	Lowercase p
113	q	Lowercase q
114	r	Lowercase r
115	s	Lowercase s
116	t	Lowercase t
117	u	Lowercase u
118	v	Lowercase v
119	w	Lowercase w
120	x	Lowercase x
121	y	Lowercase y
122	z	Lowercase z
123	{	Opening (left) brace
124	|	Vertical line
125	}	Closing (right) brace
126	~	Tilde
127	△	Delete

47.6

Roman 8 Character Set

Roman 8 characters cannot all be displayed on the IBM PC screen. The following table shows the graphic character that is actually displayed on the PC screen.

Character Code	ROMAN 8 Graphic	Non-EGA Monitor
161	À	A
162	Â	A
163	È	E
164	Ê	E
165	Ë	E
166	Î	I
167	Ï	I
168	´	’
169	`	‘
170	ˆ	^
171	¨	■
172	˜	˜
173	Ù	U
174	Û	U
175	£	£
176	¯	¯
177	Ý	Y
178	ý	y
179	°	°
180	Ç	Ç
181	ç	ç

Character Code	ROMAN 8 Graphic	Non-EGA Monitor
182	Ñ	Ñ
183	ñ	ñ
184	¡	¡
185	¿	¿
186	¤	ø
187	£	£
188	¥	¥
189	§	§
190	ƒ	ƒ
191	¢	¢
192	â	â
193	ê	ê
194	ô	ô
195	û	û
196	á	á
197	é	é
198	ó	ó
199	ú	ú
200	à	à
201	è	è
202	ò	ò

Character Code	ROMAN 8 Graphic	Non-EGA Monitor	Character Code	ROMAN 8 Graphic	Non-EGA Monitor
203	ù	ù	229	Í	I
204	ä	ä	230	Ì	I
205	ë	ë	231	Ó	O
206	ö	ö	232	Ò	O
207	ü	ü	233	Õ	O
208	Å	Å	234	õ	o
209	î	î	235	Š	S
210	Ø	Ö	236	š	s
211	Æ	Æ	237	Ú	U
212	å	å	238	Ÿ	Y
213	í	í	239	ÿ	ÿ
214	ø	ö	240	Þ	Θ
215	æ	æ	241	þ	Θ
216	Ä	Ä	242		
217	ì	ì	243		
218	Ö	Ö	244		
219	Ü	Ü	245		
220	É	É	246	—	—
221	ï	ï	247	¼	¼
222	ß	ß	248	½	½
223	Ô	O	249	ª	ª
224	Á	A	250	º	º
225	Ã	A	251	«	«
226	ã	a	252	■	■
227	Đ	D	253	»	»
228	đ	d	254	±	±

47.6 The HP Line Drawing Character Set

Reflection has an alternate line drawing character set, useful for creating forms. The HP line drawing character set is a group of 28 graphic elements that can be used in combination to create forms and other lined images on the screen. The charts in this reference show an enlarged version of each of the 28 characters in the set, along with its corresponding keystrokes. Each letter key generates only one line drawing character, whether [Shift] is depressed or not. However, the same character may be generated by more than one letter key. Most of the numeric and punctuation keys can generate two line drawing characters: one with [Shift] and a different one when only the character key is pressed.

Press [Ctrl]-[N] to access the line drawing character set; press [Ctrl]-[O] to return to normal characters. Reflection reverts to the normal character set when the cursor moves to a new row.

47.6

Line Drawing Characters

Table 54
Line Drawing Character Choices

Keystroke To Produce	HP Terminal	Code Page 437 (Standard)	Code Page 850	Reflection Defaults
@	╞	╞	├	╞
A/a	┗	└	└	╚
B/b	╋	┼	┼	┼
C/c	█	█	█	█
D/d	┃	│	│	┃
E/e	┃	│	│	║
F/f	└	└	└	└
G/g	┘	┘	┘	┘
H/h	┐	┐	┐	┐
I/i	┫	╕	┐	┐
J/j	┬	┬	┬	┬
K/k	╥	╥	┐	┐
L/l	╥	╥	┐	┐
M/m	╋	┼	┼	┼
N/n	╋	┼	┼	┼
O/o	╡	╕	┐	┐

HP Appendices

Line Drawing Character Choices (continued)

Keystroke To Produce	HP Terminal	Code Page 437 (Standard)	Code Page 850	Reflection Defaults
P/p	┘	┘	┘	┘
Q/q	┏	┌	┌	╓
R/r	┌	┌	┌	┌
S/s	┛	┘	┘	╜
T/t	┐	┐	┐	┐
U/t	┤	┤	┤	┤
V/v	╋	┼	┼	┼
W/w	┓	┐	┐	╖
X/x	█	█	█	█
Y/y	┐	┐	┐	┐
Z/z	▬	█	█	▬
[/{	╡	╡	┤	╡
\/|	╌	─	─	─
]/}	╨	╨	┴	╨
^/~	─	▬	▬	▬
—/–	╥	╥	┬	╥

Line Drawing Character Choices (continued)

Keystroke To Produce	HP Terminal	Code Page 437 (Standard)	Code Page 850	Reflection Defaults
0	╋	┼	┼	╬
1	┣	├	├	╟
2	┫	┤	┤	╢
3	┳	┬	┬	╥
4	┻	┴	┴	╨
5	├	├	├	├
6	┤	┤	┤	┤
7	┬	┬	┬	┬
8	┴	┴	┴	┴
9	═	═	─	═
:	┃	│	│	║
;	━	─	─	═
<	╫	╫	┼	╫
=	┌	┌	┌	┌
>	╪	╪	┼	╪
?	╬	╬	┼	╬

Line Drawing Character Choices (continued)

Keystroke To Produce	HP Terminal	Code Page 437 (Standard)	Code Page 850	Reflection Defaults
!	┣	├	├	╟
"	┫	┤	┤	╢
#	┳	┬	┬	╤
$	┻	┴	┴	╧
%	╟	╟	├	╟
&	╢	╢	┤	╢
'	╤	╤	┬	╤
(	╧	╧	┴	╧
)	║	║	│	║
*	╂	┼	┼	╫
+	╋	┼	┼	╪
,	─	─	─	─
-	└	└	└	└
.	│	│	│	│
/	┼	┼	┼	┼

Appendix I

VT Control Functions Summary

VT Appendices

Sequence	Function	Page
Selecting Control Characters		
^{E}C<SP>F	Select 7-bit C1 characters	544
^{E}C<SP>G	Select 8-bit C1 characters	544
Reflection Escape Sequences		
^{E}C&bN	Next session	545
^{E}C&bR	Connection reset	545
^{E}C&oB<command>^{C}R	Reflection command, comp. code	545
^{E}C&oC<command>^{C}R	Invoke Reflection command	545
^{E}C&oD	Refresh screen	547
^{E}C&oF<command>^{C}R	Reflection command, no comp. code	545
^{D}C1234;0{<cmd>^{S}T	Invoke Reflection command device control sequence	546
^{D}C1234;1{<cmd>^{S}T	Reflection command with comp. code, device control sequence	546
^{D}C1234;2{<cmd>^{S}T	Reflection command, no comp. code, device control sequence	546
^{E}C*s12347^	Serial number request	547

VT Control Functions Summary

Terminal Class, Operating Level

^{E}C&k<n>	Terminal class	548
CSI 61"p	VT102 emulation	548
CSI 62;1"p	VT220 emulation, 7-bit	548
CSI 62;0"p	VT220 emulation, 8-bit	548
CSI 62;2"p	VT220 emulation, 8-bit	548
CSI ?2l	VT52 emulation	548

Display Enhancements

CSI <n>;...<n>m	Select display enhancement	549
^{E}C#3	Double-width, double-height line (top half)	550
^{E}C#4	Double-width, double-height (bottom half)	550
^{E}C#5	Single-width, single-height line	550
^{E}C#6	Double-width, single-height line	550

Cursor Movement

CSI ?25h	Cursor visible	550
CSI ?25l	Cursor invisible	550
CSI <r>;<c>H	Cursor position	551
CSI <r>;<c>f	Horizontal/vertical position	551
CSI <n>C	Cursor forward	551
CSI <n>D	Cursor backward	551
CSI <n>A	Cursor up	551
CSI <n>B	Cursor down	551
CSI <n>Z	Cursor backtab	552
CSI <n>', CSI <n> G	Horizontal position/position absolute	552
CSI <n>a	Horizontal position relative	552
CSI <n>d	Vertical position absolute	552
CSI <n>e	Vertical position relative	552
^{E}CD , ^{I}N	Index	552
^{E}CE, CSI <n>E, ^{N}L	Next line	553
CSI <n>F	Previous line	553
^{E}C7	Save cursor state	553
^{E}C8	Restore cursor state	553
^{E}CM, ^{R}I	Reverse index	554

Screen Display

CSI 3h	Display controls on	554
CSI 3l	Display controls off	554
CSI ?5h	Display enhancements inverse video	554
CSI ?5l	Display enhancements normal	554
CSI 12h	Local echo on	554
CSI 12l	Local echo off	554
CSI >1h	Multi-page mode	555
CSI >1l	Single-page mode	555

Moving the Screen

CSI >0s	Home up	555
CSI >1s	Home down	555
CSI <n>U	Page forward <n> pages	555
CSI <n>V	Page backward <n> pages	555
CSI <n>S	Scroll up <n> lines	556
CSI <n>T	Scroll down <n> lines	556

Display Format

CSI ?3h	Set 132 column mode	556
CSI ?3l	Set 80 column mode	556
CSI <t>;<b>r	Top/bottom margin	556
CSI ?6h	Origin mode–first row	557
CSI ?6l	Origin mode–upper left corner	557

Editing

CSI 4h	Insert mode	557
CSI 4l	Replace mode	557
CSI <n>M	Delete <n> lines	558
CSI <n>L	Insert <n> lines	558
CSI <n>P	Delete <n> characters	558
CSI <n>@	Insert <n> blanks	558

Erasing Text

CSI 0J	Erase from cursor to end of screen	558
CSI 1J	Erase from top of screen to cursor	558
CSI 2J	Erase complete screen	558
CSI 0K	Erase from cursor through end of line	559
CSI 1K	Erase from start of line through cursor	559
CSI 2K	Erase entire line	559
CSI <n>X	Erase character(s)	559

Selective Erase VT220

CSI 0"q	Select erasable characters	559
CSI 2"q	Select erasable characters	559
CSI 1"q	Select protected characters	559
CSI ?0J	Erase unprotected characters, cursor through end of screen	560
CSI ?1J	Erase unprotected characters, top of screen through cursor	560
CSI ?2J	Erase unprotected characters, entire screen	560
CSI ?0K	Erase unprotected characters through end of line	560
CSI ?1K	Erase unprotected characters, start of line through cursor	560
CSI ?2K	Erase unprotected characters, complete line	560

Keyboard

CSI 2h	Keyboard locked	561
CSI 2l	Keyboard unlocked	561
CSI 20h	Auto linefeed on	561
CSI 20l	Auto linefeed off	561
CSI ?8h	Autorepeat on	561
CSI ?8l	Autorepeat off	561
CSI ?7h	End-of-line wrap on	561
CSI ?7l	End-of-line wrap off	561
CSI ?1h	Cursor keys set to application	562
CSI ?1l	Cursor keys set to normal	562

Printer and Disk Control

CSI ?18h	Form feed after PrtSc on	563
CSI ?18l	Form feed after PrtSc off	563
CSI ?19h	Printer extent screen	563
CSI ?19l	Printer extent scrolling region	563
CSI ?5i	Log to printer on	563
CSI ?4i	Log to printer off	563
CSI 5i	Pass-through mode on	563
CSI 4i	Pass-through mode off	563
CSI i	Print screen	564
CSI ?1i	Print cursor line	564
$^{E}{}_{C}$=	Keypad mode application	562
$^{E}{}_{C}$>	Keypad mode normal	562

Requests and Reports

CSI c	Primary DA request	564
CSI >c	Secondary DA request	565
CSI >1;11;0c	Reflection secondary DA response	565
CSI 5n	Device status request	565
CSI 0n	Response no malfunction	565
CSI 6n	Cursor position report request	565
CSI <row>;<column>R	Cursor position report response	565
CSI ?15n	Printer status request	566
CSI ?13n	Printer status response: no printer	566
CSI ?10n	Printer status response: printer ready	566

Resetting and Testing

$^{E}{}_{C}$c	Hard reset	566
CSI !p	Soft terminal reset (VT220)	566
CSI 3g	Tab clear (all)	567
CSI g	Tab clear (at cursor)	567
$^{E}{}_{C}$H	Tab set at current column	567
CSI y	Self-test (disconnect)	567
$^{E}{}_{C}$#8	Screen alignment pattern	567
CSI <n>q	Control LEDs	568

VT52 Control Functions

$^{E}_{C}$<	ANSI mode	568
$^{E}_{C}F$	Character set graphics	568
$^{E}_{C}G$	Character set ASCII	568
$^{E}_{C}A$	Cursor up	568
$^{E}_{C}B$	Cursor down	568
$^{E}_{C}C$	Cursor right	568
$^{E}_{C}D$	Cursor left	568
$^{E}_{C}H$	Home cursor	569
$^{E}_{C}Y$<r><c>	Move cursor	569
$^{E}_{C}K$	Erase to end of line	569
$^{E}_{C}J$	Erase to end of screen	569
$^{E}_{C}Z$	Identification request	569
$^{E}_{C}/Z$	Identification response	569
$^{E}_{C}=$	Keypad mode application	569
$^{E}_{C}$>	Keypad mode normal	569
$^{E}_{C}\wedge$	Log to printer on	570
$^{E}_{C}\_$	Log to printer off	570
$^{E}_{C}W$	Controller mode start	570
$^{E}_{C}X$	Controller mode stop	570
$^{E}_{C}V$	Print cursor line	570
$^{E}_{C}]$	Print screen	570
$^{E}_{C}I$	Reverse linefeed	570

Character Sets

$^{E}_{C}($ *<final char>*	Selects G0	572
$^{E}_{C})$ *<final char>*	Selects G1	572
$^{E}_{C}*$ *<final char>*	Selects G2	572
$^{E}_{C}+$ *<final char>*	Selects G3	572
$^{S}_{I}$	Lock shift G0	573
$^{S}_{O}$	Lock shift G1	573
$^{E}_{C}n$	Lock shift G2	573
$^{E}_{C}o$	Lock shift G3	573
$^{E}_{C}N$	Single shift G2	573
$^{E}_{C}O$	Single shift G3	573

Supported HP Escape Sequences

$^{E}{}_{C}$&*f*	See chapter starting on page 447
$^{E}{}_{C}$&*j*	
$^{E}{}_{C}$&*k*	
$^{E}{}_{C}$&*p*	
$^{E}{}_{C}$&*q*	
$^{E}{}_{C}$&*s*	
$^{E}{}_{C}$&*w*	

VT Appendices

Appendix J

VT Control Functions

47.6 Introduction to VT Control Functions

VT *control functions* make Reflection perform special actions, such as moving the text cursor, adding a line of text, assigning character attributes, and changing character sets.

