

XL SERIES USER GUIDE

OCTOBER 1984

Ann Arbor Terminals, Inc.  
6175 Jackson Road  
Ann Arbor MI 48103

Copyright 1984 by Ann Arbor Terminals, Inc.

The material contained in this document is furnished for reference only, and is subject to change. The techniques described are proprietary and should be treated accordingly.



CONTENTS	Page
INTRODUCTION	
Introduction .....	1-1
Using the Manual .....	1-2
Notational Conventions .....	1-3
System Overview .....	1-5
CHECKOUT & INSTALLATION	
Unpacking .....	2-1
Initial Turn-On .....	2-1
Initial Checkout .....	2-1
Installation .....	2-3
Printer Installation .....	2-4
SETUP	
Using Setup Mode .....	3-1
Saving the Setups .....	3-1
Other Setup Exits .....	3-1
Host Control .....	3-2
Setup Notation .....	3-2
Line A - User Convenience Modes .....	3-3
Line B - Send Parameters .....	3-3
Line C - Print Parameters .....	3-4
Line D - Other Modes .....	3-4
KEYBOARD	
Keyboard Layout .....	4-1
Modes that Affect the Keyboard .....	4-1
Controls that Affect the Keyboard .....	4-2
Programmable Key Strings .....	4-2
Main Keypad Description .....	4-3
Control Keypad Description .....	4-5
Function Keypad Description .....	4-6
DISPLAY	
Display Features .....	5-1
Modes that Affect the Display .....	5-1
Controls that Affect the Display .....	5-2
Display Definitions .....	5-3
Writing the Display .....	5-4
Flow Control of Received Data .....	5-5
Zooming the Display .....	5-6
Moving the Window (w/ Cursor) .....	5-7
Moving the Window (w/o Cursor) .....	5-7
Using the Host Areas .....	5-8
Setting up Multiple Pages .....	5-9
Setting up Multiple Windows .....	5-9
Using the Alternate Cursor .....	5-10
Highlighting the Display .....	5-11
Using the Alternate Character Sets .....	5-12
CURSOR CONTROLS	
Modes that Affect the Cursor Controls .....	6-1
Moving the Cursor .....	6-1
Reading the Cursor Position .....	6-3

Save/Restore Cursor Position .....	6-3
Using Columnar Tabs .....	6-4
Using Page Tabs .....	6-5
EDITING CONTROLS	
Modes that Affect the Editing Controls .....	7-1
Definitions .....	7-1
Erasing the Display .....	7-2
Editing the Display .....	7-3
Using the zSDP Control in Editing .....	7-4
SEND CONTROLS	
Modes that Affect Transmission .....	8-1
Initializing Transmission .....	8-2
Flow Control of Transmitted Data .....	8-2
Aborting a Transmission .....	8-3
Header and Trailer Strings .....	8-3
Using the Transfer Pointer .....	8-3
Using Selected Areas .....	8-4
Using Guarded Areas .....	8-5
PRINTER	
Using the Printer .....	9-1
Printer Setups .....	9-1
Flow Control .....	9-2
Top-of-Page Control .....	9-3
Using Remote Copy .....	9-4
Using Local Copy .....	9-5
Using Page Print .....	9-7
Modes the Affect Page Print .....	9-8
Using the Transfer Pointer .....	9-9
Using Selected Areas .....	9-10
Using Guarded Areas .....	9-11
FORMS	
Definitions .....	10-1
Modes that Affect Form-Filling .....	10-2
Creating a Form .....	10-2
Example Form .....	10-3
Editing the Form .....	10-4
Initializing the Form .....	10-6
Entering Data into the Form .....	10-7
Sending the Entered Data .....	10-8
Correcting and Re-Sending .....	10-9
Printing the Entered Data .....	10-11
OTHER CONTROLS	
DEC Controls .....	11-1
PROGRAMMABLE STRINGS	
Introduction .....	12-1
DCS Control .....	12-1
Key String Examples .....	12-3
Operational Strings .....	12-6
Operational String Examples .....	12-6
Power-on String .....	12-7

Power-on String Examples .....	12-7
Size Limitations .....	12-9
Local Strings .....	12-10
Special Notes .....	12-10
Table 1. DCS Control Summary .....	12-11
Table 2. Default String Programming .....	12-12
DIAGNOSTICS	
Initializing the Terminal .....	13-1
Data Line Monitoring .....	13-2
Local Test .....	13-3
Self Test .....	13-3
ANSI X3.64 STANDARD	
ANSI X3.64 Standard .....	14-1
Code Notation .....	14-1
Control Codes .....	14-1
Escape Sequences .....	14-2
Control Sequences .....	14-2
Parameters .....	14-2
Numeric Parameters .....	14-3
Selective Parameters .....	14-3
Default Values .....	14-3
Private-Use Provisions .....	14-3
APPENDIX	
Introduction .....	A-1
Code Tables .....	A-2
Control and Mode Descriptions .....	A-5



CONTENTS	Page
INTRODUCTION .....	1
USING THE MANUAL .....	2
NOTATIONAL CONVENTIONS .....	3
SYSTEM OVERVIEW .....	5

## INTRODUCTION.

The Ann Arbor XL Series is a family of an advanced, user-configurable, full-featured, ANSI-standard terminals. There are three models in the family, differing primarily in memory and display capacity.

The GENIE+ provides a 60-line memory with up to 30 lines displayed.

The AMBASSADOR adds a full-page (60-line) display.

The GURU adds more memory (approximately 28K bytes, that may be partitioned up to 255 characters by up to 255 lines). and more display (up to 160 characters by up to 66 lines).

**ADVANCED.** The XL-Series gives you a 15-inch screen with large crisp characters. If you've been working at a 12-inch, you'll appreciate the difference. You can 'zoom' more data onto the screen whenever you want (to see what a printout will look like, or to see something that scrolled by too fast, or just for context), while working normally at whatever screen format you find most comfortable. No other terminal gives you that capability.

**USER-CONFIGURABLE.** The XL Series does everything possible to make your life easier. They let you set whether you want a key click, or a margin bell. Whether you want your keys to repeat, and how fast. Whether you want a block or an underline cursor, and whether you want it to blink. They give you a PAUSE key with which you can start and stop data from the host, including stepping in a line at a time or a window at a time.

They give you lots of Programmable keys, and not just the ones up in the Function pad, but the whole keyboard. You can load them with your most-used host (or application) commands, for one keystroke execution, or with control strings that execute complex special functions you want to perform.

They give you a Programmable Power-on string, so that if you don't like some of our design decisions (like our default key codes or tab stop positions), you can change them. The Power-on string is executed each time the terminal is powered on. Load it with the key codes, or tab stop positions, you want, and the terminal initializes to your choices.

**FULL-FEATURED.** The terminals do just about everything you'd like an alphanumeric terminal to do: Move/set either of two independent cursors. Read/save/restore their position. Erase/edit/move data in every conceivable way. Set up multiple pages and windows. Create/use forms, quickly and easily, designed the way you want them. Read/send screen with complete control over the region(s) sent, and with the data compressed as much or as little as you want. Print from the screen or host or both, easily, with complete control of the content and the format of the printout. And we've tried to pro-

vide these functions in a form that is natural, logically consistent, and without exceptions.

**ANSI-STANDARD.** The XL Series is based on the ANSI standard, X3.64, which standardized the coding to be used for control of display and printer terminals. The terminals provide all of the relevant ANSI controls, even some that tend to be redundant. What we consider redundant, some programmer may not, so we include them all. We've added several private-use controls, that we feel extend the versatility of the ANSI set. And we've been careful to make them useful, but not necessary. The XL Series is a powerful family of terminals, even if only pure ANSI is used.

The terminals provide all of the relevant ANSI modes, and many private-use modes. Some of the added modes are for tailoring the terminal to your individual preferences, some of them are for more versatility, and others are for fine-tuning the standard. Where a difference of interpretation might result from the standard, we've tried to provide a mode to accommodate it.

## USING THE MANUAL

The manual is divided into 14 sections and an appendix, not all of which will be of interest to every reader. The sections are summarized below to help guide you to the sections of interest. Section 2 is for the person installing the terminal. Sections 1, 3, 4, and the first couple of pages of 5, should be read by all users. The remaining sections are for the programmer, and the serious user who wishes to take full advantage of the terminal's capabilities.

1. **INTRODUCTION.** Contains an overview of the Manual and the Terminal.
2. **CHECKOUT & INSTALLATION.** How to unpack, install, and checkout the Terminal (and Printer, if applicable)
3. **SETUP MODE.** How to use the SETUP key to locally control the terminal's modes.
4. **KEYBOARD.** Describes the terminal's Keyboard, and how to use it.
5. **DISPLAY.** Describes the terminal's Display, how to write to it, and how to tailor it to your use.
6. **CURSOR CONTROLS.** How to use the terminal's controls to move and position the cursor.
7. **EDITING CONTROLS.** How to use the terminal's controls to erase and edit the display.
8. **SEND CONTROLS.** How to use the terminal's controls to send data from its display memory to the host.
9. **PRINTER.** How to set up and use the Printer.
10. **FORMS MODE.** How to use Forms mode for off-line data entry.
11. **OTHER CONTROLS.** Some non-Ann Arbor private-use controls and modes that have been added for compatibility with commonly-used software.



12. PROGRAMMABLE STRINGS. How to program the terminal's Programmable keys and other Programmable strings.

13. DIAGNOSTICS. How to initialize the terminal, and use its diagnostic features.

14. ANSI STANDARD. A brief review of the ANSI standard as it relates to the XL Series.

APPENDIX. A summary of all of the terminal's controls and modes, arranged by code and by mnemonic, for reference.

## NOTATIONAL CONVENTIONS

NOTES. The following forms are used to direct the reader's attention to important material.

NOTE: Contains restated or supplementary information.

CAUTION: Contains information necessary to correct operation.

WARNING: Contains information necessary to physical safety.

KEY NOTATIONS. The following notation is used in referring to keys and key sequences:

Keys are referred to by their legend, eg, ESC refers to the ESC key, SETUP to the SETUP key, RESET to the RESET key, etc.

Keys that are to be depressed together are shown with a hyphen between them, eg, CTRL-A means to type the A key with the CTRL key depressed.

Keys that are to be depressed sequentially are shown with a space between them, eg, ESC [ Z means to type the keys ESC, [, and Z in sequence, releasing each before depressing the next.

CODE NOTATIONS. The following notation is used in referring to codes and code sequences:

Single codes are denoted by a character followed by the character code, eg, Z (5/10). The character code notation conforms to the ANSI standard, eg, 5/10 is the same as 5A hex.

A Space-code (2/0) is denoted by the symbol Sp. A New-line code is denoted by the symbol NL.

Single-code controls (Control codes) are denoted by the ANSI control mnemonic followed by its code, eg, BS (0/8).

Multiple-code controls (escape and control sequences) are denoted by the ANSI control mnemonic followed by its code sequence, eg, CBT (ESC [ Z). The space between codes is for clarity only, and is not part of the sequence. Long sequences are sometimes shown with segments of the sequence on separate lines for clarity and ease of annotation. No NL is implied by this separation. Spaces and New-lines are included in sequences only if explicitly represented by Sp and NL.

Some controls (Control sequences) permit the inclusion of decimal parameters:

The notation, Pn, refers to a decimal numeric parameter, eg, CUF (ESC [ Pn C). A value up to decimal 255 (ie, the code sequence 3/2 3/5 3/5) may be included in the control at the Pn position. A parameter value of 0, 1, or omission of the parameter are equivalent, unless otherwise specified in the description of the control, eg, the code sequences ESC [ 0 C, ESC [ 1 C, and ESC [ C, all move the cursor one column forward.

The notation, Ps, refers to a decimal selective parameter, eg, EL (ESC [ Ps K). A selective parameter selects from a list of functions defined in the description of the control. The values are passed in decimal, but valid values are limited to the defined list. A parameter value of 0, or omission of the parameter, refer to the first function in the list, unless otherwise specified in the description of the control, eg, the code sequences ESC [ 0 K and ESC [ K both erase from cursor to end of line.

The notation, Pn..Pn or Ps..Ps, refers to multiple parameters. When multiple parameters are passed, they must be separated by a ; (3/11). In the case of numeric parameters, their position in the sequence is important. Therefore, they are usually shown written out, eg, CUP (ESC [ Pn1 ; Pn2 H). In this instance, the value of Pn1 specifies the Line number, and the value of Pn2 the Column number. In the case of selective parameters, their position in the sequence is not important. Therefore, they are usually not written out, eg, SGR (ESC [ Ps..Ps m). As many functions may be selected from the list as desired, and in any order, eg, the code sequences ESC [ 1 ; 5 ; 7 m, ESC [ 7 ; 1 ; 5 m, and ESC [ 5 ; 7 ; 1 m, all result in subsequent characters being written with a bold, blinking, reversed rendition.

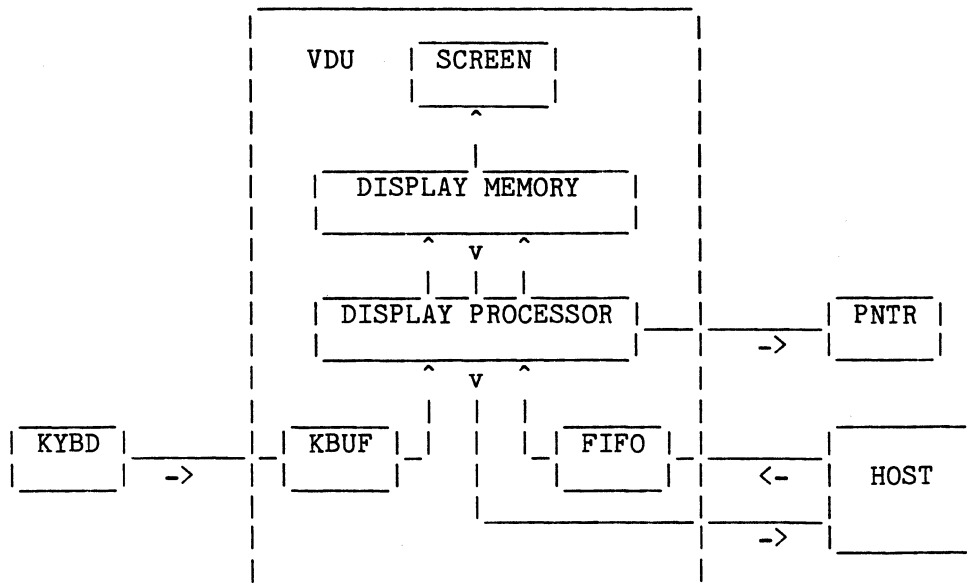
Modes are denoted by the ANSI mode mnemonic followed by its Setup location and mode number, eg, LNM (D,20) means that the mode LNM can be found on Setup line D, and its mode number (for use with the Set/Reset Mode controls) is 20. A dash (-) indicates that the mode is not provided on a Setup line.

A lower-case 'z' preceding any mnemonic means that the control (or mode) is 'private-use', ie, its function was not provided for in the ANSI standard. For example, zKCM (A,26) is an Ann Arbor private-use mode that has been added to the terminals to permit key click to be turned on and off.

NOTE: The ANSI standard anticipated such additions, and reserved code space for them. However, it is likely that the functions, and their codes, will differ among various manufacturers. It is also likely that many of these functions will be incorporated into later revisions of the standard, and assigned different codes in the code space reserved for future standardization. Ann Arbor intends to support future revisions of the standard, adding newly-assigned standard codes, and maintaining existing private-use codes, wherever possible.

## SYSTEM OVERVIEW.

A simplified block diagram of the Terminal system is shown below. The Terminal consists of a Keyboard (KYBD) and a Visual Display Unit (VDU). The Terminal communicates with a Computer (HOST), and may have a Printer (PNTR) attached.



The Keyboard is used to enter data. The entries are buffered in a Keystroke Buffer (KBUF) in the VDU, and go either to the Display Processor (in Monitor SRM, see below) or to the Host (in Simultaneous SRM). Data sent, or echoed back, by the Host are buffered in a received-data FIFO (first-in, first-out buffer), and go to the Display Processor. The Display Processor acts on the data according to the codes it receives. Displayable character codes are passed on to the Display Memory (and, typically, displayed on the Screen), and/or to the Printer. Control codes (and code sequences) are acted upon, and perform their specified function according to the present settings of the Terminal's modes.

SRM SEND-RECEIVE MODE (B,12). This is a basic mode of the Terminal, as it determines the method of communication to be used with the host.

In the set state, the Terminal is said to be in Simultaneous SRM (aka conversational mode, on-line entry, etc). Most users will use the Terminal in this mode. In Simultaneous SRM, the Keyboard is logically disconnected from the VDU. Keyboard inputs are sent to the Host as they are keyed. In the diagram above, the output of the Keystroke Buffer goes directly to the Host. The Display Processor does nothing with the data (except to pass it on.) The host will typically echo the inputs back to the VDU, whereupon they are buffered through the FIFO, acted upon by the Display Processor, and displayed (if graphic characters) or executed (if controls).

In the reset state, the Terminal is said to be in Monitor SRM (aka block mode, off-line entry, etc). In Monitor SRM, the keyboard inputs are not sent to the host as they are keyed. They are, instead, displayed and exe-

cuted locally, and the resultant display subsequently sent to the host as a block. In the diagram above, the keyboard inputs are acted upon by the Display Processor directly. The FIFO is not used by the operator. It may be used by the Host to buffer data to the Printer and/or to Display Memory.

There are two other communication 'modes' in which the Terminal may be used.

**zHDM HALF DUPLEX MODE (B,40).** Permits the outgoing data to be sent to the Host and wrapped back locally into the Terminal's FIFO. This mode may be used with hosts that cannot support echoback. If used with echoback, double-writing of characters results.

**LOCAL TEST MODE.** Permits the outgoing data to be wrapped back locally into the Terminal's FIFO, and not sent to the Host. The two-key sequence, SETUP T, puts the Terminal into Local Test. The two-key sequence, SETUP SETUP, returns it to normal operation. See Diagnostics section. During Local Test, the Host is unable to write to the terminal.

If the Host supports an Xon/Xoff protocol, the Terminal's FIFO can be made to 'handshake' with the Host by setting Auto Xon/Xoff Mode.

**zAXM AUTO XON/XOFF MODE (B,37).** Permits the Terminal to send Xon/Xoff codes to the Host to regulate the flow of received data. During Terminal operation, the FIFO is being filled by the Host, and emptied by the Display Processor. If the Display Processor falls behind, because of a series of slow controls or because the operator has used the PAUSE key to suspend processing, the FIFO gradually fills. When the FIFO is 'almost full', an Xoff code is sent to the Host to suspend further transmission. When the Display Processor catches up, and the FIFO becomes 'almost empty', an Xon code is sent to the Host to resume transmission. The Xon and Xoff codes used by the Host are typically the control codes DC1 (1/1) and DC3 (1/3), respectively. However, they can be set to any ASCII code required by, or convenient to, the Host by changing their values at the end of Setup line B. See Setup Mode section.

If the Host does not support an Xon/Xoff protocol, the data coming from the Host may need to be 'padded', ie, pad characters inserted for time delay. The terminals do not normally provide a pad character (so that the full ASCII code set may be passed to the Printer. However, the NUL (0/0) code can be used as a pad character, if required, by setting Ignore NUL Mode.

**zINM IGNORE NUL MODE (B,54).** Permits the NUL code (0/0) to be used as a pad character, by detecting it at the input to the FIFO and stripping it from the incoming data.

CONTENTS	Page
UNPACKING .....	1
INITIAL TURN-ON .....	1
INITIAL CHECKOUT .....	1
INSTALLATION .....	3
PRINTER INSTALLATION .....	4

## UNPACKING

Inspect the shipping carton for signs of damage before opening. If there is any evidence of damage, notify the carrier and Ann Arbor.

Remove the terminal from the shipping carton. The following items should be enclosed:

- Video display unit (VDU)
- Keyboard
- Stand
- Keyboard cable
- Power cord
- User guide

Inspect the contents for damage. If damage is found, notify the carrier and Ann Arbor. You may wish to save the shipping carton to facilitate return of the equipment for Factory Service should that become required. (Note that the carton is designed for the VDU and stand to ship detached. If returning, be sure to detach and retain the stand to avoid inadvertent shipping damage.)

## INITIAL TURN-ON

Attach the stand to the VDU. Instructions are enclosed with the stand. Add the keyboard cable. Note the small lever on the cable connectors. One end plugs into the rear of the keyboard, with the lever up. The other end plugs into the front of the VDU below the screen, with the lever toward the rear. Add the power cord to the IEC connector on the rear panel. Plug the other end into a standard three-pin 115VAC wall socket. If not possible, use a two-pin socket with proper ground wire attached and connected to ground.

Turn on power. The power switch is located at the front of the VDU below the screen. You'll hear a short audio tone as the unit turns on. Allow the tube to warm up. During this time, the terminal is performing its internal self-diagnostic tests. If it finds no errors, the cursor will appear in the upper-left corner of the screen. If it finds an error, and is able, it will display an appropriate error message on the screen. If not able, it will cause the audio to "chirp". See the Diagnostics section for an explanation of error messages and possible courses of action.

## INITIAL CHECKOUT

Depress the SETUP key on the keyboard. This will bring Setup line A to the bottom of the screen. Turn to the Setup Mode section, and review how to step

through, change, and save the contents of these lines.

Suggested settings for initial checkout are shown below:

```
A 1111 1111 0000 0010 180 20 5
B 9600 0100 1000 0000 0000 0000 0000 17 19
C 56 66 0 0 1800 0110 1100
D 0111 1000 0000 0010 1 0
```

NOTE: The last three parameters of line A are on the Guru only. They control Normal Brightness, Bold Contrast, and Volume respectively. Normal Brightness controls the brightness of the displayed characters. It is recommended that it be set at a minimum comfortable level to prolong the life of the tube and to minimize phosphor burn. Bold Contrast controls the brightness of the Bold rendition, relative to the Normal setting. The cursor is displayed in the Bold rendition. Volume controls the loudness of the key click and bell.

For the Genie+ and Ambassador, Normal Brightness is controlled by a pot at the front of the VDU below the screen. Bold Contrast is controlled by a trimpot on the Logic board inside the VDU. It should not require adjustment. If it does, it should be done only by a qualified technician. Volume is not adjustable.

After setting up the lines as shown (or as you want them), 'Save' them as explained in the Setup Mode section. This will ensure that the terminal powers up with the settings you've chosen.

Now, go back into Setup mode (by typing the SETUP key) and exit to Local Test (by typing the letter T). The terminal is now in Local. As you type keys on the keyboard, the generated character codes are passed to the terminal's communications interface but, instead of being passed on to the terminal's outgoing data line, are instead wrapped back into its data input. You are in effect simulating the host from the keyboard.

Observe the Monitor line at the bottom of the screen. This line displays the character codes as they are processed by the terminal; see Diagnostics section. As you type keys on the keyboard, you can observe what codes are being entered and processed.

Take a few minutes to get familiar with this line. It can save you time in installation, and can be a valuable tool in your use of the terminal. Try a few character keys. Note that they are displayed on the Monitor line as well as being written to the screen. Try a cursor key. The ANSI code sequence generated by the key is displayed in the Monitor line and the cursor moves on the screen. Note that control codes are displayed as reverse-video characters, eg, ESC (CTRL-[) is displayed as a reverse-video left-bracket. Try some other control keys and see what's generated. Try typing in some control sequences, and observe their effect. Examples:

```
ESC [ 10 C           -Moves the cursor right 10 columns
ESC [ 10 ; 10 H      -Moves the cursor to line 10, column 10
ESC [ 1 K            -Erases from beginning of line to cursor
ESC [ 10 @           -Inserts 10 spaces
ESC [ 10 P           -Deletes 10 characters
```

At this point you can skip to INSTALLATION if you wish to continue on-line. If you wish to continue off-line, try programming some of the keyboard keys. Any control sequence, that can be typed in, can be loaded into a key and, thereafter, executed with a single keystroke.

By using the Setup lines to change terminal modes, and programmed keys to facilitate entry of terminal controls, you can explore their interaction to whatever extent you wish before involving the host.

## INSTALLATION

The back panel connector labeled COMPUTER is wired as shown below.

PIN	SIGNAL
1	Protective Ground
2	Transmit Data
3	Receive Data
4	Request to Send
5	Clear to Send
7	Signal Ground
20	Data Terminal Ready

NOTE: The Clear to Send input to the terminal may be used (to enable/disable transmission from the terminal) or left open (= 'on' condition).

A male-male RS232 cable is required. The cable should be shielded for FCC compliance. Acceptable cables are available from any computer supply outlet. A 6-foot cable is available from Ann Arbor, p/n 210641.

Connect the RS232 cable between the terminal and host or modem. Bring Setup line B to the screen. Set the Baud Rate, Parity, and number of Stop Bits, and exit by typing the letter M.

NOTE: Exiting Setup mode with the letter M, returns the terminal to normal (ie, on-line) operation, with the Monitor line displayed at the bottom of the screen, permitting the codes echoed from the host to be observed; see Diagnostics section.

Try operating with the host. The Monitor line displays the codes being actually received from the host. If garbage is being received, check your Baud Rate, or check (or try a different) Parity or Stop Bit setting.

If nothing is being received, make certain (1) you're not in Setup mode, (2) the SRM (Send Receive) setting in Line B is 1, and (3) the KAM (Keyboard Action) setting in Line B is 0.

If you're still receiving nothing, disconnect the RS232 cable at the host end, short pins 2 and 3 together (which wraps the terminal's outgoing data line back into its incoming data line), and observe on the Data Monitor whether the terminal is actually sending your key entries and sending them correctly. This isolates the problem to the terminal or the external equipment.

If you've changed any of your Setup settings in the process of going on-line

with the host, you'll want to 'Save' them before proceeding. Check each of the Setup lines against the diagrams given previously to make sure you've changed only settings you're familiar with. Remember that when you 'Save' the settings you save all four lines at their present state.

