

CHAPTER 2 DISPLAY MANAGER

2.1 INTRODUCTION

This chapter describes the basic work station operations, the terminal emulation modes and corresponding keyboard functions, the multiple window facility, and the help facility for SCALDsystem operations.

The SCALDsystem work station is a high-resolution, low-glare graphics terminal. On the SCALDsystem I, up to four work stations are connected to an S-32 computer or "cluster controller" that runs the UNIX operating system, while on a SCALDsystem II, either one or two work stations are incorporated into a standalone system. Both the S-32 based and standalone systems optionally can be connected to a computer from the DEC VAX-11 or IBM System/370 families.

The work station is primarily used to support the SCALD Graphics Editor, a schematic drawing tool for logic design. With display manager software and the multiple window facility, however, the work station can be expanded to perform seemingly concurrent operations by allowing independent screens or "windows" to be defined. Each window supports a separate process or operation; up to twelve windows can be defined for work station operations and independently displayed. When the system is connected to a remote host, an additional window can be assigned to communications with the remote host. Since both the SCALDsystem I and II run under the UNIX operating system, windows supporting work station operations are referred to as local windows or "windows into UNIX." Each of these windows operates like a separate terminal, and can run the Graphics Editor, other SCALDsystem analysis tools and programs, or any UNIX program.

Windows into UNIX can be full screen (64 lines by 102 columns) or can occupy only a portion of the screen. The work station supports the full ASCII character set; characters can be displayed in normal or reverse video and can be underlined. Numerous other options and modes can be set as described in this chapter.

2.2 WORK STATION OPERATING MODES

A work station or, when more than one window is defined, each window on a work station, will be in one of the following operating modes:

- o UNIX mode: The work station is logically connected to the SCALDsystem host and emulates a DEC VT100 terminal.
- o Host/VT100 mode: The work station is logically connected to a VAX host computer and emulates a DEC VT100 terminal.
- o Host/3270 mode: The work station is logically connected to an IBM host computer and emulates an IBM 3277 (Model 2) terminal running under a 3271 in local mode.
- o Graphics mode: The work station is connected to the SCALDsystem host and functions as an interactive graphics terminal for the SCALD Graphics Editor.

2.3 THE WORK STATION KEYBOARD

Figure 2-1 shows the key layout of the workstation keyboard.

			TF1	TF2	TF3	TF4	TF5	TF6				DE								
												L								
LF	LF	LF	ES	!	@	#	\$	%	^	&	*	()	-	+	~		RF	RF	RF
1	2	3	CP	1	2	3	4	5	6	7	8	9	0	=	=	`	\	1	2	3
LF	LF	LF	TAB	Q	W	E	R	T	Y	U	I	O	P	{	}	BKSP	RF	RF	RF	
4	5	6												[]		4	5	6	
LF	LF	LF	CP	A	S	D	F	G	H	J	K	L	^	:		"	RET	RF	RF	RF
7	8	9	LK							<-	->			;	v	'		7	8	9
LF	LF	LF	SHI	Z	X	C	V	B	N	M	<	>	?	SHI	LF	RF	RF	RF		
10	11	12	FT								,	.	/	FT			10	11	12	
			CON	SPACE										CON						
			TRL											TRL						

Figure 2-1. The Work Station Keyboard

As shown in the figure, the keyboard includes three groups of special function keys: TF1-TF6 are the Top Function keys, LF1-LF12 are the Left Function keys, and RF1-RF12 are the Right Function Keys. These keys are defined by the work station operating mode of the current or "active" window. The LF1 or "DISPLAY MODE SELECT" key is used in all work station operating modes to display the window menu.

2.4 KEYBOARD TEMPLATES

Three keyboard overlays or "templates" are supplied with each work station: a large, reversible template for VT100 and 3277 key definitions, and two small templates for the left and right function key definitions when running the Graphics Editor. In addition, running the UNIX program <maketemplate> prints an IBM 3277 or DEC VT100 template on the Versatec plotter or prints the Graphics Editor overlays. To print a template for the VT100 or 3277 keyboard, enter:

```
maketemplate vt100 or maketemplate 3277
```

Similarly, to print the Graphics Editor templates enter:

```
maketemplate soft
```

2.5 MULTIPLE WINDOW FACILITY

The window facility allows up to twelve windows into UNIX and a window into a remote host to be displayed on each SCALDsystem workstation screen. Each UNIX window appears as a separate terminal (i.e., there is no communication or interaction among windows), and each window can run the Graphics Editor, any of the other SCALDsystem analysis tools and programs, or any UNIX program. Windows may occupy the full screen or may occupy only a portion of the screen (windows that are less than full screen are called "tiled" windows).

CREATING AN INITIAL UNIX WINDOW

When no windows are defined, the screen is filled with a pattern and there is no response from the keyboard (attempted keyboard entry results in the display of the message "No window picked" across the bottom of the screen). To log in to the system, a window first must be defined. To create a window, either press the LF1 key in the key cluster to the left of the main keyboard or position the cross-hair cursor anywhere along the bottom of

Display Manager
Window Facility

the screen and press any puck button. When the LF1 key or a puck button is pressed, the following menu appears across the bottom of the screen aligned with the "TF" (top function) keys in the top row of the keyboard.

New Unix New Host/VT100 New Host/3270 Delete Pick Name Cancel

To create a UNIX window, press the TF1 key (New Unix) or position the cross-hair cursor over the "New Unix" block on the bottom line of the screen and press one of the puck buttons. When the key or button is pressed, the words "Fullscreen" and "Tiled" appear below the screen menu.

Full Screen Window

To create a full-screen window, position the cross-hair cursor over the word "Fullscreen" and press one of the puck buttons. When the puck button is pressed, the window is drawn, the pattern is erased, and the system banner and log in prompt message are displayed. If desired, log on to the system by entering your user name following the log in prompt.

Tiled Window

There are two methods for creating a tiled window: the two corner method and the width/height method. To use either method, select "New Unix" from the screen menu and then position the cursor over the word "Tiled" and press one of the puck buttons. When a puck button is pressed, the message

Full screen: puck HERE. Tiled: puck two corners, or enter width, height, and puck upper left corner.

is displayed below the screen menu. To create a tiled window using the two corner method, two diagonally-opposed corners are defined with the cross-hair cursor and puck. Position the cross-hair cursor at the screen location of one of the desired window's corners and press one of the puck buttons. Move the cursor to the screen location of the diagonally-opposite corner and again press one of the puck buttons. When the second button is pressed, the outline of the window is drawn, the pattern within the window is erased, and the system banner and log in prompt message are displayed.

