EK-VT382-RM-001

VT382

Kanji Display Terminal

Programmer Reference Manual

CONTENTS

ABOUT THIS MANUAL

This reference manual is for people with a general knowledge of computer programming. The manual describes the control functions you can use when writing applications for the VT382 terminal. The manual also shows all the character set built into the terminal and describes how the VT382 processes characters.

For general operating information, refer to "Installing and Using the VT382 Video Terminal (shorten as "The VT382 User Guide" : EK-VT382-UG)".

ORGANIZATION

This manual is divided into three parts. It has 13 chapters, 3 appendices.

Part 1, "Introduction to Your VT382 Terminal," covers the information you need to know before you begin programming the terminal.

- Chapter 1, "VT382 Features," provides an overview of the terminal's major features and operating modes.

- Chapter 2, "Character Encoding," describes the character-encoding concepts used in the VT382 terminal. The chapter also describes the terminal's character sets and control function format.

Part 2, "Control Functions Sent to the Host," covers the codes sent from the keyboard.

- Chapter 3, "Keyboard codes," describes the characters and control functions that the terminal sends to the host.

Part 3, "Control Functions Received from the Host," covers all the control functions you can use to program the terminal.

- Chapter 4, "Emulating VT Series Terminals," describes the control functions used to emulate Digital's other VT series terminals.

- Chapter 5, "Using Character Sets," describes the control functions used to select the terminal's built-in character sets and your own soft character sets.

- Chapter 6, "Screen Display Control Functions," describes the control functions used to control how data appears on the screen.

- Chapter 7, "Setting Visual Character and Line Attributes," describes the control functions used to select such attributes as bold and underlining.

- Chapter 8, "Editing," describes the control functions used to edit text on the screen.

- Chapter 9, "Controlling the Cursor," describes the control functions used to move the cursor.

- Chapter 10, "Keyboard and Printing Commands," describes the control functions used to program the terminal's keyboard and printer port features.

- Chapter 11, "VT382 Reports," describes the control functions used to request reports from the VT382. The chapter also describes the format of VT382 reports and the control functions used to restore the VT382 to a previous state.

- Chapter 12, "Sixel Graphics" describes the control functions used to program about Sixel graphics.

- Chapter 13, "Resetting and Testing the Terminal," describes the control functions used to reset and test the terminal.

- Appendix A, "VT52 Mode Control Codes," describes the control codes used in VT52 mode only.

- Appendix B, "Compatibility with Other Digital Terminals," compares the VT382 to other VT series terminals.

- Appendix C, "Communication," describes the preventing a buffer overflow and the fill characters used for the VT382 communication.

CONVENTIONS

- Notes and programming tips appear throughout this manual.

  - Notes provide general operating information.
  - Programming tips provide helpful suggestions to consider when writing applications.

- Set-Up features and keyboard keys appear with " ".

  (Examples)

  Press the "Return" key. Use the "Clear Communications feature" in the "Set-Up Directory" screen.

- Below each character is a column/row number that indicates the character's position in a standard code table.

  (Example)

```
ESC   #    6          <--   Control function
1/11  2/3  3/6        <--   Column/row numbers
```

PART 1
INTRODUCTION TO YOUR VT382 TERMINAL

CHAPTER 1

VT382 FEATURES

1.1 THE VT382 TERMINAL

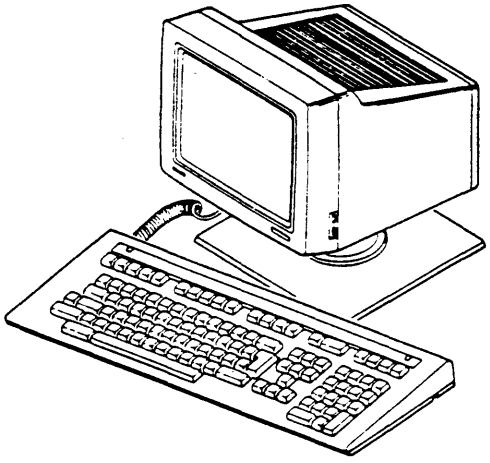

Figure 1-1 VT382 Video Terminal

This chapter provides an overview of the VT382 video terminal. The chapter briefly describes the major features and operating modes of the terminal. Each section tells you where to look in the manual for more information on that feature.

The VT382 has two major components, a monitor/terminal unit and keyboard. See "The VT382 User Guide" description of these components.

This manual covers the programming information you need to use the

features of the VT382. The terminal uses control functions that conform to the American National Standards Institute (ANSI) and the International Organization for Standardization (ISO).

Figure 1-1 shows a VT382 terminal. The next section describes some important features of the VT382.

1.2 GENERAL FEATURES

The VT382 is compatible with Digital's VT200 series terminal and provides enhancements. This section describes the general operating and communication features of the VT382. You can set many of these features from the keyboard, using Set-Up.

1.2.1 Set-Up

Set-Up is a series of display screens. Each screen lists a group of features, such as communications or printing.

You can use Set-Up screens to examine and change the current settings for features. For example, you can select the keyclick feature, transmit or receive speeds, and character set mode.

The VT382 Set-Up feature is similar to the VT282 Set-Up feature. "The VT382 User Guide" describes the Set-Up screens in detail.

1.2.2 Display Features

The VT382 screen has the following features.

Monitor	14 inch flat screen (paper white)
Display area	25 lines x 80 or 132 columns(ASCII) 25 lines x 40 or 66 columns(Kanji) For 80 columns: 960 horizontal pixels x 750 vertical pixels For 132 columns: 924 horizontal pixels x 750 vertical pixels
Status line	On the 25th display line

Character size	For 80 columns: 10 x 22 pixel matrix in 12 x 30 character cell (as ASCII) 22 x 22 pixel matrix in 24 x 30 character cell (as Kanji) For 132 columns: 6 x 22 pixel matrix in 7 x 30 character cell (as ASCII) 12 x 22 pixel matrix in 14 x 30 character cell (as Kanji)
Scrolling	Vertical scrolling on any column boundary

1.2.3 Text Features

The VT382 provides a variety of text and editing features.

Character sets (See next section.)	6 sets of 94 characters each 1 set of 96 characters 2 sets of 2-Byte characters Down-line-loadable character set (94 or 96 characters)
Top-row function keys	5 local function keys 15 user-definable keys
Editing functions	All VT282 editing functions
Visual character attributes	Normal, bold, underline, blinking, and reverse video
Line attributes	Single-width/single-height lines Double-width/single-height lines Double-width/double-height lines
Control functions	7-bit and 8-bit control characters ANSI control functions DEC private control functions Ability to display control functions

1.2.4 Character Sets

The VT382 has the following built-in character sets.

ASCII	(94 characters)
DEC Special Graphic (VT100 line drawing)	(94 characters)
DEC Supplemental Graphic	(94 characters)
ISO Latin alphabet Nr 1 supplemental graphic	(96 characters)
DEC Technical	(94 characters)
JIS-Roman	(94 characters)
JIS-Katakana	(94 characters)
DEC Kanji (1978)	(94 x 94 characters)
DEC Kanji (1983)	(94 x 94 characters)

You can also design a soft character set and load it from the host system into the terminal.

Chapter 2 describes the VT382 character sets. Chapter 5 describes how to select and use different character sets.

1.2.5 Communication Features

The VT382 provides the following features for communicating with the host system.

Character format	7-bit or 8-bit
Communication speed	Asynchronous communication speeds up to 19.2K bits per second
Connectors	One 6-pin DEC423 host port, allowing longer distances between the terminal and host
	One RS232 host port, with a 25-pin D-subminiature connector for a host or external modem
	One 6-pin DEC423 printer port

1.3 OPERATING STATES

The VT382 has two major operating states. You select the operating state in Set-Up.

On-line
Local

1.3.1 On-Line

The on-line state lets the terminal communicate with a host system. The terminal sends data entered at the keyboard to the host. The terminal displays data received from the host on the screen.

1.3.2 Local

The local state effectively puts the host system on hold. Data entered at the Keyboard is sent to the screen when you set, but not to the host. The terminal stores data received from the host, but the terminal does not display the data. The terminal displays the stored data when you return to the on-line state.

1.4 OPERATING MODES

The VT382 has four major operating modes. You can select each mode from the keyboard by using Set-Up, or from the host system by using control codes. The VT382 uses standard ANSI function in all operating modes, except VT52 mode.

VT300 mode, 7-bit controls (default)
VT300 mode, 8-bit controls
VT100 mode
VT52 mode

Each mode lets the VT382 have compatibility with other Digital terminals as follows:

	VT52 Mode	VT100 Mode	VT300 Mode VT200 Mode
Katakana Terminal	VT52J	VT102J	VT220J
Kanji Terminal	N/A	VT80	VT382 VT282

VT300 mode, 7-bit controls is the default operating mode. This mode provides the full range of VT382 capabilities, using 8-bit characters and 7-bit control characters. VT300 mode is fully compatible with applications designed for VT200 series terminal. All character sets are available. Digital recommends this mode for most applications.

VT300 mode, 8-bit controls provides the full range of VT382 capabilities, using 8-bit characters and 8-bit control characters. All character sets are available, and terminal recognizes both 7-bit and 8-bit control characters. This mode is fully compatible with applications designed for VT200 series terminals. The terminal operates most efficiently in this mode, but many systems and applications do not yet support 8-bit operation.

VT100 mode provides full compatibility with Digital's VT80 or VT102J terminal. The keyboard is restricted to VT100 series keys. You use this mode with applications written for the VT80 or VT102J.

VT52 mode provides full compatibility with Digital's VT52J terminal. This mode only uses Digital's private control functions, not standard ANSI functions. In VT52 mode, the terminal does not recognize ANSI control functions. You use this mode with applications written for the VT52J.

Chapter 2 describes the format for 7-bit and 8-bit character codes. Chapter 4 describes how the VT382 can emulate other VT series terminals.

CHAPTER 2

CHARACTER ENCODING

The VT382 uses a communication line to exchange information with a host system, The terminal and the host do not send data in the form you see on your screen. They must encode the information first. They also must be able to decode the information received from each other.

This chapter describes the character-encoding system that the VT382 uses. You must have a basic understanding of the character-encoding system described in this chapter before you use the control functions in the rest of this manual.

The chapter also describes the VT382 character sets and the format for sending control functions to the terminal. You can select character sets for special uses, such as line-drawing characters. You use control functions to make the terminal perform special functions, such as editing or printing.

2.1 CODING STANDARDS

All terminals and computers encode information as binary digits, or bits. Older systems use 7 bits to encode each character. Newer systems such as the VT382 use 8 bits, which provide more codes. The newer systems can also use the 7-bit codes.

The VT382 uses an 8-bit character-encoding system and a 7-bit code extension technique. The "7-Bit Code Extension Technique" section in this chapter explains what 7-bit code extensions are.

CHARACTER ENCODING

The American National Standards Institute (ANSI) and International Organization for Standardization (ISO) specify standards for character encoding in the information processing industry. The VT382 terminal conforms to the following ANSI and ISO standards. The VT382 also uses two-bytes character encoding scheme for Kanji characters.

Standard	Description
dpANS X3.134.1 - 1985	8-bit ASCII structure and rules
dpANS X3.134.2 - 1985	Code for information interchange of 7-bit and 8-bit ASCII supplemental multilingual graphic character set
ANSI X3.4 - 1977	American Standard Code for Information Interchange (ASCII)
ANSI X3.41 - 1974	Code Extension Techniques for Use with the 7-Bit Coded Character Set of American National Code Information Interchange
ANSI X3.64 - 1979	Additional Controls for Use with American National Standard for Information Interchange
ISO 646 - 1977	7-Bit Coded Character Set for Information Processing Interchange
ISO 2022 - 1986	7-Bit and 8-Bit Coded Character Sets - Code for Extension Techniques
ISO 6429 - 1983	Additional Control Functions for Character Imaging Devices
ISO 8859-1.2	8-Bit single byte code graphic character sets-Part 1: Latin alphabet Nr 1
ISBN 2-12-953907-0	ISO international register of character sets used with escape sequences

Standard	Description
JIS X0201 - 1976	Code for Information Interchange (JIS-Roman Character Set and JIS-Katakana Character Set)
JIS X0208 - 1978	Code of the Japanese Graphic Character Set for Information Interchange
JIS X0208 - 1983	Code of the Japanese Graphic Character Set for Information Interchange
JIS X0202 - 1975	Code Extension Techniques for Use with the Code for Information Interchange

2.2 CHARACTERS AND CHARACTER SETS

In Digital's computing environment, a character is a symbol represented by an 8-bit binary code. These symbols include letters, digits, and punctuation marks, as well as other symbols used to organize, control, or represent data.

Here are a few examples of characters and their corresponding 8-bit codes.

Character	Code
A	01000001
}	01111101
CSI	10011011

There are two types of computing environments, 7-bit and 8-bit. In a 7-bit environment, only the last 7 bits of the character code define the character. In an 8-bit environment, all 8 bits define the character.

The A character above is defined in a 7-bit or 8-bit environment, because the eighth bit of the code is 0. The CSI character is defined only on an 8-bit environment, because its eighth bit is 1. To send the CSI character in a 7-bit environment, a two character sequence is required.

A coded character set is a group of characters that conform to certain rules and standards. Each character in a character set is represented by a different code or bit combination. Many of the character sets used in the VT382 have been standardized by organizations such as ANSI, ISO and JIS.

2.3 CODE TABLE

A code table is a convenient way to show all the characters in a character set with their codes. Most standard character sets put similar characters into groups, so they have similar codes. A code table lets you see groups of characters and their relative codes clearly.

There are two basic types of characters, graphic characters and control characters.

Graphic characters are characters you can display. Graphic characters include letters, numbers, punctuation marks, and any other characters you can display.

Control characters are characters you do not usually display. They make the terminal or host system perform specific functions in data communications and text processing.

> **NOTE: You can display control characters on the screen, to help you debug your applications. To display control characters, use the "Interpret/Display Controls" feature in the "Display Set-Up" screen. See the "Display Controls Mode" section at the end of this chapter.**

This section describes the format for 7-bit and 8-bit code tables.

2.3.1 7-Bit ASCII Code Table

Figure 2-1 is the 7-bit ASCII code table. The table has 128 character codes, arranged in 8 columns and 16 rows.

Every character in a row uses the same binary code for its four least significant bits (Figure 2-2). This value appears at the left of each row. For example, every character in row 0 uses the binary code 0000 for its four least significant bits.

Every character in a column uses the same binary code or its three most significant bits. This value appears at the top of each column. For example, every character in column 0 uses the binary code 000 for its three most significant bits.

The ASCII table also shows the octal, decimal, and hexadecimal code for each character. Different programmers may prefer using octal, decimal, or hexadecimal codes for different purposes.

This manual refers to characters by their position in the table. For example, the character H is at 4/8 (column 4, row 8). You can use the column/row number to find a character and its codes in the table. For example

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

means

The ESC character is at column 1, row 11.
The # character is at column 2, row 3.
The 6 character is at column 3, row 6.

The ASCII graphic characters are in positions 2/1 through 7/14 of the ASCII table. ASCII graphic characters include all American and English alphanumeric characters, plus punctuation marks and various text symbols. Examples are c, n,",!,+, and $. (The English pound sign is not an ASCII graphic character.)

The ASCII control characters are in positions 0/0 through 1/15 (columns 0 and 1) of the ASCII table. The SP character (2/0) may act as a graphic space character or a control character, depending on the context. DEL (7/15) is always a control character.

ANSI and ISO standards define control character codes and their functions. These standards also define the mnemonic used to represent each control character in a code table. Here are some examples of ASCII control characters with their mnemonics.

ASCII Control Character	Mnemonic (Appears in Code Table)
Carriage return	CR
Form feed	FF
Cancel	CAN

ROW	0	1	2	3	4	5	6	7
0	NUL	DLE	SP	0	@	P	`	p
1	SOH	DC1 (XON)	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3 (XOFF)	#	3	C	S	c	s
4	EOT	DC4	$	4	D	T	d	t
5	ENQ	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB	'	7	G	W	g	w
8	BS	CAN	(	8	H	X	h	x
9	HT	EM	)	9	I	Y	i	y
10	LF	SUB	*	:	J	Z	j	z
11	VT	ESC	+	;	K	[	k	{
12	FF	FS	,	<	L	\	l	|
13	CR	GS	-	=	M	]	m	}
14	SO	RS	.	>	N	^	n	~
15	SI	US	/	?	O	_	o	DEL

— C0 CODES —

GL CODES
(ASCII GRAPHIC)

Figure 2-1 7-Bit ASCII Code Table

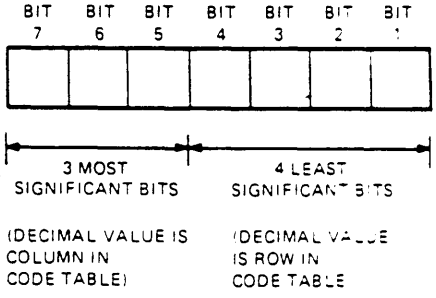

Figure 2-2 7-Bit Code

2.3.2 8-Bit Code Table

Figure 2-3 shows the format for an 8-bit code table. It has the same number of rows as the 7-bit table, but twice as many columns and character code positions.

Each character in a row of the 8-bit table uses the same binary code for its four least significant bits (Figure 2-4). Each character in a column uses the same binary code for its four most significant bits.

The codes on the left half of the 8-bit table (columns 0 through 7) work like the codes in the 7-bit table. You can use these codes in a 7-bit or 8-bit environment. The eighth bit of these codes is 0.

The codes on the right half of the table (columns 8 through 15) have an eighth bit of 1. You can only use these codes in an 8-bit environment.

The 8-bit code table has two sets of control characters, C0 (control zero) and C1 (control one). The VT382 uses the ANSI definitions for the functions of C0 and C1 controls. The C0 controls are in columns 0 and 1. The C0 controls are the same as the ASCII control characters in the 7-bit table. You can use C0 controls in a 7-bit environment.

The C1 controls are in columns 8 and 9. They perform different functions than the C0 controls. You can only use C1 controls directly in an 8-bit environment. You can select C1 codes indirectly in a 7-bit environment. The "7-Bit Code Extension Technique" section in this chapter explains how to select C1 controls indirectly. Some C1 code positions are blank, because their functions are not yet standardized.

> **NOTE: The VT382 does not recognize all C0 and C1 codes. Tables 2-1 and 2-2 list the codes the terminal recognizes. The terminal ignores all other control codes.**

The table also has two sets of graphic characters, GL (graphic left) and GR (graphic right). There are 94 GL codes in positions 2/1 through 7/14. You can use GL codes in 7-bit or 8-bit environments.

There are 96 GR codes in positions 10/0 through 15/15. Some 8-bit character sets only use 94 of these GR codes. You can use GR codes only in an 8-bit environment.

Together, the GL and GR sets make up the terminal's in-use table. The in-use table contains the graphic characters the terminal uses to interpret 8-bit codes. Before the terminal can display characters from a

character set, the set must be mapped into the in-use table. Chapter 5 describes the in-use table in detail.

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	SOS						
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						

C0 CODES — GL CODES — C1 CODES — GR CODES

7-BIT CODE TABLE

Figure 2-3 8-Bit Code Table

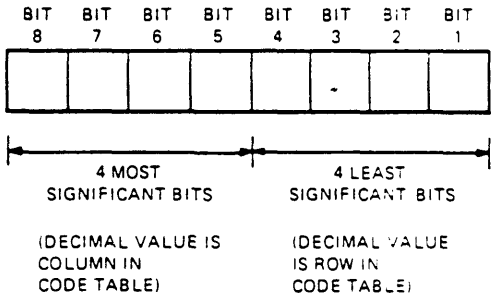

Figure 2-4 8-Bit Code

2.4 VT382 CHARACTER SETS

The VT382 provides the following built-in graphic character sets.

ASCII
DEC Supplemental Graphic
ISO Latin-1 Supplemental Graphic
DEC Special Graphic
DEC Technical
JIS-Roman
JIS-Katakana
DEC Kanji (1978)
DEC Kanji (1983)

You can also design and load a soft character set into the terminal from the host system.

Down-line-loadable (soft) set

The VT382 also has two control character sets, C0 and C1. These control character sets can be used with any of the graphic character sets. The control character sets never change, no matter what graphic character set you use. The terminal always interprets C0 and C1 control codes as defined by ANSI.

The terminal uses GL and GR tables to interpret the codes it receives. Selecting a new character set changes the characters associated with the GL or GR codes. When you turn on or reset the terminal, you automatically select the following character sets.

When use the terminal as "Kanji Terminal",

ASCII or JIS-Roman in GL
DEC Kanji in GR

When use the terminal as "Katakana Terminal",

ASCII or JIS-Roman in GL
JIS-Katakana in GR

Together, the ASCII set and one of the supplemental sets make up a multinational character set.

- The ASCII set and DEC Supplemental Graphic sets are known as the DEC Multinational character set.

- The ASCII set and the ISO Latin-1 supplemental set are known as the ISO Latin Alphabet Nr 1 set.

You select the supplemental set with the "User-Preferred Supplemental Set" feature in the "Terminal Set-Up" screen.

2.4.1 DEC Supplemental Graphic Character Set (DEC Multinational Character Set)

The DEC Multinational Character Set is shown in Figure 2-5.

The 7-bit compatible left half of DEC Multinational Set is the ASCII Graphics Set: the C0 codes are the ASCII control characters and the GL codes are 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 the DEC Supplemental Graphics Set. The DEC Supplemental Graphics Set has alphabetic characters with accents and diacritical marks that appear in the major Western European alphabets. It also has other symbols not included in the ASCII Graphics Set.

> **NOTE: All control function descriptions in this manual assume that the terminal is using the DEC Multinational set.**
> **Available in VT300 mode only.**

You can select the DEC Supplemental Graphic set as the default by using control functions or Set-Up.

ROW	COLUMN 0	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6	COLUMN 7
	BITS b7 b6 b5 / 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	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 GRAPHIC) →

Figure 2-5 DEC Multinational Character Set
(Left Half - C0 and GL Codes : ASCII)

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

← C1 CODES → ← GR CODES (DEC SUPPLEMENTAL GRAPHIC) →

Figure 2-5 DEC Multinational Character Set (Right Half - C1 and GR Codes : DEC Supplemental Set)

2.4.2 ISO Latin-1 Supplemental Graphic Character Set (ISO Latin Alphabet Nr 1 Supplemental Set)

This 8-bit character set has 96 graphic characters. The graphic characters are similar to those in the DEC Supplemental Graphic set. The ISO Latin-1 supplemental set includes letters with accents and diacritical marks, used in many European languages. It also has other special symbols and letters, not included in the DEC Supplemental Graphic set.

Figure 2-6 shows the ISO Latin-1 supplemental set. The C1 controls are in columns 8 and 9. The graphic characters are in columns 10 through 15.

You can select the ISO Latin-1 supplemental set as the default by using control functions (Chapter 5) or Set-Up. The combination of the ASCII character set in GL and the ISO Latin-1 supplemental set in GR is called the ISO Latin Alphabet Nr 1 character set.

NOTE: You can only use the ISO Latin-1 set in VT300 mode.

8	9	10	11	12	13	14	15	COLUMN / BITS / ROW
	DCS	NBSP		À	Ð	à	ð	0
	PU1	¡	±	Á	Ñ	á	ñ	1
	PU2	¢	2	Â	Ò	â	ò	2
	STS	£	3	Ã	Ó	ã	ó	3
IND	CCH	¤	´	Ä	Ô	ä	ô	4
NEL	MW	¥	µ	Å	Õ	å	õ	5
SSA	SPA	¦	¶	Æ	Ö	æ	ö	6
ESA	EPA	§	·	Ç	×	ç	÷	7
HTS	SOS	¨	¸	È	Ø	è	ø	8
HTJ		©	¹	É	Ù	é	ù	9
VTS		ª	º	Ê	Ú	ê	ú	10
PLD	CSI	«	»	Ë	Û	ë	û	11
PLU	ST	¬	¼	Ì	Ü	ì	ü	12
RI	OSC		½	Í	Ý	í	ý	13
SS2	PM	®	¾	Î	Þ	î	þ	14
SS3	APC	¯	¿	Ï	ß	ï	ÿ	15

C1 CODES ← → GR CODES (ISO LATIN-1 SUPPLEMENTAL GRAPHIC)

Figure 2-6 ISO Latin Alphabet Nr 1 Supplemental Set (Right Half - C1 and GR Codes : ISO Latin-1 Supplemental Set)
This Figure omits Left Half (ASCII).

2.4.3 DEC Special Graphic Character Set

This 7-bit character set has 94 graphic characters. Most of the graphic characters are also in the ASCII character set. The other graphic characters include special symbols and line-drawing characters.

Figure 2-7 shows the DEC Special Graphic set. The C0 controls are in columns 0 and 1. The graphic characters are in columns 2 through 7.

Another name for this character set is the VT100 line-drawing character set. The line-drawing characters let you create a limited range of pictures when you use the VT382 as a text terminal.

You can use the DEC Special Graphic set to either GL or GR. Chapter 5 describes how to select character sets.

ROW	COLUMN 0	1	2	3	4	5	6	7
	BITS B7 B6 B5 / 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	— 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	▮ SCAN 157 111 6F	DEL 177 127 7F

← C0 CODES ← → GL CODES (DEC SPECIAL GRAPHIC) →

Figure 2-7 DEC Special Graphic Character Set

2.4.4 DEC Technical Character Set

This 7-bit character set has 94 graphic characters. The DEC Technical character set has characters and symbols often used in technical applications, such as schematic and logic diagrams.

Figure 2-8 shows the DEC Technical character set. The C0 controls are in columns 0 and 1. The graphic characters and symbols are in columns 2 through 7. You can use the characters in positions 2/1 through 3/7 to form large composite characters.

You can use the DEC Technical character set into either GL or GR.

NOTE: You can only use the DEC Technical set in VT300 mode.

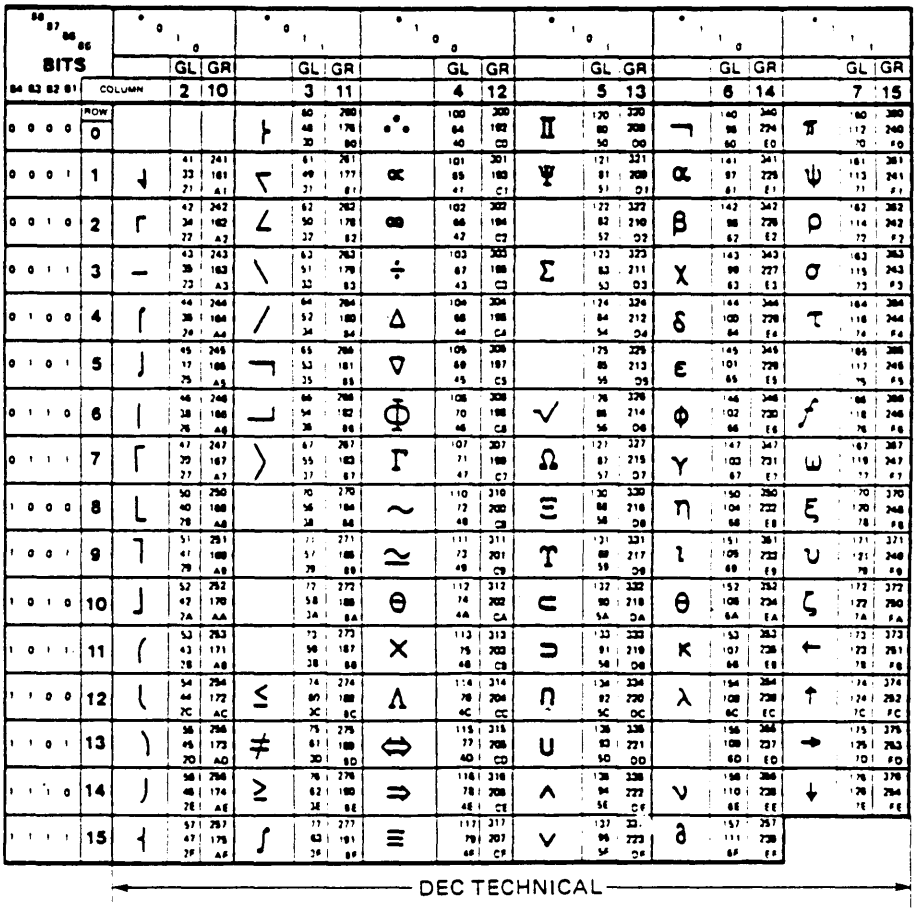

Figure 2-8 DEC Technical Character Set

2.4.5 JIS-Roman Character Set

This 7-bit character set has 94 graphic characters.

