Terminal Subsystem Manual

Order No. AA-N623B-TK

May 1983

This document describes the Terminal Subsystem of Professional 300 Series Personal Computers. It is a reference manual for programmers writing applications for these computers.

OPERATING SYSTEM AND VERSION:	P/OS
SOFTWARE VERSION:	1.5

DIGITAL EQUIPMENT CORPORATION
Maynard, Massachusetts 01754

1st Printing, December 1982
2nd Printing (Rev), May 1983

CONTENTS

INDEX

FIGURES

TABLES

PREFACE

WHO SHOULD READ THIS MANUAL

You should read this manual if you are developing an application for the Professional computer and need information about the Professional's Terminal Subsystem. The Terminal Subsystem is the subset of the Professional that provides functions typically performed by a separate video display terminal,like the VT100.

SCOPE OF MANUAL

The manual covers two possible types of Professional application environments:

- ☐ Application running on the Professional.
- ☐ Application running on a remote host system while the Professional is being operated in terminal emulation mode. Terminal emulation is available through the optional PRO/Communications application.

In both environments, the Professional processes and displays data either in text mode (character-cell-oriented processing) or graphics mode (pixel-oriented processing), as described in Chapter 1. This manual covers text mode almost exclusively. Graphics mode is described in other documents:

- ☐ If you are using the Professional 300 Series Developer's Tool Kit to develop an application that will run on the Professional, refer to the CORE Graphics Library Manual in the Tool Kit for graphics programming information.
- ☐ If you are developing a PRO/BASIC application on the Professional, refer to the Professional 300 Series PRO/BASIC Language Manual for graphics programming information.

- ☐ If you are developing a Professional application that will run on a remote host system, refer to the VT125 User's Guide for graphics programming information. VT125 emulation, available through the PRO/Communications application, features the Remote Graphics Instruction Set (ReGIS). Use of the ReGIS programming language is documented in the VT125 User Guide. Because there are some differences between a VT125 terminal and the features emulated on the Professional, you should also refer to Appendix D of this manual for a summary of these differences.
- ☐ In any of the above cases, you should also refer to Appendix C of this manual for information regarding mixing text and graphics in your application.

This manual does not cover the Professional's terminal driver or its effect on terminal I/O. Also, this manual does not describe the environment for applications interfacing directly with the Professional's asynchronous communications driver (only the terminal emulation environment is covered). Refer to the P/OS System Reference Manual for information on the two drivers.

ORGANIZATION OF MANUAL

The manual has three chapters and four appendices. The contents are summarized below. Because of many similarities, and to facilitate comparison, the environments for applications running on the Professional and on a remote host are treated simultaneously. Remember, however, that terminal emulation requires the optional PRO/Communications application running on the Professional.

Chapter 1—Introduction to the Terminal Subsystem

Introduces the environment both for applications running on the Professional and for those running on a remote host while the Professional is being operated as a terminal emulator. Defines the various text and graphics modes of operation. Describes character encoding in the Professional, including the DEC Multinational Character Set and how the Professional accommodates 7-bit and 8-bit application environments.

Chapter 2—Codes Transmitted from Keyboard to Application Program

Describes the characters and codes generated by the keyboard keys and transmitted to an application for all text modes of operation.

Chapter 3—Response of Professional to Received Codes

Describes the response of the Professional to received codes in text mode. Describes all control functions you can use in your application for all text modes of operation.

Appendix A—Character Set Summary

Shows the code tables for the Professional character sets, including the octal, decimal, and hexadecimal numerical code conversions.

Appendix B—Keyboard Illustrations

Includes illustrations of all national keyboards available for the Professional.

Appendix C—Text and Graphics Mode Interaction

Offers guidelines for mixing text and graphics in Professional applications.

Appendix D—Differences in Emulated Features

Lists the differences between VT102 and VT125 video terminals and the emulated features on the Professional.

RELATED DOCUMENTATION

The following documents are mentioned in this manual as sources of additional related information.

Professional 300 Series User Documentation

User's Guide for Hard Disk System
User's Guide for Diskette System
PRO/BASIC Language Manual
PRO/Communications Manual

VT102 and VT125 Documentation

VT102 Vides Terminal User Guide
VT125 User Guide

If you are using the Professional 300 Series Developer's Tool Kit to develop an application for the Professional, refer to the Tool Kit Documentation Directory for a description of other Tool Kit documents.

Also refer to the *Tool Kit Installation Guide and Release Notes*. That manual describes any software and/or documentation errors that were discovered or changes that were made late in the development cycle.

CHAPTER 1
INTRODUCTION TO THE TERMINAL SUBSYSTEM

The Professional's Terminal Subsystem displays text and graphics on the monitor screen and returns keyboard input to an application program. This manual describes how you interact with the Terminal Subsystem, through your application program, to control the way data is processed and displayed.

The Terminal Subsystem operates either in text mode or graphics mode, at the request of the application. Text mode is the default. A brief comparison of the two modes is in Section 1.1. Except for that overview and the information in Appendices C and D, this manual covers only the text-mode interactions between your application and the Terminal Subsystem. For example, it shows how to control the way text looks on the screen or how to change character sets with various text-mode commands. Graphics mode interactions are executed through high-level language interfaces (CORE Graphics Library, PRO/BASIC, and ReGIS). Use of the graphics interfaces is documented in the manuals indicated in the Preface.

Several application environments are discussed in this manual. They are defined in Section 1.2. The terminology introduced in that section is used throughout the manual when referring to various modes of operation.

Section 1.3 describes the character encoding scheme used on the Professional when in text mode. The Professional uses an ANSI-compatible 8-bit default character set (the DEC Multinational Character Set) that offers advantages over the traditional 7-bit ASCII code. The DEC Multinational set is a superset of the ASCII set.

Section 1.3 also introduces the use of control functions in your application: control characters, escape sequences, and control sequences. The last subject covered in this chapter is how to work with 7-bit and 8-bit application environments.

1.1 COMPARISON OF TEXT MODE AND GRAPHICS MODE

The Professional's screen display consists of closely spaced picture elements (pixels) arranged as 960 columns and 240 rows. In graphics mode, your application can turn pixels on and off individually or in groups to create a variety of images, subject only to the pixel resolution of the screen. The mapping of software addresses to pixels is determined by the software being used. In graphics mode, a pixel is the smallest addressable unit of the screen.

In text mode, data is processed on a coded character basis, and characters are displayed on the screen within discrete rectangular cells. The screen is divided into 80×24 or 132×24 such character cells, depending on whether the screen is set for 80 or 132 columns. Each cell consists of numerous pixels. Your application puts a character on the screen by telling the Professional to address a cell and display a character within that cell. The character is encoded according to the character set mapping the Professional is using at the time, for example ASCII coding. In text mode, a character cell is the smallest addressable unit of the screen.

Similar considerations apply to printing in text or graphics mode. In text mode, a serial printer attached to the Professional prints specified character cells. In graphics mode, the printer prints specified pixels.

In text mode, your application can interact directly with the Terminal Subsystem to address individual character cells. In graphics mode, the CORE Graphics Library, PRO/BASIC, or ReGIS are used to interact with the Terminal Subsystem. These interfaces minimize the tedious coding that would otherwise be necessary to control individual pixels. They also provide various degrees of hardware independence, depending on the coding standards implemented.

Each mode of operation, text or graphics, has advantages and limitations. In text mode, the screen can be written to faster than in graphics mode because there are only 1920 (80×24) or 3168 (132×24) character cell addresses in text mode. By contrast, there are 230,400 (960×240) pixels in graphics mode (the actual number of addresses to be controlled depends on the mapping of software addresses to pixels). The same speed considerations apply to printing in text or graphics mode.

Your application can create a limited selection of images while still operating in the faster, text mode. The Professional supports the DEC Special Graphics Character Set (described in Section 1.3). In that character set, various small line elements (straight line, corners) are encoded as individual characters. You create images by connecting characters together.

Although slower than text mode, graphics mode on the Professional lets you create a much greater variety of images on the screen or printer, subject only to the pixel resolution. In particular you can create text in graphics mode. In principle, you have an unlimited range of character sizes, styles, and orientations when working with text in graphics mode. Graphics-mode text is handled like any other type of graphics and offers the same advantages and disadvantages when compared to text-mode text.

You can mix text and graphics in your application. However, to do it properly you need a detailed understanding of graphics and text modes. Appendix C offers some guidelines.

If the extended bitmap option (EBO) is not installed on the Professional, your application has one bitmap plane available (a bitmap plane can be addressed on a pixel basis by your application). The application can either write to the plane in text mode or use it in graphics mode at any point in the program. Each pixel can have only two possible states.

If the EBO is installed, your application has two additional bitmap planes. When in text mode, the application still always writes to a single plane. However, in graphics mode, the application can use the three planes in a variety of ways. The exact possibilities depend on the graphics language you are using.

When your application uses two planes in graphics mode, a pixel can have one of four (2^2) possible states. This gives up to four shades of "gray" with the monochrome display monitor. With a color monitor, the two planes let you have up to four colors from a palette of 256 colors.

When your application uses all three planes in graphics mode, a pixel can have one of eight (2^3) possible states. This gives up to eight shades of "gray" with the monochrome monitor. With a color monitor, the three planes let you have up to eight colors from a palette of 256 colors.

An application using the CORE Graphics Library can use any one plane, or any pairs of planes, or all three planes in graphics mode.

When in graphics mode, a PRO/BASIC application always uses all the available planes. You cannot use one plane in text mode and use some other plane(s) in graphics mode.

VT125 terminal emulation is possible on the Professional through the PRO/Communications application. A VT125 terminal is a VT100 (text-mode) terminal physically combined with a bitmap addressable graphics terminal. Yet each is functionally independent. There are three bitmap planes. One is used only for text mode and the other two only for graphics mode. Their contents are combined only on the monitor screen.

VT125 (terminal emulation) mode on the Professional requires the EBO, for compatibility with a VT125 terminal. When in text mode, your application always writes to one plane. By default, for compatibility with a VT125 terminal, your application uses only the other two planes when in graphics mode. However, if you wish, the Professional also lets your application use all three planes in graphics mode to obtain more degrees of brightness or more colors. In addition, your application can use any one plane, or any pairs of planes, or all three planes in graphics mode.

1.2 ENVIRONMENTS FOR PROFESSIONAL APPLICATIONS

Several application environments are discussed in this manual. They are defined in this section as modes of operation for the Professional (or, equally, for an application). Because of many similarities, these modes are not treated individually in separate chapters. Rather, the information is organized and tabulated so you can compare the functions of all modes readily in each chapter. Some of the mode terminology is arbitrary and is used only to simplify the presentation of information (for example native mode, ANSI mode). Other mode names refer to terminal emulation (VT102 mode, Professional mode, VT125 mode, VT52 mode). Those names appear on the PRO/Communications operator menu and are used consistently in this manual.

Professional applications include those running on the Professional, in "native" mode, and those running on a remote host system when the Professional is being used as a terminal emulator.

You will note that various default states of operation are mentioned throughout this manual. Chapter 3 describes how to control the programmable states. Some of these can also be controlled manually by the Professional computer's operator through "Set-Up" menus. Although these menus are mentioned in the discussion of programmable states, you should refer to the user manuals for complete operating instructions. The P/OS Set-Up menus, which affect applications running in terminal emulation modes as well as native mode, are described in the *User's Guide for (Diskette/Hard Disk) System*. If you are writing an application for terminal emulation, you should also refer to the *PRO/Communications Manual* for details on the various PRO/Communications Set-Up menus. Chapter 3 contains a full comparison listing of the Set-Up characteristics and (programmable) mode selection sequences.

1.2.1 Native Mode

Native mode is the environment for an application running on the Professional. Applications running in native mode include

- ☐ applications developed with the Tool Kit,
- ☐ PRO/BASIC applications (developed using PRO/BASIC), and
- ☐ optional Digital-supplied applications, such as PRO/Communications and PRO/BASIC.

Native (text) mode is ANSI compatible. For graphics-mode interactions, an application uses either the CORE Graphics Library or PRO/BASIC graphics statements in native mode.

The native mode environment is illustrated in Figure 1–1. The application interacts with the Terminal Subsystem by way of the terminal driver. The terminal driver is described in detail in the *P/OS System Reference Manual*. It handles the queuing and de-queuing of requests from the application to the Terminal Subsystem. It also screens certain control characters under various conditions specified by the software.

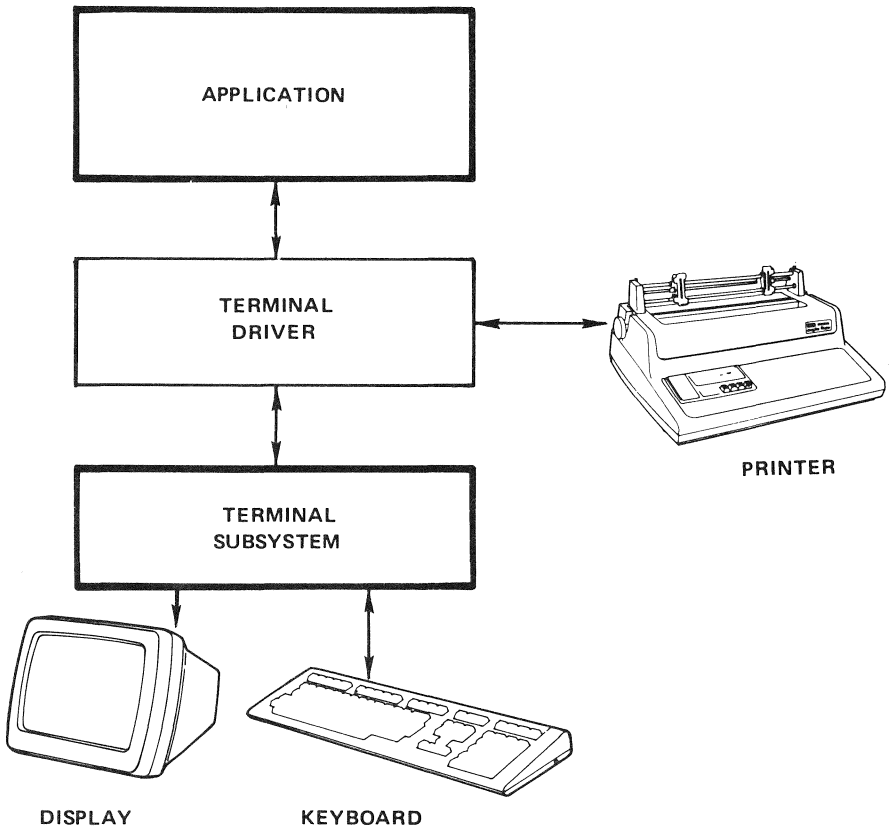


Figure 1-1
Application Environment for Native Mode

1.2.2 Terminal Emulation Modes

Figure 1-2 shows the application environment for terminal emulation. The PRO/Communications application runs on the Professional in native mode. The host application communicates with the Terminal Subsystem through the PRO/Communications terminal emulator.

A PRO/Communications menu lets the Professional computer operator choose one of four possible terminal emulation modes:

- ☐ VT102 mode
- ☐ Professional mode
- ☐ VT52 mode
- ☐ VT125 mode

Selecting one of these modes causes the Professional to behave like a terminal of that type (Professional mode is defined later in this section). The Professional communicates with a host application or host system as if it were a terminal of that type.

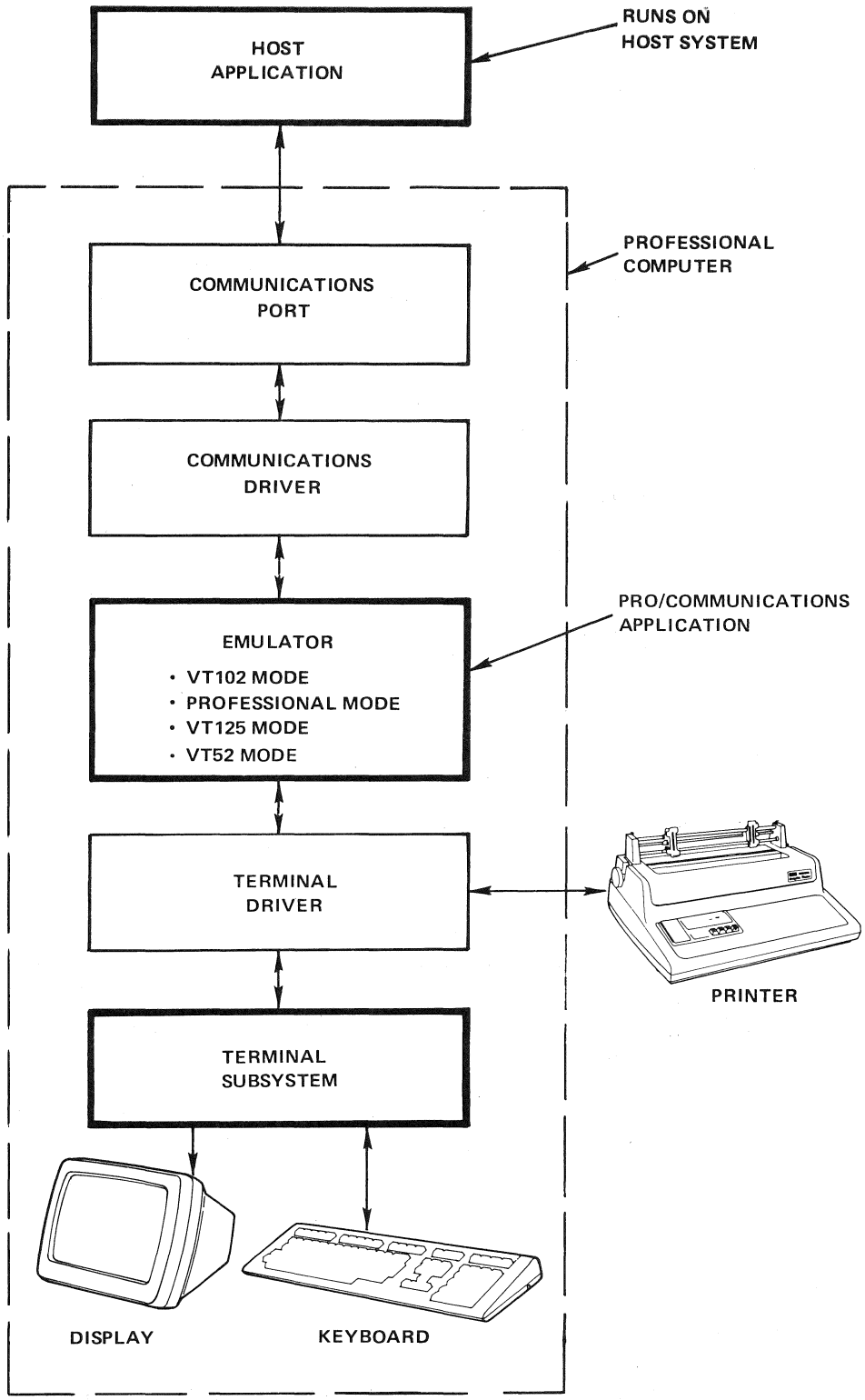


Figure 1-2
Application Environment for Terminal Emulation Modes

By using the above menu selections, the operator ensures that the Professional will not only behave like one of the four terminals but will also identify itself as one to any host application or system requesting a device identifier (the identifying responses are detailed in Chapter 3).

Complete operating instructions on the emulator, including Set-Up features and default conditions are documented in the *PRO/Communications Manual*.

Noteworthy features that are at the operator's control through the menu system include the following:

- ☐ The operator can select a default terminal type.
- ☐ The operator can reset the emulator. Resetting the emulator causes an RIS (Reset to Initial State) command to be issued which resets various programmable and/or manually controllable Set-Up states, as described in Chapter 3.
- ☐ For compatibility with earlier and newer software environments, the operator can choose to enable the transmission/reception of either 7-bit characters (only) or both 7- and 8-bit characters over the communications line (see Section 1.3.5).

An overview of additional key information specific to each emulation mode follows.

1.2.2.1 VT102 Mode—When operating the Professional in VT102 mode, you can communicate with the host system and the application as if you were at a VT102 terminal. Like a VT102 terminal, VT102 mode is ANSI compatible in normal operation and also includes VT52 emulation as an ANSI-incompatible subset that is available through program control.

There are some functional differences between a VT102 terminal and VT102 mode on the Professional. They are pointed out throughout this manual and listed in Appendix D.

1.2.2.2 Professional Mode—Professional mode is a new ANSI-compatible terminal emulation text mode. When operating the Professional in Professional mode, you can communicate with the host system and the application as if you were at a "Professional" terminal. Professional mode does not emulate any existing terminal. It is a superset of VT102 mode and includes all the functions available through VT102 mode. In addition, keys on the Professional keyboard that are not on a VT102 terminal are available to the application.

1.2.2.3 VT52 Mode—VT52 mode is an ANSI-incompatible text mode and is offered on the Professional only for compatibility with earlier software.

VT52 mode is implemented on the Professional as a subset of VT102 mode: when operating the Professional in VT52 mode, you can communicate with the host system and the application as if you were at a VT102 terminal emulating a VT52 terminal.

VT52 mode can be selected through program control as well as through a PRO/Communications menu selection. Chapter 3 describes how to invoke VT52 mode in a program and how to use the VT52-compatible commands.

1.2.2.4 VT125 Mode

> Note: VT125 emulation requires that the extended bitmap option (EBO) be installed on the Professional. If the EBO is not installed, the response of the Professional to a Device Attributes (DA) request is the same as for VT102 mode (see Chapter 3).

VT125 mode emulates a VT125 terminal, which can operate in both text mode and graphics mode. A VT125 terminal is a VT100 (text-mode) terminal combined with a ReGIS interpreter and a bitmap addressing capability.

ReGIS is the Digital Remote Graphics Instruction Set. You use ReGIS in a program to create and store images as ASCII text, called ReGIS strings. The program, running on a host system, can then send them to a remote VT125 terminal for execution. Prior to processing a ReGIS string, the VT125 terminal operates in text mode. Receipt of a command introducing one or more ReGIS strings automatically switches the VT125 to graphics mode so it can process and display ReGIS graphics. After the ReGIS code is executed, a string terminator switches the VT125 back to text mode.

These VT125 functions are emulated on the Professional in VT125 mode. The commands you use to enter and exit VT125 graphics mode and the default states are documented in Chapter 3. Complete programming information on ReGIS is in the *VT125 User's Guide*.

There are some significant functional differences between VT125 graphics mode on the Professional and using ReGIS on a VT125 terminal. You should refer to Appendix D for a summary of these differences and their impact on software compatibility.

1.3 TEXT MODE

This section describes the character encoding used on the Professional when operating in text mode.

1.3.1 Coding Standards

The Professional uses an 8-bit character encoding scheme and 7-bit code extension techniques that are compatible with the following ANSI and ISO standards. ANSI (American National Standards Institute) and ISO (International Organization for Standardization) specify the current standards for character encoding used in the communications industry.

ANSI X3.4-1977

American Code for Information Interchange (ASCII)

ISO 646-1977

7-Bit Coded Character Set for Information Processing Interchange

ANSI X3.41-1974

Code Extension Techniques for Use with the 7-Bit Coded Character Set of American National Code for Information Interchange

ISO Draft International Standard 2022.2

7-Bit and 8-Bit Coded Character Sets—Code Extension Techniques

ANSI X3.32-1973

Graphic Representation of the Control Characters of American National Code for Information Interchange

ANSI X3.64-1979

Additional Controls for Use with American National Standard for Information Interchange

ISO Draft International Standard 6429.2

Additional Control Functions for Character Imaging Devices

ISO Draft International Standard 6937

Coded Character Sets for Text Communication

The default character set on the Professional is the 8-bit DEC Multinational Character Set. It is compatible with the traditional 7-bit ASCII character encoding scheme: the ASCII character set is a subset of DEC Multinational.

The Professional is designed to accommodate 7-bit software, when necessary, in all text modes. By default, the Professional will transmit 7-bit code extensions to an application instead of 8-bit control characters. But you can make the Professional transmit 8-bit control characters if your application environment can take advantage of the faster processing possible with 8-bit codes. Sections 1.3.4 and 1.3.5 explain control functions and how to work with 7- and 8-bit codes.

1.3.2 Code Table

A code table is a convenient way to represent 7- and 8-bit characters, because groupings of characters and their relative codes can be seen readily.

> Note: The familiar ASCII 7-bit character set is used below to introduce some terminology and coding conventions. The 8-bit coding scheme used on the Professional is described after this introduction.

1.3.2.1 7-Bit Code Table—Table 1-1 is the 7-bit ASCII code table. There are 128 (2^7) positions, corresponding to 128 character codes, arranged in a matrix of 8 columns and 16 rows.

Each row represents a possible value of the 4 least significant bits of a 7-bit code. Each column represents a possible value of the 3 most significant bits.

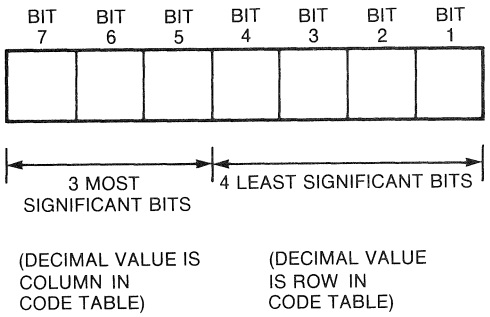


Table 1-1 shows the octal, decimal, and hexadecimal code for each ASCII character. You can also represent any character by its position in the table. For example, the character "H" (column 4, row 8) can be represented as 4/8. The column/row representation is like hexadecimal notation. For example, the character "n" (6/14) is 6E in hex code.

> Note: The column/row notation is used to represent characters and codes throughout this manual. Appendix A lists all equivalent codes in various notations.

The Professional processes received characters based on two character types defined by ANSI: graphic characters and control characters.

- *Graphic characters* are the characters you write, print, or display on a video screen. The ASCII graphic characters are in positions 2/1 through 7/14 of Table 1-1. They include all American and English alphanumeric characters, plus punctuation marks and various text symbols. Examples are: C, n, ", !, +, $ (the English pound sign is not an ASCII graphic character).

Table 1-1
7-Bit ASCII Code Table

ROW	BITS b7 b6 b5 b4 b3 b2 b1	0		1		2		3		4		5		6		7	
	COLUMN	0 0 0		0 0 1		0 1 0		0 1 1		1 0 0		1 0 1		1 1 0		1 1 1	
0	0 0 0 0	NUL	0 0 0	DLE	20 16 10	SP	40 32 20	0	60 48 30	@	100 64 40	P	120 80 50	`	140 96 60	p	160 112 70
1	0 0 0 1	SOH	1 1 1	DC1 (XON)	21 17 11	!	41 33 21	1	61 49 31	A	101 65 41	Q	121 81 51	a	141 97 61	q	161 113 71
2	0 0 1 0	STX	2 2 2	DC2	22 18 12	"	42 34 22	2	62 50 32	B	102 66 42	R	122 82 52	b	142 98 62	r	162 114 72
3	0 0 1 1	ETX	3 3 3	DC3 (XOFF)	23 19 13	#	43 35 23	3	63 51 33	C	103 67 43	S	123 83 53	c	143 99 63	s	163 115 73
4	0 1 0 0	EOT	4 4 4	DC4	24 20 14	$	44 36 24	4	64 52 34	D	104 68 44	T	124 84 54	d	144 100 64	t	164 116 74
5	0 1 0 1	ENQ	5 5 5	NAK	25 21 15	%	45 37 25	5	65 53 35	E	105 69 45	U	125 85 55	e	145 101 65	u	165 117 75
6	0 1 1 0	ACK	6 6 6	SYN	26 22 16	&	46 38 26	6	66 54 36	F	106 70 46	V	126 86 56	f	146 102 66	v	166 118 76
7	0 1 1 1	BEL	7 7 7	ETB	27 23 17	'	47 39 27	7	67 55 37	G	107 71 47	W	127 87 57	g	147 103 67	w	167 119 77
8	1 0 0 0	BS	10 8 8	CAN	30 24 18	(	50 40 28	8	70 56 38	H	110 72 48	X	130 88 58	h	150 104 68	x	170 120 78
9	1 0 0 1	HT	11 9 9	EM	31 25 19	)	51 41 29	9	71 57 39	I	111 73 49	Y	131 89 59	i	151 105 69	y	171 121 79
10	1 0 1 0	LF	12 10 A	SUB	32 26 1A	*	52 42 2A	:	72 58 3A	J	112 74 4A	Z	132 90 5A	j	152 106 6A	z	172 122 7A
11	1 0 1 1	VT	13 11 B	ESC	33 27 1B	+	53 43 2B	;	73 59 3B	K	113 75 4B	[	133 91 5B	k	153 107 6B	{	173 123 7B
12	1 1 0 0	FF	14 12 C	FS	34 28 1C	,	54 44 2C	<	74 60 3C	L	114 76 4C	\	134 92 5C	l	154 108 6C	|	174 124 7C
13	1 1 0 1	CR	15 13 D	GS	35 29 1D	-	55 45 2D	=	75 61 3D	M	115 77 4D	]	135 93 5D	m	155 109 6D	}	175 125 7D
14	1 1 1 0	SO	16 14 E	RS	36 30 1E	.	56 46 2E	>	76 62 3E	N	116 78 4E	^	136 94 5E	n	156 110 6E	~	176 126 7E
15	1 1 1 1	SI	17 15 F	US	37 31 1F	/	57 47 2F	?	77 63 3F	O	117 79 4F	_	137 95 5F	o	157 111 6F	DEL	177 127 7F

KEY

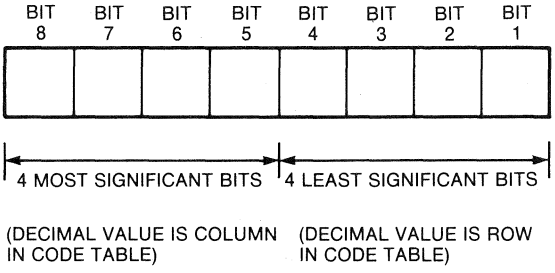
CHARACTER	ESC	33	OCTAL
		27	DECIMAL
		1B	HEX

- *Control characters* are non-display single-byte codes that perform traditional functions in data communications and text processing. The ASCII control characters are in positions 0/0 through 1/15 (columns 0 and 1) of Table 1–1. The SP character ("space", 2/0) can be considered either a graphic character or a control character depending on the context. DEL (7/15) is always used as a control character.

  The codes and functions of control characters have been standardized by ANSI. Examples of ASCII control characters, with their ANSI-standard mnemonics, are: CR (carriage return), FF (form feed), CAN (cancel). All standard ANSI definitions for control characters are summarized in Appendix A.

1.3.2.2 8-Bit Code Table —The above conventions can be generalized to the 8-bit character encoding used on the Professional. Table 1–2 shows the 8-bit code table. It has twice as many columns as the 7-bit table, because it contains 256 (2^8) versus 128 (2^7) code values.

As with the 7-bit table, each row represents a possible value of the 4 least significant bits of an 8-bit code. Each column represents a possible value of the 4 most significant bits.

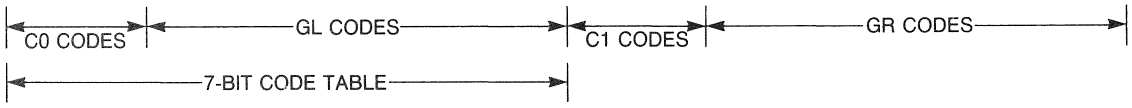


All codes on the left half of the 8-bit table (columns 0 through 7) are 7-bit compatible: their 8th bit is not set and can be ignored or assumed to be 0. You can use these codes in either a 7-bit or an 8-bit environment. All codes on the right half of the table (columns 8 through 15) have their 8th bit set. You can use these codes only in an 8-bit compatible environment.

The 8-bit code table has two sets of control characters (C0, "control zero", and C1, "control one") and two sets of graphic characters (GL, "graphic left" and GR, "graphic right").

Table 1–2
8-Bit Code Table

COLUMN / ROW	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
00	NUL	DLE	SP							DCS	///					
01	SOH	DC1								PU1						
02	STX	DC2								PU2						
03	ETX	DC3								STS						
04	EOT	DC4							IND	CCH						
05	ENQ	NAK							NEL	MW						
06	ACK	SYN							SSA	SPA						
07	BEL	ETB							ESA	EPA						
08	BS	CAN							HTS							
09	HT	EM							HTJ							
10	LF	SUB							VTS							
11	VT	ESC							PLD	CSI						
12	FF	FS							PLU	ST						
13	CR	GS							RI	OSC						
14	SO	RS							SS2	PM						
15	SI	US						DEL	SS3	APC						///



On the Professional, the basic functions of the C0 and C1 codes are as defined by ANSI. C0 codes represent the ASCII control characters described earlier. The C0 codes are 7-bit compatible. The C1 codes represent 8-bit control characters that let you perform additional functions beyond those possible with the C0 codes. C1 codes can be used only in an 8-bit environment. Some C1 code positions have been left blank: their functions have not yet been standardized. The standard ANSI definitions for the C0 and C1 characters are summarized in Appendix A.

> Note: Not all C0 and C1 codes are recognized by the Professional. Chapter 3 identifies the ones that are. The others are simply ignored by the Professional (no action taken).

The GL and GR sets of codes are reserved for graphic characters. There are 94 GL codes in positions 2/1 through 7/14 and 94 GR codes in positions 10/1 through 15/14. By ANSI standards, positions 10/0 and 15/15 are reserved. You can use GL codes in 7-bit or 8-bit environments. You can use GR codes only in an 8-bit environment.

1.3.3 The Professional Character Sets

You can't change the functions of the C0 or C1 codes on the Professional. However, you can "map" different sets of graphic characters onto the GL and/or GR codes. The sets are stored "on call" within the Professional as a sort of graphic repertoire. But, they are not available for use until mapped onto the GL or GR codes. The commands for mapping graphic character sets onto GL or GR are described in Chapter 3.

For this release, the Professional's graphic repertoire consists of three 94-character sets. Their relationship is described in the next two sections:

- ☐ ASCII Graphics Set
- ☐ DEC Supplemental Graphics Set
- ☐ DEC Special Graphics Set, also known as the "VT100 Line Drawing Set".

1.3.3.1 DEC Multinational Character Set—By default, when you power up (boot) the Professional or reset the PRO/Communications terminal emulator, the DEC Multinational Character Set is mapped onto the 8-bit code matrix. The DEC Multinational Character Set is shown on Table 1–3.

The 7-bit-compatible left half of the DEC Multinational Set is the ASCII character set: The C0 codes are the ASCII control characters, and the GL codes are mapped onto the ASCII Graphics Set.

The 8-bit compatible right half of the DEC Multinational Set includes the C1 8-bit control characters in columns 8 and 9. The GR codes are mapped onto the DEC Supplemental Graphics Set. This set has alphabetic characters with diacritical marks that appear in the major Western European alphabets. It also has other symbols not included in the ASCII Graphics Set.

The Professional supports over a dozen national (Western European) keyboards. Use of all Professional keyboards assumes the default DEC Multinational Character Set mapping. In addition, the code descriptions in the rest of this manual assume that default mapping.

Various characters from the DEC Supplemental Graphics Set appear as standard (printing character) keys on different keyboards. By supplementing the standard keys with "compose sequences", described in Chapter 2, you can create any DEC Multinational graphic character from any Professional keyboard.

1.3.3.2 Other Character Sets —In addition to ASCII and DEC Supplemental, the Professional's graphic repertoire has the DEC Special Graphics Set, also known as the "VT100 Line Drawing" Character Set. It is shown in Table 1–4. This set has about two thirds of the ASCII graphic characters. In addition it has special symbols and short line segments. The line segments let you create a limited range of pictures while still using text mode.

Note that the following characters appear in both the DEC Supplemental and DEC Special Graphics sets but under different codes.

	Supplemental Graphics	*Special Graphics*
Pound sign (£)	10/3	7/13
Degree sign (°)	11/0	6/6
Plus/Minus sign (±)	11/1	6/7

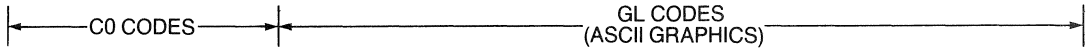
The DEC Supplemental Graphics middle dot (11/7) is a different character from the DEC Special Graphics raised dot (7/14). The former is typically used to separate numeric digits. The latter is used with line drawing segments in the DEC Special Graphics Set.