To create a tiled window using the width and height method, enter the desired window width (i.e., the number of screen columns). When the first digit is entered, the following message is displayed:

"Enter width (press return when done):

Following the colon, the first digit entered will be displayed. Complete the column width entry (the maximum number of columns is 100) and press the return key. When the return key is pressed, the message "Enter height, then puck in upper left corner" is displayed. Enter the height of the window (i.e., the number of screen rows), position the cursor at the location of the window's upper left corner, and press one of the puck buttons. The window outline, the system banner and log in prompt will be displayed as previously described. If desired, log on to the system by entering your user name following the log in prompt.

NOTE

A full-screen window can be created (after selecting "tiled") by positioning the cursor anywhere in the bottom line and pressing one of the puck buttons.

CREATING ADDITIONAL WINDOWS

To create an additional UNIX window, redisplay the window menu by pressing the LF1 key (or by positioning the cross-hair cursor at the bottom of the screen and then pressing one of the puck buttons). Note that in the menu, the "Pick" (pick window) block is now displayed in normal video (i.e., the pick function is selected from the menu). On the line immediately below the menu, either the letter "A" or the letters "FS A" will be displayed; the letter "A" is the name of the tiled window previously created, and "FS A" is the name of the full-screen window previously created. From the menu, select "New Unix" by either pressing the TF1 key or by positioning the cursor over the "New Unix" block and pressing one of the puck buttons. When the "New Unix" block is selected, the words "Fullscreen" and "Tiled" will appear across the bottom of the screen as previously described. Select the type of window to be created by positioning the cursor over the word "Fullscreen" or "Tiled" and then pressing one of the puck buttons. If "Fullscreen" is selected, a new full-screen window will be created, and if "Tiled" is selected, the message "Full screen: puck HERE. Tiled: puck two corners, or enter width, height, and puck upper left corner." will be redisplayed to allow a tiled window (or a full-screen window) to be created.

Display Manager Window Facility

If you are currently logged in to the initial screen window, the message "logging in as <username>" and the system prompt and cursor are displayed in the new window. Note that the new window is now the active or "picked" window and that the cursor no longer is present in the initial window. Once a user has logged in on the system, it is not necessary to log in again in the new window; the new window is, however, treated as a second terminal on the system (if a "who" command is entered, two different "tty" entries for the same user will be listed). Additional windows can be created by repeating the above procedure; the maximum number of windows that can be created is twelve.

NAMING WINDOWS

Each time a full-screen or tiled window is created, it is automatically assigned a letter (name) by the system in alphabetical order; tiled windows are named with a single letter (A, C, etc.) while full-screen windows are named with three letters (FS B, FS D, etc.). The window names appear in the line below the window menu. Full-screen window names are displayed on all screens; tiled window names are displayed only when the "Pick" block is selected from the menu or when the screen of tiled windows is displayed (the names of the tiled windows collectively appear as "Tiled" when the active window is full screen and the "Pick" block is not selected). Note that the individual tiled window names appear according to their position on the screen (left to right; top to bottom if left edges are aligned).

Any window can be renamed with a user-defined character string of up to nine characters. To rename a window, display the window menu and select the "Name" block by pressing the TF6 key or by positioning the cursor over the block and pressing one of the puck buttons. Next position the cursor over the name of the window to be changed in the bottom screen line and press one of the puck buttons. When the puck button is pressed, the message

Old name was "x". Enter new name:

will appear on the bottom screen line. Enter the new name from the keyboard and press the RETURN key. When the RETURN key is pressed, the window menu will disappear; redisplay the menu to verify the name change.

NOTE

To determine the name of a window, select the "Name" block, position the cursor within the window, and press one of the puck buttons (the name of the window will appear as "old name" in the message). Rather than entering a new name, select the "Cancel" block from the screen menu (i.e., press the DEL key) to abort the operation.

CHANGING THE ACTIVE WINDOW

Each time a new window is created, it becomes the active window for keyboard and puck input (a previously active tiled window still can receive system output as long as no portion of the window is covered or "overlapped" by another window; a process running in an overlapped window is blocked). To reestablish a previously-defined window as the active window, select the "Pick" block (if necessary) from the window menu and, using the puck, position the cross-hair cursor over the name of the window to be reactivated and press one of the puck buttons.

NOTE

Tiled windows additionally can be reactivated by selecting "Pick" from the menu, positioning the cursor anywhere within the window to be redefined as the active window, and then pressing one of the puck buttons.

When a window is picked, the cursor will reappear in the window.

CREATING PASS-THROUGH WINDOWS

The "New HOST/VT100" (TF2) and "New HOST/3270" (TF3) menu functions select a pass-through window that emulates a DEC VT100 terminal on a VAX host or an IBM 3270 terminal on an IBM 370 host, respectively. Note that unlike UNIX windows, only one pass-through window can be created.

To create a VT100 window into the VAX, press the TF2 key or select the "New HOST/VT100" block from the menu with the cursor. The VT100 window prompting is identical to the UNIX window prompting previously described to allow either a full-screen window to be defined or to allow both the size and location of a tiled window to be specified. Note that after a VT100 window is defined, press the return key to initiate the log in message. If the system is not connected to the VAX, the window still can be defined; a block cursor appears in the window, but there is no response to keyboard entry.

Display Manager
Window Facility

To create a 3270 window, press the TF3 key or select the "New HOST/ 3270" block from the menu with the cursor. When "New HOST/3270" is selected, the message "Puck in upper left corner." is displayed across the bottom of the screen. Position the cross-hair cursor at the desired screen location of the window's upper left corner and press one of the puck buttons. When the button is pressed, a fixed size (85 columns by 25 rows) 3270 window is created; if the system is not connected to the host, the message "The 3270/BSC line isn't running" is displayed in the center of the window.

DELETING WINDOWS

Windows are deleted by selecting "Delete" from the menu, identifying the name of the window to be deleted with the cross-hair cursor, and then pressing one of the puck buttons. If the window to be deleted is not the active window, the window simply vanishes when the puck button is pressed and the menu disappears. If the window to be deleted is the active window, the window is deleted and the cursor disappears, but the menu remains at the bottom of the screen. When the last window is deleted, the user is automatically logged off of the system. If the active window is full screen and a tiled window is to be deleted, select "Tiled" from the bottom screen line (after selecting "Delete") to display the tiled screen; tiled windows can be deleted by name or by positioning the puck within the window to be deleted and pressing one of the puck buttons. When the last tiled window is deleted and a full screen window exists, the full screen window is automatically displayed (if no other windows are defined, the patterned screen is displayed). Similarly, if a full screen window is deleted and a tiled window exists, the tiled window is automatically displayed (if no tiled windows are defined, the next full-screen window is displayed; if no other windows are defined, the patterned screen is displayed).

Note that when a user logs off of the system (i.e., enters Control-D, logout, or exit), the window is automatically deleted. Also, when logging out of the last window, an 80-column by 24-row tiled window is created in the upper left corner of the screen.

CAUTION

Deleting a window with an actively running process can lock-up the work station. Logging out of a window that has runnable processes submitted in the background also can lock-up the work station. Care must be taken to avoid these conditions.