Figure 2-9 shows the JIS-Roman Character Set. This set differs from the ASCII Character Set only in that the Yen sign(¥)replaces the backslash(\) in column 5, row 12 (5/12) and the over line (‾) replaces the tilde (~) in column 7, row 14 (7/14).

ROW	COLUMN 0	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6	COLUMN 7
0	NUL	DLE	SP	0	@	P	`	p
1	SOH	DC1 (XON)	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3 (XOFF)	#	3	C	S	c	s
4	EOT	DC4	$	4	D	T	d	t
5	ENQ	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB	'	7	G	W	g	w
8	BS	CAN	(	8	H	X	h	x
9	HT	EM	)	9	I	Y	i	y
10	LF	SUB	*	:	J	Z	j	z
11	VT	ESC	+	;	K	[	k	{
12	FF	FS	,	<	L	¥	l	|
13	CR	GS	-	=	M	]	m	}
14	SO	RS	.	>	N	^	n	‾
15	SI	US	/	?	O	_	o	DEL

← C0 CODES → ← GL CODES (JIS-ROMAN) →

Figure 2-9 JIS-Roman Character Set

2.4.6 JIS-Katakana Character Set

This 8-bit character set has 94 graphic characters.

Figure 2-10 shows the JIS-Katakana Character Set. When the VT382 receives 14/0 through 15/14, the VT382 displays a reverse question mark on the screen.

8	9	10	11	12	13	14	15	COLUMN / BITS b8 b7 b6 b5 b4 b3 b2 b1	ROW
200 128 80	DCS 220 144 90	240 160 A0	ー 260 176 B0	タ 300 192 C0	ミ 320 208 D0	? 340 224 E0	? 360 240 F0	0 0 0 0	0
201 129 81	PU1 221 145 91	。 241 161 A1	ア 261 177 B1	チ 301 193 C1	ム 321 209 D1	? 341 225 E1	? 361 241 F1	0 0 0 1	1
202 130 82	PU2 222 146 92	「 242 162 A2	イ 262 178 B2	ツ 302 194 C2	メ 322 210 D2	? 342 226 E2	? 362 242 F2	0 0 1 0	2
203 131 83	STS 223 147 93	」 243 163 A3	ウ 263 179 B3	テ 303 195 C3	モ 323 211 D3	? 343 227 E3	? 363 243 F3	0 0 1 1	3
IND 204 132 84	CCH 224 148 94	、 244 164 A4	エ 264 180 B4	ト 304 196 C4	ヤ 324 212 D4	? 344 228 E4	? 364 244 F4	0 1 0 0	4
NEL 205 133 85	MW 225 149 95	・ 245 165 A5	オ 265 181 B5	ナ 305 197 C5	ユ 325 213 D5	? 345 229 E5	? 365 245 F5	0 1 0 1	5
SSA 206 134 86	SPA 226 150 96	ヲ 246 166 A6	カ 266 182 B6	ニ 306 198 C6	ヨ 326 214 D6	? 346 230 E6	? 366 246 F6	0 1 1 0	6
ESA 207 135 87	EPA 227 151 97	ァ 247 167 A7	キ 267 183 B7	ヌ 307 199 C7	ラ 327 215 D7	? 347 231 E7	? 367 247 F7	0 1 1 1	7
HTS 210 136 88	SOS 230 152 98	ィ 250 168 A8	ク 270 184 B8	ネ 310 200 C8	リ 330 216 D8	? 350 232 E8	? 370 248 F8	1 0 0 0	8
HTJ 211 137 89	231 153 99	ゥ 251 169 A9	ケ 271 185 B9	ノ 311 201 C9	ル 331 217 D9	? 351 233 E9	? 371 249 F9	1 0 0 1	9
VTS 212 138 8A	232 154 9A	ェ 252 170 AA	コ 272 186 BA	ハ 312 202 CA	レ 332 218 DA	? 352 234 EA	? 372 250 FA	1 0 1 0	10
PLD 213 139 8B	CSI 233 155 9B	ォ 253 171 AB	サ 273 187 BB	ヒ 313 203 CB	ロ 333 219 DB	? 353 235 EB	? 373 251 FB	1 0 1 1	11
PLU 214 140 8C	ST 234 156 9C	ャ 254 172 AC	シ 274 188 BC	フ 314 204 CC	ワ 334 220 DC	? 354 236 EC	? 374 252 FC	1 1 0 0	12
RI 215 141 8D	OSC 235 157 9D	ュ 255 173 AD	ス 275 189 BD	ヘ 315 205 CD	ン 335 221 DD	? 355 237 ED	? 375 253 FD	1 1 0 1	13
SS2 216 142 8E	PM 236 158 9E	ョ 256 174 AE	セ 276 190 BE	ホ 316 206 CE	゛ 336 222 DE	? 356 238 EE	? 376 254 FE	1 1 1 0	14
SS3 217 143 8F	APC 237 159 9F	ッ 257 175 AF	ソ 277 191 BF	マ 317 207 CF	゜ 337 223 DF	? 357 239 EF	377 255 FF	1 1 1 1	15

← C1 CODES → ← GR CODES (JIS-KATAKANA) →

Figure 2-10 JIS-Katakana Character Set

2.4.7 DEC Kanji Character Set

NOTE: This set is not available in VT52 mode.

The terminal's graphic repertoire includes the DEC Kanji Character Set. Regarding this character set, please refer to the DEC Kanji Code Book . JIS X0208 was first issued on 1978 and a revision was issued on 1983. The terminal supports both versions. You can select the "Kanji version" in the "Terminal Set-Up".

DEC Kanji-1978 : JIS X0208-1978
DEC Kanji-1983 : JIS X0208-1983 (Code Book : AA-A056B-TE-J0)

If the selection is different from the version supported by the host, unexpected characters may appear on the screen.

When the terminal is turned on, the Kanji version is determined by the Set-Up selection. When the terminal receives one of the Kanji designating escape sequences, the Kanji version is determined by the Set-Up selection. (Refer to Chapter 5.)

The terminal transmits the secondary DA response including the information of the Kanji version in response to request from the host. (Refer to Chapter 11.)

2.4.8 Down-Line-Loadable (Soft) Character Set

The VT382 lets you down-line-load a character set from the host system. The character set can have up to 96 graphic characters. You can design your own character set, then load the set into the terminal. You can use the set in GL or GR. Chapter 5 describes how to load and use a soft character set.

NOTE: You can only use this character set in VT300 mode.

2.5 CONTROL CHARACTERS

The purpose of a control character is to control an action such as line spacing or data flow. The terminal does not display control characters unless you select display controls mode (described at the end of this chapter). There are two groups of control characters.

C0 7-bit control characters, in columns 0 and 1 of the 8-bit code table

C1 8-bit control characters, in columns 8 and 9 of the 8-bit code table

Table 2-1 lists the C0 control characters the VT382 recognizes. Table 2-2 lists the C1 control characters the VT382 recognizes. You can also code C1 control characters as 7-bit escape sequences. Table 2-3 lists the equivalent 7-bit sequences for 8-bit control characters. All three tables give column/row locations to help you find the characters in the character sets.

Table 2-1 C0(7Bit) Control Characters Recognized

Name	Mnemonic Column/Row	Function
Null	NUL 0/0	Ignored.
Enquiry	ENQ 0/5	Sends the answerback message ("Terminal Set-Up" screen).
Bell	BEL 0/7	Sounds the bell tone if the bell is enabled in the "General Set-Up" screen.
Backspace	BS 0/8	Moves the cursor one character position to the left. If the cursor is at the left margin, no action occurs.
Horizontal Tab	HT 0/9	Moves the cursor to the next tab stop. If there are no more tab stops, the cursor moves to the right margin. HT does not cause text to auto wrap.
Line feed	LF 0/10	Causes a line feed or a new line operation. depending on the setting of "line feed/new line" mode.
Vertical tab	VT 0/11	Treated as LF.

Table 2-1 C0(7Bit) Control Characters Recognized (cont)

Name	Mnemonic Column/Row	Function
Form feed	FF 0/12	Treated as LF.
Carriage return	CR 0/13	Moves the cursor to the left margin on the current line.
Shift out (Locking shift 1)	SO(LS1) 0/14	Maps the G1 character set into GL. You designate G1 by using a select character set (SCS) sequence (Chapter 5).
Shift in (Locking shift 0)	SI(LS0) 0/15	Maps the G0 character set into GL. You designate G0 by using a select character set (SCS) sequence (Chapter 5).
Device control 1 (XON)	DC1 1/1	Also known as XON. If XON/XOFF flow control is enabled in the "Communications Set-Up" screen, receiving DC1 causes the VT382 to continue sending characters.
Device control 3 (XOFF)	DC3 1/3	Also known as XOFF. If XON/XOFF flow control is enabled in Set-Up, DC3 causes the VT382 cannot resume sending characters until it receives a DC1 control character.
Cancel	CAN 1/8	Immediately cancels an escape sequence or control sequence in progress. The VT382 does not display any error characters.
Substitute	SUB 1/10	Immediately cancels an escape sequence or control sequence in progress. The VT382 displays a reverse question mark (؟) for an error character.

Table 2-1 C0(7Bit) Control Characters Recognized (cont)

Name	Mnemonic Column/Row	Function
Escape	ESC 1/11	Introduces an escape sequence. ESC also cancels any escape sequence or control sequence in progress.
Delete	DEL 7/15	Ignored when received. DEL is not used as a fill character. Digital does not recommend using DEL as a fill character. Use NUL instead.

Table 2-2 C1 (8-Bit) Control Characters Recognized

Name	Mnemonic Column/Row	Function
Index	IND 8/4	Moves the cursor down one line in the same column. If the cursor is at the bottom margin, a scroll up occurs.
Next line	NEL 8/5	Moves the cursor to the first position on the next line. If the cursor is at the bottom margin, a scroll up occurs.
Horizontal tab set	HTS 8/8	Sets a horizontal tab stop at the column where the cursor is.
Reverse index	RI 8/13	Moves the cursor up one line in the same column. If the cursor is at the top margin, a scroll down occurs.
Single shift 2	SS2 8/14	Temporarily maps the G2 character set into GL, for the next graphic character. You designate the G2 set by using a select character set (SCS) sequence (Chapter 5).

Table 2-2 C1 (8-Bit) Control Characters Recognized (cont)

Name	Mnemonic Column/Row	Function
Single shift 3	SS3 8/15	Temporarily maps the G3 character set into GL, for the next graphic character. You designate the G3 set by using a select character set (SCS) sequence (Chapter 5).
Device control string	DCS 9/0	Introduces a device control string.
Start of string	SOS 9/8	Introduces a string.*
Control sequence introducer	CSI 9/11	Introduces a control sequence.
String terminator	ST 9/12	Ends a control string. You use ST in combination with DCS, APC, OSC, PM, or SOS strings.
Operating system command	OSC 9/13	Introduces an operating system command.*
Privacy message	PM 9/14	Introduces a privacy message string.*
Application program command	APC 9/15	Introduces an application program command.*

* The VT382 ignores all following characters, until it receives an ST control character. ESC,CAN and SUB no longer cancel device control strings.

Table 2-3 8-Bit Control Characters and Their 7-Bit Equivalents

Name	8-Bit Character	7-Bit Sequence	
Index	IND 8/4	ESC 1/11	D 4/4
Next line	NEL 8/5	ESC 1/11	E 4/5
Horizontal tab set	HTS 8/8	ESC 1/11	H 4/8
Reverse index	RI 8/13	ESC 1/11	M 4/13
Single shift 2	SS2 8/14	ESC 1/11	N 4/14
Single shift 3	SS3 8/15	ESC 1/11	O 4/15
Device control string	DCS 9/0	ESC 1/11	P 5/0
Start of string	SOS 9/8	ESC 1/11	X 5/8
Control sequence introducer	CSI 9/11	ESC 1/11	[5/11
String terminator	ST 9/12	ESC 1/11	\ 5/12
Operating system command	OSC 9/13	ESC 1/11	] 5/13
Privacy message	PM 9/14	ESC 1/11	^ 5/14
Application program command	APC 9/15	ESC 1/11	_ 5/15

2.6 CONTROL FUNCTIONS

You use control functions to make the terminal perform special actions in your applications. These functions range from the simple - editing data - to the complex - reporting the terminal's operating state. The rest of this manual covers the many uses for control functions. Here are some examples.

Move the cursor.
Delete a line of text.
Select bold or underlined text.
Change character sets.
Make the terminal emulate a VT52 or VT100 terminal.

There are single-character and multiple-character control functions.

The single-character functions are the C0 and C1 control characters. You can use C0 characters in a 7-bit or 8-bit environment. C1 characters provide a few more functions than C0 characters, but you can only use C1 characters directly in an 8-bit environment.

Multiple-characters functions provide many more functions than the C0 and C1 characters. Multiple-character functions can use control characters and graphic characters. There are three basic types of multiple-character functions.

escape sequences
control sequences
device control strings

Many sequence are based on ANSI and ISO standards, and used throughout the industry. Others are private sequences created by manufacturers like Digital for specific families of products. ANSI sequences and private sequences follow ANSI and ISO standards for control functions.

In this manual, private control functions created by Digital have the prefix DEC in their mnemonic name. For example, column mode has the mnemonic DECCOLM. All other control functions are standardized.

The following sections describe the format for escape sequences, control sequence, and device control strings.

PROGRAMMING TIP:
When you use control functions, remember that the binary codes define a function - not the graphic characters. This manual uses graphic characters from the DEC Multinational character set to show control functions. If you use another character set, the graphic characters for control functions may change, but the code is always the same.

2.6.1 Sequence Format

This manual shows escape and control sequences in their 8-bit format. You can also use equivalent 7-bit sequences (Table 2-3).

The 8-bit format uses the C0 and C1 control characters and ASCII characters from the DEC Multinational character set. The sequences also show each character's column/row position in the character set table. below the character. The column/row code eliminates confusion over similar looking characters such as O (3/0) and 0 (4/15).

> **NOTE: Spaces appear between characters in a sequence for clarity. These spaces are not part of the sequence. If a space is part of the sequence, the SP (2/0) character appears.**

2.6.2 Escape Sequences

An escape sequence uses two or more bytes to define a specific control function. Escape sequences do not include variable parameters, but may include intermediate characters. Here is the format for an escape sequence.

ESC	I	F
1/11	2/0 to 2/15	3/0 to 7/14
Escape	Intermediate characters (zero or more characters)	Final character (one character)

"ESC" introduces escape sequences. After receiving the ESC control character, the terminal interprets the next received characters as part of the sequence.

"I" represents zero or more intermediate characters that can follow the ESC character. Intermediate characters come from the 2/0 through 2/15 range of the code table.

"F" is the final character. This character indicates the end of the sequence. The final character comes from the 3/0 through 7/14 range of the code table. The intermediate and final characters together define a single control function.

For example, the following escape sequence changes the current line of text to double-width, single-height characters.

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

2.6.3 Control Sequences

A control sequence uses two or more bytes to define a specific control function. Control sequences usually include variable parameters. Here is the format for a control sequence.

CSI	P...P	I...I	F
9/11	3/0 to 3/15	2/0 to 2/15	4/0 to 7/14
Control sequence introducer	Parameter (zero or more characters)	Intermediate (zero or more characters)	Final (one character)

"CSI" is the control sequence introducer. You can also use the equivalent 7-bit sequence, ESC [(1/11, 5/11), as a substitute for CSI. After receiving CSI, the terminal interprets the next received characters as part of the sequence.

"P...P" are parameter characters received after CSI. These characters are in the 3/0 to 3/15 range in the code table. Parameter characters modify the action or interpretaion of the sequence. You can use up to 16 parameters per sequence. You must use the ; (3/11) character to separate parameters.

All parameters are unsigned, positive decimal integers, with the most significant digit sent first. Any parameter greater than 9999 (decimal) is set to 9999 (decimal). If you do not specify a value, a default value is assumed. A 0 value or omitted parameter indicates a default value for the sequence, For most sequences, the default value is 1.

> **NOTE: All parameters must be positive decimal integers. Do not use a decimal point in a parameter - the terminal will ignore the command.**

If the first character in a parameter string is the ? (3/15) character, it indicates that DEC private parameters follow. The syntax for conforms to the rules in ANSI and ISO standards.

The VT382 processes two types to parameters, numeric and selective.

< Numeric Parameters >

A numeric parameter indicates a number value such as a margin location. In this manual, numeric parameters appear as actual values or as Pn, Pn1, Pn2, and so on.

The following is an example of a control sequence with numeric parameters.

CSI	5	;	2	0	r
9/11	3/5	3/11	3/2	3/0	7/2
Control sequence introducer	First numeric parameter	Delimiter	Second numeric parameter		Final character

This sequence sets the top and bottom margins to define the scrolling region. The top margin is at line 5, the bottom is at line 20. The ; (3/11) separates the two parameters.

< Selective Parameters >

A selective parameter selects an action associated with the specific parameter. In this manual, selective parameters usually appear as Ps, Ps1, Ps2 and so on.

The following is an example of a control sequence using selective parameters.

CSI	1	;	4	m
9/11	3/1	3/11	3/4	6/13
Control sequence introducer	First selective parameter	Delimiter	Second selective parameter	Final character

This control sequence turns on the bold and underline attribute at the cursor position. The parameters are 1 (indicating the bold attribute) and 4 (indicating the underline attribute). The ; (3/11) delimiter separates the two parameters.

"I...I" are zero or more intermediate characters received after CSI. These characters are in the 2/0 to 2/15 range.

"F" is the final character from the 4/0 to 7/14 range. The final character indicates the end of the sequence. The intermediate and final characters together define a control function. If there are no intermediate characters, the final character defines the function.

2.6.4 Device Control Strings

Device control strings (DCS), like control sequences, use two or more bytes to define specific control functions. However, a DCS also includes a data string. Here is the format for a device control string.

DCS	P...P	I...I	F	Data string	ST
9/0	3/0 to 3/15	2/0 to 2/15	4/0 to 7/15	**********	9/12
Device control string introducer	Zero or more parameters	Zero or more intermediates	Final	String	String terminator

DCS is the device control string introducer. DCS is the C1 control character at position 9/0. You can also use the equivalent 7-bit sequence, ESC (1/11) P (5/0). After receiving DCS, the terminal processes the next received characters as part of the string function.

"P...P" are parameter characters received after DCS. The use of parameter characters in a device control string is a Digital extension to the ANSI syntax. According to ANSI standards, any elements included after DCS are part of the data string.

Parameter characters are in the 3/0 to 3/15 range. They modify the action or interpretation of the device control string. You can use up to 16 parameters per string. Each parameter is separated with a ; (3/11) character. These characters follow the same rules as in a control sequence. See the previous "Control Sequences" section in this chapter.

"I...I" are zero or more intermediate character received after CSI. These characters are in the 2/0 to 2/15 range.

"F" is the final character in the 4/0 to 7/14 range. The final character indicates the end of the string. The intermediate and final characters define the string. If there are no intermediate, the final character defines the string.

"Data string" follows the final character and usually includes several definition strings. Each definition string can be several characters in length. Individual strings are separated by the ; (3/11) delimiter.

"ST" is the string terminator. ST (9/12) indicates the end of a string. You can also use the equivalent 7-bit sequence, ESC \ (1/11, 5/12).

The following is an example of a device control string.

DCS	0	!	u	%	5	ST
9/0	3/0	2/1	7/5	2/15	3/5	9/12
Device control string introducer	Parameter	Intermediate	Final	Data string		String terminator

This device control string assigns the DEC Supplemental Graphic set as the user-preferred supplemental set.

2.6.5 Using Control Characters In Sequences

You can use control characters - ESC, CAN, and SUB - to interrupt or recover from errors in escape sequences and control sequences.

- You can send ESC (1/11) to cancel a sequence in progress and begin a new sequence.
- You can send CAN (1/8) to indicate the present data is in error or to cancel a sequence in progress. The VT382 interprets the characters following CAN as usual.
- You can send SUB (1/10) to cancel a sequence in progress. The VT382 interprets the characters following SUB as usual.

The VT382 does not lose data when errors occur in escape sequences, control sequences, or device control strings. The terminal ignores unrecognized sequences and strings, unless they end a current escape sequence.

2.6.6 7-Bit Code Extension Technique

This technique provides a way to use 8-bit C1 controls in a 7-bit environment. You can represent all C1 control characters as 7-bit escape sequences. You can use the C1 characters indirectly, by representing them as 2-character escape sequences. ANSI calls this technique a 7-bit code extension. The 7-bit code extension provides a way of using C1 characters in applications written for a 7-bit environment. Here are some examples.

8-bit C1 Character	7-bit Code Extension Escape Sequence	
CSI 9/11	ESC 1/11	[5/11
SS3 8/15	ESC 1/11	O 4/15
IND 8/4	ESC 1/11	D 4/4
DCS 9/0	ESC 1/11	P 5/0

In general, you can use the 7-bit code extension technique in two ways.

- You can represent any C1 control character as a 2-character escape sequence. The second character of the sequence has a code that is 40 (hexadecimal) and 64 (decimal) less than that of the C1 character.

- You can make any escape sequence whose second character is in the range of 4/0 through 5/15 one byte shorter, by removing the ESC character and adding 40 (hexadecimal) to the code of the second character. This generates an 8-bit control character. For example, you can change ESC [to CSI with this method.

2.7 WORKING WITH 7-BIT AND 8-BIT ENVIRONMENTS

There are three requirements for using one of the terminal's 8-bit character sets.

- Your program and communication environment must be 8-bit compatible.
- The terminal's character set mode must be set to 8-bit characters (Chapter 4).
- The terminal must operate in VT300 mode. When the terminal operates VT100 mode or VT52 mode, you are limited to working in a 7-bit environment (Chapter 4).

The following sections describe conventions that only apply in VT300 mode.

2.7.1 Conventions For Codes Received By The Terminal

The terminal expects to receive character codes in a form compatible with 8-bit coding. Your application can use the C0 and C1 control characters, as well as the 7-bit C1 code extensions, if necessary. The terminal always interprets these codes correctly.

When your program sends GL or GR codes, the terminal interprets the character codes according to the graphic character sets in use. When you turn on or reset the terminal, you automatically select the ASCII or JIS-Roman character set in GL and DEC Kanji (as Kanji terminal) or JIS-Katakana (as Katakana terminal) character set in GR. You select the character sets in the "Terminal Set-Up" screen. This mapping assumes the current terminal mode is VT300 or VT100 mode.

2.7.2 Conventions For Codes Sent By The Terminal

The terminal can send data to an application in two ways.

- Directly from the keyboard
- In response to commands from the host (application or operating system)

Most function keys on the keyboard send multiple-character control functions. Many of these functions start with CSI (9/11) or SS3 (8/15), which are C1 characters. If your application cannot handle 8-bit characters, you can make the terminal automatically convert all C1 characters to their equivalent 7-bit code extensions before sending them to the application. To convert C1 characters, you use the DECSCL commands described in Chapter 4.

By default, the terminal is set to automatically convert all C1 characters sent to the application to 7-bit code extensions. However, to ensure the correct mode of operation, always use the appropriate DECSCL commands.

> **NOTE: In VT300 mode, the terminal can send GR graphic characters to an application, even if the application cannot handle 8-bit codes. However, in a 7-bit environment, the terminal sends C1 controls as 7-bit escape sequences and does not send 8-bit graphic characters.**
>
> **New programs should accept both 7-bit and 8-bit forms of the C1 control characters.**

2.8 DISPLAY CONTROLS MODE

The VT382 lets you display control characters as graphic characters, when you want to debug your applications. In this mode, the terminal does not perform all control functions.

To select this mode, you must set the "Interpret/Display Controls" feature in the "Display Set-Up" screen to "Display Controls". You can also select this mode with an escape sequence.

The effect of the "Display Controls" setting depends on the operating mode you use.

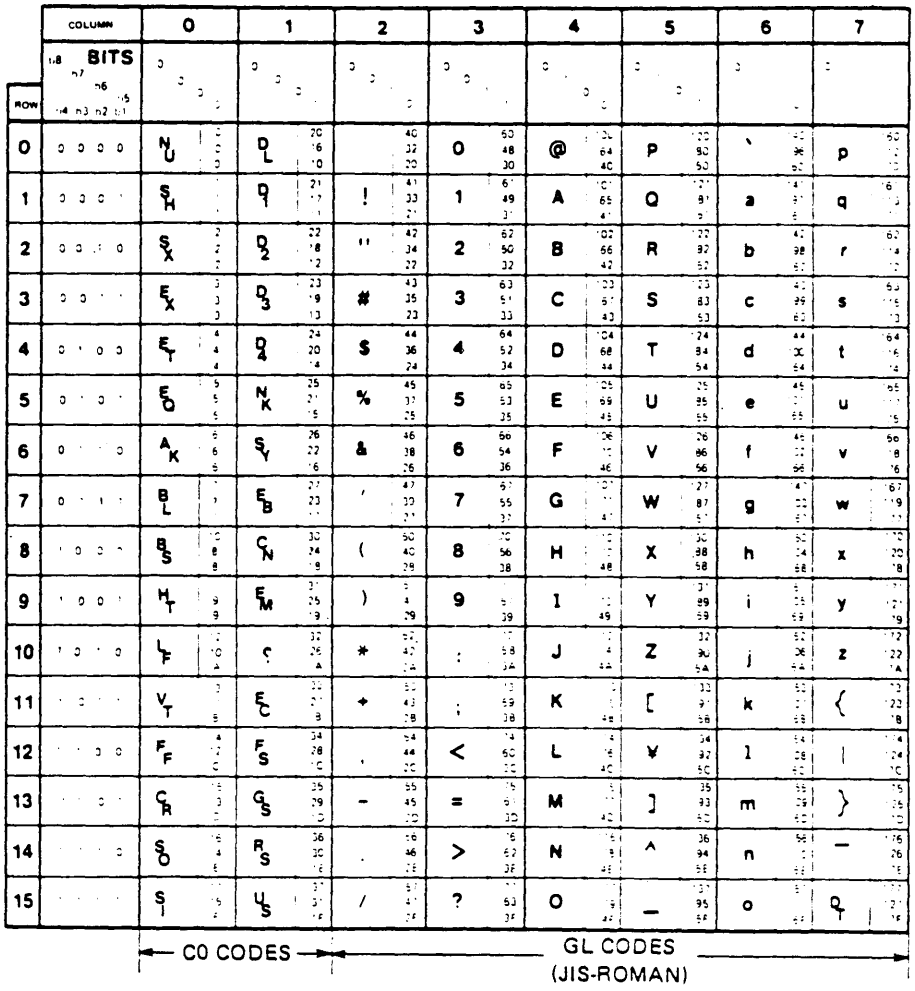

Figure 2-11 Display Controls Font(Left Half)

8	9	10	11	12	13	14	15	COLUMN / BITS	ROW
8 0	9 0	A 0	—	タ	ミ	E 0	F 0	0 0 0 0	0
8 1	9 1	。	ア	チ	ム	E 1	F 1	0 0 0 1	1
8 2	9 2	「	イ	ツ	メ	E 2	F 2	0 0 1 0	2
8 3	9 3	」	ウ	テ	モ	E 3	F 3	0 0 1 1	3
8 4	9 4	、	エ	ト	ヤ	E 4	F 4	0 1 0 0	4
8 5	9 5	・	オ	ナ	ユ	E 5	F 5	0 1 0 1	5
8 6	9 6	ヲ	カ	ニ	ヨ	E 6	F 6	0 1 1 0	6
8 7	9 7	ァ	キ	ヌ	ラ	E 7	F 7	0 1 1 1	7
8 8	9 8	ィ	ク	ネ	リ	E 8	F 8	1 0 0 0	8
8 9	9 9	ゥ	ケ	ノ	ル	E 9	F 9	1 0 0 1	9
8 A	9 A	ェ	コ	ハ	レ	E A	F A	1 0 1 0	10
8 B	9 B	ォ	サ	ヒ	ロ	E B	F B	1 0 1 1	11
8 C	9 C	ャ	シ	フ	ワ	E C	F C	1 1 0 0	12
8 D	9 D	ュ	ス	ヘ	ン	E D	F D	1 1 0 1	13
8 E	9 E	ョ	セ	ホ	゛	E E	E E	1 1 1 0	14
8 F	9 F	ッ	ソ	マ	゜	E F	□	1 1 1 1	15

← C1 CODES → ← GR CODES (JIS-KATAKANA) →

Figure 2-11 Display Controls Font(Right Half)

When you select "Display Controls", the terminal temporarily loads a special graphic character set into C0, GL, C1, and GR. Figure 2-11 shows this special set, called the display controls font. The terminal uses this font to display control characters on the screen.