## PRINTER INSTALLATION

The back panel connector labeled PRINTER is wired as shown below.

Pin 1. Protective Ground

Pin 2. Transmit Data. Data output from the printer. The terminal recognizes DC3 as a command to disable or suspend printing, and DC1 to enable or resume printing. It ignores all other codes.

Pin 3. Receive Data. Data input to the printer.

Pin 6. Data Set Ready. Control input to the printer. The terminal holds this line in the 'on' condition.

Pin 8. Carrier Detect. Control input to the printer. The terminal holds this line in the 'on' condition.

Pin 7. Signal Ground

Pin 20. Data Terminal Ready. Control output from the printer. An 'off' (ie, -V) condition on this pin causes the terminal to disable or suspend printing. An 'on' (ie, +V) condition causes it to enable or resume printing. The terminal recognizes no connection at the interface (ie, open) as an 'on' condition.

A male-female RS232 cable is required. The cable should be shielded for FCC compliance. Acceptable cables are available from any computer supply outlet. A 6-foot cable is available from Ann Arbor, p/n 210643.

Connect the RS232 cable between the terminal and the printer. Bring Setup line C to the screen, and make the settings required for your printer; see Printer section, Printer Setups.

Exit back on-line (by typing either the SETUP or M key), and try printing. Type the PRINT key to start a Page Print. Type the BREAK key to abort it. If you're using the suggested settings, you're printing from top of screen to cursor (ie, Transfer Termination Mode = 0) so be certain you have the cursor down into the text.

If you're not printing, check the interface requirements in your printer manual. Make sure your printer isn't holding pin 20 low (open is OK), and that it isn't expecting 'on' levels at other than pins 6 and 8. (If it is, you'll have to tie them up to pins 6 or 8, or some other 'on' source.)

If you have a second XL terminal available, try printing to it. Plug the host-end of the 2nd terminal's Computer cable into the 1st terminal's Printer port. This enables you to observe the printed data codes on the 2nd terminal's Monitor line. Typing CTRL-S on the 2nd terminal's keyboard, ie, generating DC3, should cause the printing to stop. Typing



CTRL-Q, ie, generating DC1, should cause it to resume.

Alternatively, use Remote Copy to type directly to the printer; see Printer section, Using Remote Copy. Go into Local Test (by typing the 2-key sequence, SETUP T). Then go into Remote Copy (by typing the 3-key sequence, ESC [ v). Now characters and control codes entered from the keyboard go directly to the printer (and will not display on the screen). Hold down a key (eg, with a pencil between keys) to generate a repetitive character stream to the printer for debug. Typing CTRL-C returns the terminal from Remote Copy. Typing SETUP SETUP returns the terminal from Local Test.

Use of either of these will help you isolate the problem.

Once you've set up your printer, don't forget to 'Save' the parameters.



CONTENTS	Page
USING SETUP MODE .....	1
SAVING THE SETUPS .....	1
OTHER SETUP EXITS .....	1
HOST CONTROL .....	2
SETUP NOTATION .....	2
LINE A - USER CONVENIENCE MODES .....	3
LINE B - SEND PARAMETERS .....	3
LINE C - PRINT PARAMETERS .....	4
LINE D - OTHER MODES .....	4

## USING SETUP MODE

The XL Series' Setup mode is an especially versatile feature. It permits you to review and alter most of the terminal's features and characteristics by bringing functionally-defined Setup lines (A-D) to the screen.

To enter Setup mode, type the SETUP key located above the main keypad. This brings Setup line A to the screen at the bottom of your work area. If there was data there, it's still there. It was just pushed down into memory below the screen. It will return when you leave Setup mode. The cursor is displayed over the letter A. You may use the DOWN-ARROW key on the Control keypad to step through the lines A through D sequentially, and the UP-ARROW key to step back through them.

To change a setting, simply move the cursor to the setting using the RIGHT- and LEFT-ARROW keys on the Control keypad. Note that the setting names automatically display in the right-half of the line. When the cursor is positioned over the desired setting, use the + (plus) key to set (or advance) the setting, or the - (minus) key to reset (or decrease) it.

To exit Setup mode, simply type the SETUP key again. The Setup line will disappear and the line it displaced will return to the screen.

## SAVING THE SETUPS

The setting changes made above are temporary. They will be lost when the terminal is turned off (or reset), and will be restored to their power-on values when the terminal is turned back on. If you wish to change their power-on value as well, simply type the letter S (Save), instead of the SETUP key, to exit Setup mode. This saves all of your present Line A-D settings in non-volatile memory.

## OTHER SETUP EXITS

There are other Setup exits that you will find useful. See the Diagnostics section for a more complete description.

Exiting Setup mode by typing the letter M (Monitor) brings a Monitor line to the bottom of the screen. The Monitor line lets you view what you are receiving from the host (including control codes), in real time, without affecting

your display or normal terminal operation. Depressing the SETUP key twice removes the line from the screen.

Exiting Setup mode by typing the letter T (Test) puts the terminal into Local and brings up the Monitor line. Keyboard entries are wrapped back into the terminal, as received data, and are displayed on the Monitor line so that you can see exactly what you are entering. Depressing the SETUP key twice returns the terminal on-line, and removes the line from the screen.

Exiting Setup mode by typing the letter X (Xparent) puts the terminal into Transparent mode. All codes subsequently entered or received are displayed (on the screen) and not acted upon. Typing the SETUP key twice returns the terminal to normal operation.

Exiting Setup mode by typing the letter Z causes the terminal to go into self-test. In either case the tests can be aborted by holding down the RESET key.

## HOST CONTROL

Most of the settings can also be changed by the host. Notice that some settings have a number displayed in parentheses following their name. The host may control these settings by sending that number as part of an SM/RM (Set/Reset Mode) control sequence. The host may control the page and margin settings of the printer (Line C) with the control zSPF (Set Print Format). The host cannot control the baud rate or parity settings on either the host (Line B) or printer (Line C) ports.

Since the SETUP key is programmable (see Programmable Strings section), the host may disable the key to prevent operator access to the Setup lines.

## SETUP NOTATION

The contents of the Setup lines are listed on the following pages. Terminal modes are shown with their mnemonic, name, mode number (in parentheses), and a short-form description. A full description of the modes can be found in the Appendix, arranged alphabetically by their mnemonic.

A mode is referred to as 'reset' if its value is zero, 'set' if its value is one. In most cases, the mode name implies the set state, eg, setting Key Click Mode (zKCM) enables key click. For some ANSI modes the opposite is true, eg, resetting Format Effector Transfer Mode (FETM) enables the transfer.

LINE A - USER CONVENIENCE MODES. This line displays a number of terminal modes that are primarily for the convenience of the user.

zBKCM	Block Cursor Mode (31)	0= Underline cursor
zBNCM	Blinking Cursor Mode (32)	0= Steady cursor
zKCM	Key Click Mode (26)	0= No click
zMBM	Margin Bell Mode (25)	0= No bell
zKPCM	Key Pad Control Mode (27)	0= Numeric pad
zRLM	Return Key CRLF Mode (29)	0= No LF
zKRM	Key Repeat Mode (28)	0= No repeat
zFRM	Fast Repeat Mode (47)	0= Slow
zAPM	Auto Pause Mode (38)	0= Manual only
zSSM	Slow Scroll Mode (39)	0= No delay
zHAM	Hold in Area Mode (48)	0= Auto skip
	Reserved	
zTPDM	Transfer Pointer Display Mode (41)	0= No display
zCLIM	Caps Lock Invert Mode (53)	0= Upper case only
	Reserved	

The following modes are active on the Guru only:

zIVM	Inverse Video Mode (59)	0= Dark background
	Normal Brightness	0-254
	Bold Contrast	0-31
	Volume	0-7

LINE B - SEND PARAMETERS. This line displays the parameters that tailor the input/output channel to the host computer.

	Baud Rate	110-19200
	Parity	00=Odd 01=Even 10=Space 11=Mark
	Stop Bits	0=1 1=2
	Parity Detection	0=Off 1=On
SRM	Send-Receive Mode (12)	0= Monitor SRM
zHDM	Half-Duplex Mode (40)	0= No local echo
KAM	Keyboard Action Mode (2)	0= Enable keyboard
zAKDM	Auto Keyboard Disable Mode (46)	0= Auto re-enable
TTM	Transfer Termination Mode (16)	0= Stop at cursor
GATM	Guarded Area Transfer Mode (1)	0= Honor guard
SATM	Selected Area Transfer Mode (17)	0= Selected areas only
MATM	Multiple Area Transfer Mode (15)	0= Active area only
FETM	Format Effector Transfer Mode (14)	0= Line separators sent
zFSTM	Field Separator Transfer Mode (44)	0= No page tabs sent
zCSTM	Column Separator Transfer Mode (43)	0= No column tabs sent
zGRM	Graphic Rendition Transfer Mode (45)	0= No SGR's sent
zAXM	Auto Xon/Xoff Mode (37)	0= Disable
zLTM	Line Transfer Mode (42)	0= No pause at EOL

zMKM	Meta Key Mode (52)	0= Disable
zMMM	Meta Monitor Mode (57)	0= Disable
zINM	Ignore NUL Mode (54)	0= Pass NUL's
	XON Code	0-127
	XOFF Code	0-127

LINE C - PRINT PARAMETERS. This line displays the parameters that tailor the output channel to the printer.

	Printed Lines	0-126
	Total Lines	0-126
	Left Margin	0-254
	New-Line Pads	0-254
	Baud Rate	110-19200
	Parity	00=Odd 01=Even 10=Space 11=Mark
	Stop Bits	0=1 1=2
zGAPM	Guarded Area Print Mode (49)	0= Print as spaces
	New-Line Character	01=LF 10=CR 11=CRLF
	Form Feed	0=NLS 1=FF
z8RCM	8-Bit Remote Copy Mode (61)	0= 7-bit

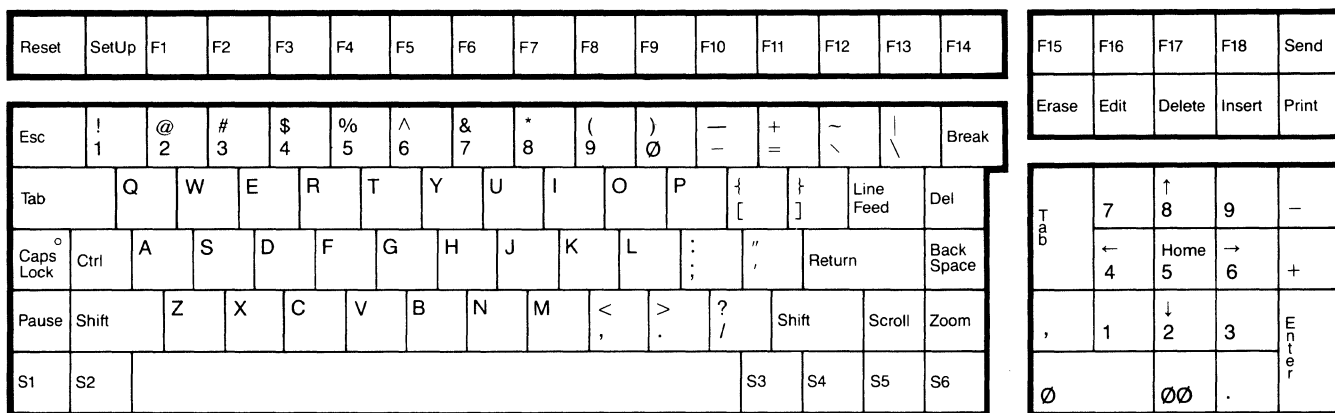
LINE D - OTHER MODES. This line displays the remaining terminal modes that are not displayed on the previous lines.

zICM	Invisible Cursor Mode (56)	0= Display cursor
zWFM	Wrap Forward Mode (33)	0= No wrap
zWBM	Wrap Backward Mode (34)	0= No wrap
zDBM	Destructive Backspace Mode (30)	0= Non-destructive
zDDM	DEL-character Display Mode (35)	0= Don't display
zSPM	Scroll-Page Mode (36)	0= Scroll
TSM	Tabulation Stop Mode (18)	0= Column tabs
ERM	Erasure Mode (6)	0= Honor protect
LNM	LF New Line Mode (20)	0= No CR
zCNM	CR New Line Mode (55)	0= No LF
	Reserved	
	Reserved	
	Reserved	
	Reserved	
dVAM	VT52-ANSI Mode	0= ANSI
dOM	Origin Mode	0= Page
SEE	Select Editing Extent Error Report Code	0=Page 1=Line 2=Field 3=Area See Diagnostics

CONTENTS	Page
KEYBOARD LAYOUT .....	1
MODES THAT AFFECT THE KEYBOARD .....	1
CONTROLS THAT AFFECT THE KEYBOARD .....	2
PROGRAMMABLE KEYS .....	2
MAIN KEYPAD DESCRIPTION .....	3
CONTROL KEYPAD DESCRIPTION .....	5
FUNCTION KEYPAD DESCRIPTION .....	6

KEYBOARD LAYOUT

The XL Series features a low-profile typewriter-style keyboard. The keyboard contains a 68-key Main keypad, a 17-key Control pad (to the right), and two Function pads (above) containing 26 keys. The operation of these keypads, under the various modes available to the user, is described in this section.



MODES THAT AFFECT THE KEYBOARD

The following modes affect your use of the keyboard. Their settings may be reviewed, and changed as desired, on the Setup lines; see Setup section. The Setup line letter and mode number are shown in parentheses following the name. Detailed descriptions of these modes can be found in the Appendix.

- zKCM KEY CLICK MODE (A,26). Turns on key click.
- zMBM MARGIN BELL MODE (A,25). Turns on margin bell.
- zKPCM KEY PAD CONTROL MODE (A,27). Inverts the Control keypad in respect to the SHIFT key.
- zRLM RETURN KEY CRLF MODE (A,29). Lets the RETURN key generate CR or CRLF. Note: The modes LNM (LF New Line) and zCNM (CR New Line) determine which code(s) do a new-line action when received by the terminal.
- zKRM KEY REPEAT MODE (A,28). Enables the keys to auto-repeat when held down for more than 1/2 second.

zFRM FAST REPEAT MODE (A,47). Selects a faster repeat rate.

zCLIM CAPS LOCK INVERT MODE (A,54). Enables lower-case entry, with the SHIFT key, when the CAPS LOCK key is down.

KAM KEYBOARD ACTION MODE (B,2). Locks the keyboard (see Controls below). The mode cannot be 'saved' in the set (locked) state.

zMKM META KEY MODE (B,52). Lets the S1 key act as a META SHIFT key.

zDBM DESTRUCTIVE BACKSPACE MODE (D,30). Lets the BACKSPACE key do a destructive backspace.

#### CONTROLS THAT AFFECT THE KEYBOARD

The keyboard may be locked and unlocked with the controls DMI/EMI. When the keyboard is locked, key entry is inhibited (attempted entries result in an error 'beep') and the cursor is not displayed. If this happens accidentally, go into Setup (the SETUP key is not inhibited) and reset the KAM (Keyboard Action Mode) setting on Line B.

DMI DISABLE MANUAL INPUT (ESC `). Locks the keyboard.

EMI ENABLE MANUAL INPUT (ESC b). Unlocks the keyboard.

#### PROGRAMMABLE KEYS

The keyboard can be programmed with the DCS control; see Programmable Strings section. This control allows you to download strings into specified keys and thereafter enter them with a single keystroke. The strings may be sequences of characters you regularly type, eg, operating system, editor, or application commands. Or they may be terminal controls, or sequences of controls, that you find useful. The control may be used to add strings to the keyboard or to rearrange or disable any or all of the default key programming described in the following sections.

Each key may be programmed on multiple levels, ie, a single key may be made to generate different strings depending on what other keys are depressed with it. The top four levels are unshifted, SHIFT, CTRL, and CTRL-SHIFT. All keys, except the 48 graphic character keys on the Main keypad, are programmable on these levels.

Up to 28 additional lower levels may be used by simply designating certain keys to be shifters. All keys, including the graphic character keys, are programmable on these levels. For example, you might designate S1 to be a shift key, and store operating system commands on the S1-level under alphabetic keys that relate mnemonically to the commands. Similarly, you might store editor commands on an S2-level, application commands on an S3 level, etc.

Note, however, that although the terminal allows you to store your strings at over 3400 different key/level positions, it's actual string memory is limited (896 bytes). The intent of this facility is to give you a great deal of freedom in where you put your strings; not to store a large number of them.



In practice, the strings are best stored in the host, as files containing separate DCS controls, and downloaded by your command or prepended to your applications. For example, a file containing operating system commands might be downloaded as part of your log-in process. A file containing commands for a particular application may then be downloaded, on top of the first set, as part of the process of initiating the application. If the the application strings are loaded into different key/level positions, both are active. If the strings for a subsequent application are loaded into the same key/level positions as the previous application, the new application strings replace the old, and no additional string memory is required.

MAIN KEYPAD DESCRIPTION

The Main keypad consists of alphabetic keys (single legend), non-alphabetic keys (2 legends), the CTRL key, and several other keys for terminal control. For each group, the code (or control) generated by the keys is tabulated and, for the terminal control keys, a short description is given of the function performed by the key (when its control is received by the terminal). A more complete description of the controls may be found in the Appendix.

ALPHABETIC KEYS. The 26 alphabetic keys generate lower-case codes unless either of the SHIFT keys, or the CAPS LOCK key, are down. With the CAPS LOCK key down, the keys generate upper-case codes, unshifted and shifted. (The keys may be made to generate lower-case codes, shifted, by setting zCLIM.)

UNSHIFTED	SHIFTED	UNSHIFTED	SHIFTED	UNSHIFTED	SHIFTED
a (6/1)	A (4/1)	j (6/10)	J (4/10)	s (7/3)	S (5/3)
b (6/2)	B (4/2)	k (6/11)	K (4/11)	t (7/4)	T (5/4)
c (6/3)	C (4/3)	l (6/12)	L (4/12)	u (7/5)	U (5/5)
d (6/4)	D (4/4)	m (6/13)	M (4/13)	v (7/6)	V (5/6)
e (6/5)	E (4/5)	n (6/14)	N (4/14)	w (7/7)	W (5/7)
f (6/6)	F (4/6)	o (6/15)	O (4/15)	x (7/8)	X (5/8)
g (6/7)	G (4/7)	p (7/0)	P (5/0)	y (7/9)	Y (5/9)
h (6/8)	H (4/8)	q (7/1)	Q (5/1)	z (7/10)	Z (5/10)
i (6/9)	I (4/9)	r (7/2)	R (5/2)		

NON-ALPHABETIC KEYS. The non-alphabetic keys generate the lower-legend codes unless either of the SHIFT keys are down. The CAPS LOCK key does not affect the codes generated by these keys. The SPACE BAR generates the code 2/0, both unshifted and shifted.

UNSHIFTED	SHIFTED	UNSHIFTED	SHIFTED	UNSHIFTED	SHIFTED
1 (3/1)	! (2/1)	8 (3/8)	* (2/10)	[ (5/11)	{ (7/11)
2 (3/2)	@ (4/0)	9 (3/9)	( (2/8)	] (5/13)	} (7/13)
3 (3/3)	# (2/3)	0 (3/0)	) (2/9)	;	: (3/10)
4 (3/4)	\$ (2/4)	- (2/13)	_ (5/15)	' (2/7)	" (2/2)
5 (3/5)	% (2/5)	= (3/13)	+ (2/11)	,	< (3/12)
6 (3/6)	^ (5/14)	` (6/0)	~ (7/14)	.	> (3/14)
7 (3/7)	& (2/6)	\ (5/12)	(7/12)	/ (2/15)	? (3/15)

CTRL (CONTROL) KEY. With the CTRL key depressed, the preceding keys generate control codes. (The CTRL key resets the two high-order bits of the normal code.)

CODE	KEY	CODE	KEY	CODE	KEY
NUL (0/0)	CTRL-@	VT (0/11)	CTRL-K	SYN (1/6)	CTRL-V
SOH (0/1)	CTRL-A	FF (0/12)	CTRL-L	ETB (1/7)	CTRL-W
STX (0/2)	CTRL-B	CR (0/13)	CTRL-M	CAN (1/8)	CTRL-X
ETX (0/3)	CTRL-C	SO (0/14)	CTRL-N	EM (1/9)	CTRL-Y
EOT (0/4)	CTRL-D	SI (0/15)	CTRL-O	SUB (1/10)	CTRL-Z
ENQ (0/5)	CTRL-E	DLE (1/0)	CTRL-P	ESC (1/11)	CTRL-[
ACK (0/6)	CTRL-F	DC1 (1/1)	CTRL-Q	FS (1/12)	CTRL-\
BEL (0/7)	CTRL-G	DC2 (1/2)	CTRL-R	GS (1/13)	CTRL-]
BS (0/8)	CTRL-H	DC3 (1/3)	CTRL-S	RS (1/14)	CTRL-^
HT (0/9)	CTRL-I	DC4 (1/4)	CTRL-T	US (1/15)	CTRL-_ _
LF (0/10)	CTRL-J	NAK (1/5)	CTRL-U		

OTHER KEYS. The remaining keys on the Main keypad generate terminal control functions as tabulated and described below. All are programmable at all levels.

KEY	UNSHIFTED	SHIFTED
ESC	ESC (1/11)	CSI (ESC [)
TAB	HT (0/9)	CBT (ESC [ Z)
PAUSE	Local string	Local string
BREAK	Short break	Long break
LINE FEED	LF (0/10)	Same as unshifted
DEL	DEL (7/15)	Same as unshifted
RETURN	CR (0/13)	Same as unshifted
BACKSPACE	BS (0/8)	Same as unshifted
SCROLL	Alt shift	None
ZOOM	Alt shift	None
S1-S6	None	None

TAB. Moves the cursor forward to the next tab stop (unshifted), or backward to the last tab stop (shifted).

PAUSE. If zAXM (Auto Xoff/Xon Mode) is set, permits start/stop of displayed data from the host; see Display section, Flow Control of Received Data.

BREAK. Forces the outgoing data line to its Space level for about 1/4 second (unshifted) or 3-1/2 seconds (shifted).

LINE FEED. Moves the cursor to the same column in the next line, or to the beginning of the next line, depending on the setting of LNM (LF New Line Mode).

DEL. The DEL character is ignored, or displayed, depending on the setting of zDDM (DEL-character Display Mode).

RETURN. Moves the cursor to the beginning of the active line, or to the

beginning of the next line, depending on the setting of zCNM (CR New Line Mode). If zRLM (Return Key CRLF Mode) is set, the key generates CRLF (instead of just CR).

BACKSPACE. Moves the cursor left one. If zDBM (Destructive Backspace Mode) is set, it also erases the character at that position.

SCROLL & ZOOM. If the operator's window is smaller than the page size set by the host, these keys permit viewing the full page by either scrolling the window over the page, or by zooming more of the page into the window. The keys are used in conjunction with the ARROW keys on the Control keypad; see Display section, Zooming the Display and Moving the Window.

S1-S6. If zMKM (Meta Key Mode) is set, S1 acts as a Meta shift key; see zMKM. Otherwise, these keys generate no codes and perform no internal functions unless programmed by the user.

CONTROL KEYPAD

The Control pad consists of a block of 17 keys located to the right of the Main keypad. If zKCPM is reset, it provides for numeric data entry (unshifted) and terminal control functions (shifted). If zKCPM is set, it provides for terminal control functions (unshifted) and numeric data entry (shifted). The table below assumes zKCPM reset. All of these keys are programmable at all levels.

=====	=====	=====
KEY	UNSHIFTED	SHIFTED
=====	=====	=====
HOME/5	5 (3/5)	CUP (ESC [ H)
UP-ARROW/8	8 (3/8)	CUU (ESC [ A)
DN-ARROW/2	2 (3/2)	CUB (ESC [ B)
RT-ARROW/6	6 (3/6)	CUF (ESC [ C)
LT-ARROW/4	4 (3/4)	CUD (ESC [ D)
9	9 (3/9)	None
7	7 (3/7)	None
1	1 (3/1)	None
3	3 (3/3)	None
0	0 (3/0)	None
00	00 (3/0 3/0)	None
+	+ (2/11)	None
,	, (2/12)	None
-	- (2/13)	None
TAB	HT (0/9)	Same as unshifted
ENTER	CR (0/13)	Same as unshifted

HOME. Moves the cursor to the beginning of the page.

UP-ARROW. Moves the cursor up one line.

DN-ARROW. Moves the cursor down one line.

RT-ARROW. Moves the cursor right one column.

LT-ARROW. Moves the cursor left one column.

TAB. Moves the cursor forward to the next tab stop.

ENTER. Moves the cursor to the beginning of the active line, or to the beginning of the next line, depending on the setting of zCNM (CR New Line Mode).

## FUNCTION KEYPAD

The Function keypad contains 26 keys arranged in two groups above the Main and Control keypads. All of these keys are programmable at all levels.

=====	=====	=====
KEY	UNSHIFTED	SHIFTED
=====	=====	=====
RESET	RIS (ESC c)	Same as unshifted
SETUP	Local string	Same as unshifted
F1-18	None	None
ERASE	EL (ESC [ K)	ED (ESC [ J)
EDIT	zTI (ESC 6)	None
DELETE	DCH (ESC [ P)	DL (ESC [ M)
INSERT	ICH (ESC [ @)	IL (ESC [ L)
PRINT	MC (ESC [ i)	Local string
SEND	None	None

RESET. Resets the terminal to its power-on state.

SETUP. Puts the terminal into Setup mode, permitting review or alteration of the terminal's characteristics.

F1-18. These keys generate no codes and perform no internal functions unless programmed by the user.

ERASE. Erases from cursor to end of line (unshifted), or to end of page (shifted).