OVERLAPPING WINDOWS

When creating tiled windows, a window can overlap a portion (or all) of another window. Note that a window cannot overlap an overlapped area (i.e., a "triple overlap" is not permitted). When an overlap occurs, the most recently-defined window is on top (i.e., the overlapped portion of an existing window is covered). To view the overlapped area, the overlapped window must be "picked" (making it the active window).

ABORTING WINDOW OPERATIONS

If, when using the window menu, a function is incorrectly or inadvertently selected, the intended operation can be aborted either by pressing the DEL (delete) key, space bar, or LF1 key or by selecting the Cancel block from the menu with the cursor and puck button. When the DEL key, space bar, or LF1 key is pressed or when the Cancel block is selected, any menu operation in progress is immediately aborted and the window menu is cleared from the screen. Also, after a function from the window menu is selected (but before the function is initiated), the selected function can be aborted by simply selecting another function from the menu. Specifically:

- o changing a screen name can be aborted before the RETURN is entered.
- o creating a "New Unix" or "New Host/VT100" tiled window can be aborted before the second puck press (two-corner method) or before the upper left corner is defined (width/height method).

WINDOW MANAGER SYSTEM RESOURCES

The window manager software uses a temporary file (/u0/wmtmp) to save and restore screen images. The device driver that controls these operations is shared by the system plotter. Accordingly, performing any window operation that involves the saving or restoring of screen images while performing a "hardcopy" operation will cause one of the following messages to be displayed

Couldn't save screen on disk -- try again later

Couldn't restore screen from disk -- try again later

to indicate that the requested window operation could not be performed at this time (the window operation attempted does not occur; the user remains in the state that existed prior to the intended operation).

2.6 SETUP MODE

Setup is a utility program that allows a number of the Display Manager terminal-configuration variables to be set. The setup mode is entered by typing "setup" following the UNIX prompt. Note that there are two versions of the setup program; the version used is automatically selected according to the size of the active window. When the active window is at least 70 columns wide by 14 lines high, the screen menu shown in figure 2-2 is displayed.

```
          SETUP MODE (Type ? for help; SPACEBAR to exit)
                ANSI Mode <<<          VT-52 Mode
    Keypad Application Mode          >>> Keypad Normal Mode
Cursor Key Mode <Arrow Keys> <<<      Cursor Application Mode
                Auto-wrap <<<          No Auto-wrap
                Parity          >>> No Parity
                Odd Parity          >>> Even Parity
Map BS -> DEL in VT100/Host Mode      >>> Normal
Map Right CNTL -> RET <VT100>        >>> Normal
                Line Cursor          >>> Block Cursor
Graphics Characters          >>> ASCII Characters
                Auto Repeat <<<          No Auto Repeat
                Baud Rate:9600
Scrolling Increment 1
```

Figure 2-2. Setup Screen Menu

Referring to the figure, the first eleven menu items are "toggle" type entries (the parameter currently-selected is pointed to by the triple arrow heads), and the last two menu items are numeric entries. The currently-selected menu item is indicated by the cursor between the parameters (when the setup mode is entered, the cursor is positioned at the first menu item). To select another menu item, use the RETURN key to move down through the menu and use the BACKSPACE key to move up through the menu. To change a toggle-type item, press the "<" key to select the parameter on the left or press the ">" key to select the parameter on the right (the triple arrows will point to the parameter selected). To change a numeric entry, press the "+" key to increase the value or press the "-" key to decrease the value. To exit the setup menu screen, press the space bar; for a summary of the menu selection controls, press the "?" key.

When a window is less than 70 columns wide or less than 14 lines high, the small-window version of the setup program is automatically run. This version offers all the menu items shown in figure 2-2, but displays only one menu item at a time and only the selected parameter. Note that on some menu parameters, the wording is less ambiguous (e.g., "No map BS -> DEL" rather than "normal" and that with the toggle entries, pressing any normal keyboard key will toggle the entry. Table 2-1 describes the setup menu functions and default states.

NOTE

Whenever a user logs off and back on to the system or whenever a new window is created, the default setup parameters are selected. If the parameters for a window are changed and a second window is then created, reselecting (picking) the first window does not alter the first window's setup parameters (i.e., the parameters remain as previously set).

Table 2-1. Setup Screen Menu

Menu Item	Description
ANSI Mode - VT-52 Mode	Selects ANSI (VT100) or VT52 mode escape sequence protocol; default is ANSI mode.
Keypad Application Mode - Keypad Numeric Mode	Selects keypad application mode or keypad numeric mode key codes (see section 2.9); default is keypad numeric mode.
Cursor Key Mode - Cursor Application Mode	Selects cursor key mode or cursor application mode key codes (see section 2.9); default is cursor key mode.
Auto-wrap - No Auto-wrap	When auto-wrap is selected, causes automatic carriage return/line feed when screen row is filled and subsequent character is entered; when no auto-wrap is selected, all characters entered after row is filled are ignored; default is auto-wrap.

Table 2-1. Setup Screen Menu (Con't)

Menu Item	Description
Parity - No Parity	In Host/VT100 mode, enforces character parity with all host communications (parity selected) or ignores character parity (no parity selected); default is no parity.
Odd Parity - Even Parity	In Host/VT100 mode, selects odd or even character parity for host communications if "parity" is specified; default is even parity.
Map BS -> DEL in VT100/ Mode - Normal	Causes delete key code (7FH) to be sent when BACKSPACE key is pressed (map BS -> DEL) or normal backspace key code (08H) to be sent; default is normal.
Map Right CNTL -> RET <VT100> - Normal	Causes carriage return (0DH) to be sent when right CNTL key is pressed (map right CNTL -> RET) or normal control key modifier; default is normal.
Line Cursor - Block Cursor	Selects cursor type (block or underline); default is block cursor.
Graphics Characters - ASCII Characters	Selects VT100 special graphics characters (see table 2-5) or normal ASCII characters; default is normal ASCII characters.
Auto Repeat - No Auto Repeat	Causes character code to be repeated while key is pressed (auto repeat) or single character code (no auto repeat); default is auto repeat. Note that the auto repeat/no auto repeat parameter remains in effect when the active window is changed.

Table 2-1. Setup Screen Menu (Con't)

Menu Item	Description
Baud Rate	Specifies the communications baud rate for the VAX (host/VT100 mode); default is 9600. Note that the baud rate is ignored in the UNIX mode.
Scrolling Increment	Specifies the number of lines that the screen moves up when a scroll operation is performed. The number entered ranges from 1 (the default) to the number of lines in the window.

2.7 HELP FACILITY

An interactive help facility is available to allow a user to obtain on-line tutorial and functional information on SCALDsystem operations. Within the help facility, information is arranged hierarchically and proceeds from a general topic to detailed subtopics. User-defined topics can be added to the help facility.