Commands described in Chapter 3 let you map the DEC Special Graphics Set into either GL or GR, replacing either the ASCII Graphics Set or the DEC Supplemental Graphics Set. The recommended mapping is to switch between ASCII and DEC Special Graphics in GL, since the latter has most of the ASCII graphic characters. Also, this mapping is compatible with a VT102 terminal.

Other character sets are planned for the Professional. User loadable character sets are not supported in text mode for this release. Graphics mode does, however, support user loadable character sets.

Table 1-3
DEC Multinational Character Set (C0 and GL Codes)

ROW	b4 b3 b2 b1	0	1	2	3	4	5	6	7
0	0 0 0 0	NUL 0 0 0	DLE 20 16 10	SP 40 32 20	0 60 48 30	@ 100 64 40	P 120 80 50	` 140 96 60	p 160 112 70
1	0 0 0 1	SOH 1 1 1	DC1 (XON) 21 17 11	! 41 33 21	1 61 49 31	A 101 65 41	Q 121 81 51	a 141 97 61	q 161 113 71
2	0 0 1 0	STX 2 2 2	DC2 22 18 12	" 42 34 22	2 62 50 32	B 102 66 42	R 122 82 52	b 142 98 62	r 162 114 72
3	0 0 1 1	ETX 3 3 3	DC3 (XOFF) 23 19 13	# 43 35 23	3 63 51 33	C 103 67 43	S 123 83 53	c 143 99 63	s 163 115 73
4	0 1 0 0	EOT 4 4 4	DC4 24 20 14	$ 44 36 24	4 64 52 34	D 104 68 44	T 124 84 54	d 144 100 64	t 164 116 74
5	0 1 0 1	ENQ 5 5 5	NAK 25 21 15	% 45 37 25	5 65 53 35	E 105 69 45	U 125 85 55	e 145 101 65	u 165 117 75
6	0 1 1 0	ACK 6 6 6	SYN 26 22 16	& 46 38 26	6 66 54 36	F 106 70 46	V 126 86 56	f 146 102 66	v 166 118 76
7	0 1 1 1	BEL 7 7 7	ETB 27 23 17	' 47 39 27	7 67 55 37	G 107 71 47	W 127 87 57	g 147 103 67	w 167 119 77
8	1 0 0 0	BS 10 8 8	CAN 30 24 18	(50 40 28	8 70 56 38	H 110 72 48	X 130 88 58	h 150 104 68	x 170 120 78
9	1 0 0 1	HT 11 9 9	EM 31 25 19	) 51 41 29	9 71 57 39	I 111 73 49	Y 131 89 59	i 151 105 69	y 171 121 79
10	1 0 1 0	LF 12 10 A	SUB 32 26 1A	* 52 42 2A	: 72 58 3A	J 112 74 4A	Z 132 90 5A	j 152 106 6A	z 172 122 7A
11	1 0 1 1	VT 13 11 B	ESC 33 27 1B	+ 53 43 2B	; 73 59 3B	K 113 75 4B	[133 91 5B	k 153 107 6B	{ 173 123 7B
12	1 1 0 0	FF 14 12 C	FS 34 28 1C	, 54 44 2C	< 74 60 3C	L 114 76 4C	\ 134 92 5C	l 154 108 6C	| 174 124 7C
13	1 1 0 1	CR 15 13 D	GS 35 29 1D	- 55 45 2D	= 75 61 3D	M 115 77 4D	] 135 93 5D	m 155 109 6D	} 175 125 7D
14	1 1 1 0	SO 16 14 E	RS 36 30 1E	. 56 46 2E	> 76 62 3E	N 116 78 4E	^ 136 94 5E	n 156 110 6E	~ 176 126 7E
15	1 1 1 1	SI 17 15 F	US 37 31 1F	/ 57 47 2F	? 77 63 3F	O 117 79 4F	_ 137 95 5F	o 157 111 6F	DEL 177 127 7F



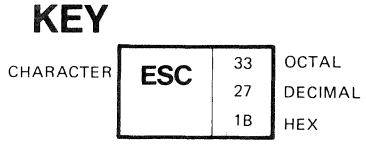


Table 1-3
DEC Multinational Character Set (C1 and GR Codes)

8		9		10		11		12		13		14		15		COLUMN	
1 0 0 0		1 0 0 1		1 0 1 0		1 0 1 1		1 1 0 0		1 1 0 1		1 1 1 0		1 1 1 1		b8 b7 BITS b6 b5 b4 b3 b2 b1	ROW
	200 128 80	DCS	220 144 90		240 160 A0	°	260 176 B0	À	300 192 C0		320 208 D0	à	340 224 E0		360 240 F0	0 0 0 0	0
	201 129 81	PU1	221 145 91	¡	241 161 A1	±	261 177 B1	Á	301 193 C1	Ñ	321 209 D1	á	341 225 E1	ñ	361 241 F1	0 0 0 1	1
	202 130 82	PU2	222 146 92	¢	242 162 A2	2	262 178 B2	Â	302 194 C2	Ò	322 210 D2	â	342 226 E2	ò	362 242 F2	0 0 1 0	2
	203 131 83	STS	223 147 93	£	243 163 A3	3	263 179 B3	Ã	303 195 C3	Ó	323 211 D3	ã	343 227 E3	ó	363 243 F3	0 0 1 1	3
IND	204 132 84	CCH	224 148 94		244 164 A4		264 180 B4	Ä	304 196 C4	Ô	324 212 D4	ä	344 228 E4	ô	364 244 F4	0 1 0 0	4
NEL	205 133 85	MW	225 149 95	¥	245 165 A5	µ	265 181 B5	Å	305 197 C5	Õ	325 213 D5	å	345 229 E5	õ	365 245 F5	0 1 0 1	5
SSA	206 134 86	SPA	226 150 96		246 166 A6	¶	266 182 B6	Æ	306 198 C6	Ö	326 214 D6	æ	346 230 E6	ö	366 246 F6	0 1 1 0	6
ESA	207 135 87	EPA	227 151 97	§	247 167 A7		267 183 B7	Ç	307 199 C7	Œ	327 215 D7	ç	347 231 E7	œ	367 247 F7	0 1 1 1	7
HTS	210 136 88		230 152 98	¤	250 168 A8		270 184 B8	È	310 200 C8	Ø	330 216 D8	è	350 232 E8	ø	370 248 F8	1 0 0 0	8
HTJ	211 137 89		231 153 99	©	251 169 A9	1	271 185 B9	É	311 201 C9	Ù	331 217 D9	é	351 233 E9	ù	371 249 F9	1 0 0 1	9
VTS	212 138 8A		232 154 9A	a	252 170 AA	o	272 186 BA	Ê	312 202 CA	Ú	332 218 DA	ê	352 234 EA	ú	372 250 FA	1 0 1 0	10
PLD	213 139 8B	CSI	233 155 9B	«	253 171 AB	»	273 187 BB	Ë	313 203 CB	Û	333 219 DB	ë	353 235 EB	û	373 251 FB	1 0 1 1	11
PLU	214 140 8C	ST	234 156 9C		254 172 AC	¼	274 188 BC	Ì	314 204 CC	Ü	334 220 DC	ì	354 236 EC	ü	374 252 FC	1 1 0 0	12
RI	215 141 8D	OSC	235 157 9D		255 173 AD	½	275 189 BD	Í	315 205 CD	Ý	335 221 DD	í	355 237 ED	ý	375 253 FD	1 1 0 1	13
SS2	216 142 8E	PM	236 158 9E		256 174 AE		276 190 BE	Î	316 206 CE		336 222 DE	î	356 238 EE		376 254 FE	1 1 1 0	14
SS3	217 143 8F	APC	237 159 9F		257 175 AF	¿	277 191 BF	Ï	317 207 CF	ß	337 223 DF	ï	357 239 EF		377 255 FF	1 1 1 1	15

C1 CODES

GR CODES
(DEC SUPPLEMENTAL GRAPHICS)

Table 1-4
DEC Special Graphics Set Mapped into GL

ROW	B4 B3 B2 B1	0 (B7 B6 B5 = 0 0 0)	1 (0 0 1)	2 (0 1 0)	3 (0 1 1)	4 (1 0 0)	5 (1 0 1)	6 (1 1 0)	7 (1 1 1)
0	0 0 0 0	NUL 0 0 0	DLE 20 16 10	SP 40 32 20	0 60 48 30	@ 100 64 40	P 120 80 50	♦ 140 96 60	— 160 112 70 SCAN 3
1	0 0 0 1	SOH 1 1 1	DC1 (XON) 21 17 11	! 41 33 21	1 61 49 31	A 101 65 41	Q 121 81 51	▒ 141 97 61	— 161 113 71 SCAN 5
2	0 0 1 0	STX 2 2 2	DC2 22 18 12	" 42 34 22	2 62 50 32	B 102 66 42	R 122 82 52	␉ 142 98 62	— 162 114 72 SCAN 7
3	0 0 1 1	ETX 3 3 3	DC3 (XOFF) 23 19 13	# 43 35 23	3 63 51 33	C 103 67 43	S 123 83 53	␌ 143 99 63	¯ 163 115 73 SCAN 9
4	0 1 0 0	EOT 4 4 4	DC4 24 20 14	$ 44 36 24	4 64 52 34	D 104 68 44	T 124 84 54	␍ 144 100 64	├ 164 116 74
5	0 1 0 1	ENQ 5 5 5	NAK 25 21 15	% 45 37 25	5 65 53 35	E 105 69 45	U 125 85 55	␊ 145 101 65	┤ 165 117 75
6	0 1 1 0	ACK 6 6 6	SYN 26 22 16	& 46 38 26	6 66 54 36	F 106 70 46	V 126 86 56	° 146 102 66	┴ 166 118 76
7	0 1 1 1	BEL 7 7 7	ETB 27 23 17	' 47 39 27	7 67 55 37	G 107 71 47	W 127 87 57	± 147 103 67	┬ 167 119 77
8	1 0 0 0	BS 10 8 8	CAN 30 24 18	(50 40 28	8 70 56 38	H 110 72 48	X 130 88 58	␤ 150 104 68	│ 170 120 78
9	1 0 0 1	HT 11 9 9	EM 31 25 19	) 51 41 29	9 71 57 39	I 111 73 49	Y 131 89 59	␋ 151 105 69	≤ 171 121 79
10	1 0 1 0	LF 12 10 A	SUB 32 26 1A	* 52 42 2A	: 72 58 3A	J 112 74 4A	Z 132 90 5A	┘ 152 106 6A	≥ 172 122 7A
11	1 0 1 1	VT 13 11 B	ESC 33 27 1B	+ 53 43 2B	; 73 59 3B	K 113 75 4B	[133 91 5B	┐ 153 107 6B	π 173 123 7B
12	1 1 0 0	FF 14 12 C	FS 34 28 1C	, 54 44 2C	< 74 60 3C	L 114 76 4C	\ 134 92 5C	┌ 154 108 6C	≠ 174 124 7C
13	1 1 0 1	CR 15 13 D	GS 35 29 1D	- 55 45 2D	= 75 61 3D	M 115 77 4D	] 135 93 5D	└ 155 109 6D	£ 175 125 7D
14	1 1 1 0	SO 16 14 E	RS 36 30 1E	. 56 46 2E	> 76 62 3E	N 116 78 4E	^ 136 94 5E	┼ 156 110 6E	· 176 126 7E
15	1 1 1 1	SI 17 15 F	US 37 31 1F	/ 57 47 2F	? 77 63 3F	O 117 79 4F	(BLANK) 137 95 5F	¯ 157 111 6F SCAN 1	DEL 177 127 7F

C0 CODES

GL CODES (DEC SPECIAL GRAPHICS)

KEY

CHARACTER: ESC | 33 OCTAL | 27 DECIMAL | 1B HEX

1.3.4 Control Functions

You use control functions in your application to specify how the Professional should handle data. There are many uses for control functions, and more are being created. Here are some of the things you can do with control functions:

- ☐ Change the monitor display from 80-column to 132-column format.
- ☐ Move the cursor on the display.
- ☐ Delete a line of text from the display.
- ☐ Change graphic character sets.
- ☐ Make the printer print in text mode (or graphics mode),
- ☐ Make the auxiliary keypad keys available for specific functions that you define within your application.
- ☐ Make the Professional operate in 7-bit mode or 8-bit mode, to accommodate different application environments.
- ☐ Ask the Professional to identify itself to an application running on a host system, including whether an option like the color monitor or the extended bitmap option is being used.
- ☐ Switch the Professional to VT125 (graphics) mode from VT102 (text) mode.
- ☐ Switch the Professional to VT52 mode from VT102 or Professional default modes.

You use all control functions in text mode and express them as single-byte or multi-byte codes.

The single-byte codes are the C0 and C1 control characters introduced earlier in Section 1.2. Your application can perform a very limited number of functions using the C0 characters. C1 characters give you a few more functions, but your application can use them only in an 8-bit environment.

Multi-byte control codes represent far more functions because of the variety of code combinations possible. These codes are called *escape sequences* and *control sequences*. Some sequences are ANSI standardized and used throughout the industry. Others are "private" sequences created by manufacturers like Digital for specific families of products. Private sequences, like the ANSI standardized sequences, obey ANSI standards governing the composition of character codes used.

1.3.4.1 Escape Sequences —An escape sequence is a sequence of one or more ASCII graphic characters preceded by the C0 character ESC (1/11). For example,

1/11	2/3	3/7
ESC	#	7

is an escape sequence that causes a printer attached to the Professional to print the contents of the current screen display.

Because escape sequences use only 7-bit characters, you can use them in 7-bit or 8-bit environments.

> Note: When using escape or control sequences, remember that it is the code that defines a sequence, not the graphic representation of the characters. The characters are shown for readability only and presume the DEC Multinational Character Set mapping (ASCII Graphics Set in GL and DEC Supplemental Graphics Set in GR).

There is an important special use of escape sequences on the Professional. ANSI permits code extension techniques to extend the functionality of 7-bit control functions. In particular, you can use 2-byte escape sequences as 7-bit code extensions to express each of the C1 control codes. This is valuable when your application must be compatible with a 7-bit environment. For example, the C1 characters CSI, SS3, and IND can be expressed as follows:

C1 Character	*7-bit Code Extension (Escape Sequence)*	
9/11	1/11	5/11
CSI	ESC	[
8/15	1/11	4/15
SS3	ESC	O
8/4	1/11	4/4
IND	ESC	D

The above code extension technique can be generalized as follows: You can express any C1 character as a two-character escape sequence whose second character has a code that is 40, hexadecimal (64, decimal), less than that of the C1 character. Conversely, you can make any supported escape sequence whose second character is in the range 4/0 through 5/15 one byte shorter by removing the ESC and adding 40, hexadecimal, to the code of the second character.

1.3.4.2 Control Sequences —A control sequence is a sequence of one or more ASCII graphic characters preceded by the control sequence introducer, the C1 character CSI (9/11). But CSI can be expressed as the 7-bit code extension ESC [(1/11 5/11). So you can express all control sequences as escape sequences whose second character code is [(5/11). For example, the following two sequences

9/11	3/15	3/3	6/8	
CSI	?	3	h	
1/11	5/11	3/15	3/3	6/8
ESC	[	?	3	h

are equivalent sequences that perform the same function. They cause the display to use 132 columns per line rather than 80.

Whenever possible, you should use CSI instead of ESC [to introduce a control sequence. CSI uses one less byte than ESC [, so you gain processing speed. But a sequence starting with CSI can be used only in an 8-bit environment, because CSI is a C1 character.

The next section explains how to handle 7- and 8-bit application environments.

1.3.5 Working with 7- and 8-Bit Environments

To take advantage of the Professional's 8-bit character set, your application and communications environment must be 8-bit compatible. The application can then transmit and receive the 8-bit GR graphic codes and C1 control codes as well as the 7-bit GL and C0 codes. The advantages of working in an 8-bit environment are:

- ☐ An expanded graphic character set is readily available to the user.
- ☐ Processing efficiency is gained by using 8-bit control characters in the application instead of the functionally equivalent two-byte 7-bit code extensions.

The 8-bit environment is the normal one for applications running in native mode.

In the terminal emulation modes, the application and/or communications environment may be only 7-bit compatible. Therefore, a PRO/Communications menu lets the operator enable the transmission of either only the 7-bit codes or both the 7- and 8-bit codes. This manual control is available for all terminal emulation modes, to satisfy the requirements of earlier and newer software.

Detailed conventions for working with 7- and 8-bit environments follow. The behavior of the Terminal Subsystem is emphasized. This is the behavior apparent to native mode applications and to applications running in terminal emulation modes when 8-bit characters are enabled by the operator.

When the operator chooses to enable only 7-bit codes, to accommodate a 7-bit communications environment, the PRO/Communications application alters the Terminal Subsystem's behavior. All such differences in behavior are noted.

1.3.5.1 Conventions for Codes Transmitted from Application—The Terminal Subsystem expects to receive character codes in a form consistent with the 8-bit coding convention described earlier in this section. This is the most efficient coding convention.

When operating in an 8-bit environment, your application can freely use the C0 and C1 control codes (including 7-bit code extensions). The Terminal Subsystem will always interpret these codes properly. When your application sends GL or GR graphic codes, the Terminal Subsystem interprets these according to the graphic character mapping currently being used. The default mapping, set when you power up (boot) the Professional, or reset the PRO/Communications

terminal emulator, or use a reset-to-initial-state (RIS) command (see Chapter 3), is the DEC Multinational Character Set: ASCII Graphics in GL and DEC Supplemental Graphics in GR.

The Terminal Subsystem always interprets the 8th bit of any received character as a legitimate value. In a 7-bit terminal emulation environment, this could cause the Terminal Subsystem to accept invalid data. To avoid this problem, the operator can choose to enable only 7-bit codes. By restricting the communications link to a 7-bit format, the operator ensures that any potentially invalid 8th bit value will be dropped before reaching the Terminal Subsystem. In terminal emulation, if the operator enables only 7-bit codes, the 8th bit of any graphic or control character transmitted from the host to the Professional is dropped.

1.3.5.2 Conventions for Codes Transmitted/Returned to Application—Codes transmitted to an application come primarily from keyboard input. In addition some codes are returned in response to commands such as requests for device identification or status.

Keyboard input includes:

- ☐ individual graphic character codes, both 7-bit (GR) and 8-bit (GL), produced by the standard (printing character) keys.
- ☐ individual control character codes and multi-byte sequences produced by the various function keys.

Multi-byte sequences are also returned in response to commands.

The next few paragraphs describe the behavior of the Terminal Subsystem, which is the behavior apparent to applications running in native mode. The behavior of the PRO/Communications application follows.

Behavior of Terminal Subsystem

The Terminal subsystem always transmits all 7-bit (GL) and 8-bit (GR) graphic character codes exactly as they are generated from the keyboard.

Most function keys on the Professional's keyboard generate multi-byte control codes (see Chapter 2). They can be expressed in 8-bit format starting either with CSI (9/11) or SS3 (8/15). These codes can also be expressed in the less efficient 7-bit-compatible format, as 7-bit code extensions. This is also true of returned codes for device status, etc.

The Terminal Subsystem can be programmed to transmit control codes in the 8-bit format, as needed. It can also be programmed to convert all 8-bit control codes to their equivalent 7-bit code extensions prior to transmission.

In an 8-bit environment, your application can issue the following command (called a "code extension announcer") to cause the Terminal Subsystem to return 8-bit control codes whenever possible.

1/11	2/0	4/7
ESC	SP	G

The above command ensures that no 7-bit code extensions are returned.

If your application environment cannot handle 8-bit control codes, receipt of an 8-bit control character with the 8th bit stripped off could cause unpredictable behavior. To avoid this possibility, your application should issue the following code extension announcer. It will cause the Terminal Subsystem to automatically convert all 8-bit control codes to their equivalent 7-bit code extensions prior to sending them to the application.

1/11	2/0	4/6
ESC	SP	F

This command would convert CSI to ESC [and SS3 to ESC O, for example.

Because 7-bit code extensions are compatible with both 7- and 8-bit environments, the command ESC SP F is invoked by default whenever the Professional is powered up (booted) or a reset-to-initial-state (RIS) command is issued by an application. To ensure the proper mode of operation, always use the appropriate code extension announcer in your application.

Behavior of PRO/Communications Application

The ability of the Professional to handle 8-bit codes in the terminal emulation modes is a superset feature that may not be compatible with earlier software.

To avoid potential problems, the emulator sends a reset-to-initial-state (RIS) command to the Terminal Subsystem whenever the operator enters terminal emulation mode, resets the emulator, or presses the **SET UP** key. RIS invokes the code extension announcer ESC SP F, causing the Terminal Subsystem to return all 8-bit control codes as 7-bit code extensions. Thus, by default, only 7-bit control codes are returned to the application after the operator has entered terminal emulation mode, reset the emulator, or pressed the **SET UP** key.

If the operator has enabled 8-bit codes, both 8-bit (GL) and 8-bit (GR) graphic codes can always be sent from the keyboard to the application.

Even if the operator has enabled 8-bit codes, by default only 7-bit control codes and code extensions will still be transmitted or returned to the application. To receive 8-bit control codes, the application must first send the ESC SP G code extension announcer to the Terminal Subsystem. Note that 8-bit codes may be transmitted with potentially unpredictable results if the application environment or communications link cannot handle them.

If the operator has enabled 7-bit codes, only 7-bit graphic codes can be transmitted from the keyboard to the application. If an 8-bit GR graphic character is generated at the keyboard, the 8th bit will be stripped by the emulator. The resultant code will represent the corresponding GL character on the code table.

If the operator has enabled 7-bit codes, the application will not be able to send an ESC SP G code extension announcer to the Terminal Subsystem. The command will be trapped by the emulator. Thus, since by default only 7-bit control codes can be returned by the Terminal Subsystem, no 8-bit control codes can be returned and altered by a 7-bit communications link.

The above behavior applies to all terminal emulation modes.

CHAPTER 2
CODES TRANSMITTED FROM KEYBOARD TO APPLICATION PROGRAM

This chapter describes the codes transmitted to an application program from the keyboard. All of the text modes of operation defined in Chapter 1 are covered.

It is assumed you are familiar with the character encoding conventions and terminology covered in Chapter 1, including the differences related to 7- and 8-bit environments.

Most of the function keys on the keyboard generate multi-byte codes. In this chapter, these codes are documented using 8-bit C1 control codes instead of their 7-bit code extensions, because this is the preferred, most efficient coding convention. The last part of Chapter 1 explains how your application can control the representation of C1 codes sent from the keyboard, to satisfy 7-bit or 8-bit environments.

The descriptions of the key codes generated are organized as follows in this chapter:

- ☐ Modes of operation and key usage (Section 2.1)
- ☐ Standard keys (the "printing", graphic character keys on the main array, Section 2.2)
- ☐ Compose sequences (Section 2.3)
- ☐ Control character keys (Section 2.4)
- ☐ Top-row function keys (Section 2.5)
- ☐ Editing keypad keys (Section 2.6)
- ☐ Auxiliary keypad keys (Section 2.7)

Key groupings are shown on Figure 2–1. Only illustrations of the U.S./Canada keyboard are used in this chapter. However, all significant differences among national keyboards are explained. Illustrations of all keyboards appear in Appendix B.

Sections 2.8 through 2.12 describe miscellaneous keyboard-related topics such as locking/unlocking the keyboard and autorepeat.

2.1 MODES OF OPERATION AND KEY USAGE

Note: Whenever a key is not available to an application, no code is transmitted to the application when the key is pressed. Pressing that key causes the keyboard bell to sound, if enabled (except for **HOLD SCREEN** and **PRINT SCREEN**, which are always controlled by P/OS).

2.1.1 Native and Professional Modes

All keys on the Professional keyboard are used in native and Professional modes. Except for the **BREAK** key, the key codes generated are identical in both modes (see Table 2–4).

2.1.2 VT102 Mode

For compatibility with a VT102 terminal, VT102 mode supports only a subset of the keys available on the Professional keyboard (see Figure 2–2). VT102 mode does not support the following Professional keys and key combinations.

- ☐ the six editing keys
- ☐ the top-row function keys except for **HOLD SCREEN**, **PRINT SCREEN**, **BREAK**, **SET-UP**, **F11 (ESC)**, **F12 (BS)**, and **F13, (LF)**

The **HOLD SCREEN** key on the Professional is equivalent to the **NO SCROLL** key on the main array of a VT102 terminal.

The **PRINT SCREEN** key on the Professional is available in all modes of operation to print the screen contents (text, graphics). On a VT102, there is no exact equivalent to the **PRINT SCREEN** key. You print the (text-mode) screen contents on a VT102 by pressing the **SHIFT** and **(PRINT)/ENTER** keys simultaneously.

The **BREAK** key generates a short break signal in VT102 mode.

The **SET-UP** key is used in VT102 mode to display PRO/Communications Set-Up menus.

The **F11 (ESC)**, **F12 (BS)**, and **F13, (LF)** keys on the Professional are equivalent to the **ESC**, **BACK SPACE**, and **LINE FEED** keys on the main array of a VT102 terminal. On the Professional, these three top-row function keys generate the ESC, BS, and LF control character codes when the Professional is being used in VT102 mode.

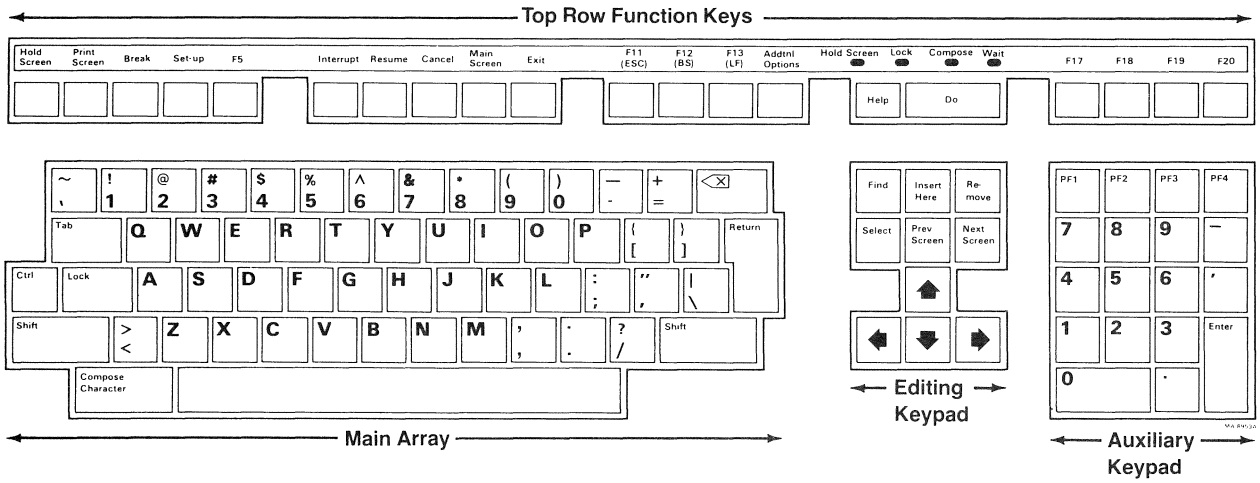


Figure 2-1
Key Groupings (U.S./Canada Keyboard)

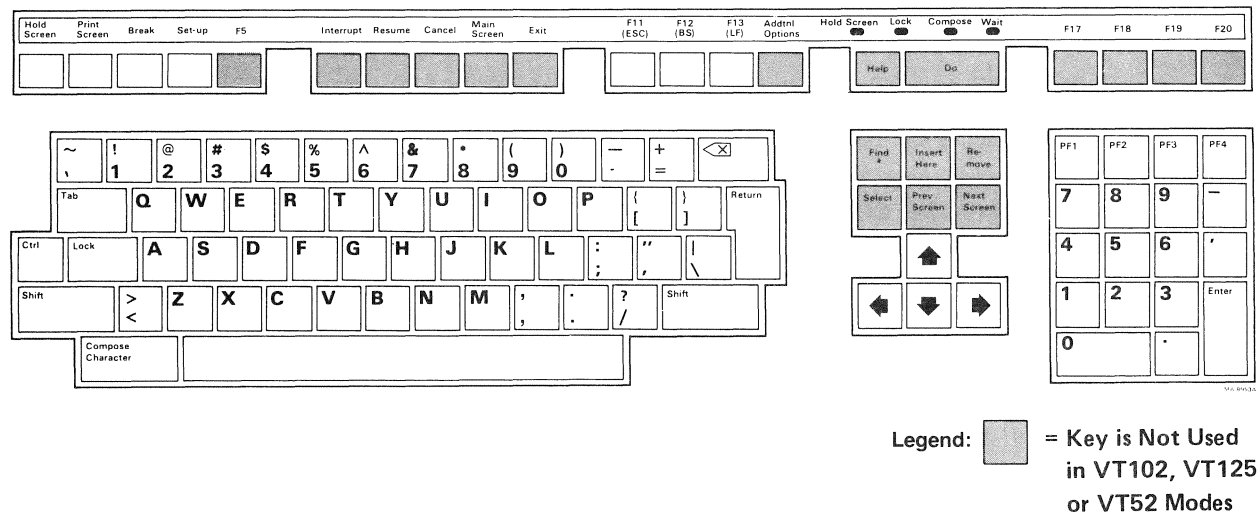


Figure 2-2
Keys Used in VT102, VT125, and VT52 Modes (U.S./Canada Keyboard)

2.1.3 VT125 Mode

For compatibility with a VT125 terminal, VT125 mode uses the same subset of the Professional keyboard as VT102 mode (see Figure 2–2). In VT125 mode, the keys generate the same codes as for VT102 mode (see "VT102 mode," above). All key codes documented for VT102 mode in this manual also apply to VT125 mode.

2.1.4 VT52 Mode

A VT102 terminal is able to emulate a VT52 terminal. Similarly, VT52 mode is implemented on the Professional as a subset of VT102 mode. VT52 mode on the Professional uses the same keys as VT102 mode (see Figure 2–2) .

Because VT102 mode is ANSI-compatible and VT52 mode is not, the codes generated by some function keys differ: The codes produced by the auxiliary keypad keys and the cursor control keys (on the editing keypad) are different for VT52 and VT102 modes.

These differences are identified throughout the manual. Because VT102, VT125, Professional, and native modes all share the ANSI-compatible coding conventions, the term "ANSI mode" is used to differentiate them from VT52 mode.

VT52 codes not specifically identified as being different are as documented for VT102 mode (see "VT102 mode," above).

2.2 STANDARD KEYS

The standard keys, located on the main array, are used to generate graphic characters, either singly or in combination with other keys.

When used with **SHIFT** and **LOCK**, standard keys generate shifted or upper-case characters, as described further in this section. When used with **CTRL**, standard keys generate 7-bit control characters, as described in Section 2.4. (**CTRL** takes precedence over **SHIFT**, which takes precedence over **LOCK**.) When used with **COMPOSE CHARACTER** and "dead" keys, standard keys generate graphic characters that may not be available from the standard keys on your keyboard (see Section 2.3, Compose Sequences).

On all keyboards, the characters shown on the standard keys and the codes generated correspond to the default (DEC Multinational) graphic character mapping: ASCII Graphics in GL and DEC Supplemental Graphics in GR, independently of the current character set selection. The DEC Multinational Character Set is shown in Appendix A. If you were to select another character set (e.g. DEC Special Graphics in GL), the GL codes would be mapped into (assigned to) the new graphic character set for display purposes. The standard keys would no longer indicate the actual graphic characters you would be creating from the keyboard. On any keyboard you would have to compare the relative position of DEC Multinational and DEC Special Graphics characters in the code table to determine what character any key might produce. The DEC Special Graphics Set is shown in Appendix A. Chapter 3 describes how to change character sets.

In the remainder of this chapter it is assumed that the default (DEC Multinational) character set is being used: ASCII Graphics in GL and DEC Supplemental Graphics in GR.

On the U.S./Canada keyboard, the standard keys show only ASCII characters. There are no DEC Supplemental characters among the standard keys. Thus, only ASCII codes are generated by these keys.

This is a special case. Most keyboards have some standard keys that generate DEC Supplemental as well as ASCII characters (see the keyboard illustrations in Appendix B). The standard-key patterns vary among keyboards, and some graphic characters (either special symbols or characters with diacritical marks) may or may not be available as standard keys on various keyboards. Nevertheless, on any keyboard, you can create any DEC Multinational graphic character that is not available through a standard key by using a compose sequence, as described in Section 2.3.

Regardless of how a graphic character is created (standard key or compose sequence), and regardless of which keyboard is used, each character is represented by a unique code according to the character's position in the code table. All GL characters can be used in both 7-bit and 8-bit environments. GR characters can be used only in an 8-bit environment.

2.2.1 SHIFT Key, LOCK Key, Shift-Lock/Caps-Lock Mode

The operations described are identical for all keyboards. It is assumed the **CTRL** key is released (up).

If the **SHIFT** key is released and the Lock indicator is turned off, any standard key with a single alphabetic character on the keycap generates the lowercase character. Any standard key with more than one character on the keycap generates the "lower" character shown. See also "Office Mode and Data Processing Mode" for keys with more than two characters on their keycaps.

When you press **SHIFT**, any standard key with a single alphabetic character on the keycap generates the uppercase character. Any standard key with more than one character on the keycap generates the "upper" character shown.

You use the **LOCK** key in conjunction with the Shift-Lock/Caps-Lock mode on one of the Professional's P/OS Set-Up menus.

When Caps-Lock mode is selected, if you press **LOCK**, the Lock indicator turns on and any standard key with a single alphabetic character on the keycap generates the uppercase character. Any standard key with more than one character on the keycap generates the "lower" character shown. You disable "Caps-Lock mode" by pressing **LOCK** again. This turns off the Lock indicator.

When Shift-Lock mode is selected, if you press **LOCK**, the Lock indicator turns on and any standard key with a single alphabetic character on the keycap generates the uppercase character. Any standard key with more than one character on the keycap generates the "upper" character shown. You disable "Shift-Lock mode" by pressing one of the **SHIFT** keys.

The **SHIFT** and **LOCK** keys have no effect on the non-standard keys.

2.2.2 Office Mode and Data Processing Mode

All keyboards except the U.S./Canada keyboard have one or more standard keys that create different graphic characters (and corresponding codes), depending on whether Office Mode or Data Processing (DP) Mode has been selected on a P/OS Set-Up menu.

Keys affected by Office or Data Processing Mode are marked on the keyboard illustrations in Appendix B. All such keys have more than two characters shown on their keycaps. The character(s) on the right side of the keycaps are generated in DP Mode. The character(s) on the left are generated in Office Mode. You select either shifted (upper) or unshifted (lower) character codes for these keys in the same way as for the other nonalphabetic keys.

2.3 COMPOSE SEQUENCES

You use compose sequences to create DEC Multinational graphic characters that do not exist as standard keys on your keyboard. There are two types of compose sequences: three-stroke sequences and two-stroke sequences. You should use two-stroke compose sequences whenever possible, because they are easier to use. However, two-stroke sequences create only certain characters and vary among national keyboards (not available on U.S./Canada keyboard).

You can use three-stroke sequences on all keyboards to create any DEC Multinational graphic character not on your keyboard. You create a three-stroke sequence by first pressing **COMPOSE CHARACTER** and then pressing two standard keys whose characters form a valid sequence.

Two-stroke sequences are faster to use than three-stroke sequences but are limited to characters with the following diacritical marks: grave accent, acute accent, circumflex, tilde, and diaeresis (umlaut). You don't use the **COMPOSE CHARACTER** key in a two-stroke sequence. Instead, you use a ''dead diacritical'' key (also known as dead key) to both initiate the sequence and enter a diacritical mark. You then enter a standard character that, together with that diacritical mark, results in a legitimate DEC Multinational character.

Dead keys are available on all but the U.S./Canada keyboard (see the keyboard illustrations in Appendix B). Dead keys vary among keyboards according to the relative usage of characters with diacritical marks. What may be a dead key on one keyboard may not be one on another. Also, only one of several characters shown on a keycap may be a dead key character.

All valid compose sequences are listed on Table 2–1. The rules for creating them are stated under ''Creating a Three-Stroke (Two-Stroke) Compose Sequence''. Because of variations among keyboards, you can create characters listed in Column 1 of Table 2–1 in one or more ways:

- with a standard key (if it is available on that keyboard)

Table 2-1
Valid Compose Sequences

In three-stroke sequences, the order in which required characters are entered does not matter unless otherwise stated. All two-stroke sequences are order sensitive.