EDIT. Toggles the terminal into and out of Edit mode. Characters typed during Edit mode are inserted at the cursor position and push following characters to the right instead of overwriting them.

DELETE. Deletes the character at the cursor position and moves the following characters left to fill the space. With SHIFT depressed, deletes the entire line containing the cursor and moves the following lines up.

INSERT. Moves the character at the cursor position to the right, opening a space. With SHIFT depressed, moves the entire line containing the cursor down, opening an empty line.

PRINT. Transmits the displayed data to a local printer. With SHIFT depressed, toggles the printer into and out of copying the received data stream.

SEND. This key generates no code and performs no internal function unless programmed by the user.

CONTENTS	Page
DISPLAY FEATURES .....	1
MODES THAT AFFECT THE DISPLAY .....	1
CONTROLS THAT AFFECT THE DISPLAY .....	2
DISPLAY DEFINITIONS .....	3
WRITING THE DISPLAY .....	4
FLOW CONTROL OF RECEIVED DATA .....	5
ZOOMING THE DISPLAY .....	6
MOVING THE WINDOW (w/ Cursor) .....	7
MOVING THE WINDOW (w/o Cursor) .....	7
USING THE HOST AREAS .....	8
SETTING UP MULTIPLE PAGES .....	9
SETTING UP MULTIPLE WINDOWS .....	9
USING THE ALTERNATE CURSOR .....	10
HIGHLIGHTING THE DISPLAY .....	11
USING THE ALTERNATE CHARACTER SETS .....	12

## DISPLAY FEATURES

The XL Series provides an exceptionally versatile display facility. Its modes and controls permit the user to tailor the display for maximum convenience, efficiency, and individual comfort. The use of these modes and controls is described and illustrated in this section.

### MODES THAT AFFECT THE DISPLAY

The following modes affect your use of the display. Their settings can be reviewed, and changed as desired, on the Setup lines; see Setup section. The Setup line letter, and mode number, are shown in parentheses following the name. Detailed descriptions of these modes can be found in the Appendix.

zBKCM BLOCK CURSOR MODE (A,31). Selects an underline or block cursor symbol.

zBNCM BLINKING CURSOR MODE (A,32). Causes the cursor symbol to blink.

zAPM AUTO PAUSE MODE (A,38). Permits the operator to regulate the flow of data onto the Screen with the PAUSE key (if the host supports an Xon/Xoff protocol; see zAXM).

zSSM SLOW SCROLL MODE (A,39). Regulates the flow of data onto the Screen to a fixed rate of about 4 lines/second (if the host supports an Xon/Xoff protocol).

zIVM INVERSE VIDEO MODE (A, 59). For Guru only, enables inverse video display, ie, 'black' characters on a 'white' screen.

zAXM AUTO XON/XOFF MODE (B,37). Permits start/stop codes to be sent to the host to regulate the flow of data to the terminal.

zICM INVISIBLE CURSOR MODE (D,56). Inhibits display of the cursor symbol. Useful when using programs that generate their own cursor (such as

spread-sheets). This mode cannot be 'saved' in the set (inhibit) state.

zWFM WRAP FORWARD MODE (D,33). Permits the cursor to wrap forward at end of line (to the beginning of the next line).

zDDM DEL-CHARACTER DISPLAY MODE (D,35). Permits display of the DEL-character (for systems that don't use DEL as a pad, or time-out, character).

zSPM SCROLL-PAGE MODE (D,36). Inhibits additional data entry at end of Page.

LNM LF NEW LINE MODE (D,20). Selects CTRL-J (LF) as the new-line code (instead of CRLF).

zCNM CR NEW LINE MODE (D,55). Selects CTRL-M (CR) as the new-line code (instead of CRLF).

zACM ALTERNATE CURSOR MODE (-,51). Enables an alternate (aka blind) cursor. The alternate cursor may write anywhere in Display Memory, whereas the primary cursor is limited to the (working) Page.

#### CONTROLS THAT AFFECT THE DISPLAY

The following controls affect the display. The code sequence (that executes them) is shown in parentheses following the name. The code sequence may be generated by the host, typed from the keyboard, or programmed into a function key; see Programmable Strings section. Detailed descriptions of these controls can be found in the Appendix.

zTFC TOGGLE FAST-BLINK CURSOR (ESC 5). Toggles the cursor symbol between its 'normal' display (as selected by zBKCM and zBKNM) and a fast-blinking block, for operator attention or warning.

zSDP SET DISPLAY PARAMETERS (ESC [ Pn..Pn p). Partitions both the Display Memory and Screen according to the parameters, Pn; see Display Definitions below. This control encompasses functions variously known as line lock, memory lock, split screen, etc.

SD SCROLL DOWN (ESC [ Pn T). Moves the displayed text down (or the Window up) the number of lines specified by Pn (if Window is smaller than Page; see Display Definitions below).

SU SCROLL UP (ESC [ Pn S). Moves the displayed text up (or the Window down) the number of lines specified by Pn (if Window is smaller than Page).

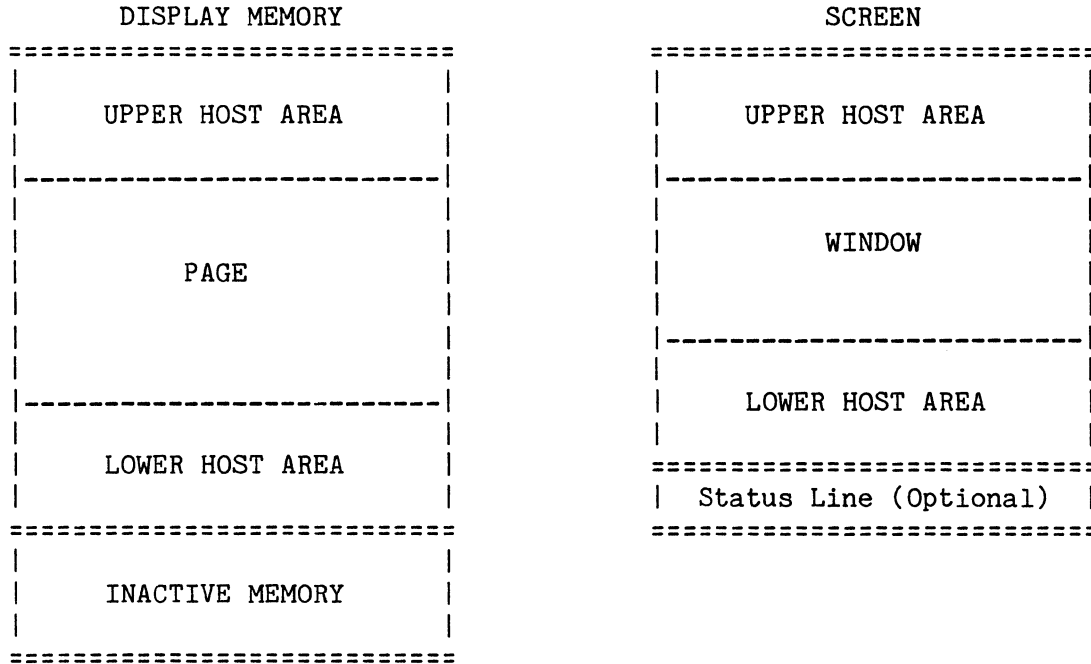
SGR SET GRAPHIC RENDITION (ESC [ Ps..Ps m). Selects the graphic rendition (aka highlights, visual attributes, etc.) with which all following characters received will be displayed; see Highlighting the Display, below.

zCGR CHANGE GRAPHIC RENDITION (ESC 9). Changes the graphic rendition of all following characters in the Page to that selected by the last SGR con-

trol.

DISPLAY DEFINITIONS

Certain terms are used in describing the terminal's display features. They are shown in the diagrams below, and defined following, to clarify their usage.



DISPLAY MEMORY. That portion of memory that is available to hold the results of keyboard and data transfer operations. The Genie+ and Ambassador provide 60 lines of Display Memory. The Guru provides up to 255 lines. Display Memory contains the following segments (starting from Display Memory address line 1): Upper Host Area, Page, Lower Host Area, and Inactive Memory.

ACTIVE MEMORY. That portion of Display Memory that is assigned to hold the results of subsequent keyboard and data transfer operations. It may be set to a specified number of lines with the 1st parameter of the zSDP (Set Display Parameters) control.

INACTIVE MEMORY. The remaining (ie, unassigned) portion of Display Memory.

NOTE: The zSDP control does not affect the contents of Display Memory; only the partitioning. Any data that was partitioned into Inactive Memory by a zSDP control may be retrieved by another zSDP control partitioning it into Active Memory.

UPPER HOST AREA. The upper-most portion of Active Memory, declared to be outside of the operator's working Page. It may be set to a specified number of lines with the 2nd parameter of the zSDP control.

LOWER HOST AREA. The lower-most portion of Active Memory, declared to be outside of the operator's working Page. It may be set to a specified number of lines with the 3rd parameter of the zSDP control.

**PAGE.** The portion of Active Memory between the Upper and Lower Host Areas. The Page is the only portion of Display Memory that may be accessed, written, or affected (by erase/edit operations) with the primary cursor.

**SCREEN.** The visible portion of Active Memory (ie, what's seen on the CRT). It may be set to a specified number of lines with the 4th parameter of the zSDP control. The Screen contains the following segments (starting from the top of the CRT): Upper Host Area, Window, Lower Host Area, and an optional Status Line.

**WINDOW.** The portion of the Screen between the Upper and Lower Host Areas. The Host Areas, and optional Status Line, are displayed in their entirety. What's left over is called the Window. The Window is the only portion of the Screen that is affected by scroll and zoom operations.

**STATUS LINE.** A line may be brought up to the bottom of the Screen to display terminal setups or to monitor received data; see Setup section. The content of this line is not stored in Display Memory (ie, it does not use up one of the Display Memory lines). When the line is brought up, it displaces one line from the Window. When the line is removed, the displaced line returns to the Window.

The following definitions apply only to the Guru. The screen and memory width of the Guru are user-selectable whereas, for the Genie+ and Ambassador, they are fixed at 80 columns.

**SCREEN WIDTH.** The number of columns of Active Memory displayed on the screen. It may be set with the 5th parameter of the zSDP control.

**ACTIVE MEMORY WIDTH.** The number of columns of Active Memory. It may be set with the 6th parameter of the zSDP control. If this parameter is changed, the Display Memory is cleared, as well as being re-partitioned.

#### WRITING THE DISPLAY

The cursor marks the active position in the Page, ie, the position at which the next character received will be displayed. The address space (for cursor positioning) is relative to the Page. Line 1 is always the top line of the Page. Line 1, column 1 (address 1,1) is the upper left corner of the Page (corresponding to cursor at Home). The cursor is automatically positioned to Home at power-on (with the Display Memory erased), and by any zSDP control (with the Display Memory unchanged).

As each character is received, it is displayed, and the cursor moves to the next column on the line. A new-line control terminates the line and moves the cursor to the 1st column of the next line. New-line controls are NEL, and CRLF (if LNM and zCNM are reset) or CR (zCNM set) or LF (LNM set). If no new-line control is received, the cursor wraps from the last column to the 1st column of the next line (zWFM set), or remains in the last column and is overwritten by subsequent characters (zWFM reset).

If the Window is smaller than the Page, when the cursor reaches the bottom line of the Window, a new-line control (or cursor wrap) moves the Window down one line to retain the cursor in the Window.



When the cursor reaches the bottom line of the Page, a new-line control (or cursor wrap) moves all lines in the Page up one line. The (former) content of the top line of the Page is lost. The bottom line of the Page is erased, and the cursor positioned in its 1st column. Note: If zSPM is set, the terminal is said to be in Page mode, and this 'memory scroll' action is inhibited.

Thereafter (unless the cursor is re-positioned), all new lines received are written into the bottom line of the Page, the older lines scrolling upward in the Page, the oldest lines being lost off the top of the Page.

This action is viewed through the Window, which is displaying the bottom-most segment of the Page. More data can be 'zoomed' into the Window at any time, without interrupting the data flow; see Zooming the Display. The data flow can be suspended at any time; see Flow Control of Received Data. When the data flow is suspended, the Page can be viewed by moving the Window (either with or without moving the cursor); see Moving the Window.

#### FLOW CONTROL OF RECEIVED DATA

If the host supports an Xon/Xoff protocol (see zAXM), the terminal provides a powerful flow control capability that permits the operator to regulate the rate of display of incoming data. This includes an ability to start and stop the data flow (with the PAUSE key), to step the data onto the screen a line (or several lines) at a time (zAPM, Auto Pause Mode), and to regulate the display to a fixed line rate (zSSM, Slow Scroll Mode),

**PAUSE KEY.** Incoming data is normally displayed as fast as it is received. At higher baud rates, especially with short lines, the data may pass through the window much too rapidly to be viewed. When this happens, simply hit the PAUSE key, to 'freeze' the display. If you missed something, zoom (or scroll) up to see it. Hit the PAUSE key again, and the display continues.

**AUTO PAUSE.** The normal PAUSE key operation is useful for scanning a file, or getting quickly to a part you want, but to read the file, go into Auto Pause Mode. Hit CTRL-PAUSE (instead of just PAUSE). Again the display freezes but, this time, typing the PAUSE key just steps the next line onto the screen. Type PAUSE again, and the next line appears. You can step through the data, a line at a time, by repeatedly typing the PAUSE key. Note: The PAUSE key can be programmed to repeat; see Programmable Strings section. This is useful in Auto Pause Mode, but can be confusing in the normal operation of the PAUSE key.

Alternatively, you can step through the file a window at a time by typing SHIFT-PAUSE. This lets in (two lines less than) a window-full of new data. The bottom lines of the previous window are retained as the top lines of the new window for context, and assurance that nothing was missed. When you've finished reading what you wished, hit CTRL-SHIFT-PAUSE which resets Auto Pause Mode, and returns you to normal PAUSE key operation. Summary:

- CTRL-PAUSE. Turns on Auto Pause.
- PAUSE. Admits a new line of data to the screen.
- SHIFT-PAUSE. Admits a new window of data to the screen.
- CTRL-SHIFT-PAUSE. Turns off Auto Pause.

The Pause functions are accomplished with local strings, and may be freely

rearranged to other key/level positions; see Programmable Strings section.

**SLOW SCROLL.** Before initiating display of the file, or any time you're Paused, you can set Slow Scroll Mode (Setup line A). This regulates the display of incoming data to a fixed line rate (of about 4 lines/second). This is particularly useful if the file consists mostly of short lines.

**SETUP KEY.** The SETUP key also suspends the incoming data flow, to let you change settings in the middle of a listing (eg, to set and reset Slow Scroll Mode). Typing SETUP again (the normal Setup exit) re-starts it. Exiting with the letter M (Monitor), re-starts it with a Data Monitor line at the bottom of the screen (that permits viewing the actual codes, including control codes, being received; see Diagnostics section).

### ZOOMING THE DISPLAY

The XL Series' unique 'zoom' capability permits either the operator or the host to control the number of lines actually displayed in the Window.

**OPERATOR CONTROL.** The operator controls the Window by using the ZOOM key in conjunction with the ARROW keys on the Control keypad. When ZOOM is depressed:

- DN-ARROW removes lines from the window.
- UP-ARROW adds lines into the window.
- RT-ARROW removes columns from the window (Guru only).
- LT-ARROW adds columns into the window (Guru only).

As lines (or columns) are removed, both the interline (or intercolumn) spacing and the character size increase. The operator may adjust the Screen for individual comfort and ease of viewing, yet instantly zoom more data onto the Screen for overall context when desired. These actions are transparent to the host; they do not affect the transfer of data or Display Memory addresses. The Zoom functions are accomplished with local strings, and may be freely rearranged to other key/level positions; see Programmable Strings section.

**HOST CONTROL.** The host controls the Window size with the zSDP control. The Screen parameter (Pn4) specifies the number of lines to be displayed on the Screen, The number of lines displayed in the Window is simply that number less any Host Area lines presently displayed. The Screen parameter may be omitted if it is desired to re-partition Display Memory without changing the number of lines displayed. Similarly, omitting the Active Memory parameter (Pn1), permits the Screen size to be changed (ie, 'zoomed') without affecting the Display Memory partitions. Examples:

```
ESC [ 30 ; ; ; 30 p
```

Partitions Display Memory for a 30-line Page. The Screen (and Window) displays 30 lines.

```
ESC [ 60 p
```

Increases Display Memory to 60 lines. The Screen (and Window) remains unchanged.

```
ESC [ ; ; ; 60 p
```

Changes the Screen size to 60 lines (Ambassador and Guru). The Display

Memory partition remains unchanged.

For the Guru, the Screen Width parameter (Pn5) specifies the number of columns to be displayed. If the parameter is omitted, its value is not changed. If the Active Memory parameter (Pn1) is omitted, the number of columns may be changed without affecting the value of other parameters. Examples:

```
ESC [ 66 ; ; ; 66 ; 132 ; 132 p
```

Partitions Display Memory for a 132-character by 66-line Page. The Screen (and Window) displays 132 characters by 66 lines.

```
ESC [ 132 p
```

Increases Display memory to 132 lines. The Screen (and Window) remain unchanged.

```
ESC [ ; ; ; 40 ; 80 p
```

Changes the Screen size to 80 characters by 40 lines. The Display Memory partition remains unchanged.

#### MOVING THE WINDOW (w/ CURSOR)

If the Window is set to be smaller than the Page, the Window moves naturally with the cursor. Any attempt, by the operator or the host, to move the cursor outside of the Window causes the Window to move within the Page so as to retain the cursor symbol within the Window. Note: The host may write outside of the Window, whenever desired, by using the alternate cursor. The Window position is not affected by alternate cursor movements; see zACM (Alternate Cursor Mode).

**OPERATOR CONTROL.** The operator will normally move the Window with the HOME and cursor ARROW keys on the Control keypad. The cursor keys are non-destructive; no data can be lost through their use.

**HOST CONTROL.** The host will normally position the Window with the CUP (Cursor Position) control. Examples:

```
ESC [ H ESC [ Pn H           Pn >= Window size
```

This string positions the Window such that the line specified by the parameter, Pn, is displayed in the bottom line of the Window. Note: The Window may not be positioned above the top line of the Page.

```
ESC [ 255 H ESC [ Pn H       Pn <= Page size - Window size +1
```

This string positions the Window such that the line specified by the parameter, Pn, is displayed in the top line of the Window. Note: The Window may not be positioned below the bottom line of the Page.

#### MOVING THE WINDOW (w/o CURSOR)

The Window may also be moved without moving the cursor, ie, without changing its position within the Page. However, if the Window is moved so that it no longer contains the cursor symbol, any attempt at entry by the operator, or cursor positioning by the host, will automatically re-position the Window to contain the cursor symbol.

OPERATOR CONTROL. The operator may move the Window by using SCROLL key in conjunction with the ARROW keys on the Control keypad. When SCROLL is depressed:

- DN-ARROW moves the window down toward end of page.
- UP-ARROW moves the window up toward top of page.
- RT-ARROW moves the window right toward end of line.
- LT-ARROW moves the window left toward beginning of line.

When SHIFT-SCROLL is depressed, the window moves in the direction indicated above, but in (almost) window-full steps. The Scroll functions are accomplished with local strings, and may be freely rearranged to other key/level positions; see Programmable Strings section.

HOST CONTROL. The host may use the controls SU (Scroll Up) and SD (Scroll Down) to move the Window. Scroll up moves the text up (corresponding to moving the Window down). Scroll down moves the text down (corresponding to moving the Window up). In both cases, the text moves by the number of lines specified by the parameter. If there are not that many lines remaining to the top (or bottom) of page, the Window moves to the top (or bottom) of Page and stops.

#### USING THE HOST AREAS

The terminal's Host Area capability permits one or more lines to be reserved at the top and/or the bottom of the Screen for display of system status, operator menu data, table headings, etc. These lines are 'locked' on the Screen, ie, they remain always displayed. Data in the Window scrolls behind them. They cannot be accessed, written, or affected (by erase/edit operations) with the primary cursor (ie, the operator's normal cursor).

Host areas are set (reserved) with the zSDP control (2nd and 3rd parameters). They may be reserved first, and then written into with the alternate cursor, or the selected lines may be written into first (with the primary cursor), and then reserved. Recall that the zSDP control does not affect the contents of Display Memory -- only its partitioning. If the terminal is used in Monitor SRM (ie, local entry), the host may freely write these areas (using the alternate cursor) without disrupting, or interfering in any way with, the operator's local entry. Examples:

```
ESC [ 60 ; 2 p
```

Reserves a 2-line Host Area at the top of the Screen. This might be used, for example, to capture the heading from a long file so that it remains displayed while you scroll through the rest of the file. With the file heading displayed at the top of the screen, type this control to lock the heading on the screen. As with all XL controls, if this were a frequent action, you could program the control into a function key, and execute it with a single keystroke.

```
ESC [ 60 ; 3 ; 5 p
```

Reserves a 3-line Host Area at the top of the Screen and a 5-line Host Area at the bottom. The displayed data scrolls between them.

```
ESC [ 60 p
```

Removes all Host Areas. The data that was in the Host Areas is still

displayed, but is no longer protected from primary cursor operations.

#### SETTING UP MULTIPLE PAGES

The Display Memory may be partitioned and used as independent pages with the zSDP, zPSH, and zPOP controls. The examples below partition the memory into 2 pages, with a 'swap page' string. Many other paging operations are possible. These examples merely illustrate the technique (and the versatility of the terminal's control set).

```
ESC [ 60 p ESC [ 30 t ESC [ 30 p
```

This string partitions the memory into two 30-line pages that may be used independently. Each time the string is executed, the pages are swapped, ie, the active page becomes inactive, and the inactive page becomes active (and displayed). Keyboard and data transfer operations affect only the active page. The inactive page cannot be accessed, written, or affected (by erase/edit operations) with the primary cursor. The inactive page may be written by the host, if desired, with the alternate cursor; see zACM. The string may be sent by the host, or programmed into a function key for local execution (ie, a SWAP PAGE key; see below).

```
ESC [ 60 ; 2 p ESC [ 29 t ESC [ 29 ; 2 p
```

This string partitions the memory into two 29-line pages with a 2-line Upper Host Area that is always displayed regardless of which page is active.

```
ESC [ 60 ; 2 p ESC [ 28 t ESC [ 29 H ESC [ 28 s ESC [ 32 ; 2 ; 2 p
```

This string partitions the memory into two 28-line pages with 2-line Upper and Lower Host Areas that are always displayed regardless of which page is active.

```
ESC P > | 122;0;2 | ~[ [ 60 p ~[ [ 30 t ~[ [ 30 p ESC \
```

This string loads the first example into F1 (making F1 a SWAP PAGE key).

#### SETTING UP MULTIPLE WINDOWS

In the examples above, only the active page is displayed. It is also possible to have multiple pages displayed, only one of which is active. In the first example above, if we had terminated the string with ESC [ 30 ; ; ; 60 p (instead of ESC [ 30 p), we would have found both pages displayed. In the example below, we divide the screen into three 20-line windows with strings that select which window is active, ie, contains the cursor. The strings may be sent by the host, or programmed into function keys for local execution.

```
ESC [ 60 ; ; 40 ; 60 p      -Selects window #1 (Lines 1-20)
ESC [ 60 ; 20 ; 20 ; 60 p  -Selects window #2 (Lines 21-40)
ESC [ 60 ; 40 ; ; 60 p     -Selects window #3 (Lines 41-60)
```

The disadvantage of the example above is that the windows have no visual separation. In the example below, we divide the screen into three 19-line windows with a separator line (of dashes).

```
ESC [ 60 p ESC [ J          -Initializes the screen
- ESC [ 79 b                -Fills Line 1 with dashes
```

```

ESC [ 21 H - ESC [ 79 b      -Fills Line 21 with dashes
ESC [ 41 H - ESC [ 79 b      -Fills Line 41 with dashes
ESC [ 60 ; 1 ; 40 ; 60 p     -Selects window #1 (Lines 2-20)
ESC [ 60 ; 21 ; 20 ; 60 p    -Selects window #2 (Lines 22-40)
ESC [ 60 ; 41 ; ; 60 p       -Selects window #3 (Lines 42-60)

```

The following string loads the first four strings into F14, and the last three strings into F1, 2, and 3. F14 then initializes the screen, and F1, 2, and 3 select the desired window. Note: The string is shown spaced out and on separate lines for clarity only. The spaces should not be included in the string; the new-lines are optional.

```

ESC P >
| 135;0;2 | ~[ [ 60 p ~[ [ J
           - ~[ 79 b
           ~[ [ 21 H - ~[ [ 79 b
           ~[ [ 41 H - ~[ [ 79 b
| 122     | ~[ [ 60 ; 1 ; 40 ; 60 p
|         | ~[ [ 60 ; 21 ; 20 ; 60 p
|         | ~[ [ 60 ; 41 ; ; 60 p
ESC \

```

#### USING THE ALTERNATE CURSOR

The terminal is equipped with two independent cursors -- primary and alternate. Only one or the other may be in use at any given time (except, in Monitor SRM, local entries from the keyboard use only the primary cursor, whereas the host may use either). The alternate cursor is selected by going into Alternate Cursor Mode. This mode is not available on the Setup lines. It can be changed only with the SM/RM (Set/Reset Mode) controls. The string, ESC [ > 51 h, sets it; the string, ESC [ > 51 l, resets it.

The significant characteristics of the primary cursor are:

- 1) Its position is confined to the Page.
- 2) Its position (in the Window) is noted with a visible cursor symbol.
- 3) Erase and edit operations using the primary cursor affect only the contents of the Page.