USING THE HELP FACILITY

The help facility is entered by typing "help" following the UNIX shell prompt. When help is entered, a list of the major topics appears on the screen. To select a major topic, enter the topic name displayed.

NOTE

Topic (and subtopic) names are not case sensitive; names can be abbreviated to their first one or two non-ambiguous characters.

When the topic name is entered, a list of the related subtopics is displayed. Similarly, to select one of the available subtopics, enter the subtopic name. The subtopic information will be displayed on the screen and, if subtopics to the subtopic exist, a list of the further subtopics will be displayed.

Display Manager Help Facility

At any time within the help facility, the following operations are available:

```
<CR>  -- return (pop up) one level in the hierarchy
q      -- exit (quit) the help facility
?      -- redisplay the available subtopics at this level
..     -- return one level (equivalent to <CR>)
.      -- no-op (does nothing)
name   -- name of subtopic to be displayed
```

SPECIFYING PATHS

Once a user becomes familiar with the topics/subtopics available within the facility, a directory or "path" format can be used to traverse the topic hierarchy without having to display any of the intervening topics or subtopics.

Topic paths are specified as follows:

```
/topic/subtopic/.../subtopic
```

or, from the current subtopic:

```
subtopic/.../subtopic
```

As shown, any number of help topics can be specified with each topic/subtopic entry separated by a slash; only the information associated with the last subtopic specified is displayed. As an example of path specification, entering

```
scald/packager/directives
```

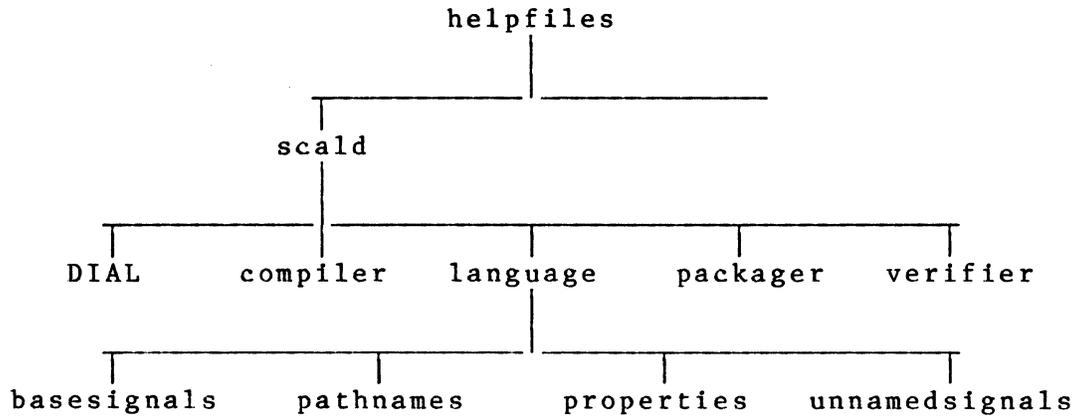
displays only the help information on the packager directives.

If an error is encountered within the path (i.e., a non-existent or ambiguous topic/subtopic is referenced), the help facility displays the last valid subtopic; the prompt line at the bottom of the screen indicates the current topic/subtopic level within the help facility hierarchy.

ADDING USER TOPICS

The help facility can be expanded to include user-defined topics/subtopics (users should not attempt to modify the "SCALD" topic hierarchy). Note that in order to add a topic structure to the help facility, you must be logged in as "root." The help facility is made up of a hierarchical set of directories and

text files located in the directory "u0/scald/help/helpfiles." Under this directory is the directory "scald" which is the root of the Valid-supplied help facility. The scald/language node of this facility is shown in the following illustration.



At the bottom level are the actual descriptions of the subtopics (e.g., the text file that describes "pathnames"). At the top and intermediate levels are directories (e.g., scald, language, etc.). Introductory text to the subtopics can be added at any level by including a text file named "INDEX" at that level of the hierarchy (the contents of the INDEX file are displayed while the names of the directories at that level are displayed as the list of subtopics). To create a user-defined help topic hierarchy, follow the steps outlined below.

1. Log in as root.
2. Change the working directory to "u0/scald/help/helpfiles."
3. Create the top level directory with the mkdir command. The directory name entered is the topic name displayed when the help facility is entered. Note that directory names are limited to a single word (no intervening spaces).
4. Change the working directory to the topic directory created.
5. If the help facility is to be hierarchical (i.e., if an intermediate level of subtopics is to be included), create a directory for each subtopic; if there are to be no subtopics below this level (i.e., if the subtopics are to be explained in text), create a text file for each subtopic as described in step 6.

Display Manager
Help Facility

6. If creating a hierarchical help structure, change the working directory to each of the subdirectories previously created and create the next level of subtopics (directories) or, if at the bottom level of the hierarchy, create a text file for each subtopic. When creating text files, the file (subtopic) name must be a single word (spaces are not allowed). Note that the file contents are displayed literally (no file formatting is performed by the help facility). Also, since the file is concatenated to the screen, the number of lines should not exceed the number of screen rows (64 rows for a full-screen window).

At any level within the hierarchy, introductory comments to the list of subtopics can be included. To include comments, create a text file named "INDEX" at the desired directory level. Note that as a text file, the contents of INDEX are displayed literally.

NOTE

The text files created by Valid are initially formatted using nroff; the "nroffed" versions of the files are then used as the subtopic text and INDEX files.

2.8 COMMUNICATING WITH UNIX -- THE ENVIRONMENT AND TERMCAP

The Display Manager, in conjunction with the Window Facility, sets up several environmental variables that are read by UNIX programs. These variables are:

- o TERM -- the terminal type used as an index into "termcap." Under windows on the Display Manager, term is always Ss (SCALDsystem).
- o TERMCAP -- the termcap (terminal capabilities) entry for Ss which contains, among other parameters, the number of lines and columns in the window. UNIX programs read termcap to determine the number of lines and columns in the active window to allow display-dependent programs to function properly (e.g., Vi uses this information as a reference for all screen-oriented operations).
- o XB, YB -- the pixel coordinate pair describing the lower left window boundary.
- o XT, YT -- the pixel coordinate pair describing the upper right window boundary.

2.9 VT100 KEYBOARD EMULATION

When the work station is logically connected to the SCALDsystem host (UNIX mode) or to a VAX host (Host/VT100 mode) the VT100 keyboard shown in figure 2-3 is emulated.