			Characters Required in Sequence	
	(1) Resultant Character	*(2) Resultant Code*	*(3) Three-Stroke Sequence*	*(4) Two-Stroke Sequence*
"	(quotation mark)	2/02	" ⟨sp⟩	" ⟨sp⟩
#	(number sign)	2/03	+ +	
'	(apostrophe)	2/07	' ⟨sp⟩	' ⟨sp⟩
@	(commercial at)	4/00	a a or A A	
[	(opening bracket)	5/11	((	
\	(backslash)	5/12	/ /	
]	(closing bracket)	5/13	))	
^	(circumflex accent)	5/14	^ ⟨sp⟩	^ ⟨sp⟩
`	(grave accent)	6/00	` ⟨sp⟩	` ⟨sp⟩
{	(opening brace)	7/11	(-	
|	(vertical line)	7/12	/ ^	
}	(closing brace)	7/13	) -	
~	(tilde)	7/14	~ ⟨sp⟩	~ ⟨sp⟩
¡	(inverted !)	10/01	! !	
¢	(cent sign)	10/02	c / or C / or c | or C |	
£	(pound sign)	10/03	l - or L - or l = or L =	
¥	(yen sign)	10/05	y - or Y - or y = or Y =	
§	(section sign)	10/07	s o or S O or s 0 or S 0	
¤	(currency sign)	10/08	x o or X O or x 0 or X 0	
©	(copyright sign)	10/09	c o or C O or c 0 or C 0	
ª	(feminine ordinal indicator)	10/10	a _ or A _	
«	(angle quotation mark left)	10/11	⟨ ⟨	

Table 2-1 (Cont.)

			Characters Required in Sequence	
	(1) Resultant Character	*(2) Resultant Code*	*(3) Three-Stroke Sequence*	*(4) Two-Stroke Sequence*
°	(degree sign)	11/00	0 ^	
±	(plus/minus sign)	11/01	+ -	
2	(superscript 2)	11/02	2 ^	
3	(superscript 3)	11/03	3 ^	
µ	(micro sign)	11/05	/ u or / U (order sensitive)	
¶	(paragraph sign)	11/06	p ! or P !	
·	(middle dot)	11/07	. ^	
1	(superscript 1)	11/09	1 ^	
º	(masculine ordinal indicator)	11/10	o _ or O _	
»	(angle quotation mark right)	1/11	))	
¼	(fraction one-quarter)	11/12	1 4 (order sensitive)	
½	(fraction one-half)	11/13	1 2 (order sensitive)	
¿	(inverted ?)	11/15	? ?	
À	(A grave)	12/00	A `	` A
Á	(A acute)	12/01	A '	' A
Â	(A circumflex)	12/02	A ^	^ A
Ã	(A tilde)	12/03	A ~	~ A
Ä	(A umlaut)	12/04	A "	" A
Å	(A ring)	12/05	A * or A ° (degree sign)	
Æ	(A E ligature)	12/06	A E (order sensitive)	
C	(C cedilla)	12/07	C ,	
È	(E grave)	12/08	E `	` E
É	(E acute)	12/09	E '	' E
Ê	(E circumflex)	12/10	E ^	^ E
Ë	(E umlaut)	12/11	E "	" E

Table 2-1 (Cont.)

			Characters Required in Sequence	
	(1) Resultant Character	*(2) Resultant Code*	*(3) Three-Stroke Sequence*	*(4) Two-Stroke Sequence*
Ì	(I grave)	12/12	I `	` I
Í	(I acute)	12/13	I '	' I
Î	(I circumflex)	12/14	I ^	^ I
Ï	(I umlaut)	12/15	I "	" I
Ñ	(N tilde)	13/01	N ~	~ N
Ò	(O grave)	13/02	O `	` O
Ó	(O acute)	13/03	O '	' O
Ô	(O circumflex)	13/04	O ^	^ O
Õ	(O tilde)	13/05	O ~	~ O
Ö	(O umlaut)	13/06	O "	" O
Œ	(O E ligature)	13/07	O E (order sensitive)	
Ø	(O slash)	13/08	O /	
Ù	(U grave)	13/09	U `	` U
Ú	(U acute)	13/10	U '	' U
Û	(U circumflex)	13/11	U ^	^ U
Ü	(U umlaut)	13/12	U "	" U
Ÿ	(Y umlaut)	13/13	Y "	" Y
ß	(German small sharp s)	13/15	s s	
à	(a grave)	14/00	a `	` a
á	(a acute)	14/01	a '	' a
â	(a circumflex)	14/02	a ^	^ a
ã	(a tilde)	14/03	a ~	~ a
ä	(a umlaut)	14/04	a "	" a
å	(a ring)	14/05	a * or a ° (degree sign)	
æ	(a e ligature)	14/06	a e (order sensitive)	
ç	(c cedilla)	14/07	c ,	

Table 2-1 (Cont.)

			Characters Required in Sequence	
	(1) Resultant Character	*(2) Resultant Code*	*(3) Three-Stroke Sequence*	*(4) Two-Stroke Sequence*
è	(e grave)	14/08	e `	` e
é	(e acute)	14/09	e '	' e
ê	(e circumflex)	14/10	e ^	^ e
ë	(e umlaut)	14/11	e "	" e
ì	(i grave)	14/12	i `	` i
í	(i acute)	14/13	i '	' i
î	(i circumflex)	14/14	i ^	^ i
ï	(i umlaut)	14/15	i "	" i
ñ	(n tilde)	15/01	n ~	~ n
ò	(o grave)	15/02	o `	` o
ó	(o acute)	15/03	o '	' o
ô	(o circumflex)	15/04	o ^	^ o
õ	(o tilde)	15/05	o ~	~ o
ö	(o umlaut)	15/06	o "	" o
œ	(o e ligature)	15/07	o e (order sensitive)	
ø	(o slash)	15/08	o /	
ù	(u grave)	15/09	u `	` u
ú	(u acute)	15/10	u '	' u
û	(u circumflex)	15/11	u ^	^ u
ü	(u umlaut)	15/12	u "	" u
ÿ	(y umlaut)	15/13	y "	" y

- ☐ with a three-stroke compose sequence (always)
- ☐ with a two-stroke compose sequence (if the dead key is available on that keyboard)

In all cases, the resultant code sent to the application is the DEC Multinational code for that graphic character (column 2 of Table 2–1).

Use of the **SHIFT** and **LOCK** keys and of Office and DP modes in compose sequences is as defined for the standard keys.

2.3.1 Creating a Three-Stroke Compose Sequence

To create a three-stroke compose sequence:

1. Find the desired character in column 1 of Table 2–1.
2. Press **COMPOSE CHARACTER**. The Compose indicator turns on, showing you are in compose mode.
3. Type the two required characters from column 3.

For example, to create E with acute accent, press **COMPOSE CHARACTER**, then type E and apostrophe, in any order.

When a valid sequence is completed, the Compose indicator turns off, and the resultant character code (column 2) is sent to the application.

If the sequence consists of two alphabetic characters, the case of both must be the same (upper or lower) and determines the case of the resultant character, if relevant. Table 2–2 summarizes how the Professional handles all valid and invalid conditions.

2.3.2 Creating a Two-Stroke Compose Sequence

To form a two-stroke sequence (applies to all keyboards except U.S./Canada):

1. Find the desired character in column 1 of Table 2–1.
2. Verify from column 4 that it can be created with a two-stroke sequence.
3. Verify from Appendix B that your keyboard has a dead key for the needed diacritical mark.
4. Press the dead key to enter the diacritical mark. The Compose indicator turns on, showing you are in compose mode.
5. Type the remaining character indicated in column 4.

For example, to create E with a grave accent on a Danish keyboard, press the dead key that has the grave accent and then type E.

When a valid sequence is completed, the Compose indicator turns off, and the resultant character code (column 2) is sent to the application. Table 2–2 summarizes how the Professional handles all valid and invalid conditions.

Table 2-2
Valid and Invalid Compose Sequence Summary

Three-stroke sequence: **COMPOSE CHARACTER**, key X, key Y
Two-stroke sequence: dead key, key Z

Condition	*Action*
1. X, Y, Z are valid characters per Column 3 or 4 of Table 2-1	Valid sequence: Resultant code is sent to application.
2. X, Y, Z are standard characters but do not conform to Table 2-1	Invalid sequence: Fails after second key. Bell sounds,if enabled. Exit compose mode. Nothing is sent to application or displayed on screen.
3. X, Y, or Z is a dead key	Diacritical mark on dead key is treated as if it were its "equivalent" ASCII character, defined as follows: *Diacritical Mark* — *Equivalent ASCII Char.* diaeresis — double quote " (2/2) acute accent — apostrophe ' (2/7) grave accent — grave accent ` (6/0) circumflex — circumflex ^ (5/14) tilde — tilde ~ (7/14) Sequence is then handled per item 1 or 2, above, and may be valid or invalid.
5. X, Y, or Z is a non-standard key or key combination (e.g. **TAB**, **RETURN, CTRL**/sequence, top-row function key, editing keypad key, auxiliary keypad key)	Invalid sequence: Fails immediately. Sequence handled per item 2, above. The non-standard code is then interpreted as if it were external to the sequence and is handled normally, as defined elsewhere in this chapter.
6. X, Y, or Z is **DEL**	Exit compose mode. Nothing is sent to application or displayed on screen.
7. X, Y, or Z is **COMPOSE CHARACTER**	New three-stroke sequence is started. Nothing is sent to application or displayed on screen.

2.3.3 Aborting or Restarting a Compose Sequence

If you enter compose mode inadvertently, either by pressing **COMPOSE CHARACTER** or a dead key, press the Delete key to immediately terminate the compose sequence and exit compose mode. No code is transmitted to the application.

If you press **COMPOSE CHARACTER** during a compose sequence, a new three-stroke sequence is started from that point. The previous sequence is aborted with no effect on the application. If you press a dead key during a compose sequence, the action is as indicated in Table 2-2.

2.3.4 More about Dead Keys

The illustrations in Appendix B show that some keyboards have keys that contain both a "normal" (standard) and a "dead" character. As with standard keys, you select the character you want with the **SHIFT** and **LOCK** keys and the appropriate modes on the P/OS Set-Up menus. You must be in native mode to use a P/OS Set-Up menu.

Some keys may appear to be dead keys because they have ASCII characters on their keycaps (`, ', ^, ~, ") that look like diacritical marks. However, these keys are standard keys that generate their own unique ASCII graphic character codes.

By themselves, diacritical marks are not DEC Multinational characters. When a DEC Multinational character has a diacritical mark, it is the entire character (including the mark) that is assigned a unique code. Similarly, dead keys do not generate a code by themselves. They must be used with a standard character, either a printing character or the "space" character (2/0).

If a dead key is used within a three-stroke compose sequence, the diacritical mark is treated as if it were its "equivalent" ASCII character, defined as follows.

Diacritical Mark	*Equivalent ASCII Character*		
diaeresis (umlaut)	double quote	"	(2/2)
acute accent	apostrophe	'	(2/7)
grave accent	grave accent	`	(6/0)
circumflex	circumflex	^	(5/14)
tilde	tilde	~	(7/14)

See Table 2–2 for more information.

2.4 KEYS USED TO GENERATE 7-BIT CONTROL CHARACTERS

You can generate 7-bit (C0) control characters in three ways from any keyboard. Brief definitions of each character are in Appendix A.

1. In all modes of operation, you can generate the Horizontal Tab (HT, 0/9), Carriage Return (CR, 0/13), or Delete (DEL, 7/15) codes by pressing dedicated function keys on the main array: **TAB**, **RETURN**, (Delete).

   - ☐ The code generated by **RETURN** depends on the set/reset state of Line Feed/New Line Mode (LNM). Use of this mode is described in Chapter 3. When LNM is reset, **RETURN** generates CR only. When LNM is set, **RETURN** generates CR followed by LF. In terminal emulation modes, you can also set and reset LNM manually from a PRO/Communications menu. Refer to the *PRO/Communications Manual* for details.

   - ☐ Depending on the state of Auxiliary Keypad Mode (DECKPNM/DECKPAM), **RETURN** and **ENTER** (on the auxiliary keypad) may generate the same control codes. Refer to Chapter 3 for information on using the DECKPNM and DECKPAM commands.

2. In VT102, VT125, and VT52 modes you can generate the ESC (1/11), BS (0/8), and LF (0/10) codes directly, by pressing top-row function keys **F11 (ESC)**, **F12 (BS)**, and **F13 (LF)**, respectively. This feature is provided for compatibility with VT102 and VT125 terminals. There are

Table 2-3
Keys Used to Generate 7-bit Control Characters

Control Character Mnemonic	*Code*	*Key Pressed with* **CTRL** *(all modes)*	*Key Pressed with* **CTRL** *(VT102 terminal)*	*Dedicated Function Key*
NUL	0/00	2	Space bar	
SOH	0/01	A	A	
STX	0/02	B	B	
ETX	0/03	C	C	
EOT	0/04	D	D	
ENQ	0/05	E	E	
ACK	0/06	F	F	
BEL	0/07	G	G	
BS	0/08	H	H	**F12 (BS)***
HT	0/09	I	I	**TAB**
LF	0/10	J	J	**F13 (LF)***
VT	0/11	K	K	
FF	0/12	L	L	
CR	0/13	M	M	**RETURN**
SO	0/14	N	N	
SI	0/15	O	O	
DLE	1/00	P	P	
DC1	1/01	Q	Q	
DC2	1/02	R	R	
DC3	1/03	S	S	
DC4	1/04	T	T	
NAK	1/05	U	U	
SYN	1/06	V	V	
ETB	1/07	W	W	
CAN	1/08	X	X	
EM	1/09	Y	Y	
SUB	1/10	Z	Z	
ESC	1/11	3	[	**F11 (ESC)***
FS	1/12	4	/	
GS	1/13	5	]	
RS	1/14	6	~	
US	1/15	7	?	
DEL	7/15	8		**DELETE**

* Keys F11, F12, and F13 generate these 7-bit control characters only when the Professional is operated in VT102, VT125, and VT52 modes.

no **Escape**, **Backspace**, or **Line Feed** keys on the main array of the Professional keyboard.

3. In all modes of operation, you can generate any of the 32 C0 7-bit control characters, plus DEL, from the keyboard by simultaneously pressing the **CTRL** key and a specified standard key.

The keys and key combinations used to generate 7-bit control characters for all modes of operation are listed in Table 2-3. These keys and combinations are valid on all keyboards. Note that there are differences between the Professional and a VT102 terminal in the way you generate 7-bit control characters with the **CTRL** key.

There is no similar mechanism for generating C1 (8-bit) control characters.

Invalid key combinations cause the bell to sound if enabled. No error code is returned to the application.

2.5 TOP-ROW FUNCTION KEYS

The function keys at the top row of the keyboard generate the codes listed in Table 2-4. The following paragraphs provide additional details on these keys.

The **HOLD SCREEN** and **PRINT SCREEN** keys are always reserved for their intended functions, and their codes are never available to your application. Pressing **HOLD SCREEN** turns on the Hold Screen indicator and causes output to the display to be suspended until any key, including **HOLD SCREEN**, is pressed. Pressing **PRINT SCREEN** causes the contents of the screen to be printed, if a printer is connected to the Professional. Refer to the description of the printing commands in Chapter 3 for more information.

The **BREAK** key's CSI 1 3 ~ code is generated only in native mode. In terminal emulation modes, the **BREAK** key generates a "short" break on the communications line. There is no long break, such as is available on a VT102 terminal by pressing the **SHIFT** and **BREAK** keys simultaneously.

The **SET-UP** key is used to display P/OS Set-Up menus in native mode and PRO/Communications Set-Up menus in terminal emulation modes. It is also the only means of unlocking the keyboard manually. See Section 2.8 for details.

In native mode, some of the other top-row function keys are used with P/OS routines related to menus, forms, and Help frames (see the *Tool Kit User's Guide*). The codes listed are available to your application whenever the associated P/OS routines are not active. See the *P/OS System Reference Manual* for a complete description of the **INTERRUPT** key when used with various keys, including **DO**.

In VT102, VT125, and VT52 modes, keys **F11 (ESC)**, **F12 (BS)**, and **F13 (LF)** emulate the **ESC**, **BACK SPACE**, and **LINE FEED** keys on the main array of a VT102 terminal and generate the corresponding 7-bit codes: ESC (1/11), BS (0/8), and LF (0/10).

Table 2-4
Codes Generated by Top Row Function Keys

Name on Legend Strip	*Generic Name*	*Code Generated*				
		Native/Professional Modes				*VT102/VT125/ VT52 Modes*
HOLD SCREEN	F1			—		—
PRINT SCREEN	F2			—		—
BREAK	F3					
		9/11	3/1	3/3	7/14	
Native Mode		CSI	1	3	~	—
Terminal Emul. Modes			Break			Break
		9/11	3/1	3/4	7/14	
SET-UP	F4	CSI	1	4	~	—
		9/11	3/1	3/5	7/14	
F5	F5	CSI	1	5	~	—
		9/11	3/1	3/7	7/14	
INTERRUPT	F6	CSI	1	7	~	—
		9/11	3/1	3/8	7/14	
RESUME	F7	CSI	1	8	~	—
		9/11	3/1	3/9	7/14	
CANCEL	F8	CSI	1	9	~	—
		9/11	3/2	3/0	7/14	
MAIN SCREEN	F9	CSI	2	0	~	—
		9/11	3/2	3/1	7/14	
EXIT	F10	CSI	2	1	~	—
		9/11	3/2	3/3	7/14	1/11
F11 (ESC)	F11	CSI	2	3	~	ESC
		9/11	3/2	3/4	7/14	0/8
F12 (BS)	F12	CSI	2	4	~	BS
		9/11	3/2	3/5	7/14	0/10
F13 (LF)	F13	CSI	2	5	~	LF
		9/11	3/2	3/6	7/14	
ADDTNL OPTIONS	F14	CSI	2	6	~	—
		9/11	3/2	3/8	7/14	
HELP	F15	CSI	2	8	~	—
		9/11	3/2	3/9	7/14	
DO	F16	CSI	2	9	~	—
		9/11	3/3	3/1	7/14	
F17	F17	CSI	3	1	~	—
		9/11	3/3	3/2	7/14	
F18	F18	CSI	3	2	~	—

Table 2-4 (Cont.)

Name on Legend Strip	*Generic Name*	*Code Generated* — *Native/Professional Modes*	*VT102/VT125/ VT52 Modes*
F19	F19	9/11 CSI 3/3 3 3/3 3 7/14 ~	—
F20	F20	9/11 CSI 3/3 3 3/4 4 7/14 ~	—

2.6 EDITING KEYPAD

This keypad has six editing keys and four cursor control keys. In native mode, some of these keys are used with P/OS routines related to menus,forms, and help frames (see the *Tool Kit User's Guide*). The codes listed here are available to your application whenever the associated P/OS routines are not active.

2.6.1 Editing Keys

The six editing keys can be used only in native and Professional modes. No codes from these keys are transmitted to the application in VT102, VT125, or VT52 modes. Table 2-5 lists the codes generated by the editing keys for all modes of operation.

Table 2-5
Codes Generated by Editing Keys

Key	*Native/Professional Mode*	*VT102/VT125/ VT52 Modes*
FIND	9/11 CSI 3/1 1 7/14 ~	—
INSERT HERE	9/11 CSI 3/2 2 7/14 ~	—
REMOVE	9/11 CSI 3/3 3 7/14 ~	—
SELECT	9/11 CSI 3/4 4 7/14 ~	—
PREV SCREEN	9/11 CSI 3/5 5 7/14 ~	—
NEXT SCREEN	9/11 CSI 3/6 6 7/14 ~	—

2.6.2 Cursor Control Keys

The cursor control keys can be used in all modes. The codes generated by these keys depend on whether the Professional is being operated in an ANSI-compatible mode (native, Professional, VT102, or VT125 modes) or in VT52 mode.

Table 2-6 shows the possible codes generated by the cursor control keys.

In ANSI mode, the codes depend on whether the auxiliary keypad is being used in application mode (DECKPAM) or numeric mode (DECKPNM) and also depend on the state of cursor key mode (DECCKM).

- ☐ If keypad application mode (DECKPAM) is selected, the cursor control keys generate either application codes or ANSI cursor control codes. The exact codes depend on the state of cursor key mode (DECCKM). You can use and interpret the application control codes specifically for your application.
- ☐ If keypad numeric mode (DECKPNM) is selected, the cursor control keys generate only ANSI cursor control codes, regardless of the state of DECCKM.

In VT52 mode, the cursor control keys always generate the VT52-compatible cursor control sequences.

Table 2-6
Codes Generated by Cursor Control Keys

	ANSI Mode [A]				*VT52 Mode* [A]	
	Cursor Key Mode Reset		*Cursor Key Mode Set*			
Key	*(Cursor control)* [B]		*(Application)* [C]			
↑	9/11	4/1	8/15	4/1	1/11	4/1
	CSI	A	SS3	A	ESC	A
↓	9/11	4/2	8/15	4/2	1/11	4/2
	CSI	B	SS3	B	ESC	B
→	9/11	4/3	8/15	4/3	1/11	4/3
	CSI	C	SS3	C	ESC	C
←	9/11	4/4	8/15	4/4	1/11	4/4
	CSI	D	SS3	D	ESC	D

A. ANSI mode applies to all modes except VT52 mode. VT52 mode is an ANSI-incompatible subset of VT102 mode.

B. These codes can be generated if either keypad application mode (DECKPAM) or keypad numeric mode (DECKPNM) is selected.

C. These codes can be generated only if keypad application mode (DECKPAM) is selected.

Refer to Chapter 3 for information on using the DECCKM, DECKPAM, DECKPNM, and DECANM commands (sequences).

2.7 AUXILIARY KEYPAD

The auxiliary keypad keys can be used in all modes. The codes generated by these keys depend on whether the Professional is being operated in an ANSI-compatible mode (native, Professional, VT102, or VT125 modes) or in VT52 mode.

Table 2–7 shows the possible codes generated by the auxiliary keypad keys.

In both ANSI and VT52 modes, the codes generated depend on whether the auxiliary keypad is being used in application mode or numeric mode. In numeric mode the auxiliary keypad keys generate the numeric, comma, period, and minus sign character codes used by the corresponding standard keys. In application mode, the keypad keys generate control codes that you can use and interpret specifically for your application.

In ANSI mode, keypad application mode is selected through the DECKPAM sequence, and keypad numeric mode is selected through the DECKPNM sequence.

Refer to Chapter 3 for the ANSI-compatible and VT52-compatible commands needed to select keypad application mode and keypad numeric mode.

Table 2-7
Codes Generated by Auxiliary Keypad Keys

	ANSI Mode [A]		*VT52 Mode* [A]	
Key	*Keypad Numeric Mode*	*Keypad Application Mode*	*Keypad Numeric Mode*	*Keypad Application Mode*
0	3/0 0	8/15 7/0 SS3 p	3/0 0	1/11 3/15 7/0 ESC ? p
1	3/1 1	8/15 7/1 SS3 q	3/1 1	1/11 3/15 7/1 ESC ? q
2	3/2 2	8/15 7/2 S3 r	3/2 2	1/11 3/15 7/2 ESC ? r
3	3/3 3	8/15 7/3 SS3 s	3/3 3	1/11 3/15 7/3 ESC ? s
4	3/4 4	8/15 7/4 SS3 t	3/4 4	1/11 3/15 7/4 ESC ? t
5	3/5 5	8/15 7/5 SS3 u	3/5 5	1/11 3/15 7/5 ESC ? u
6	3/6 6	8/15 7/6 SS3 v	3/6 6	1/11 3/15 7/6 ESC ? v
7	3/7 7	8/15 7/7 SS3 w	3/7 7	1/11 3/15 7/7 ESC ? w
8	3/8 8	8/15 7/8 SS3 x	3/8 8	1/11 3/15 7/8 ESC ? x
9	3/9 9	8/15 7/9 SS3 y	3/9 9	1/11 3/15 7/9 ESC ? y
-	2/13 - (minus)	8/15 6/13 SS3 m	2/13 -	1/11 3/15 6/13 [B] ESC ? m
,	2/12 , (comma)	8/15 6/12 SS3 l	2/12 ,	1/11 3/15 6/12 [B] ESC ? l
.	2/14 . (period)	8/15 6/14 SS3 n	2/14 .	1/11 3/15 6/14 ESC ? n
Enter [C]	0/13 CR or 0/13 0/10 CR LF	8/15 4/13 SS3 M	0/13 CR or 0/13 0/10 CR LF	1/11 3/15 4/13 ESC ? M

A. ANSI mode applies to all modes except VT52 mode. VT52 mode is an ANSI-incompatible subset of VT102 mode.

B. You cannot generate these sequences on a VT52 terminal.

C. In Keypad Numeric Mode, **ENTER** generates the same codes as **RETURN**. You can change the code generated by **RETURN** with the Linefeed/New Line Mode. When reset, the Linefeed/New Line Mode causes **RETURN** to generate a single control character (CR). When set, the mode causes **RETURN** to generate two control characters (CR,LF).

Table 2-7 (Cont.)

	ANSI Mode [A]				*VT52 Mode* [A]			
Key	*Keypad Numeric Mode*		*Keypad Application Mode*		*Keypad Numeric Mode*		*Keypad Application Mode*	
	8/15	5/0	8/15	5/0	1/11	5/0	1/11	5/0
PF1	SS3	P	SS3	P	ESC	P	ESC	P
	8/15	5/1	8/15	5/1	1/11	5/1	1/11	5/1
PF2	SS3	Q	SS3	Q	ESC	Q	ESC	Q
	8/15	5/2	8/15	5/2	1/11	5/2	1/11	5/2
PF3	SS3	R	SS3	R	ESC	R	ESC	R
	8/15	5/3	8/15	5/3	1/11	5/3	1/11	5/3 [B]
PF4	SS3	S	SS3	S	ESC	S	ESC	S

A. ANSI mode applies to all modes except VT52 mode. VT52 mode is an ANSI-incompatible subset of VT102 mode.

B. You cannot generate these sequences on a VT52 terminal.

C. In Keypad Numeric Mode, **ENTER** generates the same codes as **RETURN**. You can change the code generated by **RETURN** with the Linefeed/New Line Mode. When reset, the Linefeed/New Line Mode cause **RETURN** to generate a single control character (CR). When set, the mode causes **RETURN** to generate two control characters (CR,LF).

2.8 LOCKING AND UNLOCKING THE KEYBOARD

2.8.1 Locking the Keyboard

When the keyboard is "locked", no codes can be transmitted from the keyboard to the application (and/or to the host system in the terminal emulation modes). To alert the user, whenever the keyboard is locked, the Wait indicator at the top of the keyboard is turned on and the keyclick feature is disabled.

The keyboard can become locked under the following conditions.

In all modes, if the application sends a command to set Keyboard Action Mode (KAM), the keyboard locks immediately and no further keyboard input is sent to the application. Chapter 3 describes the KAM command.

A DC3 control character (1/3, also known as XOFF) sent to the Professional will also lock the keyboard. However, the exact action depends on the mode of operation, as described below. DC3 is cleared by sending a DC1 control character (1/1, also known as XON) to the Professional.

1. If the Terminal Subsystem receives DC3 directly, the keyboard locks immediately and no further keyboard input is sent to the application. This is the only response to DC3 in native mode.
2. In the terminal emulation modes, the host system will send DC3 to the Professional if no further input from the Professional can be handled at that moment. When the host can accept more input, it clears DC3 by sending DC1 to the Professional. The response of the Professional to DC3 and DC1 depends on whether XON/XOFF recognition has been enabled or disabled through a PRO/Communications menu.

   If XON/XOFF recognition is enabled, when the host or application sends DC3, no more characters are transmitted to the host. A type-ahead buffer temporarily stores any additional characters generated from the Professional keyboard. The capacity of the buffer is 256 characters. If the host sends DC1 before the buffer is filled, transmission is enabled, and the stored characters are sent to the host. If the buffer fills up before the host can clear DC3, the keyboard locks for the next and subsequent characters. No more characters can be generated from the keyboard.

   If XON/XOFF recognition is disabled, the PCF passes DC3 directly to the Terminal Subsystem. When the host or application sends DC3, the keyboard locks immediately and no more characters are sent to the host (the response is the same as for native mode).

2.8.2 Unlocking the Keyboard

Two conditions must be satisfied for the keyboard to become unlocked:

1. KAM must be reset, and
2. Any uncleared DC3 command must be cleared by a DC1 command.

The Wait indicator is turned off as soon as the keyboard is unlocked.

The keyboard can be unlocked manually as follows. The procedure is different for native mode and for the terminal emulation modes.

To unlock the keyboard manually in native mode, press the **SET-UP** key. Note that the **SET-UP** key also sends its own code to the application (see Section 2.5), so this method may cause unwanted side effects in native mode.

To unlock the keyboard manually in terminal emulation mode, press the **SET-UP** key, then reset the emulator through the PRO/Communications menu selection. Resetting the emulator unlocks the keyboard, resets the default emulator characteristics, and sends a Reset-to-Initial-State (RIS) command to the Terminal Subsystem. RIS resets the P/OS programmable modes and the keyboard states to their default values. Refer to the *PRO/Communications Manual* for details on the default emulator characteristics. Refer to Chapter 3 of this manual for a description of the RIS command and the default states it controls.

2.9 ENABLING AND DISABLING AUTOREPEAT

You can enable the autorepeat feature either by selecting Autorepeat On through a P/OS Set-Up menu, or by setting Autorepeat Mode (DECARM) in your application. You disable autorepeat either by selecting Autorepeat Off on the Set-Up menu or by resetting DECARM in your application. Chapter 3 describes the use of the DECARM command.

The autorepeat rate is 30 keystrokes (not characters generated) per second. Some repeating keys, such as the cursor keys, generate more than one character per keystroke.

If autorepeat is enabled,

- ☐ the Delete key and the four cursor keys on the editing keypad automatically repeat if pressed longer than 0.3 second.
- ☐ all standard keys and all auxiliary keypad keys automatically repeat if pressed longer than 0.5 second.
- ☐ if two or more keys are held down, the most recently depressed key autorepeats.

The following keys never autorepeat: all top row function keys; all editing keypad keys except for the cursor keys; and the **TAB**, **RETURN**, **COMPOSE CHARACTER**, **LOCK**, **SHIFT**, and **CTRL**keys. Also, when **CTRL** is pressed, autorepeating is temporarily inhibited when a "control-able" key (see Table 2–3) is pressed simultaneously.

2.10 BELL SIGNAL

You disable and enable the keyboard bell from a P/OS Set-Up menu.

If you select Bell Off on the Set-Up menu, no bell sound will be generated under any conditions.

If you select Bell On, the bell will sound under any of the following conditions:

1. If a BEL code (0/7) is sent to the Professional, either from the keyboard (Ctrl/G) or from the application.
2. If Margin Bell On was selected on the appropriate P/OS Set-Up menu and you try to type characters past the right margin.
3. If the keyboard input is invalid in the current context. The following are examples of invalid input:
   - ☐ Using an invalid compose sequence
   - ☐ Using an invalid function key in VT102, VT125, or VT52 mode

2.11 MONITOR SCREEN BLANKING AND RESTORE

Any display on the screen will disappear (screen will go blank) if there has been no activity in the Terminal Subsystem for 30 minutes. This includes keyboard input and any other code processing on the part of the Terminal Subsystem, whether displayable on the screen or not. This is to reduce the chance of damage to the screen phosphor from a fixed display, such as a menu.

The original screen image will be redisplayed automatically as soon as any key on the Professional keyboard is pressed. This includes keys like **CTRL**, **SHIFT**, and **COMPOSE** which don't by themselves transmit codes to the application. Use of the **CTRL** key is the recommended way of redisplaying the image. The image will also be redisplayed if the Terminal Subsystem receives any codes from any source, including characters and sequences that do not normally affect the display.

2.12 CHANGING THE KEYBOARD

Changing the keyboard has no effect on the current keyboard states (keyboard locked/unlocked, autorepeat on/off, keyclicks on/off, etc.). The Hold Screen, Lock, Compose, and Wait indicators at the top of the keyboard maintain their current status.

CHAPTER 3
RESPONSE OF PROFESSIONAL TO RECEIVED CODES

This chapter describes the Professional's response to codes it may receive from an application or a host system when operating in text mode. It is assumed you are familiar with the character encoding conventions and terminology covered in Chapter 1. The last part of that chapter describes how to work with 7- and 8-bit environments.

All of the text modes of operation defined in Chapter 1 are covered. Most of this chapter is devoted to the ANSI-compatible modes, namely native mode, Professional mode, VT102 mode, and VT125 mode. VT52 mode, being ANSI-incompatible, is treated separately at the end of the chapter. The term "ANSI mode" is used to differentiate the ANSI-compatible modes from VT52 mode. Differences in response between the Professional and a VT102 terminal are pointed out throughout the chapter.

All data received by the Professional in text mode consists of single- and multiple-character codes: graphic (printing or display) characters, control characters, escape sequences, and control sequences. Much of that data consists of graphic characters that simply appear on the screen with no other effect. Control characters, escape sequences, and control sequences are all "control functions" that you use in your application to specify how the Professional should process, transmit, and display characters. Each control function has a unique name and each name has a unique abbreviation. Both the name and abbreviation are standardized. Abbreviations are in the form of mnemonics to help you remember them.

By default, the Professional interprets individual control and graphic character codes according to the 8-bit DEC Multinational Character Set code mapping (see Table 3–1).