47.6 A Word about Notation

The following notation is used throughout this appendix:

- Where appropriate, a slash is used through a 0 to distinguish it from an uppercase O. It looks like this: Ø.
- Be sure to note the difference between a lowercase L (*l*) and the number 1 (*1*).
- Reference is occasionally given to a character's position (column/row) on the character set table. For example, the space character is 2/0 (column 2, row 0).
- Parameters you may supply for a sequence are enclosed in angled brackets. For example, when using the control function *CSI <n>A*, replace the *<n>* parameter with the number of lines you want the cursor to move. If *<n>* is omitted, the default is used. For most sequences, the default is 1 (when there is not a corresponding Ø sequence).
- Numeric parameters are represented by ASCII strings. For example, in the sequence *CSI 10;13H*, the numbers are the strings *10* and *13*, not the binary values 10 (ASCII LF) and 13 (ASCII CR).

Numeric parameters are constrained to the range of 0-9999, inclusive. Any numeric parameter with a value greater than 9999 will be interpreted as 9999.

Since the minus sign and the plus sign are not within the range of legal parameter characters, signed numbers will cause a CSI or ^{D}C sequence to be rejected. Therefore, do not include signs with numeric parameters (numeric parameters cannot be negative).

47.6 Control Characters

Control characters are divided into two sets called C0 and C1 controls. Control character functions can be single-character (such as ^{L}F) or multiple-character control functions (such as ^{E}CD to move the cursor down one line). These are described on page 543.

47.6 Single-Character Control Functions

The single-character functions use C0 and C1 control codes, as seen in the Display Control Character Set on page 574. C0 (7-bit) control characters can be used in both 7- and 8-bit environments. C1 (8-bit) control characters are for use only in an 8-bit environment. If you are working in a 7-bit environment, you can still use C1 control functions by converting a C1 function to an equivalent C0 function. The chart on page 542 lists these 7-bit equivalents.

Recognized Control Characters

The table below lists the C0 control characters that Reflection recognizes.

Table 55
C0 (7-bit) Recognized Control Characters

Decimal	Name	Char.	Action
5	Enquiry	^{E}Q	The answerback message is transmitted.
7	Bell	^{B}L	The bell sounds.
8	Backspace	^{B}S	Moves cursor left one position on the current line.
9	Horizontal tab	^{H}T	Moves cursor to the next tab stop.

10	Linefeed	$^{L}{}_{F}$	Causes a linefeed or new line operation.
11	Vertical tab	$^{V}{}_{T}$	Same as linefeed.
12	Form feed	$^{F}{}_{F}$	Same as linefeed.
13	Carriage return	$^{C}{}_{R}$	Moves the cursor to the left margin of the current line.
14	Shift out	$^{S}{}_{O}$	Maps the G1 character set into GL. You designate G1 by using a Select Character Set (SCS) sequence—see page 573.
15	Shift in	$^{S}{}_{I}$	Maps the G0 character set into GL. You designate G0 by using a Select Character Set (SCS) sequence.
17	Device control 1 (XON)	$^{D}{}_{1}$	When **Transmit pacing** in Datacomm Configuration is set to *XON/XOFF*, a $^{D}{}_{1}$ causes Reflection to continue sending characters.
19	Device control 3 (XOFF)	$^{D}{}_{3}$	When **Transmit pacing** is set to *XON/XOFF*, a $^{D}{}_{3}$ causes Reflection to stop sending characters. Reflection cannot resume sending characters until it receives a $^{D}{}_{1}$ control character.
24	Cancel	$^{C}{}_{N}$	When received during an escape or control sequence, the sequence is canceled.
26	Substitute	?	Same as cancel, except that it displays a backwards question mark (error).
27	Escape	$^{E}{}_{C}$	Introduces an escape sequence. Cancels any escape or control sequence in progress.

VT Control Functions

The following 8-bit control characters are interpreted when **Terminal type** in VT Terminal Configuration, Page 1, is set to *VT220-{7/8}*. The 7-bit equivalent is available in VT52 and VT102 environments.

Table 56
C1 Recognized Control Characters

Name	Character	Hex	7-Bit Equiv.	Action
Index	$^{I}_{N}$	84	$^{E}_{C}$D	Moves cursor down 1 line in same column and scrolls at bottom margin.
Next line	$^{N}_{L}$	85	$^{E}_{C}$E	Moves cursor to first position of next line and scrolls at bottom margin.
Horizontal tab set	$^{H}_{S}$	88	$^{E}_{C}$H	Sets a tab stop at at the cursor position.
Reverse index	$^{R}_{I}$	8D	$^{E}_{C}$M	Moves cursor up one line in the same column. Scrolls at the top margin.
Single shift G2	$^{S}_{2}$	8E	$^{E}_{C}$N	Invokes G2 into GL for the next character.
Single shift G3	$^{S}_{3}$	8F	$^{E}_{C}$O	Invokes G3 into GL for the next character.
Device control string	$^{D}_{C}$	90	$^{E}_{C}$P	Introduces a device control string.
Control sequence introducer	CSI	9B	$^{E}_{C}$[	Introduces a control sequence.
String terminator	$^{S}_{T}$	9C	$^{E}_{C}$\	Ends a device control string.

47.6 Multiple-Character Control Functions

There are three types of multiple-character functions:

- Escape sequences, introduced by EC.
- Control sequences, introduced by *CSI* .
- Device control strings, introduced by DC.

Escape Sequences

An escape sequence is a character string beginning with the ASCII escape character EC 1B hex). After receiving the EC control character, Reflection interprets the next received characters as part of the sequence. Escape and control sequences are usually sent to Reflection from the host, but you can also enter them from the keyboard.

Supported HP Escape Sequences

VT102/220 mode supports HP escape sequences (starting on page 447) that begin with the following prefixes:

EC&f	*EC&q*
EC&j	*EC&s*
EC&k	*EC&w*
EC&p	

Control Sequences

A control sequence is a character string beginning with the ASCII control sequence introduction character CSI (98 hex). The 7-bit equivalent is EC[. Control sequences can include variable parameters. For example, *CSI 7;18r* sets the scrolling region, where the top margin is at line 7 and the bottom is at line 18.

Device Control Sequences

A device control sequence begins with the C1 device control sequence introduction character DC (90 hex). The 7-bit equivalent is ECP. A device control string always includes a data string, and ends with a string terminator (ST).

47.6

Using Control Characters in Sequences

You can use the following control characters to interrupt or recover from errors in escape and control sequences:

E_C

Cancels a sequence in progress and begins a new sequence.

C_N

Indicates the present data is in error or cancels a sequence in progress. Reflection interprets the characters following a C_N as usual.

SUB

Cancels a sequence in progress. Reflection interprets the characters following SUB as usual.

47.6

Sending C1 Control Characters to the Host

C1 control characters can be sent to the host in one of two forms:

- As single 8-bit characters (VT220 terminal type required)
- As 7-bit escape sequences

To use 8 bit control characters, set up Reflection as follows:

- Set **Terminal type** to *VT200-8*.

  VT52 and VT100 modes strip the high data bit, so C1 controls are not correctly interpreted in these two modes.
- **Parity** must allow 8 bits (set to *NONE*).

For all other mode settings, the CSI character is replaced with the two-character string E_C[. To send all C1 control characters as 7-bit escape sequences, set **Terminal type** to *VT200-7*, or use the escape sequence below.

Select 7-bit characters

E_C<SP>F

<SP> is the space character SP. To send all C1 control characters as single 8-bit characters, set **Terminal type** to *VT200-8*, or use:

Select 8-bit characters

E_C<SP> G

These sequences have no effect if **Terminal type** is set to *VT52* or *VT102*.

47.6 Reflection Escape Sequences

Reflection offers its own set of control functions.

Next Session

$^{E}_{C}$&bN

Selects the next session when using LAT.

Connection Reset

$^{E}_{C}$&bR

This sequence does one of the following, depending on the environment:

- Resets an internal Hayes Smartmodem 1200B modem card by setting the *out 1* bit of the modem control register to 1 for about 2 seconds.
- Suspends a LAN session.

Invoke Reflection Command

$^{E}_{C}$&oC<command>$^{C}_{R}$

The parameter $^{C}_{R}$ represents a carriage return. The *<command>* parameter is passed to the command language processor as if it had been entered from a batch file. Reflection indicates the success or failure of the command by returning a completion code of **S** or **F** followed by a carriage return.

Reflection Command with Completion Code

$^{E}_{C}$&oB<command>$^{C}_{R}$

This sequence ensures that a completion code (**S** or **F**) is *always* returned.

Reflection Command with No Completion Code

$^{E}_{C}$&oF<command>$^{C}_{R}$

This sequence ensures that a completion code is *never* returned. Reflection acts just like a terminal in this case, and ignores Reflection specific escape sequences.

Reflection Command: Device Control Sequence

You can also use a device control sequence (^{D}C): *

^{D}C1234;Ps{<cmd>^{S}T *

^{D}C

The device control sequence introduction character. You can also use the equivalent 7-bit sequence *^{E}CP*.

1234

This identifies the following sequence as a Reflection-specific sequence.

Ps

Optional condition codes:

0

Execute Reflection command and conditionally return a completion code. This is the default when no code is entered. Reflection indicates the success or failure of the command by returning a completion code of **S** or **F**, followed by a carriage return. You can use Reflection's command language to disable completion codes; See SET DISABLE-COMP-CODES in the *Command Language Manual.* This sequence is similar to *^{E}C&oC<command>^{C}R.*

1

This sequence is similar to the previous one, except that it *always* sends a completion code, regardless of whether completion codes have been enabled or disabled. This sequence is similar to *^{E}C&oB<command>^{C}R.*

2

This sequence *never* returns a completion code. This sequence is similar to *^{E}C&oF<command>^{C}R.*

{

This symbol is the required terminator.

<cmd>

The *<cmd>* parameter is a Reflection command that is passed to the command language processor as if it had been entered from the keyboard or a command file.

^{S}T

The string terminator. You can also use the equivalent 7-bit sequence *^{E}C* .

Refresh Screen

EC&oD

This causes a refresh of the Reflection screen.

Serial number request

*EC*s12347^*

The serial number request is used so the host computer can request the serial number of the Reflection program running on the PC. After exchanging the appropriate block transfer handshake, Reflection sends a 14-character serial number in the form I01+400I123456.* See $SERIAL in the Reflection *Command Language Manual.*

Type of workstation being used

- *H* HP150
- *I* IBM
- *M* Macintosh
- *W* Wang

Product number

- *01* Reflection 1
- *02* Reflection 2
- *03* Reflection 3
- *04* Reflection 4
- *07* Reflection 7

PLUS or non-PLUS

- \+ Program is a *PLUS* version: it has backup/restore capability and LAN support.
- \- Non-PLUS versions have a minus sign.

Version number
In example above, the version number is *4.00*.

License type
I is an individual license, *S* is a server license, and *L* is a LAN license.

* *EC*s12345^* and *EC*s33^* are previous versions of this sequence.

VT Appendices

47.6

Selecting a Terminal Class and Operating Level

You can set up your PC to emulate a VT series terminal using the control functions described here.

Reflection can emulate VT102 and VT52 terminals, and provides partial emulation of VT220 terminals. This flexibility allows you to emulate any of these terminals when an application you need to use is terminal-specific.

Terminal Class

EC&k<n>

Switches between VT and HP terminal class, as follows:

<n> = Ø change to HP
<n> = 1 change to VT

Changing terminal class reactivates the last saved or last activated configuration values for the new terminal class. It also causes an error if background mode has been entered.

Select the operating level by using the control functions below. When you change the operating level, Reflection performs a soft reset; see page 566 for more information.

VT102 Emulation

CSI 61"p

Sets Reflection to emulate a VT102 terminal.

VT200 Emulation

These functions set Reflection to VT200 emulation with 7-bit controls. All 8-bit controls are converted to their 7-bit equivalent before transmitting.

CSI 62;1"p

Set VT200 mode, 7-bit controls as the operating level.

Either of the following sets VT220 mode, 8-bit controls as the operating level.

- *CSI 62;Ø"p*
- *CSI 62;2"p*

VT52 Emulation

CSI ?2l

Set VT52 mode as the operating level (only VT52 control functions are recognized).

47.6 Display Enhancements

You can use control functions for selecting visual attributes for characters and lines. Visual character attributes change the way characters appear on the screen without changing the actual characters.

Select Display Enhancement

CSI <n>;...<n>m

(DEC Mnemonic: SGR) This control function allows you to specify one or more character attributes at the same time.

<*n*> is a number representing a visual attribute listed in the following table. If more than one attribute is present, separate each one with a semicolon. The default is *0*, to clear all attributes.

Table 57
Values for Display Enhancements (<n>)

Enhancement	On	Off
Bold	1	22
Underline	4	24
Blink	5	25
Inverse video	7	27
Clear all attributes		0

After selecting an attribute, Reflection applies that attribute to all new characters received. If you move characters by scrolling, the attributes move with the characters.

For example, use the following to display blinking text in inverse video: *CSI 5;7m.*

Double Width, Double Height Line

^{E}C#3 (top half)

^{E}C#4 (bottom half)

(DEC Mnemonic: DECDHL) Makes the line containing the cursor into either the top or bottom half of a double-width, double-height line.

Only the top or bottom half of the characters on the line are used to display the enlarged characters; for best results make sure the characters used to generate the top half of the line match exactly the characters on the next line that are used to generate the bottom half.

Single-Width, Single-Height Line

^{E}C#5

(DEC Mnemonic: DECSWL) Makes the line with the cursor single-width and single-height (this line attribute is the default for all new lines on the screen).