SET UP												^ <- ->		PF 1	PF 2	PF 3	PF 4		
ES CP	! 1	@ 2	# 3	\$ 4	% 5	^ 6	& 7	* 8	(9) 0	_ -	+ =	~ '	BK SP	BR K	K7	K8	K9	K-
TAB		Q	W	E	R	T	Y	U	I	O	P	{ [}]	DE LT	K4		K5	K6	K,
CT RL	CAPS LOCK	A	S	D	F	G	H	J	K	L	:	" '	RET 	 	K1		K2	K3	E N T
NO SC	SHIFT	Z	X	C	V	B	N	M	< ,	> .	? /	SHIFT	LF	K0		K.	E R		
SPACE																			

Figure 2-3. Typical VT100 Key Layout

The SCALDsystem work station keyboard (see figure 2-1) is divided into four key groups, the three function key groups (LF, TF, and RF) and the central keyboard. The central keyboard key functions are identical to the corresponding VT100 key functions with the following exceptions:

- o Auto repeat, when enabled, is effective in all windows.
- o There is no keyclick mode.
- o CAPS LOCK only causes alphabetic keys to be upshifted.
- o There is no BREAK key. To generate a BREAK, type CONTROL-@ or CONTROL-space (i.e., hold the CONTROL and SHIFT keys down and press the "@" key or hold the CONTROL key down and press the space bar).

Display Manager
VT100 Emulation

- o Several of the central keyboard keys appear at different locations on the two keyboards (e.g., CONTROL and DEL).
- o The VT100 SETUP and NO SCROLL key functions are not implemented.

With the exception of the LF1 (DISPLAY MODE SELECT) and LF12 (META) keys, the left function key group is undefined for VT100 keyboard emulation (the LF2 through LF9 keys have predefined command functions when the work station is operated in the graphics mode). The right function key group and the top function key group combine to form the standard 18-key cluster at the right of the VT100 keyboard. Figures 2-4 and 2-5 show the VT100 key functions for the right and top function key groups respectively (work station key labels are shown in parentheses).

K7	K8	K9
(RF1)	(RF2)	(RF3)
K4	K5	K6
(RF4)	(RF5)	(RF6)
K1	K2	K3
(RF7)	(RF8)	(RF9)
K0	K.	ENTER
(RF10)	(RF11)	(RF12)

Figure 2-4. Right Function Key Group Key Functions

PF1	PF2	PF3 ↑	PF4 ↓	K-	K,
(TF1)	(TF2)	(TF3)	(TF4)	(TF5)	(TF6)

Figure 2-5. Top Function Key Group Key Functions

The top function keys perform dual functions; when pressed by themselves, these keys are the VT100 PF1 through PF4 function keys and the keypad minus (-) and comma (,) keys. When the META key (LF12) is also pressed, the TF3 through TF6 keys emulate the VT100 cursor keys. Note that the TF keys also are used in conjunction with the LF1 key in all operating modes as the window menu selection keys (see section 2.5).

DEC VT100 and VT100 EMULATION

The SCALDsystem workstation emulates a VT100 both in UNIX mode and HOST/VT100 mode. As a VT100 terminal, the work station uses a subset of ANSI standards X3.64-1977 and X3.41-1974 (with DEC and Valid extensions) for VT100 operation. For compatibility with earlier DEC software, the work station also supports a limited subset of VT52 commands. These two modes will be referred to as ANSI Mode and VT52 Mode in the remainder of this section. Most of the terminal operations described in this section are initiated through a series of one or more characters that are preceded by the ASCII escape character (1BH). These character strings are referred to as "escape sequences"; the escape character is represented as <esc> in the individual sequences.

CONTROL KEY FUNCTIONS

The following keys produce control codes when shifted with the CONTROL key: the alphabetic keys, the "@" key, the "^" key, the "_" key, the "|" key, the "[" key, the "]" key, and the SPACE bar. Of these keys, the following subset is recognized by the terminal.

Table 2-2. Control Codes

Hex Code	ASCII Function	Keyboard Sequence	Terminal Operation
1B	ESC	ESC	Introduces a control sequence
07	BEL	CTRL-G	Sounds tone
08	BS	CTRL-H	Backspaces one character position
09	HT	CTRL-I	Moves cursor to next tab stop
0A	LF	CTRL-J	Moves cursor down one line
0B	VT	CTRL-K	Interpreted as line feed (CTRL-J)
0C	FF	CTRL-L	Interpreted as line feed (CTRL-J)
0D	CR	CTRL-M	Returns cursor to column 1
0E	SO	CTRL-N	Invokes G1 character set
0F	SI	CTRL-O	Invokes G0 character set
11	XON	CTRL-Q	Resumes character transmission
13	XOFF	CTRL-S	Stops character transmission
18	CAN	CTRL-X	Abnormally terminates sequence
19	SUB	CTRL-Z	Abnormally terminates sequence

The TAB key generates an ASCII TAB (0BH). Receipt of this control character causes the cursor to advance to the next tab stop or the end of the line, whichever is first. Tabs are set at the ninth column position and every eighth column thereafter (e.g., columns 9, 17, 25, 33, etc.).

CURSOR KEYS

The VT100 cursor keys (the TF3-TF6 keys in combination with the META key on the SCALDsystem keyboard) operate in either the Cursor Key Mode (CKM) or in the Cursor Application Mode (CAM). The cursor mode is selected from the setup menu or from an escape sequence; the default mode is Cursor Key Mode. The following table defines the cursor key output codes generated in these two modes.

Table 2-3. Cursor Key Modes

Cursor Key	VT52 Mode	ANSI Mode and CKM	ANSI Mode and CAM
Up(TF3)	<ESC>A	<ESC>[A	<ESC>OA
Down(TF4)	<ESC>B	<ESC>[B	<ESC>OB
Right(TF6)	<ESC>C	<ESC>[C	<ESC>OC
Left(TF5)	<ESC>D	<ESC>[D	<ESC>OD

KEYPAD KEYS

The VT100's right-hand keypad is controlled by another pair of modes: Keypad Numeric Mode (KNM), and Keypad Application Mode (KAM). The keypad mode is selected from the setup menu or from an escape sequence; the default mode is Keypad Numeric Mode. The following table indicates the output codes generated in the two keypad modes.

Table 2-4. Keypad Modes

Keypad Key	KNM (VT52/ANSI)	VT52 Mode and KAM	ANSI Mode and KAM
K0(RF10)	0	<ESC>?p	<ESC>Op
K1(RF7)	1	<ESC>?q	<ESC>Oq
K2(RF8)	2	<ESC>?r	<ESC>Or
K3(RF9)	3	<ESC>?s	<ESC>Os
K4(RF4)	4	<ESC>?t	<ESC>Ot
K5(RF5)	5	<ESC>?u	<ESC>Ou
K6(RF6)	6	<ESC>?v	<ESC>Ov
K7(RF1)	7	<ESC>?w	<ESC>Ow
K8(RF2)	8	<ESC>?x	<ESC>Ox
K9(RF3)	9	<ESC>?y	<ESC>Oy
K-(TF5)	-	<ESC>?m	<ESC>Om
K,(TF6)	,	<ESC>?l	<ESC>Ol
K.(RF11)	.	<ESC>?n	<ESC>On
ENTER(RF12)	CTRL-M	<ESC>?M	<ESC>OM
PF1(TF1)	<ESC>P	<ESC>?P	<ESC>OP
PF2(TF2)	<ESC>Q	<ESC>?Q	<ESC>OQ
PF3(TF3)	<ESC>R	<ESC>?R	<ESC>OR
PF4(TF4)	<ESC>S	<ESC>?S	<ESC>OS