Table 3-1
DEC Multinational Character Set (C0 and GL Codes)

ROW	COLUMN 0		COLUMN 1		COLUMN 2		COLUMN 3		COLUMN 4		COLUMN 5		COLUMN 6		COLUMN 7	
	b8 b7 b6 b5 BITS b4 b3 b2 b1	0 0 0 0	0 0 0 1		0 0 1 0		0 0 1 1		0 1 0 0		0 1 0 1		0 1 1 0		0 1 1 1	
0	0 0 0 0	NUL	0 0 0	DLE	20 16 10	SP	40 32 20	0	60 48 30	@	100 64 40	P	120 80 50	`	140 96 60	p	160 112 70
1	0 0 0 1	SOH	1 1 1	DC1 (XON)	21 17 11	!	41 33 21	1	61 49 31	A	101 65 41	Q	121 81 51	a	141 97 61	q	161 113 71
2	0 0 1 0	STX	2 2 2	DC2	22 18 12	"	42 34 22	2	62 50 32	B	102 66 42	R	122 82 52	b	142 98 62	r	162 114 72
3	0 0 1 1	ETX	3 3 3	DC3 (XOFF)	23 19 13	#	43 35 23	3	63 51 33	C	103 67 43	S	123 83 53	c	143 99 63	s	163 115 73
4	0 1 0 0	EOT	4 4 4	DC4	24 20 14	$	44 36 24	4	64 52 34	D	104 68 44	T	124 84 54	d	144 100 64	t	164 116 74
5	0 1 0 1	ENQ	5 5 5	NAK	25 21 15	%	45 37 25	5	65 53 35	E	105 69 45	U	125 85 55	e	145 101 65	u	165 117 75
6	0 1 1 0	ACK	6 6 6	SYN	26 22 16	&	46 38 26	6	66 54 36	F	106 70 46	V	126 86 56	f	146 102 66	v	166 118 76
7	0 1 1 1	BEL	7 7 7	ETB	27 23 17	'	47 39 27	7	67 55 37	G	107 71 47	W	127 87 57	g	147 103 67	w	167 119 77
8	1 0 0 0	BS	10 8 8	CAN	30 24 18	(	50 40 28	8	70 56 38	H	110 72 48	X	130 88 58	h	150 104 68	x	170 120 78
9	1 0 0 1	HT	11 9 9	EM	31 25 19	)	51 41 29	9	71 57 39	I	111 73 49	Y	131 89 59	i	151 105 69	y	171 121 79
10	1 0 1 0	LF	12 10 A	SUB	32 26 1A	*	52 42 2A	:	72 58 3A	J	112 74 4A	Z	132 90 5A	j	152 106 6A	z	172 122 7A
11	1 0 1 1	VT	13 11 B	ESC	33 27 1B	+	53 43 2B	;	73 59 3B	K	113 75 4B	[	133 91 5B	k	153 107 6B	{	173 123 7B
12	1 1 0 0	FF	14 12 C	FS	34 28 1C	,	54 44 2C	<	74 60 3C	L	114 76 4C	\	134 92 5C	l	154 108 6C	|	174 124 7C
13	1 1 0 1	CR	15 13 D	GS	35 29 1D	-	55 45 2D	=	75 61 3D	M	115 77 4D	]	135 93 5D	m	155 109 6D	}	175 125 7D
14	1 1 1 0	SO	16 14 E	RS	36 30 1E	.	56 46 2E	>	76 62 3E	N	116 78 4E	^	136 94 5E	n	156 110 6E	~	176 126 7E
15	1 1 1 1	SI	17 15 F	US	37 31 1F	/	57 47 2F	?	77 63 3F	O	117 79 4F	_	137 95 5F	o	157 111 6F	DEL	177 127 7F

C0 CODES — GL CODES (ASCII GRAPHICS)

KEY

CHARACTER		
ESC	33	OCTAL
	27	DECIMAL
	1B	HEX

Table 3-1
DEC Multinational Character Set (C1 and GR Codes)

8 (1 0 0 0)		9 (1 0 0 1)		10 (1 0 1 0)		11 (1 0 1 1)		12 (1 1 0 0)		13 (1 1 0 1)		14 (1 1 1 0)		15 (1 1 1 1)		COLUMN b8 b7 b6 b5 BITS b4 b3 b2 b1	ROW
	200 128 80	DCS	220 144 90		240 160 A0	°	260 176 B0	À	300 192 C0		320 208 D0	à	340 224 E0		360 240 F0	0 0 0 0	0
	201 129 81	PU1	221 145 91	¡	241 161 A1	±	261 177 B1	Á	301 193 C1	Ñ	321 209 D1	á	341 225 E1	ñ	361 241 F1	0 0 0 1	1
	202 130 82	PU2	222 146 92	¢	242 162 A2	2	262 178 B2	Â	302 194 C2	Ò	322 210 D2	â	342 226 E2	ò	362 242 F2	0 0 1 0	2
	203 131 83	STS	223 147 93	£	243 163 A3	3	263 179 B3	Ã	303 195 C3	Ó	323 211 D3	ã	343 227 E3	ó	363 243 F3	0 0 1 1	3
IND	204 132 84	CCH	224 148 94		244 164 A4		264 180 B4	Ä	304 196 C4	Ô	324 212 D4	ä	344 228 E4	ô	364 244 F4	0 1 0 0	4
NEL	205 133 85	MW	225 149 95	¥	245 165 A5	µ	265 181 B5	Å	305 197 C5	Õ	325 213 D5	å	345 229 E5	õ	365 245 F5	0 1 0 1	5
SSA	206 134 86	SPA	226 150 96		246 166 A6	¶	266 182 B6	Æ	306 198 C6	Ö	326 214 D6	æ	346 230 E6	ö	366 246 F6	0 1 1 0	6
ESA	207 135 87	EPA	227 151 97	§	247 167 A7	·	267 183 B7	Ç	307 199 C7	Œ	327 215 D7	ç	347 231 E7	œ	367 247 F7	0 1 1 1	7
HTS	210 136 88		230 152 98	¤	250 168 A8		270 184 B8	È	310 200 C8	Ø	330 216 D8	è	350 232 E8	ø	370 248 F8	1 0 0 0	8
HTJ	211 137 89		231 153 99	©	251 169 A9	1	271 185 B9	É	311 201 C9	Ù	331 217 D9	é	351 233 E9	ù	371 249 F9	1 0 0 1	9
VTS	212 138 8A		232 154 9A	a	252 170 AA	o	272 186 BA	Ê	312 202 CA	Ú	332 218 DA	ê	352 234 EA	ú	372 250 FA	1 0 1 0	10
PLD	213 139 8B	CSI	233 155 9B	«	253 171 AB	»	273 187 BB	Ë	313 203 CB	Û	333 219 DB	ë	353 235 EB	û	373 251 FB	1 0 1 1	11
PLU	214 140 8C	ST	234 156 9C		254 172 AC	¼	274 188 BC	Ì	314 204 CC	Ü	334 220 DC	ì	354 236 EC	ü	374 252 FC	1 1 0 0	12
RI	215 141 8D	OSC	235 157 9D		255 173 AD	½	275 189 BD	Í	315 205 CD	Ý	335 221 DD	í	355 237 ED	ý	375 253 FD	1 1 0 1	13
SS2	216 142 8E	PM	236 158 9E		256 174 AE		276 190 BE	Î	316 206 CE		336 222 DE	î	356 238 EE		376 254 FE	1 1 1 0	14
SS3	217 143 8F	APC	237 159 9F		257 175 AF	¿	277 191 BF	Ï	317 207 CF	ß	337 223 DF	ï	357 239 EF		377 255 FF	1 1 1 1	15

C1 CODES — GR CODES (DEC SUPPLEMENTAL GRAPHICS)

3.1 GRAPHIC CHARACTERS

When receiving GL or GR graphic character codes from an application, the Professional interprets these according to the graphic character set mapping currently being used. The default (DEC Multinational) mapping is ASCII graphics in GL and DEC Supplemental Graphics in GR. The default graphic character set mapping is established whenever you power up (boot) the Professional or reset the PRO/Communications terminal emulator. Your application can also select the default mapping by means of the reset-to-initial-state (RIS) command.

Your application can use control functions to map different sets of graphic characters into GL or GR (see "Character Sets and Selection" in Section 3.5, ANSI-Compatible Sequences). For example, you can substitute the DEC Special Graphics (VT100 Line Drawing) Set for the ASCII Graphics set. In that case, the codes formerly assigned to the ASCII graphic characters would be mapped onto the DEC Special Graphics characters. The DEC Special Graphics Set is shown in Appendix A.

Character set selection for VT52 mode is more limited (refer to Section 3.6, VT52-compatible sequences).

3.2 CONTROL CHARACTERS

Tables 3–2 and 3–3 define the actions taken by the Professional when receiving C0 and C1 control characters. The response shown is for all modes, unless stated otherwise. The Professional does not recognize all C0 or C1 characters. Those not shown in either table are ignored (no action taken). Section 3.4 defines how the Professional handles control codes that are invalid or not recognized.

Table 3–2 shows that SO (0/14) and SI (0/15) are also called LS1 and LS0, respectively. SO and SI (shift out and shift in) are the traditional ASCII names or mnemonics. LS1 and LS0 (lock shift G1 and lock shift G0) are more precise names that are useful when dealing with the variety of character set mappings possible on the Professional. LS1 and LS0 are therefore the names used in this chapter under "Character Sets and Selection" (Section 3.5).

Table 3–3 shows the equivalent 7-bit-compatible code extension for each 8-bit C1 code. The code extensions require one more byte than the C1 codes. The last part of Chapter 1 describes when to use C1 codes and when to use 7-bit code extensions. The "code extension announcers" described in Chapter 1 are also listed in this chapter, at the beginning of Section 3.5 (ANSI-compatible sequences). Only the C1 codes are documented throughout this chapter, because this is the most efficient coding convention. You can, however, always use the equivalent code extensions indicated in Table 3–3.

Table 3-2
C0 (ASCII) Control Characters Recognized by Professional

Mnemonic	*Code*	*Name*	*Action Taken*
NUL	0/0	Null	Ignored when received.
ENQ	0/5	Enquiry	No answerback message is generated. A VT102 terminal generates an answerback message, if stored.
BEL	0/7	Bell	Generates bell tone if bell is enabled through a P/OS Set-Up menu.
BS	0/8	Backspace	Moves cursor to the left one character position; if cursor is at left margin, no action occurs.
HT	0/9	Horizontal tabulation	Moves cursor to next tab stop, or to right margin if there are no more tab stops.
LF	0/10	Linefeed	Causes a linefeed or a new line operation.
VT	0/11	Vertical tabulation	Processed as LF
FF	0/12	Form feed	Processed as LF.
CR	0/13	Carriage return	Moves cursor to left margin on current line.
SO (LS1)	0/14	Shift out (Lock shift G1)	Invokes G1 character set into GL. G1 is designated by a select-character-set (SCS) sequence.
SI (LS0)	0/15	Shift in (Lock shift G0)	Invoke G0 character set into GL. G0 is designated by a select- character-set sequence (SCS).
DC1	1/1	Device Control 1	Also referred to as XON. DC1 clears DC3 (XOFF), causing the Professional to continue transmitting characters (keyboard unlocks) unless KAM mode is currently set.
DC3	1/3	Device Control 3	DC3 causes the Professional to stop transmitting characters (keyboard becomes locked for the next and subsequent keystrokes, and no reports are generated) until a DC1 control character is received. The exact response varies in the terminal emulation modes and depends on whether XON/XOFF has been enabled or disabled through a PRO/Communications Set Up menu (see Chapter 2). On a VT102 terminal, receipt of DC3 causes a buffer to store any characters typed subsequently. The keyboard becomes locked only if the buffer fills up before DC3 is cleared by DC1. On a VT102 terminal the keyboard becomes locked on a byte (versus keystroke) boundary.

Table 3-2 (Cont.)

Mnemonic	*Code*	*Name*	*Action Taken*
CAN	1/8	Cancel	If received during an escape or control sequence, cancels the sequence. No error character is displayed. On a VT102 terminal, the error character is displayed.
SUB	1/10	Substitute	If received during an escape or control sequence, cancels the sequence. Causes a reverse question mark to be displayed. On a VT102 terminal, the error character is displayed.
ESC	1/11	Escape	Processed as escape sequence introducer.
DEL	7/15	Delete	(Not a C0 character) Ignored when received

Table 3-3
C1 Control Characters Recognized by Professional

Mnemonic	*8-bit Code*	*Equivalent 7-bit Code Extension*		*Name*	*Action Taken*
IND	8/4	1/11 ESC	4/4 D	Index	Moves cursor down one line in same column. If cursor is at bottom margin, screen performs a scroll up.
NEL	8/5	1/11 ESC	4/5 E	Next line	Moves cursor to first position on next line. If cursor is at bottom margin, screen performs a scroll up.
HTS	8/8	1/11 ESC	4/8 H	Horizontal tab set	Sets one horizontal tab stop at the column where the cursor is.
RI	8/13	1/11 ESC	4/13 M	Reverse index	Moves cursor up one line in same column. If cursor is at top margin, screen performs a scroll down.
SS2	8/14	1/11 ESC	4/14 N	Single shift G2	Temporarily invokes G2 character set into GL for the next graphic character. G2 is designated by a select- character-set (SCS) sequence.
SS3	8/15	1/11 ESC	4/15 O	Single shift G3	Temporarily invokes G3 character set into GL for the next graphic character. G3 is designated by a select- character-set (SCS) sequence.

Table 3–3 (Cont.)

Mnemonic	*8-bit Code*	*Equivalent 7-bit Code Extension*		*Name*	*Action Taken*
DCS	9/0	1/11 ESC	5/0 P	Device control string	Processed as opening delimiter of a control string for device control use. Any control string introduced by DCS and terminated by ST is ignored and is not displayed on the screen, except for ReGIS (VT125 mode) control strings: They are processed in VT125 mode.
CSI	9/11	1/11 ESC	5/11 [	Control sequence introducer	Processed as control sequence introducer.
ST	9/12	1/11 ESC	5/12 \	String terminator	Processed as closing delimiter of a string opened by DCS, OSC, PM, APC.
OSC	9/13	1/11 ESC	5/13]	Operating system command	Processed as opening delimiter of a control string for operating system use. Any control string introduced by OSC and terminated by ST is ignored and is not displayed on the screen.
PM	9/14	1/11 ESC	5/14 ^	Privacy message	Processed as opening delimiter of a control string for privacy message use. Any control string introduced by PM and terminated by ST is ignored and is not displayed on the screen.
APC	9/15	1/11 ESC	5/15 _	Application program command	Processed as opening delimiter of a control string for application program use. Any control string introduced by APC and terminated by ST is ignored and is not displayed on the screen.

3.3 ESCAPE AND CONTROL SEQUENCE OVERVIEW

Escape and control sequences were introduced in Chapter 1. They are defined by ANSI X3.41-1977 and X3.64-1979.

The Professional uses ANSI mode to respond to a subset of escape and control sequences specified by ANSI. There are two types of sequences: ANSI-standardized and private. ANSI-standardized sequences have a mnemonic assigned by ANSI. Private sequences are those not presently specified by ANSI. However, they are created to comply with the extensions permitted by ANSI. Private sequences created by DIGITAL have a mnemonic that begins with "DEC".

Table 3–4 lists all ANSI-compatible sequences recognized by the Professional. They are grouped functionally. The table shows which modes of operation recognize any particular sequence. Detailed descriptions of the ANSI-compatible sequences, grouped in the same way, are in Section 3.5.

VT52 mode is available by program control as well as through an operator menu selection. When operating in VT102 mode, you can invoke VT52 mode from your application by using the DECANM command, which is listed at the end of Table 3–4. The Professional recognizes the Digital-private ANSI-incompatible VT52 escape sequences when VT52 mode is announced in your application. Table 3–5 is a summary list of the VT52-compatible sequences recognized by the Professional. They are all described in Section 3.6. VT52 mode is provided on the Professional only for compatibility with earlier software. Because future DIGITAL equipment may not accept ANSI-incompatible sequences, you should use only the ANSI-compatible sequences in all new software.

VT125 mode includes text mode (essentially VT102 mode) and graphics mode (ReGIS processing). When the Professional is being operated in VT125 mode, it will process and run ReGIS code like a VT125 terminal, with some restrictions (see Appendix D). A ReGIS application causes the Professional to operate in VT125 graphics mode by sending ReGIS "control strings" to the Professional. In 8-bit format, ReGIS control strings are introduced by the C1 control character DCS. The C1 character ST is the closing delimiter of the string. DCS is one of a general class of control string delimiters (DCS, OSC, PM, APC, ST), which are all C1 characters. Refer to Table 3–4 and Section 3.5 for information on control strings in general, and on the particular ReGIS strings recognized by the Professional. The VT125 User's Guide has ReGIS programming information.

The last part of Chapter 1 describes how to work with 7- and 8-bit application environments. For a 7-bit environment, the 7-bit code extensions you may need to use in place of 8-bit C1 characters are listed in Table 3–3. Some of the sequences your application sends may cause other codes to be returned to your application. The code extension announcers you use to control the representation of returned codes (7- or 8-bit) are described at the beginning of Section 3.5.

Section 3.4 defines how the Professional handles invalid and unrecognized control function codes.

> Note: When forming escape or control sequences, remember that it is the code that defines a sequence, not the graphic representation of the characters: characters shown in the detailed descriptions are included for readability only and presume the DEC Multinational Set mapping (ASCII in GL and DEC Supplemental Graphics in GR).

3.4 ERROR RECOVERY

Current standards do not specify the action to be performed when the Professional receives a control function with an error. Errors are incorrect parameters, invalid control functions, or control characters embedded in control function sequences. The Professional usually recovers from these errors by performing as much of the function as possible. The specific error recovery procedures are as follows.

- ☐ Unrecognized control functions are usually ignored.
- ☐ Unsupported control functions (valid control functions not recognized by the Professional in some particular mode of operation) are usually ignored in that mode but may produce unexpected results.
- ☐ If a C0 control character other than ESC (1/11) is received within a sequence, the Professional performs the function of the control character, followed by the function of the sequence.
- ☐ If ESC is received within a sequence, the previous function is aborted and the Professional assumes it is receiving a new escape sequence initiated by the ESC character.
- ☐ If a C1 control character is received within a sequence, the previous function is aborted and the Professional performs the function of the control character.
- ☐ If CAN (1/8) or SUB (1/10) is received within a sequence, the sequence is aborted. SUB causes a reverse question mark (substitute character) to be displayed, followed by the characters in the sequence received after SUB. CAN does not display any error character but does display the remaining characters, like SUB.

Table 3-4
Summary of ANSI-Compatible Sequences

Name and Mnemonic (by Functional Group)	See Page	Used In: Native Mode	Used In: Terminal Emulation Modes	Used In: VT102 Terminal
Code Extension Announcers	3-12			
Return C1 codes to application		yes	yes	—
Return 7-bit code extensions to application		yes	yes	—
Set-Up Characteristics and Mode Selection	3-13			
Set Mode (SM) and reset Mode (RM)		yes	yes	yes
Scrolling	3-18			
Scroll Mode (DECSCLM)		yes	yes	yes
Scrolling Region	3-18			
Set Top and Bottom Margins (DECSTBM)		yes	yes	yes
Origin	3-19			
Origin Mode (DECOM)		yes	yes	yes
Cursor Positioning	3-19			
Cursor Up (CUU)		yes	yes	yes
Cursor Down (CUD)		yes	yes	yes
Cursor Forward (CUF)		yes	yes	yes
Cursor Backward (CUB)		yes	yes	yes
Cursor Position (CUP)		yes	yes	yes
Horizontal and Vertical Position (HVP)		yes	yes	yes
Index (IND)		yes	yes	yes
Reverse Index (RI)		yes	yes	yes
Next Line (NEL)		yes	yes	yes
Save Cursor (DECSC)		yes	yes	yes
Restore Cursor (DECRC)		yes	yes	yes
Cursor Characteristics	3-22			
Cursor Enable Mode (DECCEM)		yes	yes	—
Columns per Line	3-22			
Column Mode (DECCOLM)		yes	yes	yes
Auto Wrap	3-23			
Auto Wrap Mode (DECAWM)		yes	yes	yes
Screen Background	3-23			
Screen Mode (DECSCNM)		yes	yes	yes
Linefeed/New Line	3-24			
Linefeed/New Line Mode (LNM)		yes	yes	yes
Keyboard Action	3-25			
Keyboard Action Mode (KAM)		yes	yes	yes
Auto Repeat	3-26			
Auto Repeat Mode (DECARM)		yes	yes	yes
Local Echo	3-26			
Send-Receive Mode (SRM)		—	yes	yes
Cursor Key Character Selection	3-27			
Cursor Key Mode (DECCKM)		yes	yes	yes
Auxiliary Keypad Character Selection	3-28			
Keypad Numeric Mode (DECKPNM)		yes	yes	yes
Keypad Application Mode (DECKPAM)		yes	yes	yes
Character Sets and Selection	3-30			
Select (designate) Character Set (SCS)		yes	yes	yes
Invoke Character Set				
Lock Shift G0 (LS0 or SI)		yes	yes	yes
Lock Shift G1 (LS1 or SO)		yes	yes	yes
Lock Shift G1, right (LS1R)		yes	yes	—
Lock Shift G2 (LS2)		yes	yes	—

* Control strings are recognized but ignored in native mode and, except for ReGIS strings, in terminal emulation modes.

Table 3-4 (Cont.)

Name and Mnemonic (by Functional Group)	See Page	Used In: Native Mode	Used In: Terminal Emulation Modes	Used In: VT102 Terminal
Lock Shift G2, right (LS2R)		yes	yes	—
Lock Shift G3 (LS3)		yes	yes	—
Lock Shift G3, right (LS3R)		yes	yes	—
Single Shift G2 (SS2)		yes	yes	yes
Single Shift G3 (SS3)		yes	yes	yes
Character Attributes	3-34			
Select Graphic Rendition (SGR)		yes	yes	yes
Line Attributes	3-36			
Double-height Line (DECDHL)		yes	yes	yes
Single-width Line (DECSWL)		yes	yes	yes
Double-width Line (DECDWL)		yes	yes	yes
Tab Stops	3-37			
Horizontal Tab Set (HTS)		yes	yes	yes
Tabulation Clear (TBC)		yes	yes	yes
Erasing	3-37			
Erase in Line (EL)		yes	yes	yes
Erase in Display (ED)		yes	yes	yes
Editing	3-38			
Delete Character (DCH)		yes	yes	yes
Insert Line (IL)		yes	yes	yes
Delete Line (DL)		yes	yes	yes
Inserting and Replacing Characters	3-39			
Insertion-Replacement Mode (IRM)		yes	yes	yes
Printing	3-40			
Media Copy (MC)		yes	yes	yes
Printer Extent	3-41			
Printer Extent Mode (DECPEX)		yes	yes	yes
Print Termination Character	3-42			
Printer Form Feed Mode (DECPFF)		yes	yes	yes
Print Screen (Text or Graphics Mode)	3-42			
Hardcopy (DECHCP)		yes	yes	—
Reports	3-43			
Device Status Report (DSR)		yes	yes	yes
Cursor Position Report (CPR)		yes	yes	yes
Device Attributes (DA)		yes	yes	yes
Identify Terminal (DECID)		yes	yes	yes
Reset	3-46			
Reset to Initial State (RIS)		yes	yes	yes
Interlace	3-46			
Interlace Mode (DECINLM)		yes	yes	—
Tests and Adjustments	3-48			
Screen Alignment Display (DECALN)		yes	yes	yes
Invoke Confidence Test (DECTST)		—	—	yes
Keyboard Indicator				
Load LED (DECLL)		—	—	yes
Control Strings	3-48			
String Delimiters (DCS,OSC,PM,APC,ST)		yes*	yes*	—
VT125 Graphics Mode (Regis)	3-50			
DCS p		—	yes	—
DCS 1 p		—	yes	—
ANSI/VT52 Compatibility	3-50			
ANSI/VT52 Mode (DECANM)		—	yes	yes

* Control strings are recognized but ignored in native mode and, except for ReGIS strings, in terminal emulation modes.

Table 3-5
Summary of VT52-Compatible Sequences

3.5 ANSI-COMPATIBLE SEQUENCES

ANSI-compatible sequences meet ANSI standards X3.64-1979 and X3.41-1974. This section describes the ANSI control functions recognized by the Professional in native mode, Professional mode, VT102 mode, and VT125 mode.

> Note: Default states for various modes are described throughout this section. Only the following conditions are documented: when you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence. No P/OS default states (such as resetting between applications) are described.

3.5.1 Code Extension Announcers

You use code extension announcers in the application to control the representation of 8-bit (C1) control codes returned to the application by the Professional (see Chapter 1 for more information on working with 7-and 8-bit environments).

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, the Professional converts all C1 control codes returned subsequently to the application to their equivalent 7-bit code extensions.

Return 8-Bit (C1) Control Codes to Application

1/11	2/0	4/6
ESC	SP	G

Lets the Terminal Subsystem return C1 codes to the application without converting them to their equivalent 7-bit code extensions.

Return 7-Bit Code Extensions to Application

1/11	2/0	4/7
ESC	SP	F

Causes the Terminal Subsystem to convert all 8-bit (C1) control codes to their equivalent 7-bit code extensions before returning them to the application.

3.5.2 Set-Up Characteristics and Mode Selection

The Professional has two groups of Set-Up menus (P/OS and PRO/Communications) that the operator uses to manually change certain default operating characteristics.

The P/OS Set-Up menus are used to control operations like keyclick (on/off) or smooth/jump scroll. P/OS Set-Up menu selections set defaults for applications running in terminal emulation modes as well as those running in native mode.

The PRO/Communications Set-Up menus are part of the PRO/Communications application. They are used to set communications parameters such as enabling auto XON/XOFF, or 7- or 8-bit character codes, or local echo. PRO/Communications Set-Up menu selections set defaults for applications running in terminal emulation modes.

Use of the Set-Up menus is documented in the *User's Guide for (Diskette/Hard Disk) System* for P/OS menus, and the *PRO/Communications Manual* for PRO/communications menus.

In your application, you can use "mode selection sequences" to change some of these P/OS and communications characteristics. The sequences cause the Professional to operate in one of two possible ways (for example, normal/reverse screen background).

Table 3–6 lists the Professional operating characteristics controllable either through a Set-Up menu or through a mode selection sequence (not all PRO/Communications Set-Up characteristics are listed, only those discussed in this manual). Some characteristics can be controlled through both a menu selection and a mode selection sequence. In that case, the menu selection is the default setting. The Professional uses the selected mode until you change it, either manually through a Set-Up menu or through your application.

Table 3-6 shows that two modes are changed by means of dedicated sequences (set/clear tabs, select auxiliary keypad mode). All the other modes, however, are based on two general-purpose sequences: set mode (SM) and reset mode (RM). These have variable parameters so you can apply them to different functions. The general formats of SM and RM are shown below.

SM — Set Mode

9/11		3/11		3/11		6/8
CSI	Ps	;	...	;	Ps	h

Sets one or more modes specified by selective parameters (Ps) in the parameter string.

RM — Reset Mode

9/11		3/11		3/11		6/12
CSI	Ps	;	...	;	Ps	l

Resets one or more modes specified by selective parameters (Ps) in the parameter string.

You can set and reset one or more modes in one sequence. Ps is a variable, ASCII coded, numeric parameter associated with a particular mode. When you set or reset more than one mode in the same sequence, you must separate the parameters with a semicolon code (3/11). You can set or reset up to 16 modes in one sequence. Additional parameters are ignored. Each parameter may have a value in the range of zero (0) to 65535. Any value over this range is treated as the maximum value. Also, within the context of a particular command, values which exceed the maximum values for the command (for example, 24 or 80) are treated as the contextual maximum.

Table 3-6
Set-Up Characteristics and Mode Selection Sequences

Set-Up Characteristic or Programmable Mode	*Change from Application*	*Set Default Value from*	
		P/OS Set-Up Menus	*PRO/Communications Set-Up Menus*
Autorepeat on/off	Yes (DECARM)	Yes	
Bell on/off		Yes	
Current directory		Yes	
Current keyboard		Yes	
Office/DP keyboard mode		Yes	
Columns per line 80/132	Yes (DECCOLM)	Yes	
Keyclick on/off		Yes	
Shift lock/Caps lock		Yes	
Monitor monochrome/color		Yes	
Scroll jump/smooth	Yes (DECSCLM)	Yes	
Set tabs[1]	Yes (HTS/TBC)	Yes	
Video normal/reverse	Yes (DECSCNM)	Yes	
Cursor block/underline		Yes	
Margin bell on/off		Yes	
Printer LA100/LA50/LQP02		Yes	
Set date and time (clock)		Yes	
Origin mode	Yes (DECOM)		
Cursor enable mode	Yes (DECCEM)		
Auto wrap mode	Yes (DECAWM)		
Linefeed/new line mode	Yes (LNM)		Yes
Keyboard action mode	Yes (KAM)		
Send-receive mode[2] (local echo)	Yes (SRM)		Yes

1. These modes are changed using dedicated sequences, not the set mode (SM) and reset mode (RM) sequences. See Tab Stops, and Auxiliary Keypad Character Selection for more information.
2. Send-receive mode is used only in terminal emulation modes.
3. Resetting the terminal emulator invokes the reset-to-intitial-state (RIS) sequence (see note below).

Note: The reset-to-initial-state (RIS) sequence resets programmable modes and keyboard states to default values. If a Set-Up menu also controls that mode, the menu selection determines the default value. Refer to Table 3-12 for a list of the default states selected by the RIS sequence.

Table 3-6 (Cont.)

Set-Up Characteristic or Programmable Mode	*Change from Application*	*Set Default Value from*	
		P/OS Set-Up Menu	*PRO/Communications Set-Up Menus*
Cursor key mode	Yes (DECCKM)		
Auxiliary keypad mode[1]	Yes (DECKPAM/ DECKPNM)		
Insertion-replacement mode	Yes (IRM)		
Print termination character	Yes (DECPFF)		
Printer extent mode	Yes (DECPEX)		
Interlace mode	Yes (DECINLM)		
Auto XON/XOFF on/off			Yes
Bits per character (7 or 8)			Yes
Reset terminal emulator[3]			Yes

1. These modes are changed using dedicated sequences, not the set mode (SM) and reset mode (RM) sequences. See Tab Stops, and Auxiliary Keypad Character Selection for more information.
2. Send-receive mode is used only in terminal emulation modes.
3. Resetting the terminal emulator invokes the reset-to-intitial-state (RIS) sequence (see note below).

Note: The reset-to-initial-state (RIS) sequence resets programmable modes and keyboard states to default values. If a Set-Up menu also controls that mode, the menu selection determines the default value. Refer to Table 3-12 for a list of the default states selected by the RIS sequence.

The modes affected by SM and RM are listed in Tables 3-7 and 3-8, along with the parameter values (decimal and column/row format). Table 3-7 lists the ANSI-standardized modes. Table 3-8 lists the DEC private modes. When you specify private sequences, the first character in the first parameter string must be a question mark code (3/15). Then all parameters in the sequence will be interpreted as DEC-private parameters. Because standardized and private sequences are interpreted differently, when setting several modes within a single sequence, you must choose them to be either all standardized modes or all private modes.

The following example shows the use of the question mark (used with DEC-private sequences) and semicolon (used with multiple parameters). This sequence sets both column and scroll modes.

9/11	3/15	3/3	3/11	3/3	6/8
CSI	?	3	;	4	h

Each of the modes listed in Tables 3-7 and 3-8 is described in detail in the rest of this chapter (all sequences are grouped functionally as listed in Table 3-4).

Table 3-7
ANSI-Standardized Modes

Name	*Mnemonic*	*Parameter (Ps)*	
Error (ignored)		0	(3/0)
Keyboard Action	KAM	2	(3/2)
Insertion-replacement	IRM	4	(3/4)
Send-receive	SRM	12	(3/1 3/2)
Linefeed/New Line	LNM	20	(3/2 3/0)

Table 3-8
ANSI-Compatible DEC Private Modes

Name	*Mnemonic*	*Parameter (Ps)*	
Error (ignored)		0	(3/0)
Cursor Key	DECCKM	1	(3/1)
Column	DECCOLM	3	(3/3)
Scroll	DECSCLM	4	(3/4)
Screen	DECSCNM	5	(3/5)
Origin	DECOM	6	(3/6)
Auto Wrap	DECAWM	7	(3/7)
Auto Repeat	DECARM	8	(3/8)
Interlace	DECINLM	9	(3/9)
Printer Form Feed	DECPFF	18	(3/1 3/8)
Printer Extent	DECPEX	19	(3/1 3/9)
Cursor Enable	DECCEM	25	(3/2 3/5)

3.5.3 Scrolling

Scrolling is the upward or downward movement of existing lines on the screen. This makes room for more display lines at either the top or bottom of the scrolling region. There are two methods of scrolling, jump scroll and smooth scroll. Select the type of scrolling by using the following sequences.

On the Professional, smooth scroll is possible only for the entire screen. Smooth split-screen scroll is not possible, in which case the Professional uses jump split-screen scroll instead. Full-screen scroll affects all bitmap planes when the extended bitmap option is being used. Split-screen scroll affects only the plane that is used for text mode. No other video activity occurs while smooth scroll is in progress. This includes blinking of the cursor or other characters.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, scroll mode defaults to the state selected in the P/OS Set-Up menu.

DECSCLM — Scroll Mode

9/11	3/15	3/4	6/8
CSI	?	4	h

Set selects smooth scroll. Smooth scroll lets the Professional add no more than 6 lines per second to the screen.

9/11	3/15	3/4	6/12
CSI	?	4	l

Reset selects jump scroll. Jump scroll lets the Professional add lines to the screen as fast as possible.

3.5.4 Scrolling Region

The scrolling region is the area of the screen between the top and bottom margins. The margins determine which screen lines move during scrolling. Characters added outside the scrolling region do not cause the screen to scroll. The minimum size of the scrolling region is one line. Therefore, the line number of the bottom margin must be greater than or equal to the line number of the top margin. Origin mode affects the line numbers.

After the margins are selected, the cursor moves to the home position. Origin mode affects the home position. Select the top and bottom margins of the scrolling region by using the following sequence.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, the scrolling region becomes the full screen.

DECSTBM — Set Top and Bottom Margins

9/11		3/11		7/2
CSI	Pt	;	Pb	r

Selects top and bottom margins, defining the scrolling region. Pt is the (ASCII coded) line number of the first line in the scrolling region. Pb is the line number of the bottom line. If either Pt or Pb is not selected, they default to top and bottom respectively. Lines are counted from "1".

3.5.5 Origin

Origin mode determines line numbering and whether the cursor can move outside the scrolling region (the area between the top and bottom margins).

Lines on the screen are numbered according to the location of the home position. Home position is always line 1, column 1. The cursor moves to the new home position whenever origin mode is selected. Select origin mode by using the following sequences.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, origin mode resets.

DECOM — Origin Mode

9/11	3/15	3/6	6/8
CSI	?	6	h

Set selects the home position in the scrolling region. Line numbers start at the top margin of the scrolling region (relative line numbers). The cursor cannot move out of the scrolling region.

9/11	3/15	3/6	6/12
CSI	?	6	l

Reset selects home position in the upper-left corner of the screen. Line numbers are independent of the scrolling region (absolute line numbers). Use the CUP and HVP sequences to move the cursor out of the scrolling region.

3.5.6 Cursor Positioning

The cursor indicates the active screen position where the next character will appear. A number of operations implicitly affect cursor positioning. In addition, your application can control cursor movement by means of the following sequences.

> Note: Pn is a variable, ASCII coded, numeric parameter. If you select no parameter or a parameter value of 0, the Professional assumes the parameter equals 1.

CUU — Cursor Up

9/11		4/1
CSI	Pn	A

Moves the cursor up Pn lines in the same column. The cursor stops at the top margin.

CUD — Cursor Down

9/11		4/2
CSI	Pn	B

Moves the cursor down Pn lines in the same column. The cursor stops at the bottom margin.

CUF — Cursor Forward

9/11		4/3
CSI	Pn	C

Moves the cursor right Pn columns. The cursor stops at the right margin.

CUB — Cursor Backward

9/11		4/4
CSI	Pn	D

Moves the cursor left Pn columns. The cursor stops at the left margin.

CUP — Cursor Position

9/11		3/11		4/8
CSI	Pl	;	Pc	H

Moves the cursor to line Pl, column Pc. The numbering of the lines and columns depends on the state (set/reset) of origin mode (DECOM).

Note: CUP operates the same as the horizontal and vertical position (HVP) sequence.

CUP — Cursor Position (Home)

9/11	4/8
CSI	H

Moves the cursor to the home position, selected by origin mode (DECOM).

HVP — Horizontal and Vertical Position

9/11		3/11		6/6
CSI	Pl	;	Pc	f

Moves the cursor to line Pl, column Pc. The numbering of the lines and columns depends on the state (set/reset) of origin mode (DECOM).

Note: HVP operates the same as the cursor postion (CUP) sequence.

HVP — Horizontal and Vertical Position (Home)

9/11	6/6
CSI	f

The cursor moves to the home position selected by origin mode (DECOM).

IND — Index

IND is an 8-bit control character (8/4). It can also be expressed as an escape sequence for a 7-bit environment:

1/11	4/4
ESC	D

Moves the cursor down one line in the same column. If the cursor is at the bottom margin, the screen performs a scroll-up.

RI — Reverse Index

RI is an 8-bit control character (8/13). It can also be expressed as an escape sequence for a 7-bit environment:

1/11	4/13
ESC	M

Moves the cursor up one line in the same column. If the cursor is at the top margin, the screen performs a scroll-down.

NEL — Next Line

NEL is an 8-bit control character (8/5). It can also be expressed as an escape sequence for a 7-bit environment:

1/11	4/5
ESC	E

Moves the cursor to the first position on the next line. If the cursor is at the bottom margin, the screen performs a scroll-up.

DECSC — Save Cursor

1/11	3/7
ESC	7

Saves the cursor position, origin mode selection, character attribute (graphic rendition), and character set. See Restore Cursor.

DECRC — Restore Cursor

1/11	3/8
ESC	8

Restore the cursor position, character attribute (graphic rendition), character set, and origin mode selections saved since the last time the Professional was powered up (booted), or the terminal emulator was reset, or a reset-to-initial-state (RIS) sequence was issued. If none of these characteristics were saved,

- ☐ the cursor moves to home position,
- ☐ origin mode is reset (absolute line numbers, independent of scrolling region),
- ☐ no character attributes are assigned, and
- ☐ the default character set mapping is established: ASCII in G0, DEC Special Graphics in G1, DEC Supplemental Graphics in G2, ASCII in G3, G0 mapped to GL, G2 mapped to GR.

3.5.7 Cursor characteristics

This sequence affects the cursor in both text mode and graphics mode. A graphics-mode pixel-oriented cursor (for example, a cross hair) appears on the screen if the most recent data sent to the Professional was in graphics mode. A text-mode character-cell-oriented cursor appears otherwise. The period for cursor blinking is one second (2/3 on, 1/3 off).

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, cursor enable mode is set.

DECCEM — Cursor Enable Mode

9/11	3/15	3/2	3/5	6/8
CSI	?	2	5	h

Set causes the cursor to appear.

9/11	3/15	3/2	3/5	6/12
CSI	?	2	5	l

Reset causes the cursor to disappear.

3.5.8 Columns per Line

Column mode selects the number of columns in each display line, 80 or 132. With either selection, the screen can display 24 lines. Select the number of columns per line by using the following sequences.

> Note: When you change the number of columns per line, the screen is erased and the cursor moves to the home position. This also sets the scrolling region for full screen (24 lines).

Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, column mode defaults to the state selected in the P/OS Set-Up menu.

DECCOLM — Column Mode

9/11	3/15	3/3	6/8
CSI	?	3	h

Set selects 132 columns per line.

9/11	3/15	3/3	6/12
CSI	?	3	l

Reset selects 80 columns per line.

3.5.9 Auto Wrap

Auto wrap mode selects where a received character will appear when the cursor is at the right margin. Select auto wrap by using the following sequences.

Note: Regardless of the auto wrap Set-Up selection, the tab character never moves the cursor to the next line.

Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, auto wrap mode is set.

DECAWM — Auto Wrap Mode

9/11	3/15	3/7	6/8
CSI	?	7	h

Set selects auto wrap. Any display characters received when the cursor is at the right margin appear on the next line. The display scrolls up if the cursor is at the bottom of the scrolling region.

9/11	3/15	3/7	6/12
CSI	?	7	l

Reset turns auto wrap off. Display characters received when the cursor is at the right margin replace the previously displayed character.

3.5.10 Screen Background

Screen mode selects either light (reverse) or dark display background on the screen. Select screen mode by using the following sequences.

Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, screen mode defaults to the state selected in the P/OS Set-Up menu.

DECSCNM — Screen Mode

9/11	3/15	3/5	6/8
CSI	?	5	h

Set selects reverse screen, a light screen background with dark characters.

9/11	3/15	3/5	6/12
CSI	?	5	l

Reset selects normal screen, a dark screen background with light characters.

3.5.11 Linefeed/New Line

Linefeed/new line mode selects the control character(s) transmitted to the application by the **RETURN** and **ENTER** keys. **ENTER** transmits the same code as **RETURN** only when the auxiliary keypad is in keypad numeric mode (DECKPNM).

Linefeed/new line also selects the action taken by the Professional when receiving linefeed (LF), form feed (FF), or vertical tab (VT) codes. These three codes are always processed identically.

Table 3-9 provides a summary of the code processing.

You set and reset linefeed/new line mode by using the following sequences. In the terminal emulation modes, you can also manually set and reset linefeed/new line mode through a PRO/Communications Set-Up menu.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, linefeed/new line mode resets. In the terminal emulation modes, this is the default setting indicated on the PRO/Communications Set-Up menu.

LNM — Linefeed/New Line Mode

9/11	3/2	3/0	6/8
CSI	2	0	h

Set causes a received LF, FF, or VT code to move the cursor to the first column of the next line. **RETURN** transmits both a CR and a LF code. This selection is also called "new line option".

9/11	3/2	3/0	6/12
CSI	2	0	l

Reset causes a received LF, FF, or VT code to move the cursor to the next line in the current column. **RETURN** transmits a CR code only.

Table 3-9
Linefeed/New Line Mode Characteristics

Mode Setting	*Key Pressed*	*Codes Generated*	
Reset	Return/Enter	CR	(0/13)
Reset	LF*	LF	(0/10)
Set	Return/Enter	CR LF	(0/13 0/10)
Set	LF*	LF	(0/10)

Mode Setting	*Code Received From Application*	*Action Taken*
Reset	CR	Cursor moves to left margin of current line
Reset	LF, FF, or VT	Cursor moves to next line but stays in same column
Set	CR	Cursor moves to left margin of current line
Set	LF, FF, or VT	Cursor moves to left margin of next line

* There is no **LINE FEED** key on the Professional. The LF code (0/10) is generated as follows: In VT102, VT125, or VT52 modes, press top-row function key **F13 (LF)**. In Professional or native mode, press **CTRL**/J.

3.5.12 Keyboard Action

Keyboard action mode lets your application lock and unlock the keyboard. When the keyboard is locked, no codes can be transmitted from the keyboard to the application (and/or to the host system in the terminal emulation modes). To alert the user, whenever the keyboard is locked, the Wait indicator at the top of the keyboard is turned on and the keyclick feature is disabled. You should refer to Chapter 2 for complete details on locking and unlocking the keyboard, because the control characters DC3 and DC1 are also used to suppress and re-enable the transmission of data. Select keyboard action mode by using the following sequences.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, keyboard action mode resets.

KAM — Keyboard Action Mode

9/11	3/2	6/8
CSI	2	h

Set locks the keyboard for the next and subsequent keystrokes.

9/11	3/2	6/12
CSI	2	l

Reset unlocks the keyboard, unless it is currently locked by DC3.

3.5.13 Auto Repeat

Auto repeat mode selects automatic key repeating. A key pressed for more than 0.5 second automatically repeats the transmission of the character (more than 0.3 second for Delete and cursor control keys). The following keys never auto-repeat: all top row function keys; all editing keypad keys except for the cursor keys; **RETURN**, **COMPOSE CHARACTER**, **LOCK**, **SHIFT**, **CTRL**. Also, when the **CTRL** key is pressed, auto-repeating is temporarily inhibited when a "control-able" key is pressed simultaneously. Select auto repeat mode by using the following sequences.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, auto repeat mode defaults to the state selected in the P/OS Set-Up menu.

DECARM — Auto Repeat Mode

9/11	3/15	3/8	6/8
CSI	?	8	h

Set selects auto repeat. A key pressed for more than one-half second automatically repeats.

9/11	3/15	3/8	6/12
CSI	?	8	l

Reset turns off auto repeat. Keys do not automatically repeat.

3.5.14 Local Echo (Keyboard Send-Receive)

Send-receive mode, available only in the terminal emulation modes, turns local echo on or off. When send-receive mode is reset (local echo on), every character transmitted by the Professional automatically appears on the screen. Therefore, the host terminal driver or application does not have to transmit (echo) the character back to the Professional for display. When send-receive mode is set (local echo off), the Professional transmits characters only to the application. The host terminal driver or application must echo the characters back to the Professional for display.

You set and reset send-receive mode by using the following sequences. You can also manually set and reset send-receive mode through a PRO/Communications Set-Up menu.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, send-receive mode is set. This is the default setting indicated on the PRO/Communications Set-Up menu.

SRM — Send-Receive Mode

9/11	3/1	3/2	6/8
CSI	1	2	h

Set turns off (disables) local echo. When the Professional transmits characters to the host application, the host must echo characters back to the Professional for display on the screen.

9/11	3/1	3/2	6/12
CSI	1	2	l

Reset selects (enables) local echo. Characters transmitted to the application automatically appear on the screen.

3.5.15 Cursor Key Character Selection

The characters generated by the cursor keys on the editing keypad depend on the state of cursor-key mode and on whether the auxiliary keypad, at the right of the keyboard, is being used in keypad application mode (DECKPAM) or keypad numeric mode (DECKPNM). In keypad application mode, cursor key mode selects the set of codes generated by the cursor keys. In keypad numeric mode, the current setting of cursor key mode is ignored. See Table 3-10 for the codes generated by the cursor keys. Select cursor key mode by using the following sequences.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, cursor key mode resets.

DECCKM — Cursor Key Mode

9/11	3/15	3/1	6/8
CSI	?	1	h

Set causes the cursor keys to generate control functions if keypad application mode (DECKPAM) was selected.

9/11	3/15	3/1	6/12
CSI	?	1	l

Reset causes the cursor keys to generate ANSI cursor control sequences.

Table 3-10
Codes Generated by Cursor Control Keys (ANSI mode)

Key	*Cursor Key Mode Reset (Cursor Control)*		*Cursor Key Mode Set (Application)*	
	9/11	4/1	8/15	4/1
↑	CSI	A	SS3	A
	9/11	4/2	8/15	4/2
↓	CSI	B	SS3	B
	9/11	4/3	8/15	4/3
→	CSI	C	SS3	C
	9/11	4/4	8/15	4/4
←	CSI	D	SS3	D

3.5.16 Auxiliary Keypad Character Selection

The keys on the auxiliary keypad, at the right of the keyboard, generate either numeric characters or control (application) functions. Selecting keypad numeric mode (DECKPNM) or keypad application mode (DECKPAM) determines the type of characters. The keypad mode may also affect the cursor keys. See Cursor Key Character Selection for more information. The program function (PF) keys transmit the same characters to the application regardless of the keypad character selection. See Table 3–11 for the character codes generated by the auxiliary keypad keys. Select the auxiliary keypad mode by using the following sequences.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, the Professional selects keypad numeric mode.

DECKPAM — Keypad Application Mode

1/11	3/13
ESC	=

Selects keypad application mode. The auxiliary keypad keys generate control functions. Cursor key mode (DECCKM) selects the type of character codes generated by the cursor keys on the editing keypad.

DECKPNM — Keypad Numeric Mode

1/11	3/14
ESC	>

Selects keypad numeric mode. The auxiliary keypad generates characters that match the numeric, comma, period, and minus sign keys on the main keyboard. The cursor keys transmit ANSI cursor control sequences.

Table 3-11
Codes Generated by Auxiliary Keypad Keys (ANSI Mode)

Key	*Keypad Numeric Code*	*Keypad Application Mode*
	3/0	8/15 7/0
0	0	SS3 p
	3/1	8/15 7/1
1	1	SS3 q
	3/2	8/15 7/2
2	2	SS3 r
	3/3	8/15 7/3
3	3	SS3 s
	3/4	8/15 7/4
4	4	SS3 t
	3/5	8/15 7/5
5	5	SS3 u
	3/6	8/15 7/6
6	6	SS3 v
	3/7	8/15 7/7
7	7	SS3 w
	3/8	8/15 7/8
8	8	SS3 x
	3/9	8/15 7/9
9	9	SS3 y
	2/13	8/15 6/13
–	– (minus)	SS3 m
	2/12	8/15 6/12
,	, (comma)	SS3 l
	2/14	8/15 6/14
.	. (period)	SS3 n
	0/13	8/15 4/13
Enter*	CR	SS3 M
or	0/13 0/10	
	CR LF	

*In keypad numeric mode, **ENTER** generates the same codes as **RETURN**. You can change the code generated by **RETURN** with the Linefeed/New Line Mode. When reset, the Linefeed/New Line Mode causes **RETURN** to generate a single control character (CR). When set, the mode causes **RETURN** to generate two control characters (CR,LF).

Table 3-11 (Cont.)

Key	*Keypad Numeric Code*		*Keypad Application Mode*	
	8/15	5/0	8/15	5/0
PF1	SS3	P	SS3	P
	8/15	5/1	8/15	5/1
PF2	SS3	Q	SS3	Q
	8/15	5/2	8/15	5/2
PF3	SS3	R	SS3	R
	8/15	5/3	8/15	5/3
PF4	SS3	S	SS3	S

*In keypad numeric mode, **ENTER** generates the same codes as **RETURN**. You can change the code generated by **RETURN** with the Linefeed/New Line Mode. When reset, the Linefeed/New Line Mode causes **RETURN** to generate a single control character (CR). When set, the mode causes **RETURN** to generate two control characters (CR,LF).

3.5.17 Character Sets and Selection

Character encoding in the Professional was introduced in Chapter 1. The control functions you need to select different graphic character sets are described below. Differences with a VT102 terminal are pointed out where they may affect sofware compatibility between the Professional and VT102-type terminals.

The Professional's graphic repertoire consists of the following three graphic sets:

- ☐ ASCII Graphics
- ☐ DEC Supplemental Graphics
- ☐ DEC Special Graphics

In the most general case, you select character sets as follows.

First, using SCS sequences, you "designate" each of the above three graphic sets to one of the Professional's four character sets — G0, G1, G2, G3. This makes the graphic sets available "on call" for your program. To actually map any one of these sets into GL or GR you must then "invoke" any of G0 through G3 into GL or GR by using lock shifts (LS0, LS1, LS2, LS3, LS1R, LS2R, LS3R) or single shifts (SS2, SS3).

Character sets remain designated until the Professional receives another SCS sequence. All lock shifts remain active until the Professional receives another lock shift or single shift. All single shifts remain active for only the next single graphic character.

You don't need to select character sets in this manner every time you use the Professional because there is a default (DEC Multinational) mapping: ASCII Graphics in GL and DEC Supplemental Graphics in GR. The default graphic character set mapping is reset whenever you power up (boot) the Professional or reset the PRO/Communications terminal emulator. Your application can also select the default mapping by means of the reset-to-initial-state (RIS) sequence (this is a good general practice, because you should not rely on default states). Under these conditions the Professional's graphic character sets are designated and invoked automatically as follows:

Designate	*as*	*and invoke*	*into*
ASCII Graphics	G0	G0	GL
DEC Special Graphics	G1		
DEC Suppl. Graphics	G2	G2	GR

There is no default designation of G3 (unspecified).

The default designation of DEC Special Graphics in G1 lets you switch from ASCII Graphics to DEC Special Graphics in GL by using LS1 (SO) and LS0 (SI). LS1 invokes G1 (DEC Special Graphics) into GL, and LS0 invokes G0 (ASCII Graphics) into GL.

For comparison, the conventions for selecting character sets on a VT102 terminal are summarized below.

1. The VT102 terminal's graphic repertoire consists of the following graphic sets: ASCII, U.K., and DEC Special Graphics.
2. On a VT102 terminal, the 7-bit ASCII or U.K. set is the default set, depending on the Set-Up selection.
3. A VT102 terminal supports only the LS0 (SI) and LS1 (SO) lock shifts. Like the Professional, a VT102 terminal support the SS2 and SS3 single shifts.
4. On a VT102 terminal, G0 and G1 are the only sets you can designate. You then use SI to invoke G0 into GL and SO to invoke G1 into GL. G2 and G3 can not be designated by program control on a VT102 terminal. A VT102 terminal always interprets G2 and G3 as the default graphic character sets. They are selected in Set Up. You can invoke G2 and G3 for only one character at a time, using SS2 and SS3, after which you return to the previous character set. Software that relies on the VT102's "permanent" designation of G2 and G3 may be incompatible with the Professional.

   Note: Certain character set selections, although feasible on the Professional, may cause incompatibility with DEC software. To avoid potential problems, do not invoke G0 or G1 into GR, and do not invoke G2 or G3 into GL. The specific selections to avoid are indicated in the descriptions of how to invoke character sets.

All control functions for designating and invoking graphic character sets on the Professional are described below.

SCS — Select (Designate) Character Set

Use the following SCS sequences to designate character sets.

1/11	2/8	4/2
ESC	(	B

Designate ASCII graphic set as G0 (default).

1/11	2/9	4/2
ESC	)	B

Designate ASCII graphic set as G1.

1/11	2/10	4/2
ESC	*	B

Designate ASCII graphic set as G2.

1/11	2/11	4/2
ESC	+	B

Designate ASCII graphic set as G3.

1/11	2/8	3/12
ESC	(	<

Designate DEC Supplemental Graphics set as G0.

1/11	2/9	3/12
ESC	)	<

Designate DEC Supplemental Graphics set as G1.

1/11	2/10	3/12
ESC	*	<

Designate DEC Supplemental Graphics set as G2 (default).

1/11	2/10	3/12
ESC	+	<

Designate DEC Supplemental Graphics set as G3.

1/11	2/8	3/0
ESC	(	0

Designate DEC Special Graphics set as G0.

1/11	2/9	3/0
ESC	)	0

Designate DEC Special Graphics set as G1 (default).

1/11	2/10	3/0
ESC	*	0

Designate DEC Special Graphics set as G2.

1/11	2/11	3/0
ESC	+	0

Designate DEC Special Graphics set as G3.

Invoke Character Set Using Lock Shifts

Once you have designated your character sets, you can invoke G0, G1, G2, or G3 into GL or GR, by using the lock shift control functions described below.

LS0 — Lock Shift G0

0/15

Invoke G0 character set into GL (default). Also known as Shift In (SI).

LS1 — Lock Shift G1

0/14

Invoke G1 character set into GL. Also known as Shift Out (SO).

LS1R — Lock Shift G1, Right

1/11	7/14
ESC	~

Invoke G1 character set into GR. Do not use this sequence as it may create software compatibility problems.

LS2 — Lock Shift G2

1/11	6/14
ESC	n

Invoke G2 character set into GL. Do not use this sequence as it may create software compatibility problems.

LS2R — Lock Shift G2, Right

1/11	7/13
ESC	_

Invoke G2 character set into GR (default).

LS3 — Lock Shift G3

LS3 — Lock Shift G3

1/11	6/15
ESC	o

Invoke G3 character set into GL. Do not use this sequence as it may create software compatibility problems.

LS3R — Lock Shift G3, Right

1/11	7/12
ESC	|

Invoke G3 character set into GR.

Invoke Character Set Using Single Shifts

You can also invoke the G2 or G3 character sets for a single graphic character at a time, by means of single shift control functions. The Professional returns to the previous character set after displaying a single character.

Note: To avoid software compatibility problems do not use SS2 or SS3.

SS2 — Single Shift G2

SS2 is an 8-bit control character (8/14). You can also express it as an escape sequence when coding for a 7-bit environment:

1/11	4/14
ESC	N

Invoke G2 character set into GL for the next graphic character.

SS3 — Single Shift G3

SS3 is an 8-bit control character (8/15). You can also express it as an escape sequence when coding for a 7-bit environment:

1/11	4/15
ESC	O

Invoke G3 character set into GL for the next graphic character.

3.5.18 Character Attributes

The Professional can display the following character attributes. These attributes change the appearance of a displayed character without changing the character.

- ☐ Underline
- ☐ Reverse video (reverses foreground and background)

- ☐ Blink
- ☐ Bold (appearance of increased intensity)
- ☐ Any combination of these attributes

You can select one or more character attributes at one time. Selecting an attribute does not turn off other attributes already selected. After you select an attribute, all characters received by the Professional appear with that attribute. If you move the characters by scrolling, the attribute moves with the characters. Select the character attributes by using the following sequences.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, all character attributes are turned off.

SGR — Select Graphic Rendition

9/11	6/13			9/11	3/0	6/13
CSI	m		or	CSI	0	m

Turns off character attributes.

9/11	3/1	6/13
CSI	1	m

Selects bold (increased intensity) in 80-column mode only. Bold is not available in 132-column mode and is ignored.

9/11	3/4	6/13
CSI	4	m

Selects underline.

9/11	3/5	6/13
CSI	5	m

Selects blink.

You can have up to 100 blink fields (a field consists of adjacent blinking character cells within a single horizontal line). If you exceed this number, all blink fields stop blinking until the overflow condition ceases. Blinking character cells continue to blink while displaying graphics. If this is not desired, you should erase such cells before displaying graphics. The period for character cell blinking is four thirds of a second, and the duty cycle is 50 percent. Normal-intensity characters are produced.

9/11	3/7	6/13
CSI	7	m

Selects reverse video relative to the current screen state, as selected with the DECSNM sequence or P/OS Set-Up menu.

3.5.19 Line Attributes

Line attributes are display features that affect a complete display line. The cursor selects the line affected by the attribute. The cursor stays in the same character position when the attribute changes, unless the attribute would move the cursor past the right margin. In that case, the cursor stops at the right margin. When you move lines on the screen by scrolling, the attribute moves with the line. Select line attributes by using the following sequences.

Note: If you erase an entire line while using the erase in display (ED) sequence, the line attribute changes to single-height and single-width.

Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, all lines become single width and single height.

DECDHL — Double Height Line

Top Half			*Bottom Half*		
1/11	2/3	3/3	1/11	2/3	3/4
ESC	#	3	ESC	#	4

Makes the line with the cursor the top or bottom half of a double-height, double-width line. The sequences work in pairs on adjacent lines. The same character must be used on both lines to form full characters. If the line was previously single-width, single-height, all characters to the right of center are lost.

DECSWL — Single-Width Line

1/11	2/3	3/5
ESC	#	5

Makes the line with the cursor single-width, single-height. This is the line attribute for all new lines on the screen.

DECDWL — Double Width Line

1/11	2/3	3/6
ESC	#	6

Makes the line with the cursor double-width, single-height. If the line was previously single-width, single-height, all characters to the right of center screen are lost.

3.5.20 Tab Stops

You select tab stop positions on the horizontal lines of the screen. The cursor advances (tabs) to the next tab stop when the Professional receives a horizontal tab code (HT, 0/9). If there is no next tab, HT moves the cursor to the right margin. Set and clear the tab stops by using the following sequences.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, tab stops default to the settings selected in the P/OS Set-Up menu.

HTS — Horizontal Tab Set

HTS is an 8-bit control character (8/8). You can also express it as an escape sequence when coding for a 7-bit environment:

1/11	4/8
ESC	H

Sets a horizontal tab stop at cursor position.

TBC — Tabulation Clear

9/11	6/7		9/11	3/0	6/7
CSI	g	or	CSI	0	g

Clears a horizontal tab stop at cursor position

9/11	3/3	6/7
CSI	3	g

Clears all horizontal tab stops.

3.5.21 Erasing

Erasing removes characters from the screen without affecting other characters on the screen. Erased characters are lost. The cursor position does not change when erasing characters or lines.

If you erase an entire line while using the erase-in-display (ED) sequence, the line attribute becomes single-height, single-width. If you erase a line by using the erase-in-line (EL) sequence, the line attribute is not affected. The cursor determines the line for all EL sequences.

Erasing a character also erases any character attribute of the character. Erase characters by using the following sequences.

EL — Erase in Line

9/11	4/11		9/11	3/0	4/11
CSI	K	or	CSI	0	K

Erases from the cursor to the end of the line, including the cursor position.

9/11	3/1	4/11
CSI	1	K

Erases from the beginning of the line to the cursor, including the cursor position.

9/11	3/2	4/11
CSI	2	K

Erases complete line.

ED — Erase in Display

9/11	4/10		9/11	3/0	4/10
CSI	J	or	CSI	0	J

Erases from the cursor to the end of the screen, including the cursor position.

9/11	3/1	4/10
CSI	1	J

Erases from the beginning of the screen to the cursor, including the cursor position.

9/11	3/2	4/10
CSI	2	J

Erases the complete display. All lines are erased and changed to single-width. The cursor does not move.

3.5.22 Editing

You use editing sequences to insert or delete characters and lines of characters at the cursor position. The cursor position does not change when inserting or deleting lines. Delete characters or insert and delete lines by using the following sequences.

> Note: Insertion-replacement mode (IRM) selects how characters are added to the screen. See Inserting and Replacing Characters, in this chapter, for more information. Pn is a variable, ASCII coded, numeric parameter. If you select no parameter or a parameter value of 0, the Professional assumes a parameter value of 1.

DCH — Delete Character

9/11		5/0
CSI	Pn	P

Deletes Pn characters, starting with the character at the cursor position. When a character is deleted, all characters to the right of the cursor move to the left. This creates a space character at the right margin for each character deleted. Character attributes move with the characters. The spaces created at the end of the line have all their character attributes off.

IL — Insert Line

9/11		4/12
CSI	Pn	L

Inserts Pn lines at the line with the cursor. If fewer than Pn lines remain from the current line to the end of the scroll region, the number of lines inserted is the lesser number. Lines within the scroll region at and below the cursor move down. Lines moved past the bottom margin are lost. The cursor is reset to the first column. This sequence is ignored when the cursor is outside the scrolling region.

DL — Delete Line

9/11		4/13
CSI	Pn	M

Deletes Pn lines, starting at the line with the cursor. If fewer than Pn lines remain from the current line to the end of the scroll region, the number of lines deleted is the lesser number. As lines are deleted, lines within the scroll region and below the cursor move up, and blank lines are added at the bottom of the scroll region. The cursor is reset to the first column. This sequence is ignored when the cursor is outside the scrolling region.

3.5.23 Inserting and Replacing Characters

The Professional displays received characters at the cursor position. Insertion-replacement mode determines how the Professional adds characters to the screen. Insert mode displays the character and moves previously displayed characters to the right. Replace mode adds characters by replacing the character at the cursor position. Select insertion-replacement mode by using the following sequences.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, insertion-replacement mode resets.

IRM — Insertion-Replacement Mode

9/11	3/4	6/8
CSI	4	h

Set selects insert mode. New display characters move old display characters to the right. Characters moved past the right margin are lost.

9/11	3/4	6/12
CSI	4	l

Reset selects replace mode. New display characters replace old display characters at the cursor position. The old character is erased.

3.5.24 Printing

The Professional has a serial printer interface for local printing. Your application can select all text mode print operations by using Media Copy (MC) sequences. They are for text mode only and may not be used to reproduce the graphics-mode content of the display.

> Note: To print graphics, use the DECHCP (Hard Copy) sequence. Pressing the **PRINT SCREEN** key on the Professional keyboard has the same effect as using the DECHCP (Hard Copy) sequence.

A line of double-height characters print as two identical lines of single-width characters. Double-width characters print as single-width characters on a single line. No character attributes are preserved.

Before selecting a print operation, check the printer status by using the printer status report (DSR). Do not select a print operation if the serial printer is not ready to print. If the printer is unavailable or busy, the print request will still be assumed completed.

Select print operations by using the following sequences.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, MC auto print mode is turned off. Also, in native mode, RIS initializes the Terminal Subsystem only if MC printer controller mode is off. If MC printer controller mode is on, RIS resets the printer, not the Terminal Subsystem.

MC — Media Copy, Auto Print Mode

9/11	3/15	3/5	6/9
CSI	?	5	i

Turns on auto print mode. A display line prints after you move the cursor off the line, using a linefeed, form feed, or vertical tab, which is also transmitted to the printer. The line also prints during an auto wrap. Auto wrap lines end with a line feed.

9/11	3/15	3/4	6/9
CSI	?	4	i

Turns off auto print mode.

MC — Media Copy, Printer Controller

9/11	3/5	6/9
CSI	5	i

Turns on printer controller mode and sends CAN (1/8) to the printer. The Professional transmits received characters, including RIS sequences, to the printer without displaying them on the screen. All characters and character sequences except NUL, CSI 5 i, and CSI 4 i are sent to the printer. The Professional does not insert or delete spaces, or provide line delimiters, or select the correct

printer character set. This data path is unidirectional, and therefore an application cannot receive data from the printer while printer controller mode is on.

On the Professional, you always enter printer controller mode whenever the command is issued. On a VT102 terminal, you enter printer controller mode only if the printer is ready.

9/11	3/4	6/9
CSI	4	i

Turns off printer controller mode and sends a CR (0/13) to the printer.

> Note: Printer controller mode sequences that are used to turn that mode on or off are not sent to the printer. All other sequences are. To turn printer controller mode off, you must send the above sequence exactly as indicated or its equivalent 7-bit format (ESC [4 i).

MC — Media Copy, Print Cursor Line

9/11	3/15	3/1	6/9
CSI	?	1	i

Prints the display line containing the cursor. The cursor position does not change. The print-cursor-line sequence is completed when the line prints.

MC — Media Copy, Print Screen

9/11	6/9		9/11	3/0	6/9
CSI	i	or	CSI	0	i

Prints the screen display. Printer extent mode (DECPEX) selects either the full screen or the scrolling region to be printed. Select the scrolling region by using the set-top- and-bottom-margins (DECSTBM) sequence. Printer form feed mode (DECPFF) selects either a form feed (FF) or a line feed (LF) as the print termination character. The print-screen sequence is completed when the screen prints.

3.5.25 Printer Extent

Printer extent mode selects either the full screen or the scrolling region to be printed during a MC print screen sequence. It does not affect the DECHCP sequence. Select printer extent mode by using the following sequences.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, printer extent mode is set.

DECPEX — Printer Extent Mode

9/11	3/15	3/1	3/9	6/8
CSI	?	1	9	h

Set causes the full screen to be printed during a print screen.

9/11	3/15	3/1	3/9	6/12
CSI	?	1	9	l

Reset causes the scrolling region to be printed during a print screen.

3.5.26 Print Termination Character

Printer form feed mode determines which print termination character is transmitted after a print screen. The form feed (FF) and line feed (LF) control characters serve as the print termination characters. Select printer form feed mode by using the following sequence.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, printer form feed mode is set.

DECPFF — Printer Form Feed Mode

9/11	3/15	3/1	3/8	6/8
CSI	?	1	8	h

Set selects a form feed (FF) code as the print termination character. The Professional transmits this character to the printer after each print screen.

9/11	3/15	3/1	3/8	6/12
CSI	?	1	8	l

Reset selects a line feed (LF) code as the print termination character. On the Professional, no space is cleared between print screens if DECPFF is reset. On a VT102 terminal, there is one additional linefeed between print screens.

3.5.27 Print Screen (Text or Graphics Mode)

You can use the DECHCP (Hard Copy) sequence to print the screen contents. The DECPEX and DECPFF modes have no effect on DECHCP. The entire screen is printed, without any spacing between printouts. Pressing the **PRINT SCREEN** key has the same effect as using the DECHCP sequence. If the printer is unavailable or busy when the DECHCP sequence is sent, the print request will still be assumed completed.

If there is a possibility that the display contains graphics, a pixel-wise dump is used rather than a character-cell-wise dump. Whenever entering graphics mode, all text-mode lines are considered to contain a graphics image. This attribute is maintained on a line-by-line basis. It is cleared for lines which scroll off during full-screen scroll, or for lines which are erased or scrolled off if the extended bitmap option is not installed on the Professional, or for all lines after the Terminal Subsystem receives an RIS (reset to initial state) sequence. When a pixel-wise dump to the screen is performed, all pixels that are not turned off (black) on the display have a corresponding black dot on the printout.

A pixel-wise dump of text appears very differently than a cell-wise dump of the same text. A cell-wise dump is used for text-mode data. For text-mode data, a coded representation of each video character cell is sent to the printer. The printer decodes this information and generates the specified text for optimum readability, in accordance with the requirements and restrictions of the printer technology. Character and line attributes, such as bold or double wide, are not encoded and are therefore not reproduced in the printout.

For a pixel-wise dump, a coded representation of individual video pixels is sent to the printer, which reproduces the image pixel for pixel. The printout is apt to be distorted because of differences in aspect ratio, pixel size, etc. between the video display and the printout.

DECHCP — Hard Copy

1/11	2/3	3/7
ESC	#	7

Prints entire contents of screen.

3.5.28 Reports

The Professional transmits reports in response to program requests. Reports specify type and status, and cursor position. The report requests and responses are as follows.

> Note: The Professional does not respond to the DSR, DA, or DECID sequences during printer operation. Also, no reports are transmitted if DC1 is pending (DC3 suppresses all transmission).

DSR — Device Status Report

9/11	3/5	6/14
CSI	5	n

The application requests a status report.

9/11	3/0	6/14
CSI	0	n

The Professional's response is: Ready, no malfunctions detected.

9/11	3/15	3/1	3/5	6/14
CSI	?	1	5	n

The application requests a printer status report. The Professional checks the status of the printer. This report should be requested before any print operation.

9/11	3/15	3/1	3/3	6/14
CSI	?	1	3	n

The Professional's response is: The printer is not ready and never has been, when tested, since the last time the Professional was booted. Specifically, the data-terminal-ready (DTR) signal of the printer has not been on (asserted) when tested since the last time the Professional was booted. On a VT102 terminal, the status report indicates that the DTR signal of the printer has not been asserted since the terminal was last turned on or reset.

9/11	3/15	3/1	3/1	6/14
CSI	?	1	1	n

The printer is not ready to print. The printer DTR signal was on earlier when tested, but is now off.

9/11	3/15	3/1	3/0	6/14
CSI	?	1	0	n

The printer is ready to print. The printer DTR signal is on.

9/11	3/6	6/14
CSI	6	n

The application requests a cursor position report from the Professional.

CPR — Cursor Position Report

9/11		3/11		5/2
CSI	Pl	;	Pc	R

The Professional reports the cursor's position in response to a DSR sequence request. Pl indicates the line number and Pc indicates the column number. Line numbering depends on the state (set/reset) of origin mode (DECOM). The left-most column number is "1" and topmost line number is "1".

DA — Device Attributes

9/11	6/3		9/11	3/0	6/3
CSI	c	or	CSI	0	c

The application requests the Professional to identify itself.

In native mode, the response of the Professional to a DA request is in the following format (P1 through P4 are variable, ASCII coded, numeric parameters):

9/11	3/15	3/2	3/1	3/11		3/11		3/11		3/11		6/3
CSI	?	2	1	;	P1	;	P2	;	P3	;	P4	c

where

P1 hardware option:

0 no options present, monochrome monitor present
1 extended bitmap option and monochrome monitor present
2 extended bitmap option and color monitor present

P2 reserved for future use (0 for this release)

P3 revision level of the Professional (was 1 for first release version, changes with each release)

P4 reserved for future use (0 for this release)

Note: In the terminal emulation modes, the response of the Professional to a DA request depends on the particular emulation mode selected by the operator from the PRO/Communications menu. Only the ANSI compatible responses are described here. Refer to Section 3.6 for the VT52-mode response.

If the operator has selected Professional mode, the response of the Professional to a DA request is in the following format (P1 through P4 are variable, ASCII coded, numeric parameters):

9/11	3/15	3/2	3/3	3/11		3/11		3/11		3/11		6/3
CSI	?	2	3	;	P1	;	P2	;	P3	;	P4	c

where

P1 hardware option:

0 no options present, monochrome monitor present
1 extended bitmap option and monochrome monitor present
2 extended bitmap option and color monitor present

P2 reserved for future use (0 for this release)

P3 revision level of the Professional (was 1 for first release version, changes with each release)

P4 reserved for future use (0 for this release)

If the operator has selected VT102 mode, the response of the Professional to a DA request is as follows and is identical to the response from a VT102 terminal (the response is the same, regardless of whether the extended bitmap option is installed or not):

1/11	5/11	3/15	3/6	6/3
ESC	[	?	6	c

If the operator has selected VT125 mode and the extended bitmap option is installed, the response of the Professional to a DA request is as follows (the response is compatible with that from a VT125 terminal):

1/11	5/11	3/15	3/1	3/2	3/11	3/7	3/11	3/0	3/11	3/1	3/0	3/2	6/3
ESC	[	?	1	2	;	7	;	0	;	1	0	2	c

If the operator has selected VT125 mode and the extended bitmap option is not installed, the response of the Professional to a DA request is identical to that for VT102 mode.

DECID — Identify Terminal

1/11	5/10
ESC	Z

The application requests the Professional to identify itself. The Professional uses the device attributes (DA) sequence to respond. Future DIGITAL equipment may not support DECID. Therefore, new software should use the device-attributes sequence format.

3.5.29 Reset

The reset-to-initial-state (RIS) sequence initializes the Terminal Subsystem to default states and clears the screen.

RIS affects the text-mode states, graphics-mode states, and keyboard. Except for two keyboard states (Compose mode on/off, keyboard lock/unlock), RIS affects only the states that can be set by an application. RIS has no effect on states such as printer status or cursor block/underline, that cannot be changed from an application. On a VT102 terminal, RIS resets all states (programmable or not, such as "keyclick").

On the Professional, when RIS is selected, Compose mode is off and the keyboard becomes unlocked even if it were locked by KAM or DC3. Also, the graphics output map is reset to the VT125-compatible color mapping.

The defaults states selected by the RIS sequence are listed in Table 3–12. Reset the Terminal Subsystem by using the following sequence.

> Note: The RIS sequence is invoked whenever you power up (boot) the Professional or reset the PRO/Communications terminal emulator. In native mode, RIS initializes the Terminal Subsystem only if MC printer controller mode is off. If MC printer controller mode is on, RIS resets the printer, not the Terminal Subsystem.

RIS — Reset to Initial State

1/11	6/3
ESC	c

Resets the Professional to its initial default states.

3.5.30 Interlace

You use interlace mode for high resolution options. To reduce screen flicker, interlace should be turned off (reset) if such an option is not installed.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, interlace mode resets.