< Exceptions >

Some control functions still work in display controls mode.

- LF, FF, and VT cause a carriage return and line feed (CR LF) that move the cursor to a new line. The terminal displays the LF, FF, or VT character before performing the new line function.

- XOFF (DC3) and XON (DC1) maintain flow control, if enabled in Set-Up. The terminal displays the DC1 or DC3 character after performing the control function.

PART 2
CONTROL FUNCTIONS SENT TO THE HOST

CHAPTER 3

KEYBOARD CODES

This chapter describes the codes that the terminal can send to an application program. The chapter assumes that you are familiar with the character-encoding concepts described in chapter 2.

In VT300 or VT100 mode, the keyboard keys send codes that are compatible with ANSI standards. In VT52 mode, some keys send codes that differ from those sent in the ANSI-compatible modes. This chapter lists VT52 codes that differ from the ANSI-compatible codes.

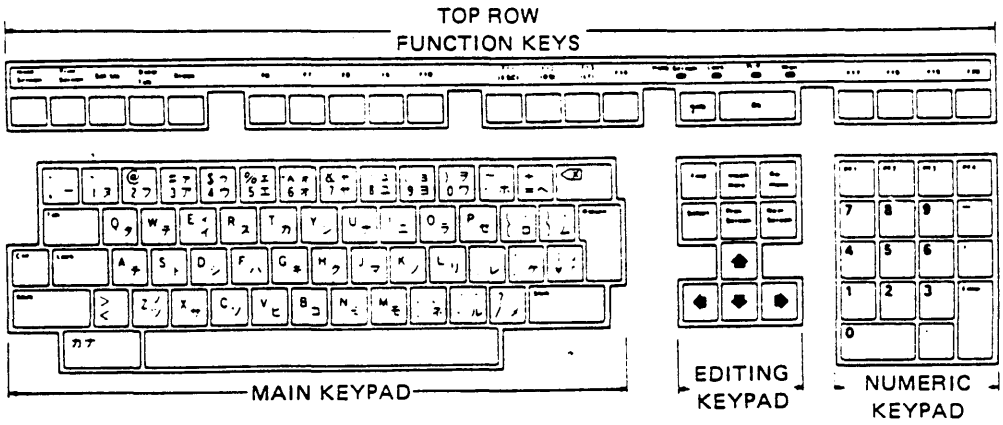

Figure 3-1 Four Key Groups

3.1 KEYBOARD CODES

The keyboard (Figure 3-1) has four groups of keys: a main keypad, an editing keypad, a numeric keypad, and the top-row function keys.

3.1.1 Main Keypad

The main keypad has standard keys and function keys. You use the standard keys to send letters, numbers, and other symbols. You can use the function keys to send special function codes.

3.1.1.1 Standard Keys

The standard keys generate alphanumeric and Katakana characters either singly or in combination with other keys.

On the Katakana keyboard, the standard keys show both JIS-Roman and Katakana characters. There are no DEC Supplemental characters among the standard keys. JIS-Roman characters are located on the left side of the each standard keycap. The "KANA" key is used to change the keyboard state from JIS-Roman to Katakana or vice-versa. You can also select either shifted (upper) or unshifted (lower) character codes for these keys by using the "Shift" key.

In an 8-bit environment (8-bit host line), JIS-Roman and Katakana character is represented by a unique code according to the character's position in the code table. On the other hand, in a 7-bit environment (7-bit host line), any keystroke which would normally produce an 8-bit character produce an 8th bit stripped 7-bit code with SI(0/15)/SO(0/14).

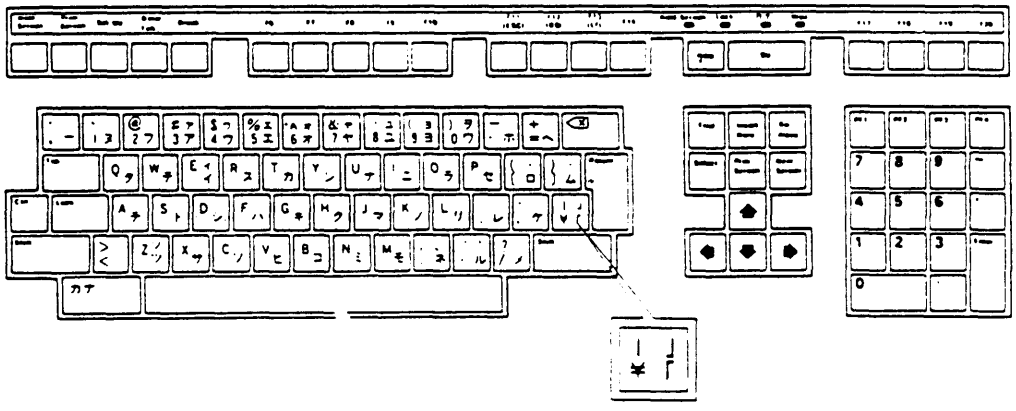

Figure 3-2 Katakana Keyboard

3.1.1.2 Function Keys

This section describes the function keys on the main keypad. Remember, the column/row numbers that appear after a character tell you the position of the character in the code table (Chapter 2). For example, the DEL character is at column 7 / row 15.

Table 3-1 Function Keys in Main Keypad

Key	Function
⌫	The ⌫ key sends a delete character (DEL, 7/15) or a backspace character (BS, 0/8), depending on the state of setting of the backarrow key mode (DECBKM). You can also select the code sent by key in the "Keyboard Set-Up" screen. **NOTE: In this manual ⌫ key is called as "Backarrow key", on the other hand, "Delete key" in "The VT382 User Guide".**
Tab	The Tab key sends a horizontal tab character (HT, 0/9).
Return	The Return key sends either a carriage return (CR, 0/13), or a carriage return (CR, 0/13) and line feed (LF,0/10), depending on the state of line feed/new line mode (LNM). You can also select the code(s) sent by Return key in the "Display Set-Up" screen.
Ctrl	The Ctrl key alone does not send a code. You use Ctrl with another key to send a control code.
Lock	The Lock key alone does not send a code. You use Lock to set or clear the "Caps Lock" or "Shift Lock" state. You select "Caps Lock" or "shift Lock" in the "Keyboard Set-Up" screen.
Shift (2keys)	To Shift key alone does not send a code. You use Shift with another standard key to send an upper case character or the top character shown on the key.
Space bar	The space bar sends a space character (SP, 2/0).

Table 3-1 Function Keys in Main Keypad(cont)

Key	Function
KANA	The KANA key does not transmit a code. This key is used to change the Keyboard state from JIS-Roman to Katakana or vice-versa. When the KANA key is depressed while the keyboard is in JIS-Roman state, the KANA LED is turned on, and next keystrokes on the standard keys generate Katakana codes. When the KANA key is depressed while the keyboard is in the Katakana state, the KANA LED is turned off, and next keystrokes on the standard keys generate JIS-Roman codes. This key is also used in combination with the CTRL key to create arbitrary hex characters.

3.1.2 Editing Keypad

The editing keypad includes the editing keys and arrow keys. Table 3-2 lists the codes sent by the editing keys, and Table 3-3 list the codes sent by the arrow keys. Normally, you use the arrow keys to move the cursor on the screen. See "Cursor Keys Mode (DECCKM)" in Chapter 10.

Chapter 4 describes how to select VT300 or VT100 mode. Appendix A describes how to select VT52 mode.

Table 3-2 Codes Sent by Editing Keys

Key	Code Sent VT300 Mode			VT100,VT52 Modes
Find	CSI 9/11	1 3/1	~ 7/14	The editing keys do not send codes in these two modes.
Insert Here	CSI 9/11	2 3/2	~ 7/14	
Remove	CSI 9/11	3 3/3	~ 7/14	
Select	CSI 9/11	4 3/4	~ 7/14	

Table 3-2 Codes Sent by Editing Keys (cont)

Key	Code Sent VT300 Mode			VT100,VT52 Modes
Prev Screen	CSI 9/11	5 3/5	~ 7/14	The editing keys do not send codes in these two modes.
Next Screen	CSI 9/11	6 3/6	~ 7/14	

Table 3-3 Codes Sent by Arrow Keys

	Cursor Key Mode Setting (DECCKM)					
	ANSI Mode				VT52 Mode*	
Key	Cursor		Application		Cursor or Application	
↑	CSI 9/11	A 4/1	SS3 8/15	A 4/1	ESC 1/11	A 4/1
↓	CSI 9/11	B 4/2	SS3 8/15	B 4/2	ESC 1/11	B 4/2
→	CSI 9/11	C 4/3	SS3 8/15	C 4/3	ESC 1/11	C 4/3
←	CSI 9/11	D 4/4	SS3 8/15	D 4/4	ESC 1/11	D 4/4

* ANSI mode applies to VT300 and VT100 modes. VT52 mode is not compatible with ANSI mode.

3.1.3 Numeric Keypad

The characters sent by the numeric keypad depend on the setting of VT52 mode and numeric keypad mode. The application usually selects the application keypad codes. However, you can select the application keypad codes in the "Keyboard Set-Up" screen.

For more information, see "Numeric Keypad Mode (DECNKM)" in Chapter 10. Chapter 4 describes how to select VT300 or VT100 mode. Appendix A describes how to select VT52 mode.

Table 3-4 lists the character codes sent by the numeric keypad in ANSI modes (VT100 and VT300) and in VT52 mode.

Table 3-4 Codes Sent by Numeric Keypad Keys

	Numeric Keypad Mode Setting (DECNKM)						
	ANSI Mode *1			VT52 Mode			
Key	Numeric	Application		Numeric	Application		
0	0 3/0	SS3 8/15	p 7/0	0 3/0	ESC 1/11	? 3/15	p 7/0
1	1 3/1	SS3 8/15	q 7/1	1 3/1	ESC 1/11	? 3/15	q 7/1
2	2 3/2	SS3 8/15	r 7/2	2 3/2	ESC 1/11	? 3/15	r 7/2
3	3 3/3	SS3 8/15	s 7/3	3 3/3	ESC 1/11	? 3/15	s 7/3
4	4 3/4	SS3 8/15	t 7/4	4 3/4	ESC 1/11	? 3/15	t 7/4
5	5 3/5	SS3 8/15	u 7/5	5 3/5	ESC 1/11	? 3/15	u 7/5
6	6 3/6	SS3 8/15	v 7/6	6 3/6	ESC 1/11	? 3/15	v 7/6
7	7 3/7	SS3 8/15	w 7/7	7 3/7	ESC 1/11	? 3/15	w 7/7
8	8 3/8	SS3 8/15	x 7/8	8 3/8	ESC 1/11	? 3/15	x 7/8
9	9 3/9	SS3 8/15	y 7/9	9 3/9	ESC 1/11	? 3/15	y 7/9

Table 3–4 Codes Sent by Numeric Keypad Keys (cont)

	Numeric Keypad Mode Setting (DECNKM)							
	ANSI Mode *1				VT52 Mode			
Key	Numeric		Application		Numeric		Application	
-	(minus)		SS3	m	-		ESC ?	m
	2/13		8/15	6/13	2/13		1/11 3/15	6/13 *2
,	(comma)		SS3	l	,		ESC ?	l
	2/12		8/15	6/12	2/12		1/11 3/15	6/12 *2
.	(period)		SS3	n	.		ESC ?	n
	2/14		8/15	6/14	2/14		1/11 3/15	6/14
Enter	CR		SS3	M	CR		ESC ?	M
	0/13		8/15	4/13	0/13		1/11 3/15	4/13
	or				or			
	CR	LF *3			CR	LF *3		
	0/13	0/10			0/13	0/10		
PF1	SS3	P	SS3	P	ESC	P	ESC P	
	8/15	5/0	8/15	5/0	1/11	5/0	1/11 5/0	
PF2	SS3	Q	SS3	Q	ESC	Q	ESC Q	
	8/15	5/1	8/15	5/1	1/11	5/1	1/11 5/1	
PF3	SS3	R	SS3	R	ESC	R	ESC R	
	8/15	5/2	8/15	5/2	1/11	5/2	1/11 5/2	
PF4	SS3	S	SS3	S	ESC	S	ESC S	
	8/15	5/3	8/15	5/3	1/11	5/3	1/11 5/3 *2	

*1 ANSI mode applies to VT300 and VT100 modes. VT52 mode is not compatible with ANSI standards.
*2 You cannot use these sequences on a VT52 terminal.
*3 Keypad numeric mode. "Enter" sends the same codes as "Return". You can use line feed/new line mode (LNM) to change the code sent by "Return". When LNM is reset, pressing "Return" sends one control character (CR). When LNM is set, pressing Return sends two control characters (CR,LF).

3.1.4 Top-Row Function Keys

There are 20 top-row function keys, F1 through F20. The first five keys - labeled "Hold Screen", "Print Screen", "Set-Up", "Data/Talk", and "Break" - are local function keys that do not send codes. You use these keys to perform predefined function local to the terminal. Keys F6 through F20 send the codes listed in Table 3-5. For more information, see Chapter 3 of "The VT382 User Guide".

Chapter 4 describes how to select VT300 or VT100 mode. Appendix A describes how to select VT52 mode.

Table 3-5 Codes Sent by the Top-Row Function Keys

Name on Legend Strip	Key Number	Code Sent: VT300 Mode				Code Sent: VT100, VT52 Modes
Hold Screen	(F1)*	-				-
Print	(F2)*	-				-
Set-Up	(F3)*	-				-
Data/Talk	(F4)*	-				-
Break	(F5)*	-				-
F6	F6	CSI 9/11	1 3/1	7 3/7	~ 7/14	-
F7	F7	CSI 9/11	1 3/1	8 3/8	~ 7/14	-
F8	F8	CSI 9/11	1 3/1	9 3/9	~ 7/14	-
F9	F9	CSI 9/11	2 3/2	0 3/0	~ 7/14	-
F10	F10	CSI 9/11	2 3/2	1 3/1	~ 7/14	-

* F1 through F5 are local function keys that do not send codes.

Table 3-5 Codes Sent by the Top-Row Function Keys (cont)

Name on Legend Strip	Key Number	Code Sent: VT300 Mode				Code Sent: VT100, VT52 Modes
F11 (ESC)	F11	CSI 9/11	2 3/2	3 3/3	~ 7/14	ESC 1/11
F12 (BS)	F12	CSI 9/11	2 3/2	4 3/4	~ 7/14	BS 0/8
F13 (LF)	F13	CSI 9/11	2 3/2	5 3/5	~ 7/14	LF 0/10
F14	F14	CSI 9/11	2 3/2	6 3/6	~ 7/14	-
Help	F15	CSI 9/11	2 3/2	8 3/8	~ 7/14	-
Do	F16	CSI 9/11	2 3/2	9 3/9	~ 7/14	-
F17	F17	CSI 9/11	3 3/3	1 3/1	~ 7/14	-
F18	F18	CSI 9/11	3 3/3	2 3/2	~ 7/14	-
F19	F19	CSI 9/11	3 3/3	3 3/3	~ 7/14	-
F20	F20	CSI 9/11	3 3/3	4 3/4	~ 7/14	-

3.1.5 7-Bit Control Codes

Table 3-6 lists the key or keys you use to send each 7-bit control characters. The 7-bit control characters are the C0 characters. You can send some 8-bit C1 control characters from the keyboard, but they do not have dedicated keystrokes.

Table 3–6 Keys Used to Send 7-Bit Control Codes

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

*1 7-bit control codes sent in VT100 and VT52 modes only.
*2 7-bit control codes sent only when XON/XOFF support is off.

3.2 SPECIAL CASES

This section describes special functions and modes that affect the keyboard.

3.2.1 Turning Autorepeat On And Off

The autorepeat feature make most keys send their character repeatedly when you hold the key down. You can turn the autorepeat feature on and off by using the "General Set-Up" screen or the autorepeat mode (DECARM) control function (Chapter 10).

The following keys do not repeat: "Hold Screen", "Print Screen", "Set-Up", "Data/Talk", "Break", "KANA", "Shift", "Return", "Lock", and "Ctrl". Shifted keys and keys pressed with "Ctrl" can repeat.

Keys that can auto repeat usually start repeating after a delay of 0.5 seconds. The autorepeat speed depends on the baud rate of the host system and the type of key. At speeds of 2400 baud or above, all keys repeat 30 times per seconds. At lower speeds, the keyboard is divided into three groups.

Group A	Main keyboard
Group B	Cursor keys and keypad keys
Group C	Top-row function keys and editing keys

The keys in each group repeat at the fixed rate set by the baud rate of the host, regardless of how many codes the key actually sends.

Host Baud Rate	Autorepeat Rate (Characters/Second) Group A	Group B	Group C
2400 or above	30	30	30
1200	30	30	24
600	30	20	12
300	30	10	6
150	13	6	6
110	10	6	6
75	6	6	6

In general, the "Transmit Rate Limit" feature in the "Communications Set-Up" screen does not affect repeat rates. The terminal can send codes at the speed of 150 characters per second at most baud rates. In local mode, keys repeat at 30 keystrokes per second.

3.2.2 Unlocking The Keyboard

Two conditions can cause the keyboard to lock.

- An application sends a control function to set the keyboard action mode (KAM), as described in Chapter 10.
- The keyboard input buffer is full.

When the keyboard is locked, all keys except "Hold Screen", "Print Screen", "Set-Up", "Data/Talk", and "Break" are disabled, Also, the keyboard's "Wait indicator" turns on.

Any of the following events can unlock the keyboard.

- The output buffer becomes less than full (assuming KAM is not set).
- The terminal receives KAM when the output buffer is not full (Chapter 10).
- You select the "Clear Comm", "Reset Terminal", or "Recall" features from the "Set-Up Directory" screen. (Entering Set-Up unlocks the keyboard, If you do not select one of these functions in Set-Up, the keyboard locks again when you leave Set-Up.)
- The terminal performs the power-up self-test (DECTST) or a hard reset (RIS). See Chapter 13.

PART 3
CONTROL FUNCTIONS RECEIVED FROM THE HOST

CHAPTER 4

EMULATING VT SERIES TERMINALS

The VT382 terminal can operate like VT200, VT100, and VT52 series terminals. This feature lets you use the VT382 with applications designed for these terminals. You can select from two possible levels of operation.

Level 1 for VT100 operation
Level 3 for VT200 and VT300 operation (default)

When you operate the terminal at level 1, you cannot use some VT382 control functions. Table 4-1 lists the functions you cannot use. Level 3 includes all the characteristics of level 2. This means that applications designed for level 2 terminals (such as the VT200 series) run in level 3.

The following paragraphs describe other limits that apply to each operating level. The chapter also describes how to select an operating level and how to send 7-bit or 8-bit C1 controls to the host.

Appendix A describes how to use VT52 mode.

4.1 LEVEL 1 (VT100 MODE)

The following limits apply to operating level 1.

- The following keys do not operate.
  - special-function keys, except F11 (ESC), F12 (BS), and F13 (LF)
  - six editing keys

- user-defined keys

. Soft character sets are not available.

. The terminal sends all C1 control characters as 7-bit escape sequences (ESC Fe).

Table 4-1 Control Functions Ignored in Level 1 (VT100 Mode)

Name	Mnemonic
DECAUPSS	Assign User-Preference Supplemental Character Set
DECCIR	Cursor information report
DECDLD	Down-line-loadable set
DECRPM	Report mode
DECRPSS	Report selection or setting
DECRQM	Request mode
DECRQPSR	Request presentation state
DECRQSS	Request selection or setting
DECRQTSR	Request terminal state
DECRQUPSS	Request User-Preference Supplemental Character Set
DECRSPS	Restore presentation state
DECRSTS	Restore terminal state
DECSASD	Select active status display
DECSCA	Select character attribute
DECSED	Selective erase in display
DECSEL	Selective erase in line
DECSSDT	Select status display type
DECSTR	Soft terminal reset
DECTABSR	Tabulation stop report
DECTSR	Terminal state report
DECUDK	User-defined keys
DSR	UDK and keyboard language
ECH	Erase character
ICH	Insert character
S7C1T	Send 7-bit C1 controls
S8C1T	Send 8-bit C1 controls

4.2 LEVEL 3 (VT300 MODE)

In VT300 mode, you can use all VT382 features. All keyboard functions are available. You can use all control functions and device control strings described in this manual. VT300 mode is fully compatible with VT200 series terminals.

4.3 SELECTING AN OPERATING LEVEL (DECSCL)

You select the terminal's level by using the following select conformance level (DECSCL) control sequences. The factory default is level 3 (VT300 mode, 7-bit controls).

NOTE: When you change the operating level, the terminal performs a hard reset (RIS). See Chapter 13 for details.

Table 4-2 DECSCL Sequence

Sequence							Level Selected
							< Level 1 >
CSI 9/11	6 3/6	1 3/1	" 2/2	p 7/0			VT100 mode
							< Level 3 >
CSI 9/11	6 3/6	3 3/3	" 2/2	p 7/0			VT300 mode, 8-bit controls
CSI 9/11	6 3/6	3 3/3	; 3/11	0 3/0	" 2/2	p 7/0	VT300 mode, 8-bit controls
CSI 9/11	6 3/6	3 3/3	; 3/11	2 3/2	" 2/2	p 7/0	VT300 mode, 8-bit controls
CSI 9/11	6 3/6	2 3/2	" 2/2	p 7/0			VT300 mode, 8-bit controls
CSI 9/11	6 3/6	2 3/2	; 3/11	0 3/0	" 2/2	p 7/0	VT300 mode, 8-bit controls
CSI 9/11	6 3/6	2 3/2	; 3/11	2 3/2	" 2/2	p 7/0	VT300 mode, 8-bit controls
CSI 9/11	6 3/6	3 3/3	; 3/11	1 3/1	" 2/2	p 7/0	VT300 mode, 7-bit controls (default)
CSI 9/11	6 3/6	2 3/2	; 3/11	1 3/1	" 2/2	p 7/0	VT300 mode, 7-bit controls

4.4 SENDING C1 CONTROLS TO THE HOST

The VT382 can send C1 control characters to the host as single 8-bit characters or as 7-bit escape sequences. You should select the format that matches the operating level you are using. You can use the following sequences to select the format for C1 control characters. See Chapter 2 for information on working with 7-bit and 8-bit environments.

4.4.1 Select 7-Bit C1 Control Characters (S7C1T)

The following sequence causes the terminal to send all C1 characters as 7-bit escape sequences.

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

This sequence changes the terminal mode as follows.

Mode before	Mode After
VT300 mode, 8-bit controls	VT300 mode, 7-bit controls.
VT300 mode, 7-bit controls	Same, Terminal ignores sequence.
VT100 mode or VT52 mode	Same, Terminal ignores sequence.

4.4.2 Select 8-Bit C1 Control Characters (S8C1T)

The following sequence causes the terminal to send C1 control characters to the host as single 8-bit characters.

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

This sequence changes the terminal mode as follows.

Mode Before	Mode After
VT300 mode, 8-bit controls	Same, Terminal ignores sequence.
VT300 mode, 7-bit controls	VT300 mode, 8-bit controls.
VT100 mode or VT52 mode	Same, Terminal ignores sequence.

CHAPTER 5

USING CHARACTER SETS

This chapter describes how you can select different character sets to use with your VT382 terminal. This chapter assumes you are familiar with the character-encoding concepts described in Chapter 2.

You can use two types of character sets in the terminal, hard sets and soft sets. Hard character sets are the character sets built into the VT382, such as the ASCII and DEC Supplemental Graphic sets. Chapter 2 shows the built-in character sets. Soft character sets are sets that you down-line-load into the terminal from a computer. You can design your own soft character sets. The number of hard sets available depends on the operating mode you select: VT100 or VT300 (Chapter 4).

5.1 SELECTING CHARACTER SETS

To understand how to select character sets, you must first understand the function of the terminal's in-use table. The in-use table is an 8-bit code table the terminal uses to interpret received characters. The in-use table defines the character sets the terminal can currently use. You can define any two character sets in the terminal's in-use table.

The in-use table has four parts, based on character position in the code table.

C0 control character set	positions 0/0 through 1/15
Left graphic (GL) set	positions 2/1 through 7/14
C1 control character set	positions 8/0 through 9/15
Right graphic (GR) set	positions 10/0 through 15/15

USING CHARACTER SETS

Together, the ASCII and DEC Supplemental Graphic sets make up the DEC Multinational set. The ASCII and ISO Latin-1 supplemental sets make up the ISO Latin-1 set, the international standard.

You can select a different character set by following these two steps.

1. Designate the set as G0, G1, G2, or G3.
   G0 through G3 are logical sets that the terminal uses to access character sets. You can designate up to four character sets and have them ready for use in the in-use table.

2. Map the designated set into the in-use table.
   After you map the set into the in-use table, you can display or send any character from that set using 8-bit codes.

Each time you turn on the terminal, the terminal places the following default character sets in the in-use table.

Kanji Terminal	JIS-Roman or ASCII	as G0	
	DEC Special	as G1	
	JIS-Katakana	as G2	
	DEC Kanji	as G3	
		G0 into GL	G3 into GR
Katakana Terminal	JIS-Roman or ASCII	as G0	
	JIS-Katakana	as G1	
	JIS-Katakana	as G2	
	DEC Special	as G3	
		G0 into GL	G2 into GR
in VT52 mode (Katakana only)	JIS-Roman or ASCII	into GL	
	JIS-Katakana	into GR	

* G0(JIS-Roman or ASCII) is defined in the "Terminal Set-Up" screen.

Figure 5-1 shows how you select character sets. The following sections describe the control functions you use to designate and map character sets.

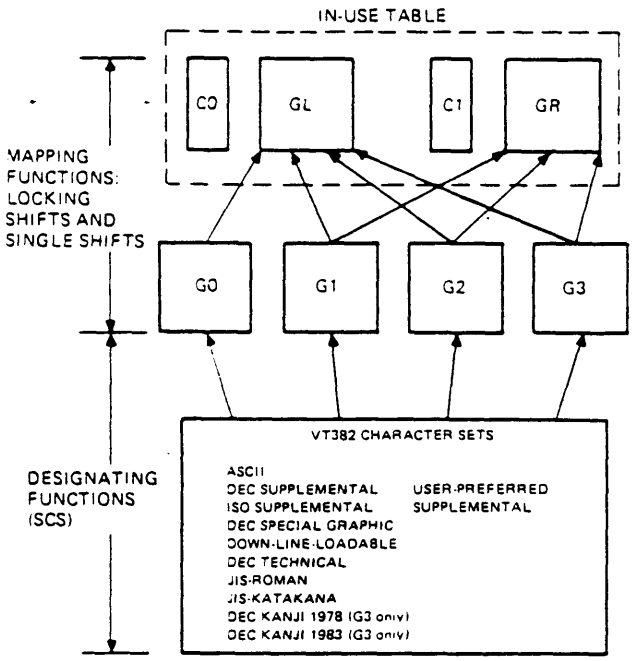

Figure 5-1 Character Set Selection

5.1.1 Designating Character Sets (SCS Sequences)

You designate a hard character set as G0 through G3 by using a select character set (SCS) escape sequence. You cannot designate a 96-character set as G0.

SCS sequences use the format shown in Table 5-1. The table lists the code used to select each available character set.