Double-Width, Single-Height Line

^{E}C#6

(DEC Mnemonic: DECDWL) Makes the line containing the cursor double-width and single-height. Characters already on the line are spaced to every other position. The cursor moves two positions at a time when it is on the line. Any characters that extend beyond the right edge of the double-width, single-height line are lost, even if the line is returned to single width.

47.6 Cursor Movement

Text Cursor Visible/Invisible

CSI ?25h

(DEC Mnemonic: DECTCEM) Makes the cursor visible.

CSI ?25l

Makes the cursor invisible.

Cursor Position

CSI <r>;<c>H

(DEC Mnemonic: CUP) Moves the cursor to row <*r*> and column <*c*>. Normally row 1, column 1 is the upper left corner of the screen. However, if top and bottom margins have been defined and origin mode is set, then the top row of the scrolling region is row 1. The cursor is never positioned beyond the bottom margin or right margin. If the screen width is greater than 80 columns, the display is scrolled horizontally, if necessary, to bring the cursor into view. The default is: *CSI 1;1H*.

See **Origin mode** on page 557.

Horizontal and Vertical Position

CSI <r>;<c>f

(DEC Mnemonic: HVP) The horizontal and vertical position sequence works the same as CUP (described above).

Cursor Forward

CSI <n>C

(DEC Mnemonic: CUF) Moves the cursor right (forward) <*n*> columns. If the screen width is greater than 80 columns, the display is scrolled, if necessary, to bring the cursor into view. If the cursor is at the right margin, it does not move.

Cursor Backward

CSI <n>D

(DEC Mnemonic: CUB) Moves the cursor left (backward) <*n*> columns. If the screen width is greater than 80 columns, the display is scrolled, if necessary, to bring the cursor into view. If the cursor is at the left margin, it does not move.

Cursor Up

CSI <n>A

(DEC Mnemonic: CUU) Moves the cursor up <*n*> lines in the same column. If the cursor is within the scrolling region, it stops at the top margin. If the cursor is above the top margin of the scrolling region, it stops at the top of the screen.

Cursor Down

CSI <n>B

(DEC Mnemonic: CUD) Moves the cursor down <*n*> lines in the same column. If the cursor is within the scrolling region, it stops at the bottom margin. If the cursor is below the bottom margin of the scrolling region, it stops at the bottom of the screen.

Cursor Backtab

CSI <n>Z

Moves the cursor backward along the active line to the <*n*>*th* preceding tab position. The cursor stops at column 1 if the <*n*>*th* tab stop is not found.

Horizontal Position Absolute

CSI <n>' or *CSI <n> G*

Moves the cursor to the <*n*>*th* column without changing the current row.

Horizontal Position Relative

CSI <n>a

Moves the cursor right <*n*> columns without changing the current row.

Vertical Position Absolute

CSI <n>d

Moves the cursor to the <*n*>*th* line without changing the column.

Vertical Position Relative

CSI <n>e

Moves the cursor down <*n*> lines without changing the column.

Index

^{E}CD or ^{I}N

(DEC Mnemonic: IND) Moves the cursor down one row. If the cursor is at the bottom margin, the display scrolls up one line.

Next Line

^{E}*CE, CSI <n>E,* or ^{N}*L*

(DEC Mnemonic: NEL) Moves the cursor to column 1 of the next line. If the cursor is at the bottom margin, the display scrolls up one line.

Previous Line

CSI <n>F

Moves the cursor to the first position of the *<n>th* previous line. If line *<n>* is above the first line, a roll down is performed.

Save Cursor State

^{E}*c7*

(DEC Mnemonic: DECSC) Saves the following Reflection settings:

- Cursor position
- Color and display enhancements
- Character sets (G0, G1, G2, or G3) currently in GL
- The end-of-line wrap setting
- State of origin mode
- Selective erase attribute
- Single shift 2 or 3 functions set

Restore Cursor State

^{E}*c8*

(DEC Mnemonic: DECRC) Restores Reflection to the state saved by ^{E}*c7*. If nothing was saved, this sequence does the following occurs:

- Homes the cursor (upper left)
- Resets origin mode
- Turns all character attributes off
- Maps the ASCII character set into GL

Reverse Index

^{E}CM or *^{R}I*

(DEC Mnemonic: RI) Moves the cursor up one row. If the cursor is at the top margin, the display scrolls down one line and a blank line is inserted.

47.6 Screen Display

These display control functions affect how the screen looks, and how it receives (and sends) information.

Display Controls

CSI 3h

Display control codes (such as ^{E}C and ^{L}F) instead of acting on them.

CSI 3l

Set control codes to function normally. For example, ^{L}F, ^{V}T, and ^{F}F display their codes then move the cursor to the first column of the next line. The *CSI 3l* sequence first displays, then acts upon them.

On the VT terminal, you cannot select this mode with a control function. You must use **DISPLAY FUNCTNS** on the applicable configuration menus.

Normal/Inverse Video

CSI ?5h

(DEC Mnemonic: DECSCNM) Sets display enhancements to inverse video. The screen background is light.

CSI ?5l

Sets display enhancements to normal. The screen background is dark.

Local Echo: Send/Receive Mode

CSI 12h

(DEC Mnemonic: SRM) Turn off local echo so the PC only sends characters to the host. Characters entered on the keyboard are not sent to the display. In remote mode the characters are sent to the host and the host must echo them back to the display.

CSI 12l

Turn on local echo. Characters entered on the keyboard are sent directly to the display. In remote mode, if the host is echoing characters, the characters appear twice on the display.

Multi page/Single page Mode

CSI >1h

Data that scrolls off the screen is retained in display memory. This data can be brought back into view by the Scroll up, Scroll down, Previous page, Next page, Home up, and Home down functions.

CSI >1l

Data that scrolls off the screen is erased from display memory.

47.6 Moving the Screen

The following control functions move the position of the screen window into display memory. They are disabled if **Multi page** mode is *NO*, or top and bottom margins are set.

Home Up/Home Down

CSI >0s = Up

CSI >1s = Down

Moves the screen up and down.

Page Forward <n> Pages

CSI <n>U

Moves the page forward <*n*> pages. When the <*n*> parameter is omitted, this defaults to *1* (i.e., you'll move forward one page).

Page Backward <n> Pages

CSI <n>V

Moves the previous page backward <*n*> pages. When the <*n*> parameter is omitted, or input as 1, this defaults to *1* (i.e., you'll move the page backward one page).

Scroll Up <n> Lines

CSI <n>S

Scrolls the display up *<n>* lines. When the *<n>* parameter is omitted, or input as *1*, this defaults to *1* (i.e., you'll scroll up one line).

Scroll Down <n> Lines

CSI <n>T

Scrolls the display down *<n>* lines. When the *<n>* parameter is omitted, or input as 1, this defaults to *1* (i.e., you'll scroll down one line).

Controlling Display Format

There are control functions that allow you to change the width of the display, the height of the scrolling region, and specify whether or not the cursor can move outside the scrolling margins.

Setting 80 or 132 Columns

This setting selects a display width of 80 or 132 columns.

CSI ?3h

(DEC Mnemonic: DECCOLM) Sets the left margin to 1 and the right margin to 132.

CSI ?3l

Sets the left margin to 1 and the right margin to 80.

If you change the DECCOLM setting, the display is cleared and the cursor is moved to the upper-left corner.

Set Scrolling Region (Top and Bottom Margins)

The scrolling region is the area between the top and bottom margins. This is the area that moves during vertical scrolling. When the margins are selected, the cursor moves to the home position as determined by the origin mode. Rows are counted from 1.

CSI <t>;<b>r

(DEC Mnemonic: DECSTBM) Set the top margin to row *<t>* and the bottom margin to row *<b>*. The *<t>* parameter must be at least two less than *<b>*.

Setting a top or bottom margin has the same effect as setting **Multi page** mode to *NO*; i.e., lines that are scrolled off the screen are deleted from display memory. In addition, the following functions are disabled:

- Previous page
- Next page
- Scroll up/down
- Home up/down

Origin Mode

This setting allows cursor addressing relative to the scrolling margins or the complete display. When you start Reflection, you reset origin mode.

CSI ?6h

(DEC Mnemonic: DECOM) The first row in the scrolling region is the home position. The first row in the scrolling region is now row 1. The cursor cannot move outside the margins.

CSI ?6l

The upper left corner of the screen is the home position. The first row on the screen is now row 1. The cursor can move outside the margins.

47.6 Editing

You can insert and delete data in the scrolling region (the area on the display between the top and bottom margins).

Insert/Replace Mode

CSI 4h

(DEC Mnemonic: IRM) Characters are inserted at the cursor position, and all characters to the right of the cursor move one column to the right. The insert indicator is turned on.

CSI 4l

Characters replace characters at the cursor position. The insert indicator is turned off.

Delete Line(s)

CSI <n>M

(DEC Mnemonic: DL) Delete <*n*> lines at the cursor position and shift lower lines up. Blank lines with normal character attributes are added at the bottom of the scrolling region.

Insert Line(s)

CSI <n>L

(DEC Mnemonic: IL) Insert <*n*> blank lines at the cursor position.

Delete Character(s)

CSI <n>P

(DEC Mnemonic: DCH) Delete <*n*> characters, starting with the cursor position and then deleting characters to the right. As characters are deleted, characters to the right of the cursor move left.

Insert Blank(s)

CSI <n>@

(DEC Mnemonic: ICH) Insert <*n*> blanks before the cursor position on the current line only. This is not used in VT52 mode.

Erasing Text

The erase text functions can affect data inside or outside the scrolling region: they are not restricted by margins.

Erase in Display

(DEC Mnemonic: ED) When you use these sequences, display memory above the top of the screen is retained.

CSI 0J

Erase from the cursor to the end of the screen.

CSI 1J

Erase from the top of the visible screen to the cursor.

CSI 2J

Erase all of the visible screen.

Erase in Line

(DEC Mnemonic: EL) These sequences erase characters on the line that the cursor is on, and clears all character attributes from the erased character positions.

CSI 0K

Erase from the cursor through the end of the line.

CSI 1K

Erase from the start of the line through the cursor.

CSI 2K

Erase the entire line the cursor is on.

Erase Character(s)

CSI <n>X

(DEC Mnemonic: ECH) Erase <*n*> characters from the cursor position to the right (without moving the cursor). A value of *0* or *1* erases one character. This sequence clears character attributes from erased cursor positions.

Note: ECH is not available in VT52 mode.

Selectively Erasing Text in VT220 Mode

With selective erase, you can only erase characters that you define as erasable.

Select Erasable Characters

(DEC Mnemonic: DECSCA) These sequences define characters as erasable or not erasable. It does not affect visual character attributes set by display enhancements (CSI <n>;...<n>m).

Note: The selective erase control functions cannot erase characters defined as not erasable.

CSI 0"q or *CSI 2"q*

Characters following either of these control functions will be erased by the selective erase control function.

Select Protected Characters

CSI 1"q

Protects characters from erasure by the selective erase control functions (see below).

The characters can still be erased by a normal erase sequence.

Selective Erase in Display

(DEC Mnemonic: DECSED) When you use these sequences, display memory above the top of the screen is retained.

CSI ?0J

Erase unprotected characters from the cursor through the end of the screen.

CSI ?1J

Erase unprotected characters from the top of the screen to the cursor.

CSI ?2J

Erase unprotected characters from the entire screen.

Selective Erase in Line

(DEC Mnemonic: DECSEL) These sequences erase only those characters defined as erasable.

CSI ?0K

Erase unprotected characters from the cursor through the end of the line.

CSI ?1K

Erase unprotected characters from the beginning of the line through the cursor.

CSI ?2K

Erase unprotected characters from the complete line.

47.6

Keyboard

Lock/Unlock (Keyboard Action Mode)

CSI 2h

(DEC Mnemonic: KAM) Lock the keyboard so it cannot send characters to the host (the PC ignores all keystrokes that send characters to the host).

CSI 2l

Unlock the keyboard.

Auto Linefeed Mode

(DEC Mnemonic: LNM) Auto linefeed mode selects the characters sent to the host when you press Return. It also controls how the PC interprets linefeed (^{L}F), form feed (^{F}F), and vertical tabs (^{V}T).

CSI 20h

Turns auto linefeed on. Return and keypad Enter send both a ^{C}R and a ^{L}F. A received ^{L}F, ^{F}F, or ^{V}T character moves the cursor to the first column of the next line.

CSI 20l

Turns auto linefeed off. Return and keypad Enter only send a ^{C}R character. A received ^{L}F, ^{F}F, or ^{V}T character moves the cursor down one line in the current column.

Autorepeat Mode

CSI ?8h

(DEC Mnemonic: DECARM) Turn on autorepeat. When a key is held down, it repeatedly sends a character until released.

CSI ?8l

Turn off autorepeat.

End-of-Line Wrap (Autowrap) Mode

CSI ?7h

(DEC Mnemonic: DECAWM) Enable autowrap mode. Received characters automatically wrap to the next line when the cursor reaches the right margin of the display.

CSI ?7l

Turn off autowrap. When the cursor reaches the right margin, it does not move. Additional characters overwrite the character at the right margin until the cursor is explicitly moved.

Cursor Key Mode

CSI ?1h

(DEC Mnemonic: DECCKM) Cursor keys send special *application* sequences.

CSI ?1l

Cursor keys send normal cursor positioning sequences.

Keypad Mode

ESC=

(DEC Mnemonic: DECKPAM) Numeric keypad keys send special *application* escape sequences.

ESC>

Numeric keypad keys send the normal numeric values. The [+] or [Enter] key acts as [Return], and [PrtSc] sends a comma.

Note: See page 581 for the codes sent by the numeric keypad.

47.6 Printer and Disk Control

If a printer is configured, the following control functions send data to the printer. If a disk file has been selected as a "to" device, data is sent to both the printer and the disk file. If the **Printer interface** field is configured to *NONE*, data is sent only to the selected disk file.

Form Feed After PrtSc

CSI ?18h

(DEC Mnemonic: DECPFF) Send a form feed to the printer after either *CSI i* or Alt-PrtSc prints the screen.

CSI ?18l

No form feed is sent to the printer after a print-screen operation.

Printer Extent Mode

CSI ?19h

(DEC Mnemonic: DECPEX) Print the entire screen in a print-screen operation.

CSI ?19l

Print just the scrolling region (data inside the top and bottom margins) during a print-screen operation.

Log to Printer (Auto Print)

CSI ?5i

(DEC Mnemonic: MC) Turn on log to printer mode. Data received into display memory, either from datacomm or from the keyboard, is sent to the printer one line at a time.