Table 3-12
States Selected by RIS Sequence (1)

Escape or Control Sequence	*State after RIS Sequence*
Code extension announcer	Return 7-bit code extensions
Scroll mode (DECSCLM)	Default value from P/OS Set-Up menu
Set top/bottom margins (DECSTBM)	Full screen
Origin mode (DECOM)	Reset (absolute line numbers)
Cursor enable mode (DECCEM)	Set (cursor blinks)
Column mode (DECCOLM)	Default value from P/OS Set-Up menu
Auto wrap mode DECAWM	Set (autowrap on)
Screen mode (DECSCNM)	Default value from P/OS Set-Up menu
Linefeed/new line mode (LNM)	Reset (send CR only)
Keyboard action mode (KAM)	Reset (unlock keyboard)
Auto repeat mode (DECARM)	Default value from P/OS Set-Up menu
Send-receive mode (SRM) (2)	Set (disable local echo)
Cursor key mode (DECCKM)	Reset (ANSI cursor control)
Auxiliary keypad mode	
(DECKPAM/DECKPNM)	DECKPNM (numeric mode)
Character sets	G0 = ASCII Graphics
	G1 = DEC Special Graphics
	G2 = DEC Supplemental Graphics
	G3 = unspecified
	G0 mapped to GL
	G2 mapped to GR
Select graphic rendition (SGR)	All graphic attributes off
Line attributes	
(DECDHL/DECSWL/DECDWL)	Single-height/single-width line
Horizontal tab stops (HTS/TBC)	Default value from P/OS Set-Up menu
Insertion-replacement mode (IRM)	Reset (replace)
Media copy auto print mode (MC)	Auto print off
Printer extent mode (DECPEX)	Set (print full screen)
Printer form feed mode (DECPFF)	Set (FF is print termination char.)
Interlace mode (DECINLM)	Reset (interlace off)
ANSI/VT52 mode (DECANM)	Reset (ANSI mode)
Keyboard	Unlocked (if locked by KAM or DC3)
Compose mode	Off
Output map (graphics)	Reset to VT125-compatible color map

1. The RIS sequence is invoked whenever you power up (boot) the Professional or reset the PRO/Communications terminal emulator. In native mode, RIS initializes the Terminal Subsystem only if MC printer controller mode is off. If MC printer controller mode is on, RIS resets the printer, not the Terminal Subsystem in native mode.
2. The SRM sequence is used only in terminal emulation modes.

DECINLM—Interlace Mode

9/11	3/15	3/9	6/8
CSI	?	9	h

Set turns on interlace mode. The raster scanning of the display is alternately interlaced.

9/11	3/15	3/9	6/12
CSI	?	9	l

Reset turns off interlace mode. The raster scanning of the display is not interlaced.

3.5.31 Tests and Adjustments

The Professional has a screen alignment pattern that service personnel use to adjust the screen. Display the screen alignment pattern by using the following sequence.

DECALN—Screen Alignment Display

1/11	2/3	3/8
ESC	#	8

Fills the screen with rectangles for screen focus and alignment (a VT102 terminal uses capital E's). This sequence is used by DIGITAL Manufacturing and Field Service personnel.

3.5.32 Control Strings

A control string is a delimited string of characters which may occur in the data stream as a logical entity for control purposes. A control string consists of an opening delimiter, a command string, and a closing delimiter.

The opening delimiter indicates the class of the component which is sending or receiving the control string. The following C1 control characters are used as opening delimiters: DCS (9/0), OSC (9/13) PM (9/14), APC (9/15). The closing delimiter (ST, 9/12) is used for all four classes of control strings.

On the Professional control strings are recognized but ignored (not displayed), except for terminal emulation modes, where certain ReGIS control strings are recognized and processed. The term "recognized" is used because, on a VT102 terminal, control strings are not recognized for their intended function and are not ignored: they do appear on the screen.

DCS—Device Control String Introducer

DCS is an 8-bit control character (9/0). It can also be expressed as an escape sequence for a 7-bit environment:

1/11	5/0
ESC	P

DCS is the opening delimiter of a control string for device control use. Any control string introduced by DCS and terminated by ST is recognized but ignored and is not displayed on the screen, with the following exception. In VT125 mode, DCS is used to introduce one or more ReGIS command strings. The control string is closed by the string terminator (ST) delimiter.

ST—String Terminator

ST is an 8-bit control character (9/12). It can also be expressed as an escape sequence for a 7-bit environment:

1/11	5/12
ESC	\

ST terminates any control string introduced by DCS, OSC, PM, or APC. When used as the closing delimiter of a ReGIS control string, ST causes the application to exit VT125 graphics mode and return to text mode.

OSC—Operating System Command Introducer

OSC is an 8-bit control character (9/13). It can also be expressed as an escape sequence for a 7-bit environment:

1/11	5/13
ESC	]

OSC is the opening delimiter of a control string for operating system use. Any control string introduced by OSC and terminated by ST is recognized but ignored and is not displayed on the screen.

PM—Privacy Message Introducer

PM is an 8-bit control character (9/14). It can also be expressed as an escape sequence for a 7-bit environment:

1/11	5/14
ESC	^

PM is the opening delimiter of a control string for privacy message use. Any control string introduced by PM and terminated by ST is recognized but ignored and is not displayed on the screen.

APC—Application Program Command Introducer

APC is an 8-bit control character (9/15). It can also be expressed as an escape sequence for a 7-bit environment:

1/11	5/15
ESC	_

APC is the opening delimiter of a control string for application program use. Any control string introduced by APC and terminated by ST is recognized but ignored and is not displayed on the screen.

3.5.33 VT125 Graphics Mode (ReGIS)

When the Professional is being operated in VT125 mode, receipt of a ReGIS control string causes the Professional to switch to graphics mode and process graphics commands. The device control string introducer DCS (9/0) marks the beginning of one or more ReGIS strings and a switch from text mode to graphics mode. All characters after DCS are passed to the ReGIS interpreter in the PRO/Communications application until the string terminator ST (9/12) is received. Upon receipt of ST, the Professional switches back to text mode. Refer to the section on control strings for more information on DCS and ST. Refer to the VT125 User's Guide for complete programming information on ReGIS. Appendix D of this manual lists differences between a VT125 terminal and the emulated features on the Professional.

> Note: When you power up (boot) the Professional, or reset the PRO/Communications terminal emulator, or use the reset-to-initial-state (RIS) sequence, the Professional's graphics output map is reset to the VT125-compatible color mapping.

ReGIS Control String Parameters

The characters following DCS, below, are ReGIS parameters as specified in the VT125 *User's Guide.*

9/0	7/0
DCS	p

Exit text mode and enter VT125 graphics mode, accepting data from the same data path as this sequence, at the command level that was in effect at the end of the last ReGIS control string. (ReGIS is at the highest command level if the Professional was booted [powered up] after the last device control string.)

9/0	3/1	7/0
DCS	1	p

Exit text mode and enter VT125 graphics mode, accepting data from the same data path as this sequence, and force immediate synchronization to the highest command level in the same way that the ";" character acts in a ReGIS control string.

3.5.34 ANSI/VT52 Compatibility

On the Professional, you can select VT52 mode from a PRO/Communications menu. You can also switch to VT52 compatibility by program control from VT102 mode. In VT52 mode, you can use DIGITAL-private VT52-compatible sequences. All VT52 compatible sequences are described in Section 3.6. Your application selects VT52 mode by means of the sequence shown below.

Programmable and Set-Up features and modes selected in ANSI mode are also in effect in VT52 mode. However, usually you cannot change these features and modes in VT52 mode.

In ANSI mode, use the following sequence to switch to VT52 mode.

Note: When you reset the PRO/Communications terminal emulator, the Professional is reset to ANSI-compatible VT102 mode.

DECANM—Enter VT52 Mode

9/11	3/15	3/2	6/12
CSI	?	2	l

In ANSI mode, reset selects VT52 mode. In VT52 mode, the Professional responds like a VT102 terminal emulating a VT52 terminal.

3.6 VT52-COMPATIBLE SEQUENCES

This section describes the VT52-compatible sequences recognized by the Professional.

Your application can use VT52-compatible sequences if it is operating in VT52 mode. VT52 mode is available to the Professional computer operator through a PRO/Communications menu selection. You can also select VT52 mode from VT102 mode by using the ANSI sequence documented in Section 3.5 (CSI ? 2 l). VT52-compatible sequences are DIGITAL-private sequences. They are not ANSI compatible. Future DIGITAL equipment may not accept ANSI-incompatible sequences. Therefore, you should not use VT52-compatible sequences in new software.

Programmable and Set-Up features and modes selected in ANSI mode are also in effect in VT52 mode. However, usually you cannot change these features and modes in VT52 mode. In VT52 mode, you can select only three modes with sequences: ANSI mode (DECANM), and the two auxiliary keypad modes—keypad application mode and keypad numeric mode.

Note: When you reset the PRO/Communications terminal emulator, the Professional is reset to ANSI-compatible VT102 mode.

3.6.1 ANSI/VT52 Compatibility

When in VT52 mode you cannot set or reset most Professional modes and features. You must switch to ANSI mode for that purpose. Once selected in ANSI mode, the states of modes and features are maintained when you switch back to VT52 mode. Select ANSI (compatibility) mode by using the following sequence.

DECANM—Enter ANSI Mode

1/11	3/12
ESC	<

The Professional interprets all subsequent sequences according to ANSI standards X3.64-1979 and X3.41-1974. The VT52 escape sequences described in the balance of this chapter are not recognized and are ignored by the Professional.

3.6.2 Cursor Positioning

The cursor indicates the active screen position where the next character will appear. A number of operations implicitly affect cursor positioning. In addition, your application can control cursor movement by means of the following sequences.

> Note: You must select the margins for VT52 mode in ANSI mode (either Professional or VT102 mode), as indicated in Section 3.5. If you do not select margins, the Professional uses the complete screen. The cursor's home position is dependent on the margin selection.

Cursor Up

1/11	4/1
ESC	A

Moves cursor up one line in same column. Cursor stops at top margin.

Cursor Down

1/11	4/2
ESC	B

Moves cursor down one line in same column. Cursor stops at bottom margin.

Cursor Right

1/11	4/3
ESC	C

Moves cursor one column to right. Cursor stops at right margin.

Cursor Left

1/11	4/4
ESC	D

Moves cursor one column to left. Cursor stops at left margin.

Cursor to Home

1/11	4/8
ESC	H

Moves cursor to home position, which is dependent on margin selection.

Direct Cursor Address

1/11	5/9	line	column
ESC	Y	**	**

Moves cursor to specified line and column. Line and column numbers are ASCII characters whose codes are calculated by adding the numerical line or column

number (in octal, decimal, or hexadecimal notation) to 37 (if octal), or 31 (if decimal), or 1F (if hexadecimal). For example, line 1 and column 8 (decimal) are decimal code values 32 and 39, respectively.

Reverse Linefeed

1/11	4/9
ESC	I

Moves cursor up one line in same column. If cursor is at top margin, screen performs scroll-down.

3.6.3 Auxiliary Keypad Character Selection

The auxiliary keypad transmits to the application either numeric characters or control (application) functions. Select keypad application mode to generate control functions. Select keypad numeric mode to generate numeric characters. See Table 3-13 for the codes generated by the keypad. Select the keypad mode by using the following sequences.

Select Keypad Application Mode

1/11	3/13
ESC	=

Keypad transmits control codes to application.

Select Keypad Numeric Mode

1/11	3/14
ESC	>

Keypad transmits characters that match the numeric, comma, period, and minus sign keys on main keyboard.

3.6.4 Character Sets and Selection

In VT52 mode, the Professional maps the ASCII Graphics Set into GL by default. You can also map the DEC Special Graphics Set into GL, by using the Enter Graphics Mode sequence. On a VT52 terminal, the VT52 Graphics Mode Character Set is used instead of the DEC Special Graphics Set. See Table 3-14 for a comparison of these two sets. The VT52 Graphics Mode Character Set is not available either in VT52 mode on the Professional, or on a VT102 terminal when it is emulating a VT52 terminal. Use the following sequences to map DEC Special Graphics into GL and to return to the default mapping (ASCII Graphics in GL).

Enter Graphics Mode

1/11	4/6
ESC	F

Designate DEC Special Graphics Set as G1 and invoke G1 into GL.

Table 3-13
Codes Generated by Auxilliary Keypad Keys (VT52 Mode)

Key	*Keypad Numeric Mode*		*Keypad Application Mode*		
	3/0		1/11	3/15	7/0
0	0		ESC	?	p
	3/1		1/11	3/15	7/1
1	1		ESC	?	q
	3/2		1/11	3/15	7/2
2	2		ESC	?	r
	3/3		1/11	3/15	7/3
3	3		ESC	?	s
	3/4		1/11	3/15	7/4
4	4		ESC	?	t
	3/5		1/11	3/15	7/5
5	5		ESC	?	u
	3/6		1/11	3/15	7/6
6	6		ESC	?	v
	3/7		1/11	3/15	7/7
7	7		ESC	?	w
	3/8		1/11	3/15	7/8
8	8		ESC	?	x
	3/9		1/11	3/15	7/9
9	9		ESC	?	y
	2/13		1/11	3/15	6/13[A]
–	– (minus)		ESC	?	m
	2/12		1/11	3/15	6/12[A]
,	, (comma)		ESC	?	l
	2/14		1/11	3/15	6/14
.	. (period)		ESC	?	n
	0/13		1/11	3/15	4/13
Enter[B]	CR		ESC	?	M
	or 0/13	0/10			
	CR	LF			
	1/11	5/0	1/11	5/0	
PF1	ESC	P	ESC	P	
	1/11	5/1	1/11	5/1	
PF2	ESC	Q	ESC	Q	
	1/11	5/2	1/11	5/2	
PF3	ESC	R	ESC	R	
	1/11	5/3	1/11	5/3 (A)	
PF4	ESC	S	ESC	S	

A. You cannot generate these sequences on a VT52 terminal.

B. In keypad numeric mode, **ENTER** generates the same codes as **RETURN**. You can change the code generated by **RETURN** with the Linefeed/New Line Mode. When reset, the Linefeed/New Line Mode causes **RETURN** to generate a single control character (CR). When set, the mode causes **RETURN** to generate two control characters (CR,LF).

TABLE 3-14
Comparison of ASCII Graphics, DEC Special Graphics and VT52 Graphics Mode Character Sets

Key	*ASCII Graphics*	*DEC Special Graphics*		*VT52 Graphics Mode*
5/15	_		Blank	Blank
6/00	`	◆	Diamond	Reserved
6/01	a	▒	Checkerboard	Solid rectangle
6/02	b	HT	Horizontal tab	1/
6/03	c	FF	Form Feed	3/
6/04	d	CR	Carriage return	5/
6/05	e	LF	Linefeed	7/
6/06	f	°	Degree sign	Degree sign
6/07	g	±	Plus/minus sign	Plus/minus sign
6/08	h	NL	New line	Right arrow
6/09	i	VT	Vertical tab	Ellipsis (dots)
6/10	j	┘	Lower-right corner	Divide by
6/11	k	┐	Upper-right corner	Down arrow
6/12	l	┌	Upper-left corner	Bar at scan 0
6/13	m	└	Lower-left corner	Bar at scan 1
6/14	n	┼	Crossing lines	Bar at scan 2
6/15	o	⎺	Horizontal line-scan 1	Bar at scan 3
7/00	p	⎻	Horizontal line-scan 3	Bar at scan 4
7/01	q	─	Horizontal line-scan 5	Bar at scan 5
7/02	r	⎼	Horizontal line-scan 7	Bar at scan 6
7/03	s	⎽	Horizontal line-scan 9	Bar at scan 7
7/04	t	├	Left "T"	Subscript 0
7/05	u	┤	Right "T"	Subscript 1
7/06	v	┴	Bottom "T"	Subscript 2
7/07	w	┬	Top "T"	Subscript 3
7/08	x	│	Vertical bar	Subscript 4
7/09	y	≤	Less than or equal to	Subscript 5
7/10	z	≥	Greater than or equal to	Subscript 6
7/11	{	π	Pi	Subscript 7
7/12	|	≠	Not equal to	Subscript 8
7/13	}	£	UK pound sign	Subscript 9
7/14	~	·	Centered dot	Paragraph

Exit Graphics Mode

1/11	4/7
ESC	G

Designate ASCII Graphics Set as G0 and invoke G0 into GL. This selects the default character set into GL.

3.6.5 Erasing

Erasing removes characters from the screen. Erased characters are lost. Erase characters by using the following sequences.

Erase to End of Line

1/11	4/11
ESC	K

Erases all characters from the cursor to the end of the current line, including the cursor position. The cursor does not move.

Erase to End of Screen

1/11	4/10
ESC	J

Erases all characters from the cursor to the end of the screen, including the cursor position. The cursor does not move.

3.6.6 Printing

The Professional has a serial printer interface for local printing. The application can select text mode print operations by using the two sequences documented below.

> Note: These sequences are for text mode only and may not be used to reproduce the graphics-mode content of the display. To print graphics, use the DECHCP (Hard Copy) sequence in ANSI mode. Pressing the **PRINT SCREEN** key on the Professional keyboard has the same effect as using the DECHCP (Hard Copy) sequence.

A line of double-height characters print as two identical lines of single-width characters. Double-width characters print as single-width characters on a single line. No character attributes are preserved.

Before selecting a print operation, check the printer status by using the printer status report (DSR) in ANSI mode. Do not select a print operation if the serial printer is not ready to print. If the printer is unavailable or busy, the print request will be assumed completed.

Select print operations by using the following sequences.

Auto Print

1/11 5/14
ESC ^

Turns on auto print. A display line prints after you move the cursor off the line, using a linefeed, form feed, or vertical tab (also transmitted to the printer). The line also prints during an auto wrap. Auto wrap lines end with CR, LF.

1/11 5/15
ESC _

Turns off auto print.

Print Controller

1/11 5/7
ESC W

Turns on print controller. The Professional transmits received characters to the printer without displaying them. The Professional does not insert or delete spaces, or provide delimiters, or select the printer character set.

1/11 5/8
ESC X

Turns off printer controller.

Print Cursor Line

1/11 5/13
ESC]

Prints the display line with the cursor. The cursor position does not change. The print-cursor-line sequence is completed when the line prints.

Print Screen

1/11 5/6
ESC V

Prints the screen display.

In ANSI mode, use printer extent mode (DECPEX) to select either the full screen or the scrolling region to be printed. Select the scrolling region by using the set-top- and-bottom-margins (DECSTBM) sequence. The print-screen sequence is completed when the screen prints.

3.6.7 Reports

The Professional transmits reports in response to program requests. The Professional generates only one report in VT52 mode. The report requests and response are as follows.

Identify

1/11	5/10
ESC	Z

The application requests the Professional to identify itself.

1/11	2/15	5/10
ESC	/	Z

The Professional identifies itself as a VT102 emulating a VT52.

APPENDIX A
CHARACTER SET SUMMARY

This appendix contains the code tables for the DEC Multinational Character Set and for the DEC Special Graphics Set (Tables A–1 and A–2, respectively). The latter is shown mapped onto a 7-bit code table: the DEC Special Graphics Set is in GL (columns 2 through 7), and the C0 control characters are in columns 0 and 1. Octal, decimal, and hexadecimal numeric codes appear next to each character on all code tables.

All DEC Multinational characters are also listed in Table A–3, ordered by column/row position from 0/0 through 15/15. The "official" name of the character and the octal, decimal, and hexadecimal code values are given for each column/row position.

Table A-1
DEC Multinational Character Set (C0 and GL Codes)

COLUMN	0		1		2		3		4		5		6		7	
ROW / BITS b8 b7 b6 b5 b4 b3 b2 b1	0 0 0 0		0 0 0 1		0 0 1 0		0 0 1 1		0 1 0 0		0 1 0 1		0 1 1 0		0 1 1 1	
0	0 0 0 0	NUL 0 0 0	DLE	20 16 10	SP	40 32 20	0	60 48 30	@	100 64 40	P	120 80 50	`	140 96 60	p	160 112 70
1	0 0 0 1	SOH 1 1 1	DC1 (XON)	21 17 11	!	41 33 21	1	61 49 31	A	101 65 41	Q	121 81 51	a	141 97 61	q	161 113 71
2	0 0 1 0	STX 2 2 2	DC2	22 18 12	"	42 34 22	2	62 50 32	B	102 66 42	R	122 82 52	b	142 98 62	r	162 114 72
3	0 0 1 1	ETX 3 3 3	DC3 (XOFF)	23 19 13	#	43 35 23	3	63 51 33	C	103 67 43	S	123 83 53	c	143 99 63	s	163 115 73
4	0 1 0 0	EOT 4 4 4	DC4	24 20 14	$	44 36 24	4	64 52 34	D	104 68 44	T	124 84 54	d	144 100 64	t	164 116 74
5	0 1 0 1	ENQ 5 5 5	NAK	25 21 15	%	45 37 25	5	65 53 35	E	105 69 45	U	125 85 55	e	145 101 65	u	165 117 75
6	0 1 1 0	ACK 6 6 6	SYN	26 22 16	&	46 38 26	6	66 54 36	F	106 70 46	V	126 86 56	f	146 102 66	v	166 118 76
7	0 1 1 1	BEL 7 7 7	ETB	27 23 17	'	47 39 27	7	67 55 37	G	107 71 47	W	127 87 57	g	147 103 67	w	167 119 77
8	1 0 0 0	BS 10 8 8	CAN	30 24 18	(	50 40 28	8	70 56 38	H	110 72 48	X	130 88 58	h	150 104 68	x	170 120 78
9	1 0 0 1	HT 11 9 9	EM	31 25 19	)	51 41 29	9	71 57 39	I	111 73 49	Y	131 89 59	i	151 105 69	y	171 121 79
10	1 0 1 0	LF 12 10 A	SUB	32 26 1A	*	52 42 2A	:	72 58 3A	J	112 74 4A	Z	132 90 5A	j	152 106 6A	z	172 122 7A
11	1 0 1 1	VT 13 11 B	ESC	33 27 1B	+	53 43 2B	;	73 59 3B	K	113 75 4B	[	133 91 5B	k	153 107 6B	{	173 123 7B
12	1 1 0 0	FF 14 12 C	FS	34 28 1C	,	54 44 2C	<	74 60 3C	L	114 76 4C	\	134 92 5C	l	154 108 6C	|	174 124 7C
13	1 1 0 1	CR 15 13 D	GS	35 29 1D	-	55 45 2D	=	75 61 3D	M	115 77 4D	]	135 93 5D	m	155 109 6D	}	175 125 7D
14	1 1 1 0	SO 16 14 E	RS	36 30 1E	.	56 46 2E	>	76 62 3E	N	116 78 4E	^	136 94 5E	n	156 110 6E	~	176 126 7E
15	1 1 1 1	SI 17 15 F	US	37 31 1F	/	57 47 2F	?	77 63 3F	O	117 79 4F	_	137 95 5F	o	157 111 6F	DEL	177 127 7F

C0 CODES

GL CODES (ASCII GRAPHICS)

KEY

CHARACTER: ESC | 33 OCTAL | 27 DECIMAL | 1B HEX

Table A-1
DEC Multinational Character Set (C1 and GR Codes)

8		9		10		11		12		13		14		15		COLUMN	
1 0 0 0		1 0 0 1		1 0 1 0		1 0 1 1		1 1 0 0		1 1 0 1		1 1 1 0		1 1 1 1		b8 b7 BITS b6 b5 b4 b3 b2 b1	ROW
	200 128 80	DCS	220 144 90		240 160 A0	°	260 176 B0	À	300 192 C0		320 208 D0	à	340 224 E0		360 240 F0	0 0 0 0	0
	201 129 81	PU1	221 145 91	¡	241 161 A1	±	261 177 B1	Á	301 193 C1	Ñ	321 209 D1	á	341 225 E1	ñ	361 241 F1	0 0 0 1	1
	202 130 82	PU2	222 146 92	¢	242 162 A2	2	262 178 B2	Â	302 194 C2	Ò	322 210 D2	â	342 226 E2	ò	362 242 F2	0 0 1 0	2
	203 131 83	STS	223 147 93	£	243 163 A3	3	263 179 B3	Ã	303 195 C3	Ó	323 211 D3	ã	343 227 E3	ó	363 243 F3	0 0 1 1	3
IND	204 132 84	CCH	224 148 94		244 164 A4		264 180 B4	Ä	304 196 C4	Ô	324 212 D4	ä	344 228 E4	ô	364 244 F4	0 1 0 0	4
NEL	205 133 85	MW	225 149 95	¥	245 165 A5	µ	265 181 B5	Å	305 197 C5	Õ	325 213 D5	å	345 229 E5	õ	365 245 F5	0 1 0 1	5
SSA	206 134 86	SPA	226 150 96		246 166 A6	¶	266 182 B6	Æ	306 198 C6	Ö	326 214 D6	æ	346 230 E6	ö	366 246 F6	0 1 1 0	6
ESA	207 135 87	EPA	227 151 97	§	247 167 A7	·	267 183 B7	Ç	307 199 C7	Œ	327 215 D7	ç	347 231 E7	œ	367 247 F7	0 1 1 1	7
HTS	210 136 88		230 152 98	¤	250 168 A8		270 184 B8	È	310 200 C8	Ø	330 216 D8	è	350 232 E8	ø	370 248 F8	1 0 0 0	8
HTJ	211 137 89		231 153 99	©	251 169 A9	1	271 185 B9	É	311 201 C9	Ù	331 217 D9	é	351 233 E9	ù	371 249 F9	1 0 0 1	9
VTS	212 138 8A		232 154 9A	ª	252 170 AA	º	272 186 BA	Ê	312 202 CA	Ú	332 218 DA	ê	352 234 EA	ú	372 250 FA	1 0 1 0	10
PLD	213 139 8B	CSI	233 155 9B	«	253 171 AB	»	273 187 BB	Ë	313 203 CB	Û	333 219 DB	ë	353 235 EB	û	373 251 FB	1 0 1 1	11
PLU	214 140 8C	ST	234 156 9C		254 172 AC	¼	274 188 BC	Ì	314 204 CC	Ü	334 220 DC	ì	354 236 EC	ü	374 252 FC	1 1 0 0	12
RI	215 141 8D	OSC	235 157 9D		255 173 AD	½	275 189 BD	Í	315 205 CD	Ý	335 221 DD	í	355 237 ED	ý	375 253 FD	1 1 0 1	13
SS2	216 142 8E	PM	236 158 9E		256 174 AE		276 190 BE	Î	316 206 CE		336 222 DE	î	356 238 EE		376 254 FE	1 1 1 0	14
SS3	217 143 8F	APC	237 159 9F		257 175 AF	¿	277 191 BF	Ï	317 207 CF	ß	337 223 DF	ï	357 239 EF		377 255 FF	1 1 1 1	15

C1 CODES

GR CODES
(DEC SUPPLEMENTAL GRAPHICS)

Table A-2
DEC Special Graphics Set Mapped into GL

COLUMN	0		1		2		3		4		5		6		7	
BITS B7 B6 B5 / ROW B4 B3 B2 B1	0 0 0		0 0 1		0 1 0		0 1 1		1 0 0		1 0 1		1 1 0		1 1 1	
0 (0 0 0 0)	NUL	0 0 0	DLE	20 16 10	SP	40 32 20	0	60 48 30	@	100 64 40	P	120 80 50	♦	140 96 60	— SCAN 3	160 112 70
1 (0 0 0 1)	SOH	1 1 1	DC1 (XON)	21 17 11	!	41 33 21	1	61 49 31	A	101 65 41	Q	121 81 51	▒	141 97 61	— SCAN 5	161 113 71
2 (0 0 1 0)	STX	2 2 2	DC2	22 18 12	"	42 34 22	2	62 50 32	B	102 66 42	R	122 82 52	␉	142 98 62	— SCAN 7	162 114 72
3 (0 0 1 1)	ETX	3 3 3	DC3 (XOFF)	23 19 13	#	43 35 23	3	63 51 33	C	103 67 43	S	123 83 53	␌	143 99 63	⎺ SCAN 9	163 115 73
4 (0 1 0 0)	EOT	4 4 4	DC4	24 20 14	$	44 36 24	4	64 52 34	D	104 68 44	T	124 84 54	␍	144 100 64	├	164 116 74
5 (0 1 0 1)	ENQ	5 5 5	NAK	25 21 15	%	45 37 25	5	65 53 35	E	105 69 45	U	125 85 55	␊	145 101 65	┤	165 117 75
6 (0 1 1 0)	ACK	6 6 6	SYN	26 22 16	&	46 38 26	6	66 54 36	F	106 70 46	V	126 86 56	°	146 102 66	⊥	166 118 76
7 (0 1 1 1)	BEL	7 7 7	ETB	27 23 17	'	47 39 27	7	67 55 37	G	107 71 47	W	127 87 57	±	147 103 67	⊤	167 119 77
8 (1 0 0 0)	BS	10 8 8	CAN	30 24 18	(	50 40 28	8	70 56 38	H	110 72 48	X	130 88 58	␤	150 104 68	│	170 120 78
9 (1 0 0 1)	HT	11 9 9	EM	31 25 19	)	51 41 29	9	71 57 39	I	111 73 49	Y	131 89 59	␋	151 105 69	≤	171 121 79
10 (1 0 1 0)	LF	12 10 A	SUB	32 26 1A	*	52 42 2A	:	72 58 3A	J	112 74 4A	Z	132 90 5A	┘	152 106 6A	≥	172 122 7A
11 (1 0 1 1)	VT	13 11 B	ESC	33 27 1B	+	53 43 2B	;	73 59 3B	K	113 75 4B	[	133 91 5B	┐	153 107 6B	π	173 123 7B
12 (1 1 0 0)	FF	14 12 C	FS	34 28 1C	,	54 44 2C	<	74 60 3C	L	114 76 4C	\	134 92 5C	┌	154 108 6C	≠	174 124 7C
13 (1 1 0 1)	CR	15 13 D	GS	35 29 1D	-	55 45 2D	=	75 61 3D	M	115 77 4D	]	135 93 5D	└	155 109 6D	£	175 125 7D
14 (1 1 1 0)	SO	16 14 E	RS	36 30 1E	.	56 46 2E	>	76 62 3E	N	116 78 4E	^	136 94 5E	┼	156 110 6E	·	176 126 7E
15 (1 1 1 1)	SI	17 15 F	US	37 31 1F	/	57 47 2F	?	77 63 3F	O	117 79 4F	(BLANK)	137 95 5F	⎻ SCAN 1	157 111 6F	DEL	177 127 7F

C0 CODES — GL CODES (DEC SPECIAL GRAPHICS)

KEY

CHARACTER		
ESC	33	OCTAL
	27	DECIMAL
	1B	HEX

Table A-3
DEC Multinational Character Set and Numeric Codes

Char	*Col/Row*	*Hex.*	*Decimal*	*Octal*	*Description*
NUL	0/00	0	0	0	Null
SOH	0/01	1	1	1	Start of heading
STX	0/02	2	2	2	Start of text
ETX	0/03	3	3	3	End of text
EOT	0/04	4	4	4	End of transmission
ENQ	0/05	5	5	5	Enquiry
ACK	0/06	6	6	6	Acknowledge
BEL	0/07	7	7	7	Bell
BS	0/08	8	8	10	Backspace
HT	0/09	9	9	11	Horizontal tab
LF	0/10	A	10	12	Line feed
VT	0/11	B	11	13	Vertical tab
FF	0/12	C	12	14	Form feed
CR	0/13	D	13	15	Carriage return
SO (LS1)	0/14	E	14	16	Shift out (Lock shift G1)
SI (LS0)	0/15	F	15	17	Shift in (Lock shift G0)
DLE	1/00	10	16	20	Data link escape
DC1	1/01	11	17	21	Device control 1
DC2	1/02	12	18	22	Device control 2
DC3	1/03	13	19	23	Device control 3
DC4	1/04	14	20	24	Device control 4
NAK	1/05	15	21	25	Negative acknowledge
SYN	1/06	16	22	26	Synchronous idle
ETB	1/07	17	23	27	End of transmission block
CAN	1/08	18	24	30	Cancel
EM	1/09	19	25	31	End of medium
SUB	1/10	1A	26	32	Substitute
ESC	1/11	1B	27	33	Escape
FS	1/12	1C	28	34	File separator
GS	1/13	1D	29	35	Group separator
RS	1/14	1E	30	36	Record separator
US	1/15	1F	31	37	Unit separator
SP	2/00	20	32	40	Space
!	2/01	21	33	41	Exclamation point
"	2/02	22	34	42	Double quotation mark
#	2/03	23	35	43	Number sign, octothorp

Table A-3 (Cont.)

Char	*Col/Row*	*Hex.*	*Decimal*	*Octal*	*Description*
$	2/04	24	36	44	Dollar sign
%	2/05	25	37	45	Percent sign
&	2/06	26	38	46	Ampersand
'	2/07	27	39	47	Apostrophe
(	2/08	28	40	50	Opening parenthesis
)	2/09	29	41	51	Closing parenthesis
*	2/10	2A	42	52	Asterisk
+	2/11	2B	43	53	Plus
,	2/12	2C	44	54	Comma
-	2/13	2D	45	55	Hyphen, minus
.	2/14	2E	46	56	Period, decimal point
/	2/15	2F	47	57	Slash
0	3/00	30	48	60	Digit 0
1	3/01	31	49	61	Digit 1
2	3/02	32	50	62	Digit 2
3	3/03	33	51	63	Digit 3
4	3/04	34	52	64	Digit 4
5	3/05	35	53	65	Digit 5
6	3/06	36	54	66	Digit 6
7	3/07	37	55	67	Digit 7
8	3/08	38	56	70	Digit 8
9	3/09	39	57	71	Digit 9
:	3/10	3A	58	72	Colon
;	3/11	3B	59	73	Semicolon
<	3/12	3C	60	74	Less than
=	3/13	3D	61	75	Equals
>	3/14	3E	62	76	Greater than
?	3/15	3F	63	77	Question mark
@	4/00	40	64	100	Commercial at
A	4/01	41	65	101	Capital A
B	4/02	42	66	102	Capital B
C	4/03	43	67	103	Capital C
D	4/04	44	68	104	Capital D
E	4/05	45	69	105	Capital E
F	4/06	46	70	106	Capital F
G	4/07	47	71	107	Capital G
H	4/08	48	72	110	Capital H
I	4/09	49	73	111	Capital I

Table A-3 (Cont.)

Char	*Col/Row*	*Hex.*	*Decimal*	*Octal*	*Description*
J	4/10	4A	74	112	Capital J
K	4/11	4B	75	113	Capital K
L	4/12	4C	76	114	Capital L
M	4/13	4D	77	115	Capital M
N	4/14	4E	78	116	Capital N
O	4/15	4F	79	117	Capital O
P	5/00	50	80	120	Capital P
Q	5/01	51	81	121	Capital Q
R	5/02	52	82	122	Capital R
S	5/03	53	83	123	Capital S
T	5/04	54	84	124	Capital T
U	5/05	55	85	125	Capital U
V	5/06	56	86	126	Capital V
W	5/07	57	87	127	Capital W
X	5/08	58	88	130	Capital X
Y	5/09	59	89	131	Capital Y
Z	5/10	5A	90	132	Capital Z
[	5/11	5B	91	133	Opening bracket
\	5/12	5C	92	134	Backslash
]	5/13	5D	93	135	Closing bracket
^	5/14	5E	94	136	Circumflex accent
_	5/15	5F	95	137	Underline
`	6/00	60	96	140	Opening single quotation mark, grave accent
a	6/01	61	97	141	Small a
b	6/02	62	98	142	Small b
c	6/03	63	99	143	Small c
d	6/04	64	100	144	Small d
e	6/05	65	101	145	Small e
f	6/06	66	102	146	Small f
g	6/07	67	103	147	Small g
h	6/08	68	104	150	Small h
i	6/09	69	105	151	Small i
j	6/10	6A	106	152	Small j
k	6/11	6B	107	153	Small k
l	6/12	6C	108	154	Small l
m	6/13	6D	109	155	Small m
n	6/14	6E	110	156	Small n

Table A-3 (Cont.)

Char	*Col/Row*	*Hex.*	*Decimal*	*Octal*	*Description*
o	6/15	6F	111	157	Small o
p	7/00	70	112	160	Small p
q	7/01	71	113	161	Small q
r	7/02	72	114	162	Small r
s	7/03	73	115	163	Small s
t	7/04	74	116	164	Small t
u	7/05	75	117	165	Small u
v	7/06	76	118	166	Small v
w	7/07	77	119	167	Small w
x	7/08	78	120	170	Small x
y	7/09	79	121	171	Small y
z	7/10	7A	122	172	Small z
{	7/11	7B	123	173	Opening brace
|	7/12	7C	124	174	Vertical line
}	7/13	7D	125	175	Closing brace
~	7/14	7E	126	176	Tilde
DEL	7/15	7F	127	177	Delete
	8/00	80	128	200	Reserved
	8/01	81	129	201	Reserved
	8/02	82	130	202	Reserved
	8/03	83	131	203	Reserved
IND	8/04	84	132	204	Index
NEL	8/05	85	133	205	Next line
SSA	8/06	86	134	206	Start of selected area
ESA	8/07	87	135	207	End of selected area
HTS	8/08	88	136	210	Horizontal tab set
HTJ	8/09	89	137	211	Horizontal tab with justify
VTS	8/10	8A	138	212	Vertical tab set
PLD	8/11	8B	139	213	Partial line down
PLU	8/12	8C	140	214	Partial line up
RI	8/13	8D	141	215	Reverse index
SS2	8/14	8E	142	216	Single shift G2
SS3	8/15	8F	143	217	Single shift G3
DCS	9/00	90	144	220	Device control string
PU1	9/01	91	145	221	Private use one
PU2	9/02	92	146	222	Private use two
STS	9/03	93	147	223	Set transmit state
CCH	9/04	94	148	224	Cancel character

Table A-3 (Cont.)