> **NOTE: The ISO Latin-1 supplemental character set is the only built-in set with 96 characters.**

Table 5-1 Designating Character Sets

ESC 1/11	Intermediate *********	Final ****	
Intermediate		**Final**	
To Select	**Use**	**To Select**	**Use**
< 94-Character Sets >		**< 94-Character Sets >**	
G0	(2/8	ASCII	B 4/2
G1	) 2/9	DEC Supplemental Graphic (*1)	% 2/5 5 3/5
G2	* 2/10	User preferred Supplemental (*1)	< 3/12
G3	+ 2/11	DEC Special Graphic	0 3/0
		DEC Technical (*1)	> 3/14
		JIS-Roman	J 4/10
		JIS-Katakana	I 4/9
< 96-Character Sets >		**< 96-Character Sets >**	
G1	- 2/13	ISO Latin-1 Supplemental (*1)	A 4/1
G2	. 2/14		
G3	/ 2/15		
		(*1) Available in VT300 mode only	

Table 5-1 Designating Character Sets (cont)

ESC 1/11	Intermediate *********	Final ****

Intermediate		Final		
To Select	Use	To Select	Use	
< 2-Byte Character Sets >		< 2-Byte Character Sets >		
G3	S + 2/4 2/11	DEC Kanji (1978) (*2)	1 3/1	or @ 4/0
		DEC Kanji (1983) (*2)	3 3/3	or B 4/2

(*2) DEC Kanji 1978 or 1983 is determined only in the "Terminal Set-Up" screen. You cannot select by escape sequence.

(Examples)

. The following sequence designates the DEC Special Graphic character set as G1 logical set.

ESC) 0

. The following sequence designates the ISO Latin-1 Supplemental character set as the G3 logical set.

ESC / A

5.1.2 Mapping Character Sets

After you designate a character set as G0, G1, G2, or G3, you must map the set into the in-use table as GL or GR. To map a set, you use locking-shift or single-shift control functions.

Figure 5-2 shows how you use locking shifts and single shifts.

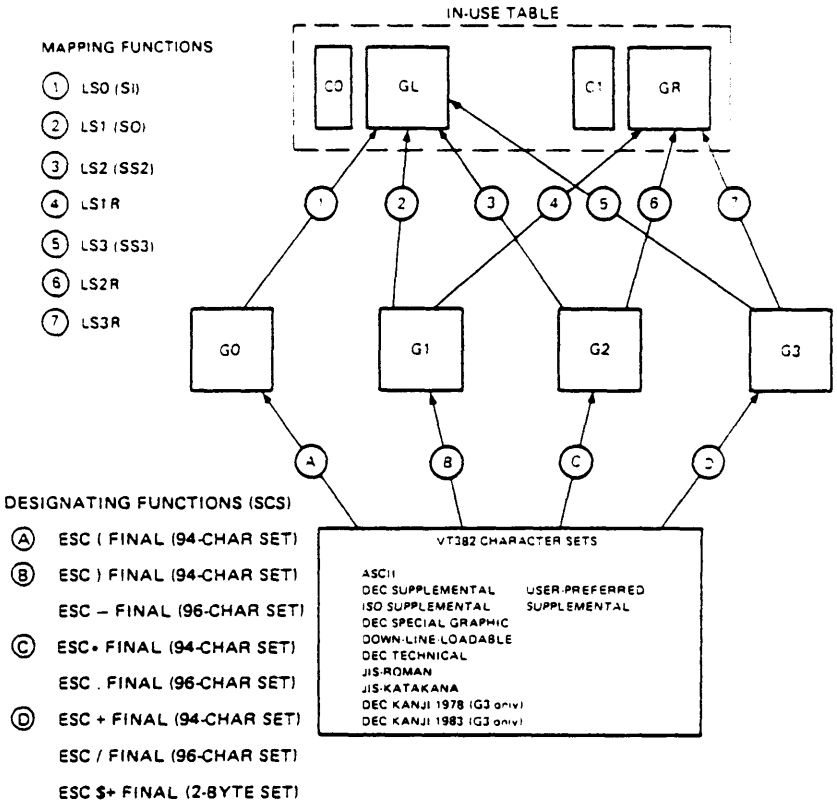

Figure 5-2 Designating and Mapping Character Sets

5.1.2.1 Locking Shifts

When you use a locking shift, the character set remains in GL or GR until you use another locking shift. Table 5-2 lists all locking shifts available.

Table 5-2 Mapping Character Sets with Locking Shifts

Locking Shift	Code		Function
LS0 (locking shift 0)	SI 0/15		Map G0 into GL. (default)
LS1 (locking shift 1)	S0 0/14		Map G1 into GL.
LS1R (locking shift 1, right)	ESC 1/11	~ 7/14	Map G1 into GR. *
LS2 (locking shift 2)	ESC 1/11	n 6/14	Map G2 into GL.
LS2R (locking shift 2, right)	ESC 1/11	} 7/13	Map G2 into GR. *
LS3 (locking shift 3)	ESC 1/11	o 6/15	Map G3 into GL.
LS3R (locking shift 3, right)	ESC 1/11	| 7/12	Map G3 into GR. *

* Available in in 8-bit environment.

(Examples)

. The following sequences designate the DEC Special Graphic character set as G1, then map G1 into GL.

ESC) 0	SO
designate as G1	map G1 into GL

- The following sequences designate the ISO Latin-1 supplemental character set as G2, then map G2 into GR.

ESC . A	ESC }
designate as G2	map G2 into GR

5.1.2.2 Single Shifts

You use a single shift when you want to display the next character from a different character set. A single shift maps the G2 or G3 set into GL. The character set is active for only one character. Then the terminal returns to the previous character set in GL.

The terminal has two single-shift control functions available.

Table 5-3 shows about single-shift.

Table 5-3 Single Shift

Single-Shift Control	8-Bit Character	7-Bit Equivalent Sequence		Function
Single shift 2	SS2 8/14	ESC 1/11	N 4/14	Maps G2 into GL for the next character.
Single shift 3	SS3 8/15	ESC 1/11	O 4/15	Maps G3 into GL for the next character.

(Example)
Suppose the ASCII character set is in GL. You want to display the line feed character from the DEC Special Graphic character set, already designated as G3. You do not want to replace the ASCII set just to display one character. Instead, you can use single shift 3 to temporarily map the DEC Special Graphic set (G3) into GL.

SS3	LF
8/15	0/10
single shift 3	line feed character

After displaying the line feed character, the terminal maps the ASCII set (G1) back into GL, replacing the DEC Special Graphic set (G3).

5.1.3 User-Preferred Supplemental Character Sets

You can assign the supplemental character set you use most often as a special standby set. This standby set is called the user-preferred supplemental set. This feature provides applications with an easy way to access the user's preferred supplemental set.

You can assign the DEC Supplemental Graphic or ISO Latin-1 supplemental set as the standby set. After you assign a set, you must designate and map the set before using it.

1. Designate the set as G1, G2, or G3.
2. Map the set into GR.

For more information on designating and mapping sets, see "Selecting Character Sets" in this chapter.

You can assign a supplemental character set as follows.

Assign User-Preferred Supplemental Set (DECAUPSS)

Default: DEC Supplemental Graphic

Sequence							Function
DCS 9/0	0 3/0	! 2/1	u 7/5	% 2/5	5 3/5	ST 9/12	Assigns the DEC Supplemental Graphic set as the preferred supplemental set.
DCS 9/0	1 3/1	! 2/1	u 7/5	A 4/1	ST 9/12		Assigns the ISO Latin-1 supplemental set as the preferred supplemental set.

5.2 ANSI CONFORMANCE LEVELS

This control function lets an application map certain character sets into the terminal's in-use table as default sets. The character sets are based on ANSI conformance levels, listed below. These conformance levels are from the dpANSI X.3134.1 standard.

ANSI conformance levels represent an agreement between the sender and receiver for compatible data exchange. The control function acts an announcer for the data exchange that follows between the terminal and application software. The control function selects which character sets the terminal used by default in the data exchange.

The VT382 supports three ANSI conformance levels.

ANSI Levels 1 and 2
- . ASCII designated as G0.
- . ISO Latin-1 supplemental designated as G1.
- . G0 mapped into GL.
- . G1 mapped into GR.

ANSI Level 3
- . ASCII designated as G0.
- . G0 mapped into GL.

The announcer function is function is as follows.

NOTE: Available in VT300 mode only.

ESC	sp	Final
1/11	2/0	4/?

where

Final indicates the ANSI conformance level for the following data exchange.

Final	ANSI Conformance Level
L	Level 1
M	Level 2
N	Level 3

Notes

- If the terminal is reset, turned off, or changed with a set conformance level (DECSCL) sequence, software must send another announcer sequence to the terminal. Otherwise, the terminal uses the default character sets (ASCII in GL, DEC Hanzi in GR).

- Do not confuse ANSI conformance levels with Digital conformance levels (Chapter 4).

5.3 SOFT CHARACTER SETS

You can down-line-load a soft character set from the host computer into the terminal. This feature lets you design your own soft character sets for use with the terminal. You can only load soft character sets in VT300 mode.

The soft character set is also known as a dynamically redefinable character set (DRCS). The terminal stores the soft characters in its DRCS buffer.

NOTE: The terminal does not store the soft character set in nonvolatile RAM. When you turn off the terminal, the soft characters are lost.

The next section describes the guidelines for designing a soft set. The sections that follow describe how to code, load, designate, and clear a soft set.

5.3.1 Designing A Soft Character Set

Your terminal displays each character by turning on a series of pixels. A pixel (picture element) is the smallest displayable unit on the screen. Each character must fit in a limited area, called the character cell. The VT382 uses a default character cell size of 10 X 20 (ASCII, 80 columns) pixels.

When you design a character, you should lay out a character cell on grid paper. The little boxes on the grid paper represent pixels. You fill in the pixels that male up the character. The next section shows an example of a character design.

You can design characters for an 80-column or 132-column font. The largest character cell you can use is the maximum size of 12 X 30 pixels (360 pixels) for an 80-column font. Figure 5-3 shows the cell sizes for 80- and 132-column fonts. The built-in fonts supplied by Digital follow the guidelines in Table 5-4.

You must design your characters to fit the cell, The terminal ignores characters that are larger than the cell size.

Figure 5-4 shows an example of this spacing for an uppercase D character. In this example, the character for the 80-column font has two pixel columns reserved for spacing.

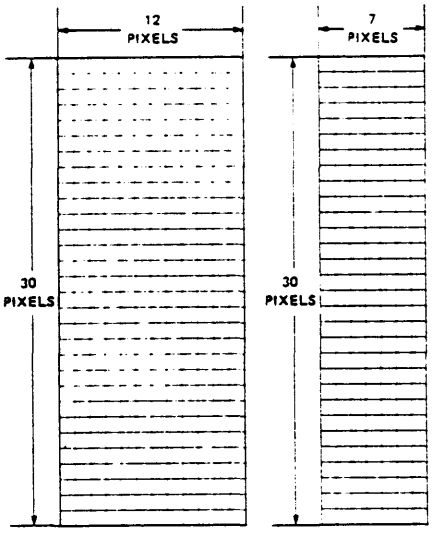

Figure 5-3 Character Cell Sizes for 80- and 132-Column Fonts

Table 5–4 Guidelines for Designing Soft Characters

Character Dimension	80-Column Font	132-Column Font
Cell width	12 pixels	7 pixels
Cell height	30	30
Body width	10	6
Body height	17	17
Ascender height	5	5
Descender height	8	8
Spacing before character	1	1
Spacing after character	1	0

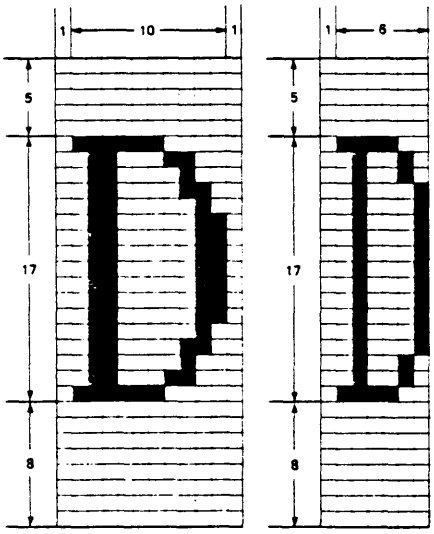

Figure 5–4 Character Body Sizes for 80- and 132-Column Fonts

5.3.2 Coding The Soft Character Set

After you design your characters, you must code them for the terminal. This section describes how to code soft characters. The next section describes how to load the character codes into the terminal.

Each pixel of a soft character cell receives a binary value of 0 or 1. A 1 bit indicates the pixel is on, and a 0 bit indicates the pixel is off.

The terminal receives the code for a soft character in sections, called sixels. A sixel is a 6-bit binary code that represents a vertical column of 6 pixels on the screen. Each bit in a sixel corresponds to a pixel on the screen. The following example describes how to design and code a soft character.

(Example)
Suppose you want to design an uppercase D for an 80-column font.

1. Draw your design on a grid.

   Use the grid for an 80-column character cell to draw your design. Mark which pixels will be on and which pixels will be off. Your design may look like Figure 5-4.

2. Divide the character cell into columns of 6 bits each.

   Use the format shown in Figure 5-5. Each 6-bit pattern represents 6 pixels or a sixel. The least significant bit is at the top, and the most significant bit is at the bottom. The terminal would receive the sixel columns in order (1 to 12), starting with Group A.

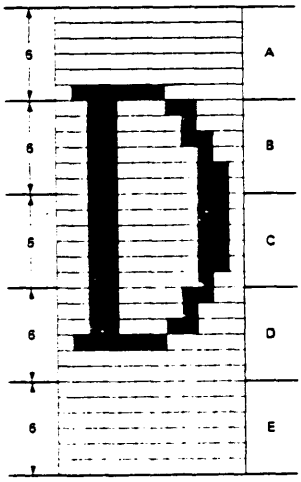

Figure 5-5 Example of an Uppercase D in an 80-Column Font

3. Convert the binary value of each sixel to its hexadecimal value.

   Sixel codes are restricted to characters in the range of ? (hex 3F) to ~ (hex 7E), so you must add an offset of hex 3F to the hex value of each column. For example,

000000(2)	= 00(16)	110101	= 35(16)	111111(2)	= 3F(16)
	+ 3F(16)		+ 3F(16)		+ 3F(16)
	3F(16)		74(16)		7E(16)

4. Use Table 5-5 to convert each binary number to the equivalent ASCII character.

   Table 5-5 lists the results of steps 3 and 4 for each possible binary value. All you have to do is find the 6-digit binary number for each sixel bit pattern in your character design. Figure 5-6 shows this conversion for the uppercase D in this example.

You use this procedure to convert each character of your soft character set into a string of sixel bit patterns. Then you can down-line-load your DRCS characters into the terminal, using the DECDLD device control string described in the next section.

Table 5-5 Converting Binary Code to an ASCII Character

Binary Value	Hex Value	Hex Value + 3F Offset	Character Equivalent
000000	00	3F	?
000001	01	40	@
000010	02	41	A
000011	03	42	B
000100	04	43	C
000101	05	44	D
000110	06	45	E
000111	07	46	F
001000	08	47	G
001001	09	48	H
001010	A	49	I
001011	B	4A	J
001100	C	4B	K
001101	D	4C	L
001110	E	4D	M
001111	F	4E	N
010000	10	4F	O
010001	11	50	P
010010	12	51	Q
010011	13	52	R
010100	14	53	S
010101	15	54	T
010110	16	55	U
010111	17	56	V
011000	18	57	W
011001	19	58	X
011010	1A	59	Y
011011	1B	5A	Z
011100	1C	5B	[
011101	1D	5C	\
011110	1E	5D	]
011111	1F	5E	^
100000	20	5F	_

Table 5-5 Converting Binary Code to an ASCII Character (cont)

Binary Value	Hex Value	Hex Value + 3F Offset	Character Equivalent
100001	21	60	`
100010	22	61	a
100011	23	62	b
100100	24	63	c
100101	25	64	d
100110	26	65	e
100111	27	66	f
101000	28	67	g
101001	29	68	h
101010	2A	69	i
101011	2B	6A	j
101100	2C	6B	k
101101	2D	6C	l
101110	2E	6D	m
101111	2F	6E	n
110000	30	6F	o
110001	31	70	p
110010	32	71	q
110011	33	72	r
100100	34	73	s
110101	35	74	t
110110	36	75	u
110111	37	76	v
111000	38	77	w
111001	39	78	x
111010	3A	79	y
111011	3B	7A	z
111100	3C	7B	{
111101	3D	7C	|
111110	3E	7D	}
111111	3F	7E	~

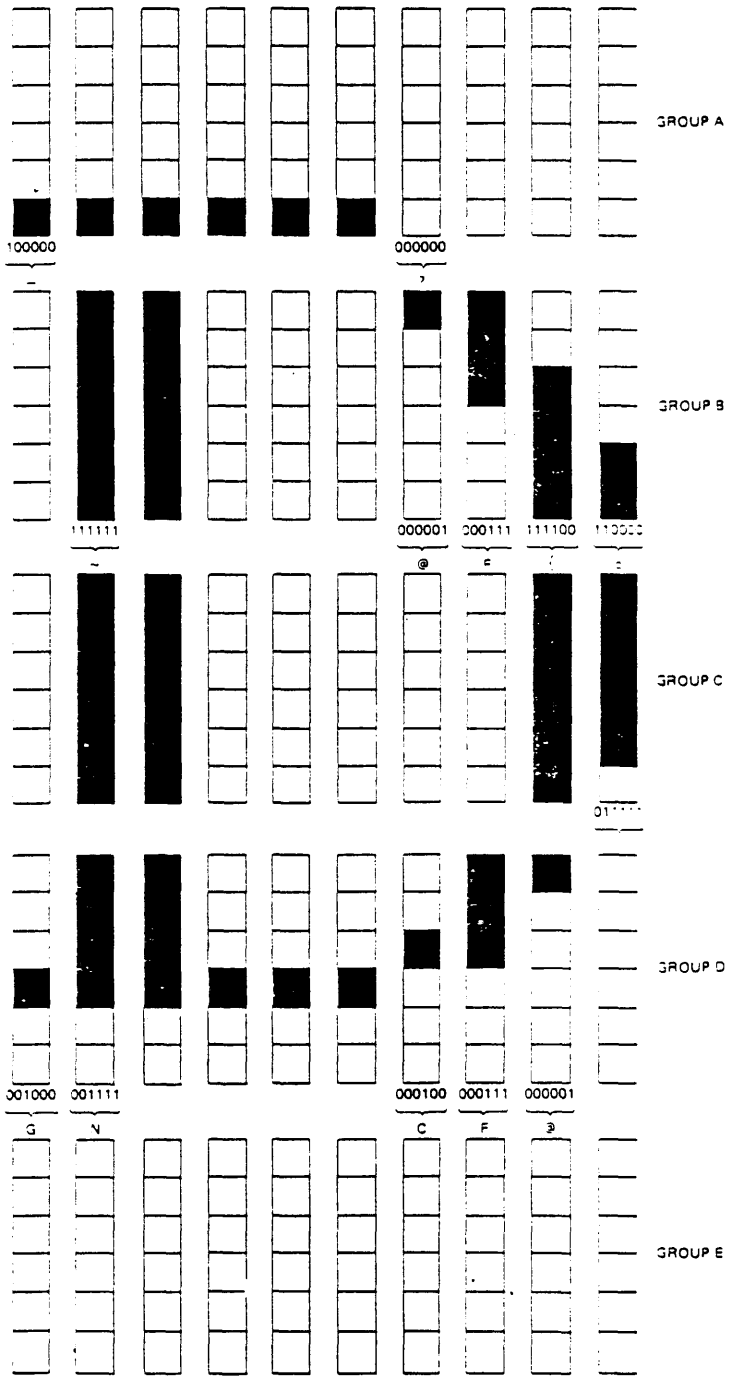

Figure 5-6 Sixel-to-ASCII Conversion

5.3.3 Down-Line-Loading Soft Characters

You can load two font renditions of your soft character set, an 80-column font or 132-column font.

You should load both an 80-column and a 132-column rendition of your soft set. Then the terminal can select the correct rendition if you change the width of the scrolling region (Chapter 6).

You load your soft character set with a DECDLD device control string. This control string has the following format.

NOTE: See Chapter 2 for general information about device control strings.

```
DCS Pfn ; Pcn; Pe ; Pcmw ; Pw ; Pt ; Pcmh ; Pcss {

Dscs Sxbp1 ; Sxbp2 ;...; Sxbpn ST
```

where

DCS (9/0)
: is the device control string introducer. DCS is an 8-bit C1 character. You can use the equivalent 7-bit sequence ESC P (1/11, 5/0) when coding for a 7-bit environment.

Pfn ; Pcn ; Pe ; Pcmw ; Pw ; Pt ; Pcmh ; Pcss
: are parameter characters, separated by semicolons ;(3/11). Table 5-6 describes these parameters and list their possible values. Table 5-7 describes the combinations of Pcmw. PT, and Pcmh you can use for 80- and 132-column fonts. If you use any other combinations, the terminal ignores the DECDLD string.

{ (7/11)
: is the final character. It makes the end of the parameter characters and indicates that this string is a DECDLD function.

Dscs
: defines the name for the soft character set. You use this name in the select character set (SCS) escape sequence. You use the following format for the Dscs name.

I F

where

I
is 0, 1 or 2 intermediate characters from the range 2/0 to 2/15 in the ASCII character set.

F
is a final character in the range 3/0 to 7/14.

(Examples of Dscs Names)

Name	Function
sp @ 2/0 4/0	Defines the character set as an unregistered soft set. This value is the recommended default for user-defined sets. The value of Pcss defines whether this set has 94 or 96 characters.
& % C 2/6 2/5 4/3	Defines the soft character set as &%C, which is currently an unregistered set. The value of Pcss defines whether this set has 94 or 96 characters.

Sxbp1 ; Sxbp2 ;...; Sxbpn
are the sixel bit patterns for individual characters, separated by semicolons (3/11). Your character set can have 1 to 94 patterns or 1 to 96 patterns, depending on the setting of the character set size parameter (Pcss). Each sixel bit pattern is in the following format.

S...S/S...S

where

< each S...S >
represents the sixels in each group of the soft character (Figure 5-5)

/ (2/15)
advances the sixel pattern to next group of the soft character

ST (9/12)
is the string terminator. ST is an 8-bit C1 character. You can use the equivalent 7-bit sequence ESC \ (1/11, 5/12) when coding for a 7-bit environment.

After you load your soft character set, you must designate the set as G0, G1, G2, or G3.

Table 5-6 DECDLD Parameter Characters

Parameter	Name	Description
Pfn	Font number	Selects the DRCS font buffer to load. The VT382 has one DRCS font buffer. Pfn has two valid values, 0 and 1. Both values refer the same DRCS buffer.
Pcn	Starting character	Selects where to load the first character in the DRCS font buffer. The location corresponds to a location in the ASCII code table (Figure 2-1). Pcn is affected by the character set size. (See Pcss below.) In a 94-character set, a Pcn value of 0 or 1 means that the first soft character is loaded into position 2/1 of the character table. In a 96-character set, a Pcn value of 0 means the first character is loaded into position 2/0 of the character table. The greatest Pcn value is 95 (position 7/15).
Pe	Erase	Selects which characters to erase from the DRCS buffer before loading the new font. 0 or 2 = erase all characters in the DRCS buffer with this number, width and rendition. (default) 1 = erase only characters in locations being reloaded.

Table 5-6 DECDLD Parameter Characters (cont)

Parameter	Name	Description
Pcmw	Character matrix width	Selects the maximum character cell width. 0 = 10 pixels wide for 80 columns. 6 pixels wide for 132 columns. (default) 1 = illegal. 2 = 5 x 10 pixel cell (VT200 mode compatible). 3 = 6 x 10 pixel cell (VT200 mode compatible). 4 = 7 x 10 pixel cell (VT200 mode compatible). 5 = 5 pixel wide. 6 = 6 pixel wide. . . 12 = 12 pixel wide. If you omit a Pcmw value, the terminal uses the default character width. Any Pcmw value over 12 is illegal. Use Pcmw values 2 through 4 with VT200 compatible software. Remember that VT200 fonts appear different on the VT382.
Pw	Font Width	Selects the number of columns per line (font set size). 0 = 80 columns. (default) 1 = 80 columns. 2 = 132 columns.

Table 5-6 DECDLD Parameter Characters (cont)

Parameter	Name	Description
Pt	Text or full-cell	Defines the font as a text font or full-cell font. 0 = text. (default) 1 = text. 2 = full cell. Full-cell fonts can individually address all pixels in a cell. Text fonts cannot individually address all pixels. If you specify a text cell, the terminal automatically performs spacing and centering of the characters.
Pcmh	Character matrix height	Selects the maximum character cell height. 0 or omitted = 20 pixels high. (default) 1 to 10 = 10 pixel high. 11 to 20 = 20 pixel high. 21 to 30 = 30 pixel high. Pcmh values over 31 are illegal. If the value of Pcmw is 2, 3, or 4, Pcmh is ignored.
Pcss	Character set size	Defines the character set as a 94- or 96-character graphic set. 0 = 94-character set. (default) 1 = 96-character set.

The value of Pcss changes the meaning of the Pcn (starting character) parameter above.

(Examples)

. If Pcss = 0 (94-character set)

The terminal ignores any attempt to load character into the 2/0 or 7/15 table positions.

Pcn	Specifies
1	column 2/row 1
.	
.	
94	column 7/row 14

. If Pcss = 1 (96-character set)

Pcn	Specifies
0	column 2/row 0
.	
.	
95	column 7/row 15

Table 5-7 Valid DECDLD Parameter Combinations

Pt	Pcmw	Pcmh	Pw
<80-Column Fonts>			
0,1	0 to 10	0 to 30	0, 1
2	0 to 12	0 to 30	0, 1
<132-Column Fonts>			
0,1	0 to 6	0 to 30	2
2	0 to 7	0 to 30	2

5.3.4 Designating The Soft Character Set

You designate your soft character set the same way you designate the hard character sets - using a select character set (SCS) sequence. You also use the same format for the SCS sequence.

ESC	Intermediate(s)	Final
1/11	**********	*****

where

Intermediate(s)
: are one or more characters that designate the soft character set as one of the logical sets, G0 through G3. You use the same intermediate characters that you use for hard character sets (Table 5-1).

  The intermediate character(s) also indicates that the soft character set is a 94- or 96-character set. Make sure you use an intermediate character that matches the setting of the character set size parameter (Pcss) in the DECDLD string (Table 5-6).

Final
: is the Dcsc name you used for the soft character set in the DECDLD string.

Notes on Designating Soft Character Sets

- Replacing a soft set with a soft set
  If you use a new Dscs name when you replace the current soft set with another soft set, then the following occurs.

  - Characters from the old soft set are undefined. If you redefine the soft set, characters currently on the screen may change.

  - Any logical sets (G0, G1, G2, G3) used to designate the old soft set are undefined. The in-use table is also undefined.

  After you load a new soft set, use a select character set (SCS) Sequence to designate the soft set, Using SCS eliminates the confusion involved with undefined characters.

- Replacing a hard set with a soft set
  You can define a soft set that replaces one of the hard sets (such as ASCII or DEC Special Graphic).
  A soft set that replaces a hard set remains in effect until you perform one of the following actions.

  - Clear the soft (set using the Recall Set-Up feature, or the power-up self-test).

- Redefine the soft set (using another DECDLD string).

5.3.5 Soft Character Set Example

Suppose you want to create a soft character set containing a solid rectangle, a blank, a rectangular box, and a striped rectangle. This example shows how you would

- down-line-load the set,
- designate the set as G1, and
- map the G1 set into GL.

NOTE: Make sure the terminal is in VT300 mode before you try to load a soft character set. You cannot load soft sets in VT100 and VT52 mode.