The significant characteristics of the alternate cursor are:

- 1) It can access all of Display Memory, including the Host Areas and Inactive Memory.
- 2) Its position is not visible, even if writing into the Window.
- 3) Erase and edit operations using the alternate cursor affect all of Display Memory.
- 4) Address 1 of the alternate cursor points to the top line of Display Memory. Address 1 of the primary cursor points to the top line of the Page. Note that to point to line N in the operator's Page with the alternate cursor, it is necessary to use a parameter value of N + M, where M is

the number of lines of Upper Host Area. This is required because the ANSI standard does not provide for the transfer of negative parameters.

#### HIGHLIGHTING THE DISPLAY

The terminal provides five distinct renditions for highlighting the display. The renditions may be used to highlight a particular group of characters (to call attention to them or to visually differentiate them from others on the screen), or it may be used to convey additional information about the characters (eg, type style). This information may be transmitted to the host; see zGRTM (Graphic Rendition Transfer Mode).

OPERATOR CONTROL. No keys are assigned to the control of graphic rendition. If desired, the programmable keys may be used. Example:

```
ESC P > | 122;0;2 | ~[ [ m | | ~[ [ 4 m | | ~[ [ 7 m ESC \
This string makes F1 a 'Start Normal' key, F2 a 'Start Underscore' key,
and F3 a 'Start Reverse' key; see Programmable Strings section.
```

HOST CONTROL. The host uses the SGR control to control the rendition with which subsequent characters are written.

SGR SET GRAPHIC RENDITION (ESC [ Ps..Ps m). Sets a Graphic Rendition (GR) register according to the parameters. The contents of this register are appended to all characters subsequently entered or received, causing them to be displayed as Normal (Ps= 0), Bold (Ps= 1), Underscored (Ps= 4), Blinking (Ps= 5), Reverse (Ps= 7), or Concealed (Ps= 8), in any combination. Note: The Guru does not support the Concealed rendition.

Bold is simply brighter than Normal. The difference is adjustable; see Checkout & Installation section, Initial Checkout). Reverse gives black characters on a 'white' background. Concealed causes the characters to be displayed as Spaces (2/0). Concealed characters are stored correctly, and may be transmitted or printed; they're just not visible. Examples:

ESC [ m	Start Normal
ESC [ 1 m	Start Bold
ESC [ 4 m	Start Underscored
ESC [ 5 m	Start Blinking
ESC [ 7 m	Start Reverse
ESC [ 8 m	Start Concealed (black spaces)
ESC [ 7 ; 8 m	Start Concealed (white spaces)
ESC [ 1 ; 4 ; 7 m	Start Bold, Underscored, and Reverse

The renditions do not require character locations in Display Memory. They may be freely applied by individual character if desired. Also, The rendition is defined to be an attribute of the character being written, not of a character position in the Page. The rendition is simply part of the character. When the character is moved or erased, its rendition is moved or erased with it.

Unlike earlier generations of many terminals, the rendition carries no control attributes (eg, 'dim' is not associated with 'protected'). The ANSI control DAQ (Define Area Qualifications) is reserved for setting control attributes; see Forms section. The SGR control sets visual attributes, the DAQ control sets control attributes, and the user is free to associate them in any way

that may be desired.

The content of the GR-register is also appended to all Space codes (2/0) written or inserted by the Erase/Edit controls (see Editing Controls section), or inserted as the result of a memory scroll (see Writing the Display, above). The desired result follows naturally, whether working with a Normal or Reverse display. Note: Care is required when using a Reverse rendition to highlight a Normal display (or vice versa) and memory scroll is permitted (ie, zSPM reset). If the last character of the line is written Reverse, the terminal assumes the intention is to continue Reverse (as it would be if working with a Reverse display) and scrolls up the blank line in that rendition.

The host may use the zCGR control to change the rendition of characters already written.

zCGR CHANGE GRAPHIC RENDITION (ESC 9). Changes the graphic rendition of all following characters in the Active Area to that selected by the last SGR control. The Active Area is the Page at power-on; see Forms section. Example:

ESC [ 7 m ESC 9

This string, if given with the cursor in the Home position, changes the rendition of the Page to Reverse (ie, black characters on a 'white' background).

#### USING ALTERNATE CHARACTER SETS

The terminal provides three character sets: ASCII, UK, and Graphics. Any two may be selected at one time. The ASCII set is the standard character set defined by ANSI X3.4. The UK set replaces the character corresponding to code 2/3 (#) with the British pound sign. The Graphics set replaces the lower-case characters (codes 6/0 (`) through 7/14 (~)) with the symbols shown in the table below. Caution: These symbols are transmitted as 2/0 (Space) codes during send and print operations.

6/0	(')	Diamond	7/0	(p)	Scan line 3
6/1	(a)	Checkerboard	7/1	(q)	Scan line 5
6/2	(b)	Space	7/2	(r)	Scan line 7
6/3	(c)	Space	7/3	(s)	Scan line 9
6/4	(d)	Space	7/4	(t)	Left 'T'
6/5	(e)	Space	7/5	(u)	Right 'T'
6/6	(f)	Degree symbol	7/6	(v)	Bottom 'T'
6/7	(g)	Plus/minus	7/7	(w)	Top 'T'
6/8	(h)	Space	7/8	(x)	Vertical bar
6/9	(i)	Space	7/9	(y)	Less than or equal
6/10	(j)	Lower-right corner	7/10	(z)	Greater than or equal
6/11	(k)	Upper-right corner	7/11	{ }	Space
6/12	(l)	Upper-left corner	7/12		Not equal
6/13	(m)	Lower-left corner	7/13	{ }	British pound
6/14	(n)	Crossing lines	7/14	(~)	Space
6/15	(o)	Scan line 1			

The SCSG0 and SCSG1 controls determine which two (of the three) characters sets are selected. The SI and SO controls determine which one (of the two) are active. The terminal powers on with ASCII selected for both the G0 and G1



sets, and the G0 set active.

SCSG0 SELECT CHARACTER SET G0 (ESC ( Fc). Designates the G0 character set from one of the available sets depending on the final character Fc:

A (4/1) United Kingdom set  
 B (4/2) ASCII set  
 0 (3/0) Graphics set

SCSG1 SELECT CHARACTER SET G1 (ESC ) Fc). Designates the G1 character set from one of the available sets depending on the final character Fc:

A (4/1) United Kingdom set  
 B (4/2) ASCII set  
 0 (3/0) Graphics set

SI SHIFT IN (CTRL-O). Causes characters subsequently entered or received to be displayed from the G0 character set.

SO SHIFT OUT (CTRL-N). Causes characters subsequently entered or received to be displayed from the G1 character set.

The Graphics character set may be used to outline regions of screen. For example, the following string draws a box 21 characters wide by 7 lines high. Note: In this case, the spaces and new-lines should be included.

```
ESC ) 0 SO
lqqqqqqqqqwwqqqqqqqqk
x          x          x
x          x          x
tqqqqqqqqqqqqqqqqqqqu
x          x          x
x          x          x
mqqqqqqqqqvqqqqqqqqqj
SI
```



CONTENTS	Page
MODES THAT AFFECT THE CURSOR CONTROLS .....	1
MOVING THE CURSOR .....	1
READING THE CURSOR POSITION .....	3
SAVE/RESTORE CURSOR POSITION .....	3
USING COLUMNAR TABS .....	4
USING PAGE TABS .....	5

## MODES THAT AFFECT THE CURSOR CONTROLS

The following modes affect the terminal's response to some of the cursor controls. If the mode is not mentioned in the description of a control, the control is not affected by the mode. Detailed descriptions of these modes can be found in the Appendix.

**zWFM WRAP FORWARD MODE (D,33).** Permits the cursor to wrap forward at end of line (to beginning of next line).

**zWBM WRAP BACKWARD MODE (D,34).** Permits the cursor to wrap backward at beginning of line (to end of preceding line).

**zSPM SCROLL-PAGE MODE (D,36).** Inhibits additional data entry at end of Page.

**TSM TABULATION STOP MODE (D,18).** Determines whether columnar tab stops (TSM reset) or page tab stops (TSM set) are affected by the Tab Set/Clear controls.

**LNM LF NEW LINE MODE (D,20).** Selects CTRL-J (LF) as the new-line code (instead of CRLF).

**zCNM CR NEW LINE MODE (D,55).** Selects CTRL-M (CR) as the new-line code (instead of CRLF).

**zACM ALTERNATE CURSOR MODE (-,51).** Enables the alternate cursor. The primary cursor is confined to the (working) Page and is displayed; the alternate cursor is not.

## MOVING THE CURSOR

The terminal provides all of the ANSI standard cursor controls.

**OPERATOR CONTROL.** The operator moves the cursor with the HOME and ARROW keys on the Control keypad. HOME moves the cursor to the first character position in the Page. UP-ARROW moves the cursor up one line. DOWN-ARROW moves it down one line. RIGHT-ARROW moves it right one column. LEFT-ARROW moves it left one column. The ARROW keys repeat when held down, regardless of the setting of zKRM (Key Repeat Mode), unless specifically re-programmed to not repeat; see Programmable Strings section.

**HOST CONTROL.** The host moves the cursor with the following controls. All cursor controls (except those that reference zSPM) are non-destructive, ie,

they cannot affect the contents of the Page. Those that reference zSPM can move the contents of the Page, if zSPM is reset, losing one line from the Page. If zACM is set, they are non-destructive in all cases, ie, the alternate cursor cannot cause the data to scroll regardless of the setting of zSPM. Many of the controls take numeric parameters, Pn (up to a maximum value of 255). Illegal parameter values, ie, values that would move the cursor outside the boundaries of the Page, will typically move the cursor to the applicable boundary.

CR CARRIAGE RETURN (CR). Moves the cursor to beginning of the active line. If zCNM is set, it also does an implicit LF. If zCNM is set, zSPM and zACM are reset, and the cursor is in the bottom line of the Page, it does a scroll up of all data in the Page (losing the top line).

LF LINE FEED (LF). Moves the cursor down one line. If LNM is set, it also does an implicit CR. If zSPM and zACM are reset and the cursor is in the bottom line of the Page, it does a scroll up of all data in the Page (losing the top line).

BS BACKSPACE (BS). Moves the cursor left one column. If zDBM (Destructive Backspace Mode) is set, it also erases the character at that position. If zWBM is set, it is permitted to wrap to the preceding line.

CUU CURSOR UP (ESC [ Pn A). Moves the cursor up Pn lines. Examples:

ESC [ A	Moves the cursor up 1 line.
ESC [ 12 A	Moves the cursor up 12 lines.
ESC [ 99 A	Moves the cursor to top of Page (in the same column).

NOTE: In the last example, since the cursor cannot move 99 lines (assuming a 60-line page), it moves to the page boundary and stops.

CUD CURSOR DOWN (ESC [ Pn B). Moves the cursor down Pn lines.

CUF CURSOR FORWARD (ESC [ Pn C). Moves the cursor right Pn columns. If zWFM is set, it is permitted to wrap to following lines.

CUB CURSOR BACKWARD (ESC [ Pn D). Moves the cursor left Pn columns. If zWBM is set, it is permitted to wrap to preceding lines.

HPR HORIZONTAL POSITION RELATIVE (ESC [ Pn a). Same as CUF, except limited to the active line regardless of zWFM.

VPR VERTICAL POSITION RELATIVE (ESC [ Pn e). Same as CUD.

IND INDEX (ESC D). Moves the cursor down one line. If zSPM and zACM are reset and the cursor is in the bottom line of the Page, it does a scroll up of all data in the Page (losing the top line).

RI REVERSE INDEX (ESC M). Moves the cursor up one line. If zSPM and zACM are reset and the cursor is in the top line of the Page, it does a scroll down of all data in the Page (losing the bottom line).

NEL NEXT LINE (ESC E). Moves the cursor to the beginning of the next line. If zSPM and zACM are reset and the cursor is in the bottom line of

the Page, it does a scroll up of all data in the Page (losing the top line).

CNL CURSOR NEXT LINE (ESC [ Pn E). Moves the cursor down Pn lines to beginning of line.

CPL CURSOR PRECEDING LINE (ESC [ Pn F). Moves the cursor up Pn lines to beginning of line.

HPA HORIZONTAL POSITION ABSOLUTE (ESC [ Pn `). Positions the cursor to column Pn (in the same line).

VPA VERTICAL POSITION ABSOLUTE (ESC [ Pn d). Positions the cursor to line Pn (in the same column). If zACM is set, Pn is offset by the amount of any Upper Host Area setting, eg, with a 3-line Upper Host Area, Pn= 6 (zACM reset) and Pn= 9 (zACM set) both point to line 6 in the Page.

CHA CURSOR HORIZONTAL ABSOLUTE (ESC [ Pn G). Same as HPA.

CUP CURSOR POSITION (ESC [ Pn1 ; Pn2 H). Positions the cursor to line Pn1, column Pn2 (of the Page).

If zACM is set, Pn1 is offset by the amount of any Upper Host Area setting; see VPA above. Examples:

ESC [ H	Moves the cursor to line 1, column 1 (Home).
ESC [ 6 H	Moves the cursor to line 6, column 1.
ESC [ ; 12 H	Moves the cursor to line 1, column 12.
ESC [ 6 ; 12 H	Moves the cursor to line 6, column 12.
ESC [ 255 ; 255 H	Moves the cursor to end of Page.

HVP HORIZ & VERT POSITION (ESC [ Pn1 ; Pn2 f). Same as CUP.

## READING THE CURSOR POSITION

The following controls may be used to read the present cursor position. The position reported is that of the primary cursor only, regardless of the setting of zACM.

DSR DEVICE STATUS REPORT (ESC [ Ps n). This control, sent by the host with Ps= 6, requests the terminal to report its cursor position. The terminal responds in the form of a CPR control.

CPR CURSOR POSITION REPORT (ESC [ Pn1 ; Pn2 R). This control is returned to the host in response to the DSR control. It reports the cursor position in the same decimal form used by the CUP and HVP controls. Examples:

ESC [ 01 ; 01 R	Cursor position = line 1, column 1 (Home)
ESC [ 06 ; 12 R	Cursor position = line 6, column 12

## SAVE/RESTORE CURSOR POSITION

The following controls may be used to save and restore the present cursor position (and graphic rendition setting). Either the primary or alternate cur-

sor values may be saved. However, only one set of values may be saved at a time. The 'saved' values may be restored to either cursor.

zSC SAVE CURSOR POSITION (ESC 7). Saves the cursor position (and rendition setting) in a temporary buffer for later recall. If zACM is reset, it saves the primary cursor position (and associated rendition). If zACM is set, it saves the alternate cursor position (and associated rendition).

zRC RESTORE CURSOR POSITION (ESC 8). Returns the cursor position (and rendition setting) to its 'saved' values. If zACM is reset, it restores the values to the primary cursor. If zACM is set, it restores the values to the alternate cursor.

#### Example:

```
ESC [ > 51 h ESC 7 ESC [ > 51 1 ESC 8
```

This string saves the alternate cursor position (and rendition setting) and restores it to the primary cursor. The primary cursor moves to the alternate position and subsequent entries are rendered with the alternate rendition.

#### USING COLUMNAR TABS

The terminal provides both columnar and full-page tab capabilities. A columnar tab stop applies to an entire column in the Page. A page tab stop applies to a single character position in the Page. Either may be used, but not at the same time. Setting a page tab stop suspends the effect of columnar tab stops; see Using Page Tabs below.

Columnar tabs are the more generally used, and usually at a fixed spacing (8 or 10 columns). The XL Series initialize with an 8-column spacing, ie, with tab stops at columns 1, 9, 17, 25, 33, 41, 49, 57, 65, and 73.

NOTE: If your facility uses a 10-column spacing, the string  
 ESC [ 5 W CR ESC H ESC [ 10 C .... ESC H ESC [ 10 C ESC H  
 may be used to re-initialize the settings. This string may be  
 programmed into the Power-on string so that the terminal comes  
 up with the 10-column settings each time it is powered on.

OPERATOR CONTROL. The operator tabs and backtabs with the TAB key. TAB moves the cursor right to the next tab stop. SHIFT-TAB moves the cursor left to the preceding tab stop. The TAB key repeats when held down, regardless of the setting of zKRM (Key Repeat Mode), unless specifically re-programmed to not repeat. A tab-with-justify function is available for entering right-justified data (eg, columns of numbers; see HTJ below).

The default key programming does not provide for operator setting and clearing of tab stops. If this is desired, it is easily added:

```
ESC P > | 122;0 | ~[ H | | ~[ [ g | | ~[ [ 3 g ESC \
```

This string loads the F1, 2, and 3 keys with Set tab, Clear tab, and Clear all tab controls; see below.

HOST CONTROL. The host sets and clears tab stops with the following controls:

HTS HORIZONTAL TABULATION SET (ESC H). Sets a tab stop at the present cursor position. If TSM is reset, it sets a columnar tab stop, otherwise page.

TBC TABULATION CLEAR (ESC [ Ps g). Clears one or more tab stops according to the selective parameter, Ps. If TSM is reset, it clears columnar tab stops; otherwise page. Examples:

ESC [ g	Clears tab stop at the cursor
ESC [ 2 g	Clears all tab stops in the line
ESC [ 3 g	Clears all tab stops in the Page

CTC CURSOR TABULATION CONTROL (ESC [ Ps W). Alternative control to HTS and TBC; see Appendix.

The host tabs and backtabs with the following controls:

HT HORIZONTAL TABULATION (HT). Moves the cursor right to the next tab stop. If zWFM is set, it is permitted to wrap to the next line. If zSPM and zACM are reset and the cursor is at the last tab stop in the Page, it does a scroll up of all data in the Page (losing the top line).

HTJ HORIZONTAL TAB WITH JUSTIFY (ESC I). Same as HT except that the entry is right-justified before moving to the next tab stop. The control may be loaded into a programmable key for operator use when desired.

CHT CURSOR HORIZONTAL TABULATION (ESC [ Pn I). Moves the cursor right Pn tab stops. If zWFM is set, it is permitted to wrap to following lines.

CBT CURSOR BACKWARD TABULATION (ESC [ Pn Z). Moves the cursor left Pn tab stops. If zWBM is set, it is permitted to wrap to preceding lines.

## USING PAGE TABS

Page tab stops apply to single-character positions in the Page. They may be used to facilitate entry into forms (see Forms section) or display of page-like data. Note: They are typically used with zSPM in the set state (ie, with memory scroll inhibited). A memory scroll moves the contents of the Page, but not the tab stops (by definition). Unless the tab stops are aligned in columns, an inadvertent memory scroll would move the data away from the tab stops.

Setting any page tab stop suspends the effect of columnar tab stops. Only the page tab stops are active. This is true until the Page is purposefully cleared with an RIS control, or by erasing the entire page with ERM set, or by re-defining the Page with the zSDP control. Simply clearing the last tab stop does not re-institute columnar tabs. This permits the tab clear controls to avoid the time-consuming task of searching for remaining tab stops or DAQ qualifications. When the Page is so cleared or re-defined, the columnar tab stops become active again at their previous positions.

Page tab stops are set and cleared with the same controls described above. TSM determines whether they set columnar or page tab stops. Likewise, the same tab and backtab keys and controls are used, moving the cursor according to whichever tab stops are active.





CONTENTS	Page
MODES THAT AFFECT THE EDITING CONTROLS .....	1
DEFINITIONS .....	1
ERASING THE DISPLAY .....	2
EDITING THE DISPLAY .....	3
USING THE zSDP CONTROL IN EDITING .....	4

## MODES THAT AFFECT THE EDITING CONTROLS

The mode ERM affects the terminal's response to erase controls:

ERM ERASURE MODE (D,6). Permits the erasure of characters regardless of their protected status; see Forms section.

The mode IRM is an editing mode:

IRM INSERTION-REPLACEMENT MODE (-,4). Permits character entries to be inserted into the Page, pushing the characters presently displayed to the right (instead of overwriting, or replacing, them).

Two controls set internal registers whose settings affect the terminal's response to the controls:

SGR SET GRAPHIC RENDITION (ESC [ Ps..Ps m). Sets a Graphic Rendition (GR) register according to the parameters; see Display section, Highlighting the Display. The contents of this register are appended to all characters subsequently entered or received. It is also appended to all Space (2/0) codes written by the Erase controls, inserted by the Edit controls, or inserted as a result of memory scroll; see zSPM (Scroll-Page Mode).

SEE SELECT EDITING EXTENT (ESC [ Ps Q). Sets an Editing Extent (EE) register according to the parameter. The contents of this register limit the effect of the character edit controls, DCH and ICH, to Page, Line, Field, or Area; see Definitions below.

## DEFINITIONS

Certain terms are used in describing the terminal's editing features. They are defined below to clarify their usage. These definitions follow, and are consistent with, the ANSI definitions.

PAGE. The portion of Active Memory between the Upper and Lower Host Areas; see Display section. The Page is the only portion of Display Memory that can be accessed, written, or affected (by erase/edit operations) with the primary cursor.

LINE. A portion of the Page having a common vertical position.

FIELD. A portion of the Page bounded by tab stops (either columnar or page). A Field contains the character position at the beginning tab stop and excludes the character position at the trailing tab stop. The terminal initializes to ten 8-character Fields per Line; see Cursor Controls

section, Using Columnar Tabs.

**QUALIFIED AREA.** A portion of the Page bounded by area qualifiers; see Forms section. An Area includes the character position at the beginning area qualifier and excludes the character position at the trailing area qualifier. The terminal initializes to one Area (equal to the Page).

**ACTIVE.** Applied to a portion of the Page, refers to the portion that contains the cursor, eg, the Active Line is the Line that presently contains the cursor.

#### ERASING THE DISPLAY

All erase operations consist of writing Space (2/0) codes into the selected regions of memory. Each Space code is tagged with the present setting of the GR-register (see SGR).

**OPERATOR CONTROL.** The operator may erase portions of the Page with the following keys:

**SPACE BAR.** Erases the character at the cursor. If zKRM (Key Repeat Mode) is set, it repeats if held down.

**BACKSPACE.** If zDBM (Destructive Backspace Mode) is set, backspaces and erases the character at that position. If zKRM is set, it repeats if held down.

**ERASE.** ERASE erases from cursor to end of Line. SHIFT-ERASE erases from cursor to end of Page. The key is programmable. Its functions may be disabled or changed, if desired; see Programmable Strings section. Note: In Monitor SRM, ie, off-line entry, ERASE is confined to the Active Qualified Area, and SHIFT-ERASE will not erase Areas designated as protected regardless of the setting of ERM; see Forms section.

**HOST CONTROL.** The host may erase portions of the Page with the following controls. The controls ED, EL, EF, and EA erase defined regions; see Definitions above. In addition, each permits one of three selective parameters to further specify whether the erase is to be from cursor to end of region (Ps= 0), from beginning of region to cursor (Ps= 1), or the entire region (Ps= 2).

**ED ERASE IN DISPLAY (ESC [ Ps J).** Erases some or all of the Page according to the parameter, Ps. Examples:

ESC [ J	Erases from cursor to end of Page
ESC [ 1 J	Erases from beginning of Page to cursor
ESC [ 2 J	Erases the entire Page

**EL ERASE IN LINE (ESC [ Ps K).** Erases some or all of the Active Line according to the parameter.

**EF ERASE IN FIELD (ESC [ Ps N).** Erases some or all of the Active Field according to the parameter.

**EA ERASE IN AREA (ESC [ Ps O).** Erases some or all of the Active Qualified Area according to the parameter.

The host may also erase a specified number of characters (up to 255).

ECH ERASE CHARACTER (ESC [ Pn X). Erases the following Pn characters, starting with the character at the cursor.

#### EDITING THE DISPLAY

OPERATOR CONTROL. The operator may edit portions of the Page with the following keys:

EDIT. Toggles the terminal into and out of Insertion-Replacement Mode (IRM). In Insertion mode, character entries are inserted into the Page, pushing the characters presently displayed to the right (instead of overwriting or replacing them).

DELETE. Deletes the character at the cursor and moves the following characters left to fill the space. SHIFT-DELETE deletes the entire line containing the cursor and moves the following lines up.

INSERT. Moves the character at the cursor to the right, opening a space. SHIFT-INSERT moves the entire line containing the cursor down, opening an empty line.

In all cases, the extent of the Page affected is determined by the Select Editing Extent setting near the end of Setup line D. It should normally be set to 1 (= Line). Note: In Monitor SRM, ie, off-line entry, the effect of these keys is confined to the Active Qualified Area regardless of the Select Editing Extent setting; see Forms section.

The DELETE and INSERT keys repeat when held down, regardless of the setting of zKRM (Key Repeat Mode), unless specifically re-programmed not to repeat. All of the keys are programmable. Their functions may be disabled or changed if desired; see Programmable Strings section.

HOST CONTROL. The host may edit portions of the Page with the following controls.

DCH DELETE CHARACTER (ESC [ Pn P). Deletes Pn characters, starting with the character at the cursor, by shifting the remaining characters (in the editing region) to the left. Character positions vacated at the end of the editing region are erased. Examples:

ESC [ P	Deletes 1 character
ESC [ 12 P	Deletes 12 characters

ICH INSERT CHARACTER (ESC [ Pn @). Inserts Pn spaces (2/0) at the cursor by shifting the character at the cursor, and all following characters (in the editing region), to the right. Characters shifted past the end of the editing region are lost.

Instead of using the ICH control to open up a space and then writing into it, the host may put the terminal into the Insertion (set) state of IRM and insert the characters directly. The SM sequence, ESC [ 4 h, puts the terminal into Insertion mode. The RM sequence, ESC [ 4 l, returns it to Replacement mode.

SEE SELECT EDITING EXTENT (ESC [ Ps Q). Sets an Editing Extent (EE) register according to the selective parameter. The contents of this register limit the effect of the character edit controls, DCH and ICH, and entries in the Insertion state of IRM, to Page (Ps= 0), Line (Ps= 1), Field (Ps= 2), or Qualified Area (Ps= 3); see Definitions above. Examples:

ESC [ Q	Selects end of Page (as the end of the editing region)
ESC [ 1 Q	Selects end of Line
ESC [ 2 Q	Selects end of Field
ESC [ 3 Q	Selects end of Qualified Area

DL DELETE LINE (ESC [ Pn M). Deletes the contents of Pn lines, starting with the Active Line, by shifting the remaining lines in the Page upward. Lines vacated at the end of the Page are erased.