Lines must be delimited by LF, FF, or VT to be recognized as lines. The delimiting LF, FF, or VT is also sent to the printer.

CSI ?4i

Turn off logging to the printer.

Pass-through (Printer Controller) Mode

CSI 5i

(DEC Mnemonic: MC) Turn on pass-through mode (also called *printer controller mode*), and characters are sent directly to the printer. Data received from datacomm is passed through to the printer without appearing on the screen. All codes except ^{N}L, $^{D}1$, $^{D}3$, and the *CSI 5i* and *CSI 4i* control functions are passed to the printer.

CSI 4i

Turn off pass-through mode, and return to *normal* (i.e., no logging to printer).

Print Screen

CSI i

(DEC Mnemonic: MC) If the print extent is set to the *scrolling region*, only the scrolling region is printed. If the **Form feed after PrtSc** field is set to *YES*, a ^{F}F is sent to the printer after the print operation.

Print Cursor Line

CSI ?1i

(DEC Mnemonic: MC) Print the line that currently contains the cursor.

47.6 Requests and Reports

Reflection and the host exchange device attribute (DA) sequences to provide the host with the basic operating information. The host uses this information to make the best use of Reflection's features. It also allows the host to determine the cause of any communication errors.

There are two types of DA exchanges between the host and Reflection—primary DA and secondary DA.

Primary DA Request

CSI c

(DEC Mnemonic: DA) The host can query Reflection for *primary device attributes*—the service code, conformance level, and basic attributes using this control function.

Reflection replies according to the setting of the **DA response** field on the Terminal Configuration, Page 1 menu. See the following table.

Table 58
Primary DA Responses from Reflection

Value	Reply	Meaning
VT100	*CSI ?1;0c*	VT100
VT100av	*CSI ?1;2c*	VT100 advan. video
VT100p	*CSI ?1;11c*	VT100 w/printer
VT102	*CSI ?6c*	VT102
VT220	*CSI ?62;1;2;6;7;8c*	VT220 normal
VT220x	*CSI ?62;1;2;6c*	Reflection

Secondary DA Request

CSI >c

This control code is the host request for *secondary device attributes*—Reflection's identification code and hardware options.

Reflection Secondary DA Response

CSI >1;11;0c

Reflection response.

Operating Status

CSI 5n

(DEC Mnemonic: DSR) Host request for Reflection's operating status.

CSI 0n

Reflection's response when there is no malfunction.

Cursor Position

CSI 6n

(DEC Mnemonic: CPR) Host cursor position request.

CSI <row>;<column>R

Reflection responds with the row/column cursor position.

Printer Status

CSI ?15n

(DEC Mnemonic: DSR) Host request for the current printer status.

CSI ?13n

Reflection replies, depending on the **Printer interface** configuration value. The above response indicates that Reflection does not detect a printer. The data set ready (DSR) signal has not been asserted on the printer port since the last power-up or reset.

CSI ?10n

Reflection indicates that the printer is ready. DSR (data set ready) is asserted on the printer port.

47.6 Resetting and Testing

There are three control functions you can use to reset the PC:

- Reset to initial state—hard terminal reset
- Soft terminal reset
- Tab clear

Hard Reset (Reset to Initial State)

$^{E}{}_{C}c$

(DEC Mnemonic: RIS) Reset to last activated state. Identical to the keystrokes [Alt]-[R] and [Ctrl]-[F10] (disconnect).

Soft Terminal Reset

CSI !p

(DEC Mnemonic: DECSTR) This control function selects most of the factory default settings, and is available with VT200 emulation and above.

Table 59
Reset Features

Feature	Value
Character sets	defaults
Character attributes	normal
Cursor key mode	normal
Cursor style	underline
End-of-line wrap	off
Insert mode	off
Keyboard	unlocked
Keypad mode	normal
Origin mode	off
Top/Bottom margins	whole screen
Saved cursor state	defaults
Selective erase	normal

CSI 3g

(DEC Mnemonic: TBC) Clear all tab stops.

CSI g

Clear only the tab stop at the cursor.

^{E}*CH*

Set a tab stop at the current column.

Testing the PC

Power Up Self-Test (Disconnect)

CSI y

Causes a communications line disconnect, drops DTR for 2 seconds, then sends a $^{D}1$ regardless of flow control settings.

Screen Alignment Pattern

^{E}*C#8*

(DEC Mnemonic: DECALN) Display a screen full of *E*s.

Control LEDs

CSI <n>q

Controls the simulated LEDs that appear on the **pf key** label line, in the [F8] position, as follows:

Table 60
Values for Controlling LEDs (<n>)

Value	Meaning
Ø	Clear all LEDs
1	Turn on LED 1
2	Turn on LED 2
3	Turn on LED 3
4	Turn on LED 4

47.6 VT52 Control Functions

ANSI mode

(CSI ?2l, described on page 548, sets VT52 mode.)

$^{E}{}_{C}$<

Exit VT52 mode, and enter VT100 mode.

Character Sets

$^{E}{}_{C}F$

Select the graphics character set.

$^{E}{}_{C}G$

Select the ASCII character set.

Cursor Positioning

$^{E}{}_{C}A$

Move the cursor up.

$^{E}{}_{C}B$

Move the cursor down.

$^{E}{}_{C}C$

Move the cursor right.

$^{E}{}_{C}D$

Move the cursor left.

$^{E}{}_{C}H$

Home the cursor (move the cursor to the top of the display).

$^{E}{}_{C}$*Y<r><c>*

The parameters *<r>* and *<c>* are equal to the character whose binary value is the desired row or column value plus 31. For example, row 2 and column 4 would be $^{E}{}_{C}$*Y!#*.

Erasing

$^{E}{}_{C}K$

Erase from cursor to end of line.

$^{E}{}_{C}J$

Erase from cursor to end of screen.

Identification Request

$^{E}{}_{C}Z$

Host issued identification request.

$^{E}{}_{C}/Z$

Reflection's reply to host issued identification request, as a VT100 emulating a VT52 terminal.

Keypad Mode

$^{E}{}_{C}=$

This sequence, described on page 562, is also available as a VT52 sequence to set the keypad to application mode.

$^{E}_{C}$>

This sequence, described on page 562, is also available as a VT52 sequence to set the keypad to normal mode.

Log to Printer (Auto Print)

$^{E}_{C}$^

Turn on log to printer.

$^{E}_{C}$_

Turn off log to printer.

Controller Mode

$^{E}_{C}$*W*

Start controller mode.

$^{E}_{C}$*X*

Stop controller mode.

Printing

$^{E}_{C}$*V*

Print the cursor line.

$^{E}_{C}$*]*

Print the screen.

Reverse Linefeed

$^{E}_{C}$*I*

Reverse linefeed; the cursor moves up one row in the current column.

47.6 Selecting and Mapping Character Sets

Character sets are selected by control functions and the [Ctrl]-[N], [Ctrl]-[O] keystrokes. For example, in local mode enter $^{E}_{C}$*)0* (zero), then [Ctrl]-[N], and type **abcd**. Notice the results: the characters on the screen are not the usual display representation for these keys.

Now enter [Ctrl]-[O] and type **abcd** again. The characters on the screen are the normal *abcd* characters. The codes stored in display memory and transmitted to the host are the same in both cases; it is only the representation of the code on the screen that is different.

Character Set Support

The IBM PC comes with a 256-character font defined in alphanumeric mode. Reflection divides these characters into four character sets (see page 574):

- ASCII
- UKASCII
- Special graphics
- DEC supplemental graphics

Each character set contains 96 displayable characters. A given character may appear in more than one character set. These character sets can be designated as GØ, G1, G2, or G3 by the escape sequences as shown on the following pages. Each of these may in turn be mapped into GL by control codes or escape sequences. The current values of GL are used to translate all incoming codes into a display representation on the screen. The GL character set translates codes 32 to 127. GR is the Roman 8 character set.

Selecting a Character Set

Character sets are selected and mapped into Reflection's *in-use table*, defining all character sets Reflection can choose from.

When you start Reflection, it places the following default character sets in the in-use table:

- ASCII for GL
- Roman 8 for GR

There are two steps to selecting a different character set:

1. Designate the set as G0, G1, G2, or G3. You can designate up to four character sets and have them ready for use in the in-use table.
2. Map the set. After mapping the set into the in-use table, you can display and send any character from that set using 8-bit code tables. See page 572.

Designating a Character Set

You designate a character set as G0 through G3 by using a select character set (SCS) escape sequence.

Note: You cannot designate a 96-character set as G0.

Following are tables that describe the codes used to select each available character set.

EC<Gn> <set>

(DEC Mnemonic: SCS) This is the sequence for designating a character set. where *<Gn>* specifies G0, G1, G2, or G3:

EC(specifies G0

EC) specifies G1

*EC** specifies G2 (only available in VT220 mode)

EC+ specifies G3 (only available in VT220 mode)

<set> indicates the character set selected.

The character sets that may be designated as the final selection (*<set>*) are:

Table 61
Designate Character Sets

<chr>	Character set
B	ASCII
0	Special graphics (line drawing)
<	Supplemental graphics
A	British

Mapping Character Sets

After you designate a character set as G0, G1, G2, or G3, you must map the set into the left graphic set (GL) of the in-use table. The current values of GL are used to translate all incoming codes in the range 32 to 127 and how you see them on the display.

Locking Shifts

Locking-shifts and *single-shifts* are used to map character sets. When you use a locking shift, the character set remains in GL until you use another locking shift:

${}^{S}{}_{I}$, ${}^{S}{}_{O}$

Locking Shift	Keystroke	Control Code	Function
LS0 (locking shift 0)	Ctrl - O	${}^{S}{}_{I}$	Map G0 into GL
LS1 (locking shift 1)	Ctrl - N	${}^{S}{}_{O}$	Map G1 into GL

${}^{E}{}_{Cn}$, ${}^{E}{}_{Co}$

The following locking shift functions are available in VT200 mode and above:

Locking Shift	Control Code	Function
LS2 (locking shift 2)	${}^{E}{}_{Cn}$	Map G2 into GL
LS3 (locking shift 3)	${}^{E}{}_{Co}$	Map G3 into GL

Single Shifts

When you want to display just the next single character using a different character set, use a single shift. A single shift maps the G2 or G3 set into GL for one displayable character. GL then automatically reverts to its mapping.

Single Shift	7-Bit Equivalent	Function
Single shift 2	${}^{E}{}_{CN}$	Maps G2 for next character into GL
Single shift 3	${}^{E}{}_{CO}$	Maps G3 for next character into GL

47.6 ASCII Character Set

Decimal Value	→	0	16	32	48	64	80	96	112
↓	Hexa-decimal Value	0	1	2	3	4	5	6	7
0	0	NU	DL		0	@	P	`	p
1	1	SH	D1	!	1	A	Q	a	q
2	2	SX	D2	"	2	B	R	b	r
3	3	EX	D3	#	3	C	S	c	s
4	4	ET	D4	$	4	D	T	d	t
5	5	EQ	NK	%	5	E	U	e	u
6	6	AK	SY	&	6	F	V	f	v
7	7	BL	EB	'	7	G	W	g	w
8	8	BS	CN	(	8	H	X	h	x
9	9	HT	EM	)	9	I	Y	i	y
10	A	LF	SB	*	:	J	Z	j	z
11	B	VT	EC	+	;	K	[	k	{
12	C	FF	FS	,	<	L	\	l	|
13	D	CR	GS	-	=	M	]	m	}
14	E	SO	RS	.	>	N	^	n	~
15	F	SI	US	/	?	O	_	o	DT

47.6

UKASCII Character Set

Decimal Value	→	32	48	64	80	96	112
↓	Hexa-decimal Value	2	3	4	5	6	7
0	0		0	@	P	`	p
1	1	!	1	A	Q	a	q
2	2	"	2	B	R	b	r
3	3	£	3	C	S	c	s
4	4	$	4	D	T	d	t
5	5	%	5	E	U	e	u
6	6	&	6	F	V	f	v
7	7	'	7	G	W	g	w
8	8	(	8	H	X	h	x
9	9	)	9	I	Y	i	y
10	A	*	:	J	Z	j	z
11	B	+	;	K	[	k	{
12	C	,	<	L	\	l	|
13	D	-	=	M	]	m	}
14	E	.	>	N	^	n	~
15	F	/	?	O	_	o	

VT Appendices

47.6

DEC Special Graphics

Decimal Value	→	32	48	64	80	96	112
↓	Hexadecimal Value	2	3	4	5	6	7
0	0		0	@	P	◆	— SCAN 3
1	1	!	1	A	Q	▒	— SCAN 5
2	2	"	2	B	R	HT	— SCAN 7
3	3	#	3	C	S	FF	— SCAN 9
4	4	$	4	D	T	CR	├
5	5	%	5	E	U	LF	┤
6	6	&	6	F	V	°	┴
7	7	'	7	G	W	±	┬
8	8	(	8	H	X	NL	│
9	9	)	9	I	Y	VT	≤
10	A	*	:	J	Z	┘	≥
11	B	+	;	K	[	┐	π
12	C	,	<	L	\	┌	≠
13	D	-	=	M	]	└	£
14	E	.	>	N	^	┼	·
15	F	/	?	O		— SCAN 1	

47.6 DEC Supplemental Graphics Character Set

Decimal Value / ↓	→ / Hexadecimal Value	32 / 2	48 / 3	64 / 4	80 / 5	96 / 6	112 / 7
0	0		°	A	¿	à	¿
1	1	¡	±	A	Ñ	á	ñ
2	2	¢	±	A	O	â	ò
3	3	£	$^{C}{}_{N}$	A	O	a	ó
4	4	¿	¿	Ä	O	ä	ô
5	5	¥	µ	Å	O	å	o
6	6	¿	¶	Æ	Ö	æ	o
7	7	§	•	Ç	O	ç	ö
8	8	¤	¿	E	O	è	ø
9	9	c	$^{E}{}_{M}$	É	U	é	ù
10	A	ª	º	E	U	ê	ú
11	B	«	»	E	U	ë	û
12	C	¿	¼	I	Ü	ì	ü
13	D	¿	½	I	Y	í	ÿ
14	E	¿	¿	I	¿	î	¿
15	F	¿	¿	I	ß	ï	

47.6 Keyboard Codes

In VT mode, the typewriter keyboard generates the normal ASCII characters. Using [Ctrl] in combination with the character keys generates ASCII control characters (see page 574 for a table of ASCII control characters). This reference describes additional codes that can be generated from the keyboard.