1. You could use the following DECDLD string to load your character set. (The string is shown divided into sections for clarity.)

```
DCS

   1 ; 1 ; 0 ; 8 ; 1 ; 1 ; 20 ; 0 { sp @

   ~~~~~~~~/~~~~~~~~/~~~~~~~~/~~~~~~~~;

   ????????/????????/????????/????????;

   ~@@@@@@~/~??????~/~??????~/~AAAAAA~;

   TTTTTTTT/TTTTTTTT/TTTTTTTT/TTTTTTTT;

   ST
```

where

DCS (9/0)
introduces the device control string.

1 ; 1; 0 ; 8 ; 1 ; 1 ; 20 ; 0
is the parameter string specifying the following.
(See Table 5-6 for parameter definitions.)

Parameter	Function
Pfn = 1	Loads this soft set into the DRCS font buffer.
Pcn = 1	Selects the character at row 2/ column 1 in the ASCII table (Chapter 2) as the first character to load.
Pe = 0	Erases all characters in the font buffer for that rendition.
Pcmw = 8	Selects a maximum character width of 8 pixels.
Pw = 1	Selects a font width of 80 columns.
Pt = 1	Defines the set as a text font.
Pcmh = 20	Selects a maximum character height of 20 pixels.
Pcss = 0	Defines the set as a 94-character set.

{ (7/11)
indicates the end of the parameter characters and specifies that this sequence is a DECDLD string.

sp @ (2/0, 4/0)
defines the character set as an unregistered soft set. This value is the recommended default value for user-defined sets. The sp represents one space. You can use other values to define other specific character sets.

~~~~~~~~/~~~~~~~~/~~~~~~~~/~~~~~~~~
represents the first character (a solid rectangle).

; (3/11)
separates the soft characters.

????????/????????/????????/????????
represents the second character (a blank).