GRAPHICS CHARACTER SET

In addition to the normal ASCII character set, the VT100 supports a special graphics character set. This character set is selected by the setup mode menu or by the G0/G1 character set designators through the CTRL-O and CTRL-N control key sequences (the assigning of the G0 and G1 character set designators is described in the next section). When the special graphics character set is selected, the characters between 5FH and 7EH are defined as outlined in the following table:

Table 2-5. Special Graphics Character Set

Hex Code	ASCII Graphic	Special Graphic
5F		Blank
60	∇	Diamond
61	a	Checkerboard
62	b	HT Character
63	c	FF Character
64	d	CR Character
65	e	LF Character
66	f	Degree Symbol
67	g	Plus/Minus
68	h	NL Character
69	i	VT Character
6A	j	Lower-right corner
6B	k	Upper-right corner
6C	l	Upper-left corner
6D	m	Lower-left corner
6E	n	Crossing Lines
6F	o	Horizontal line 1
70	p	Horizontal line 3
71	q	Horizontal line 5
72	r	Horizontal line 7
73	s	Horizontal line 9
74	t	Left "T"
75	u	Right "T"
76	v	Bottom "T"
77	w	Top "T"
78	x	Vertical Bar
79	y	<= Character
7A	z	>= Character
7B	{	Pi
7C		Not equal sign
7D	}	UK Pound Sign
7E	~	Centered Dot

ANSI MODE ESCAPE SEQUENCES

The following escape sequences are recognized in the ANSI mode:

- o <ESC>[PnA -- Move Cursor Up
- <ESC>[PnB -- Move Cursor Down
- <ESC>[PnC -- Move Cursor Right
- <ESC>[PnD -- Move Cursor Left

These escape sequences move the cursor in the direction indicated. The sequences can include a numeric argument (Pn) following the "[" to move the cursor a specified number of rows or columns; if the argument is omitted or set to 0 or 1, the cursor is moved one position. Cursor movement stops at a window boundary (i.e., the cursor cannot be moved beyond the left or right margin or top or bottom row).

- o <ESC>[P1;PcH -- Direct Cursor Addressing
- <ESC>[P1;Pcf -- Direct Cursor Addressing (alternate form)

These escape sequences move the cursor to the line (P1) and column (Pc) specified. If no arguments are included, the cursor is moved to its home position at the upper left corner of the window (column 1 of line 1).

- o <ESC>[c or <ESC>[0c -- Device Attributes
- <ESC>Z

These escape sequences are sent from the host and request the terminal to identify its configuration. The terminal always returns the sequence "<ESC>[?1;0c" to indicate a "bare VT100" (no VT100 options).

- o <ESC>= -- Select Keypad Application Mode
- <ESC>> -- Select Keypad Numeric Mode

These sequences select the keypad mode (default keypad numeric mode) as defined in table 2-4. Note that the keypad mode is applicable to both the ANSI and VT52 modes.

- o <ESC>7 -- Save Cursor and Attributes
- <ESC>8 -- Restore Cursor and Attributes

These escape sequences save and restore the cursor's position, graphic rendition, and character set. Note that to use the Restore Cursor and Attributes sequence, the cursor position and attributes must have been previously saved by an <ESC>7 sequence.

Display Manager
VT100 Emulation

- o <ESC>[Psn -- Device Status Report

This escape sequence is issued by the host to request the general status of the terminal or the current location of the cursor. If "Ps" is 5, the terminal returns "<ESC>[0n" to indicate that the terminal is present and ready; if "Ps" is 6, the terminal returns a cursor position report as "<ESC>[Pn;PnR" where the first "Pn" is the cursor row position and the second "Pn" is the cursor column position.

- o <ESC>[PnJ -- Erase in Display
<ESC>[PnK -- Erase in Line

If "Pn" is 0 (or omitted), the escape sequences erase from the cursor to the end of the display (J) or line (K); if "Pn" is 1, these sequences erase from the beginning of the display or line up to (and including) the cursor position. If "Pn" is 2, the entire display or the line containing the cursor is erased. Note that following any of these escape sequences, the cursor remains in its original position.

- o <ESC>D -- Index
<ESC>E -- Next Line
<ESC>M -- Reverse Index

Index moves the cursor down one line in the current column, and Next Line moves the cursor to column 1 of the next line down; Reverse Index moves the cursor up one line in the current column. All three sequences cause the screen to scroll (if necessary).

- o <ESC>(B -- Assign ASCII Character Set to G0 Designator
<ESC>(O -- Assign Graphics Character Set to G0 Designator
<ESC>)B -- Assign ASCII Character Set to G1 Designator
<ESC>)O -- Assign Graphics Character Set to G1 Designator

These escape sequences assign the G0 and G1 character sets that can be subsequently selected by the ASCII control codes SI (select G0 character set) and SO (select G1 character set). The default configuration assigns the ASCII character set to the G0 designator and the graphics character set to the G1 designator.

- o <ESC>[Ps;...;Psm -- Select Graphic Rendition

In the escape sequence, if "Ps" is 0, normal video is selected and if "Ps" is 4 or 7, reverse video is selected. Note that if the underscore cursor and reverse video are selected, the cursor is displayed as an underline.

- o <ESC>[?Ps;...;Psh -- Set Mode
<ESC>[?Ps;...;Psl -- Reset Mode
 - Ps=1 Sets or resets the cursor key mode (when set, the cursor key normal mode is selected); the default cursor key mode is normal.
 - Ps=2 Sets (selects) ANSI mode or resets ANSI mode (selects VT52 mode); the default mode is ANSI.
 - Ps=6 Sets (selects) origin mode (when set, cursor row addressing is from the top of the scrolling region rather than from the top of the window); default is top of window.
 - Ps=7 Sets (enables) auto-wrap mode or resets (disables) auto-wrap mode; the default mode is auto-wrap enabled.
- o <ESC>[Pn;Pnr -- Set Top and Bottom Margins

This sequence sets the top and bottom page margins to define the scrolling region (the area of the screen that can receive new characters by moving previous character lines off of the screen). In the escape sequence, the first "Pn" is the line number of the top line and the second "Pn" is the bottom line. The minimum scrolling region is two lines, and the value specified for the top line always must be less than the value specified for the bottom line. The default scrolling region is the full screen or window. Note that line numbering within the scrolling region is determined by the "set origin mode" escape sequences. If the VT52 mode is subsequently selected, the scrolling region established in the ANSI mode remains in effect.

VALID ESCAPE SEQUENCE EXTENSIONS

The following sequences are Valid extensions to the VT100 escape sequences. These sequences are recognized only in the ANSI mode. Note that a majority of the sequences are used by the setup program (see section 2.6).