The following tables list the control functions sent by the following keys:

- Keyboard Key Codes
- Arrow Keys
- Function Keys
- Editing Keys
- Numeric Keypad

Codes Sent by the Keyboard

Table 62
Keyboard Key Codes

VT102	Reflection	Generated Code
Backspace	Backspace or Ctrl-Backspace	BS or DEL (configurable)
Break	Ctrl-ScrollLock	Break
Delete	Ctrl-Backspace or Backspace	DEL or BS (configurable)
Linefeed	Ctrl-Return	LF
Disconnect	Ctrl-F10	Lowers DTR and RTS for 2 seconds
Esc	Esc	EC
Return	Return	CR
Send ans msg	Ctrl-F2	Answerback

Codes Sent by the Arrow Keys

The VT terminal cursor control keys are available in two ways on the IBM PC keyboard. While NumLock is off, the cursor control keys on the PC keypad generate cursor control. When NumLock is on, Shift plus the cursor control keys on the PC keyboard generate cursor control. In addition, while the **pf keys** labels are displayed (Alt-P), F5 through F8 generate cursor control.

Table 63
Codes Sent by Arrow Keys

	*ANSI Mode**		*VT52 Mode*
Key	Cursor	Application	Cursor or Application
↑	EC[A	ECOA	ECA
↓	EC[B	ECOB	ECB
←	EC[C	ECOC	ECC
→	EC[D	ECOD	ECD

* ANSI mode applies to VT100 through VT200 modes. VT52 is not compatible with ANSI mode.

Codes Sent By the VT220 Keys

Additional VT220 keys available in VT220 mode only.

Table 64
VT220 Key Codes

VT220 Keystroke	Reflection Keystroke	Generated Code
Find	Alt - 1	^{E}C[1~
Insert Here	Alt - 2	^{E}C[2~
Remove	Alt - 3	^{E}C[3~
Select	Alt - 4	^{E}C[4~
Prev Screen	Alt - 5	^{E}C[5~
Next Screen	Alt - 6	^{E}C[6~
F6	Shift - F6	^{E}C[17~
F7	Shift - F7	^{E}C[18~
F8	Shift - F8	^{E}C[19~
F9	Shift - F9	^{E}C[20~
F10	Shift - F10	^{E}C[21~
F11	Alt - F1	^{E}C[23~
F12	Alt - F2	^{E}C[24~
F13	Alt - F3	^{E}C[25~
F14	Alt - F4	^{E}C[26~
F15 (help)	Alt - F5	^{E}C[28~
F16 (do)	Alt - F6	^{E}C[29~
F17	Alt - F7	^{E}C[31~
F18	Alt - F8	^{E}C[32~
F19	Alt - F9	^{E}C[33~
F20	Alt - F10	^{E}C[34~

Codes Sent by the Numeric Keypad

The VT numeric keypad keys are available on the IBM PC keypad when [NumLock] is on. When [NumLock] is off, the keypad generates the cursor control functions. While [NumLock] is on, the keypad may be in either normal or application mode. Normal or application mode is usually selected by host software using control functions.

[F2] from the system keys or [Alt]-[P] accesses the PF function key labels. When the **pf keys** labels are on the screen, [F1] through [F4] act as PF1 through PF4. [F5] through [F8] act as cursor control keys. The rightmost label includes the LED indicators. The PF keys are available as [F1] through [F4] while the **pf key** labels are displayed. The table below assumes an Enhanced keyboard with [NumLock] on.

Table 65
Codes Sent by Numeric Keypad

	*ANSI Mode**		*VT52 Mode*	
Key	Numeric	Applic.	Numeric	Applic.
0	0	$^{E}{}_{C}$Op	0	$^{E}{}_{C}$?p
1	1	$^{E}{}_{C}$Oq	1	$^{E}{}_{C}$?q
2	2	$^{E}{}_{C}$Or	2	$^{E}{}_{C}$?r
3	3	$^{E}{}_{C}$Os	3	$^{E}{}_{C}$?s
4	4	$^{E}{}_{C}$Ot	4	$^{E}{}_{C}$?t
5	5	$^{E}{}_{C}$Ou	5	$^{E}{}_{C}$?u
6	6	$^{E}{}_{C}$Ov	6	$^{E}{}_{C}$?v
7	7	$^{E}{}_{C}$Ow	7	$^{E}{}_{C}$?w
8	8	$^{E}{}_{C}$Ox	8	$^{E}{}_{C}$?x
9	9	$^{E}{}_{C}$Oy	9	$^{E}{}_{C}$?y
-	(minus)	$^{E}{}_{C}$Om	-	$^{E}{}_{C}$?m
,	(comma)	$^{E}{}_{C}$Ol	,	$^{E}{}_{C}$?l
.	(period)	$^{E}{}_{C}$On	.	$^{E}{}_{C}$?n
Enter	$^{C}{}_{R}$	$^{E}{}_{C}$OM	$^{C}{}_{R}$	$^{E}{}_{C}$?M
PF1	$^{E}{}_{C}$OP	$^{E}{}_{C}$OP	$^{E}{}_{C}$P	$^{E}{}_{C}$P
PF2	$^{E}{}_{C}$OQ	$^{E}{}_{C}$OQ	$^{E}{}_{C}$Q	$^{E}{}_{C}$Q
PF3	$^{E}{}_{C}$OR	$^{E}{}_{C}$OR	$^{E}{}_{C}$R	$^{E}{}_{C}$R
PF4	$^{E}{}_{C}$OS	$^{E}{}_{C}$OS	$^{E}{}_{C}$S	$^{E}{}_{C}$S

* ANSI mode applies to VT100 through VT200 modes. VT52 is not compatible with ANSI mode.

Appendix K

Error and Other Screen Messages

The following pages list and explain error and other screen messages that may appear during Reflection operation. Messages are listed in alphabetical order, although *The* at the beginning of a message is ignored.

(HP 3000 error message and number) (Capital Letters)

A file transfer was aborted by the PCLINK2 program running on the HP 3000. The error is indicated by the HP 3000 error message text and error number, typically an MPE file system error (see your MPE manual). This error is frequently due to file size. Purge the existing file if the file you are attempting to overwrite is smaller.

Aborted by user

A file transfer was aborted by the user.

Abort or Retry

When an error is encountered, you are requested to either abort or retry the operation. If you are able to fix the problem by inserting the correct disk in the drive or turning on the printer, retry the operation.

Access denied

You cannot write to a read-only protected file.

Bad character in node name

AdvanceNet message —the node name (name of host computer) has a syntax error. An illegal character (a control character) was entered in the network node name given.

Bad field length in node name
AdvanceNet Message - The name entered as the network node name is too long. Check it and reenter the name. The node name has 3 fields, each of which is restricted to 15 characters.

Bad or missing command file password
You attempted to invoke an encrypted command file without the required password, or entered the password incorrectly.

Build of host program file failed
The HP 3000 BUILD command failed when Reflection tried to create the PCLINK2 program file during PCLINK2 uploading to the HP 3000. You may not have save access to the group and account.

Can't change term class AFTER installing a resident program
If you SHELL to DOS and install a memory resident program, you cannot (once you return to Reflection) change terminal class.

Can't change term class once background mode has been entered
Once Reflection has been put into the background the terminal class cannot be changed from HP to DEC or vice versa.

Can't find PCLINK2.PUB
In trying to upload the program PCLINK2.PUB from the PC diskette to the HP 3000, Reflection could not find the PCLINK2.PUB program on the disk.

Can't find R?.EXE
As soon as Reflection starts up, it searches for the R?.EXE file and *
opens it to read overlays. This message appears if it fails to find the file. *
If Reflection cannot load its overlays, it cannot run at all. *

Can't find R1.HLP
The file R1.HLP (or R7.HLP) contains the text for all command line help functions. R1.HLP must be in the current directory in order for the HELP command to function, unless the DOS SET environment command indicates where the R1.HLP file can be found.

Communications driver software not found
The PC software for the network has not been loaded onto the PC or cannot be found.

Connected
AdvanceNet message or other LAN connection has been established with the requested node.

Could not create DOS file

This error can occur at the start of a file transfer from the host to the PC. The output file could not be created on the PC's disk. The disk or directory may be full, the write-protect tab may be on, or the disk may be damaged.

Could not invoke FCOPY utility

Reflection uses the HP 3000 FCOPY program to upload the PCLINK2 program by issuing the command RUN FCOPY.PUB.SYS. For some reason, the HP 3000 was not able to run FCOPY.

Could not save program file

A PCLINK2 upload error message. You do not have write access or save access to the group and account you specified for the PCLINK2 program.

Critical DOS error #<n>

Check the DOS Technical Reference Manual for the explanation of the error number indicated.

Data error (CRC)

DOS error message. Your diskette is probably damaged.

Devices may not share the same com port

You have set up your datacomm and printer to use the same port. Change the port designation on either one.

Disconnected

You have been disconnected from the network software. The LAN connection was either not established or lost.

Disk (DOS) error

DOS error message. A DOS error was detected.

The disk is full — change disks and try again

Insert a new, formatted floppy diskette to continue the operation you are performing.

Disk read failed

Reflection attempted to read from the disk and received an error. Perform a CHKDSK operation at the DOS prompt.

Disk space exhausted - insert new disk
This message is displayed when a disk file fills up the disk during a file transfer from the host. The function key labels are changed, giving you the option of continuing the file on a new disk or cancelling the file transfer.

DOS 2.0 or later required
Reflection does not run on versions of DOS previous to 2.0.

Drive not ready
Insert a diskette and close the door of the requested disk drive.

Duplicate filename or file not found
The specified filename is already in use or the file cannot be found.

Encryption version mismatch
The command file you are trying to run was encrypted with a CRYPT utility that is now obsolete. The command file can only be run with older copies of Reflection. Use the current CRYPT utility to re-encrypt the original command file.

Enter filename:
This message appears when you save a configuration file or open a disk file for output. If a default filename is listed, you can use that name by pressing Return. If you want to clear the message without entering a filename, press the function key labeled **CANCEL**. Otherwise, enter a filename in the DOS format (**drive:filename.ext**) and press Return.

Error opening file
The DOS file could not be opened.

Error opening log file
Reflection automatically creates separate log files for backups and restores, BACKUP.LOG and RESTORE.LOG. If you receive this error message, perform a CHKDSK to see if there is a problem with the disk.

Error loading state file
Reflection attempted to read the state file and received an error. The file *
may have been corrupted in some way or a disk error has occurred. *
Reflection will not run. Rename or delete the state file and try running *
Reflection again. *

Fatal error attempting to load overlay
This message appears if Reflection is unable to access the R?.EXE file *
for loading overlays. Press Return to clear the message; Reflection *

then exits immediately to DOS. This error can also occur if you *
download a new copy of Reflection, overwriting the copy you are using. *

Feature not included in this version

Reflection products vary in capability. Depending on what product you purchased, you may or may not have the part of the program (overlay) necessary for what you would like to do. Call Walker Richer & Quinn regarding an upgrade or exchange.

File corrupted, but backed up

During the backup process, a discrepancy was found between the file's actual size and the size as indicated in the directory entry. In the rare case that file corruption is detected after the file has begun to be backed up, Reflection always sends the exact number of bytes as indicated in the file's directory entry. This may mean padding with binary zeros or truncating the file. You may wish to use the DOS CHKDSK command to check for corruption on this disk.

File corrupted, NOT backed up

Before a file is backed up, it is checked to ensure that the file size matches the file's directory entry. This error message indicates that there is a discrepancy between the actual size of the file and the size as indicated in the directory. You may wish to use the DOS CHKDSK command to check for corruption on your disk.

File exists

The file already exists. Either delete the file or use another name.

<Filename> does not exist

The file you are attempting to send to the host could not be found. Check your drive and directory information as well as the filename.

File not found

The requested file was not found. If this is a Reflection support file and you installed only the minimum number of Reflection files, the requested file may be on your master disks.

File R#PRINT.CFG not found

The printer configuration file was not found. Default values are used for printing display enhancements (see **Esc xfer** on page 118), and for compressed or expanded printing.

File transfer link failed

This message can be the result of several conditions:

- A read fault, when an error is encountered when trying to read from the disk.
- Bad data. When Reflection consecutively receives strings of unsolicited or indecipherable data. The **Error retry limit** field of the File Transfer Configuration menu specifies the number of consecutive occurrences of this condition before the message is displayed and the transfer is terminated (see page 368).
- An unprocessable message. This occurs when Reflection receives a packet that passes the CRC check, but Reflection doesn't understand what to do with it. This occurs when Reflection and PCLINK2 get out of synch in the transfer protocol. The file transfer was not completed; try the file transfer again.
- A receive timeout. This can be caused by consecutive timeout periods without receiving any data response from the host, or Reflection does not get the normal DC1 prompt from the host. The transfer is terminated.
- Too many NAKs. During a file transfer, more than 20 consecutive NAKs were received. That is, the host program was unable to read 20 consecutive packets. Change **File transfer link** on the File Transfer Configuration screen to *7-BIT*, and confirm that the Reflection parity matches the host parity.

File would not fit in memory - the first part has been loaded
This message is displayed at the end of a **READ DISK** operation if the file being loaded into display memory is larger than the memory space. The program loads as much of the file as possible, then quits reading.

Host software requires OLD-WRQ transfer protocol
This message is accompanied by error code 26. You have a host startup sequence that specifies PCLINK, VAXLINK, or UNIXLINK, but you have *WRQ* set as your protocol. If you want to use this startup sequence, change **File transfer protocol** on the File Transfer Configuration menu to *OLD-WRQ*. * * * * *

Incompatible configuration file version - using default values
Reflection stores a version number as part of each configuration file. If the version number of the configuration file does not match the expected version number, this message is displayed. If the default values are not satisfactory, you must reconfigure.

Incompatible host software - you must do an upload

The version of the host file transfer program as named in the **Host startup sequence** field of the File Transfer Configuration menu is not compatible with the version of Reflection you're running. You must upload the current version of the host software from the distribution diskette (see page 347).

Insufficient memory

There is not enough memory to run Reflection. If you have a TSR program loaded, exit it and try to load Reflection again.