IL INSERT LINE (ESC [ Pn L). Inserts Pn erased lines by shifting the contents of the Active Line, and all following lines, downward. Lines shifted past the end of the Page are lost.

zPSH PUSH LINE (ESC [ Pn s). Moves the contents of Pn lines, starting with the Active Line, to the bottom of the Page, by shifting the bottom lines of the Page upward.

zPOP POP LINE (ESC [ Pn t). Moves the contents of Pn lines at the bottom of the Page to (start at) the Active Line, by shifting the Active Line, and all following lines, downward.

NOTE: The zPSH and zPOP controls permit re-arranging the data in the Page without re-transmitting; see also Display section, Setting Up Multiple Pages.

#### USING THE zSDP CONTROL IN EDITING

The standard controls provide a great deal of versatility in confining the effect of erase and edit operations to defined regions. The zSDP control extends this versatility even more.

The zSDP control defines the 'Page'. It partitions memory, but does not change its contents; see Display section, Definitions. Erase and edit operations, by definition, act only on the contents of the Page. The zSDP control permits the effect of these operations to be confined to specified lines.

#### Example:

Assume a 60-line Page, SEE set to 0 (edit to end of Page), and a 9-line paragraph in lines 20 through 28. The zSDP control, ESC [ 28 ; 19 p, narrows the Page to contain only the paragraph. The erase and edit controls now act only on the paragraph; the remaining text is unaffected. When the paragraph has been edited, the zSDP control, ESC [ 60 p, returns the Page to its prior setting.

CONTENTS	Page
MODES THAT AFFECT TRANSMISSION .....	1
INITIATING TRANSMISSION .....	2
FLOW CONTROL OF TRANSMITTED DATA .....	2
ABORTING A TRANSMISSION .....	3
HEADER AND TRAILER STRINGS .....	3
USING THE TRANSFER POINTER .....	3
USING SELECTED AREAS .....	4
USING GUARDED AREAS .....	5

## MODES THAT AFFECT TRANSMISSION

The following modes affect the transmission of data from the terminal's display memory to the host. Their settings can be reviewed, and changed as desired, on the Setup lines; see Setup section. The Setup line letter, and mode number, are shown in parentheses following the name. Detailed descriptions of these modes can be found in the Appendix.

zTPDM TRANSFER POINTER DISPLAY MODE (A,41). During a transmission, the cursor symbol is not displayed (since the keyboard is disabled). This mode permits a transfer pointer symbol to be displayed (as feedback to the operator that the terminal is in the transmit state and that transmission is occurring).

zAKDM AUTO KEYBOARD DISABLE MODE (B,46). Upon completion of a transmission, the keyboard is automatically re-enabled (and the cursor symbol returns). This mode permits the keyboard to remain disabled, until purposefully released by the host with an EMI (Enable Manual Input) control.

zLTM LINE TRANSFER MODE (B,42). A transmission normally continues until completion. For hosts that cannot buffer a full Page, this mode permits the the terminal to stop transmitting at the end of each line (requiring the host to send a DC1-code (1/1) to obtain the next line).

NOTE: The terminal will respond normally, in all modes, to a DC1/DC3 protocol (or use of the Clear-to-Send interface signal) to start/stop transmission.

TTM TRANSFER TERMINATION MODE (B,16). A transmission will normally terminate after sending the character at the cursor position. This mode permits the transmission to continue to end of Page.

zGRM GRAPHIC RENDITION TRANSFER MODE (B,45). The characters in the Page are normally transmitted without information regarding their rendition. This mode permits this information to be transmitted, by including the necessary SGR controls in the transmissions.

GATM GUARDED AREA TRANSFER MODE (B,1). Areas of the Page may be designated as guarded (ie, their contents are not normally to be transmitted); see Forms section. This mode permits the contents of these Areas to be included in the transmissions.

The following modes permit trailing Space codes (2/0) to be suppressed in the

transmissions.

FETM FORMAT EFFECTOR TRANSFER MODE (B,14). Permits trailing Spaces to be suppressed at end of line (if FETM is reset). A New-Line code is inserted to mark their omission. LNM (LF New Line Mode) selects CRLF (LNM reset) or LF (LNM set) as the New-Line code.

zCSTM COLUMN SEPARATOR TRANSFER MODE (B,43). Permits trailing Spaces to be suppressed at each Columnar tab stop. An HT-code (0/9) is inserted to mark their omission.

zFSTM FIELD SEPARATOR TRANSFER MODE (B,44). Permits trailing Spaces to be suppressed at each Page tab stop. An HT-code is inserted to mark their omission.

The following modes permit Selected Area transmission; see Using Selected Areas below.

SATM SELECTED AREA TRANSFER MODE (B,17). Permits transmission of the Active Selected Area (if SATM is reset).

MATM MULTIPLE AREA TRANSFER MODE (B,15). Permits transmission of all Selected Areas in the Page (if SATM is reset).

#### INITIATING TRANSMISSION

OPERATOR CONTROL. The operator will normally initiate transmission with the SEND key. The SEND key is initialized to null at power-on. It can be programmed with whatever control string is desired (and can be accommodated by the host); see, for example, Using the Transfer Pointer, below. If the string is always wanted, the programmable Power-on string may be used to load the key with the string each time the terminal is powered on.

HOST CONTROL. The host initiates transmission with the STS control.

STS SET TRANSMIT STATE (ESC S). Initiates transmission, provided the terminal's communications interface is ready; see Flow Control of Transmitted Data below. The transmission starts at the top of Page and terminates with the cursor (TTM reset) or end of Page (TTM set). The region transmitted can be narrowed (to as small as a single character) with the zSTP and zSTE controls; see Using the Transfer Pointer. Sub-regions may be excluded from the transmission; see Using Guarded Areas and Using Selected Areas.

#### FLOW CONTROL OF TRANSMITTED DATA

The transmission of data from the terminal's memory to the host is controlled by two flags in the terminal and a communications input.

A Transmit Ready Flag (TRF) indicates that transmission is desired. It is set by the STS control, whether entered by the operator or received from the host. It is reset at power-on and at the end of each transmission. It may also be reset by the operator with the BREAK or SETUP keys, or by the host with the INT control; see Aborting the Transmission below.

A Communications Ready Flag (CRF) indicates that the communications interface is ready to send the data. It is set at power-on. It can be set and reset by the host with the DC1 (1/1) and DC3 (1/3) controls. If zLTM is set, it is automatically reset after each line is transmitted.

A Clear-to-Send (CTS) input signal at the communications connector (pin 5) indicates that the external communications system is ready to accept the data. This input may be controlled by the external system, or it may be left open (if the external system is always ready); see Checkout & Installation section.

All three conditions must be ready for transmission to occur. Therefore, the actual transmission may be initiated by any one of the conditions by first setting the others. The transmission may be suspended by resetting CRF or by pulling the CTS input low. The transmission may be terminated only by resetting TRF.

#### ABORTING A TRANSMISSION

**OPERATOR CONTROL.** The operator may abort a transmission by typing the BREAK or SETUP key.

**HOST CONTROL.** The host may abort a transmission by sending the INT control.

INT INTERRUPT (ESC a). Terminates a transmission in progress.

NOTE: This control is detected at the input of the terminal's FIFO buffer. The host may send other controls during the transmission (to be executed upon completion of the transmission) without interfering with its ability to abort the transmission.

If a trailer string has been programmed (see Header and Trailer Strings below), aborting a transmission aborts only the text portion of the transmission. The trailer string is still sent.

#### HEADER AND TRAILER STRINGS

The terminal provides programmable Header and Trailer strings; see Programmable Strings section. Both are initialized to null at power-on. If programmed, the Header string precedes, and the Trailer string follows, all STS-initiated transmissions. Example:

```
ESC P > | 150; 0; 1 | ~B | | ~C ESC \
```

This string programs an STX (0/2) into the Header string and an ETX (0/3) into the Trailer string. All STS-initiated transmissions will now begin with an STX and end with an ETX.

#### USING THE TRANSFER POINTER

Transmission normally starts at the beginning of Page, and ends with the cursor (TTM reset) or end of Page (TTM set). The terminal provides two controls that permit the starting and ending points to be specified anywhere within the Page.

zSTP SET TRANSFER POINTER (ESC [ Pn1 ; Pn2 u). Sets the starting address of the transfer pointer to line Pn1, column Pn2. The transmission includes the character at this position. The default value (for line and/or column) is the present cursor position.

zSTE SET TRANSFER END (ESC [ Pn1 ; Pn2 z). Sets the ending address of the the transfer pointer to line Pn1, column Pn2. The transmission includes the character at this position. The default value (for line and/or column) is the present cursor position. Note: If TTM is reset (ie, transmit to cursor), if the cursor is encountered before reaching the specified endpoint, the transmission will terminate after transmitting the character at the cursor. This may be avoided by setting TTM.

#### Examples:

ESC [ u ESC [ z ESC S

This string transmits the character at the cursor. If a Header and Trailer string are programmed, they will precede and follow the character.

ESC [ ; 1 u ESC [ ; 80 z ESC S

This string transmits the entire contents of the active line (ie, the line containing the cursor). If FETM is reset, any trailing spaces are suppressed and the transmission is terminated with the new-line character selected by LNM. Alternatively, FETM may be set and the new-line character programmed into the Trailer string.

ESC P > | 140 | ~[ [ ; 1 u ~[ [ ; 80 z ~[ S ESC \

This string programs the preceding string into the SEND key to provide a Line Transmit function. The contents of the active line are now sent to the host whenever the SEND key is typed.

ESC [ ; 2 u ESC [ ; 80 z ESC S

If your system uses a 1-character prompt, this string transmits the line without transmitting the prompt.

ESC [ 6 ; 1 u ESC [ 12 ; 80 z ESC S

This string transmits the contents of lines 6 through 12 of the Page.

Note: The last example could also have been done using the zSDP control. The zSDP control defines the 'Page'. It partitions memory, but does not change its contents; see Display section, Definitions. The STS control sends only the contents of the Page (TTM set). The string, ESC [ 12 ; 5 p ESC S, would have accomplished the same function with fewer codes (ignoring the fact that the Page still has to be restored). The transfer pointer controls are more versatile (because they permit column parameters).

#### USING SELECTED AREAS

The terminal provides a Selected Area capability that permits the region(s) of the Page (to be transmitted) to be selected by visible symbols in the display. A Selected Area is defined as starting with an SSA symbol (forward-pointing triangle) and ending with an ESA-symbol (backward-pointing triangle).

The host enters the symbols with the SSA and ESA controls.



SSA START OF SELECTED AREA (ESC F). Writes an SSA-symbol to the Page.

ESA END OF SELECTED AREA (ESC G). Writes an ESA-symbol to the Page.

The operator can enter (or insert) the symbols by programming the strings into the desired keys. Example:

```
ESC P > | 122 | ~[ F | | ~[ G ESC \
```

This string programs the F1 and F2 keys to generate the SSA and ESA symbols respectively.

The SSA and ESA symbols are characters, just like any of the other graphic characters. They may be erased with erase controls, moved with edit controls, and are included in transmissions (as their escape sequence).

The symbols assume a control attribute only when SATM is in the reset state. In that case, they affect the transmission according to the setting of MATM.

If MATM is reset, only the active Selected Area (ie, the one containing the cursor) is transmitted. If TTM is reset (ie, transmit to cursor), the transmission includes the SSA-symbol and all following text through the character at the cursor.

If MATM is set, all Selected Areas in the Page (or up to the cursor, if TTM is reset) are transmitted. The contents of each area are bracketed by SSA and ESA symbols in the transmission.

#### USING GUARDED AREAS

The terminal provides a Guarded Area capability that permits designated regions in the Page to be 'guarded' from transmission. The regions are designated by the host with the DAQ control; see Forms section. They are normally intended for use with Forms, but may also be used by the host simply to exclude designated regions of the Page from the transmission. Example:

```
ESC [ 6 H ESC [ 1 o ESC [ 9 H ESC [ o ESC S ESC [ 60 p
```

This string excludes lines 6-8 of the Page from the transmission and restores the Page (to 60 lines).

NOTE: Selected and Guarded Areas provide alternative means of accomplishing the same result, namely selective transmission of designated regions of the Page. It is envisioned that the operator will tend to use Selected Areas (because of the visible symbols) and the host Guarded Areas. Using the two types of areas in combination is not recommended. Doing so will yield predictable, but not necessarily sensible, results.



CONTENTS	Page
USING THE PRINTER .....	1
PRINTER SETUPS .....	1
FLOW CONTROL .....	2
TOP-OF-PAGE CONTROL .....	3
USING REMOTE COPY .....	4
USING LOCAL COPY .....	5
USING PAGE PRINT .....	7
MODES THAT AFFECT PAGE PRINT .....	8
USING THE TRANSFER POINTER .....	9
USING SELECTED AREAS .....	10
USING GUARDED AREAS .....	11

## USING THE PRINTER

The XL Series provides three different methods of printing:

**REMOTE COPY.** Permits the terminal and printer to be used as two separate devices, merely sharing a common communications line.

**LOCAL COPY.** Permits printing all or part of the data that the host sends or echoes to the terminal.

**PAGE PRINT.** Permits printing all or part of the displayed data.

The operation and use of these printing methods is described in this section. It assumes that the printer has already been attached to the terminal and properly set up (baud rate, parity, new-line character, new-line pads, etc.); see Checkout & Installation, Printer Installation.

## PRINTER SETUPS

The terminal permits a wide variety of printers to be used. There are few requirements of the printer (other than that it accept a new-line control, either CR, LF, or CRLF). If the printer accepts an FF (0/12) control (to advance the paper to top-of-page), it may be used (with all forms of print). If the printer is capable of an Xon/Xoff protocol (either codes or signal), it also may be used. Any other printer features are usable only with Remote Copy (as the terminal does not support embedded control codes in either Local Copy or Page Print).

The settings associated with the printer are consolidated on Setup line C; see Setup section. Besides adapting the terminal to various printers, they provide some local control of the printout.

The following settings are applicable to all forms of print. They control the basic signal form of data to the printer.

**BAUD RATE.** Sets the baud rate (110-19200) required by the printer. This will generally be set to the maximum that the printer will accommodate.

**PARITY.** Sets the parity (00=Odd 01=Even 10=Space 11=Mark) required by the

printer. If none, set for Mark.

STOP BITS. Sets the number of stop bits required by the printer. This will generally be set to the minimum that the printer will accommodate.

The following settings are applicable to Local Copy and Page Print only. In Remote Copy, the host determines what codes are used for new-line and new-page actions, and whether pad characters are required.

NEW-LINE PADS. Sets the number of NUL-codes (0/0) to be sent to the printer following each new-line character (to allow time for carriage return, if needed).

NEW-LINE CR/LF. Selects the codes (01=LF 10=CR 11=CRLF) required by the printer to perform a new-line action.

FORM FEED. Selects the code (0=NLS 1=FF) required by the printer to perform a new-page action. If the printer recognizes an FF-code (0/12) as a top-of-page control, set to 1. Otherwise, set to 0, and the terminal will generate the necessary number of new-line codes to advance the paper to the top of the next page.

The following settings are also applicable to Local Copy and Page Print only. In Remote Copy, the page and margins are controlled by the host. These settings may also be set by the host with the zSPF control; see Top-of-Page Control, below.

PRINTED LINES. Sets the number of lines to be printed per page (eg, 60 leaves a 3-line top and bottom margin on a 66-line printer).

TOTAL LINES. Sets the number of lines possible per page (eg, 66, for a 6 line/inch printer with 11-inch paper).

LEFT MARGIN. Sets the number of Spaces (2/0) to be sent to the printer at the beginning of each line for a left margin (eg, 12 gives a 1-inch left margin for a 12 character/inch printer, provided that the paper is positioned so that, with no margin, the first character prints at the left edge of the paper.)

## FLOW CONTROL

The terminal accommodates printers that use either DC1/DC3 codes, or an RS232 control signal, for ready/busy control.

It will suspend further transfer of data to the printer on receipt of a 'busy' signal from the printer (either a DC3 control received through pin 2 of the printer connector, or an RS232 'off' level at pin 20). It will resume transferring the data on receipt of a 'ready' signal from the printer (either a DC1 control received through pin 2 of the connector, or an RS232 'on' level at pin 20).

This is applicable to all three methods of printing; see the Flow Control paragraph in the description of each method, below.

## TOP-OF-PAGE CONTROL

If the printer uses paper other than roll stock, it is necessary that the printer and printing source agree on the location of top-of-(printed)page.

When using Remote Copy, this is a matter between the host and printer; the terminal is not involved in the paging operation. The operator typically positions the paper to top-of-page at (printer) power-on, and thereafter the host takes care of the paging. If the printer has a printed line counter, it is typically initialized to top-of-page at power-on. If the operator aborts a print, he/she will typically use the Form Feed button on the printer (if it has one, and is convenient), or a pre-arranged code sequence with the host, to advance the paper to the top of the next page.

When using Local Copy or Page Print, the terminal is the printing source, and controls the paging operation. The terminal knows what constitutes a printer page by virtue of the Total Lines setting. It knows how many lines you want to print per page by virtue of the Printed Lines setting. Once the printer is initialized, the terminal takes care of the paging, with a printed-lines counter. The counter is initialized to top-of-page at power-on. Thereafter, it counts the lines it prints, and advances the paper to the top of the next page (by issuing an FF-code, or the necessary number of new-line codes) in accordance with your settings.

If a print is aborted, the terminal's line counter, the printer's paper position, and the printer's line counter (if applicable) are all left at other than top-of-page. The terminal's zSPF control permits all three to be advanced to the top of the next page, or only the terminal's line counter (the printer being advanced by other means).

This control may be used by the operator by typing it in, or loading it into a programmable key.

zSPF SET PRINT FORMAT (ESC [ Pn1 ; Pn2 ; Pn3 w). Sets the number of Printed Lines (Pn1), Total Lines (Pn2), and Left Margin (Pn3) (see Printer Setups above) AND advances the terminal's printed-line counter to top-of-page (with or without advancing the printer; see examples below).

ESC [ 60 ; 66 ; 10 w

Sets Printed Lines = 60, Total Lines = 66, and Left Margin = 10, AND advances the terminal's line counter to top-of-page (without advancing the printer). The paper should be positioned to top of page (and the printer's line counter reset, if applicable) before issuing this command.

ESC [ 62 ; ; 12 w

Changes the Printed Line and Left Margin settings without changing the Total Line setting, AND advances the terminal's line counter to top-of-page, without advancing the printer.

ESC [ w

Changes no settings, but advances the terminal's line counter to top-of-page by issuing new-line codes (or an FF-code, if the Form Feed setting on Line C is selected) to advance the paper position (and line counter setting) in the printer to top-of-page. This may be used after aborting a Local Copy or Page Print to advance both terminal and

printer to the top of the next (printed) page.

```
ESC P > | 122;0;2 | ~[ [ w ESC \
```

This string loads the preceding control into F1 (making F1 a New Page key).

#### USING REMOTE COPY

Remote Copy permits data to be printed without being displayed. It may be used by either the operator or the host (although intended primarily for the host).

NOTE: Remote Copy is the only way of printing a file that contains embedded printer control codes. As such, it is the preferred method of printing (if the host is cooperative).

PRINTER SETUPS. Remote Copy uses only the Baud Rate, Parity, and Stop Bit settings on Setup line C. All other settings are ignored. (The other settings affect the content of the transfer which, in Remote Copy, is controlled entirely by the host).

FLOW CONTROL. The data transferred from the host to the printer is buffered through the terminal's FIFO. The FIFO receives characters from the host at the host's baud rate. It passes those characters on to the printer at the printer's baud rate. In addition, it responds to any 'busy' signal from the printer (either a DC3 control received through pin 2 of the printer connector, or a low level at pin 20), and suspends passing on the characters until it receives a 'ready' signal from the printer (either a DC1 control received through pin 2 of the connector, or a high level at pin 20).

If the printer's baud rate is less than the host's, or if the printer goes 'busy', the FIFO will gradually fill. When the FIFO becomes 'almost full', it sends an XOFF control to the host, suspending further transmission from the host (if the host supports an Xon/Xoff protocol; see zAXM). When enough data has been printed that the terminal's FIFO becomes 'almost empty', the terminal sends an XON control to the host to resume transmission.

OPERATOR CONTROL. The operator may use the zSRC control to write directly from the keyboard to the printer (either in Local Test mode, see Diagnostics section, or echoed through the host). The control may be typed in, or loaded into a Programmable key. Depressing CTRL-C (ie, ETX) terminates writing to the printer. Example:

```
ESC [ v
```

Starts Remote Copy. All characters subsequently typed are sent to the printer (instead of the display). Depressing CTRL-C returns the terminal to normal operation.

```
ESC P > | 122 | ~[ [ v ESC \
```

This string loads the Start Remote Copy control into F1.

HOST CONTROL. The host may use the zSRC control to write to the printer. If it is desired to write also to the display, Local Copy should be used. The data to be printed may be interspersed (in small packets) with the normal communications to the display (by using the numeric parameter), or printed in a

batch by using an ETX to terminate the print.

zSRC START REMOTE COPY (ESC [ Pn v). Causes the next Pn characters (up to 255) to be directed to the printer (instead of the display). If no parameter is given, all subsequent characters are directed to the printer until the first occurrence of an ETX-code (0/3) in the data stream. All formatting of the data is the responsibility of the host. The terminal merely passes the buffered data to the printer interface. Examples:

ESC [ 128 v

Causes the next 128 characters sent to the terminal to be directed to the printer only. The 129th, and subsequent characters, are directed back to the display.

ESC [ v

Causes all subsequent characters sent to the terminal to be directed to the printer only, until an ETX character is sent to direct the characters back to the display.

An 8-Bit Remote Copy Mode (z8RCM) permits passing 8-bit data to the printer port. However, this mode places certain restrictions on terminal operation and should be used with caution; see Appendix, z8RCM.

#### USING LOCAL COPY

Local Copy permits data to be displayed and printed simultaneously. It may be used by either the operator or the host. Local copy lacks the versatility of Remote Copy (for formatted printing), but requires no cooperation from the host. The host needs no information about the printer (or even whether there is one).

Local Copy can be faster than Remote Copy when the comm rate is slower than the printer (eg, using a 40 char/sec. printer with a 300 baud modem), because the terminal expands the controls used by the host into the characters required by the printer. For example, tab codes from the host are expanded into the appropriate number of spaces for the printer. The new-line character used by the host is translated into the new-line character(s) required by the printer, including new-line pads. Etc.

Local Copy may also be used for logging the transactions between operator and host, for debug or historical reference.

PRINTER SETUPS. All settings are applicable. Since the printing is done from the Page, the host needs no information about the printer being used (unlike Remote Copy). Each line is printed as it finally appears on the screen, regardless of what formatting and editing controls were used in creating it. The only control needed by the printer is a new-line control (selected on Setup line C). If the printer recognizes an FF-code (0/12) as a top-of-page control, the Form Feed setting should be selected. The Left Margin setting may be used to position the printout laterally on the paper. The Printed Lines setting may be used to control the number of lines per page to be printed.

FLOW CONTROL. Whenever a new-line control is received by the Display Processor (ie, after the FIFO, but before moving the cursor), the Display Processor checks to see if Local Copy is 'on'. If so, it tags the active line (ie, the

line containing the cursor) as a line to be printed. New-line controls are LF, IND, NEL, or CR (if zCNM is set).

During printing, the terminal's FIFO and its entire Page (see Display section, Definitions) serve as a buffer. The Page is being filled at the host's baud rate. The contents of the Page (ie, the lines tagged to be printed) are being copied to the printer at the printer's baud rate. In addition, the terminal responds to any 'busy' signal from the printer (either a DC3 control received through pin 2 of the printer connector, or a low level at pin 20), and suspends copying further characters until it receives a 'ready' signal from the printer (either a DC1 control received through pin 2 of the connector, or a high level at pin 20).

If the printer's baud rate is less than the host's, or if the printer goes 'busy', the Page will gradually fill. When a new-line control is received by the Display Processor that, if processed, would cause a line that has not yet been printed to scroll off the top of the Page, the scroll is inhibited, and further processing from the FIFO is suspended.

As the host continues to send characters, the FIFO will gradually fill. When the FIFO becomes 'almost full', it sends an XOFF control to the host, suspending further transmission from the host (if the host supports an Xon/Xoff protocol; see zAXM).

When the line at the top of the Page has been printed, the Page is allowed to scroll, FIFO processing is resumed, and another line of data is admitted to the Page.

When enough lines of data have been admitted to the Page that the FIFO becomes 'almost empty', the terminal sends an XON control to the host to resume transmission.

**OPERATOR CONTROL.** The operator uses SHIFT-PRINT to toggle the terminal into and out of Local Copy. This is a Local-only control, ie, no code is transmitted to the host; see Programmable Strings section.

To log transactions, depress SHIFT-PRINT (to toggle Local Copy on). All subsequent transactions (ie, everything that's written to the display) is also printed. Depress SHIFT-PRINT again (to toggle Local Copy off) to terminate logging.

To print a file, depress SHIFT-PRINT, and initiate display of the file. When the file has been printed, depress SHIFT-PRINT again (or initiate display of another file to be printed).

Toggling Local Copy on before initiating the display of a file results in the host command (used to initiate the display) to also be printed. To avoid this, SHIFT-PRINT must be depressed AFTER the host command has been echoed and BEFORE the first line of the file has been (completely) written to the display. This requires a fast finger, or a slow host, or a host with a delay (or sleep) command.