~@@@@@@~/~??????~/~??????~/~AAAAAA~
represents the third character (a hollow rectangle).
~~~~~~~~

TTTTTTTT/TTTTTTTT/TTTTTTTT/TTTTTTTT
represents the fourth character
(a set of horizontal stripes).

ST (9/12)
indicates the end of the DECDLD string.

2. Now you are ready to designate the character set as G1. You can use the following SCS escape sequence.

ESC) sp @

where

ESC (1/11)
introduces the SCS sequence.

) (2/9)
designates the character set as G1.

sp @ (2/0, 4/0)
selects the soft set as the set to designate as G1. Remember, sp @ was the name used for the soft set in the DECDLD string.

3. Finally, you want to map the G1 set into the in-use table as GL. You can map the set by sending a shift out (SO) control character. To send the SO character, you hold down the Ctrl key and press the N key.

NOTE: For information on using shift characters, see Chapter 3. For information on mapping sets, see "Mapping Character Sets" in this chapter.

The soft character set should now be loaded and ready for use.

5.3.6 Clearing A Soft Character Set

You can clear a soft charter set that you loaded into the terminal by using the following DECDLD control string.

DCS	1	;	1	;	2	{	sp	@	ST
9/0	3/1	3/11	3/1	3/11	3/2	7/11	2/0	4/0	9/12

- Performing the power-up self-test.
- Selecting the "Recall" or "Reset Terminal" Set-Up features.
- Using a reset to initial state (RIS) or ESC c sequence.

CHAPTER 6

SCREEN DISPLAY CONTROL FUNCTIONS

This chapter describes the control functions that affect how the terminal displays data, including the status line. The chapter also describes the control functions that let you change the format of the display.

This chapter assumes you are familiar with the character-encoding concepts described in Chapter 2.

6.1 DISPLAY CONTROL FUNCTIONS

This section describes control functions that determine the screen background, the scrolling speed, the type of status line, when to display the status line, and whether or not to display keyboard data.

6.1.1 Local Echo: Send/Receive Mode (SRM)

This control function determines whether or not the terminal displays keyboard data. When local echo is on, the terminal sends keyboard characters to the screen and the host. The host does not have to send (echo) the characters back to the terminal display. When local echo is off, the terminal only sends characters to the host. It is up to the host to echo characters back to the screen.

Default: No local echo

Mode	Sequence				Action
Set (No local echo)	CSI 9/11	1 3/1	2 3/2	h 6/8	Turns local echo off. The terminal sends keyboard characters to the host only. The host can echo the characters back to the screen.
Reset (Local echo)	CSI 9/11	1 3/1	2 3/2	l 6/12	Turns local echo on. The terminal sends keyboard characters to the host and to the screen. The host does not have to echo characters back to the terminal.

6.1.2 Light Or Dark Screen: Screen Mode (DECSCNM)

This control function selects a dark or light background on the screen.

Default: Dark background

Mode	Sequence				Action
Set (Light background)	CSI 9/11	? 3/15	5 3/5	h 6/8	Selects reverse video. The screen displays dark characters on a light background.
Reset (Dark background)	CSI 9/11	? 3/15	5 3/5	l 6/12	Selects a normal display. The screen displays light characters on a dark background.

6.1.3 Scrolling Mode (DECSCLM)

This control functions selects the rate at which the terminal scrolls lines. You can select one of two scrolling styles, smooth or jump.

Default: Smooth scroll

Mode	Sequence				Action
Set (Smooth)	CSI 9/11	? 3/15	4 3/4	h 6/8	Selects smooth scroll. The terminal adds lines to the screen at a rate of 6 line per second.
Reset (Jump)	CSI 9/11	? 3/15	4 3/4	l 6/12	Selects jump scroll. The terminal can add lines to the screen as fast as it receives them.

6.1.4 Selecting The Indicator Or Host Status Line

The twenty-fifth line of the screen is reserved for the status line. The terminal lets you use the status line in two ways - as an indicator of the terminal's current state, or as a window the host can use to display application-specific messages.

The indicator status line displays information about the current state of the terminal. If this status line is enabled, it appears in reverse video on the twenty-fifth screen line. The default setting of the indicator status line is enabled. This status line always appears in Set-Up.

The indicator status line displays the following information about the terminal.

- Text cursor position (line, column)
- Selected character sets
- Printer status
- Modem status

The host-writable status line can display specific information from the host. That is, you can program this status line to display any information you want. You use the following control sequences to send data to the status line and select the type of status line.

6.1.4.1 Select Status Line Type (DECSSDT)

This control function lets the host select the type of status line displayed on line 25 of the screen.

NOTE: Available in VT300 mode only.

Default: Indicate status line

CSI	Ps	$	~
9/11	3/?	2/4	7/14

where

Ps indicates which status line the host selects, as follows.

Ps	Status Line Selected
0	No status line (The 25th line is blank.)
1 (default)	Indicator status line
2	Host-writable status line

Notes on DECSSDT

- If you change from an indicator to a host-writable status line, the new host-writable status line is empty.
- When you select the host-writable status line, most control functions that affect the main display also affect the status line. Table 6-1 describes the exceptions.
- DECSSDT does not change the type of status line displayed in Set-Up. In Set-Up, the VT382 always uses the indicator status line.

Table 6-1 Effect of Control Functions on the Status Line

Control Function	Mnemonic	Action
ANSI mode	DECANM	Ignored if received in the status line.
Cursor position controls	-	Only the column parameters in cursor positioning commands operate in the status line.
Hard terminal reset	RIS	Erase and exits the status line.
Invoke confidence test	DECTST	Exits the status line. Erase the host-writable status line.
Select status display type	DECSSDT	If the status display type is chosen other than "Host-Writable", the status line is exited.
Set conformance level	DECSCL	Exits the status line.
Sixel	-	Ignored if received in the status line.
Soft terminal reset	DECSTR	Exits the status line.
Text cursor enable mode	DECTCEM	You can individually enable the cursor in the main display or the status line.

6.1.4.2 Select Active Status Display (DECSASD)

This control function selects whether the terminal sends data to the main display or the status line. The main display is the first 24 lines on the screen. The status line is the twenty-fifth line.

NOTE: Available in VT300 mode only.

Default: Main display

CSI	Ps	$	}
9/11	3/?	2/4	7/13

where

Ps represents the display the terminal sends data to, as follows.

Ps		Action
0	(default)	Selects the main display. The terminal sends data to the main display only.
1		Selects the status line. The terminal sends data to the status line only.

6.1.5 Sixel Display Mode (DECSDM)

The sixel display mode (DECSDM) controls the Sixel and ANSI text interactions and whether Sixels scroll when the bottom margin is reached.

Default: Enable sixel scroll

Mode	Sequence					Action
Set (Disable sixel scroll)	CSI 9/11	? 3/15	8 3/8	0 3/0	h 6/8	Disabale sixel scroll.
Reset (Enable sixel scroll)	CSI 9/11	? 3/15	8 3/8	0 3/0	l 6/12	Enable sixel scroll.

6.1.6 Control Representation Mode (CRM)

The control representation mode (CRM) determines whether the terminal displays control codes as graphic character or interpret them for display control. See chapter 2 "DISPLAY CONTROLS MODE" for detail.

Default: Interpret controls

Mode	Sequence			Action
Set (Display controls)	CSI 9/11	3 3/3	h 6/8	Display controls on the screen.
Reset (Interpret controls)	CSI 9/11	3 3/3	l 6/12	Interpret controls.

6.1.7 Kanji/Katakana Display Mode (DECKKDM)

The Kanji/Katakana Mode determines if the VT382 acts as a Kanji Terminal or a Katakana Terminal. The reset or set state will set the terminal's current character sets of G0, G1, G2, G3, GL and GR as follows.

Defaults: Kanji terminal

Mode	Sequence					Action
Set (Kanji terminal)	CSI 9/11	? 3/15	5 3/5	9 3/9	h 6/8	Kanji Terminal G0 JIS-Roman or ASCII G1 DEC Special Graphic G2 JIS-Katakana G3 DEC Kanji G0 into GL G3 into GR
Reset (Katakana terminal)	CSI 9/11	? 3/15	5 3/5	9 3/9	l 6/12	Kanji Terminal G0 JIS-Roman or ASCII G1 JIS-Katakana G2 JIS-Katakana G3 DEC Special Graphic G0 into GL G2 into GR

6.2 CONTROLLING THE DISPLAY FORMAT

This section describes the control functions that let you change the height and width of the scrolling region, and determine whether or not the cursor can move outside the scrolling margins.

6.2.1 Setting 80 Or 132 Columns (DECCOLM)

The column mode (DECCOLM) control function sets the width of the scrolling region to 80 or 132 columns.

Default: 80 columns

Mode	Sequence				Action
Set (132)	CSI 9/11	? 3/15	3 3/3	h 6/8	Selects the 132-column font to display text on the screen.
Reset (80)	CSI 9/11	? 3/15	3 3/3	l 6/12	Selects the 80-column font to display text on the screen.

Notes on DECCOLM

- If you change the DECCOLM setting, the terminal
  - sets the top and bottom scrolling margins to their default positions.
  - erases all data on the screen including the status line.

6.2.2 Set Top And Bottom Margins (DECSTBM)

This control function sets the top and bottom margins to define the scrolling region. You cannot scroll outside the margins.

Default: Margins at screen limits.

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

where

Pt is the line number for the top margin.
Default: Pt = 1.

Pb is the line number for the bottom margin.
Default: Pb = 24.

Notes on DECSTBM

- The value of Pt must be less than Pb.
- The maximum size of the scrolling region is the screen size.
- DECSTBM moves the cursor to column 1, line 1 of the screen.

6.2.3 Origin Mode (DECOM)

This control function allows cursor addressing relative to the scrolling margins or the complete screen. DECOM determines if the cursor position is restricted to inside the margins. When you turn on or reset the terminal, you reset origin mode.

Default: Origin at upper-left of screen, independent of margins.

Mode	Sequence				Action
Set (Margin-dependent)	CSI 9/11	? 3/15	6 3/6	h 6/8	Sets the home cursor position at the upper-left corner of the screen, within the margins. The starting point for line numbers depends on the current margin settings. The cursor cannot move outside of the margins.
Reset (Margin-independent)	CSI 9/11	? 3/15	6 3/6	l 6/12	Sets the home cursor position at the upper-left corner of for line numbers is independent of the margins. The cursor can move outside of the margins.

6.3 SUMMARY

Tables 6-2 and 6-3 list the control functions described in this chapter.

Table 6-2 Screen Display Sequences

Name	Mnemonic	Sequence
Send/receive mode	SRM	Set: CSI 1 2 h Local echo off. (D) Reset: CSI 1 2 l Local echo on.
Screen mode	DECSCNM	Set: CSI ? 5 h Light background. Reset: CSI ? 5 l Dark background. (D)
Scrolling mode	DECSCLM	Set: CSI ? 4 h Smooth scroll. (D) Reset: CSI ? 4 l Jump scroll.
Select status line type*	DECSSDT	CSI Ps $ ~ Ps = 0, none. Ps = 1, indicator. (D) Ps = 2, host-writable.
Select active status display*	DECSASD	CSI Ps $ } Ps = 0, main display. (D) Ps = 1, status line.
Sixel display mode	DECSDM	Set: CSI ? 8 0 h Disable sixel scroll. Reset: CSI ? 8 0 l Enable sixel scroll. (D)

* Available in VT300 mode only.

Table 6-2 Screen Display Sequences (cont)

Name	Mnemonic	Sequence
Control representation mode	CRM	Set: CSI 3 h Display controls. Reset: CSI 3 l Interpret controls. (D)
Kanji/Katakana	DECKKDM	Set: CSI ? 5 9 h Kanji terminal (D) Reset: CSI ? 5 9 l Katakana terminal

(D) = default.

Table 6-3 Screen Format Sequences

Name	Mnemonic	Sequence
Column mode	DECCOLM	Set: CSI ? 3 h 132 columns. Reset: CSI ? 3 l 80 columns. (D)
Set top and bottom margins	DECSTBM	CSI Pt ; Pb r Pt = top line. (D)=1 Pb = bottom line. (D)=24
Origin mode	DECOM	Set: CSI ? 6 h Move within margins. Reset: CSI ? 6 l Move outside margins. (D)

(D) = default.

CHAPTER 7

SETTING VISUAL CHARACTER AND LINE ATTRIBUTES

This chapter describes how to select visual attributes for display characters. Visual character attributes change the way characters appear on the screen, without changing the actual characters. For example, the bold character attributes for a complete display line on the screen.

7.1 SETTING VISUAL CHARACTER ATTRIBUTES

This section describes how to select and change visual character attributes. You can set the following attributes.

bold
underline
blinking
reverse video

7.2 SELECT GRAPHIC RENDITION (SGR)

This control function selects one or more character attributes at the same time.

Default: Clear all attributes (Ps = 0).

CSI	Ps	;	Ps	...	m
9/11	3/?	3/11	3/?	...	6/13

where

Ps is a number representing a certain visual attribute. You can use more than one Ps value to select different character attributes.
Table 7-1 lists Ps values and the attributes they select.

Table 7-1 Visual Character Attribute Values

Mode	Ps	Attribute
< VT100 or VT300 mode >		
	0	All attributes off
	1	Bold
	4	Underline
	5	Blinking
	7	Reverse video
< VT300 mode only >		
	22	Bold off
	24	Underline off
	25	Blinking off
	27	Reverse video off

(Examples)

- When you select more than one attribute in an SGR sequence, they are executed in order. For example, you can use the following sequence to display text that is bold, blinking, and underlined.

  CSI 0 ; 1 ; 5 ; 4 m

- The following sequence displays text in reverse video.

  CSI 7 m

Notes on SGR

- After you select an attribute, the terminal applies that attribute to all characters received. If you move characters by scrolling, the attributes move with the characters.

7.3 SETTING LINE ATTRIBUTES

Line attributes are display features that affect the way a line of characters appears on the screen. For example, the double-width, single-height line (DECDWL) attribute makes a line of characters appear twice as wide as a normal line of characters. This section describes how to select line attributes.

7.3.1 Single-Width, Single-Height Line (DECSWL)

This control function makes the line with the cursor single-width and single- height. This line attribute is the standard for all new lines on the screen.

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

7.3.2 Double-Width, Single-Height Line (DECDWL)

This control function makes the line with the cursor double-width and single- height. If the line was single-width and single-height, all characters to right of the screen's center are lost.

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

7.3.3 Double-Width, Double-Height Line (DECDHL)

These two control functions make the line with the cursor the top or bottom half of a double-height, double-width line. You must use these sequences in pairs on adjacent lines. In other words, the same display characters must appear in the same positions on both lines to form double-height characters. If the line was single-width and single-height, all characters to the right of the screen center are lost.

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

(Example)

The following sequences make the phrase "VT382 Video Terminal" double-height and double-width.

ESC#3 VT382 Video Terminal
ESC#4 VT382 Video Terminal

7.4 SUMMARY

Table 7-2 lists the control functions described in this chapter.

Table 7-2 Character and Line Attribute Sequences

Name	Mnemonic	Sequence
Select graphic rendition	SGR	CSI Ps...Ps m Ps = character attribute value(s). (Table 7-1)
Single-width, single-height line	DECSWL	ESC # 5
Double-width, single-height line	DECDWL	ESC # 6
Double-width, double-height line	DECDHL	ESC # 3 (top half) ESC # 4 (bottom half)

CHAPTER 8

EDITING

This chapter describes how to edit characters on the screen. You use editing control functions to insert, delete, and erase characters and lines of characters at the cursor position.

8.1 INSERTING AND DELETING TEXT

This section describes control functions that let you insert or delete data in the scrolling region. The scrolling region is the area of the screen inside the top and bottom margins (Chapter 6).

8.1.1 Insert/Replace Mode (IRM)

This control function selects how the terminal adds characters to the screen. The terminal always adds new characters at the cursor position.

Default: Replace.

Mode	Sequence			Action
Set (Insert)	CSI 9/11	4 3/4	h 6/8	Selects insert mode. New characters move characters on the screen to the right. Characters moved past the screen's right border are lost.
Reset (Replace)	CSI 9/11	4 3/4	l 6/12	Selects replace mode. New characters replace the characters replace the character at the cursor position.

8.1.2 Delete Line (DL)

This control function deletes one or more lines in the scrolling region, starting with the line that has the cursor.

CSI	Pn	M
9/11	3/?	4/13

where

Pn is the number of lines to delete.
Default: Pn = 1.

As lines are deleted, lines below the cursor and in the scrolling region move up. Blank lines without character attributes are added at the bottom of the scrolling region. If Pn is greater than the number of lines remaining in the scrolling region, DL deletes only the remaining lines.

8.1.3 Insert Line (IL)

This control function inserts one or more blank lines, starting at the cursor.

CSI	Pn	L
9/11	3/?	4/12

where

Pc is the number of lines to insert.
Default: Pn = 1

As lines are inserted, lines below the cursor and in the scrolling region move down. Lines scrolled past the margins are lost.

8.1.4 Delete Character (DCH)

This control function deletes one or more characters, from the cursor position to the right.

CSI	Pn	P
9/11	3/?	5/0

where

Pn is the number of characters to delete. If Pn is greater than the number of characters remaining on the line, DCH only deletes the remaining characters.
Default: Pn = 1.

As characters are deleted, characters to the right of the cursor move left. Character attributes move with the characters. The spaces created at the end of the line have all attributes off.

8.1.5 Insert Character (ICH)

This control function inserts one or more space (SP) characters, starting at the cursor position.

NOTE: Available in VT300 mode only.

CSI	Pn	@
9/11	3/?	4/0

where

Pn is the number of characters to insert.
Default: Pn = 1.

The ICH sequence inserts Pn blank characters with the normal character attribute. The cursor remains at the beginning of the blank characters. Text to the right of the cursor moves right. Characters scrolled past the margins are lost.

8.2 ERASING TEXT

This section describes control functions that let you erase from the display. These control functions can affect data inside or outside the scrolling region. They are not restricted by margins.

8.2.1 Erase In Display (ED)

This control function erase characters from part or all of the display. When you erase complete lines, they become single-height and single-width, with all character attributes cleared.

CSI	Ps	J
9/11	3/?	4/10

where

Ps represents the amount of the display to erase, as follows.

Ps	Area Erased
0 (default)	From the cursor through the end of the display
1	From the beginning of the display through the cursor
2	The complete display

PROGRAMMING TIP:
Use a Ps value of 2 to erase the complete display in a fast, efficient manner.

8.2.2 Erase In Line (EL)

This control function erases characters in the line that has the cursor. EL clears all characters attributes from erased character positions.

CSI	Ps	K
9/11	3/?	4/11

where

Ps represents the section of the line to erase, as follows.

Ps	Section Erased
0 (default)	From the cursor through the end of the line
1	From the beginning of the line through the cursor
2	The complete line

8.2.3 Erase Character (ECH)

This control function erases one or more characters, from the cursor position to the right. ECH clears character attributes from erased character positions.

NOTE: Available in VT300 mode only.

CSI	Pn	X
9/11	3/?	5/8

where

Pn is the number of characters to erase. A Pn value of 0 or 1 erases one character.
Default: Pn = 1.

8.3 SELECTIVELY ERASING TEXT

This section describes control functions that let you selectively erase data on the screen. You can only erase character that you define as erasable.

8.3.1 Select Character Protection Attribute (DECSCA)

This control function defines the characters that come after it as erasable or not erasable from the screen. The selective erase control functions (DECSED, DECSEL) cannot erase characters defined as not erasable.

NOTE: Available in VT300 mode only.

CSI	Ps	"	q
9/11	3/?	2/2	7/1

where

Ps defines all characters that follow the DECSCA function as erasable or not erasable.

Ps	Meaning
0 (default)	All attributes off
1	Not erasable by DECSED or DECSEL
2	Erasable by DECSED or DECSEL

Notes on DECSCA

- DECSCA does not affect visual character attributes set by the select graphic rendition (SGR) function.

8.3.2 Selective Erase In Display (DECSED)

This control function lets you erase some or all of the erasable characters in the display. DECSED can only erase characters defined as erasable by the DECSCA control function.

NOTE: Available in VT300 mode only.

CSI	?	Ps	J
9/11	3/15	3/?	4/10

where

Ps represents the area of the display to erase, as follows.

Ps	Area Erased
0 (default)	From the cursor through the end of the display
1	From the beginning of the display through the cursor
2	The complete display

Notes on DECSED

- DECSED does not affect visual character attributes set by the select graphic rendition (SGR) function.

8.3.3 Selective Erase In Line (DECSEL)

This control function lets you erase some or all of the erasable characters in a single line of text. DECSEL erases only those characters defined as erasable by the DECSCA control function.

NOTE: Available in VT300 mode only.

CSI	?	Ps	K
9/11	3/15	3/?	4/11

where

Ps represents the section of the line to erase, as follows,

Ps	Section Erased
0 (default)	From the cursor through the end of the line
1	From the beginning of the line through the cursor
2	The complete line

Notes on DECSEL

- DECSEL does not affect visual character attributes set by the select graphic rendition (SGR) function.

8.4 SUMMARY

Table 8-1 lists he control functions described in this chapter.

Table 8-1 Editing Sequences

Name	Mnemonic	Sequence
< Inserting and Deleting Text >		
Insert/replace mode	IRM	Set: CSI 4 h Insert characters. Reset: CSI 4 l Replace characters. (D)
Delete line	DL	CSI Pn M Pn lines.
Insert line	IL	CSI Pn L Pn lines.
Delete character	DCH	CSI Pn P Pn characters.
Insert character	ICH	CSI Pn @ Pn characters.

Table 8-1 Editing Sequences (cont)

Name	Mnemonic	Sequence
< Erasing Text >		
Erase in display	ED	CSI Ps J Ps = 0, cursor to end. (D) Ps = 1, beginning to cursor. Ps = 2, complete display.
Erase in line	EL	CSI Ps K Ps = 0, cursor to end. (D) Ps = 1, beginning to cursor. Ps = 2, complete line.
Erase character *	ECH	CSI Pn X Pn characters.
< Selectively Erasing Text >		
Select character attribute *	DECSCA	CSI Ps " q Ps = 0, all attributes off Ps = 1, not erasable. Ps = 2, erasable
Selective erase in display *	DECSED	CSI ? Ps J Ps = 0, cursor to end. (D) Ps = 1, beginning to cursor. Ps = 2, complete display.
Selective erase in line *	DECSEL	CSI ? Ps K Ps = 0, cursor to end. (D) Ps = 1, beginning to cursor. Ps = 2, complete line.

(D) = default.
* Available in VT300 mode only.

CHAPTER 9

CONTROLLING THE CURSOR

This chapter describes the control functions you use to move the cursor on the screen.

9.1 THE CURSOR

The cursor is a marker that indicates the active position on the screen. The active position is the point on the screen where the next character is written.

The default text cursor style is a blinking box. The cursor can also be a blinking underline, a no blinking box and a no blinking underline. You can select one of these cursor styles in the "Display Set-Up" screen. See Chapter 4 of "The VT382 User Guide" for details.

9.1.1 Cursor Enable Mode (DECTCEM)

This control function makes the cursor visible or invisible.

Default: Visible

Mode	Sequence					Action
Set (Visible)	CSI 9/11	? 3/15	2 3/2	5 3/5	h 6/8	Makes the cursor visible.
Reset (Invisible)	CSI 9/11	? 3/15	2 3/2	5 3/5	l 6/12	Makes the cursor invisible.

9.1.2 Moving The Cursor On The Screen (CUP, HVP, CUF, CUB, CUU, CUD)

This section describes the control functions you can use to move the cursor on the screen.

> **NOTE: Some C0 and C1 control characters not covered in this chapter also move the cursor. See Chapter 2 for detailed descriptions of these control characters.**

In the following sequences, the parameters Pn, Pl, and Pc indicate cursor position. If you omit a parameter or use a value of 0, the terminal uses a default value of 1.

Name	Sequence					Action
Cursor position (CUP)	CSI 9/11	Pl 3/?	; 3/11	Pc 3/?	H 4/8	Moves the cursor to line Pl, column Pc. The starting point for lines and columns depends on the setting of origin mode (DECOM). If Pl is 0 or 1, the cursor moves to the first line on the screen. If Pc is 0 or 1, the cursor moves to the first column on the screen.
Horizontal and vertical position (HVP)	CSI 9/11	Pl 3/?	; 3/11	Pc 3/?	f 6/6	Works the same as CUP. New applications should use CUP instead of HVP, to be compatible with future Digital products.
Cursor forward (CUF)	CSI 9/11	Pn 3/?	C 4/3			Moves the cursor right Pn columns. The cursor stops at the right margin. CUF clears the autowrap flag.

Name	Sequence			Action
Cursor backward (CUB)	CSI 9/11	Pn 3/?	D 4/4	Moves the cursor left Pn columns. The cursor stops at the left margin. CUB clears the autowrap flag.
Cursor up (CUU)	CSI 9/11	Pn 3/?	A 4/1	Moves the cursor up Pn lines in the same column. The cursor stops at the top margin. If the cursor stops at the top margin. If the cursor is already above the top margin, the cursor stops at the top line.
Cursor down (CUD)	CSI 9/11	Pn 3/?	B 4/2	Moves the cursor down Pn lines in the same column. The cursor stops at the bottom margin. If the cursor is already below the bottom margin, the cursor stops at the bottom line.

9.1.3 Saving And Restoring The Cursor State (DECSC, DECRC)

The save cursor function (DECSC) stores many of the terminal's selections and settings. The restore cursor function (DECRC) restores the terminal to the state saved by DECSC.

PROGRAMMING TIP:
Applications can use DECSC to save the current settings of many modes and control functions. Later, the application can use DECRC restore the control functions and modes to their saved state.

This operation is useful for applications that need to temporarily change the settings of some of the terminal's modes and control functions. When the application is finished, it can restore the modes and control functions to their previous state.

Name	Sequence	Action
Save cursor (DECSC)	ESC 7 1/11 3/7	Saves the following in the terminal's memory. . Cursor position . Character attributes set by the SGR command . Character sets (G0, G1, G2, or G3) currently in GL and GR . Wrap flag (autowrap or no autowrap) . State of origin mode (DECOM) . Selective erase attribute . Any single shift 2 (SS2) or single shift 3 (SS3) functions sent
Restore cursor (DECRC)	ESC 8 1/11 3/8	Restores the terminal to the state saved by the save cursor (DECSC) function. If nothing was saved by DECSC, then DECRC performs the following actions. . Moves the cursor to the home position (upper left of screen). . Resets origin mode (DECOM). . Turns all character attributes off (normal setting). . Maps the ASCII or JIS-Roman character set into GL, and the DEC Kanji (when Kanji terminal) or JIS-Katakana (when Katakana terminal) character set into GR.

Notes on DECSC and DECRC

- The terminal maintains a separate DECSC buffer for the main display and the status line. This feature lets you save a separate operating state for the main display and the status line.

9.2 SUMMARY

Table 9-1 lists the control functions described in this chapter.

Table 9-1 Cursor Movement Sequences

Name	Mnemonic	Sequence
< Enabling the Cursor >		
Text cursor enable mode	DECTCEM	Set: CSI ? 2 5 h Visible cursor. (D) Reset: CSI ? 2 5 l Invisible cursor.
< Moving the Cursor >*		
Cursor position	CUP	CSI Pl ; Pc H Line Pl, column Pc.
Horizontal and vertical position	HVP	CSI Pl ; Pc f Line Pl, column Pc.
Cursor forward	CUF	CSI Pn C Pn columns right.
Cursor backward	CUB	CSI Pn D Pn columns left.
Cursor up	CUU	CSI Pn A Pn lines up.
Cursor down	CUD	CSI Pn B Pn lines down.
< Saving and Restoring the Cursor State >		
Save cursor state	DECSC	ESC 7
Restore cursor state	DECRC	ESC 8

(D) = default.
* In these sequences, the default value for Pn, Pl, and Pc is 1.

CHAPTER 10

KEYBOARD AND PRINTING COMMANDS

10.1 KEYBOARD CONTROL FUNCTIONS

This section describes control functions that affect keyboard operation.

10.1.1 Keyboard Action Mode (KAM)

This control function locks or unlocks the keyboard.

Default: Unlocked

Mode	Sequence			Action
Set (Locked)	CSI 9/11	2 3/2	h 6/8	Locks the keyboard. The keyboard cannot send characters to the host. The "Wait indicator" comes on. The terminal ignores all keystrokes that send characters to the host.
Reset (Unlocked)	CSI 9/11	2 3/2	l 6/12	Unlocks the keyboard. The Keyboard can send characters to the host.

10.1.2 Line Feed/New Line Mode (LNM)

This control function selects the characters sent to the host when you press the "Return" key. LNM also controls how the terminal interprets line feed. (LF), form feed (FF), and vertical tab (VT) characters.

NOTE: For compatibility with Digital's software, you should keep LNM reset (line feed).

Default: Line feed

Mode	Sequence				Action
Set (New line)	CSI 9/11	2 3/2	0 3/0	h 6/8	When the terminal receives an LF, FF, or VT character, the cursor moves to the first column of the next line. When you press "Return", the terminal sends both a carriage return (CR) and line feed (LF).
Reset (Line feed)	CSI 9/11	2 3/2	0 3/0	l 6/12	When the terminal receives an LF,FF, or VT character. the cursor moves to the current column of the next line. When you press "Return", the terminal sends a CR only.

Notes on LNM

- When the numeric keypad is in keypad numeric mode (DECKPNM), the "Enter" key sends the same character(s) as the Return key.

10.1.3 Autorepeat Mode (DECARM)

This control function determines whether or not keys automatically repeat their character when held down, If DECARM is set, most keys you press for more than 0.5 seconds send a character repeatedly until you release the key.

Default: Repeat

Mode	Sequence				Action
Set (Repeat)	CSI 9/11	? 3/15	8 3/8	h 6/8	Keys autorepeat when pressed pressed for more than 0.5 seconds.
Reset (No repeat)	CSI 9/11	? 3/15	8 3/8	l 6/8	Keys do not autorepeat.

Notes on DECARM

- The following keys never repeat: "Hold screen", "Print Screen", "Set-Up","Data/Talk", "Break", "Return", "KANA", "Lock", "Shift", and "Ctrl".

10.1.4 Autowrap Mode (DECAWM)

This control function determines whether or not received characters automatically wrap to the next line when the cursor reaches the right border of the screen.

Default: No autowrap

Mode	Sequence				Action
Set (Autowrap)	CSI 9/11	? 3/15	7 3/7	h 6/8	Selects autowrap. Graphic characters received when the the cursor is at the screen's right border appear at the beginning of the next line. Any text on the screen scrolls up if the cursor is at the end of the scrolling region.
Reset (No autowrap)	CSI 9/11	? 3/15	7 3/7	l 6/12	Turns off autowrap. Graphic characters received when the cursor is at the screen's right border replace characters already on the screen.

10.1.5 Cursor Keys Mode (DECCKM)

This control function selects the sequences the arrow keys send. You can use the four arrow keys to move the cursor on the screen or to send special application commands. See Chapter 3 for the sequences the keys send.

Default: Cursor

Mode	Sequence				Action
Set (Application)	CSI 9/11	? 3/15	1 3/1	h 6/8	Arrow keys send application sequences to the host.
Reset (Cursor)	CSI 9/11	? 3/15	1 3/1	l 6/12	Arrow keys send ANSI cursor sequences to the host.

10.1.6 Numeric Keypad

The following control functions select whether the numeric keypad sends numeric characters or application sequences. See Chapter 3 for the sequences the key send.

10.1.6.1 Keypad Application And Numeric Modes (DECKPAM And DECKPNM)

Default: Numeric characters

Mode	Sequence	Action
Application (DECKPAM)	ESC = 1/11 3/13	Numeric keypad sends application sequences.
Numeric (DECKPNM)	ESC > 1/11 3/14	Numeric keypad sends the characters shown on the key (number, comma, period, or minus sign). Keys "PF1" through "PF4" send application sequences.

Notes on DECKPAM and DECKPNM

- When you turn on or reset the terminal, the terminal automatically selects numeric keypad mode.

10.1.6.2 Numeric Keypad Mode (DECNKM)

This control function works like the DECKPAM and DECKPNM functions above. DECNKM is provided mainly for use with the request and report mode (DECRQM/ DECRPM) control functions (Chapter 11).

Default: Numeric characters

Mode	Sequence	Action
Set (Application)	CSI ? 6 6 h 9/11 3/15 3/6 3/6 6/8	Numeric keypad sends application sequences.
Reset (Numeric)	CSI ? 6 6 l 9/11 3/15 3/6 3/6 6/12	Numeric keypad sends characters shown on the key (number, comma, period, or minus sign). Keys "PF1" through "PF4" send control functions.

10.1.7 Backarrow Key Mode (DECBKM)

This control function selects the characters sent to host when you press the ⌫ key.

Default: Delete

Mode	Sequence					Action
Set (Backspace)	CSI 9/11	? 3/15	6 3/6	7 3/7	h 6/8	⌫ key sends BS code.
Reset (Delete)	CSI 9/11	? 3/15	6 3/6	7 3/7	l 6/12	⌫ key sends DEL code.

10.1.8 Katakana Shift Mode (DECKANAM)

This mode determines if the terminal keyboard generates JIS-Roman (or ASCII) or JIS-Katakana codes.

Default: ASCII or JIS-Roman

Mode	Sequence					Action
Set (JIS-Katakana)	CSI 9/11	? 3/15	1 3/1	2 3/2	h 6/8	The keyboard generate JIS-Katakana codes without pressing "KANA" key. The "KANA LED" will be turned on.
Reset (ASCII or JIS-Roman)	CSI 9/11	? 3/15	1 3/1	2 3/2	l 6/12	The keyboard generate ASCII or JIS-Roman codes and the "KANA LED" will be turned off.

10.2 USER-DEFINED KEYS (DECUDK)

The keyboard has 20 function keys on its top row. You can define the codes sent by 15 of these top-row keys.

- F6 through F14
- Help
- Do
- F17 through F20

The other five keys - "Hold Screen", "Print Screen", "Set-Up", "Data/Talk", and "Break" - have dedicated local functions that you cannot change.

10.2.1 Using UDKs

User-defined keys (UDKs) are only available in VT300 mode. UDKs do not work in VT100 and VT52 modes.

You define the function of keys by using a DECUDK device control string, as described in "Programming UDKs" section that follows. After you define a key, you can use the new function by pressing "Shift-(function key)" or the function key alone, which is defined in "User-Defined Keys Set-Up" screen, while the normal control sequence values are accessed by pressing the function key alone or "Shift-(function key)".

10.2.2 UDK Memory Space

There are 256 bytes of memory available for the 15 user-defined keys. Space is supplied on a first-come/first-serve basis. When the 256 bytes are full, you cannot define any more keys until you clear some of the memory space. There are five ways you can clear space.

- Redefine one or more UDKs by using the "UDK Set-Up" screen.
- Redefine one or more UDKs by using a DECUDK control string.
- Clear one or more UDKs by using a DECUDK control string.
- Clear one or more UDKs by using the "UDK Set-Up" screen.
- Clear all UDKs with a terminal power-up or reset (RIS) operation.

NOTE: All key definitions can be stored in non-volatile RAM. Loss of terminal power will not result in loss of UDK key definitions.

10.2.3 Programming UDKs

You use the following device control string format to down-line-load definitions for user-defined keys. See Chapter 2 for general information about device control strings.

DECUDK Device Control String Format

NOTE: Available in VT300 mode only.

DCS	Pc ; Pl	|	Ky1/St1;...Kyn/Stn	ST
Device Control String Introducer	Clear and Lock Parameters	Final Character	Key Definition String	String Terminator

where

DCS (9/0)
indicates the beginning of a device control string. DCS is an 8-bit C1 character. You can use ESC P (1/11 5/0) for a 7-bit environment.

Pc
is the clear parameter. Pc selects how to clear key definitions.

Pc	Action
0 (default) or none	Clear all keys before loading new values.
1	Clear one key at a time, before loading a new value.

When Pc is 1, the terminal only clears the keys you are loading. By using a Pc value of 1, you can redefine some keys without redefining them all.

NOTE: There are 256 bytes of memory for all user-defined keys. A key definition can only use the number of bytes available when that key is loaded.