Insufficient memory for print buffer

The print buffer and display memory share the memory available in your PC after DOS, Reflection, and any other memory-resident programs have been loaded. The first time you configure a printer buffer it is shared with display memory. Only as much room as is currently available is allocated. If display memory is full or nearly full, there is very little room left for the print buffer. You can free space in display memory by homing the cursor ([Ctrl]-[Home]) and clearing memory ([Alt]-[J]). The next time you load Reflection the memory allocations will be separate for the printer buffer and display memory.

Integrity test failed - start over

When Reflection uploads the host software, it sends the file to the host twice, then compares the two files. If they are not identical, this message is displayed. Perform the upload again.

Invalid command

Reflection did not recognize the name of the command or command file you entered on the command line.

Invalid directory

Reflection could not find the directory you specified.

Invalid MPE filename

You must enter a valid MPE filename.

Invalid syntax

Reflection recognized the command you entered; however, the form of the parameters that go with the command is incorrect.

Local file does not exist

During a file transfer to the host, you specified a local filename that Reflection could not find in the specified directory of the specified disk.

Local file exists

This message is displayed when you're transferring a file from the host to the PC, and the file already exists on the PC's disk. See page 340.

Memory full

This message is displayed when the overflow protect feature (memory lock enabled with the cursor in the top row) is enabled, and memory is full (see page 319). Reflection does not allow the entry of more data into display memory until you disable memory lock or until you delete one or more lines from display memory.

Network error

The network software reports an unexpected error. There is a problem with the network. See your system manager.

Network software missing

The communications driver software has been found, but the network software could not be located.

Node name missing

AdvanceNet error—You tried to connect to the network without entering a node name. Reenter the connect sequence with the node name.

No help available for this topic

Type **HELP** and press $\boxed{\text{Return}}$ for a list of help topics.

No host response

This message is displayed at the start of a file transfer if Reflection does not receive the expected start-of-text character from the host system. This message appears if the host program was not uploaded or if the **Host startup sequence** parameter is incorrect. It is also displayed if another program is running on the host, so that the RUN command cannot be executed.

No target label

The label specified in a GOTO command does not exist in the command file.

Not a valid state file

Your Reflection serial number does not match the one in the state file *
you are trying to load. Reflection comes up with a blank screen with *
whatever settings are contained in the configuration file (if there is no *
configuration file, it will use its defaults). The invalid state file *

(*R<n>STATE.STA*) is automatically deleted (unless the /SK startup *
switch was used) and you should not see the message again. *

No "TO" devices

The printer or a disk file must be specified as a "to" device before you can start logging. See the OPEN command.

Not enough available memory to run Reflection

Either your computer does not have enough memory, or DOS has allocated some memory to another program. See page 327 for memory management options.

Not enough memory to run with selected video driver

In Reflection 7, you can decrease memory requirements by loading Reflection with **/G0**; see page 327 for other memory management options.

NOT restored, cannot open file

The file indicated in the RESTORE command is a DOS device on the receiving PC. There is a filename conflict with the PC device name.

NOT restored, file changed since backup

The file indicated in the RESTORE command was not restored because the file has been modified since the time it was backed up. This message occurs only if the **/K** parameter is used on the RESTORE command.

NOT restored, not enough disk space

There was not enough room on the disk to restore the file indicated in the RESTORE command. This message is stored in the RESTORE.LOG file next to any file not restored. The restore operation continues so that smaller files may be restored.

NOT restored, system, hidden or read-only file

The file indicated in the RESTORE command exists on the PC as a system, hidden, or read-only file. In the default case, RESTORE does not take place over an existing system, hidden, or read-only file. Use the **/H** parameter if you wish to restore this file.

No workstation session resources

AT&T LAN error. The workstation network software cannot connect to a node because the workstation can not support any more virtual circuits.

Out of paper

DOS error – check the paper in your printer.

Out of space in backup file

The BACKUP file on the HP 3000 is too small. Restore the backed up files and create a new or second backup file.

Path not found

The path indicated does not exist on the current drive.

When you are printing to disk (**Printer interface** is set to *FILE*), this *
message indicates that the path in the **File name** field is invalid. Check *
the Printer Configuration menu, or bring up the command line and type *
VERIFY PRINTER-FILE. *

Print buffer is not empty - is it OK to erase it?

Whenever you change the size of the printer buffer in the Printer Configuration menu, the buffer is completely rebuilt, which requires deleting the old buffer. When this message is displayed, you must press either the *YES* or the *NO* function key. If you press *NO*, the buffer is not changed. If you press *YES*, the buffer is rebuilt, and its old contents are destroyed (see page 141).

Printer fault

Check your printer.

Printer timeout

If you direct output to the printer when the printer is offline, out of paper, or otherwise inoperable, this message is displayed after several minutes have elapsed.

Read fault

An error was encountered when trying to read from the disk.

Received bad data

During file transfer, Reflection consecutively received strings of unsolicited or indecipherable data. The **Error retry limit** field of the File Transfer Configuration menu specifies the number of consecutive occurrences of this condition before the message is displayed and the transfer is terminated (see page 368).

Received unexpected packet type

The PC has received an unexpected packet type. The file transfer was not completed; try the file transfer again.

Record size must be positive numeric

The record size given for a file transfer to the host has been entered incorrectly. Make sure the size indicated is a number greater than zero.

Sector not found

DOS error message. The directory on your disk is probably damaged. Consult your DOS manual and try using the CHKDSK function.

Terminated by user

The file transfer was ended by the user.

There is no configured plotter

You attempted to direct output to a plotter, but none has been configured.

There is no configured printer

You attempted to direct output to a printer, but the Printer Configuration menu has a value of *NONE* in the **Printer interface** field (see page 137).

This function LOCKED

The terminal function you are trying to gain access to has been locked by an escape sequence from the host computer. You may be able to clear the terminal with a hard reset, but this could adversely affect the host software system you are using. Consult your system manager.

Too many open files

The DOS limit for number of open files has been exceeded. The limit can be adjusted by use of the DOS configuration file (CONFIG.SYS); FILES=20 and BUFFERS=20 are recommended. This message also occurs if a command file invokes too many file levels. Consult your DOS manual.

Too much display memory requested; only xxK allocated.

Your PC did not have the requested amount of memory available. The maximum amount was allocated.

Transfer method must be A or B

File transfer method can only be **A** (ASCII) or **B** (binary) .

Transmit error

During file transfer, Reflection expected to find a signal on the CTS, DSR, or DCD pins, but did not. A poor phone connection may have caused your modem to disconnect and drop carrier. If this is not the case, go to the appropriate COM configuration menu and set **CTS required** to *NO* and **DSR required** to *NO*. After doing this, try the transfer again.

Unexpected disk (DOS) error detected
A disk error occurred when trying to perform the operation. Check the disk.

Unknown node
AdvanceNet or other network. The name entered for the node is either not a valid node name or the connection request was unsuccessful because the requested node did not respond.

Valid host filename required
The host filename field must be completed.

Valid local filename required
The local filename field must be completed.

VTCOM internal error
There is a problem with the network software. See your system manager.

VT/DOS error
There is a problem with the network. See your system manager.

Waiting for response from host
When you start a file transfer, Reflection sends out a startup sequence to begin running the host program. It displays the above message while waiting for the host to respond. If you are currently running another program on the host or do not have an active session at all, the host cannot respond to the startup sequence. (Check AUTO LF on the modes keys also; it should be off.)

Warning: this configuration has not been saved or activated
You attempted to exit a configuration menu in which you have made a change without activating the configuration or saving it to disk.

Wildcard process failed
This message is issued by an HP host if PCLINK2 was uploaded to an *
account that does not have process handling (PH) capability; the ability *
to use wildcards to receive files on the PC is disabled. See page 348. *

Write fault
DOS error message. Your disk is probably damaged.

Write-protected diskette
DOS error message. You have a write-protect tab on your diskette.

You cannot run Reflection:
The current number of users is already at the maximum of <n>

If you exceed the allowed number of users for a server version of *
Reflection, the program does not run and this message is displayed at *
the DOS prompt. *

You can increase the number of users by purchasing a "bump" disk *
from Walker Richer & Quinn. *

You must configure at least xxK of free memory

Memory to use the hot-key must include room for COMMAND.COM and for storing the video image when not in use. Set a lower display memory size or eliminate the other DOS application from memory.

Glossary

ACK

A special character sent to acknowledge that a previous action or operation is correct.

alphanumeric cursor (alpha cursor)

The cursor that indicates your screen working position when in alpha text mode. Normally displayed as a blinking underline character. When Ins is enabled, the alpha cursor becomes a blinking rectangle; when the graphics display is on, the cursor doesn't blink. See *alpha text.*

alphanumeric display (alpha display)

The screen window into alphanumeric memory. The alpha display can be turned on and off without erasing the contents of alphanumeric memory. Alpha display may be active alone or simultaneously with graphics display.

alphanumeric memory (alpha memory)

The temporary storage area that holds alphanumeric data. Alpha memory may be several screen *pages* long. See *graphics memory.*

alphanumeric text (alpha text)

Normal display text characters. These characters appear when you type in alpha text mode. They are stored in alphanumeric memory. See *graphics text.*

Alt

Alternate. A key that, when pressed, signals the computer of a change in the meanings of other keys struck in conjunction with it.

ASCII

The most common coding system for converting letters, numerals, and other symbols into a form usable by a computer and its peripherals. The ASCII code is also used to transmit characters between computers and from a computer to its peripherals (printers, modems, etc.). *ASCII* stands for *American Standard Code for Information Interchange.* See page 519 for a complete table of the ASCII character set.

asynchronous communications

In the transmission of data, the time between transmission of each character varies. To get the receiving end ready each character is introduced with a bit or two of information that mean *start.* So that the receiving end knows that this piece of transmission has ended, each transmission is ended with a bit or two that mean *stop.*

background

A *background* program is not currently present on the PC screen, but is available. Input from the keyboard does not affect the background program.

background processing

Performing tasks while in the background. This is sometimes called *multitasking.* These tasks might include a simple connection to the host computer, file transfers from the PC to the host, or the execution of Reflection command files.

baud rate

The rate at which data is transmitted. Baud rate is often used to mean bits per second (BPS). Strictly speaking, baud rate is the number of times per second that the carrier signal is modulated.

binary

Pertaining to base 2 numeric representations. The decimal number 7 is represented as 111 in the binary number system.

bit

Short for binary digit, the smallest unit of information in a computer system. Each bit has a value of 0 or 1. A collection of eight bits is a byte.

block mode

A mode of terminal operation in which data is edited locally (within the terminal) before being transferred as a multi-character block to the host computer. The opposite of *character* mode.

boot

Load the operating system into memory.

BPS

Bits per second, the rate at which data is sent over a communications link. See *baud* rate.

byte

A group of eight binary digits (bits) that can be used to represent a letter, special character, or numeral.

carriage return

An ASCII character that causes the character following it to be displayed and/or printed at the first position of a line. When combined with a linefeed character, the carriage return causes the character following to be displayed and/or printed at the first position of the following line.

carrier signal

A signal that is set by a modem on pin 8 of an RS-232 cable to indicate that a connection has been established with another modem, thus permitting data communication.

character mode

A mode of terminal operation in which each character is transmitted to a host computer as it is keyed through the keyboard. Opposite of *block mode.*

clear to send (CTS)

A signal transferred from terminal to modem stating that the modem is ready to accept data for transmission.

command variable

Reflection command language variable. See *variable.*

configuration

The process of customizing a program to a user's preferences and equipment requirements. For example, a prepackaged terminal emulation program, such as Reflection, can be configured to communicate with a host computer either by direct cable connection or by modem.

control

See Ctrl.

control character

A non-text character that signals the computer to perform an action or acknowledge a condition, e.g., a carriage return or linefeed.

Ctrl

The Control key, also expressed as ^. Holding down the Control key changes the meaning of other keys struck in conjunction with it.

CTS (see *clear to send*)

cursor

A symbol on the screen, in the form of a blinking line or rectangle, that indicates where the next typed character will be displayed.

datacomm

Short for data communication, the act of transmitting data from its source to peripheral equipment (e.g., a disk file or printer) or to another computer.

data set

A modem.

data set ready (DSR)

A signal that is set by a modem on pin 6 of an RS-232 cable to indicate that the modem is ready to receive and transmit data. Most modems automatically set this signal *on* as soon as power is applied to the modem.

data terminal ready (DTR)

A signal set *on* by a terminal on pin 20 of an RS-232 cable to indicate that the terminal is ready for data communication. Some modems terminate a telephone connection after this signal has been absent for a certain period of time.

data communications equipment (DCE)

Installed equipment that provides all the functions required to establish, maintain and terminate a connection, and do the signal conversion and coding between the data terminal equipment and the common carrier's line, e.g., a data set or modem.

default

A value or instruction that operates automatically until a user specifies some other value or instruction. For example, the default disk drive (usually drive A) of a PC is made active (logged on) when a computer is turned on. Also, default configuration values are operating instructions built into a software package. Default configuration values can be changed to conform to a user's preferences or equipment requirements.

display
The computer's screen.

display memory
That portion of a computer's memory that receives data or the contents of a file. The contents of display memory can be routed to a hard disk, a diskette, a printer, or a host computer.

DOS
Disk Operating System (pronounced *doss*). Operating system to run on many types of PCs.

download
The process of transferring data from another system to your own.

data terminal equipment (DTE)
Or *data terminating equipment.* Equipment at which communications begin or end, e.g., a computer or terminal.

echo
A host computer is said to *echo* if it immediately transmits back to the source each character that it receives.