Alternatively, zAPM (Auto Pause Mode) may be used; see Display section, Flow Control of Received Data. Depress CTRL-PAUSE (to go into Auto Pause Mode). Initiate display of the file. Depress PAUSE (to permit the echoed command line to display). Depress SHIFT-PRINT (to toggle Local Copy on). Now, each



time you depress PAUSE, another line is admitted to the screen -- and printed. You may step through the file, a line at a time, printing what you want and leaving out what you don't want (by toggling Local Copy on and off), or you may depress CTRL-SHIFT-PAUSE (to leave Auto Pause Mode) and the file will complete printing by itself (and without the command line).

To stop the printing of a file (when you only wanted part of it), simply depress SHIFT-PRINT. The line being written to the display at the time the key is depressed will not be printed; all previous lines will.

To abort the printing of a file (because something's gone wrong), depress CTRL-SHIFT-RESET (which clears the display and resets Local Copy), re-position the printer to top-of-page (see Top-Of-Page Control), and re-initiate display of the file.

If Local Copy is used frequently, you may wish to program separate Copy On and Copy Off keys. Example:

```
ESC P > | 145;1;2 | ~[ [ 4 i | ;2 | ~[ [ 5 i ESC \  
This string programs SHIFT-PRINT to turn Local Copy on and CTRL-PRINT to  
turn Local Copy off.
```

HOST CONTROL. The host uses the MC control to turn Local Copy on and off.

MC MEDIA COPY (ESC [ Ps i). Controls the transfer of data from the terminal's Display Memory to the printer according to the selective parameter. Ps= 5 turns on Local Copy; Ps= 4 turns off Local Copy.

Lines longer than the (display) Page width may be printed by allowing the (display) line to wrap at end of line. Display wrap does not cause the line to be printed; only a new-line control does. Top-of-page control is available to the host by issuing an FF (0/12) control in the transmission; see FF. The terminal buffers the FF-code, with the data to be printed, and passes it on to the printer either as an FF-code or the necessary number of new-line codes, depending on the Form Feed setting of Line C.

#### USING PAGE PRINT.

Page print permits all or selected portions of the displayed data to be printed. A full (printed) page of data may be assembled, edited, formatted, reviewed, and then printed. The data is printed exactly as it is displayed.

PRINTER SETUPS. The Left Margin setting shifts the printout to the right. The Printed Lines setting should be greater (eg, 66) than the number of lines set for the (display) Page size; see Display section, Definitions. The displayed data may be divided into two separate printed pages by setting the Printed Lines to be smaller than the Page size.

FLOW CONTROL. During a Page Print, characters are transferred from the terminal's Display Memory to the printer at the baud rate selected in Setup line C. The terminal responds to any 'busy' signal from the printer (either a DC3 control received through pin 2 of the printer connector, or a low level at pin 20), and suspends transferring further characters until it receives a 'ready' signal from the printer (either a DC1 control received through pin 2 of the connector, or a high level at pin 20).

The keyboard is disabled during the print operation (except for the BREAK, SETUP, and RESET keys; see KAM). Host input is not disabled. The host may send a DSR control, ESC [ 5 n, to inquire whether a Page Print is in progress. The terminal responds with the DSR control, ESC [ 1 n, if yes, or ESC [ 0 n if no.

If the host sends the DSR control, ESC [ > 5 n, the terminal suspends further processing from its FIFO until the print has been completed, and then returns the ready response. This permits the host to continue to send characters to the terminal without affecting the Page print.

If the host continues to send characters to the terminal, the FIFO will gradually fill. When the FIFO becomes 'almost full', it sends an XOFF control to the host, suspending further transmission (if the host supports an Xon/Xoff protocol; see zAXM). When the print operation has been completed (or aborted), the keyboard and FIFO processing are re-enabled, and any characters waiting in the FIFO are processed normally. When enough characters have been processed that the terminal's FIFO becomes 'almost empty', the terminal sends an XON control to the host to resume transmission.

**OPERATOR CONTROL.** The operator initiates Page Print with the PRINT key.

Before initiating print, make sure the printer is set to top-of-form (if applicable). Upon completion of Page Print, the terminal does not automatically advance the printer to the top of the next page (to allow you to compile segments of text on a single page, if desired). The form feed must be done manually between Page Prints. You may use the Form Feed button on your printer (if it has one and is convenient) or you may program a Form Feed key on the terminal; see Top-of-Page Control, above.

A Page Print may be aborted with either the BREAK or SETUP key (without clearing the data), or with CTRL-SHIFT-RESET.

**HOST CONTROL.** The host initiates Page Print with the MC control.

**MC MEDIA COPY (ESC [ i).** Initiates transfer of data from the terminal's Display Memory to the printer. The transfer starts at the top of the (display) Page and terminates with the cursor (TTM reset) or end of Page (TTM set). The region transmitted may be narrowed (to as small as a single character) with the zSTP and zSTE controls; see Using the Transfer Pointer. Sub-regions may be excluded from the transfer; see Using Guarded Areas and Using Selected Areas.

The host may advance the printer to the top of the next page with the zSPF control (ESC [ w). The host may abort a Page Print in progress with the INT control (ESC a). The host may request that it be notified when a Page Print (it initiates) has been completed with the DSR control (ESC [ > 5 n); the terminal returns the string, ESC [ n, upon completion of the Page Print.

#### MODES THAT AFFECT PAGE PRINT

The following modes affect Page Print. Their settings may be reviewed, and changed as desired, on the Setup lines; see Setup section. The Setup line letter, and mode number, are shown in parentheses following the name. Detailed descriptions of these modes can be found in the Appendix.

zTPDM TRANSFER POINTER DISPLAY MODE (A,41). During a Page Print, the cursor symbol is not displayed (since the keyboard is disabled). This mode permits a transfer pointer symbol to be displayed.

TTM TRANSFER TERMINATION MODE (B,16). A Page Print will normally terminate after transferring the character at the cursor position. This mode permits the transfer to continue to end of Page.

GAPM GUARDED AREA PRINT MODE (C,49). Areas of the Page may be designated as guarded (ie, their contents are not normally to be transferred); see Forms section. This mode permits the content of Guarded Areas to be printed.

The following modes permit Selected Areas to be printed; see Using Selected Areas below.

SATM SELECTED AREA TRANSFER MODE (B,17). Permits printing of the Active Selected Area (if SATM is reset).

MATM MULTIPLE AREA TRANSFER MODE (B,15). Permits printing of all Selected Areas in the Page (if SATM is reset).

#### USING THE TRANSFER POINTER

Page Print normally starts at the beginning of Page, and ends with the cursor (TTM reset) or end of Page (TTM set). The terminal provides two controls that permit the starting and ending points to be specified anywhere within the Page.

zSTP SET TRANSFER POINTER (ESC [ Pn1 ; Pn2 u). Sets the starting address of the transfer pointer to line Pn1, column Pn2. The transfer includes the character at this position. The default value is the present cursor position.

zSTE SET TRANSFER END (ESC [ Pn1 ; Pn2 z). Sets the ending address of the the transfer pointer to line Pn1, column Pn2. The transfer includes the character at this position. The default value is the present cursor position. NOTE: If TTM is reset (ie, transfer to cursor), if the cursor is encountered before reaching the specified endpoint, the transfer will terminate after transferring the character at the cursor. This may be avoided by setting TTM.

The pointers are restored to beginning and end of Page after each transfer. These controls may be used by the host to print a specified region of the Page. They may be used by the operator, by typing in the controls or loading them into a programmable key. Examples:

ESC [ u  
Sets the starting address to include the character at the cursor.

ESC [ z  
Sets the ending address to include the character at the cursor.

ESC P > | 122;0;2 | ~[ [ u | | ~[ [ z ESC \  
Programs the above controls into F1 and F2, respectively. To print a por-

tion of the Page, depress F1 when the cursor is at the first character to be printed, and F2 when the cursor is at the last character to be printed.

ESC [ u ESC [ z ESC [ i

This string prints the character at the cursor.

ESC [ ; 1 u ESC [ ; 80 z ESC [ i

This string prints the entire contents of the active line (ie, the line containing the cursor).

ESC [ 6 ; 1 u ESC [ 12 ; 80 z ESC [ i

This string prints the contents of lines 6 through 12 in the Page.

NOTE: The preceding example could also have been done using the zSDP control. The zSDP control defines the 'Page'. It partitions memory, but does not change its contents; see Display section, Definitions. The MC control transfers only the contents of the Page (TTM set). The string, ESC [ 12 ; 5 p ESC [ i, would have accomplished the same function with fewer codes (ignoring the fact that the Page still has to be restored). The transfer pointer controls are more versatile, because they permit column parameters.

#### USING SELECTED AREAS

The terminal provides a Selected Area capability that permits the region(s) of the Page (to be printed) to be selected by visible symbols in the display. A Selected Area is defined as starting with an SSA symbol (forward-pointing triangle) and ending with an ESA-symbol (backward-pointing triangle).

The operator may enter (or insert) these symbols with the SSA and ESA keys on the Control keypad. The host enters them with the SSA and ESA controls.

SSA START OF SELECTED AREA (ESC F). Writes an SSA-symbol to the Page.

ESA END OF SELECTED AREA (ESC G). Writes an ESA-symbol to the Page.

These symbols are characters, just like any of the other graphic characters. They may be erased with erase controls, moved with edit controls, and are included in transfers exactly as any other character. When transferred to the printer, they are sent as a Space code (2/0) (not as ESC F or ESC G).

The symbols assume a control attribute only when SATM is in the reset state. In that case, they affect the transfer according to the setting of MATM.

If MATM is reset, only the active Selected Area (ie, the one containing the cursor) is transferred. If TTM is reset (ie, transfer to cursor), the transfer includes the SSA-symbol and all following text through the character at the cursor.

If MATM is set, all Selected Areas in the Page (or up to the cursor, if TTM is reset) are transmitted. The contents of each area are bracketed by SSA and ESA spaces in the transfer.

## USING GUARDED AREAS

The terminal provides a Guarded Area capability that permits designated regions in the Page to be 'guarded' from transfer. The regions are designated by the host with the DAQ control; see Forms section. They are normally intended for use with Forms, but may also be used by the host simply to exclude designated regions of the Page from a printout. Example:

```
ESC [ 6 H ESC [ 1 o ESC [ 9 H ESC [ o ESC [ i
```

This string prints the contents of the Page, excluding lines 6-8.

NOTE: Selected and Guarded Areas provide alternative means of accomplishing the same result, namely selective transfer of designated regions of the Page. It is envisioned that the operator will tend to use Selected Areas (because of the visible symbols) and the host Guarded Areas. Using the two together is not recommended. Doing so will yield predictable, but not necessarily sensible, results.



CONTENTS	Page
DEFINITIONS .....	1
MODES THAT AFFECT FORM-FILLING .....	2
CREATING A FORM .....	2
EXAMPLE FORM .....	3
EDITING THE FORM .....	4
INITIALIZING THE FORM .....	6
ENTERING DATA INTO THE FORM .....	7
SENDING THE ENTERED DATA .....	8
CORRECTING AND RE-SENDING .....	9
PRINTING THE ENTERED DATA .....	11

## DEFINITIONS

Certain terms are used in describing the XL Form-Filling features. They are defined below to clarify their usage. These definitions follow, and are consistent with, the ANSI definitions.

**PAGE.** The portion of Active Memory between the Upper and Lower Host Areas; see Display section. The Page is the only portion of Display Memory that may be accessed, written, or affected (by erase/edit operations) with the primary cursor. In Forms Mode, the Page is divided into Qualified Areas, which may further limit these operations in respect to Local Entry.

**QUALIFIED AREA.** A portion of the Page bounded by Area Qualifiers. An Area includes the character position at the beginning Area Qualifier and excludes the character position at the trailing Area Qualifier.

**AREA QUALIFIER.** A control attribute(s) assigned to a single character position in the Page. The attribute(s) may apply only to that position (eg, a page tab stop) or to all character positions in the Area (eg, protected). Area Qualifiers are set with the DAQ control; see Creating a Form, below.

**FORMS MODE.** The terminal is said to be in Forms Mode when any Area Qualifier has been set within the Page.

**NOTE:** The significance of Forms Mode is that the user, by setting a control attribute at a specified character position within the Page, conceives of the Page as a page, rather than as a scrolling window into the host.

**LOCAL ENTRY.** Operator entry while the terminal is in Monitor SRM. The keyboard inputs are not sent to the host as they are keyed. They are, instead, displayed and executed locally, and the resultant display subsequently sent to the host as a block. Local Entry uses only the primary cursor, regardless of the setting of zACM.

**FIELD.** A portion of the Page bounded by tab stops. A Field contains the character position at the beginning tab stop and excludes the character position at the trailing tab stop. In Forms Mode, the tab stops are page tab stops; see Cursor Controls, Using Page Tabs. Page tab stops are also

Area Qualifiers. A Field contains at least one, and possibly more, Qualified Areas (eg, an unprotected and a protected Area).

NOTE: This differs from a common definition of 'field' (as a protected, followed by unprotected, region). We have adhered to the ANSI definition.

ACTIVE. Applied to a portion of the Page, refers to the portion that contains the cursor, eg, the Active Area is that Area that presently contains the cursor.

#### MODES THAT AFFECT FORM-FILLING

The following modes affect form-filling. Their settings can be reviewed, and changed as desired, on the Setup lines; see Setup section. The Setup line letter, and mode number, are shown in parentheses following the name. Detailed descriptions of these modes can be found in the Appendix.

TSM TABULATION STOP MODE (D,18). Determines whether columnar tab stops (TSM reset) or page tab stops (TSM set) are affected by the Tab Set/Clear controls.

ERM ERASURE MODE (D,6). Permits the erasure of characters regardless of their protected status.

ZAKDM AUTO KEYBOARD DISABLE MODE (B,46). Upon completion of a transmission, the keyboard is automatically re-enabled (and the cursor symbol returns). This mode permits the keyboard to remain disabled, until purposefully released by the host with an EMI control.

ZHAM HOLD IN AREA MODE (A,48). In normal entry, when an Area is filled, the cursor auto-skips to the next Area. This mode permits the cursor to remain in the Area, until purposefully advanced with the TAB key.

#### CREATING A FORM

Forms are created with the DAQ control.

DAQ DEFINE AREA QUALIFICATION (ESC [ > Ps..Ps o). Sets an Area Qualifier at the present cursor position, assigning one or more control attributes to the position according to the selective parameters:

PAGE TAB STOP (Ps= 7). Sets a page tab stop at the beginning of the Area. The tab stop is active in respect to entries from the host, as well as Local Entry.

PROTECTED/GUARDED (Ps= 1). Local Entry is not permitted into the Area, AND the contents of the Area will not be included in transmissions to the host (if GATM is reset) or in transfers to the printer (if zGAPM is reset).

PROTECTED/UNGUARDED (Ps= 8). Local Entry is not permitted into the Area, but the contents of the Area are included in transmissions to the host and transfers to the printer, regardless of the settings of



GATM and zGAPM.

UNPROTECTED (Ps= 0). Local Entry is permitted into the Area.

NUMERIC ONLY (Ps= 3). Local Entry is permitted into the Area, but restricted to numerics and punctuation only (codes 2/0 (Sp) to 3/15 (?) inclusive).

RIGHT JUSTIFY (Ps= 5). Local Entry is permitted into the Area, but is right justified when leaving the Area with the TAB key (ie, the entry is shifted right until a non-Space character occupies the last character position in the Area).

CONCEALED (Ps= 18). Local Entry is permitted into the Area, but is displayed only as 'white' Spaces. Note: This attribute is not implemented on the Guru.

Note that the attributes that restrict entry apply only to operator entries in an off-line (ie, Monitor SRM) situation. In that situation, the entry control functions must be accomplished in the terminal (because the host is not involved in the entry). In an on-line (ie, Simultaneous SRM) situation, the entry control functions are better accomplished in the host software.

The DAQ control is not the only control that can create a Form (and put the terminal into Forms Mode). The following controls are not as versatile as DAQ, but are more code efficient and may achieve the desired result in special situations:

SPA START OF PROTECTED AREA (ESC V). Same as DAQ control, ESC [ 1 o.

EPA END OF PROTECTED AREA (ESC W). Same as DAQ control, ESC [ 7 o.

HTS HORIZONTAL TAB SET (ESC H). Same as DAQ control, ESC [ 7 o (if TSM is set).

#### EXAMPLE FORM

This example sets up a one-line Form. A diagram of the Form is shown, followed by the string that creates it. This example will be used later to illustrate the entry of data into the Form, and the sending and printing of the entered data.

In the diagram of the Form (following), the numbers are column numbers across the line. The entry 'FORM#1' is in a Protected/Unguarded area. The operator cannot change it, but it will be included in transmissions to the host. The other entries in upper-case are in Protected/Guarded areas. They are likewise protected from the operator, but will not normally be included in transmissions to the host. The lower-case letters show the Unprotected areas, a lower-case 'a' indicating alphanumeric, and lower-case 'n' indicating Numeric-only. The unprotected area following WAGE is a Right-Justify area. The unprotected area following CODE is a Concealed area. The unprotected area following NOTES extends to end of Page. All protected areas will be displayed with a Reverse rendition. The symbol 'T' indicates the location of page tab stops.

The string that creates the Form is broken into multiple lines for clarity and ease of annotation (ie, new-line codes should not be included in the string). The spaces shown between characters are also for clarity (and should not be included, except as specifically called for with the designation, Sp). ESC denotes the ESC-code (1/11).

Form:

```

          1           2           3           4           5           6
12345678901234567890123456789012345678901234567890123456789012345678
    
```

```

FORM 1 NAME aaaaaaaaaaaa AGE nn WAGE nnnn.nn CODE aaaaaa NOTES aaa...
          T                   T       T           T           T
    
```

String:

```

ESC [ 7 m           -Sets the GR-register to Reverse rendition
ESC [ H             -Moves the cursor Home
ESC [ 8 o           -Starts a protected/unguarded Area
FORM#1              -Writes 'FORM#1' into the Area
ESC [ 1 o           -Starts a protected/guarded Area
SpNAME$Sp           -Writes ' NAME ' into the Area
ESC [ 7 o           -Sets a page tab stop and starts an unprotected Area
ESC [ 26 `          -Moves the cursor to column 26
ESC [ 1 o           -Starts a protected/guarded Area
SpAGESp             -Writes ' AGE ' into the Area
ESC [ 7 ; 3 o       -Sets a page tab stop and starts a Numeric-only Area
ESC [ 33 `          -Moves the cursor to column 33
ESC [ 1 o           -Starts a protected/guarded Area
SpWAGESp            -Writes ' WAGE ' into the Area
ESC [ 7 ; 3 ; 5 o   -Sets a page tab stop, starts a Numeric/Justify Area
ESC [ 46 `          -Moves the cursor to column 46
ESC [ 1 o           -Starts a protected/guarded Area
SpCODESp            -Writes ' CODE ' into the Area
ESC [ 7 ; 18 o      -Sets a page tab stop and starts a Concealed Area
ESC [ 58 `          -Moves the cursor to column 58
ESC [ 1 o           -Starts a protected/guarded Area
SpNOTESSp           -Writes ' NOTES ' into the Area
ESC [ 7 o           -Sets a page tab stop and starts an unprotected Area
    
```

### EDITING THE FORM

The control attributes (ie, Area Qualifiers) are not stored in the Page (with the characters and their visual attributes) but are, instead, stored separately, as a control table, indexed by character position. This not only permits the Form to be rapidly written when needed, it is also efficient when entering data into the Form, and when sending or printing data from it.

It has the added advantage of clearly separating the visual and control attributes. The person creating the Form may freely associate these attributes in any way that best accomplishes the task at hand (including no association, or different associations in different parts of the Form). The rendition of protected parts of the Form is controlled when the Form is written. The rendition of the unprotected parts of the Form is controlled by the setting of the GR-register when the unprotected areas are erased. The rendition of the

operator's entries is controlled by the setting of the GR-register during the entry.

Thus, a 'Form' exists in the terminal only by virtue of a table of Area Qualifiers vs. character position. The significance of Forms Mode, is that it causes the Display Processor to reference the table as part of processing an input. If the input is a local keyboard entry (ie, from the keystroke buffer; see Entering Data into the Form), the Display Processor references the table for each entry. If the input is received from the host (ie, from the FIFO buffer), the Display Processor references the table only when it receives one of the tab controls (HT, CHT, CBT, or HTJ).

The Area Qualifier (AQ) table contains up to 320 Area Qualifiers. This is the only factor that limits the complexity of the Form that can be created; there are no limits on number of qualifiers per line, spacing between qualifiers, etc.

The AQ table may be edited in various ways:

**CHANGING AN ENTRY IN THE AQ TABLE.** An entry may be changed by first positioning the cursor to the Area Qualifier (ie, to the first character position of the Area).

A DAQ control may be used to change the Area Qualifier in accordance with new parameters. This is likewise true of the SPA and EPA controls.

A set-tab control, HTS (ESC H) or CTC (ESC [ W), may be used to add a tab stop to the Area Qualifier. A clear-tab control, TBC (ESC [ g) or CTC (ESC [ 2 W), may be used remove a tab stop from the Area Qualifier. A clear-all-tabs-in-line control, TBC (ESC [ 2 g) or CTC (ESC [ 4 W), may be used to remove tab stops from all Area Qualifiers in the Active line. A clear-all-tabs control, TBC (ESC [ 3 g) or CTC (ESC [ 5 W), may be used to remove tab stops from all Area Qualifiers. In all cases, the tab stops are added or removed without changing any of the other control attributes of the Area Qualifier.

Whenever a tab stop is removed, the remaining attributes of the Area Qualifier are compared with the attributes of the preceding Area Qualifier. If they are the same (or the same except for a tab stop in the preceding qualifier), the entry is deleted (ie, the preceding Area is extended to encompass the present Area).

**INSERTING AN ENTRY INTO THE AQ TABLE.** An entry may be inserted into (or added to) the table, by first positioning the cursor to the desired position (that does not presently contain an Area Qualifier, ie, any position other than the first character position of an existing Area).

A DAQ control inserts an Area Qualifier with the attributes specified by its parameters. An SPA control inserts an Area Qualifier with a 'protected' attribute only. An EPA control inserts an Area Qualifier with 'unprotected' and tab stop attributes. An HTS (ESC H) or CTC (ESC [ W) control inserts an Area Qualifier having the same attributes as the preceding Area Qualifier, with a tab stop attribute added.

When an entry is inserted into the table, no comparison is made with adjacent entries. Adjacent Areas with identical attributes may be written (or inserted), if desired.

DELETING AN ENTRY FROM THE AQ TABLE. Entries may be deleted from the table by erasing the region containing the entries, with ERM (Erasure Mode) set.

An ECH (ESC [ X) control deletes the Area Qualifier at the cursor position, if any. An EA (ESC [ 2 O) control deletes the Area Qualifier associated with the Active Area. An EF (ESC [ 2 N) control deletes all Area Qualifiers in the Active Field. An EL (ESC [ 2 K) control deletes all Area Qualifiers in the Active Line. An ED (ESC [ 2 J) control deletes all Area Qualifiers.

An entry may also become deleted by virtue of removing a tab stop; see Changing an Entry in the AQ Table, above.

REMOVING THE FORM. Simply deleting the last Area Qualifier in the Page does not remove a Form (ie, reset Forms Mode). One of the following explicit controls is required to remove the Form:

A full-Page erase (ESC [ 2 J or ESC [ H ESC [ J) with ERM set, or an RIS (ESC c) control, removes the Form and erases the Page.

A zSDP (ESC [ Pn..Pn p) control removes the Form without erasing the Page.

#### INITIALIZING THE FORM

The host writes the Form to the terminal by sending the necessary string; see Example Form, above. This may be done in either Simultaneous or Monitor SRM. The host retains full control of the terminal, including the ability to write to the display, in Monitor SRM. If done in Monitor SRM, the keyboard should first be disabled (so that inadvertent operator entries are not interspersed), and re-enabled only after the form has been written, and the terminal initialized for operator entry.

The following is a typical string for initializing the Form in Simultaneous SRM:

Form string	-Writes the Form
ESC [ m	-Sets GR-register to Normal rendition
ESC [ 2 J	-Erases all unprotected Areas
ESC [ > 36 h	-Sets Page mode (to inhibit memory scroll)
ESC P >   140   ~[ S ESC \	-Loads SEND key with STS control
ESC [ 12 l	-Puts the terminal into Monitor SRM

Note: The Form initialization string will typically include other controls to tailor the operator's keyboard (and the various modes) to the application. These are discussed in the appropriate sections below.

The terminal is now in Monitor SRM, and the operator may locally enter data into the Form. The terminal's FIFO is not being used by the operator. It may be used by the host to write into the Host Areas or Inactive Memory with the Alternate cursor; see Display section. This is a totally parallel operation and does not interfere, in any way, with the operator's entry, (eg, the host could maintain a real-time clock in the Host Area, without interrupting or slowing down the operator.) The host may also write into the operator's Page, if desired, but should use the DMI/EMI controls to disable/enable the the operator's keyboard to avoid any possibility of conflict. The host may also use the terminal's FIFO to print (while the operator is entering data) using

the zSRC (Remote Copy) control; see Printer section.

NOTE: The Form itself may be written with the Alternate cursor (eg, to set page tab stops in the Host Areas and Inactive Memory for the host's use). Keep in mind, however, that the line numbers referenced by the Alternate cursor differ from those referenced by the Primary cursor, if an Upper Host Area is used; see Display section, Using the Alternate Cursor.

#### ENTERING DATA INTO THE FORM

The operator's input is processed one keystroke at a time; escape and control sequences are not assembled. Thus the operator cannot type in undesired sequences. For example, if the operator types the ESC key, an ESC-code is processed, which does nothing. If the operator follows it with a graphic character, the character is processed and written to the screen. The system programmer can provide the operator with any sequences of controls that are desired for the application, by loading them into Programmable keys as part of the Form initialization string.