Char	*Col/Row*	*Hex.*	*Decimal*	*Octal*	*Description*
MW	9/05	95	149	225	Message waiting
SPA	9/06	96	150	226	Start of protected area
EPA	9/07	97	151	227	End of protected area
	9/08	98	152	230	Reserved
	9/09	99	153	231	Reserved
	9/10	9A	154	232	Reserved
CSI	9/11	9B	155	233	Control Sequence introducer
ST	9/12	9C	156	234	String terminator
OSC	9/13	9D	157	235	Operating system command
PM	9/14	9E	158	236	Privacy message
APC	9/15	9F	159	237	Application program command
	10/00	A0	160	240	Reserved
¡	10/01	A1	161	241	Inverted exclamation mark
¢	10/02	A2	162	242	Cent sign
£	10/03	A3	163	243	Pound sign
	10/04	A4	164	244	Reserved
¥	10/05	A5	165	245	Yen sign
	10/06	A6	166	246	Reserved
§	10/07	A7	167	247	Section sign
¤	10/08	A8	168	250	General currency sign
©	10/09	A9	169	251	Copyright sign
ª	10/10	AA	170	252	Feminine ordinal indicator
<<	10/11	AB	171	253	Angle quotation mark left
	10/12	AC	172	254	Reserved
	10/13	AD	173	255	Reserved
	10/14	AE	174	256	Reserved
	10/15	AF	175	257	Reserved
°	11/00	B0	176	260	Degree sign
±	11/01	B1	177	261	Plus/minus sign
2	11/02	B2	178	262	Superscript 2
3	11/03	B3	179	263	Superscript 3
	11/04	B4	180	264	Reserved
µ	11/05	B5	181	265	Micro sign
¶	11/06	B6	182	266	Paragraph sign, pilcrow
·	11/07	B7	183	267	Middle dot
	11/08	B8	184	270	Reserved
1	11/09	B9	185	271	Superscript 1
º	11/10	BA	186	272	Masculine ordinal indicator

Table A-3 (Cont.)

Char	*Col/Row*	*Hex.*	*Decimal*	*Octal*	*Description*
>>	11/11	BB	187	273	Angle quotation mark right
¼	11/12	BC	188	274	Fraction one quarter
½	11/13	BD	189	275	Fraction one half
	11/14	BE	190	276	Reserved
¿	11/15	BF	191	277	Inverted question mark
À	12/00	C0	192	300	Capital A with grave accent
Á	12/01	C1	193	301	Capital A with acute accent
Â	12/02	C2	194	302	Capital A with circumflex accent
Ã	12/03	C3	195	303	Capital A with tilde
Ä	12/04	C4	196	304	Capital A with diaeresis or umlaut mark
Å	12/05	C5	197	305	Capital A with ring
Æ	12/06	C6	198	306	Capital AE ligature
Ç	12/07	C7	199	307	Capital C with cedilla
È	12/08	C8	200	310	Capital E with grave accent
É	12/09	C9	201	311	Capital E with acute accent
Ê	12/10	CA	202	312	Capital E with circumflex accent
Ë	12/11	CB	203	313	Capital E with diaeresis or umlaut mark
Ì	12/12	CC	204	314	Capital I with grave accent
Í	12/13	CD	205	315	Capital I with acute accent
Î	12/14	CE	206	316	Capital I with circumflex accent
Ï	12/15	CF	207	317	Capital I with diaeresis or umlaut mark
	13/00	D0	208	320	Reserved
Ñ	13/01	D1	209	321	Capital N with tilde
Ò	13/02	D2	210	322	Capital O with grave accent
Ó	13/03	D3	211	323	Capital O with acute accent
Ô	13/04	D4	212	324	Capital O with circumflex accent
Õ	13/05	D5	213	325	Capital O with tilde
Ö	13/06	D6	214	326	Capital O with diaeresis or umlaut mark
Œ	13/07	D7	215	327	Capital OE ligature
Ø	13/08	D8	216	330	Capital O with slash
Ù	13/09	D9	217	331	Capital U with grave accent
Ú	13/10	DA	218	33	Capital U with acute accent
Û	13/11	DB	219	333	Capital U with circumflex accent
Ü	13/12	DC	220	334	Capital U with diaeresis or umlaut mark

Table A-3 (Cont.)

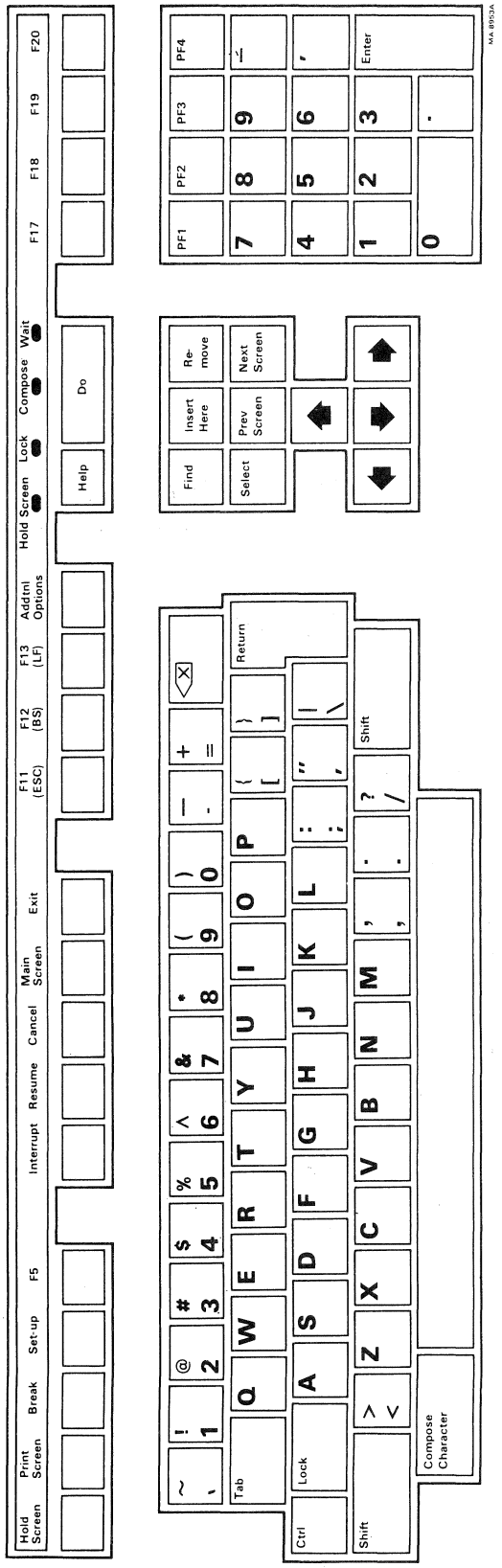
Char	*Col/Row*	*Hex.*	*Decimal*	*Octal*	*Description*
Ÿ	13/13	DD	221	335	Capital Y with diaeresis or umlaut mark
	13/14	DE	222	336	Reserved
ß	13/15	DF	223	337	German small sharp s
à	14/00	E0	224	340	Small a with grave accent
á	14/01	E1	225	341	Small a with acute accent
â	14/02	E2	226	342	Small a with circumflex accent
ã	14/03	E3	227	343	Small a with tilde
ä	14/04	E4	228	344	Small a with diaeresis or umlaut mark
å	14/05	E5	229	345	Small a with ring
æ	14/06	E6	230	346	Small ae ligature
ç	14/07	E7	231	347	Small c with cedilla
è	14/08	E8	232	350	Small e with grave accent
é	14/09	E9	233	351	Small e with acute accent
ê	14/10	EA	234	352	Small e with circumflex accent
ë	14/11	EB	235	353	Small e with diaeresis or umlaut mark
ì	14/12	EC	236	354	Small i with grave accent
í	14/13	ED	237	355	Small i with acute accent
î	14/14	EE	238	356	Small i with circumflex accent
ï	14/15	EF	239	357	Small i with diaeresis or umlaut mark
	15/00	F0	240	360	Reserved
ñ	15/01	F1	241	361	Small n with tilde
ò	15/02	F2	242	362	Small o with grave accent
ó	15/03	F3	243	363	Small o with acute accent
ô	15/04	F4	244	364	Small o with circumflex accent
õ	15/05	F5	245	365	Small o with tilde
ö	15/06	F6	246	366	Small o with diaeresis or umlaut mark
œ	15/07	F7	247	367	Small oe ligature
ø	15/08	F8	248	370	Small o with slash
ù	15/09	F9	249	371	Small u with grave accent
ú	15/10	FA	250	372	Small u with acute accent
û	15/11	FB	251	373	Small u with circumflex accent
ü	15/12	FC	252	374	Small u with diaeresis or umlaut mark
ÿ	15/13	FD	253	375	Small y with diaeresis or umlaut mark
	15/14	FE	254	376	Reserved
	15/15	FF	255	377	Reserved

APPENDIX B
KEYBOARD ILLUSTRATIONS

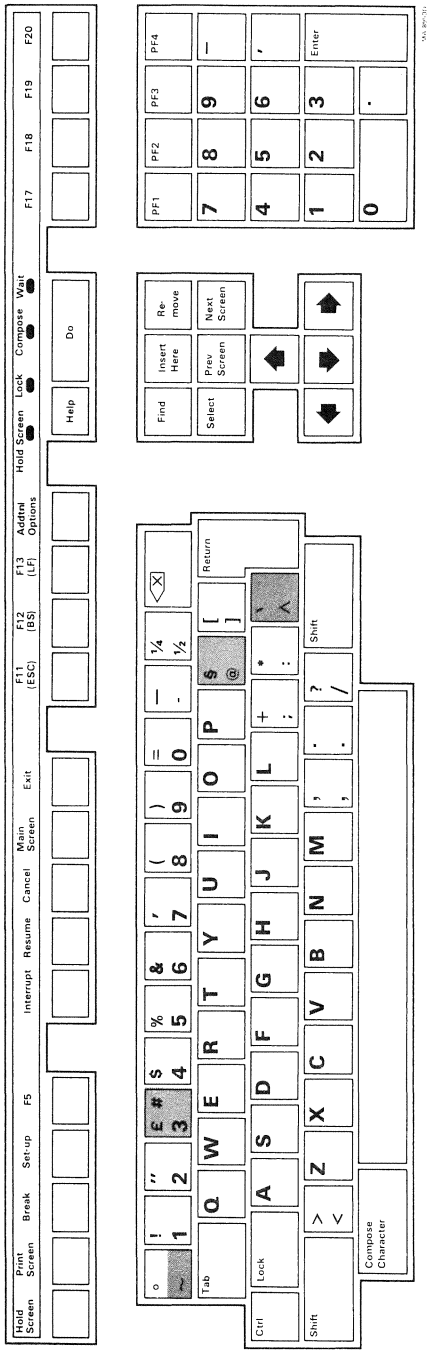
Illustrations of all national keyboards for the Professional are on the next several pages, in the following order:

United States/Canada
United Kingdom
Belgium (Flemish)
Canada (French)
Denmark
Finland
France/Belgium
Germany/Austria
Holland
Italy
Norway
Spain
Sweden
Switzerland (French)
Switzerland (German)

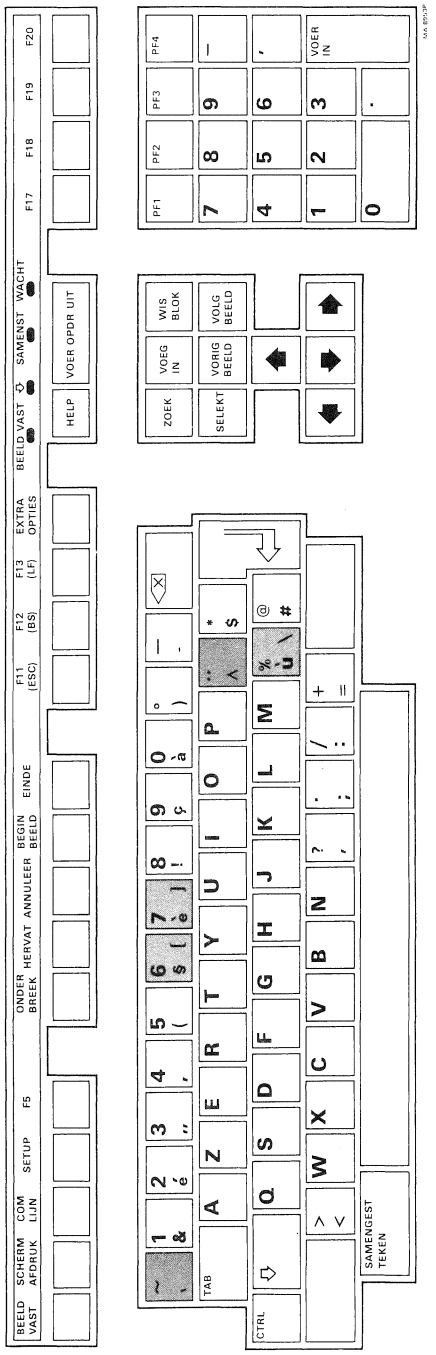
Keys that generate different characters in Office mode and Data Processing mode are shaded in red. "Dead diacritical" keys are shaded in gray. Chapter 2 describes the use of these keys.



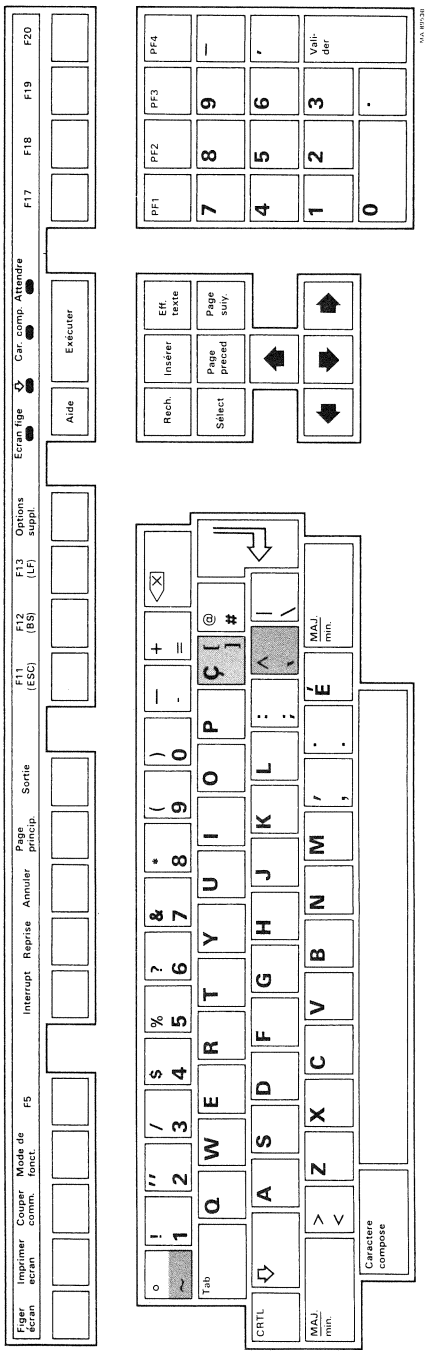
UNITED STATES/CANADA



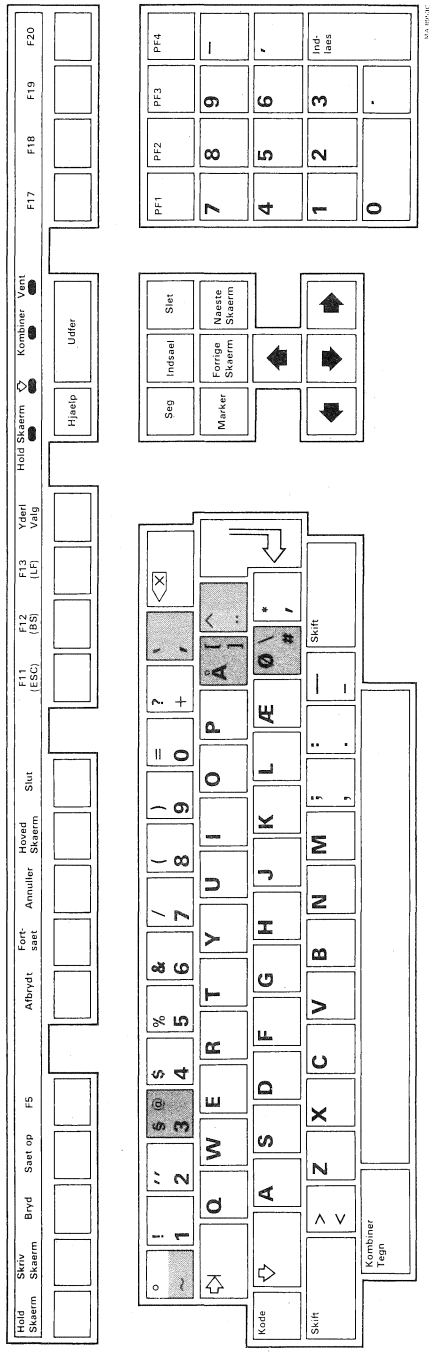
UNITED KINGDOM



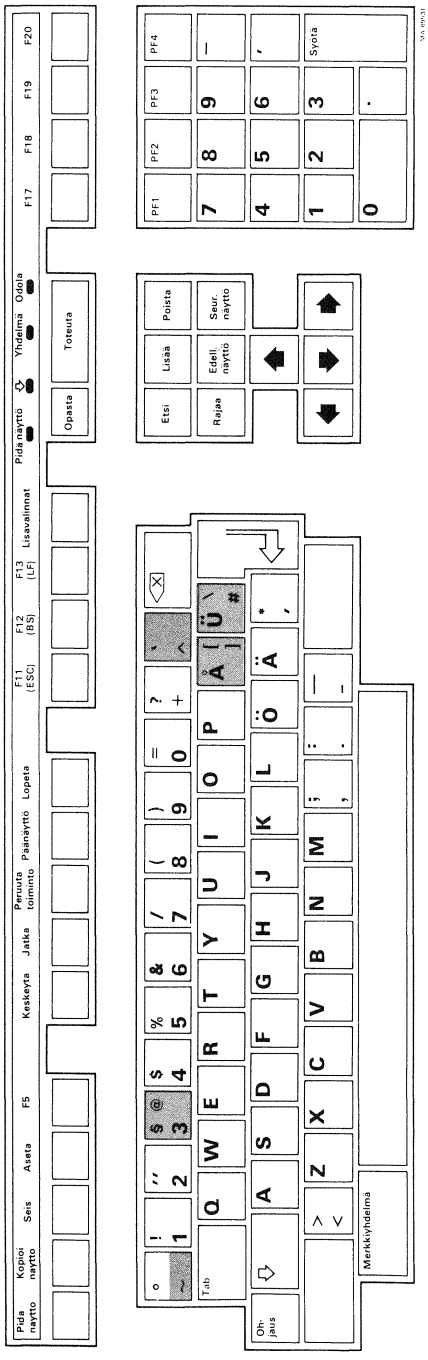
BELGIUM (FLEMISH)



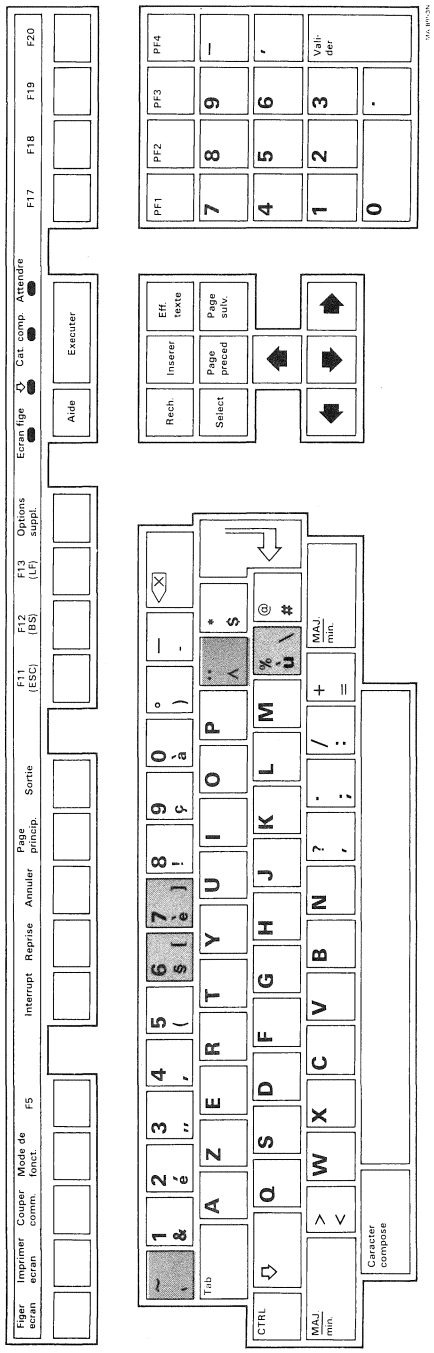
CANADA (FRENCH)


Hold Skaerm
Skriv Skaerm
Bryd
Saet op
F5
Afbrydt
Fort-saet
Annuller
Hoved Skaerm
Slut
F11 (ESC)
F12 (BS)
F13 (LF)
Yderl Valg
Hold Skaerm
Kombiner
Vent
F17
F18
F19
F20
Hjaelp
Udfer
Seg
Indsael
Slet
Marker
Forrige Skaerm
Naeste Skaerm
PF1
PF2
PF3
PF4
Kode
Skift
Kombiner Tegn
Skift
Ind-laes

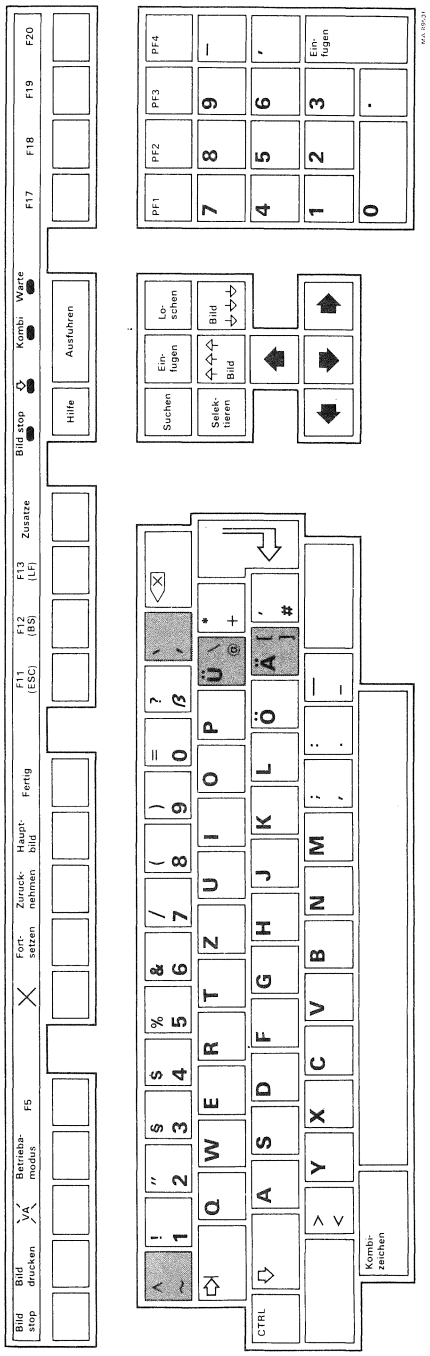
DENMARK


Pida näytto
Kopioi näytto
Seis
Aseta
F5
Keskeyta
Jatka
Peruuta toiminto
Päänäyttö
Lopeta
F11 (ESC)
F12 (BS)
F13 (LF)
Lisävalinnat
Pida näytto
Yhdeima
Odota
F17
F18
F19
F20
Opasta
Toteuta
Etsi
Lisää
Poista
Rajaa
Edell. näyttö
Saur. näyttö
PF1
PF2
PF3
PF4
Syota
Tab
Oh. jaus
Merkkivhdelma

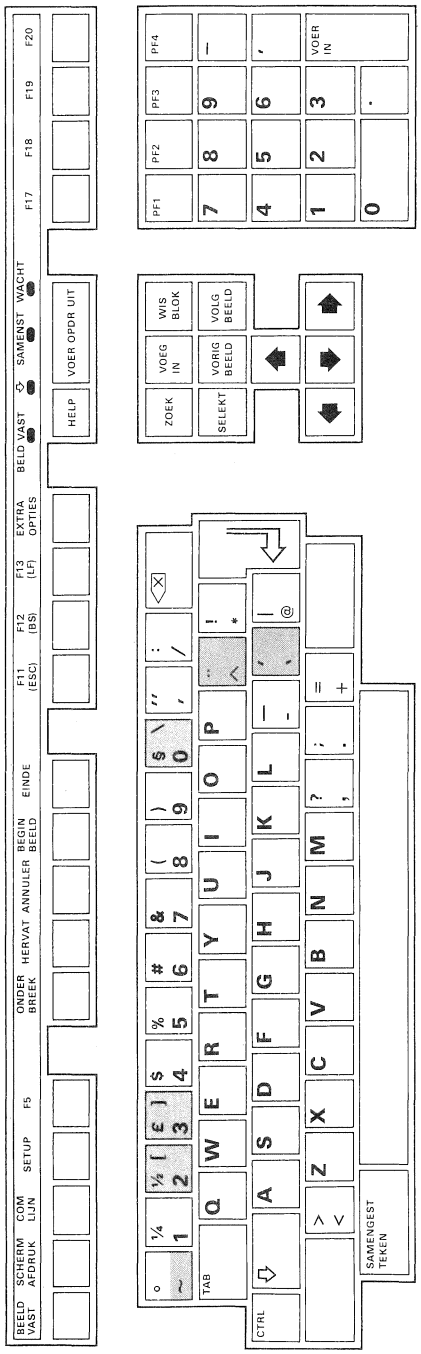
FINLAND



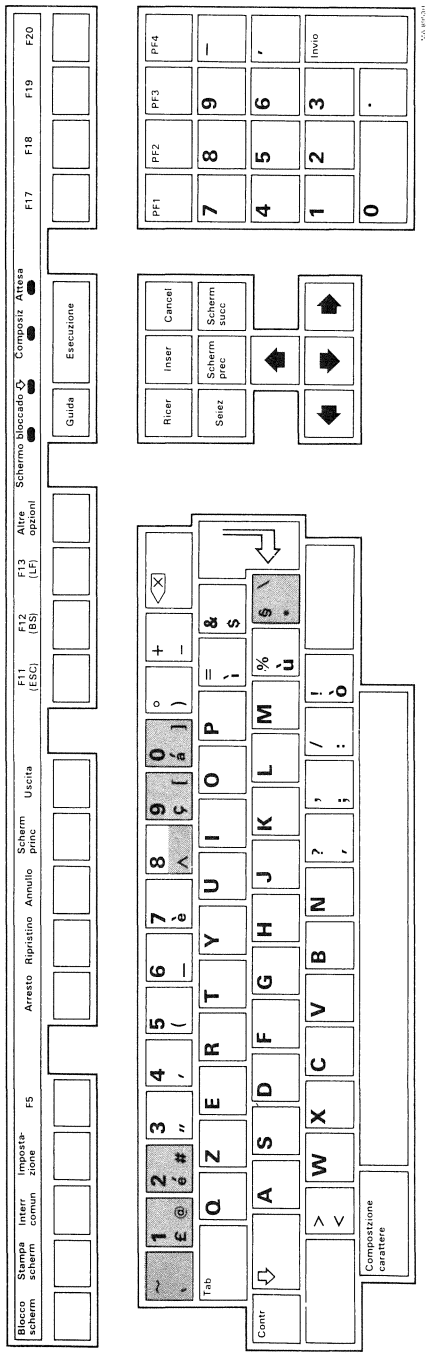
FRANCE/BELGIUM



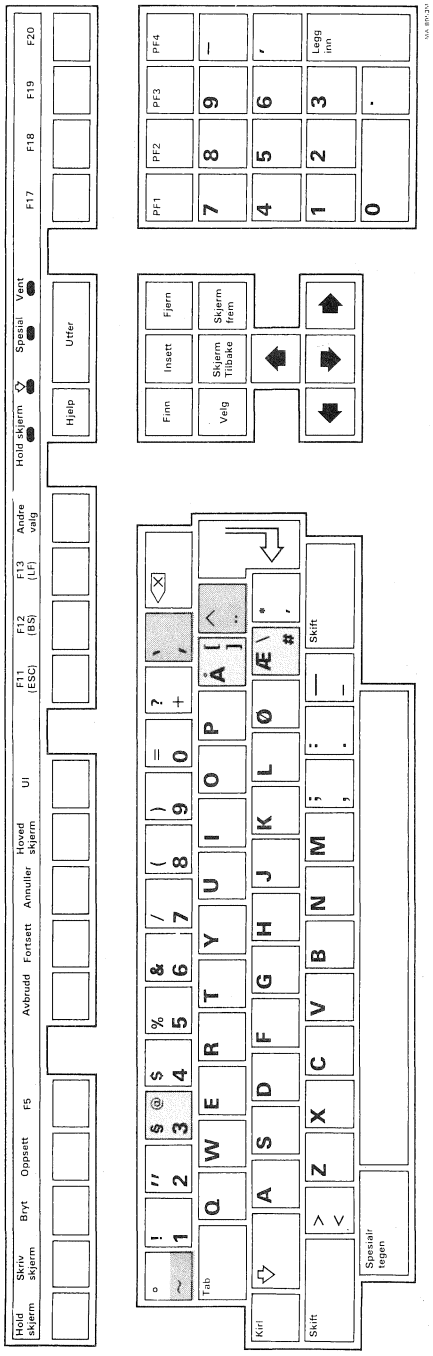
GERMANY/AUSTRIA


BEELD VAST
SCHERM AFDRUK
COM LIJN
SETUP
F5
ONDER BREEK
HERVAT
ANNULER
BEGIN BEELD
EINDE
F11 (ESC)
F12 (BS)
F13 (LF)
EXTRA OPTIES
BELD VAST
SAMENST
WACHT
F17
F18
F19
F20
HELP
VOER OPDR UIT
TAB
CTRL
SAMENGEST TEKEN
ZOEK
SELEKT
VOEG IN
VORIG BEELD
WIS BLOK
VOLG BEELD
PF1
PF2
PF3
PF4
VOER IN

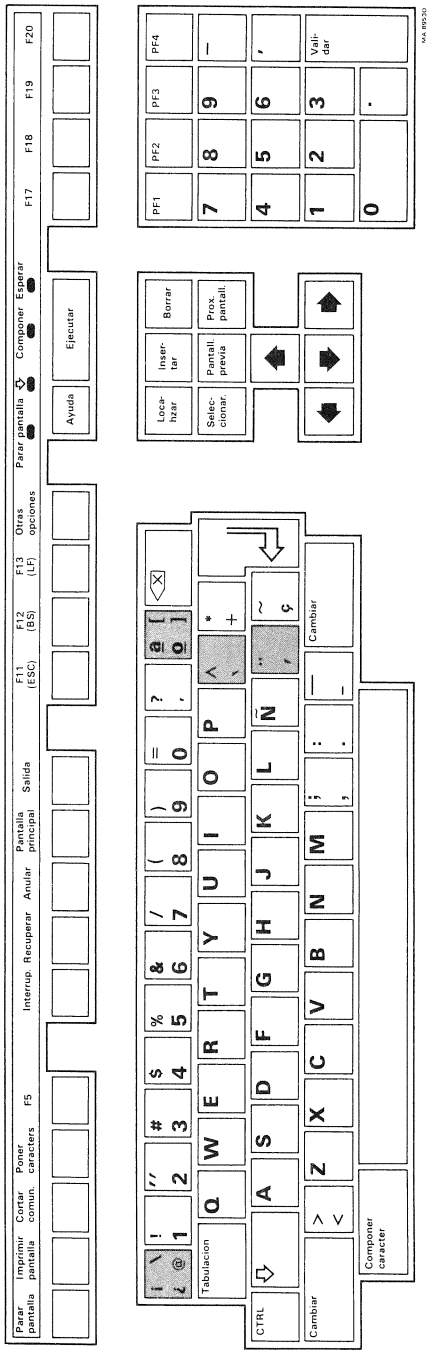
HOLLAND


Blocco scherm
Stampa scherm
Inter comun
Imposta-zione
F5
Arresto
Ripristino
Annullo
Scherm princ
Uscita
F11 (ESC)
F12 (BS)
F13 (LF)
Altre opzioni
Schermo bloccato
Composiz
Attesa
F17
F18
F19
F20
Guida
Esecuzione
Ricer
Inser
Cancel
Selez
Scherm prec
Scherm succ
PF1
PF2
PF3
PF4
Tab
Contr
Compostzione carattere
Invio

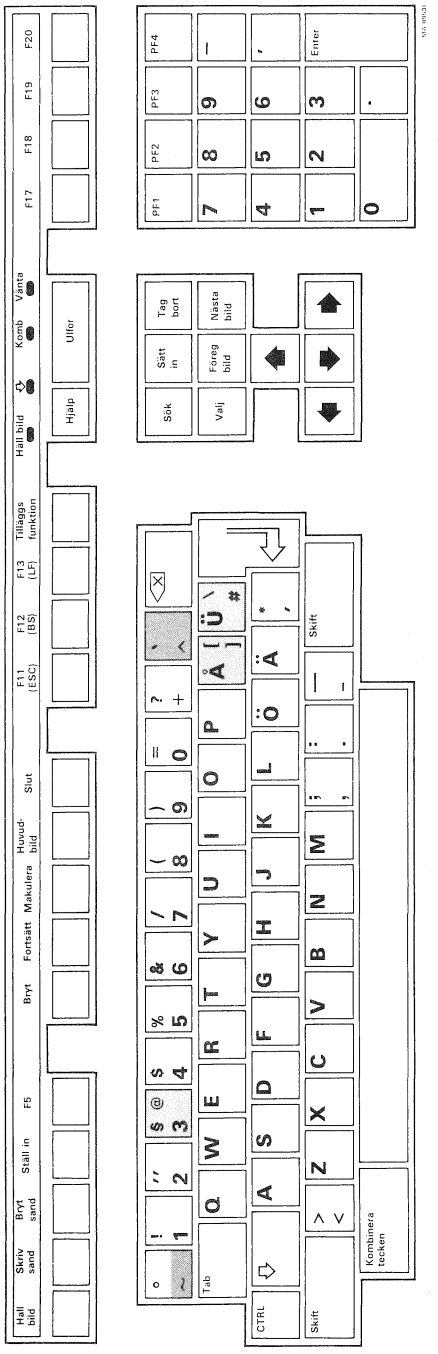
ITALY


Hold skjerm
Skriv skjerm
Bryt
Oppsett
F5
Avbrudd
Fortsett
Annuller
Hoved skjerm
UI
F11 (ESC)
F12 (BS)
F13 (LF)
Andre valg
Hold skjerm
Spesial
Vent
F17
F18
F19
F20
Tab
Kirl
Skift
Spesial/ tegen
Skift
Hjelp
Utfør
Finn
Insett
Fjern
Velg
Skjerm Tilbake
Skjerm frem
PF1
PF2
PF3
PF4
Legg inn

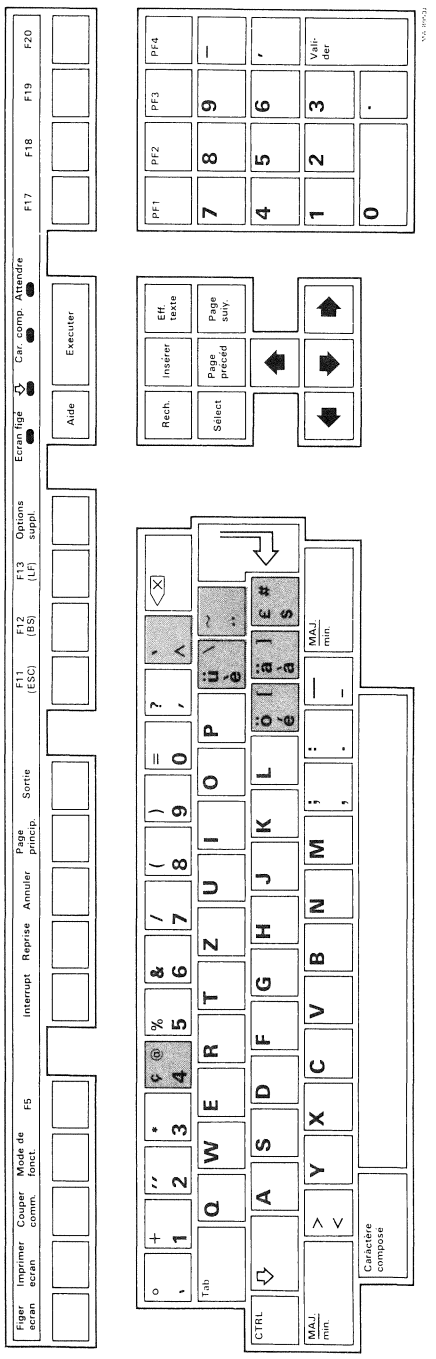
NORWAY


Parar pantalla
Imprimir pantalla
Cortar comun.
Poner caracters
F5
Interrup.
Recuperar
Anular
Pantalla principal
Salida
F11 (ESC)
F12 (BS)
F13 (LF)
Otras opciones
Parar pantalla
Componer
Esperar
F17
F18
F19
F20
Ayuda
Ejecutar
Tabulacion
CTRL
Cambiar
Componer caracter
Cambiar
Borrar
Inser-tar
Loca-lizar
Pantall previa
Prox. pantall.
Selec-cionar.
PF1
PF2
PF3
PF4
Vali-dar
MA 000320

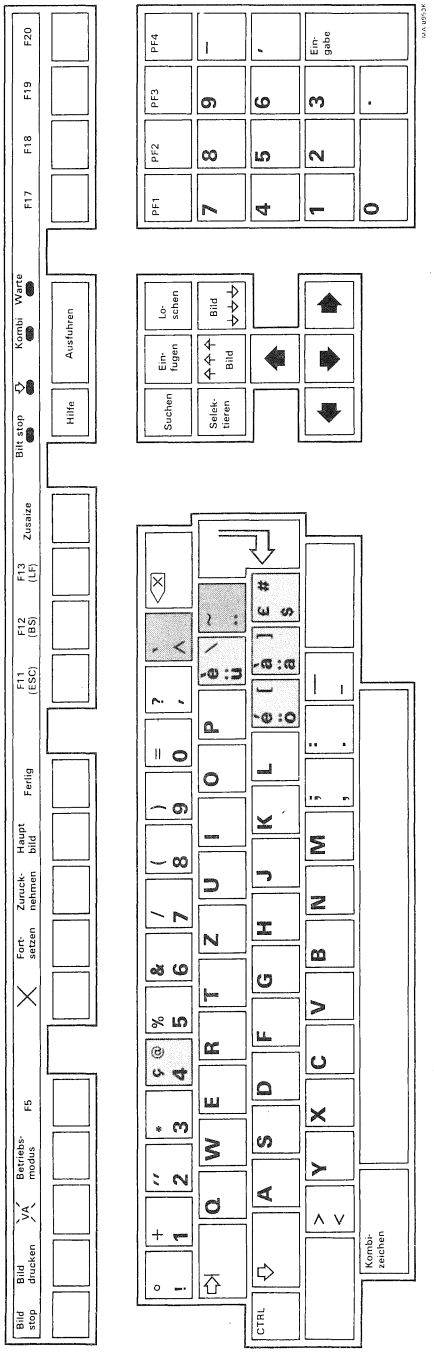
SPAIN


Hall bild
Skriv sand
Bryt sand
Stall in
F5
Bryt
Fortsätt
Makulera
Huvud-bild
Slut
F11 (ESC)
F12 (BS)
F13 (LF)
Tilläggs funktion
Hall bild
Komb
Vänta
F17
F18
F19
F20
Hjälp
Utför
Sök
Sätt in
Tag bort
Välj
Föreg bild
Nästa bild
PF1
PF2
PF3
PF4
Enter
Tab
CTRL
Skift
Kombinera tecken
Skift

SWEDEN



SWITZERLAND (FRENCH)



SWITZERLAND (GERMAN)

APPENDIX C
TEXT AND GRAPHICS MODE INTERACTION

You can mix text and graphics in your application. But to do it properly, you need to understand the behavior of text and graphics modes. Here are some guidelines.