ENQ
Enquiry (ENQ) is used to request a response. It may be part of the initialization sequence or the final character of a polling sequence.

escape
$\boxed{\text{Esc}}$ or the escape character, often abbreviated to ^{E}C.

escape sequence
A character string beginning with the ASCII escape character. Such strings constitute a command, usually to the terminal or printer, to perform some action. Escape sequences are usually sent to a terminal from a host computer, but they can also be entered through the keyboard.

field
A part of a record designed to hold one category of data. For example, a mailing list would contain fields for name, address, and zip code. See also *record, file.*

file

A body of data stored at one location on a hard disk or floppy diskette under a unique filename. A file can contain random data, such as text, or can be organized as records, which in turn would contain one or more fields.

fixed-length record format

A file format in which all records have the same length.

foreground

Any program that is currently present on the PC screen is the *foreground* application. Input from the keyboard affects the foreground program.

form feed

The movement of paper in a printer up to the next page. The term is derived from the fact that the form (paper) feeds (moves up) to the start of the next sheet.

format mode

A mode of terminal operation in which the display is made up of protected and unprotected fields. Data can be entered only in unprotected fields.

full duplex

The data communication mode that allows data to be transmitted in two directions simultaneously. Reflection always operates in full duplex, even if the host computer does not echo data entered at the terminal.

graphics cursor

The *crosshair* cursor that marks the working position on the graphics display. See *alphanumeric cursor.*

graphics display

The screen window into graphics memory (see below). Graphics display may be active alone or at the same time as alpha display. It consists of a single page; see *alphanumeric display.*

graphics memory

The temporary terminal storage area that holds graphics display data. Graphics data and alphanumeric data are stored in separate memories. Graphics memory accommodates only one screen *page* of data. See also *alphanumeric memory.*

graphics text

The text generated when you use the character keys in graphics text mode. Graphics text can be displayed on the screen in 8 sizes, 4 orientations, and *roman* (upright) or *italic* (slanted) face. Graphics text resides in graphics memory. See also *alphanumeric text.*

half duplex

A communications link that can transmit in only one direction at a time. Compare *full duplex.*

handshake

A protocol exchange of characters between a terminal and a host computer. The handshake delays transmission until a receiver is prepared to receive data.

host computer

Any computer that receives data from a terminal by direct connection or through a modem.

hot-key

The keystroke that makes a pop-up appear. Reflection allows you to define which keystroke you would like to use from a list of options to prevent interference with other pop-ups.

K

The symbol for kilo, a thousand. Used, for example, to express the capacity of computer or disk memory, in thousands of bytes (kilobytes). One kilobyte actually equals 1,024 bytes.

linefeed

An ASCII control character that moves the cursor or the form (paper) to the next line.

local mode

A mode of terminal operation in which data is not transmitted to a host computer. If data is received from a host computer while a terminal is operating in *local mode*, the data is discarded. Compare *remote mode.*

mode

A way of operating.

modem

A contraction of the words *modulator* and *demodulator.* A modem converts digital signals from a computer to analog (sound) signals for transmission over a phone line, and converts analog signals received through a phone line to digital signals.

MPE

Operating system of the HP 3000 minicomputer. MPE stands for *Multi Programming Executive.*

NAK

A character sent to indicate negative acknowledgment of the preceding action.

node

An individual computer system in a network.

null modem

A unique cable arrangement in asynchronous serial communications directly connecting two pieces of terminal equipment.

operating system

A program that tells a computer what to do and allows the user to issue commands to the computer. For example, PC DOS and MPE are operating systems.

parallel transmission

The movement of data more than one bit at a time. This is accomplished by a cable that has a separate line for each bit position. For example, the IBM PC's parallel printer interface sends eight bits (i.e., one byte) in parallel. Compare *serial transmission.*

parameter

A value that can vary but remains constant in a given context.

parity

In data communications, the addition of a non-information bit to each character transmitted. The bit may be always 0 (space parity), always 1 (mark parity), or either 0 or 1 so as to make the sum of the bits odd (odd parity) or even (even parity). Also see page 94.

PC

Personal computer.

pixel

One small dot on the screen. Many pixels together create screen images – characters, lines, and patterns. The more pixels displayed on the screen, the higher the display resolution.

pop-up

A program that resides in the PC's memory at all times and can be popped up to the screen with a simple keystroke. Programs such as Reflection and Sidekick operate as pop-ups so that they can always be available to you no matter what other program you may be running.

port

A jack or outlet on a computer into which connectors to peripheral devices can be plugged. Usually designated either serial or parallel.

prompt

A symbol on a computer screen, telling the user that the computer is ready to receive a command. Sometimes a prompt is a message on the screen, telling the user to perform some action or indicating the status of an operation.

protocol

A formal set of rules governing communication between two processes and allowing control of the input and outputs in an orderly manner.

record

A collection of related pieces of information grouped together as part of a file, e.g., the name, street address, city, state, and zip code of one individual or company in a mailing list file. See also *field, file.*

remote mode

A mode of terminal operation in which data is transmitted to and received from a host computer. Compare *local mode.*

request to send (RTS)

A signal from a terminal to a modem indicating that the terminal is wanting to send data.

RS-232

A standard cable type, used to connect a PC with its peripheral equipment, another computer, or data communications equipment. The standard specifies the meaning of each of the lines connecting the equipment, and the allowable ranges of voltage and impedance.

serial port

A communication port through which data moves over the same wire, one bit after another.

serial transmission
A method of data transfer in which one bit is sent at a time. Compare *parallel transmission.*

start bits
In data communications, bits transmitted before each data character to signal the beginning of the character.

status request
A request issued by a host computer for information from a terminal. An aid to communication between terminal and host.

stop bits
In data communication, bits transmitted after each data character to signal the end of the character.

terminal
A device that can feed information and receive data from a host computer. A PC becomes a terminal when it is connected to a host computer and running terminal emulation software.

terminal emulation
The ability to communicate with a host computer with your PC as if you were using a terminal made to connect to the host. Reflection emulates several HP and VT terminals depending on the product you are using, and can support a variety of hosts.

terminal mode
Not currently at a Reflection configuration menu or the File Transfer menu; in either remote or local mode.

timeout
The intended operation is assumed to have failed because it was not completed within the time allotted.

typeahead
A feature of Reflection that allows a user to enter data at the keyboard continuously during transmission to a host computer. Typeahead is useful with host computers, such as the HP 3000, that accept keyboard input a line at a time as it is requested by the host.

UART

Universal Asynchronous Receiver/Transmitter. This device receives and transmits data asynchronously and checks incoming data for errors. It is capable of halting transmission when certain events occur, such as the receipt of a character or line error. A common UART is the INS 8250.

UNIX

Operating system designed by AT&T Bell Laboratories. Many versions of the operation system exist; Reflection supports most host versions.

upload

The process of moving data from your system to another system.

variable

An unspecified value or quantity.

variable-length record format

A file format in which records are not necessarily the same length.

VAX

Digital's Virtual Address Extender.

vector

In simplest terms, a line.

VMS

Virtual Memory Operating System. Digital's Operating system for the DEC VAX.

wildcard

In MS-DOS, the symbol ? is used in place of single characters in filenames or extensions, and * is used in place of whole or partial filenames or extensions, indicating *all* for the given field.

Glossary

Index

Index

B

Index

Index

Index

Index

D

Index

Index

Index

E

Index

Index

Index

H

I

Index

Index

M

Index

Index

N

O

Index

Q

R

Index

S

Index

Index

T

Index

U

Index

V

Index

Index

Index

Index

PC

Escape Sequence Summary

REFLECTION® 1/7

Reflection Escape Sequences

ESC &bN	Next session (LAT or CTERM)
ESC &bR	Connection reset
ESC &oA	5-digit absolute cursor position
ESC &oB<command>CR	Reflection command/comp code
ESC &oC<command>CR	Reflection command
ESC &oD	Refresh screen
ESC &oF<command>CR	Reflection command/ no completion code
ESC &oX	Clear typeahead buffer
*ESC *33^*	Request serial number (PC 2622) (doesn't work w/ graphics emulation)
*ESC *s12345^*	Request 9-character serial #
*ESC *s12346^*	Request 11-character serial #
*ESC *s12347^*	Request 14-character serial #

HP Escape Sequences

Terminal Control

ESC !	*ESC "*	132/80 column mode (Zentec)
ESC &f<parameters>		User key control
ESC @		Delay one second
ESC [	*ESC]*	Start/End unprotected field
ESC ^	*ESC ~*	Primary/Secondary status request
ESC `		Sense cursor position, relative
ESC 0		Home up & copy
ESC 1	*ESC 2*	Set/Clear tab
ESC 3		Clear all tabs
ESC 4	*ESC 5*	Set left/right margins
ESC 9		Reset margin defaults
ESC A	*ESC B*	Move cursor up/down
ESC C	*ESC D*	Move cursor right/left
ESC E		Hard reset
ESC F	*ESC H*	Home down/up
ESC G		Cursor to left margin
ESC I		Horizontal tab
ESC J	*ESC K*	Clear display/line
ESC L	*ESC M*	Insert/Delete line
ESC N	*ESC O*	Insert/Delete with wraparound
ESC P		Delete character
ESC Q	*ESC R*	Insert mode on/off
ESC S	*ESC T*	Roll display up/down
ESC U	*ESC V*	Next/Previous page
ESC W	*ESC X*	Format mode on/off
ESC Y	*ESC Z*	Display functions on/off
ESC a		Sense cursor position, absolute
ESC b	*ESC c*	Unlock/Lock keyboard
ESC d		Block transfer request
ESC f		Disconnect

ESC g		Soft reset
ESC h		Home cursor
ESC i		Backtab
ESC j	ESC k	Display/ Remove user-key menu
ESC l	ESC m	Memory lock mode on/off
ESC p through	ESC w	Default user key values
ESC z		Terminal self-test

Cursor Positioning

ESC &a+/-<col>c+/-<row>R	Move cursor, display relative
ESC &a+/-<col>x+/-<row>Y	Move cursor, cursor relative
ESC &a<col>c<row>R	Move cursor absolute, display
ESC &a<col>x<row>Y	Move cursor absolute, screen
ESC &x<n>C	Cursor position mode

Configuration Control

ESC &f1m149P<n>		Modify Return
ESC &f211P<n>		Modify Tab
ESC &f R		Reset Tab/ Return
ESC &j<n>D		Set bell, CR, labels
ESC &k Ø \	ESC &k1\	Terminal class HP/ VT
ESC &k Ø A	ESC &k1A	Auto linefeed off/on
ESC &k Ø B	ESC &k1B	Block mode off/on
ESC &k Ø C	ESC &k1C	CapsLock off/on
ESC &k Ø D	ESC &k1D	Bell off/on
ESC &k Ø I	ESC &k1 I	Previous (7-bit)/No parity (8-bit)
ESC &k Ø J	ESC &k1J	Set frame rate 60/50 (ignored by Reflection)
ESC &k Ø K	ESC &k1K	Auto keyboard lock off/on
ESC &k Ø L	ESC &k1L	Local echo off/on
ESC &k Ø M	ESC &k1M	Modify all mode off/on
ESC &k Ø N	ESC &k1N	SPOW latch off/on
ESC &k Ø P	ESC &k1P	Caps mode off/on
ESC &k Ø R	ESC &k1R	Remote mode off/on
ESC &q Ø L	ESC &q1L	Unlock/Lock configuration
ESC &s Ø A	ESC &s1A	Transmit functions no/yes
ESC &s Ø B	ESC &s1B	SPOW no/yes
ESC &s Ø C	ESC &s1C	Inhibit EOL wrap no/yes
ESC &s Ø D	ESC &s1D	Line/Page mode
ESC &s Ø G	ESC &s1G	Inhibit handshake no/yes
ESC &s Ø H	ESC &s1H	Inhibit DC2 no/yes
ESC &s Ø N	ESC &s1N	Esc Xfer to printer no/yes
ESC &s Ø Z	ESC &s1Z	Check parity no/yes
ESC *d Ø E	ESC *d1E	Normal/Inverse display
ESC *d Ø Q	ESC *d1Q	Underline/Block cursor

Data Operations

ESC &p4^	ESC p6^	Printer status
ESC&p<a>d<b>d<c>d<x>W<string>		Transfer binary data string
ESC &p<a>d<b>d<c>dW<data string>		Transfer ASCII data string
ESC &p<a>d<Y>		Copy data
ESC &p<n>p2 Ø C		Record mode
ESC &p<x>p<y>u<z>C		Printer control action

Display Control & Enhancements

ESC &d<x>, display enhancements, where <x> is one of the following:

@ No enhancement	**H** Half bright
A Blinking	**I** Half bright, blinking
B Inverse video	**J** Half bright, inverse video
C Blinking, inverse video	**K** Half bright, blinking, inverse video
D Underline	**L** Half bright, underline
E Blinking, underline	**M** Half bright, blinking, underline
F Inverse video, underline	**N** Half bright, inverse video, underline
G Blinking, inverse video, underline	**O** Half bright, blinking, inverse video, underline

ESC &ds<x>		Display enhancements, security
ESC &q3t Ø {<x>W		Page width
ESC &r<n>L	ESC &r<n>R	Roll left/right <n> columns
ESC &r<n>U	ESC &r<n>D	Roll up/down <n> lines
ESC &w12F	ESC &w13F	Display on/off
ESC &w5f<n>W		Page width (active value only)

Character Sets & Labels

ESC &j@		Enable user keys, no labels
ESC &jA		Display modes keys
ESC &jB		Display & enable user keys
ESC &jC		Remove message, replace labels
ESC &jR	ESC &jS	Enable/ Disable system, modes, and user keys
ESC)A	ESC)C	Select space set *
ESC)B	ESC)@	Select line drawing/base set **
ESC &j<x>L<string>		Replace labels with message
ESC &f<parms>	ESC &f<x>E	Define/Trigger user keys

Forms Cache

ESC &p9^	Request status of forms buffer
ESC &p9u<form #>pF	Display form
ESC &p9u<form #>pL	Purge form from buffer
ESC &p9u<form #>p<<contents>>L	Store a form of unknown length
ESC &p9u<form #>p<length>L<contents>	Store a form of known length
ESC &p<form #>p9^	Request status of specified forms buffer

* Does not apply to 2392A, 70092, or 70094 terminal IDs.