PROGRAMMING TIP:
If Pc is 1, a key load may fail because no memory space is available. The reason for this is as follows.

With Pc set to 1, keys are cleared and loaded sequentially. If the new definition for a key is larger than the old one, you may exceed the 256 byte limit.

For example, suppose "F6" contains 120 bytes, "F7" contains 110 bytes, and "F8" contains 20 bytes. You try to load "F8" with 40 bytes, "F6" with 1 byte, and "F7" with 1 byte, in that order. This works if all keys are cleared first (Pc is 0), but not if keys are cleared one at a time (Pc is 1). When you try to load "F8" with 40 bytes, the load fails because only 26 bytes are free at that time.

256 (maximum) - 120 (in "F6") - 110 (in "F7") = 26

Pl

is the lock parameter. Pl determines whether the key definitions are locked or unlocked after you load them.

Pl	Action
0 or none	Lock the keys. If you want to load new values into the keys, you must unlock the keys by using Set-Up.
1 (default)	Do not lock the keys. The keys are unlocked and can be redefined with another DECUDK string.

NOTE: If Pl is 1 and the keys are already locked, nothing happens.

The terminal uses a special lock to allow or prevent the programming of user-defined keys. You can turn on this lock from Set-Up or from the host (with a DECUDK device control string). The lock affects all programmable keys. When you use to lock, you should following these guidelines.

- Unlock the keys to define them.
  The keys must be unlocked before you can define them. You can only unlock the keys from Set-Up. If a key is locked and an application tries to redefine the key with a DECUDK sequence, the terminal ignores the sequence.

- Lock the keys to prevent redefinition.
  You can lock the keys from Set-Up or from the host (by sending a DECUDK sequence). New key definitions are locked by default.

|

is the final character. The vertical bar (7/12) identifies this control string as a DECUDK.

Ky1/St1;...Kyn/Stn

are the key definition strings. You include these strings between the final character (|) and the string terminator (ST). Each string consists of a key selector number (Kyn) and a string parameter (Stn), separated by a slash (/ 2/15). A semicolon (; 3/11) separates different strings.

. The key selector number (Kyn) indicates which key you are defining. Here is a list of definable keys and their identifying values.

Key	Value	Key	Value	Key	Value
F6	17	F11	23	Do	29
F7	18	F12	24	F17	31
F8	19	F13	25	F18	32
F9	20	F14	26	F19	33
F10	21	Help	28	F20	34

. The string parameters (Stn) are the encoded definition of the keys. String parameters consist of hex pairs in the following ranges.

3/0 through 3/9 (0 through 9)
4/1 through 4/6 (A through F)
6/1 through 6/6 (a through f)
When you combine these hex values, they represent an 8-bit quantity. The ASCII table in Chapter 2 lists the hex values of characters.

This method lets you use any of the 256 character codes in the key string. You can enter key definition strings in any order.

Default: Empty. The key is undefined.

ST

is the string terminator. ST (9/12) is a C1 8-bit character. You can use ESC \ (1/11, 5/12) for a 7-bit environment.

Notes On Loading UDKs
Here are some general guidelines you should keep in mind when loading UDKs.

- Clear UDK memory space before loading new definitions. Make sure the previous UDK definitions are cleared before you load new ones.

- If you redefine a key, the old definition is lost.
  This may clear some space if the new definition uses less bytes than the old one.

- There are two ways to lock UDKs, but only one way to unlock them. To lock UDKs, you can use the "UDK Set-Up" screen or a DECUDK control string. To unlock UDKs, you must use the "UDK Set-Up" screen.

- The default value for each key definition is empty. When you clear UDKs, they are empty.

- An invalid hex pair in a DECUDK string stops a UDK load sequence. When a load sequence stops (due to error or other cause), the terminal saves any keys already loaded and cancels the rest of the sequence from the point where the error occurred.

Examples of DECUDK Device Control Strings

- The following sequence clears UDKs.

  DCS 0 ; 1 | ST

- The following sequence locks UDKs.

  DCS 1 ; 0 | ST

- Suppose you want to define the F20 key to be "PRINT", without clearing or locking any other keys. The first par of your sequence would look like this.

  DCS 1 ; 1 | 3 4 /

  Where 34 is the code for the F20 key.

After the slash character (2/15), you include the definition. The rest of the sequence after the slash character would look like this.

5 0 5 2 4 9 4 E 5 4 ST

where the hex encoding for "PRINT" is as follows.

50 = P
52 = R
49 = I
4E = N
54 = T

The ST character (9/12) marks the end of the control string. The complete string is as follows.

DCS 1; 1 | 34 / 50 52 49 4E 54 ST

10.3 PRINTER PORT CONTROL FUNCTIONS

10.3.1 Printer Extent Mode (DECPEX)

This control function selects what area of the screen display to print when you use the print screen function. See the "Print Screen" description in the "Printing Functions" section that follows.

Default: Screen

Mode	Sequence					Action
Set (Screen)	CSI 9/11	? 3/15	1 3/1	9 3/9	h 6/8	The print function prints the complete screen.
Reset (Scrolling region)	CSI 9/11	? 3/15	1 3/1	9 3/9	l 6/12	The print function only prints the scrolling region (data inside the margins).

10.3.2 Print Form Feed Mode (DECPFF)

This control function selects whether or not the terminal sends a form feed (FF) character to the printer at the end of a printing function.

Default: No form feed

Mode	Sequence					Action
Set (Form feed)	CSI 9/11	? 3/15	1 3/1	8 3/8	h 6/8	The terminal sends a form feed (FF) to the printer at the end of a printing function.
Reset (No form feed)	CSI 9/11	? 3/15	1 3/1	8 3/8	l 6/12	The terminal sends nothing to the printer at the end of a printing function.

Notes on DECPFF

- DECPFF does not affect the print cursor line function described in the next section.

10.4 PRINTING FUNCTIONS

This section describes control functions you use to print text from the terminal. If you do not have a printer connected to the terminal, the terminal ignores these functions.

When you print characters from the screen, the printer converts all tabs to spaces. Printed characters are spaced with the space (SP) character. The terminal sends a carriage return (CR), line feed (LF), vertical tab (VT), or form feed (FF) character to the printer after the last printed character on a line.

All the printing functions described in this section are variations of the media copy (MC) command. The VT382 can use one of two variations of the MC command, standard and DEC private. The format of each is as follows.

MC	ANSI standard	CSI	Ps	i	
		9/11	3/?	6/9	
MC	DEC private	CSI	?	Ps	i
		9/11	3/15	3/?	6/9

where Ps indicates the function of the command

10.4.1 Printing A Line At A Time: Autoprint Mode

In this mode, the printer prints a line from the screen when you move the cursor off that line an LF, FF, or VT character, or when an autowrap occurs. The printed line ends with a CR and the character (LF, FF, or VT) that moved the cursor off the previous line.

Sequence	Action
CSI ? 5 i	Turns on autoprint mode.
CSI ? 4 i	Turns off autoprint mode.

10.4.2 Sending Characters Directly To The Printer : Printer Controller Mode

This mode lets the host control the operation of the printer. The terminal sends all characters and control sequences directly to the printer, except NUL, XON, XOFF, and the printer controller mode sequences.

Sequence	Action
CSI 5 i	Turns on printer controller mode.
CSI 4 i	Turns off printer controller mode.

Notes on Printer Controller Mode

- Printer controller mode cancels autoprint mode. When the terminal leaves printer controller mode, the terminal returns to the normal method for printing operations.
- The printer's active column position should always be on the left margin before the terminal leaves printer controller mode.

10.4.3 Print Screen

This control function prints the data on the screen. The terminal stores data entered from the keyboard until printing is complete. You can use either of the following sequences to print the screen.

CSI i or CSI 0 i

Notes on Print Screen

- If printer extent mode (DECPEX) is currently reset, the print screen function only prints the scrolling region.

10.4.4 Print Cursor Line

This control function prints the line that has the cursor. The cursor does not move.

CSI ? 1 i

10.4.5 Stop/Start Relay From Printer Port To Host

This determines whether or not to pass data coming in on the printer port to the host port for transmission to the host.

Default: Stop relay from printer port to host

Mode	Sequence				Action
Stop	CSI 9/11	? 3/15	8 3/8	i 6/9	Stops transmission of data coming in on the printer port to the host.
Start	CSI 9/11	? 3/15	9 3/9	i 6/9	Starts transmission of data coming in on the printer port to the host.

10.5 PRINTING VISUAL ATTRIBUTES

This section describes how the terminal sends visual attributes, such as bold or underlining, to a local printer. To send visual attributes, you must select "All Characters" in the "Printer Set-Up" screen. See Chapter 4 of "The VT382 User Guide" for details.

The VT382 can send two types of visual attributes, line attributes and visual character attributes.

10.5.1 Sending Line Attributes

The terminal sends line attributes to a printer by (1) sending the appropriate line attribute control function, followed by (2) the characters in the current line. There are four line attribute control functions.

Single-width line	ESC # 5
Double-width line	ESC # 6
Double-width/double-height line	
Top half	ESC # 3
Bottom half	ESC # 4

10.5.2 Sending Visual Character Attributes

The terminal initializes character attributes at the beginning of each print line or print screen operation by sending the following SGR sequence to the printer (Chapter 7).

ESC [0 m

This sequence sets all character attributes to the normal rendition.

To send a visual character attribute to a printer, the VT382 (1) sends the appropriate SGR sequence for that attribute, followed by (2) the current character. The SGR sequence is as follows.

ESC [0; Ps; Ps;...Ps m

where

Ps indicates a character attribute sent.

Ps	Attribute
0	Normal (all attributes off)
1	Bold
4	Underline
5	Blinking
7	Reverse video

After each print line or print screen operation, the terminal clears all attributes by sending the following sequence.

ESC [0 m

10.6 SUMMARY

Tables 10-1 through 10-3 list the control sequences described in this chapter.

Table 10-1 Keyboard Control Sequences

Mode	Mnemonic	Sequence Set	Reset
Keyboard action mode	KAM	CSI 2 h Locked.	CSI 2 l Unlocked. (D)
Line feed/ new line mode	LNM	CSI 2 0 h New line.	CSI 2 0 l Line feed. (D)
Autorepeat mode	DECARM	CSI ? 8 h Repeat. (D)	CSI ? 8 l No repeat.
Autowrap mode	DECAWM	CSI ? 7 h Autowrap.	CSI ? 7 l No autowrap.(D)
Cursor keys mode	DECCKM	CSI ? 1 h Application.	CSI ? 1 l Cursor. (D)
Keypad application/ numeric modes	DECKPAM DECKPNM	ESC = Application.	ESC > Numeric. (D)
Numeric keypad mode	DECNKM	CSI ? 6 6 h Application.	CSI ? 6 6 l Numeric. (D)

(D) = default.

Table 10-2 Programming UDKs

Definable Keys	
F6 through F14	Help
Do	F17 through F20

< DECUDK Device Control String Format >

DCS Pc ; Pl | Ky1/St1;...Kyn/Stn ST

Pc is the clear parameter.

0 or none	=	Clear all keys before loading new values (default)
1	=	Clear one key at a time, before loading a new value.

Pl is the lock parameter.

0 or none	=	Lock the keys.
1	=	Do not lock the keys. (default)

Ky1/St1;...Kyn/Stn are the key definition strings.

The key selector number (Kyn) indicates which key you are defining.

Key	Value	Key	Value	Key	Value
F6	17	F11	23	Do	29
F7	18	F12	24	F17	31
F8	19	F13	25	F18	32
F9	20	F14	26	F19	33
F10	21	Help	28	F20	34

The string parameters (Stn) are the key definitions, encoded as pairs of hex codes.

3/0 through 3/9 (0 through 9)
4/1 through 4/6 (A through F)
6/1 through 6/6 (a through f)

Table 10-3 Printing Control Sequences

Name	Mnemonic	Sequence
Printer extent mode	DECPEX	Set: CSI ? 1 9 h Screen. (D) Reset: CSI ? 1 9 l Scrolling region.
Print form feed mode	DECPFF	Set: CSI ? 1 8 h Form feed. Reset: CSI ? 1 8 l No form feed. (D)
Autoprint mode	MC	On: CSI ? 5 i Off: CSI ? 4 i
Printer controller mode	MC	On: CSI 5 i Off: CSI 4 i
Print screen	MC	CSI i or CSI 0 i
Print cursor line	MC	CSI ? 1 i
Stop relay from printer port to host		CSI ? 8 i (D)
Start relay from printer port to host		CSI ? 9 i

(D) = default.

CHAPTER 11

VT382 REPORTS

The VT382 sends reports in response to requests from the host computer. These reports provide the host with the following kinds of information about the terminal.

identification (type of terminal)
cursor state
operating status
operating level (VT100 or VT300)
almost all terminal states that software can set

The host can use the reports to adjust the computing environment to match the terminal.

11.1 DEVICE ATTRIBUTES (DA)

The terminal and host computer exchange DA sequences to provide the host with the following information.

conformance level and extensions
basic features
identification code
firmware version level
hardware options

Based on this information, the host can

- use the information it receives to make the best use of the terminal's features.

. select the correct application software for the terminal.

. determine the cause of certain communication errors.

There are two types of DA exchanges between the host and the terminal, primary DA and secondary DA. The host can request a primary DA or secondary DA report, depending on the information the host needs.

NOTE:If Printer Controller mode is set, the DA sequence will be sent to printer which may or may not respond.

11.1.1 Primary DA

In this DA exchange, the host asks for the terminal's service class code and basic attributes.

< Host Request >

The host uses the following sequence to send this request.

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

< Terminal Response >

The terminal responds by sending its service code and basic attributes to the host. This response depends on the setting of the Terminal ID feature in the "Terminal Set-Up" screen. See Chapter 4 of "The VT382 User Guide" for details.

CSI	?	Psc	;	Ps1	;	...	Psn	c
9/11	3/15	3/?	3/11	3/?	3/11	...	3/?	6/3

where

Psc indicates the terminal's service class code. The value of Psc depends on the terminal's current operating level, as follows.

Psc	Operating Level
61	Level 1 (VT100 family)
62, 63	Level 3 (VT200 or VT300 family)

Ps1...Psn indicate which of the following extensions the terminal supports.

Ps	Meaning
1	132 columns
2	Printer port
4	Sixel I/O
5	Katakana
6	Selective erase
7	Soft character set (DRCS)
8	User-defined keys
10	Two-byte Kanji
15	Technical Character Set

< Primary DA Example >

Here is a typical primary DA exchange.

Exchange	Sequence	Meaning
Request (Host to VT382)	CSI c or CSI 0 c	The host asks for the terminal's service code, conformance level, and supported extensions.
Response (VT382 to host: international model)	CSI ? 63; 1; 2; 4; 5; 6; 7; 8 ; 10; 15 c	The terminal is a service class 3 device (63) and supports the following extensions. . 132 columns (1) . Printer port (2) . Sixel I/O (4) . Katakana (5) . Selective erase (6) . DRCS (7) . UDKs (8) . Two-bytes Kanji (10) . Technical character set (15)

Table 11-1 lists the primary DA alias responses the VT382 can send to the host. The terminal uses an alias response to identify itself to the host as some other type of terminal. Each response corresponds to one of the "Terminal ID" settings selected in the "Terminal Set-Up" screen.

Table 11-1 Alias Primary DA Responses from the VT382

Terminal	Identification Sequence	Meaning
VT80 DA	ESC [? 18 ; 2 c	VT80 terminal
VT100 DA	ESC [? 1 ; 2 c	VT100 terminal
VT100J DA	ESC [? 5 ; 2 c	VT100J terminal
VT101 DA	ESC [? 1 ; 0 c	VT101 terminal
VT102 DA	ESC [? 6 c	VT102 terminal
VT102J DA	ESC [? 15 c	VT102J terminal
VT220J DA	CSI ? 62; 1; 2; 5; 6; 7; 8 c	VT220J terminal
VT282 DA	CSI ? 62; 1; 2; 4; 5; 6; 7; 8; 10; 11 c	VT220J terminal
VT320 DA	CSI ? 63; 1; 2; 6; 7; 8 c	VT320 terminal
VT382 DA	CSI ? 63; 1; 2; 4; 5; 6; 7; 8; 10; 15 c	VT382 terminal

NOTE:To change an alias response, you must use the 'Terminal Set-Up" screen.
See Chapter 4 of "The VT382 User Guide".

11.1.2 Secondary DA

In this DA exchange, the host requests the terminal's identification code, firmware version level, and hardware options.

< Host Request >
The host uses the following sequence to send this request.

CSI	>	c	or	CSI	>	0	c
9/11	3/14	6/3		9/11	3/14	3/0	6/3

< Terminal Response >
The terminal uses the following sequence to respond.

CSI	>	Pp	;	Pv	;	Po	c
9/11	3/14	3/?	3/11	3/?	3/11	3/?	6/3

where

Pp indicates the identification code for the terminal. For the VT382, the identification code is 32.

Pv indicates the firmware version level of the terminal. Firmware is the software implementation of all the terminal's functions (for example, the editing functions).

(Examples)

Pv	Version
7	X0.7 (prereleased version 0.7)
10	V1.0 (released version 1.0)

Po indicates the hardware options installed in the terminal.
Po = 1; DEC Kanji-1978 is currently selected by Set-Up
= 2; DEC Kanji-1983 is currently selected by Set-Up

< Secondary DA example >

Here is a typical secondary DA exchange.

Exchange	Sequence	Meaning
Request (Host to VT382)	CSI > c or CSI > 0 c	The host asks for the terminal's identification, firmware version, current hardware options.
Response (VT382 to host)	CSI > 32; 14; 2 c	The terminal identifies itself as a VT382, uses version 1.4 firmware, and DEC Kanji-1983 version.

11.2 TERMINAL IDENTIFICATION (DECID)

This control function is similar to a primary device attributes (DA) request from the host. See the previous " Device attributes" section.

NOTE: Digital does not recommend using DECID. DECID may not be supported in Digital terminals. You should use the primary device attributes request for this purpose. In VT300 mode, the terminal ignores DECID.

< Host DECID Request >

ESC	Z
1/11	5/10

< Terminal Response >

The terminal uses the same response as for a primary DA request. The terminal uses this response for all operating levels (1 or 3).

11.3 DEVICE STATUS REPORTS (DSR)

The host computer and terminal exchange DSR sequences to provide the host with the operating status of the following features.

VT382 operating status	user-defined keys
cursor position	printer port
keyboard dialect	

DSR requests and reports follow one of two formats, ANSI standard or DEC private. The format for each is as follows.

ANSI standard	CSI	Ps	n	
	9/11	3/?	6/14	
DEC private	CSI	?	Ps	n
	9/11	3/15	3/?	6/14

where

Ps indicates the type of DSR requested.

There is a different DSR request for each feature. The following sections describe the possible DSR reports. If the terminal is in printer controller mode (Chapter 10), the printer receives the DSR request.

11.3.1 DSR - VT382 Operating Status

The host requests the terminal's operating status.

Exchange	Sequence	Meaning
Request (Host to VT382)	CSI 5 n	The host requests the terminal's operating status. The host asks if the terminal is in good operating condition.
Responses (VT382 to host)	CSI 0 n	The terminal indicates that it is in good operating condition.
	CSI 3 n	The terminal indicates that it has a malfunction.

11.3.2 DSR - Cursor Position Report (CPR)

The host asks the terminal for a cursor position report.

Exchange	Sequence	Meaning
Request (Host to VT382)	CSI 6 n	The host asks for a cursor position report (CPR).
CPR response (VT382 to host)	CSI Pl ; Pc R	The terminal indicates that the cursor is currently at line Pl, column Pc.

11.3.3 DSR - Printer Port

The host asks for the status of the terminal's printer.

NOTE: Host software should check the printer status before entering any print mode or starting any printing function.

Exchange	Sequence	Meaning
Request (Host to VT382)	CSI ? 1 5 n	The host asks for the current printer status.
Possible responses (VT382 to host)	CSI ? 1 3 n	No printer. The data set ready (DSR) signal has not been asserted on the printer port since the last power-up or reset.
	CSI ? 1 0 n	Printer ready. DSR is asserted on the printer port.
	CSI ? 1 1 n	Printer not ready. DSR is not currently asserted on the printer port.

11.3.4 DSR - User-Defined Keys

The host asks if the user-defined keys (UDKs) are locked or unlocked.

NOTE: Available in VT300 mode only.

Exchange	Sequence	Meaning
Request (Host to VT382)	CSI ? 2 5 n	The host asks if UDKs are locked or unlocked.
Possible responses (VT382 to host)	CSI ? 2 0 n	UDKs are unlocked.
	CSI ? 2 1 n	UDKs are locked.

11.3.5 DSR - Keyboard Dialect

The host asks for the current keyboard dialect.

NOTE: Available in VT330 mode only.

Exchange	Sequence	Meaning
Request (Host to VT382)	CSI ? 2 6 n	The host asks for the keyboard dialect.
Response (VT382 to host)	CSI ? 2 7; Pd n	The keyboard dialect is Pd. where Pd = Dialect 17 = Katakana

11.4 TERMINAL STATE REPORTS

NOTE: Available in VT330 mode only.

The host can request a report on the terminal's complete operating state. In response, the terminal sends the host a terminal state report. This report includes the current setting of all the terminal's features, except user-defined key (UDK) definitions and the soft character set. The host can use the information in the report to save the current terminal state. Later, the host can restore the terminal to the saved state.

This operation is useful for applications that need to temporarily change the terminal's operating state. When the application is finished, it can restore the terminal to the previous operating state.

A terminal state report is a device control string. The terminal sends the report in response to a request terminal state report (DECRQTSR) sequence from the host.

11.4.1 Request Terminal State Report (DECRQTSR) - Host To VT382

NOTE: Available in VT300 mode only.

The host sends this control function to request a terminal state report (DECTSR). The terminal responds by sending a report indicating the settings of all the terminal's features, except UDK definitions and the soft character set.

CSI	Ps	$	u
9/11	3/?	2/4	7/5

where

Ps must be 1 for the terminal to return a terminal state report.

Ps	Report Requested
0 or none	Ignored. No report sent.
1	Terminal state report (DECTSR)

11.4.2 Terminal State Report (DECTSR) - VT382 To Host

NOTE: Available in VT300 mode only.

The terminal sends this sequence in response to a request terminal state report (DECRQTSR) sequence, DECTSR provides the host with a complete report on the terminal's current operating state, except UDK definitions and the soft character set.

PROGRAMMING TIP:
Applications can use the information in the terminal state report to save the current terminal state. Later, the application can restore the terminal to the saved state.

This operation is useful for application that need to temporarily change the terminal operating state. When the application is finished, it can restore the terminal to the previous operating state. You use the restore terminal state (DECRSTS) function to restore the terminal state. DECRSTS is described in the next section.

The DECTSR format is as follows.

DCS	1	$	s	D1...Dnn	< checksum 1 >	< checksum 2 >	ST
9/0	3/1	2/4	7/3	....			9/12

where

D1...Dnn is a data string indicating the status of many of the terminal's functions. There are nn bytes in the data string (D1...Dnn). D1...Dnn are in the range 4/0 to 4/15 in the code table (Chapter 2). Bit 6 of each Dn byte is always on, and bits 4,5,and 7 are always off.

< checksum 1 > < checksum 2 > is a 2 byte checksum of all data (D1...Dnn) in the report. The checksum is equal to the 2's complement of the sum of all the data elements in the report (D1 + D2 +...Dn).

Notes on DECTSR

- Software should not expect the format of DECTSR to be the same for all VT300 terminals.

11.4.3 Restore Terminal State (DECRSTS)

NOTE: Available in VT300 mode only.

This sequence restores the terminal to a previous state specified in the terminal state report (DECTSR).

PROGRAMMING TIP:
Applications can use DECRSTS to restore the terminal to a previous operating state specified in a terminal state report. See the previous "Terminal State Report (DECTSR)" section in this chapter.

DCS	Ps	$	p	D...D	ST
9/0	3/?	2/4	7/0	...	9/12

where

Ps indicates whether or not the host succeeds in restoring the terminal state. Ps must be 1 for a successful restore operation.

Ps	Data string Format
0	Error, restore ignored.
1	Restore the previous terminal state, based on the terminal state report(DECTSR).

D...D is a data string that contains the restored information. This string is identical to the data string used by the terminal state report.

Notes on DECRSTS

- If there is an invalid value in the DECRSTS sequence, the terminal ignores the rest of the sequence. This action may leave the terminal in a partially restored state.
- Software should not expect the format of the terminal state report (DECTSR) to be the same for all VT300 terminals.

11.5 PRESENTATION STATE REPORTS

NOTE: Available in VT300 mode only.

The terminal can send two presentation state reports.

State Report	Function
Cursor information report (DECCIR)	Reports on the cursor position, including its visual attributes and character protection attributes. Also reports on origin mode (DECOM) and the current active active character sets.
Tab stop report (DECTABSR)	Reports the current tab stop settings.

The host can request the terminal's current presentation state. In response to this request, the terminal returns a presentation state report. The host can use the information in the report to save the current presentation state. Later, the host can restore the terminal to the saved state.

This operation is useful for applications that need to temporarily change the terminal's presentation state. When the application is finished, it can restore the terminal to the previous presentation state.

A presentation state report is a device control string. The terminal sends the report in response to a request presentation state report (DECRQPSR) sequence from the host.

11.5.1 Request Presentation State Report (DECRQPSR)- Host To VT382

NOTE: Available in VT300 mode only.

The host sends this sequence to request a cursor information report (DECCIR) or a tabulation stop report (DECTABSR).

CSI	Ps	$	w
9/11	3/?	2/4	7/7

where

Ps indicates which report the host requests.

Ps	Report Requested
0	Error, request ignored
1	Cursor information report (DECCIR)
2	Tab stop report (DECTABSR)

11.5.2 Cursor Information Report (DECCIR) - VT382 To Host

NOTE: Available in VT300 mode only.

The terminal sends this sequence in response to a request presentation state report (DECRQPSR) sequence. DECCIR reports the status of the cursor position, including visual attributes and character protection attributes. DECCIR also reports the status of origin mode (DECOM) and the current active character sets.

PROGRAMMING TIP:
Applications can use the information in the cursor information report to save the current presentation state. Later, the application can restore the terminal to the saved state.

This operation is useful for applications that need to temporarily change the terminal's presentation state. When the application is finished, it can restore the terminal to the previous presentation state. You use the restore presentation state (DECRSPS) function to restore the presentation state. DECRSPS is described later in this chapter.

DCS	1	$	u	D...D	ST
9/0	3/1	2/4	7/5	...	9/12

where

D...D is the data string containing the cursor information. The format for this data string as follows.

Pr; Pc; Pp; Srend; Satt; Sflag; Pgl; Pgr; Scss; Sdesig

where

Pr is the number of the line the cursor is on.

Pc is number of the column the cursor is at.
Pp is the number of the current page. For the VT382, Pp is always 1.

Srend is one or more characters indicating the visual attributes (such as bold and blinking) currently in use for writing (Chapter 7).

To find out what attributes are set, you must convert the character to an 8-bit binary number, You can use the code table in Chapter 2 to convert characters. After you convert a character, you can find the meaning of its 8-digit binary number in the following table. The table lists the most significant bit (8) to the least significant bit (1).

Bit	Attribute	Bit Value
8	-	Always 0 (off).
7	-	Always 1 (on).
6	-	Always 0 (off)
5	-	Always 0 (off).
4	Reverse video	0 = off. 1 = on.
3	Blinking	0 = off. 1 = on.
2	Underline	0 = off. 1 = on.
1	Bold	0 = off. 1 = on.

(Example)

If the bold and underline attributes are currently set for writing, Srend is the ASCII uppercase C character (binary 01000011).

Satt is one or more characters indicating any selective erase attributes (Chapter 8) currently set for writing.

To find what attributes are set, you must convert each character to an 8-bit binary number. Use the same method you used to convert the Srend parameter above. Then use the following table to find the meaning of the 8-bit binary number.

Bit	Attribute	Bit Value
8	-	Always 0 (off).
7	-	Always 1 (on).
6	-	Always 0 (off).
5	-	0 - Reserved for future use.
4	-	0 - Reserved for future use.
3	-	0 - Reserved for future use.
2	-	0 - Reserved for future use.
1	Selective erase (DECSCA)	0 = off. 1 = on.

(Example)

If the selective erase is currently on for writing, Satt is the ASCII uppercase A character (binary 01000001).

Sflag is one or more characters that indicate several flags and modes the terminal must save.

To see the current state of the flags and modes, you must convert each character to an 8-bit binary number. Use the same method you used to convert the Srend and Satt parameters above. Then use the following table to find the meaning of the 8-bit binary number.

Bit	Attribute	Bit Value
8	-	Always 0 (off).
7	-	Always 1 (on).
6	-	Always 0 (off).
5	-	0 - Reserved for future use.
4	Autowrap	1 = autowrap pending 0 = autowrap not pending
3	Single shift 3 (SS3) setting	1 = G3 is mapped into GL for the next typed character only. 0 = single shift 3 is off.
2	Single shift 2 (SS2) setting	1 = G2 is mapped into GL for the next typed character only. 0 = single shift 2 is off.
1	Origin mode	1 = origin mode set 0 = origin mode reset

(Example)

If origin mode is set, autowrap is pending, and a single shift 3 has been received, then Sflag is the ASCII upper case M character (binary 01001101).

Pgl indicates the number of the logical character set (G0 through G3) mapped into GL.

0 = G0 is in GL.	2 = G2 is in GL.
1 = G1 is in GL.	3 = G3 is in GL.

Pgr indicates the number of the logical character set (G0 through G3) mapped into GR.

0 = G0 is in GR.	2 = G2 is in GR.
1 = G1 is in GR.	3 = G3 is in GR.

Scss is a character indicating the size of the character sets in G0 through G3.

To find out what the character means, you must convert it to an 8-bit binary number. Use the same method you used to convert the Srend, Satt, and Sflag parameters. Then use the following table to find the meaning of the 8-bit binary number.

Bit	Indicates	Bit Value
8	-	Always 0 (off).
7	-	Always 1 (on).
6	-	Always 0 (off).
5	-	0 - Reserved for future use.
4	G3 set size	0 = 94 characters. 1 = 96 characters.
3	G2 set size	0 = 94 characters. 1 = 96 characters.
2	G1 set size	0 = 94 characters. 1 = 96 characters.
1	G0 set size	0 = 94 characters. 1 = 96 characters.

(Example)

Suppose the following conditions exist.

. ISO Latin-1 Supplemental is designated as G2 and G3.
. ASCII is designated as G0 and G1.

Then Scss is the ASCII uppercase L character (binary 01001100).

Sdesig is a string of intermediate and final characters indicating the character sets designated as G0 through G3. These final characters are same as those used in select character set (SCS) sequences (Chapter 5).

NOTE: Kanji set can have the following values depending on its version.

It is determined by Set-Up.

$1: DEC Kanji-1978
$3: DEC Kanji-1983

(Example)

Suppose the ASCII character set is designated as G0, DEC Special Graphic as G1, and DEC Supplemental Graphic as G2 and G3. The Sdesig string would be "B0%5%5". Each character corresponds to a final character in an SCS sequence, as follows.

G0	G1	G2	G3
B	0	%5	%5
ASCII	DEC Special Graphic	DEC Supplemental Graphic	DEC Supplemental Graphic

(Example)

The following is an example of a cursor information report.

DCS 1 $ u 1; 1; 1; @; @; @; 0; 3; @; JOI$3 ST

where

1; 1; 1; indicates that the cursor is at row 1, column 1 on the first page.

@; @; @; indicates that (1) no visual character attributes or selective erase attributes are set for writing, (2) DECOM is reset, and (3) there is no SS2, SS3, or autowrap pending. (ASCII @ is binary 01000000.)

0; 3; indicates that G0 is mapped into GL, and G3 is in GR.

@; all character sets have 94 characters.

JOI$3 indicates that JIS-Roman in G0, DEC Special Graphics in G1, JIS-Katakana in G2 and DEC Kanji-1983 in G3.

Notes on DECCIR

- The cursor information in a DECCIR sequence is the same information saved through a save cursor (DECSC) command.

11.5.3 Tab Stop Report (DECTABSR) - VT382 To Host

NOTE: Available in VT300 mode only.

The terminal sends this sequence to the host in response to a request presentation state report (DECRQPSR) sequence. DECTABSR informs the host of the terminal's current tab settings.

PROGRAMMING TIP:
Applications can use the information in the tab stop report to save the current tab stops. Later, the application can restore the saved tab stops.

This operation is useful for applications that need to temporarily change the terminal's tab stops. When the application is finished, it can restore the tab stops that were in effect before the application changed them. You use the restore presentation state (DECRSPS) function to restore tab stops. DECRSPS is described in the next section.

DCS	2	$	u	D...D	ST
9/0	3/2	2/4	7/5	...	9/12

where

D...D is a data string indicating the column number location of each tab stop.

(Example)

The following is an example of a DECTABSR sequence.

DCS 2 $ u 9/ 17/ 25/ 33/ 41/ 49/ 57/ 65/ 73 ST

where

9, 17, 25, 33, 41, 49, 57, 65, and 73 are the column numbers for tab stops.

11.5.4 Restore Presentation State (DECRSPS)

NOTE: Available in VT300 mode only.

This control function restores the terminal to a previous state based on one of the presentation state reports. There are two presentation state reports.

Cursor information report (DECCIR)
Tab stop report (DECTABSR)

A DECRSPS sequence can only restore the information from one report at a time, cursor or tab stop.

PROGRAMMING TIP:
Applications can use DECRSPS to restore the terminal to a previous state specified in a presentation state report. See the previous "Cursor Information Report (DECCIR)" and "Tab Stop Report (DECTABSR)" sections in this chapter.

DCS	Ps	$	t	D...D	ST
9/0	3/?	2/4	7/4	...	9/12

where

Ps indicates the format of the data string (D...D). You can use one of the two following formats for the data string. These formats correspond to the formats used in the two presentation state reports (DECPSR). Make sure you use the format of the report you are restoring.

Ps	Data String Format
0	Error, restore ignored.
1	Selects the format of the cursor information report (DECCIR).
2	Selects the format of the tab stop report (DECTABSR).

D...D is a data string that contains the restored information. This string is identical to the data string used in the report you are restoring - the cursor information report (DECCIR) or tab stop report (DECTABSR).

(Example)

The following DECRSPS sequence restores tab stops according to the tab stop report (DECTABSR).

DCS 2 $ u 9; 17; 25; 33; 41; 49; 57; 65; 73 ST

Note that the data string format above is exactly the same as the format for the tab stop report (DECTABSR).

Notes on DECRSPS

- If there is an invalid value in the DECRSPS sequence, the terminal ignores the rest of the sequence. This action may leave the terminal in a partially restored state.

11.6 MODE SETTINGS

The host can request the current settings of any ANSI or DEC private modes. In response to this request, the terminal returns a report indicating which modes are set and which are reset. The host can use the information in the report to save the current mode settings. Later, the host can restore the mode settings to their saved state.

This operation is useful for applications that need to temporarily change a number of modes. When the application is finished it can restore the modes to their previous state.

The host requests the setting of a mode with a DECRQM sequence. The terminal responds with a DECRPM sequence. The host can then restore a saved setting with an SM or RM sequence. The following sections describe these sequences.