Display Manager
VT100 Emulation

- o <ESC>P -- Send Graphics Cursor Coordinates/Function Key Code

Sent by host to cause the terminal to merge puck presses and function key presses into input stream. When this sequence is received, subsequent pressing of one of the puck buttons causes the current x-y cursor coordinates and the button code to be sent to the host, and pressing a function key causes the corresponding function key code to be sent. When a puck button is pressed, the coordinate information is encoded into four printing ASCII characters (c1, c2, c3, and c4); the individual puck button pressed is encoded into a single ASCII character (cb). These characters are sent to the host in the following sequence:

```
~}c1c2c3c4cb
```

In the SCALDsystem, pixel coordinates are specified as two 12-bit binary values in four bytes (only the low-order six bits of each byte are used), a high and low x coordinate (decimal range 0-1023) and a high and low y coordinate (decimal range 0-799). To encode the binary values into printing ASCII characters, each byte is offset by the ASCII space character (20H). As an example of ASCII character encoding, the screen location 980 (x) by 389 (y) would be packed into four bytes as follows:

```
0 0 0 0 1 1 1 1 (high x)
0 0 0 1 0 1 0 0 (low x)
0 0 0 0 0 1 1 0 (high y)
0 0 0 0 0 1 0 1 (low y)
```

Adding the ASCII space character (00100000) to each byte and converting to ASCII produces the following character string:

```
0 0 1 0 1 1 1 1 = /
0 0 1 1 0 1 0 0 = 4
0 0 1 0 0 1 1 0 = &
0 0 1 0 0 1 0 1 = %
```

The puck button pressed is encoded into a single ASCII character as follows:

```
Yellow = !
Green = (
Blue = $
White = "
```

If in the example, the yellow button was pressed while the cursor was at screen location 980,389, the following sequence would be sent to the host:

~}/4&%!

When a function key is pressed, the function key code shown in table 2-6 is sent to the host. Note that the character string assigned to a key is not sent and that the LF1 key is reserved.

Table 2-6. Function Key Codes

Key Pressed	Key Modifier					
	Alone	LF10	LF11	LF12	CTRL	SHIFT
LF2	~@!	~D!	~H!	~P!	~B!	~A!
LF3	~@"	~D"	~H"	~P"	~B"	~A"
LF4	~@#	~D#	~H#	~P#	~B#	~A#
LF5	~@\$	~D\$	~H\$	~P\$	~B\$	~A\$
LF6	~@%	~D%	~H%	~P%	~B%	~A%
LF7	~@&	~D&	~H&	~P&	~B&	~A&
LF8	~@'	~D'	~H'	~P'	~B'	~A'
LF9	~@(~D(~H(~P(~B(~A(
TF1	~@8	~D8	~H8	~P8	~B8	~A8
TF2	~@9	~D9	~H9	~P9	~B9	~A9
TF3	~@:	~D:	~H:	~P:	~B:	~A:
TF4	~@;	~D;	~H;	~P;	~B;	~A;
TF5	~@<	~D<	~H<	~P<	~B<	~A<
TF6	~@=	~D=	~H=	~P=	~B=	~A=
RF1	~@,	~D,	~H,	~P,	~B,	~A,
RF2	~@-	~D-	~H-	~P-	~B-	~A-
RF3	~@.	~D.	~H.	~P.	~B.	~A.
RF4	~@/	~D/	~H/	~P/	~B/	~A/
RF5	~@0	~D0	~H0	~P0	~B0	~A0
RF6	~@1	~D1	~H1	~P1	~B1	~A1
RF7	~@2	~D2	~H2	~P2	~B2	~A2
RF8	~@3	~D3	~H3	~P3	~B3	~A3
RF9	~@4	~D4	~H4	~P4	~B4	~A4
RF10	~@5	~D5	~H5	~P5	~B5	~A5
RF11	~@6	~D6	~H6	~P6	~B6	~A6
RF12	~@7	~D7	~H7	~P7	~B7	~A7

- o <ESC>p -- Discontinue Cursor Coordinate/Keycode Transmission

Discontinue sending cursor coordinates and function key codes.

o <ESC>[Pnb -- Set Baud Rate

Set the baud rate for host communications. In the escape sequence, "Pn" specifies the baud rate as follows:

Pn = 0 = 50	4 = 150	8 = 1800	12 = 4800
1 = 75	5 = 300	9 = 2000	13 = 7200
2 = 110	6 = 600	10 = 2400	14 = 9600
3 = 134.5	7 = 1200	11 = 3600	15 = 19200

Note that when the baud rate is changed through the escape sequence, the baud rate parameter of the setup menu is automatically updated.

o <ESC>b -- Report Baud Rate

Sent by the host to request the baud rate setting. The terminal returns the sequence <ESC>[Pnb where "Pn" is a single printing ASCII character that represents the binary value of the baud rate index (0-15) offset by the ASCII space character (e.g., a baud rate of 9600 has an index value of 14 and is represented by the "." character).

o <ESC>[Pnj -- Set Scrolling Increment

Set scrolling increment for current window. In the escape sequence, "Pn" is the number of lines that the screen moves up when a scroll operation is performed. Note that if the scrolling increment specified is greater than the number of lines in the window, the escape sequence is ignored. The sequence "<ESC>[0;0j" restores the scrolling increment to its value prior to the receipt of the last <ESC>[Pnj sequence.

o <ESC>j -- Report Scrolling Increment

Sent by the host to request the scrolling increment. The terminal returns the sequence <ESC>[Pnj. In the escape sequence, "Pn" is the binary value of the scrolling increment (offset by the ASCII space character) encoded into a single printing ASCII character.

o <ESC>v -- Display Software Version Number

Displays the release date and version of the display manager software currently installed.

- o <ESC>[PnN -- Set Valid Setup Menu Variable
- <ESC>[PnF -- Reset Valid Setup Menu Variable

The above escape sequences set and clear individual setup menu parameters as follows:

Pn No	Set Variable (<code><ESC>[PnN</code>)	Reset Variable (<code><ESC>[PnF</code>)
1	ANSI Mode	VT 52 Mode
2	Key Application Mode	Key Numeric Mode
3	Cursor Key Mode	Cursor Application Mode
4	Wrap Around Mode	No Wrap Around Mode
5	Parity On	Parity Off
6	Odd Parity	Even Parity
7	Map BS to DEL in HOST/VT100	Normal
8	Map Right CTRL to CR (VT100)	Normal
9	Underscore (Line) Cursor	Block Cursor
10	Graphics Character Set	ASCII Character Set
11	Auto Repeat	No Auto Repeat
12	Number of Lines*	