Similarly, if there are terminal control keys that the system programmer wishes to deny to the operator (such as RESET, SETUP, BREAK, etc.), these keys may be disabled by loading them with local-only NUL codes in the Form initialization string. Example:

```
ESC P > | 120;0;2 | ~@ | 121 | ~@ | 95 | ~@ ESC \
```

Disables the RESET, SETUP, and BREAK keys.

The operation of the terminal control keys, if not re-programmed, is described in the Keyboard section. The keys operate the same in Forms Mode, with the following exceptions:

Local cursor controls are restricted to unprotected areas. If the cursor is moved into a protected area, it will auto-skip right to the first unprotected character position.

Local erase controls (EA, EF, EL) are constrained to erase only in the Active Area, regardless of the control used. A Local ED control is constrained to erase only unprotected areas, regardless of the setting of ERM.

Local edit controls are constrained to edit only in the Active area, regardless of the setting of SEE.

These exceptions apply to all terminal controls entered from the keyboard during Local Entry, regardless of whether they are the original or re-programmed strings, and regardless of what keys generate them.

The operator will typically use the TAB key to move about the Form. If the Form contains tab stops at the beginning of unprotected Areas, TAB moves the cursor right to the beginning of the next unprotected Area. SHIFT-TAB moves the cursor left to the beginning of the preceding unprotected Area (if the cursor is at the beginning of the Active Area) or to the beginning of the Active Area (otherwise). Attempting to tab past the last unprotected Area (or backtab past the first unprotected Area) results in an error 'beep'.

During operator entry, zHAM (Hold in Area Mode) controls what happens when the operator writes into the last character position of an Area. If zHAM is reset, the cursor auto-skips to the beginning of the next unprotected Area. If zHAM is set, the cursor remains at that position until the TAB key is used to leave the Area.

If the operator attempts to write an alpha character into a Numeric-only area (such as AGE and WAGE in the Example Form), an error 'beep' results. When the operator terminates entry into a Right-Justify area (such as WAGE in the Example Form), with the TAB key, the entry moves right to the end of the area. When the operator writes into a Concealed area (such as CODE in the Example Form), the characters are not displayed (but 'white' Spaces are displayed instead, as markers).

The operator may correct an error before leaving an Area, or backtab to the Area to correct it. An entry (or part of it) may be erased with the Spacebar, with BACKSPACE (if zDBM is set), or with the ERASE key. An entry may be edited with the EDIT, INSERT, and DELETE keys. In all cases, only the Active Area is affected.

#### SENDING THE ENTERED DATA

When the entered data are ready to transmit, the operator depresses the SEND key. This disables the keyboard, and sends the data to the host. The keyboard is automatically re-enabled upon completion of the transmission, unless zAKDM (Auto Keyboard Disable Mode) is set, whereupon it remains disabled until purposefully re-enabled by the host (eg, with the EMI control, ESC b). This permits the host to 'protect' the displayed data (from operator actions) until it has performed whatever validity checks are desired.

The operation of the Send controls and modes is described in the Send Controls section. Their operation is the same in Forms Mode. Examples:

The format of the Example Form was:

```
FORM 1 NAME aaaaaaaaaaaaa AGE nn WAGE nnnn.nn CODE aaaaaa NOTES aaa...
```

Assume that it was filled out by the operator as shown below, the symbol '-' being used to represent an unprotected Space. Note: The characters shown in the Area following CODE are the characters entered, not displayed. The characters, A789, are displayed as 'white' spaces; the two characters that follow them are displayed as 'black' spaces.

```
FORM#1 NAME John Doe----- AGE 37 WAGE --12.34 CODE A789-- NOTES None
```

In the examples below, spaces are shown between the transmitted characters for clarity; they are not part of the transmission. The symbol '-' is used to show the transmission of a Space (2/0) from an unguarded character position. The symbol '+' is used to show the transmission of a Space (2/0) from a guarded position.

With the Transfer modes, FETM, GATM, zFSTM, and zGRTM reset, the content of the transmission is shown below. NL indicates a new-line code, as selected by LNM: CR LF (if LNM is reset) or LF (if LNM is set). An NL is included in the transmission at the end of every line (including partial lines) in the

transmitted region, regardless of whether the line contains data, is guarded, etc. The present example assumes that TTM is reset (ie, transmit to cursor) and that the cursor is in the top line (following the data).

```
F O R M # 1 J o h n - D o e - - - - 3 7 - - 1 2 . 3 4 A 7 8 9 - - N o n
e NL
```

Setting FETM causes the new-line codes to be NOT included in the transmission:

```
F O R M # 1 J o h n b D o e - - - - 3 7 - - 1 2 . 3 4 A 7 8 9 - - N o n
e
```

Setting GATM causes all Areas (including those designated as Guarded) to be included in the transmission:

```
F O R M # 1 + N A M E + J o h n - D o e - - - - + A G E + 3 7 + W A G E
+ - - 1 2 . 3 4 + C O D E + A 7 8 9 - - + N O T E S + N o n e NL
```

Setting zFSTM causes trailing Spaces to be suppressed (and HT codes inserted to mark their omission) in the transmission:

```
F O R M # 1 HT J o h n - D o e HT 3 7 HT - - 1 2 . 3 4 HT A 7 8 9 HT NL
```

Setting zGRTM causes Graphic Rendition information (in the form of SGR controls) to be included in the transmission: Note: In this example, GATM is also set, so as to illustrate multiple rendition changes. It also illustrates how a Form could be created and edited by the operator on the screen, the host software translating the SGR controls into the appropriate DAQ controls in a prescribed manner.

```
ESC [ 7 m F O R M # 1 + N A M E + ESC [ m J o h n - D o e - - - - ESC [
7 m + A G E + ESC [ m 3 7 ESC [ 7 m + W A G E + ESC [ m - - 1 2 . 3 4 ESC
[ 7 m + C O D E + ESC [ m A 7 8 9 - - ESC [ 7 m + N O T E S + ESC [ m N o
n e NL
```

The preceding examples assumed that no Header/Trailer strings were programmed. If they had been, the Header string would have preceded the transmission shown, and the Trailer string would have followed it. These strings may be used to bracket the transmissions with control codes or data needed, or desired, by the host.

#### CORRECTING AND RE-SENDING

If the host does validity checking of the transmitted data, zAKDM should be set (to 'protect' the displayed data from inadvertent operator actions until the validity checks are completed).

If there was an error in the transmission of the data, the host may send the STS (ESC S) control to re-transmit the data.

If there was an error in the entry of the data, the host, if desired, may use the zGUA control, so that after correction, only the corrected data are included in the re-transmission.

zGUA GUARD UNPROTECTED AREAS (ESC :). Causes all unprotected Areas in the Page to become guarded. Thereafter, any operator entry into an Area causes that Area to become unguarded.

Example: In the zFSTM example above, the transmission was:

```
F O R M # 1 HT J o h n - D o e HT 3 7 HT - - 1 2 . 3 4 HT A 7 8 9 HT NL
```

Suppose that the actual transmission contained 97, in the AGE Area (instead of 37), and that the host finds that to be an invalid entry:

```
F O R M # 1 HT J o h n - D o e HT 9 7 HT - - 1 2 . 3 4 HT A 7 8 9 HT NL
```

The host must inform the operator that there is an error, and where. This may be done via a Host Area; however, it may also be done with the following string:

```
ESC [ 5 m      -Sets the GR-register to Blinking
HT             -Moves the cursor to the next tab stop (the error Area)
ESC 9         -Causes the 97 to blink
ESC [ m      -Restores the GR-register to Normal
ESC :         -Guards all unprotected Areas
ESC b        -Re-enables the keyboard
```

The operator now sees a blinking Area on the screen and knows (by pre-arrangement) to correct it. As the operator overwrites the entry, it stops blinking (because the GR-register was restored to Normal). When the operator depresses the SEND key, the resulting transmission will contain:

```
F O R M # 1 HT HT 3 7 HT HT HT NL
```

If the host accepts the corrected entry as valid, it may re-initialize the Form (for the next set of entries) as follows:

```
ESC [ 6 l      -Resets ERM (erase unprotected)
ESC [ 2 J      -Erases the Page
ESC b         -Re-enables the keyboard
```

Erasing the Page indicates to the operator (by pre-arrangement) that the host is ready for a new set of entries. It also restores any guarded unprotected Areas to unguarded.

In the last string, the Reset ERM control may be omitted if the host resets ERM in the Form initialization string, and leaves it reset,

In the preceding string, if the host had used the Alternate cursor to send the string, there would have been no need to set and restore the GR-register. The Alternate cursor has its own GR-register, which could have been preset to Blinking in the Form initialization string. That would result in erasing the screen to blinking Spaces, but that is not discernible to the operator, nor to the host (unless zGRTM is used).



## PRINTING THE ENTERED DATA

The operation of the Print controls and modes is described in the Printer section. Their operation is the same in Forms Mode.

In a data entry situation, printing will normally be done by the host, using Remote Copy, during the periods while the operator is entering data; see Initializing the Form, above.

The Form may also be used, with Page Print, simply to obtain a formatted printout. Using the example entry again (the symbol '-' still being used to represent an unprotected Space):

```
FORM#1 NAME John Doe----- AGE 37 WAGE --12.34 CODE A789-- NOTES None
```

In the examples below, spaces are shown between the transferred characters for clarity; they are not part of the transfer. The symbol '-' is used to show the transfer of a Space (2/0) from an unguarded character position. The symbol '+' is used to show the transfer of a Space (2/0) from a guarded position.

With zGAPM reset, the characters in guarded Areas are transferred as Spaces. The transfer to the printer is illustrated below. NL indicates a new-line code, as selected by the setting in Setup line C: CR, LF, or CR LF. An NL is included in the transfer at the end of every line (including partial lines) in the transferred region, regardless of whether the line contains data, is guarded, etc. The present example assumes that TTM is reset (ie, transfer to cursor) and that the cursor is in the top line (following the data).

```
F O R M # 1 + + + + + J o h n - D o e - - - - + + + + + 3 7 + + + + +
+ - - 1 2 . 3 4 + + + + + A 7 8 9 - - + + + + + N o n e NL
```

With zGAPM set, all characters are transferred, regardless of their guarded status. The transfer to the printer is illustrated below.

```
F O R M # 1 + N A M E + J o h n - D o e - - - - + A G E + 3 7 + W A G E
+ - - 1 2 . 3 4 + C O D E + A 7 8 9 - - + N O T E S + N o n e NL
```

Given a Form, eg, a Telephone Contact report with protected headings like Name, Company, Subject, Action Required, etc, the data would probably be printed with zGAPM set (so that both the data and the headings are printed). However, if the Form (in the terminal) matched a pre-printed Form (on the printer), eg, an Invoice form, the data would probably be printed with zGAPM reset (so that only the data are printed). The headings on the terminal's Form would be transferred as Spaces, merely spacing over the pre-printed headings on the printer's Form.



CONTENTS

Page

DEC CONTROLS ..... 1

DEC CONTROLS

Several DEC private-use (ie, non-ANSI) functions have been added to the terminal to facilitate its use with standard DEC software. The added controls and modes are described below. Note: DEC is a registered trademark of the Digital Equipment Corporation.

dANM VT52/ANSI MODE (ESC [ ? 2 1). Puts the terminal into VT52 mode. The mode is displayed near the end of Line D. It can be changed in Setup mode but the reset (VT52) state cannot be 'Saved', ie, the terminal always comes up in ANSI mode.

In the RESET state, the terminal responds only to VT52-compatible codes, as follows:

CONTROL	CODE	CONTROL	CODE
Cursor Up	ESC A	Home	ESC H
Cursor Down	ESC B	Reverse Line Feed	ESC I
Cursor Right	ESC C	Erase to End of Page	ESC J
Cursor Left	ESC D	Erase to End of Line	ESC K
Select Graphics	ESC F	Direct Cursor Address	ESC Y l c
Select ASCII	ESC G	Exit VT52 Mode	ESC <

In the SET state, the terminal responds to the normal Ambassador control set.

NOTE: The following string may be used to load the keys with VT52 controls; see Programmable Strings section.

```

ESC P > 0
| 106; 1 | ~[ F | | ~[ B | | ~[ G
| 111    | ~[ D | | ~[ H | | ~[ C
| 117    | ~[ A
| 141    | ~[ J | ;0 | ~[ K
ESC \
    
```

dOM ORIGIN MODE (ESC [ ? 6 Fc). Sets/resets the mode according to the final character Fc: 6/8 (h) sets it, 6/12 (l) resets it. The mode is displayed near the end of Line D. It can be changed (and saved) in Setup mode.

In the RESET state, the origin is the first character position within the display Page. Line numbers are independent of the current top/bottom margin settings.

In the SET state, the origin is the first character position of the line designated to be the top margin; see dSTBM. Line numbers are relative to the top margin setting.

dSTBM SET TOP AND BOTTOM MARGINS (ESC [ Pn1 ; Pn2 r). Sets top and bottom margins according to the parameters to define a scrolling region within the display page. Pn1 is the line number of the top line of the scrolling region. Pn2 is the line number of the bottom line of the scrolling region. Default is the entire page (ie, no margins).

SAMPLE VT100 SETUP. The terminal may be programmed to execute a sequence of commands automatically at power-on; see Programmable Strings section. This may be used to initialize the terminal. Example:

```
ESC [ 2 4 ; ; ; 2 4 p
This string limits the Page size to 24 lines.
```

```
ESC P > | 122; 0 | ~[ O P | | ~[ O Q | | ~[ O R | | ~[ O S ESC \
This string programs the F1, F2, F3, F4 keys to match the VT100.
```

```
ESC P > | 153 | ~[ [ ? 1 ; 2 c ESC \
This string programs the terminal's DA response to match the VT100.
```

These strings may be loaded into the power-on string as follows.

```
ESC P > | 200 |
~[ [ 2 4 ; ; ; 2 4 p
~[ P > ~| 122 ~| ~[ O P ~| ~| ~[ O Q ~| ~| ~[ O R ~| ~| ~[ O S ~[ \
~[ P > ~| 153 ~| ~[ [ ? 1 ; 2 c ~[ \
ESC \
```

If VT52 mode is desired at power-on, the dANM control, ESC [ ? 2 1, may be included in the power-on string, but following all ANSI controls. The ANSI controls will not function after VT52 mode is selected.

CONTENTS	Page
INTRODUCTION .....	1
DCS CONTROL .....	1
KEY STRING EXAMPLES .....	3
OPERATIONAL STRINGS .....	6
OPERATIONAL STRING EXAMPLES .....	6
POWER-ON STRING .....	7
POWER-ON STRING EXAMPLES .....	7
SIZE LIMITATIONS .....	9
LOCAL STRINGS .....	10
SPECIAL NOTES .....	10
TABLE 1: DCS CONTROL SUMMARY .....	11
TABLE 2. DEFAULT STRING PROGRAMMING .....	12

## INTRODUCTION

The XL Series provides an exceptionally versatile programmable string capability;

**PROGRAMMABLE KEYS.** All of the keys on the keyboard are programmable, ie, you can load frequently used character strings into them, and thereafter generate the strings with a single keystroke. Each key is programmable on many levels. Levels 0-3 are pre-defined as unshifted, SHIFT, CTRL, and CTRL-SHIFT. The 48 graphic character keys on the Main keypad are not programmable on these levels. All other keys are. Additional levels may be defined by the user by designating desired keys to be shift keys. The added levels are fully programmable, ie, you can embed strings beneath the graphic character keys.

**PROGRAMMABLE OPERATIONAL STRINGS.** These strings are generated upon the occurrence of certain events (eg, start or end of transmission, receipt of an ENQ from the host, etc.). They can be programmed as needed to tailor the terminal to your operational requirements.

**PROGRAMMABLE POWER-ON STRING.** This string is stored in non-volatile memory and generated each time the terminal is powered on. It can be used to initialize the terminal as desired.

These strings may be loaded or changed by the user with the DCS control.

## DCS CONTROL

The DCS control is used to load the programmable strings. The DCS control (ESC P) indicates the beginning of a device control string. The device control string is limited to graphic characters in the range Sp (2/0) to 7/14 (~) inclusive, starting with the first character after the DCS control. If control codes are received during assembly of the string, they are deleted from the string, executed normally, and assembly continues. The device control string is terminated by the ST control (ESC \).

ESC P <device control string> ESC \

If the first character of the device control string is > (3/14), it is interpreted as an XL control string.

```
ESC P > <XL control string> ESC \
```

NOTE. Pre-XL versions of the terminals used a device control string in which the first character was a ` (6/0) or p (7/0). These strings are still supported in the XL Series for the convenience of existing users.

The XL control string contains instructions to load one or more of the programmable strings. Each instruction consists of a Control field, delimited by | (7/12), followed by a Text field. The Control field specifies the programmable string to be loaded and its attributes. The Text field contains the data to be loaded into the string.

```
ESC P > Ps0 | Control | Text | Control | Text .... ESC \
```

The parameter, Ps0, at the beginning of the XL control string is a global control parameter. If it is 0 (3/0), all programmable strings are returned to their default value before executing the instructions that follow. If omitted, the present contents of the strings are unchanged.

CONTROL FIELD. The Control field contains 5 parameters, delimited by ; (3/11). The parameters are initialized to 0 (3/0) at the beginning of the XL control string. Omission of a Control field parameter results in no change in the value of that parameter. Trailing semicolons in the field are not significant, and may be omitted.

```
ESC P > Ps0 | Ps1 ; Ps2 ; Ps3 ; Ps4 ; Ps5 | Text .... ESC \
```

KEY NUMBER (Ps1). Specifies the programmable string to be loaded; see Keyboard diagram, Table 1. An empty Control field increments to the next valid key number.

KEY SHIFT LEVEL (Ps2) For key strings, specifies the shift level. Valid values are levels 0-31, and 64. Levels 0-3 are pre-defined as unshifted, SHIFT, CTRL, and CTRL-SHIFT respectively. Levels 4-31 may be defined by the user with the Key shift attribute parameter, Ps5; see below. All keys are programmable on all levels, except that the 48 graphic character keys on the Main keypad are not programmable on levels 0-3. Level 64 defines the key as an algorithmic META shift key; see Default Programming, S1. When the terminal is in Meta Key mode (zMKM), this key causes the code(s) generated by other keys to be transmitted with their parity bit (8th-bit) equal to 1 (key depressed) or 0 (not depressed) regardless of the Setup parity settings.

TRANSMIT ATTRIBUTE (Ps3). Specifies the destination of the string when activated. Valid values are 0-3. A value of 0 (3/0) causes the the string to be sent according to Send-Receive mode (SRM), ie, sent to the host in Simultaneous SRM and to the terminal's display processor in Monitor SRM. A value of 1 causes it to be sent to the host only, and a value of 2 to the display processor only, regardless of the setting of SRM. A value of 3 is applicable to GXL users only. It causes the string to be sent locally to the graphics display processor only.

KEY REPEAT ATTRIBUTE (Ps4). For key strings, specifies whether the string is to be repeatedly generated if the key is held down. A value of 0 (3/0) causes the string to be repeated according to the setting of Key Repeat mode (zKRM). A value of 1 causes it to not repeat, and a value of 2 to always repeat, regardless of the setting of zKRM.

KEY SHIFT ATTRIBUTE (Ps5). For key strings, a value of 1 (3/1) designates the key as a shift key accessing the shift level specified by the Key shift level parameter, Ps. For example, the Control field string, | 1; 4; ; ; 1 |, designates the S2 key (key #1) as a shift key accessing shift level 4. The shift levels are additive, ie, SHIFT-S2 will access any strings stored in shift level 5, CTRL-S2 will access strings stored in shift level 6, and CTRL-SHIFT-S2 will access strings stored in shift level 7. More than one key may be designated to access a level. For example, the S3 key may be made to perform the same function as the 3-key depression, CTRL-SHIFT-S2, by designating it a shift key accessing level 7. Any key may be designated as a shift key. The shift designation overrides any key programming that may be resident in the key. For example, the 0 key on the Control keypad, which normally generates a 0 (3/0) code on level 0, may be designated a shift key without disabling the default programming. Of course, the key will no longer generate the 0 code. A key cannot be both a shift key and a code generator.

TEXT FIELD. Characters in the text field are limited to ASCII codes in the range 2/0 (Sp) 7/14 (~). Control codes, for example CTRL-M, must be represented as the two-character sequence, ~M. DEL must be represented as the two-character sequence, ~?.

Because the terminal uses the characters | (7/12) and ~ (7/14) as special control characters, they may not be used directly in the Text field. Instead, the ~ (7/14) is also used as an escape character, and the two special characters represented by the two-character sequence ~| or ~~ respectively.

An empty Text field returns the specified Key number/Shift level to its default programming, if any; see Table 2. To disable a key string that has default programming, it is necessary to load it with a local-only string that is not recognized by the terminal, eg, CTRL-@.

#### KEY STRING EXAMPLES

The following examples may be entered locally by going into Local Test (by typing the 2-key sequence SETUP T), and then typing in the example strings. The entry may be observed on the Monitor line as it is typed. ESC denotes the ESC-key, and is shown on the Monitor line as a reverse left-bracket ([).

The spaces shown between characters are for clarity (and should not be included in the string). Likewise, some strings are broken into multiple lines for purpose of annotation (and the new-lines should not be included in the string).

#### Example #1

```
ESC P > 0 | 122; 0; 0; 0; 0 | Hello ESC \  
This DCS sequence loads F1 (key number 122) with the string, Hello, ie,  
the word, Hello, is generated each time the F1 key is typed. The first  
zero in the control field indicates the shift level, in this case, un-
```

shifted. A zero for the transmit attribute indicates that the key should follow the setting of SRM (Send Receive Mode). The next parameter in the control field is for the repeat attribute, a zero indicates that the key should follow the setting of zKRM Keyboard Repeat Mode. The last parameter in the control field is the shift-key attribute. A zero indicates that F1 is to be a normal keystroke, not a shift key. The "zero" entries are not required, but have been included in this case for clarity. The components of the string are as follows:

```

ESC P >      -Start of a DCS string
  0          -Return strings to default
  |          -Start of a control field
  122 ;      -Index to the F1 key followed by separator
  0 ;        -Shift level, 0=unshifted followed by separator
  0 ;        -Transmit attribute, 0=follow SRM followed by separator
  0 ;        -Repeat attribute, 0=follow zKRM followed by separator
  0          -Shift-key attribute, 0=normal keystroke
  |          -End of a control field
  Hello      -Text field - Data portion of the string table entry
ESC \        -String terminator to end the DCS string

```

#### Example #2

```

ESC P > | 122; 1; 0; 2; 0 | Hello ESC \

```

This DCS sequence also loads F1 with the string, Hello, but with a change in shift level and repeat attribute. F1 will generate 'Hello' on the shifted level since we have included a '1' in the shift level and it will repeat while the key is held down because of the '2' as the repeat attribute, which says to repeat always.

#### Example #3

```

ESC P > | 123; 0; 0; 2; 0 | Ann Arbor ~M ~J ESC \

```

This DCS causes F2 (key number 123) to generate the words, Ann Arbor, followed by a carriage return CR (~M) and a line feed LF (~J). The '0' shift level programs the string into F2, unshifted. The '2' repeat attribute makes the key repeat when held down.

#### Example #4

```

ESC P > | 124; 3; 0; 0; 0 | Terminal | 3; 3; 0; 0; 1 | ESC \

```

This DCS sequence addresses two keys within the same string. It first programs F3 (key number 124) on shift level 3, to generate the word, Terminal. The second portion of the string programs S3 (key number 3) to be a shift key accessing level 3. Level 3 corresponds to CTRL-SHIFT; see Table 1. You can now generate the string with either CTRL-SHIFT-F3 or S3-F3.

At this point, if you have followed each example, you have programmed the following keys:

```

F1           Hello
SHIFT-F1     Hello (repeating)
F2           Ann Arbor CR LF (repeating)
S3-F3        Terminal

```



## Example #5

```
ESC P > | 111; 0; 0; 0; 0 | ~[ [ 60 ; ; ; 24 p ESC \
```

This DCS causes F18 (key number 139) to generate a Set Display Parameters (zSDP) control, ESC [ 60 ; ; ; 24 p, to set the terminal for a Page of 60 lines and a Window of 24 lines. Note that the ESC is represented by a ~[.

## Example #6

```
ESC P > | 122; 0; 0; 0; 0 | 123; 0; 0; 0; 0 | ESC \
```

This DCS causes F1 and F2 (key numbers 122 and 123) to return to their default values on the unshifted level, which was null; see Table 2.

Thus far the examples have shown complete entries in the control field. This is not necessary when key strings are addressed sequentially and attributes remain the same. The following examples highlight multiple key string entries.

## Example #7

```
ESC P > | 122 | Key 1 | | Key 2 | | Key 3 | | Key 4 ESC \
```

This DCS sequence begins by addressing F1 (key number 122). Note that the control field ends after the key number. All five parameters in the control field are initialized to zero at the start of the DCS string, so the remainder of the control field, by default, is set for '0' shift level = unshifted, '0' transmit attribute = follow SRM, '0' repeat attribute = follow zKRM, and '0' shift attribute = normal keystring. The text field contains 'Key 1', so that each time F1 is typed 'Key 1' is generated. The double vertical bar following the first text field automatically increments the Key number, which is F2 (key number 123), and programs 'Key 2' to be generated each time F2 is typed. The attributes remain the same as for F1. In all, this string programs F1 thru F4.