** Does not apply to 2392A terminal ID.

Graphic Escape Sequences

Graphics Sequences

ESC &kØO	ESC &k1O	Graphics keypad off/on
ESC &p7sF		Print graphics memory
ESC *d<X,Y>O	ESC *d<X,Y>P	Move graphics cursor abs/rel
ESC *dC	ESC *dD	Graphics display on/off
ESC *dE	ESC *dF	Turn alpha display on/off
ESC *dK	ESC *dL	Turn graphics cursor off/on
ESC *dM	ESC *dN	Turn on/off rubberband line
ESC *dQ	ESC *dR	Turn on/off alpha cursor
ESC *dS	ESC *dT	Turn on/off graphics text mode
ESC *jØA	ESC *j1A	Mouse offline/online

Drawing Modes & Patterns

ESC * l<text>CR		Graphics labels
ESC *m1A	ESC *m2A	Clear/Set graphics bit
ESC *m1B	ESC *m2B	Solid/User-defined line
ESC *m1G	ESC *m2G	Solid/User-defined area pattern
ESC *m3A	ESC *m4A	Complement/Jam
ESC *m3B		Current area pattern
ESC *m11B		Point plot
ESC *m<lower><upper>E		Area fill absolute
ESC *m<lower><upper>F		Area fill relocatable
ESC *m<n>B		Select line type (4-10)
ESC *m<n>G		Predefined area pattern (3-10)
ESC *m<row Ø>...<row 7>D		Define an area pattern
ESC *m<x><y>C		Define a line pattern
ESC *m<pen#>H		Set area boundary pen (0-7)
ESC *m<X,Y>J	ESC *mK	Relocate origin absolute/at pen
ESC *mL		Relocate origin at graphics cursor

Graphics Text & Drawing Modes

ESC *m1N	ESC *m2N	Rotated 0/90 degrees
ESC *m3N	ESC *m4N	Rotated 180/270 degrees
ESC *m<1>R		Restore selected graphics defaults
ESC *mR		Restore graphics defaults
ESC *m<origin>Q		Justification (0-9)
ESC *m<n>M		Character size (1-8)
ESC *mO	ESC *mP	Slanted/ Upright characters
ESC *pA	ESC *pB	Lift / Lower pen
ESC *pC		Use graphics cursor as point
ESC *pD		Draw point at pen, lift pen
ESC *pE		Set relocatable origin at pen
ESC *pF	ESC *pG	Data ASCII absolute/incremental
ESC *pH	ESC *pL	Data is ASCII/binary relocatable
ESC *pJ		Data is binary short incremental
ESC *p I	ESC *pK	Data is binary absolute/incremental
ESC *pS	ESC *pT	Start/End polygon area fill
ESC *pZ		No operation is performed

Graphics Status

ESC *s1^	ESC *s2^	Read device ID/pen position
ESC *s3^	ESC *s4^	Read graphics cursor position/ with wait
ESC *s5^	ESC *s6^	Read display size/graphics capabilities
ESC *s7^	ESC *s8^	Read graphics text/zoom status
ESC *s9^	ESC *s1Ø^	Read relocatable origin/reset status
ESC *s11^	ESC *s12^	Read area shading/modifications
ESC *s32		Request mouse status
ESC *s33		Request cursor position

Compatibility Mode

ESC &s1pØQ	ESC &sØp1Q	Scaled/Unscaled compatibility on
ESC &sØp0Q		Compatibility mode off
ESC *tØA		Set carriage return as terminator
ESC *tØB		Page full break, no break
ESC *t1B		Page full break, break
ESC *tØC	ESC *t1C	Page full busy, normal/lock
ESC *t1A		Set return & EOT as terminators
ESC *t2A		Disable all terminators
ESC *wR		Graphics hard reset

Color Graphics Sequences

ESC &f<color parameters>		User key control, Reflection 7
ESC &f<color pairs>c<vid.>vØL		User key control, Reflection 7
ESC *d<pen#>A		Clear graphics memory
ESC *d<pen#>B		Set graphics memory
ESC *e<pen#>B		Select background pen color
ESC *mØG		Dither pattern as polygon pattern
ESC *m1W		User-defined dither pattern
ESC *m<d1,d2,d3>V		Define dither pattern
ESC *m<mode>A		Select drawing mode (0-7)
ESC *m<n>W		Predef. dither patterns (2-12)
ESC *m<pen#>H		Set area boundary pen (0-7)
ESC *m<pen#>X		Select primary pen color
ESC *m<pen#>Y		Select secondary pen color
ESC *n<pen#>X		Graphics text colors
ESC *pU	ESC *pV	Lift/Lower boundary pen

Color Methods

ESC &vØm	ESC &v1m	Selecting RGB/HSL method
ESC &v<0-7>^		Color pair definition status
ESC &v<0-7>i		Color pair to be initialized
ESC &v<0-7>s		Color pair to be selected
ESC &v<decimal><a, b, or c>		Foreground: Red (a), Green (b), or Blue (c)/Hue (a), Saturation (b), or Luminosity (c)
ESC &v<decimal><x, y, or z>		Background: Red (x), Green (y), or Blue (z)/Hue (x), Saturation (y), or Luminosity (z)

Character Set Display—IBM PC with EGA, VGA, or MCGA Adapter

Decimal Value ↓	→ Hexa-decimal Value	0 0	16 1
0	0	N_U	D_L
1	1	S_H	D_1
2	2	S_X	D_2
3	3	E_X	D_3
4	4	E_T	D_4
5	5	E_Q	N_K
6	6	A_K	S_Y
7	7	B_L	E_B
8	8	B_S	C_N
9	9	H_T	E_M
10	A	L_F	S_B
11	B	V_T	E_C
12	C	F_F	F_S
13	D	C_R	G_S
14	E	S_O	R_S
15	F	S_I	U_S

N_U	Null	D_L	Data link escape
S_H	Start of header	D_1	Device control 1 (XON)
S_X	Start of text	D_2	Device control 2
E_X	End of text	D_3	Device control 3 (XOFF)
E_T	End of transmission	D_4	Device control 4
E_Q	Enquiry	N_K	Negative acknowledge (NAK)
A_K	Acknowledge	S_Y	Synchronous idle
B_L	Bell	E_B	End of transmission block
B_S	Backspace	C_N	Cancel
H_T	Horizontal tab	E_M	End of medium
L_F	Linefeed	S_B	Substitute
V_T	Vertical tab	E_C	Escape
F_F	Formfeed	F_S	Field separator
C_R	Carriage return	G_S	Group separator
S_O	Shift out	R_S	Record separator
S_I	Shift in	U_S	Unit separator

Display Functions

Decimal Value ↓	→ Hexa-decimal Value	32 2	48 3	64 4	80 5	96 6	112 7
0	0		0	@	P	\`	p
1	1	!	1	A	Q	a	q
2	2	"	2	B	R	b	r
3	3	#	3	C	S	c	s
4	4	$	4	D	T	d	t
5	5	%	5	E	U	e	u
6	6	&	6	F	V	f	v
7	7	'	7	G	W	g	w
8	8	(	8	H	X	h	x
9	9	)	9	I	Y	i	y
10	A	*	:	J	Z	j	z
11	B	+	;	K	[	k	{
12	C	,	<	L	\	|	|
13	D	-	=	M	]	m	}
14	E	.	>	N	^	n	~
15	F	/	?	O	_	o	

D_T Delete

ASCII

Decimal Value ↓	→ Hexa-decimal Value	160 A	176 B	192 C	208 D	224 E	240 F
0	0		‾	â	Å	Á	Þ
1	1	À	Ý	ê	î	Ã	þ
2	2	Â	ý	ô	Ø	ã	•
3	3	È	°	û	Æ	Đ	µ
4	4	Ê	Ç	á	å	ð	¶
5	5	Ë	ç	é	í	Í	¾
6	6	Î	Ñ	ó	ø	Ì	—
7	7	Ï	ñ	ú	æ	Ó	¼
8	8	´	¡	à	Ä	Ò	½
9	9	\`	¿	è	ì	Õ	ª
10	A	^	¤	ò	Ö	õ	º
11	B	¨	£	ù	Ü	Š	«
12	C	~	¥	ä	É	š	▬
13	D	Ù	§	ë	ï	Ú	»
14	E	Û	ƒ	ö	ß	Ÿ	±
15	F	£	¢	ü	Ô	ÿ	

Roman 8

Character Set Display—IBM PC with CGA, Monochrome, or Hercules Adapter

Decimal Value ↓	→ Hexa-decimal Value	0 0	16 1
0	0	BLANK (NULL)	►
1	1	☺	◄
2	2	☻	↕
3	3	♥	‼
4	4	♦	¶
5	5	♣	§
6	6	♠	▬
7	7	•	↨
8	8	◘	↑
9	9	○	↓
10	A	◙	→
11	B	♂	←
12	C	♀	∟
13	D	♪	↔
14	E	♫	▲
15	F	☼	▼

N_U	Null	D_L	Data link escape
S_H	Start of header	D_1	Device control 1 (XON)
S_X	Start of text	D_2	Device control 2
E_X	End of text	D_3	Device control 3 (XOFF)
E_T	End of transmission	D_4	Device control 4
E_Q	Enquiry	N_K	Negative acknowledge (NAK)
A_K	Acknowledge	S_Y	Synchronous idle
B_L	Bell	E_B	End of transmission block
B_S	Backspace	C_N	Cancel
H_T	Horizontal tab	E_M	End of medium
L_F	Linefeed	S_B	Substitute
V_T	Vertical tab	E_C	Escape
F_F	Formfeed	F_S	Field separator
C_R	Carriage return	G_S	Group separator
S_O	Shift out	R_S	Record separator
S_I	Shift in	U_S	Unit separator

Display Functions

Decimal Value ↓	→ Hexa-decimal Value	32 2	48 3	64 4	80 5	96 6	112 7
0	0	BLANK (NULL)	0	@	P	\`	p
1	1	!	1	A	Q	a	q
2	2	"	2	B	R	b	r
3	3	#	3	C	S	c	s
4	4	$	4	D	T	d	t
5	5	%	5	E	U	e	u
6	6	&	6	F	V	f	v
7	7	'	7	G	W	g	w
8	8	(	8	H	X	h	x
9	9	)	9	I	Y	i	y
10	A	*	:	J	Z	j	z
11	B	+	;	K	[	k	{
12	C	,	<	L	\	|	|
13	D	-	=	M	]	m	}
14	E	.	>	N	^	n	~
15	F	/	?	O	_	o	Δ

D_T Delete

ASCII

Decimal Value ↓	→ Hexa-decimal Value	160 A	176 B	192 C	208 D	224 E	240 F
0	0		‾	â	Å	A	Θ
1	1	A	Y	ê	î	A	Θ
2	2	A	y	ô	Ö	a	•
3	3	E	°	û	Æ	D	µ
4	4	E	Ç	á	å	d	
5	5	E	ç	é	í	I	
6	6	I	Ñ	ó	ö	I	—
7	7	I	ñ	ú	æ	O	¼
8	8	'	¡	à	Ä	O	½
9	9	'	¿	è	ì	O	ª
10	A	^	ø	ò	Ö	o	º
11	B		£	ù	Ü	S	«
12	C	~	¥	ä	É	s	▬
13	D	U	§	ë	ï	U	»
14	E	U	ƒ	ö	ß	Y	±
15	F	£	¢	ü	O	ÿ	

Roman 8

REFLECTION® 1/7

Default Reflection Keystrokes

Reflection Function	PC/AT	Enhanced
Cursor Movement		
* Beginning of line	Home	(cp)Home
* End of line	End	(cp)End
Home down	Ctrl-End	Ctrl-(cp)End
Home up	Ctrl-Home	Ctrl-(cp)Home
Move down	↓	(cp)↓
Move up	↑	(cp)↑
Move left	←	(cp)←
Move right	→	(cp)→
* Next page	PgDn	(cp)PgDn
* Move to top	Ctrl-PgUp	Ctrl-(cp)PgUp
* Move to bottom	Ctrl-PgDn	Ctrl-(cp)PgDn
* Prev page	PgUp	(cp)PgUp
Roll down	Ctrl-↑	Ctrl-(cp)↑
Roll up	Ctrl-↓	Ctrl-(cp)↓
Roll left	Ctrl-→	Ctrl-(cp)→
Roll right	Ctrl-←	Ctrl-(cp)←
Editing Keystrokes		
Clear display	Alt-J	
Clear line	Alt-K	
Delete char	Del	(cp)Del
Delete line	Alt-D	Del/Alt-D
Insert char	Ins	(cp)Ins
Insert line	Alt-I	Ins/Alt-I
Resetting Functions		
* Clear keyboard buffer	Alt-V	
Clear print buffer	Ctrl-F1	
Connection reset	Ctrl-F8	
* Function key recall	Alt-F	
Hard reset	Alt-R	
* Reset handshake	Alt-Q	
* Reset typeahead	Alt-T	
Reset video (R1 only)	Alt-=	
Soft reset	Alt-S	
Menus and Labels		
Command line	Alt-Y	
Config keys	Alt-C	
Define user keys	Ctrl-F9	
Error recap menu	Alt-E	
Help screen	Alt-H	
Modes keys	Alt-M	
System keys	Alt-A or F10	
Toggle user keys	F9	
User-key labels	Alt-U or F9	

* HP only — (cp) = Cursor Pad — (kp) = Keypad

Reflection Function	PC/AT	Enhanced

Special Functions

* Backtab	Shift-Tab	
Break	Ctrl-Scroll-Lock	Ctrl-Break
* Delete character	Ctrl-Backspace	
Delete wraparound	Ctrl-Del	
Disconnect	Ctrl-F10	
Enter	(kp)plus or Shift-F10	(kp)Enter
Extend Key	Alt-Z	
Exit to DOS	Alt-X	
Hot key	Alt-(right) Shift	
Insert wraparound	Ctrl-Ins	
Interrupt command	Ctrl-Y	
Print screen	Alt-(kp)star	PrtSc
Print screen with labels (DOS)	Shift-PrtSc	
Return	Return	
State save	Alt-B	
* Stop **	Alt/Shift-Scroll-Lock	
Toggle print logging	Ctrl-(kp)star	Ctrl-PrtSc

* HP only ** Receive pacing must be XON/XOFF

VT-Only Functions

Backspace	Ctrl-Backspace
Delete character	Backspace
Hold screen	Scroll-Lock or Shift Scroll-Lock
Next page	Ctrl-PgDn
Next session	Alt-N
PF key labels	Alt-P
Prev page	Ctrl-PgUp
ScrollLock	Alt-Scroll-Lock
Send answerback	Ctrl-F2
VT0 to VT9	(kp)0 to (kp)9
VT comma	(kp)star
VT enter	(kp)plus
VT minus	(kp)minus
VT period	(kp)period
VT PF1 (GOLD)	F1
VT PF2	F2
VT PF3	F3
VT PF4	F4

The following key functions are only valid when VT220 is the terminal type:

Find	Alt-1
Insert here	Alt-2
Next screen	Alt-6
Previous screen	Alt-5
Remove	Alt-3
Select	Alt-4
VT F6 to VT F10	Shift-F6 to F10
VT F11 to VT F14	Alt-F1 to F4
VT F15 (help)	Alt-F5
VT F16 (do)	Alt-F6
VT F17 to VT F20	Alt-F7 to F10

DC8I001.01