* Report only; cannot be set

o `<ESC>[ON` -- Report Setup Menu Settings

In response to this escape sequence, the terminal sends an `<ESC>` character and a string of 12 ASCII characters followed by a line feed (OAH). The 12 ASCII characters represent the 12 menu settings; each character is offset by 20H (ASCII space). When a menu item is set, a "1" is sent (ASCII "!") and when a menu item is reset, a "0" is sent (ASCII space).

o `<ESC>[OF` -- Report Setup Menu Settings

Identical to `<ESC>[ON`; reports setup menu settings.

o <ESC>W -- Draw Vector

There are two versions of the draw vector escape sequence; one sequence scales the vector screen coordinates to the current window, and the other sequence uses absolute screen coordinates. The version is determined by the first character following the "W." The format of the draw vector escape sequence is:

```
<ESC>WcPx0,Py0,Px1,Py1,Pn <line feed>
.
.
Px0,Py0,Px1,Py1,Pn <line feed>
<ESC>w
```

In the escape sequence, the "c" is the version-select character; if this character is a "U" (or "u"), unscaled (absolute) vectors are drawn and if "c" is any other character, vectors are scaled to the active window. "Px0,Py0,Px1,Py1," is a vector coordinate pair that describes the end points of a vector. The "x/y" values are entered as decimal numbers; "x" ranges from 0 to 1023, and "y" ranges from 0 to 799. "Pn" is a decimal number that defines the display mode/line style of the vector as follows:

- 0: Vector is drawn in replace mode (every pixel in line segment is on).
- 1: Vector is drawn in complement mode (each pixel in line segment is changed to its opposite state).
- 2: Vector is drawn in clear mode (every pixel in line segment is off).
- 3: Same as 0.
- 4-65535: Vector is drawn in bit pattern represented by 16-bit value specified (e.g., a value of 13107 draws a vector with an alternating line style of two bits on and two bits off).

Note that in the unscaled mode, any vectors outside of the window or any vectors with an end point outside of the window are not drawn. Each vector entry must be on a separate line and must be terminated with a line feed character. The sequence is terminated when an escape character, followed by any other character, is encountered or the sequence times out (no characters received for approximately five seconds).

- o <ESC>[Px0;Py0W -- Identify Window

Identifies the window containing the coordinate specified by "Px0;Py0." The number of the window is returned as a single ASCII character (0-9, a or b); if the coordinate is outside of a window, the character "n" is returned. Note that the display manager blindly writes the character into the keyboard queue (the keyboard queue appears as "standard input" to most UNIX programs) and that the queue must be empty in order for the character to be displayed.

- o <ESC>[Px0;Py0;Px1;Py1;Psz -- Draw Block

Draws a block on the screen. The size and position of the block is determined by the x-y coordinate pair (Px0;Py0;Px1;Py1;); the drawing mode is determined by the value of "Ps" as follows:

Ps=0 -- Block is drawn in replace mode (every pixel is turned on).

Ps=1 -- Block is drawn in complement mode (every pixel is changed to its opposite state).

Ps=2 -- Block is drawn in clear mode (every pixel is turned off).

If "Ps" is omitted, "0" is assumed. Note that the block can be drawn anywhere on the screen (i.e., the block does not have to be drawn within the active window).

VT52 MODE ESCAPE SEQUENCES

When the VT52 mode is selected, the terminal responds to the following escape sequences:

- o <ESC>A -- Move Cursor Up
- <ESC>B -- Move Cursor Down
- <ESC>C -- Move Cursor Right
- <ESC>D -- Move Cursor Left
- <ESC>H -- Move Cursor to Home

With the exception of the Home Cursor escape sequence, the above sequences move the cursor one position in the direction indicated. Cursor movement stops at a window boundary.

Display Manager
VT100 Emulation

- o <ESC>YPlPc -- Direct Cursor Addressing

Moves the cursor to line and column specified. In the escape sequence, "Pl" is the line number and "Pc" is the column number. Line and column numbers are sent as ASCII codes; the ASCII code value is the line or column number plus an offset of 1FH. As an example of this ASCII encoding, 20H (ASCII space) references the first line or column, and 2BH (ASCII "+" character) references the twelfth line or column.

- o <ESC>F -- Enter Graphics (Character) Mode

Select special graphics character set.

- o <ESC>G -- Exit Graphics (Character) Mode

Select normal ASCII characters.

- o <ESC>I -- Reverse Line Feed

Move the cursor up one line; if necessary, scroll the screen.

- o <ESC>J -- Erase to End of Screen
<ESC>K -- Erase to End of Line

The above escape sequences erase all characters after the cursor (including the character under the cursor); the cursor remains in its original position.

- o <ESC>= -- Enter Alternate Keypad Mode
<ESC>> -- Exit Alternate Keypad Mode
<ESC>< -- Enter ANSI Mode

The above escape sequences control mode selection; the <ESC>> sequence exits the alternate keypad mode and returns to the normal (numeric) keypad mode, and the <ESC>< sequence exits the VT52 mode and returns to the ANSI mode.

- o ESC Z -- Identify

Causes the terminal to send its identifier sequence (<ESC>/Z) to the host.

2.10 IBM 3270 EMULATION

When a window into an IBM host is created, the work station emulates a 3271 Model 2 controller running up to four 3277 Model 2 terminals in remote mode. Specifically, the 3271 CU address is 0, and the terminal device addresses range from 0 to 3. For additional information on IBM emulation, refer to the IBM documentation provided with the 3270 Information Display System.

The Display Manager provides a complete 3271/3277 emulation with one major exception: the emulation is software dependent rather than hardware dependent. When the S-32 is brought down, the Display Manager stops responding to requests from the 370 host machine, and the host automatically disables the bisynchronous communications (BSC) line. Accordingly, when the S-32 is restarted, the BSC line also must be restarted at the host. If the BSC line is not running when the Host/3270 mode is selected, the message "The 3270/BSC line isn't running" is displayed.

The 3277 PF keys are assigned to the work station's right function keys; the other keys are assigned as follows:

IBM key -----	VALID key -----
CLEAR	LF 3
ERASE INPUT	LF 6
ERASE EOF	LF 9
NOT	TILDE
-> (TAB)	TAB
<- (BACKTAB)	BACKSPACE
<_ (NEWLINE)	RETURN
DUP/PA 1	TF 1
FM/PA 2	TF 2
INSERT MODE	TF 3
DELETE	DEL
6 / CENT	6 / CARET
CURSOR UP	META (LF 12) - RF7
CURSOR DOWN	META (LF 12) - RF8
CURSOR LEFT	META (LF 12) - RF10
CURSOR RIGHT	META (LF 12) - RF11
RESET	LEFT CONTROL KEY
ENTER	RIGHT CONTROL KEY

Display Manager
3270 Emulation

The character set is standard EBCDIC with the non-standard inclusion of a pipe (|) character and the following non-EBCDIC characters:

Graphic	Hex
[AD
]	BD
{	B0
}	D0
\	E0

APL characters and the ASCII characters ~, ^, and ` are not implemented.