11.6.1 Request Mode (DECRQM) - Host To VT382

NOTE: Available in VT300 mode only.

The host sends this control function to find out if a particular mode is set or reset. The terminal responds with a report mode function (DECRPM).

There are two versions of the DECRQM function, for ANSI and DEC private modes.

< Requesting ANSI Modes >

CSI	Pa	$	p
9/11	3/?	2/4	7/0

where

Pa indicates the ANSI mode that the host is asking about. Table 11-2 lists the values for Pa.

< Requesting DEC Private Modes >

CSI	?	Pd	$	p
9/11	3/15	3/?	2/4	7/0

where

Pd indicates the DEC private mode the host is asking about. Table 11-3 lists the values for Pd.

(Examples)

. The following sequences request the setting of some ANSI modes.

Host Request	Meaning
CSI 2 $ p	What is the current state of keyboard action mode (KAM)? (KAM = 2)
CSI 4 $ p	What is the current state of insert/replace mode (IRM)? (IRM =4)

. The following sequences request the setting of some DEC private modes.

Host Request	Meaning
CSI ? 66 $ p	What is the current state of numeric keypad mode (DECNKM)? (DECNKM = 66)
CSI ? 6 $ p	What is the current state of origin mode (DECOM)? (DECOM = 6)

Notes on DECRQM

. A DECRQM sequence can only ask about one mode at a time.

Table 11-2 ANSI Modes for DECRQM, DECRPM, SM, and RM

Mode	Mnemonic	Pa
error		0
Guarded area transfer mode	GATM *1	1
Keyboard action	KAM	2
Control representation	CRM *2	3
Insert/replace	IRM	4
Status reporting transfer mode	SRTM *1	5
Erasure mode	ERM *1	6
Vertical editing mode	VEM *1	7
Horizontal editing	HEM *3	10
Positioning unit mode	PUM *1	11
Send/receive	SRM	12
Format effector action mode	FEAM *1	13
Format effector transfer mode	FETM *1	14
Multiple area transfer mode	MATM *1	15
Transfer termination mode	TTM *1	16
Selected area transfer mode	SATM *1	17
Tabulation stop mode	TSM *1	18
Editing boundary mode	EBM *1	19
Line feed/new line	LNM	20

*1 This control function is permanently reset.
*2 If CRM is set, the terminal ignores DECRQM and most other control functions.
*3 This control function is permanently set.

Table 11-3 DEC Private Modes for DECRQM, DECRPM, SM, and RM

Mode	Mnemonic	Pd
error		0
Cursor keys	DECCKM	1
ANSI	DECANM*	2
Column	DECCOLM	3
Scrolling	DECSCLM	4
Screen	DECSCNM	5
Origin	DECOM	6
Autowrap	DECAWM	7
Autorepeat	DECARM	8
Katakana shift	DECKANAM	12
Print form feed	DECPFF	18
Printer extent	DECPEX	19
Text cursor enable	DECTCEM	25
Kanji/Katakana	DECKKDM	59
Numeric keypad	DECNKM	66
Backarrow key	DECBKM	67
Sixel display	DECSDM	80

* If DECANM is reset, the terminal ignores DECRQM.

11.6.2 Report Mode (DECRPM) - VT382 To Host

NOTE: Available in VT300 mode only.

The terminal sends this control function in response to a request mode (DECRQM) function. DECRPM informs the host whether a certain mode is set or reset.

PROGRAMMING TIP:
Applications can use the information in the DECRPM report to save the current mode settings. Later, the application can restore the saved mode settings.

This operation is useful for applications that need to temporarily change some of the terminal's mode settings. When the application is finished, it can restore the mode settings that were in effect before the application changed them. You use the set mode (SM) and reset mode (RM) functions to restore mode settings. SM and RM are described in the next section.

There are two versions of DECRPM, for ANSI and DEC private modes.

< Reporting ANSI Modes >

CSI	Pa	;	Ps	$	y
9/11	3/?	3/11	3/?	2/4	7/9

where

Pa indicates which ANSI mode the terminal is reporting on. Table 11-2 lists the values for Pa.

Ps indicates the setting of the mode, as follows.

Ps	Mode Setting
0	Mode not recognized.
1	Set.
2	Reset.
3	Permanently set.
4	Permanently reset.

< Reporting DEC Private Modes >

CSI	?	Pd	;	Ps	$	y
9/11	3/15	3/?	3/11	3/?	2/4	7/9

where

Pd indicates which DEC private mode the terminal is reporting on. Table 11-3 lists the values for Pd.

Ps indicates the setting of the mode. The Ps values are the same as for the ANSI version above.

(Examples)

. The following sequences report the setting of some ANSI modes.

VT382 Report	Meaning
CSI 2 ; 1 $ y	Keyboard action mode (KAM) is currently set. (KAM = 2, set = 1)
CSI 4 ; 2 $ y	Insert/replace mode is currently reset (IRM). (IRM = 4, reset = 2)

. The following sequences report the setting of some DEC private modes.

VT382 Report	Meaning
CSI ? 66 ; 1 $ y	Numeric keypad mode is currently set. (DECNKM = 66, set = 1)
CSI ? 6 ; 2 $ y	Origin mode (DECOM) is currently reset. (DECOM = 6, reset = 2)

Notes on DECRPM

. The terminal can only report on one mode at a time.

11.6.3 Restoring Mode Settings (SM And RM)

ANSI and DEC private modes are control functions that have only two settings, set or reset. Soft terminal reset and hard terminal reset affect many control functions, including some ANSI and DEC private modes.

PROGRAMMING TIP:
Applications can use the SM and RM functions to restore any number of VT382 modes to a desired state. See the previous "Report Mode (DECRPM)" section in this chapter for details.

11.6.3.1 Set Mode (SM)

This control function has two versions. You use the ANSI version to set one or more ANSI modes. You use the DEC private version to set one or more DEC private modes. You cannot set ANSI and DEC private modes with the same SM sequence.

< Setting ANSI Modes >

CSI	Pa	;	...	;	Pa	h
9/11	3/?	3/11	...	3/11	3/?	6/8

where

Pa indicates the ANSI mode to set. Table 11-2 lists Pa values for ANSI modes. You can use more than one Pa value in a sequence.

< Setting DEC Private Modes >

CSI	?	Pd	;	...	;	Pd	h
9/11	3/15	3/?	3/11	...	3/11	3/?	6/8

where

Pd indicates a DEC private mode to set. Table 11-3 lists the Pd values for DEC private modes. You can use more than one Pd value in a sequence.

(Examples)

. ANSI Modes

The following sequence sets keyboard action mode (KAM) and insert/replace mode (IRM).

CSI 2 ; 4 h

where
2 indicates keyboard action mode.
4 indicates insert/replace mode.

. DEC Private Modes

The following sequence sets scrolling mode (DECSCLM) and numeric keypad mode (DECNKM).

CSI ? 4; 66 h

where
4 indicates scrolling mode.
66 indicates numeric keypad mode.

11.6.3.2 Reset Mode (RM)

There are two versions of this control function. You use the ANSI version to reset one or more ANSI modes. You use the DEC private version to reset one or more DEC private modes. You cannot reset ANSI and DEC private modes with the same RM sequence.

< Resetting ANSI Modes >

CSI	Pa	;	...	;	Pa	l
9/11	3/?	3/11	...	3/11	3/?	6/12

where
Pa indicates an ANSI mode to reset. Table 11-2 lists the Pa values for ANSI modes You can use more than one Pa value in a sequence.

< Resetting DEC Private Modes >

CSI	?	Pd	;	...	;	Pd	l
9/11	3/?	3/?	3/11	...	3/11	3/?	6/12

where
Pd indicates a DEC private mode to reset. Table 11-3 lists the Pd values for DEC private modes. You can use more than one Pd value in a sequence.

(Examples)

- ANSI Modes

  The following sequence resets keyboard action mode (KAM) and insert/replace mode (IRM).

  CSI 2 ; 4 l

  where
  2 indicates keyboard action mode.
  4 indicates insert/replace mode.

- DEC Private Modes

  The following sequence resets scrolling mode (DECSCLM) and numeric keypad mode (DECNKM).

  CSI ? 4; 66 l

  where
  4 indicates scrolling mode.
  66 indicates numeric keypad mode.

11.7 CONTROL FUNCTION SETTINGS

NOTE: Available in VT330 mode only.

The host can request the current selection or setting of any control function listed in Table 11-4. In response, the terminal sends the host a report with the requested information. The host can use the information in the report to save the current setting. Later, the host can restore the control function to its saved state.

This operation is useful for applications that need to temporarily change a number of control function settings. When the application is finished, it can restore the control functions to their previous state.

The host requests the setting of a control function with a DECRQSS sequence. The terminal responds with a DECRPSS sequence. The host can then restore the control function, based on the DECRPSS report. The following sections describe DECRQSS and DECRPSS.

Table 11–4 Control Functions for DECRQSS Requests

Control Function	Mnemonic	Intermediate and Final Character(s)
Select active status display	DECSASD	$ }
Set character attribute	DECSCA	" q
Set conformance level	DECSCL	" p
Set status display type	DECSSDT	$ ~
Set top and bottom margins	DECSTBM	r
Select graphic rendition	SGR	m

11.7.1 Request Selection Or Setting (DECRQSS) - Host To VT382

NOTE: Available in VT300 mode only.

The host sends this sequence to ask for the setting of a control function. The terminal responds with a report selection or setting (DECRPSS) sequence.

DCS	$	q	D...D	ST
9/0	2/4	7/1	...	9/12

where

D...D indicates the control function the host is asking about. D...D consists of the intermediate and/or final characters of the control function requested. Table 11-4 lists the control functions the host can ask about, with their final characters.

(Examples)

- The following DECRQSS sequence asks about the select graphic rendition (SGR) function.

  DCS $ q m ST

  where

  m is the final character of the SGR sequence.

- The following sequence asks about the set status display type (DECSSDT) function.

  DCS $ q $ ~ ST

  where
  $~ are the intermediate and final characters of the DECSSDT sequence.

Notes on DECRQSS

- A DECRQSS sequence can only ask about one control function at a time.

11.7.2 Report Selection Or Setting (DECRPSS) - VT382 To Host

NOTE: Available in VT300 mode only.

The terminal sends the host this sequence in response to a request selection or setting (DECRQSS) sequence. The terminal sends DECRQPSS to report the setting of a particular control function.

PROGRAMMING TIP:
Applications can use the information in the DECRPSS report to save the current selections or settings of some control functions. Later, the application can restore the control functions to their saved state.

This operation is useful for applications that need to temporarily change the settings of some of the terminal's control functions. When the application is finished, it can restore the control functions to their previous state.

DCS	Ps	$	r	D...D	ST
9/0	3/?	2/4	7/2	...	9/12

where
Ps indicates whether or not the request from the host is valid.

0 = host's request is invalid.
1 = host's request is valid.

D...D indicates the current setting of a valid control function that the host asked about. D...D consists of all the characters in the control function, except the CSI (9/11) or ESC [(1/11,5/11) introducer characters.

(Examples)

- The host requests the setting of the select graphic rendition (SGR) function. If the current graphic rendition is underline, blinking, and reverse, the terminal responds with the following DECRPSS sequence.

  DCS 1 $ r 0 ; 4 ; 5 ; 7 m ST

  where
  0; 4 ; 5 ; 7 m are all the characters in the SGR sequence, except CSI.

- The host requests the setting of the set top and bottom margin function (DECSTBM). If the current top and bottom margins are set to include the complete screen area, the terminal responds with the following DECRPSS sequence.

  DCS 1 $ r 1 ; 24 r ST

  where
  1 ; 24 r are all the characters in the DECSTBM sequence, except CSI.

- The host requests the setting of a function that the terminal does not recognize. The terminal responds with the following DECRPSS sequence.

  DCS 0 $ r ST

  The terminal does not send a data string (D...D) to the host when the terminal receives an invalid request.

11.8 USER-PREFERRED SUPPLEMENTAL SET (DECRQUPSS)

NOTE: Available in VT300 mode only.

The host can ask for the current user-preferred supplemental set. The terminal responds with the assign user-preferred supplemental set(DECAUPSS) sequence (Chapter 5).

PROGRAMMING TIP:
This operation is useful for applications that need to know what supplemental set the terminal is using.

< Host Request (DECRQUPSS) >
The host requests the current user-preferred supplemental set by sending the following sequence.

CSI	&	u
9/11	2/6	7/5

< Terminal Response >
The terminal uses the DECAUPSS device control string to report the current user-preferred supplemental set (Chapter 5). The terminal sends DECAUPSS in response to a DECRQUPSS sequence. The terminal can send one of the following reports.

DCS 0 ! u % 5 ST	The user-preferred supplemental set is DEC Supplemental Graphic.
DCS 1 ! u A ST	The user-preferred supplemental set is ISO Latin-1 supplemental.

11.9 SUMMARY

Table 11-5 lists all the sequences described in this chapter.

Table 11-5 Sequences for VT382 Reports

Name	Mnemonic	Sequence
< Primary Device Attributes >		
Primary DA request (Host to VT382)	DA	CSI c or CSI 0 c
Primary DA response (VT382 to host)	DA	CSI ? Psc; Psl;...Psn c Psc = operating level 61 level 1 (VT100 mode) 62, 63 level 3 (VT200, VT300 mode) Psl ... Psn = extensions 1 132 columns 2 Printer port 4 Sixel I/O 5 Katakana 6 Selective erase 7 Soft character set (DRCS) 8 User-defined keys 10 Two-byte Kanji 15 Technical Character Set See Table 11-1 for alias responses.
< Secondary Device Attributes >		
Secondary DA request (Host to VT382)	DA	CSI > c or CSI > 0 c
Secondary DA response (VT382 to host)	DA	CSI > Pp; Pv; Po c Pp = identification code. 32 = VT382 terminal. Pv = firmware version.
* Available in VT300 mode only.		

Table 11-5 Sequences for VT382 Reports (cont)

Name	Mnemonic	Sequence
		Po = hardware options. 1 DEC Kanji-1978 2 DEC Kanji-1983
< Device Status Reports >		
< VT382 Operating Status >		
Request (Host to VT382)	DSR	CSI 5 n
Report (VT382 to host)	DSR	CSI 0 n No malfunction. CSI 3 n Malfunction.
< Cursor Position Report >		
Request (Host to VT382)	DSR	CSI 6 n
Report (VT382 to host)	CPR	CSI Pl; Pc R Pl = line number. Pc = column number.
< Printer Status >		
Request (Host to VT382)	DSR	CSI ? 1 5 n
Report (VT382 to host)	DSR	CSI ? 1 3 n No printer. CSI ? 1 0 n Printer ready. CSI ? 1 1 n Printer not ready.

Table 11-5 Sequences for VT382 Reports (cont)

Name	Mnemonic	Sequence
< UDK Status >		
Request * (Host to VT382)	DSR	CSI ? 2 5 n
Report * (VT382 to host)	DSR	CSI ? 2 0 n UDKs unlocked. CSI ? 2 1 n UDKs locked.
< Keyboard Dialect >		
Request * (Host to VT382)	DSR	CSI ? 2 6 n
Report * (VT382 to host)	DSR	CSI ? 2 7 ; Pd n Pd = keyboard dialect. 17 = Katakana
< Terminal State Reports >		
Request * (Host to VT382)	DECRQTSR	CSI Ps $ u Ps = report requested. 0 = ignored. 1 = terminal state report.
Terminal state report * (VT382 to host)	DECTSR	DCS 1 $ s D...D <checksum 1> <checksum 2> ST D...D = report data. <checksum 1> = checksum of all data. <checksum 2> = checksum of all data.
Restore terminal state *	DECRSTS	DCS Ps $ p D...D ST Ps = data string format. 0 = error. 1 = terminal state report. D...D = restored data.

* Available in VT300 mode only.

Table 11-5 Sequences for VT382 Reports (cont)

Name		Mnemonic	Sequence
< Presentation State Reports >			
Request (Host to VT382)	*	DECRQPSR	CSI Ps $ w Ps = report requested. 0 = error. 1 = cursor information report. 2 = tab stop report.
Cursor information report (VT382 to host)	*	DECCIR	DCS 1 $ u D...D ST D...D = data string. See text for description.
Tab stop report (VT382 to host)	*	DECTABSR	DCS 2 $ u D...D ST D...D = tab stops.
Restore presentation state	*	DECRSPS	DCS Ps $ t D...D ST Ps = data string format. 0 = error. 1 = cursor information report. 2 = tab stop report. D...D = data string.
< Mode Settings >			
Request mode (Host to VT382)	*	DECRQM	CSI Pa $ p Pa = ANSI mode. (Table 11-2) CSI ? Pd $ p Pd = DEC private mode. (Table 11-3)
Report mode (VT382 to host)	*	DECRPM	CSI Pa; Ps $ y Pa = ANSI mode. (Table 11-2) Ps = mode state. 0 = unknown mode. 1 = set. 2 = reset. 3 = permanently set. 4 = permanently reset.

* Available in VT300 mode only.

Table 11-5 Sequences for VT382 Reports (cont)

Name	Mnemonic	Sequence
Set mode	SM	CSI Pa; ... Pa h Pa = ANSI mode(s). (Table 11-2) CSI ? Pd; ... Pd h Pd = DEC private mode(s). (Table 11-3)
Reset mode	RM	CSI Pa; ... Pa l Pa = ANSI mode(s). (Table 11-2) CSI ? Pd; ... Pd l Pd = DEC private mode(s). (Table 11-3)
< Control Function Settings >		
Request * (Host to VT382)	DECRQSS	DCS $ q D...D ST D...D = intermediate and/or final characters of function. (Table 11-4)
Report * (VT382 to host)	DECRPSS	DCS Ps $ r D...D ST Ps = 0, valid request. Ps = 1, invalid request. D...D = intermediate and/or final characters of function. (Table 11-4)
< User-Preferred Supplemental Set >		
Request * (Host to VT382)	DECRQUPSS	CSI & u
Report * (VT382 to host)	DECAUPSS	DCS 0 ! u % 5 ST DEC Supplemental Graphic. DCS 1 ! u A ST ISO Latin-1 supplemental

* Available in VT300 mode only.

CHAPTER 12

SIXEL GRAPHICS

12.1 WHAT ARE SIXELS?

The VT382 can send and receive sixel graphics data. You can draw monochrome images with sixel data.

A sixel is a group of six pixels in a vertical column. A pixel (picture element) is the smallest displayable unit on a video screen. Sixels represent bitmap data for a graphic image. The terminal processes sixel data as bits of information. A bit value of 1 means turn on a pixel. A bit value of 0 means turn off the pixel.

You use a single character code for each sixel. The terminal uses 6 bits of the 8-bit character code to encode bitmap data.

You can use sixels to design character sets and fonts for display. Chapter 5 of this manual describes how to design and load soft character sets into the terminal.

> **NOTE: When you enter to Set-Up, the information on the screen disappears, and appears again when you exit. But Sixel data which was displayed is lost.**

12.2 SIXEL DATA FORMAT

The VT382 uses a device control string to send and receive sixel images.

> **NOTE: See Chapter 2 of this manual for general information about device control strings.**

Many of Digital's printers recognize the control format. Here are some examples.

LA50	LA75	LA84	LA86
LA100-J	LA280	LN03S-JA	

Different printers have different output quality. For example, dot matrix printers are very different from laser printers. When you design sixel images on the terminal for printing, you should use parameter values that are appropriate for your printer. For more information, see your printer's programmer reference manual.

12.2.1 Device Control String

The format for the device control string is as follows.

DCS	P1	;	P2;	P3;	q	s...s	ST
9/0	**	3/11	**	**	7/1	***	9/12

where

DCS is a C1 control character that introduces the sixel data sequence. You can also express DCS as the 7-bit escape sequence ESC P (1/11, 5/0) for a 7-bit environment.

P1 is the macro parameter. This parameter indicates the pixel aspect ratio used by the application or terminal. The pixel aspect ratio defines the shape of the pixel dots the terminal uses to draw images. For example, a pixel that is twice as it is wide has an aspect ratio of 2:1. The following list shows the values you can use for P1.

> **NOTE: The macro parameter is provided for compatibility with existing Digital software. New applications should set P1 to 0 and use the set raster attributes control, described later in this chapter.**

P1	Pixel Aspect Ratio (Vertical:Horizontal)
Omitted	1:1 (default)
0, 7, 8, 9	1:1
1, 5, 6	2:1
2	5:1
3, 4	3:1

You can override the setting of the macro parameter by using the set raster attributes character (", 2/2) in a sixel data string. See below.

; is a semicolon (3/11). This character separates numeric parameters in a DCS string.

P2 selects how the terminal draws the background color. You can use one of three values.

P2	Meaning
Omitted, 0 or 2 (default)	Pixel positions specified as 0 are set to the current background color.
1	Pixel positions specified as 0 remain at their current color.

P3 is the horizontal grid size parameter. The horizontal grid size is the horizontal distance between two pixel dots. The VT382 ignores this parameter because the horizontal grid size is fixed.

q indicates that this device control string is a sixel command.

s...s is the sixel-encoded data string. The sixel data characters are characters in the range of ? (hex 3F) to ~(hex 7E). Each sixel data character represents six vertical pixels of data. Each sixel data character represents a binary value equal to the character code value minus hex 3F.

(Examples)

- ? (hex 3F) represents the binary value 000000.
- t (hex 74) represents binary value 110101.
- ~ (hex 7E) represents binary value 111111.

The terminal translates the six bits to a sixel - six pixels in a vertical column. The least significant bit is at the top.

NOTE: For information on how to code sixel characters, see "Soft Character Sets" in Chapter 5 of this manual.

You can also use sixel control functions in the data string. The next section describes these characters and their functions. ST is the string terminator. ST is a C1 control character. You can also express ST as the 7-bit escape sequence ESC \ (1/11, 5/12) for a 7-bit environment.

12.3 SIXEL CONTROL FUNCTIONS

You can use sixel control functions to perform special functions, such as selecting raster attributes.

12.3.1 Graphics Repeat Introducer (!)

The ! (2/1) character introduces a repeat sequence. A repeat sequence lets you repeat a graphic character a specified number of times. You use the following format for the repeat sequence.

!	Pn	character
2/1	**	****

where

Pn is the repeat count. The repeat count can be any decimal value. For example, if you use a repeat count of 23, the next character repeats 23 times.

character is the character to repeat. you can use any character in the range of ? (hex 3F) to ~ (hex 7E).

12.3.2 Raster Attributes (")

The " (2/2) character is the set raster attributes command. This command selects the raster attributes for the sixel data string that follows it. You must use the command before any sixel data string. The " command overrides any raster attributes set by the macro parameter described above. You use the following format for the " command.

"	Pan	;	Pad	;	Ph	;	Pv
2/2	**	3/11	**		**		**

where

Pan and Pad define the pixel aspect ratio for the following sixel data string. Pan is the numerator, and Pad is the denominator.

$$\frac{Pan}{Pad} = \text{pixel aspect ratio}$$

The pixel aspect ratio defines the shape of the pixels the terminal uses to draw the sixel image.

Pan defines the vertical shape of the pixel. Pad defines the horizontal shape of the pixel. For example, to define a pixel that is twice as high as it is wide, you use a value of 2 for Pan and 1 for Pad.

If you use the set raster attributes command (") in a sixel data string, you must specify a pixel aspect ratio. You can only use integer values for Pan and Pad. The VT382 rounds the pixel aspect ratio to the nearest integer.

Ph and Pv define the horizontal and vertical size of the image (in pixels), respectively.

Ph and Pv do not limit the size of the image defined by the sixel data. However, Ph and Pv let you omit background sixel data from the image definition and still have a background color . They also provide a concise way for the application or terminal to encode the size of an image.

NOTE: The VT382 uses Ph and Pv to erase the background when P2 is set to 0 or 2.

12.3.3 Graphics Carriage Return ($)

The $ (2/4) character indicates the end of the sixel line. The active position returns to the left border of the same sixel line. You can use this character to overprint lines.

12.3.4 Graphics New Line (-)

The - (2/13) character indicates the end of a sixel line. The active position moves to the left margin of the next sixel line.

12.3.5 Parameter Separator (;)

The ; (3/11) character separates numeric parameters in a device control string. If there is no number before the separator, the terminal assumes that parameter is 0. If there is a number after the separator, the terminal assumes that parameter is 0.

12.4 SUMMARY

Table 12-1 Sequence for Sixel Graphics

Sequence	Parameter
DCS P1 ; P2 ; P3 ; q s...s ST	
	P1 : The macro parameter that indicates the pixel aspect ratio. 0, 7, 8, 9 = 1:1 (Default) 1, 5, 6 = 2:1 2 = 5:1 3, 4 3:1
	P2 : Selects how the terminal draws the background color. 0, 2 = Set the current background color. (Default) 1 = Remain at their current color.
	P3 : Horizontal grid size parameter. VT382 ignores P3.
	s...s : Sixel-encoded data string. Each Sixel data character represents as 3F to 7E.

Table 12-2 Sixel Control Functions

Name	Function
Graphic repeat introducer	! Pn s Pn= repeat count. s = character to repeat.
Raster attributes	" Pan ; Pad ; Ph ; Pv Pan= numerator of pixel aspect ratio. Pad= denominator of pixel aspect ratio. Ph = horizontal size of image. Pv = vertical size of image.
Graphic carriage return	$ Indicates the end of a Sixel line. The active position returns to the left border of same sixel line.
Graphics new line	- Indicates the end of a Sixel line. The active position moves to the left margin of the next Sixel line.

CHAPTER 13

RESETTING AND TESTING THE TERMINAL

This chapter describes how to reset the settings of many VT382 control functions at the same time. The chapter also describes how to run the power-up self-test and screen alignment pattern by using control functions.

You can also reset your VT382 by using Set-Up. See Chapter 4 of "The VT382 User Guide" for information on using Set-Up.

13.1 RESETTING THE TERMINAL

There are three control functions you can use to reset the terminal.

Name	Function
Soft terminal reset (DECSTR)	Selects most of the power-up factory default settings.
Reset to initial state (RIS)	Selects the settings stored in NVR memory.
Tab clear (TBC)	Clears tab stops.

Soft terminal reset (DECSTR) and reset to initial state (RIS) affect many control functions, including some ANSI and DEC private modes. ANSI and DEC private modes are control functions that have only two settings, set or reset.

13.1.1 Soft Terminal Reset (DECSTR)

This control function changes most of the terminal's current settings to the power-up default settings listed in Table 13-1.

NOTE: Available in VT300 mode only.

CSI	!	p
9/11	2/1	7/0

You can also perform a soft terminal reset by selecting Reset Terminal in the "Set-Up Directory" screen. See Chapter 4 of "The VT382 User Guide"

Notes on DECSTR

- DECSTR affects only those functions listed in Table 13-1.

Table 13-1 Soft Terminal Reset (DECSTR) States

ANSI and DEC Private Modes	Mnemonic	State After DECSTR
Autowrap	DECAWM	No autowrap.
Cursor keys	DECCKM	Normal (arrow keys).
Insert/replace	IRM	Replace.
Keyboard action	KAM	Unlocked.
Numeric keypad	DECNKM	Numeric characters.
Origin	DECOM	Absolute (cursor origin at upper-left of screen).
Text cursor enable	DECTCEM	Cursor enabled.
All character sets	SCS	VT382 default settings.
Assign user-preferred supplemental set	DECAUPSS	Set saved in NVR.
Save cursor state	DECSC	Home position with VT382 defaults.
Select active display	DECSASD	Main display (first 24 lines).
Select graphic rendition	SGR	Normal rendition.
Selective erase attribute	DECSCA	Normal (erasable by DECSEL and DECSED).
Set top and bottom margins	DECSTBM	Top margin = 1. Bottom margin = 24.

13.1.2 Reset To Initial State (RIS)

NOTE: Digital does not recommend using RIS to reset the terminal. You should use a soft terminal reset (DECSTR) instead. RIS causes a communication line disconnect and may change the current baud rate settings. The terminal waits a few seconds before it performs a received RIS function.

The RIS control function, also called a hard reset, causes the terminal to use the saved settings for Set-Up features. The terminal recalls the saved settings from its nonvolatile memory (NVR). The saved settings replace the settings that were in use.

The saved setting for a feature is the same as the factory-default setting, unless you saved a new setting by using the "Save" feature in the "Set-Up Directory". See Chapter 4 of "The VT382 User Guide". You can also recall the saved settings from the "Set-Up Directory" screen, by selecting "Recall".

The RIS sequence is as follows.

ESC	c
1/11	6/3

RIS Actions

- Sets all features listed on Set-Up screens to their saved settings.
- Causes a communication line disconnect, and initializes communication ports.
- Restores user-defined keys to their saved values.
- Clears the soft character set.
- Clears the screen.
- Returns the cursor to the upper-left corner of the screen.
- Sets the select graphic rendition (SGR) function to normal.
- Sets the selective erase attribute (DECSCA) to erasable.

. Selects the default character sets.

13.1.3 Tab Clear (TBC)

This control function clears tab stops.

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

where

Ps indicates the tab stops to clear. There are only two values for Ps, 0 and 3.

Default: Ps = 0.

0 or none	The terminal only clears the tab stop at the cursor.
3	The terminal clears all tab stops.

13.2 TESTING THE TERMINAL

This section describes two control functions that let the host test the VT382 for possible operating errors.

. Invoke confidence test (DECTST)
. Screen alignment pattern (DECALN)

13.2.1 Invoke Confidence Test (DECTST) - Power-Up Self-Test

This control function runs one or more tests to check the terminal's major internal circuits. If possible, the terminal displays an error message on the screen for each error it finds.

CSI	4	;	Ps;	...	Ps	y
9/11	3/4	3/11	**	...	**	7/9

where

Ps indicates a particular test to run.

Ps	Test to Run.
0	All tests (1, 2, 3, 6)
1	Power-up self-test
2	RS232 port data lines loopback test
3	Printer port loopback test
6	RS232 port modem control lines loopback test
7	DEC423 port loopback test
9	Repeats the tests you included in the sequence.

13.2.2 Screen Alignment Pattern (DECALN)

This control function displays the screen alignment pattern on the screen. Service personnel use the alignment pattern to adjust the screen display. DECALN fills the screen with uppercase E's.

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

13.3 SUMMARY

Table 13-2 lists all the control functions described in this chapter.

Table 13-2 Resetting and Testing Sequences

Name	Mnemonic	Sequence
< Resetting the Terminal >		
Soft terminal reset *	DECSTR	CSI ! p
Reset to initial state	RIS	ESC c Not recommended.
Tabulation clear	TBC	CSI 0 g Clear tab at cursor position. CSI 3 g Clear all tabs.
< Testing the Terminal >		
Invoke confidence test	DECTST	CSI 4; Ps; Ps; ... y Power-up self-test. Ps Test to Run. 0 All tests (1, 2, 3, 6) 1 Power-up self-test 2 RS232 port data lines loopback test 3 Printer port loopback test 6 RS232 port modem control lines loopback test 7 DEC423 port loopback test 9 Repeats the tests you included in the sequence.
Screen alignment pattern	DECALN	ESC # 8

* Available in VT300 mode only.

APPENDIX A

VT52 MODE CONTROL CODES

The VT52 mode lets the VT382 terminal operate like a VT52 terminal. You use VT52 mode with applications designed for the VT52.

> **NOTE: VT52 mode may not be included in future Digital terminals. Programmers should only write new software for the ANSI operating mode. Software should avoid switching indiscriminately between ANSI and VT52 modes. In VT52 mode, the terminal ignores many features and settings used in the ANSI environment. To avoid confusion, write all new software for the ANSI operating mode.**

A.1 ENTERING VT52 MODE

You use the DECANM control function to change the terminal to the VT52 mode of operation. In VT52 mode, the VT382 acts like a VT52 terminal. This mode lets you use applications designed for a VT52 terminal.

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

Table A-1 lists and describes all the escape sequences you can use when the terminal is in VT52 mode.

Notes on DECANM

- ANSI private control functions are not available.
- The DEC Supplemental Graphic, ISO Latin-1 supplemental, DEC Technical and Kanji character sets are not available.

- C1 control characters are not available.
- ASCII or JIS-Roman character set to GO.
- The user-defined keys are disabled.

Table 3-3 and 3-4 defines the VT52 keypad codes.

Figure A-1 shows mapping character set in VT52 mode.

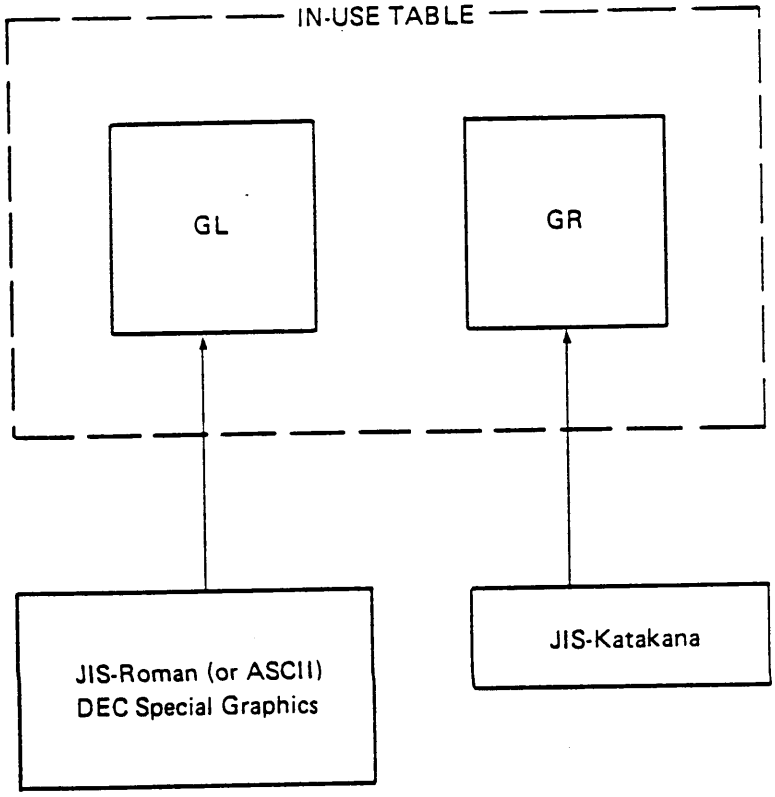

Figure A-1 Designation and Mapping Character Set in VT52 Mode.

A.2 EXITING VT52 MODE

You can exit VT52 mode by using the following escape sequence.

ESC	<
1/11	3/12

NOTE: When you exit VT52 mode, the terminal returns to VT100 mode.

Table A-1 VT52 Escape Sequences

Sequence	Action
ESC A	Cursor up.
ESC B	Cursor down.
ESC C	Cursor right.
ESC D	Cursor left.
ESC F	Enter graphics mode.
ESC G	Exit graphics mode.
ESC H	Move the cursor to the home position.
ESC I	Reverse line feed.
ESC J	Erase from the cursor to the end of the screen.
ESC K	Erase from the cursor to the end of the line.
ESC Y Pn	Move the cursor to column Pn.
ESC Z	Identify. (host to terminal)
ESC /Z	Report. (terminal to host)
ESC :	Set keyboard to KANA.
ESC ;	Set keyboard to JIS (ASCII).
ESC =	Enter alternate keypad mode.
ESC >	Exit alternate keypad mode.
ESC <	Exit VT52 mode.
ESC ^	Enter autoprint mode.
ESC _	Exit autoprint mode.
ESC W	Enter printer controller mode.
ESC X	Exit printer controller mode.
ESC]	Print the screen.
ESC V	Print the line with the cursor.

APPENDIX B

COMPATIBILITY WITH OTHER DIGITAL TERMINALS

Table B-1 Compatibility with Other Digital Terminal

Feature	VT382	VT320	VT282	VT80
< Character Attributes >				
Blinking	Yes	Yes	Yes	Yes
Bold	Yes	Yes	Yes	Yes
Double-height	Yes	Yes	Yes	Yes
Double-width	Yes	Yes	Yes	Yes
Reverse video	Yes	Yes	Yes	Yes
Underline	Yes	Yes	Yes	Yes
< Character Sets >				
ASCII	Yes	Yes	Yes	Yes
DEC Special Graphic	Yes	Yes	Yes	Yes
DEC Supplemental Graphic	Yes	Yes	Yes	No
JIS-Roman	Yes	No	Yes	Yes
JIS-Katakana	Yes	No	Yes	Yes
Down-line-loadable	Yes	Yes	Yes	No
ISO Latin-1	Yes	Yes	No	No
DEC Technical	Yes	No	No	No
DEC Kanji (1978)	Yes	No	Yes	Yes
DEC Kanji (1983)	Yes	No	Yes	No

Table B-1 Compatibility with Other Digital Terminal (cont)

Feature	VT382	VT320	VT282	VT80
< Communication >				
Baud rate up to 19.2K	Yes	Yes	Yes	Yes
Composite video output	No	No	Yes	No
Communication ports				
DEC423 serial	Yes	Yes	No	No
RS232 serial	Yes	Yes	Yes	Yes
20 milliamp	No	No	Yes	No
Printer port				
Connector	6-pin DEC423	6-pin DEC423	9-pin RS232	No -
Bidirectional	Yes	Yes	Yes	No
< Compatibility >				
VT52J	Yes	No	Yes	No
VT80	Yes	No	Yes	Yes
VT102J	Yes	No	Yes	Yes
VT220J	Yes	No	Yes	No
VT282	Yes	No	Yes	No
VT320	Yes	Yes	No	No
< Conformance Level >	3	3	3	1
< 25th Status Line >				
Local	Yes	Yes	Yes	No
Host	Yes	Yes	Yes	No
< Other Features >				
Terminal state inquiry	Yes	Yes	No	No

APPENDIX C

COMMUNICATION

This appendix provides supplemental information about the VT382 communication. For detail see Chapter 7 of "The VT382 User Guide".

C.1 PREVENTING A BUFFER OVERFLOW

If the host system does not respond to the XOFF from the terminal, the input buffer continues to fill with characters. If the host continues to send characters when the buffer is full, the buffer overflows and characters are lost. In place of the lost characters, the terminal displays reverse question mark characters (؟).

To prevent buffer overflows and loss of characters, you can use the following formulas to determine how fast the host system must respond to the first XOFF character. Calculate the overflow first, then the host response time.

NOTE: These formulas assume that you set the "Transmit Rate Limit" feature in the "Communication Port Set-Up" screen to "Unlimited".

1. Overflow

$$OVFL = (MXBF - XOFF) - [3 \times (RCDR/XMDR)] - (RCDR/600)$$

where

OVFL = the number of characters to overflow.
MXBF = the receive buffer size (1024 characters).
XOFF = the first XOFF point (64, 256 or 512).
RCDR = the received data rate (receive speed).
XMDR = the transmitted data rate (transmit speed).

2. Host Response Time

HRST = OVFL x [(DATA + STOP + PRTY + 1)/RCDR]

where

HRST = the host computer response time (in seconds).
OVFL = the number of characters to overflow.
DATA = the number of data bits per character.
STOP = the number of stop bits per character.
PRTY = the number of parity bits per character.

(Example)

Suppose you set the VT382 to send and receive 8-bit characters with no parity, at 9600 bits per second. You select 1 stop bit. You set the first XOFF point to 64 characters. You would calculate the maximum host response time as follows.

OVFL = (1024 - 64) - [3 x (9600/9600)] - (9600/600)
= 941 characters
HRST = 941 x [(8-bits + 1 bit + 0 bits + 1)/9600]
= 0.98 seconds

Therefore, the host system must stop sending data in 0.98 seconds, or the terminal input buffer will overflow.

C.2 USING FILL CHARACTERS

Software that does not support XON and XOFF characters from the terminal can still use all terminal features, by using fill characters. In some applications, you can use the terminal without XON/XOFF support or fill characters. However, the bit rate must be limited to 9600, and the software must not send the ESC (escape code), or use slow scrolling or the printer port.