There is currently no standard which addresses the interaction or integration of text mode and graphics mode. A future standard or different video hardware will probably affect the type of integration that is supported on the Professional.

On the Professional, if you write text (in text mode) over graphics, the graphics image beneath that text will be lost. This is because characters in text mode are written as cells.

Graphics written over text will destroy only the text area which it affects directly. If you want to write text over graphics, it is recommended that you use graphics mode text.

If you want to mix text (text mode) and graphics on the same display, it is recommended that you use each character line defined for text mode exclusively for text or graphics, but not both.

Before using several lines of the display for a graphics region, clear those lines with a text-mode command. This will prevent text-mode software from behaving as if there were still characters within the region.

Also, if the underline cursor is moved over a character cell containing graphics, the graphics content of the cell will be lost.

Text mode and graphics mode have different cursor presentations. Whenever the video display is idle, a cursor is usually presented on the display. You get a graphics-mode pixel-oriented cursor (for example a cross hair) if the most recent data sent to the Terminal Subsystem was for graphics. You get a text-mode character-cell-oriented cursor otherwise.

Text-mode character cells which are blinking continue to blink when graphics mode is entered, even if the characters are altered or erased by a graphics-mode command. You should erase such cells by a text-mode command before displaying the graphics.

If the extended bitmap option (EBO) is not on the Professional, only one bitmap plane is available, and graphics and text are forced to use the same plane. With the EBO present, three bitmap planes are available. The behavior of text and graphics integration may differ depending on whether or not the EBO is present. To emulate a VT125 terminal, for example, your application should restrict graphics mode to two planes.

If the EBO is installed on the Professional, receipt of an RIS (reset to initial state) sequence defaults the output map to one of the following sets of settings depending on whether a color or monochrome monitor is present.

Entry	*Color*	*Monochrome*
0	black	intensity-0
1	blue	intensity-2
2	red	intensity-4
3	green	intensity-6
4	white	intensity-6
5	white	intensity-6
6	white	intensity-6
7	white	intensity-6

These defaults are consistent with a VT125 terminal. Text mode affects only one plane corresponding to output map value of 4. Text mode operations do not modify the output map but are affected by the settings made while in graphics mode.

If the EBO is absent, the graphics content of a line is lost if the line is erased or scrolled off the screen while in text mode. If the EBO is present, the graphics content of a line is lost only if the line is scrolled off the screen when doing full-screen scroll.

APPENDIX D
DIFFERENCES IN EMULATED FEATURES

Differences between VT102 and VT125 terminals and the emulated features on the Professional are treated throughout this manual. This appendix groups the difference information in one place.

As in the rest of this manual, VT102 mode and VT125 mode refer to VT102 and VT125 terminal emulation as implemented on the Professional through the PRO/Communications application. In this appendix, only the graphics functions of VT125 mode are treated (on the Professional, VT125 text mode is essentially like VT102 mode).

D.1 VT102 MODE COMPARED WITH VT102 TERMINAL

Whenever the description of a VT102-mode characteristic applies equally to all ANSI-compatible text modes of operation on the Professional (VT102, Professional, and native modes), the expression "on the Professional" is used instead of "in VT102 mode".

For more information on a particular subject, refer to the applicable portion of this manual.

D.1.1 Seven- and Eight-Bit Support

VT102 mode is capable of fully supporting the 8-bit encoding system used on the Professional. This includes recognition of the C1 control characters and GR graphic characters.

In VT102 mode, if your application environment does not support 8-bit characters, your application can use code extension announcers causing the Terminal Subsystem to convert any C1 codes returned to the application to their equivalent 7-bit code extensions. No such conversion is possible with GR codes however: they always require an 8-bit environment.

In any mode, the Professional always interprets the eighth bit of any received character as a legitimate value. In some 7-bit application environments, this could cause the Professional to accept invalid data. To avoid this problem, the Terminal Emulator Set-Up menu lets you restrict the communications link to a 7-bit format so that any potentially invalid eighth bit values will be dropped.

D.1.2 Communications

A VT102 terminal provides both half duplex and full duplex communication modes. The communications services provided in VT102 mode on the Professional are for full duplex mode only.

D.1.3 Keyboard

All keys used on a VT102 terminal are represented on the Professional keyboard in VT102 mode and have equivalent functions/codes. The following differences exist.

On the Professional, the cursor control keys are on the editing keypad, and the **HOLD SCREEN, PRINT SCREEN**, and **BREAK** keys are top-row function keys. **HOLD SCREEN** is equivalent to **NO SCROLL** on the main array of a VT102 Terminal, except that pressing any key, including **HOLD SCREEN** will re-enable scrolling in VT102 mode. (the resulting code is sent to the application).

Pressing the **PRINT SCREEN** key on the Professional is not equivalent to printing the screen on a VT102 terminal (simultaneously pressing the **SHIFT** and **(PRINT)/ENTER** keys): **PRINT SCREEN** may print graphics, and **PRINT SCREEN** is not affected by the DECPEX or DECPFF commands.

The **ESC, BACK SPACE**, and **LINE FEED** keys on the main array of a VT102 terminal do not exist on the main array of the Professional. In VT102 mode, the following top-row function keys are used instead: **F11 (ESC), F12 (BS), F13 (LF)**. These keys generate the standard control codes for ESC, BS, and LF, respectively.

D.1.4 Codes Transmitted from Keyboard to Application

The codes transmitted to an application from the keyboard differ between VT102 mode and a VT102 terminal as follows.

The Professional supports over a dozen national keyboards. Most keyboards have keys that produce some GR (8-bit) graphic characters. Also, GR characters can be generated from all keyboards by means of Compose sequences. If the application environment does not support 8-bit codes, no GR characters can be transmitted.

The following C0 (7-bit) control characters, generated from the keyboard with the **CTRL** key, are created differently:

Control Character Mnemonic	*Key Pressed with* **CTRL** *(VT102 Mode)*	*Key Pressed with* **CTRL** *(VT102 Terminal)*
NUL	2	space bar
ESC	3	[
FS	4	/
GS	5	]
RS	6	~
US	7	?

On the Professional, your application can use code extension announcers to control whether function-key codes returned to the application use C1 control characters or their equivalent 7-bit code extensions. On a VT102 terminal, all function-key codes are transmitted using 7-bit code extensions.

On a VT102 terminal, the Set-Up function is asynchronous and may be invoked at any time without affecting the current context. On the Professional, Set-Up is not asynchronous.

On the Professional, if two or more keys are held down, the most recently depressed key is selected for autorepeat.

D.1.5 Response of "Terminal" to Received Codes

D.1.5.1 Control Characters—The following C0 (7-bit) control characters are processed differently in VT102 mode than on a VT102 terminal.

ETX	Ignored on the Professional.
EOT	Ignored on the Professional.
ENQ	No answerback message is generated on the Professional. On a VT102 terminal, ENQ causes an answerback message to be returned, if stored.
CAN	No error character is displayed on the Professional. On a VT102 terminal, CAN causes an error character to appear on the screen.
SUB	Causes a different error character (reverse question mark) to be displayed on the Professional as compared to on a VT102 terminal.

On the Professional, the following C1 (8-bit) control characters sent from the application are recognized by the Professional: IND, NEL, HTS, RI, SS2, SS3, DCS, CSI, OSC, PM, APC, ST. The equivalent 7-bit code extensions are also recognized.

D.1.5.2 Control Strings—A control string consists of an opening delimiter (DCS, OSC, PM, or APC), a command string, and the string terminator (ST) closing delimiter. A VT102 terminal does not recognize control strings: the command strings are merely displayed on the screen and no action is taken. On the Professional, control strings are recognized but ignored (hence are not displayed), except for ReGIS control strings which are recognized and processed in VT125 mode.

The opening delimiter for ReGIS control strings is DCS. DCS transmitted from a host application is used as the introducer of ReGIS strings that are then executed in graphics (VT125) mode on the Professional.

D.1.5.3 Control Sequences—On the Professional, control sequences may contain as many as sixteen (16) parameters. Additional parameters are ignored. Each parameter may have a value in the range of zero (0) to 65535. Any value over this range is treated as the maximum value. Also, within the context of a particular command, values which exceed the maximum values for the command (for example 24 or 80) are treated as the contextual maximum.

D.1.5.4 Set-Up Characteristics and Mode Selection—Much of the Set-Up functionality is different on the Professional from that on a VT102 terminal. On the Professional, the user interface to Set-Up is provided through the P/OS Set-Up menus for all modes and, additionally, through the PRO/Communications Set-Up menus for terminal emulation modes. The PRO/Communications menus handle the parameters for communications protocols. Also, Set-Up on the Professional is not asynchronous.

D.1.5.5 Scrolling—On the Professional, smooth scroll is possible only for the entire screen. Smooth split-screen scroll is not possible. The Professional uses jump split-screen scroll instead. Full-screen scroll affects all bitmap planes when the extended bitmap option is being used. Jump scroll affects only the plane that is used for text mode. No other video activity occurs while smooth scroll is in progress. This includes blinking of the cursor or other characters.

D.1.5.6 Cursor Characteristics—On the Professional, a graphics-mode pixel-oriented cursor (for example, a cross hair) appears on the screen if the most recent data sent to the Professional was in graphics mode. A text-mode character-cell-oriented cursor appears otherwise. The period for cursor blinking is one second (2/3 on, 1/3 off).

The Professional supports the cursor-enable (DECCEM) sequence, which is not supported on a VT102 terminal. This sequence affects the cursor in both text mode and graphics mode.

D.1.5.7 Character Sets and Selection—The character fonts, affecting the visual appearance of displayed characters, are different on the Professional from those on a VT102 terminal.

The 8-bit DEC Multinational Character Set is the default character set on the Professional. This maps the ASCII Graphics set in GL (7-bit) and the DEC Supplemental Graphics set in GR (8-bit). On a VT102 terminal, the 7-bit ASCII or U.K. set is the default set, depending on the Set-Up selection.

The Professional's graphic repertoire consists of the following three graphic sets: ASCII Graphics, DEC Supplemental Graphics, and DEC Special Graphics (VT100 Line Drawing). The VT102 terminal's graphic repertoire consists of the following graphic sets: ASCII, U.K., and DEC Special Graphics.

The Professional supports the following lock shifts: LS0, LS1, LS2, LS3, LS1R, LS2R, LS3R. A VT102 terminal supports only the LS0 (SI) and LS1 (SO) lock shifts. Both the Professional and a VT102 terminal support the SS2 and SS3 single shifts.

By default, the Professional's graphic character sets are designated and invoked automatically as follows:

Designate	*as*	*and invoke*	*into*
ASCII Graphics	G0	G0	GL
DEC Special Graphics	G1		
DEC Suppl. Graphics	G2	G2	GR

The Professional allows the designation of the G0, G1, G2 and G3 character sets. However, the default designation of G3 is unspecified.

On a VT102 terminal, G0 and G1 are the only sets you can designate. You then use SI to invoke G0 into GL and SO to invoke G1 into GL. G2 and G3 can not be designated by program control on a VT102 terminal. A VT102 terminal always interprets G2 and G3 as the default graphic character sets. They are selected in Set Up. You can invoke G2 and G3 for only one character at a time, using SS2 and SS3, after which you return to the previous character set. Software that relies on the VT102's "permanent" designation of G2 and G3 may be incompatible with the Professional.

D.1.5.8 Character Attributes—Differences in character attributes are as follows:

Reverse Video

On the Professional, normal-intensity reverse-video characters are black on a white background, and bold-intensity reverse-video characters are gray on white. A VT102 terminal uses black on gray and black on white respectively.

Bold

On the Professional, bold (increased intensity) is not available in 132-column mode and is ignored.

Blink

On the Professional, you can have up to 100 blink fields (a field consists of adjacent blinking character cells within a single horizontal line). If you exceed this number, all blink fields stop blinking until the overflow condition ceases. Blinking character cells continue to blink while displaying graphics. If this is not desired, you should erase such cells before displaying graphics. The period for character cell blinking is four thirds of a second, and the duty cycle is 50 percent. Normal-intensity characters are produced.

D.1.5.9 Printing—Differences in printing are as follows.

Printer Controller Mode

The media copy (MC) printer controller mode is implemented differently on the Professional versus on a VT102 terminal.

On a VT102 terminal, you enter printer controller mode only if the printer is ready. On the Professional, you always enter printer controller mode whenever the command is issued.

On the Professional, the printer controller mode sequence turns on printer controller mode and sends CAN (1/8) to the printer. The Professional transmits received characters, including RIS sequences, to the printer without displaying them on the screen. All characters and character sequences except NUL, CSI 5 i, and CSI 4 i are sent to the printer. The Professional does not insert or delete spaces, or provide line delimiters, or select the correct printer character set. This data path is unidirectional, and therefore an application cannot receive data from the printer while printer controller mode is on.

Printer controller mode sequences that are used to turn that mode on or off are not sent to the printer. All other sequences are. To turn printer controller mode off, you must send the above sequence exactly as indicated or its equivalent 7-bit format (ESC [4 i).

Printer Form Feed Mode (DECPFF)

On the Professional, no space is cleared between print screens if DECPFF is reset (off). On a VT102 terminal, there is one additional linefeed between print screens.

Print Screen (Text or Graphics Mode)

The DECHCP (Hard Copy) sequence is not recognized on a VT102 terminal. In VT102 mode, your application can use the DECHCP (Hard Copy) sequence to print the screen contents. The DECPEX and DECPFF modes have no effect on DECHCP. The entire screen is printed, without any spacing between print screens. Pressing the **Print Screen** key has the same effect as using the DECHCP sequence. If the printer is unavailable or busy when the DECHCP sequence is sent, the print request will still be assumed completed.

If there is a possibility that the display contains graphics, a pixel-wise dump is used rather than a character-cell-wise dump. Whenever entering graphics mode, all text-mode lines are considered to contain a graphics image. This attribute is maintained on a line-by-line basis. It is cleared for lines which scroll off during full-screen scroll, or for lines which are erased or scrolled off if the extended bitmap option is not installed on the Professional, or for all lines after the Professional receives an RIS (reset to initial state) sequence. When a pixel-wise dump to the screen is performed, all picture elements that are not turned off (black) on the display have a corresponding black dot on the printout. A pixel-wise dump of text appears very differently than a cell-wise dump of the same text.

D.1.5.10 Reports—Differences in reports are as follows.

D.1.5.10.1 Device Status Report (DSR)—On a VT102 terminal, one of the possible responses to a DSR printer status request indicates that the DTR (data-terminal-ready) signal of the printer has not been on (asserted) since the terminal was last turned on or reset. The Professional's corresponding response is: The printer is not ready and never has been, when tested, since the last time the Professional was booted.

D.1.5.10.2 Device Attributes (DA)—In VT102 mode, the Professional's response to a DA request is the same as the response of a VT102 terminal:

ESC [? 6 c

D.1.5.11 Reset—On the Professional, the reset-to-initial-state (RIS) sequence initializes the Terminal Subsystem to default states and clears the screen.

RIS affects the text-mode states, graphics-mode states, and keyboard. Except for two keyboard states (Compose mode on/off, keyboard lock/unlock), RIS affects only the states that can be set by an application. RIS has no effect on states such as printer status or cursor block/underline, that cannot be changed from an application. On a VT102 terminal, RIS resets all states (programmable or not, such as "keyclick").

On the Professional, when RIS is selected, Compose mode is off and the keyboard becomes unlocked even if it were locked by KAM or DC3.

The defaults states selected by the RIS sequences are listed in Chapter 3.

D.1.5.12 Interlace—On the Professional, your application can enable and disable interlacing of the raster scan by specifying the DECINLM sequence. DECINLM is not supported on a VT102 terminal.

D.1.5.13 Tests and Adjustments—The self tests initiated with the DECTST sequence on a VT102 terminal are not supported on the Professional. However, much of this functionality (including data loop-back) is provided in the Professional's power up diagnostics sequence.

On the Professional and on a VT102 terminal, an application can use the DECALN sequence to display a screen alignment pattern. On the Professional, DECALN displays rectangles on the screen. On a VT102 terminal DECALN displays capital "E"s.

D.1.5.14 Keyboard Indicator—In VT102 mode, there is no user-loadable LED indicator on the keyboard, and therefore there is no support for turning one on or off. The DECLL (load LED) sequence is used on a VT102 terminal to turn an LED indicator on or off.

D.1.6 Other Features

The bell can be disabled on the Professional.

On the Professional, any screen display will disappear (video blanking) if there has been no activity in the Terminal Subsystem for 30 minutes.

There is no support for a 50-Hz screen refresh rate on the Professional. Only the 60-Hz rate is supported.

There is no support for the "word processing flag" setup feature on the Professional. This feature reverses the codes sent by the **LINEFEED** key and the |\ (vertical bar, backslash) key.

D.2 VT125 MODE COMPARED WITH VT125 TERMINAL

The Professional's VT125 graphics mode implements a few VT125 features differently. The variations result both from recent updates to the Remote Graphics Instruction Set (ReGIS) and from differences that exist between the Professional hardware and VT125 terminal hardware.

Default values for VT125 graphics mode on the Professional are compatible with the default values for a VT125 terminal. However, because the Professional must provide graphics functionality for the CORE Graphics Library and for PRO/BASIC, the variations described in the following sections can occur when previously developed VT125 ReGIS applications execute on a Professional in VT125 graphics mode.

A VT125 terminal has a single text plane and two graphics planes. Therefore, on the Professional, the extended bitmap option (EBO) is required to emulate a VT125 terminal and prevent text and graphics overwriting each other. Also, due to hardware/software environment differences, it sometimes takes longer for the Professional to execute the same VT125 graphics function.

D.2.1 Image Distortion Factors

The Professional's 960 $\times$ 240 pixel display density provides a 2.5-to-1 pixel-aspect ratio, while a VT125 terminal has a 768 $\times$ 240 pixel density resulting in a 2.0-to-1 pixel-aspect ratio. The increased density of the Professional is

designed for high-resolution graphic images, rather than the medium-resolution graphics of the VT125. The following subsections describe some of the imaging differences caused by the Professional's improved density.

D.2.1.1 Distortion Scaling—When certain screen-addressing parameters are used with the VT125 terminal, circles (drawn using the ReGIS C command) appear round, but squares (drawn using four equal-length arguments to the ReGIS V command) do not appear square. Similarly, the VT125 does not always represent angles faithfully. These inconsistencies are not reproduced in the Professional; its circles always are round, its squares square, and its angles correctly drawn, regardless of the screen-addressing parameters.

An image that intentionally exploits the distortion in the VT125's screen-addressing mapping is not distorted in the Professional. Note that, when using the VT125 default screen-addressing parameters, this incompatibility does not arise because the default screen addressing uses the entire screen area without distortion.

D.2.1.2 Pattern Registration—Standard (predefined) VT125 patterns 7, 8, and 9 are represented in the following table. Note that on the Professional, these patterns are shifted left with the most significant bit set, an aid in pattern registration. Only minor differences are expected when these patterns display on the Professional.

Pattern No.	*VT125 Bit Pattern*	*Professional Bit Pattern*
7	00100001	10000100
8	00011001	11001000
9	10000110	10000110

D.2.2 Shading Differences

D.2.2.1 Two New Shading Options—The Professional introduces two additional shading options: (1) shading to a vertical line (the W(S[x]) command) and (2) shading to a point (the W(S[x,y]) command). Existing VT125 applications are not affected because the Professional's default shading is to a horizontal line, the VT125 terminal's only shading option.

D.2.2.2 Character Shading and Text Attributes—For shading with characters, Professional capabilities match VT125 terminal attributes with one exception. The Professional does not use the italic attribute when drawing a shaded area with character fill. Any VT125 application using italic-attribute character fill appears without the italics when run on the Professional. All other character-fill options (size attributes including height, multiplier, and integral size) are supported.

The italic effect can be achieved by loading a tilted character into a loadable character location and then specifying that character as the fill character. This change produces the same effect on both the Professional and the VT125 terminal.

D.2.3 Screen-Addressing Considerations

D.2.3.1 Pixel-Aspect Ratio—When VT125 default screen addressing is used on the Professional (768 × 480 logical pixels) some horizontal pixels are duplicated (about one in four) to produce the same VT125 picture. For applications to take advantage of the increased resolution of the Professional (with a more direct logical to physical pixel mapping ability), use an explicit screen addressing of 960 ×600.

D.2.3.2 General Screen-Scaling Method—The Professional uses a more general scaling algorithm than the VT125 terminal when mapping user-specified screen-addressing parameters with the physical screen. Therefore, images might use more of the physical display area when displayed on the Professional.

The VT125 terminal uses integral multiples or integral divisions when mapping explicit screen-addressing parameters to physical pixels. This method can waste a substantial portion of the display area. For example, a 1024H × 1024V addressing scheme uses 512H × 256V logical pixels when mapped on a VT125 terminal. The same scheme uses 480H ×480V logical pixels when mapped onto the Professional. Note that the Professional does not distort the picture-aspect ratio while the VT125 distorts at a 2:1 ratio.

D.2.3.3 Decimal-Fraction Support—The Professional permits the use of decimal fractions in screen-addressing parameters and coordinates while in VT125 graphics mode. Fractional digits are significant, subject to the specific image resolution. This feature facilitates the use of the unit-square addressing (0 to 1 coordinates).

For programs designed for the VT125, previously ignored decimal portions of screen-addressing parameters become significant on the Professional. Integer parameter values are not affected.

D.2.3.4 Clipping Offscreen Areas—The VT125 terminal's memory can store images in a 768 × 256 physical pixel array, an area addressable with ReGIS commands. The VT125's display provides a window that can display 768 × 240 physical pixels out of the 768 × 256 pixels; there is an extra 16 horizontal pixel scan lines that are addressable and displayable.

In contrast, the Professional defines the addressable bitmap memory as 960 × 240 physical pixels. Areas of memory outside the vertical 240-line range are clipped and cannot be addressed, displayed, or revealed by scrolling. With default screen-addressing parameters, this incompatibility does not occur, since there is no viewable screen area beyond the defined 768 × 240 physical pixel display. Moreover, explicit screen-scrolling commands to memory areas outside the 240-pixel vertical limit result in a display of the current background color instead of any hidden pixel image.

D.2.3.5 Graphics Processor Memory Position Stack Size—The Professional has a 16-position stack for storing graphics processor memory locations, as opposed to the VT125's 10-position stack. Note that an application using position-stack overflow keyed to overflow on the eleventh entry can execute incorrectly when running on the Professional. The ReGIS V(B)...(E) and P(B)...(E) commands are used to perform stack manipulation.

D.2.4 Scrolling Differences

D.2.4.1 Text/Graphics Independent Scrolling—The Professional does not permit independent scrolling of text and graphics images. Different bitmap planes cannot scroll independently, and, with the extended bitmap (EBO) option, a full-screen scroll affects all bitmap planes.

Although a VT125 terminal permits text and graphics to scroll separately (the ReGIS screen (S) command), programs utilizing this VT125 feature (not recommended) do not operate correctly when run on the Professional.

D.2.4.2 No Wrap-Around Scrolling—Once a graphics image scrolls off the Professional screen, the image is lost; it cannot be retrieved unless it is redrawn. The Professional does not support either horizontal or vertical image wrap-around. Although the VT125 terminal wraps around both horizontally and vertically, wrap-around is deemed undesirable on the Professional because text and graphics share the same bitmap.

D.2.4.3 Fixed-Origin Scrolling—The origin point of a graphics image remains fixed to the screen in VT125 graphics mode on the Professional. The origin is the 0,0 point defined by the ReGIS screen-addressing (S(A)) command. On a VT125 terminal, the origin point moves with the image (fixed to the bitmap) to take advantage of scrolling for offscreen memory.

While operating in the Professional's VT125 graphics mode, image handling requires that the application maintain a scroll offset in order to determine locations relative to the origin point. Note that specific screen locations on the Professional always retain the same address and simplify the generation of strip charts.

D.2.5 Graphics Text Variations

A ReGIS application program that explicitly specifies text size (rather than using the text standard sizes) should be transportable between a VT125 terminal and the Professional. It is also recommended that a ReGIS position report be solicited after drawing a long text string to determine the exact location of the current drawing point. Otherwise, misalignment of text with some other graphical images could result.

D.2.5.1 Color/Text/Graphics Interdependence—The Professional has three planes of bitmap memory and an eight-bit-wide color map, allowing display of up to eight simultaneous colors from a 256-color palette (eight levels of gray-scale on monochrome monitor). However, one of the three planes usually is dedicated for text. The VT125 provides a six-bit-wide color map with four simultaneous colors from a 64-color palette (four levels of gray-scale on monochrome).

On the Professional, when one plane is used for text the text display and graphics images can be written and erased independently. Also, a text display can be made to blink without affecting graphic images; however, independent scrolling cannot be done.

Destructive text and graphics action can occur when full graphics capability (eight simultaneous colors) is used. In this instance, text and graphic images might overwrite each other and destroy the previously drawn image.

D.2.5.2 Loadable Alphabets—Applications using VT125 loadable alphabets perform appropriately on the Professional. Up to three loadable alphabets of 95 characters each are permitted. The default VT125 alphabet characters have 10 rows of eight pixels each.

Additional alphabet features are available with the Professional. A maximum of 16 different loadable alphabets is allowed, contingent on available alphabet memory. A Professional alphabet's character count is variable, and an alphabet set can be defined as large as the memory allowed. Character height and width can be defined independently and can be specified in a range of one to 16 in each dimension.

D.2.5.3 Slashed/Unslashed Zero Character—To concur with ASCII standards, the Professional's VT125 graphics mode predefined character set contains a zero character (0) without a diagonal slash mark. On a VT125 terminal, zero characters are slashed to indicate a zero digit, rather than a capitalized letter O.

D.2.5.4 Display/Unit Cell Size Effect—A display cell is the rectangular screen area in which a drawn character resides as opposed to the unit cell, which is the rectangular screen area making up the character itself. Generally display cell size is slightly larger than unit cell size, providing a margin of space surrounding the character and preventing characters from merging together. When a graphics text character display cell size is significantly larger than the cell size defined for a unit cell, the Professional's display differs from the display on a VT125 terminal.

Figure D–1 illustrates the two different display types. This display reaction only occurs when you specify a markedly larger display cell size in relation to unit cell size. The Professional draws a single copy of the text character in the upper left corner of the display cell, as shown on the left in Figure D–1. The character is in the current unit cell size, and the rest of that display cell either fills with the background color or remains unchanged, depending on the current writing mode.

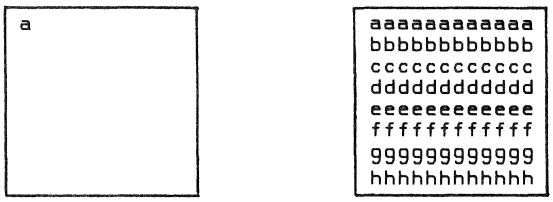


Figure D–1
Display/Unit-Cell Size Effect

A VT125 terminal draws as many copies of the text character at unit cell size as needed to fill the horizontal unit-size area of the display cell. In an attempt to complete the display cell, the VT125 terminal fills the vertical direction with whatever adjacent characters reside in character storage. Each vertical character repeats to fill the horizontal area as shown on the right in Figure D–1.

D.2.5.5 Specifying Unit-Cell Size—The Professional offers two methods for specifying the size of a graphics text unit cell. The unit-cell size can be specified indirectly with the multiplier T(M) option for both terminals. However, the Professional also permits direct specification of unit-cell size with the T(U) command. Modification of existing applications is not necessary; however, both the old and new methods are supported on the Professional.

D.2.5.6 Screen-Addressing Coordinates—Text-cell size on the Professional in VT125 graphics mode is relative to current screen addressing coordinates, while the VT125 terminal does not take current screen-addressing coordinates into account when specifying the size of graphics text cells.

An application using explicit (non-default) screen addressing and text-cell size produces text in proportion to the current screen addressing. The same application on a VT125 terminal does not take screen addressing into consideration and produces text sized by the specified constant physical pixel count.

D.2.6 Screen Time Delay Differences

D.2.6.1 Fixed Tick Length—On the Professional, the screen time delay option permits you to specify a multiple of ticks with the length of each tick fixed at 1/64 of a second, the frequency of the Professional's system clock. The screen time delay option specifies the number of ticks to wait before processing the next ReGIS command. For example, the command S(T16) specifies 16 ticks, each 1/64 of a second long, resulting in a quarter-second delay before the next command executes.

The VT125 terminal's time delay is dependent on terminal power supply (50 or 60 Hertz). An integral number of frame times (1/50th or 1/60th seconds) is used with the ReGIS S(T) command. Existing VT125 applications with delays specified have a slightly longer time delay for the S(T) command.

D.2.6.2 Maximum Wait Time—The Professional provides a longer maximum screen time delay period than that of the VT125 terminal. On the Professional, the S(T) command's tick count is stored as a word, allowing modulo 65536 interpretation and providing a maximum delay of about 17 minutes per command.

On a VT125 terminal, the tick count is stored in a byte and is interpreted modulo 256 for a maximum delay of about four seconds. Therefore, applications with S(T) command values larger than 255 invoke a longer delay when running on the Professional.

D.2.7 Report Features

D.2.7.1 Report Error—The ReGIS instruction set on the Professional contains a report-error instruction. The VT125 terminal does not have this instruction. Applications software written for the VT125 is not affected by this added feature.

D.2.7.2 Unimplemented Reports Response—On the Professional, a command requesting a report that is not implemented (i.e., not defined for the specified keyletter) returns a carriage-return character to indicate an empty report. Report requests are made through the ReGIS report (R) command. On the VT125 terminal, there is no response to a request for an unimplemented report.

D.2.8 Non-Supported VT125 Capabilities

D.2.8.1 No VT105 Emulation—The Professional does not provide a VT105 emulation mode. A VT125 terminal can emulate the VT105.

D.2.8.2 No DECwriter Data Transfer—The Professional does not support transfer of DECwriter descriptor data to a file on the host computer. A VT125 terminal transfers such data with the media copy (MC) command, but on the Professional, this command is ignored.

D.2.8.3 No Additional Monitors—The Professional supports only one monitor at a time, either a monochrome or a color monitor. A VT125 terminal has provision to support separately-controlled images on a main and an external monitor. An application that displays two separate images on the VT125's monitors will have these images superimposed on the Professional monitor.

D.2.8.4 Scaling "Zoom" Not Implemented—The Professional terminal hardware does not provide image scaling, also known as the "non-destructive zoom." This effect, done with the ReGIS screen-scale (S(S)) command and a new scaling constant, permitted enlargement (2X zoom) and shrinking of display image.

D.2.8.5 No Custom Writing Controls (Input Map)—Custom writing with the ReGIS custom-writing (W(W)) command is not implemented on the Professional. Applications that use custom writing are executed in the current writing mode, rather than in the user-specified custom writing mode.

D.2.8.6 No Concurrent Display of Graphics and ReGIS Text Mode—The Professional does not support the simultaneous display of graphics images with the ReGIS text that generates those images. While such a capability exists on a VT125 terminal, the Professional, which does not permit independent scrolling of text and graphics images, ignores requests for simultaneous graphics/ReGIS code display. On a VT125 terminal, concurrent display is achieved through the DCS 2 p and DCS 3 p control strings.

INDEX

Professional 300 Series
Developer's Tool Kit
Terminal Subsystem Manual
Order No. AA-N623B-TK

READER'S COMMENTS

NOTE: This form is for document comments only. DIGITAL will use comments submitted on this form at the company's discretion. If you require a written reply and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Did you find this manual understandable, usable, and well-organized?
Please make suggestions for improvement.

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

Please cut along this line.

Did you find errors in this manual? If so, specify the error and the page number.

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

Please indicate the type of reader that you most nearly represent.

- ☐ Assembly language programmer
- ☐ Higher-level language programmer
- ☐ Occasional programmer (experienced)
- ☐ User with little programming experience
- ☐ Student programmer
- ☐ Other (please specify) _____

Name _____ Date _____

Organization _____

Street _____

City _____ State _____ Zip Code _____
or
Country

-- Do Not Tear - Fold Here and Tape ----------

digital

No Postage
Necessary
if Mailed in the
United States

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

Professional 300 Series Publications
DIGITAL EQUIPMENT CORPORATION
146 MAIN STREET
MAYNARD, MASSACHUSETTS 01754

--- Do Not Tear - Fold Here ----------

Cut Along Dotted Line