## Example #8

```
ESC P > | 122; 3; ;2 | KEY A | | KEY B | | KEY C | | KEY D ESC \
```

This DCS sequence loads F1 (key number 122) on shift level 3; see Table 1. Following the shift level are two semi colons. This is equivalent to a '0' for the transmit attribute. This is followed by a '2' for the repeat attribute (always repeat). The shift attribute does not need to be sent as the default '0' = normal keystring is desired. The text field contains 'KEY A', so that each time the key is typed (using CTRL-SHIFT, or a key programmed programmed to be a shift key on level 3), The double vertical bar following this text field automatically increments the key number to F2, and programs 'KEY B' into it. F2 will be on the same shift level, 3, with the same repeat attribute, 2, since no change was made to the control field. In all, this string programs F1 thru F4, on shift level 3, with the repeat-always attribute, regardless of the setting of zKRM.

## Example #9

```
ESC P > | 131; 0 | LEVEL 0 | ; 1 | LEVEL 1 | ; 2 | LEVEL 2 ESC \
```

This DCS sequence loads F10 (key number 131) on shift level 0 with no change from the '0' defaults to the rest of the control field. The key will generate 'LEVEL 0' when the key is depressed. The control field immediately following retains the same key number but changes the shift lev-

e1 to 1. The text field contains 'Level 1'. The next control and text field program F10 on shift level 2. F10 now generates 'LEVEL 0', SHIFT-F10 'LEVEL 1', and CTRL-F10 'LEVEL 2'.

#### Example #10

```
ESC P > | 122 | || || || ESC \
```

This DCS sequence restores F1 thru F4 to their default values on the unshifted level. Remember that the double vertical bar increments the key number.

#### Example #11

```
ESC P > | 0; 1; ; ; 1 | | 1; 2 | | 3; 3 | | 4; 4 | | 5; 8 | | 5; 16 |
ESC \
```

This DCS sequence program S1 thru S6 to be shift keys on levels 1 thru 16. S1 is programmed to be a shift key on level 1, S2 to be a shift key on level 2, etc.

### OPERATIONAL STRINGS

The terminal provides the following operational strings. The strings can be loaded, changed, and disabled with the DCS control.

**HEADER.** The Header string precedes all transmissions from the terminal's Display Memory to the host. It is initialized to null (ie, no codes) at power-on.

**TRAILER.** The Trailer string follows all transmissions from the terminal's Display Memory to the host. If a transmission is aborted, the trailer is still sent. It is initialized to null at power-on.

**ENQ.** The ENQ string is sent in response to a received ENQ (0/5) code. It is initialized to null at power-on.

**DA.** The DA string is sent in response to a received DA (ESC [ c) control. It is initialized at power-on with the release-revision-options number of the terminal's firmware.

### OPERATIONAL STRING EXAMPLES

The following examples may be entered locally by going into Local Test (by typing the 2-key sequence SETUP T), and then typing in the example string. The entry may be observed on the Monitor line as it is typed, ESC denotes the ESC-key, and is shown on the Monitor line as a reverse left-bracket ([). The spaces shown between characters are for clarity (and should not included in the string).

#### Example #1

```
ESC P > | 153; 0; 0 | ~[ [ ? 1 ; 2 c ESC \
```

This DCS sequence loads the DA response (key number 153) so that it will match the DA response of the DEC VT100 (ESC [ ? 1 ; 2 c). Note that the shift level is a zero, which is the only valid entry for this parameter in

an Operational String. The transmit attribute may be specified as required. To execute this string while in Local Test, type ESC [ c. The new DA response will appear in the Monitor line.

#### Example #2

```
ESC P > | 153 | ESC \
```

This DCS sequence disables the DA response programmed above and restores it to its default value; see Table 2.

#### POWER-ON STRING

The terminal provides a Power-on string that permits a string of controls to be loaded into non-volatile (ie, permanent) memory, to be automatically executed each time the terminal is powered-on. This string may be used to initialize the terminal to a desired state, eg, to initialize key codes, tab stop positions, or screen format, or just to display initial instructions.

The Power-on string is loaded with the DCS Control using key number 200. All other parameters must be zero. Loading the Power-on string overwrites any previously loaded Power-on string.

The text field contains the actual programming instructions for the terminal. All of the instructions must be included in the one text field. The only limitation to the number of functions that may be programmed into the string is the size of the non-volatile memory; see Size Limitations.

All of the rules set forth previously in programming key and operational strings apply.

#### POWER-ON STRING EXAMPLES.

The following examples may be entered locally by going into Local Test (by typing the two-key sequence SETUP T), and then typing in the example strings. The entry may be observed the Monitor line as it is typed. ESC denotes the ESC-key, and is shown on the Monitor line as a reverse left-bracket ([).

The spaces shown between characters are for clarity (and should not be included in the string). Likewise, some strings are broken into multiple lines for purpose of annotation (and the new-lines should be not included in the string).

The example strings may be executed (to demonstrate their result) without having to power down, by reinitializing the terminal (by typing the two-key sequence SETUP Z). The string can also be executed without having to power down, by depressing CTRL-SHIFT-RESET (or just RESET, if the terminal is in Local Test). The string is executed if the RESET key is just momentarily typed; it is not if the RESET key is held down; see Diagnostics section.

#### Example #1

```
ESC P > | 200 | Good Morning ESC \
```

This DCS loads the Power-on string with the message 'Good Morning'. Each time the terminal is powered on, this message will be displayed on the

screen. The components of the string are as follows:

ESC P >	-DCS
	-Start of a control field
200	-Index to program a power-on string
	-End of the control field
Good Morning	-Text field - characters to be executed
ESC \	-ST control to end

#### Example #2

```
ESC P > | 200 | Good Morning ~[ [ 24 ; ; ; 24 p ESC \
```

This DCS loads the Power-on string with the previous example, plus the zSDP (Set Display Parameters) control, ESC [ 24 ; ; ; 24 p, which when executed will display the message 'Good Morning' and set the Page and Screen size to 60 and 24 lines respectively. Note that, because of the DCS coding restrictions, the ESC must be loaded as a tilde left-bracket (~[).

#### Example #3

```
ESC P > | 200 | ~[ P > ~| 122; 0; 0; 0; 0 ~| Hello ~[ \ ESC \
```

This DCS loads the Power-on string with a DCS string to program F1 to generate the word 'Hello' each time the F1 key is typed. This Power-on string has a DCS string within it. The DCS string to program F1 is:

```
ESC P > | 122; 0; 0; 0; 0 | Hello ESC \
```

Note that, because of DCS coding restrictions, the ESC's in this string must be loaded as tilde left-bracket (~[) and the vertical bar must be preceded by a tilde to indicate that the next character is to be included as part of the text. The entire DCS string to load F1 is the text field of the Power-on string.

#### Example #4

```
ESC P > | 200 | ~[ P > ~| 122 ~| Hello ~[ \ ESC \
```

This DCS is the abbreviated form of Example #3. Following the key index for F1, the remainder of the control field contains all zeros for the shift level and attributes, which are by definition, the default states, and so may be eliminated. See Examples 7-11 in Key String Examples.

#### Example #5

```
ESC P > | 200 | ~[ P > ~| 123; 0; 0; 0; 0 ~| ANN ARBOR ~M ~J  
~[ \ ESC \
```

This DCS loads the Power-on string with a DCS string to program F2 to generate the words 'ANN ARBOR' followed by a CR (Carriage Return) and LF (Line Feed) each time the F2 key is typed. The DCS string to program F2 is:

```
ESC P > | 123; 0; 0; 0; 0 | ANN ARBOR ~M ~J ESC \
```

Note that, because of DCS coding restrictions, the ESC's in this string must be loaded as tilde left-bracket (~[) and the vertical bars, and the tildes must be preceded by a tilde to indicate that the next character is to be included as part of the text. The entire DCS string to load F2 is

the text field of the Power-on String.

#### Example #6

```
ESC P > | 200 | ~[ P > ~| 123 ~| ANN ARBOR ~M ~J ~[ \ ESC \
```

This DCS string is an abbreviated form of Example #5; see also Examples #3 and #4.

#### Example #7

```
ESC P > | 200|
  Good Morning
  ~[ [ 2 4 ; ; ; 2 4 p
  ~[ P > ~| 122 ~| Hello
  ~| 123 ~| ANN ARBOR ~M ~J
  ~[ \
ESC \
```

The Power-on string may be programmed to perform more than one function. In this example, the DCS programs the Power-on string to perform all of the functions of the three previous examples #2, #4 and #6. The only limitation to the number of functions that may be programmed into the string is the size of non-volatile memory; see Size Limitations, below.

## SIZE LIMITATIONS

There is a size limitation on the total length of the programmable strings that may be stored by the terminal. For the Key and Operational strings, the total memory allotment is 896 bytes.

A string requires 1 byte per character in the Text field (after converting any tilde-shifted characters to actual). In addition, the Control field requires 1 byte if any of its parameters (except key number) are other than zero. For example, consider the DCS string used to program the F1 key to repeat the characters, Hello: ESC P > | 94 ; 0 ; 0 ; 2 ; 0 | Hello ESC \

After executing this DCS, the terminal will have assigned 6 bytes of memory to the F1 key -- one for each of the characters, Hello, plus one for the repeat attribute.

The default strings shown in Table 2 do not initially require string table space. If they are re-programmed, even if they are re-programmed to their default values, they do take up table space, unless they are returned to their default values by addressing a control field with no text field.

The Power-on string has its own 255-byte non-volatile memory. This count applies only to the characters actually executed at power-on. As with the other strings, the codes 7/12 (|), 3/11 (;) and 7/14 (~) (in the DCS which loads the memory) do not take up space in memory.

**CAUTION:** At each power-on, the Power-on string is copied into the terminal's input buffer for execution. If the Power-on string is near its maximum length of 255 characters, extra input from the host (such as a power-on log-in message), may cause the buffer to overflow, and the string will not execute. If this occurs, the Power-on string may always be executed by

typing the two-key sequence, SETUP Z; see Diagnostics section.

#### LOCAL STRINGS.

There are several strings encoded into the default programming (Table 2) that perform local operations within the terminal. These are summarized below. They can be disabled or moved to other key/levels with the DCS control.

SETUP	ESC d	-Enter Setup mode (see Note 3 below)
BREAK	ESC e	-Send short break
SHIFT-BREAK	ESC f	-Send long break
PAUSE	ESC g	-Admit a new line of data to the screen
SHIFT-PAUSE	ESC h	-Admit a new window of data to the screen
CTRL-PAUSE	ESC j	-Turn on Auto Pause mode (zAPM)
CTRL-SHIFT-PAUSE	ESC m	-Turn off Auto Pause mode (zAPM)
SHIFT-PRINT	ESC k	-Toggle Local Copy
ZOOM-UP ARW	ESC n	-Add lines into the window
ZOOM-LT ARW	ESC q	-Add columns into the window
ZOOM-DN ARW	ESC o	-Remove lines from the window
ZOOM-RT ARW	ESC p	-Remove columns from the window
SCROLL-LT ARW	ESC r	-Move window left 1 column
SCROLL-RT ARW	ESC s	-Move window right 1 column
SHIFT-SCROLL-LT ARW	ESC t	-Move window left 1 window
SHIFT-SCROLL-RT ARW	ESC w	-Move window right 1 window
SHIFT-SCROLL-DN ARW	ESC J	-Move window down 1 window
SHIFT-SCROLL-UP ARW	ESC i	-Move window up 1 window

#### SPECIAL NOTES

1. The CAPS LOCK key and CTRL-SHIFT-RESET are not programmable.
2. The RESET, SETUP, and BREAK keys are not affected by the mode KAM (Keyboard Action Mode) or the controls DMI/EMI (Disable/Enable Manual Input). These keys may be disabled only by loading them with a local-only control character not recognized by the terminal.
3. The SETUP key may be disabled and re-enabled with the DCS control, but its function may not be moved to a different key position. It is the position of the key, and not its code, that is significant in exiting from Setup.
4. NUL-codes (0/0) cannot be loaded into the Power-on string.

TABLE 1: DCS CONTROL SUMMARY.

Reset 120	SetUp 121	F1 122	F2 123	F3 124	F4 125	F5 126	F6 127	F7 128	F8 129	F9 130	F10 131	F11 132	F12 133	F13 134	F14 135
Esc 80	! 81	@ 82	# 83	\$ 84	% 85	^ 86	& 87	* 88	( 89	) 90	- 91	= 92	~ 93	94	Break 95
Tab 60	Q 61	W 62	E 63	R 64	T 65	Y 66	U 67	I 68	O 69	P 70	{ 71	}	Line Feed 73	Del 74	
Caps Lock 40	Ctrl 41	A 42	S 43	D 44	F 45	G 46	H 47	J 48	K 49	L 50	: 51	" 52	Return 53	Back Space 54	
Pause 20	Shift 21	Z 22	X 23	C 24	V 25	B 26	N 27	M 28	< 29	> 30	? 31	Shift 32	Scroll 33	Zoom 34	
S1 0	S2 1										S3 2	S4 3	S5 4	S6 5	6

F15 136	F16 137	F17 138	F18 139	Send 140
Erase 141	Edit 142	Delete 143	Insert 144	Print 145

Tab 110	7 116	↑ 8 117	9 118	- 119
	← 4 111	Home 5 112	→ 6 113	+ 114
	1 106	↓ 2 107	3 108	Enter 109
	∅ 100	∅∅ 102	· 103	104

OTHER STRINGS: Header (150) Trailer (151) ENQ (152) DA (153) Power-on (200)

The DCS control for downloading strings to the XL is of the form:

ESC P > 0 | Key # ; Shift level ; Transmit ; Repeat ; Shift | Text ESC \

KEY NUMBER

From keyboard diagram

SHIFT LEVEL

- 0 - 3 = Unshift, SHIFT, CTRL, CTRL-SHIFT
- 4 -31 = User defined
- 64 = Meta shift

TRANSMIT ATTRIBUTE

- 0 = Follow SRM
- 1 = Transmit only
- 2 = Local only

REPEAT ATTRIBUTE

- 0 = Follow zKRM
- 1 = Never repeat
- 2 = Always repeat

SHIFT ATTRIBUTE

- 0 = Code-generating key
- 1 = Shift key

SPECIAL CHARACTERS

- | = Control field delimiter
- ~ = Control shift

Note: When used in Text field, these character must be escaped: | = ~| ~ = ~~.

TABLE 2: DEFAULT STRING PROGRAMMING

XL CONTROL STRING	KEY	FUNCTION
0; 64; 0; 0; 1	S1	
20; 0; 2; 1; 0	PAUSE	-Admit new line
; 1; 2; 1; 0		-Admit new window
; 2; 2; 1; 0		-Set zAPM
; 3; 2; 1; 0		-Reset zAPM
21; 1; 0; 0; 1	SHIFT (left)	
32; 1; 0; 0; 1	SHIFT (right)	
33; 30; 0; 0; 1	SCROLL	
34; 29; 0; 0; 1	ZOOM	
41; 2; 0; 0; 1	CTRL	
53; 0; 0; 0; 0	RETURN	-See CR
; 1; 0; 0; 0		
54; 0; 0; 0; 0	BACK SPACE	-See BS
; 1; 0; 0; 0		
60; 0; 0; 2; 0	TAB	-See HT
; 1; 0; 2; 0		-See CBT
73; 0; 0; 0; 0	LINE FEED	-See LF
; 1; 0; 0; 0		
74; 0; 0; 0; 0	DEL	
; 1; 0; 0; 0		
80; 0; 0; 0; 0	ESC	
; 1; 0; 0; 0		
95; 0; 2; 1; 0	BREAK	-Short break
; 1; 2; 1; 0		-Long break
100; 0; 0; 0; 0	0	
102; 0; 0; 0; 0	00	
103; 0; 0; 0; 0	.	
104; 0; 0; 0; 0	ENTER	
; 1; 0; 0; 0		
105; 0; 0; 0; 0	,	
106; 0; 0; 0; 0	1	
; 28; 3; 2; 0		-GIN SW (GXL)
107; 0; 0; 0; 0	2	
; 1; 0; 2; 0		-See CUD
; 28; 3; 2; 0		-GIN South (GXL)
; 29; 2; 2; 0		-Zoom Down
; 30; 2; 2; 0		-See SU
; 31; 2; 2; 0		-Scroll Down Step
108; 0; 0; 0; 0	3	
; 28; 3; 2; 0		-GIN SE (GXL)
110; 0; 0; 2; 0	TAB	
; 0; 0; 2; 0		
111; 0; 0; 0; 0	4	
; 1; 0; 2; 0		-See CUB
; 28; 3; 2; 0		-GIN West (GXL)
; 29; 2; 2; 0		-Zoom Left
; 30; 2; 2; 0		-Scroll Left
; 31; 2; 2; 0		-Scroll Left Step



XL CONTROL STRING	KEY	FUNCTION
112; 0; 0; 0; 0   5	5	
; 1; 0; 1; 0   ~[ [ H		-See CUP
113; 0; 0; 0; 0   6	6	
; 1; 0; 2; 0   ~[ [ C		-See CUF
; 28; 3; 2; 0   ~[ / A		-GIN East (GXL)
; 29; 2; 2; 0   ~[ p		-Zoom Right
; 30; 2; 2; 0   ~[ s		-Scroll Right
; 31; 2; 2; 0   ~[ w		-Scroll Right Step
114; 0; 0; 0; 0   +	+	
116; 0; 0; 0; 0   7	7	
; 28; 3; 2; 0   ~[ / F		-GIN NW (GXL)
117; 0; 0; 0; 0   8	8	
; 1; 0; 2; 0   ~[ [ A		-See CUU
; 28; 3; 2; 0   ~[ / D		-GIN North
; 29; 2; 2; 0   ~[ n		-Zoom Up
; 30; 2; 2; 0   ~[ [ T		-See SD
; 31; 2; 2; 0   ~[ i		-Scroll Up Step
118; 0; 0; 0; 0   9	9	
; 28; 3; 2; 0   ~[ / F		-GIN NE (GXL)
119; 0; 0; 0; 0   -	-	
120; 0; 0; 1; 0   ~[ c	RESET	-See RIS
; 1; 0; 1; 0   ~[ c		
121; 0; 2; 1; 0   ~[ d	SETUP	-Enter Setup
; 1; 2; 1; 0   ~[ d		
141; 0; 0; 1; 0   ~[ [ K	ERASE	-See EL
; 1; 0; 1; 0   ~[ [ J		-See EP
142; 0; 0; 1; 0   ~[ 6	EDIT	-See zTI
143; 0; 0; 2; 0   ~[ [ P	DELETE	-See DCH
; 1; 0; 2; 0   ~[ [ M		-Dee DL
144; 0; 0; 2; 0   ~[ [ @	INSERT	-See ICH
; 1; 0; 2; 0   ~[ [ L		-See IL
145; 0; 0; 1; 0   ~[ [ i	PRINT	-See MC
; 1; 2; 1; 0   ~[ k		-Toggle local copy



CONTENTS	Page
INITIALIZING THE TERMINAL .....	1
DATA LINE MONITORING .....	2
LOCAL TEST MODE .....	3
SELF TEST .....	3

## INITIALIZING THE TERMINAL

**OPERATOR CONTROL.** The operator may initialize (or re-initialize) the terminal in any of the following ways:

**POWER-ON.** Each time the terminal is powered on, it goes through an Initialization procedure. In sequence:

- The ROM and RAM memories are tested; see Self Test, below.
- The Programmable keys are initialized (to their default values).
- The Power-on string is loaded into the FIFO
- The Non-Volatile Memory (NVM) checksum is verified.
- Working RAM is initialized from the NVM.
- I/O is initialized.
- Display RAM is initialized.
- The Power-on string is executed.
- The cursor is displayed.

**SETUP Z.** Typing the 2-key sequence, SETUP Z, causes the terminal to go through the same Initialization procedure (as at power-on), except that it does not drop the DTR (Data Terminal Ready) signal to the host (pin 20 on the RS232 connector). This is the preferred method for re-initializing after power-on, because it does not break the connection to the host.

**RESET KEY.** Depressing CTRL-SHIFT-RESET causes the terminal to re-initialize, but starting AFTER initialization of the Programmable keys (ie, the Self Test is not repeated, and the existing contents of the Programmable keys are left undisturbed).

In all cases, if it is desired that the Power-on string NOT be executed, hold down the RESET key during the initialization. The RESET key is checked prior to loading the Power-on string into the FIFO. If depressed, the string is not loaded and not executed.

**HOST CONTROL.** The host may initialize the terminal with the RIS (ESC c) control. This performs the same function as the operator's RESET key (ie, starts with loading of the Power-on string). The host cannot initiate Self Test. However, it can return the Programmable keys to their default values with the string ESC P > 0 ESC \ ; see Programmable Strings section).

**NOTE.** No data, including pads, should be sent to the terminal for at least 200 ms. following an RIS control, to allow the terminal time to re-initialize its I/O. If the USART comes out of initialization in the middle of a received character, that character and probably following characters will be garbled, with indeterminate effect.

## DATA LINE MONITORING

The XL Series provides an extremely powerful Data Line Monitoring capability. This includes a unique Data Monitor Line (that permits monitoring the data without disturbing the terminal's normal operation or display), as well as a conventional full-screen Data Monitor that can display up to a full Page of received characters. Couple this with the terminal's Meta Monitor Mode (that permits the display of 8-bit data), and its ability to transmit (full-page) monitored data back to the host, and you have a Data Monitor capability exceeding that of many high-priced commercial Data Monitors.

**NORMAL MONITORING.** The Received Data line (pin 3 of the Communications connector) normally accepts 7-bit data and parity. The Baud Rate, Parity, and Stop Bits expected are controlled by settings on Setup line B.

If Parity Detection is set, the parity of each incoming character is compared against the parity expected. If different, an error symbol (checkerboard) is displayed (instead of the character). The Data Monitor Line shows what character the terminal thought it received.

If Meta Monitor Mode (zMMM, B,57) is set, the terminal is able to accept 8-bit data (in lieu of parity). It may be used with the Data Monitor Line, or Transparent mode, to monitor 8-bit data. When used with the Data Monitor Line, the low-order 7-bits are processed normally by the terminal.

**DATA MONITOR LINE.** The Data Monitor Line permits you to monitor the incoming data while the terminal is in normal operation. This can be used to see what the host is sending when you're first installing the terminal, or trying to use an unfamiliar system, or trying out a new piece of software, or simply to isolate a system problem to the terminal or the host.

Typing the 2-key sequence, SETUP M, brings the Data Monitor Line to the Screen. The line displays the last 79 codes that have been processed by the terminal. The SSA symbol (forward-pointing triangle) is used as a separator on the line. The last code processed is shown immediately before the symbol and the oldest code immediately after the symbol. The symbol appears as a needle pulling the string of characters around the Monitor line.

Graphic character codes (2/0 to 7/15 inclusive) are displayed as their graphic characters, just as they are in the text in the Window above. Control codes (0/0 to 1/15 inclusive) are displayed as a Reverse rendition of their corresponding upper-case characters, for easy identification.

If Parity Detection is set, characters received with Parity error are displayed with the additional rendition of Bold Underline. Examples:

A displayed Normal	Upper-case A
A displayed Reverse	CTRL-A (SOH)
A displayed Bold, Underlined	A w/ parity error
A displayed Reverse, Bold, Underlined	SOH w/ parity error

If Meta Monitor Mode is set, the Bold, Underlined rendition means that the character was received with its 8th bit equal to 'one'.

Using the PAUSE key, in conjunction with this display, provides a powerful system debugging facility. You can operate the terminal normally yet, when an

unexpected result occurs, pause, and review the sequence of codes that led up to it.

When the Data Monitor line is no longer needed, simply type the SETUP key twice to remove it from the display.

**TRANSPARENT MODE.** Transparent mode permits the entire display Page to be used as a Data Monitor. Up to 28,000 received codes may be stored and viewed with the Guru; up to 4800 with the Genie+ and Ambassador.. In addition, the captured data may be transmitted back to the host for analysis; see Send Controls section, zGRTM (Graphic Rendition Transfer Mode).

Typing the 2-key sequence, SETUP X, puts the terminal into Transparent mode. Incoming codes are displayed exactly as they were for the Data Monitor Line, except that now they are displayed in the terminal's Window, and the terminal's entire Page is available to store them; see Display section, Definitions.

When Transparent mode is no longer needed, simply type the SETUP key twice to return the terminal to normal operation.

Toggling into and out of Transparent mode, does not affect the data presently stored and displayed (until codes are received, whereupon they scroll up naturally in the Page). Thus, the mode may be used, at any time, to capture a few lines of incoming codes for analysis, without totally eliminating the context of what you're doing.

#### LOCAL TEST MODE

Local Test mode is useful for performing terminal operations without involving the host, eg, to exercise the terminal to see how a particular control (or control sequence) works, to enter (or annotate) data prior to a Page Print, or simply to test the terminal to isolate a problem to the terminal, host, or printer. The Data Monitor line is displayed during Local Test to permit you to see exactly what codes you're entering.

Typing the 2-key sequence, SETUP T, puts the terminal into Local Test, with the Monitor line on the screen. The terminal's outgoing data line is wrapped back into its incoming data line within the terminal's communications interface. Codes are not sent to or received from the host. However, the Data-Terminal-Ready output at the Communications connector remains 'on'.

Typing the SETUP key twice, removes the Data Monitor line from the screen and returns the terminal to normal operation.

#### SELF TEST

The terminal performs a Self-Test procedure each time it is powered on (and in response to SETUP Z; see Initialization, above). In sequence:

- The bell beeps.
- The Program ROM checksums are verified.
- Working RAM is tested.
- Display RAM is tested.

If an error is found, the bell beeps, a 'MEMORY ERROR' flag is set, and the tests are re-started (from the beginning). Later in the initialization, if the NVM checksum does not verify, a 'CHECKSUM ERROR' flag is set. Upon completion of the initialization, if either error flag is set, the error message is displayed.

The following conclusions follow from observed results:

If the terminal beeps once, the cursor appears, and no error messages are displayed, the terminal has found no errors in its tests and has completed its initialization.

If the terminal beeps more than once, but the cursor appears and no error messages are displayed, the power supply is having trouble starting. This may indicate a power supply problem, but may also occur if the terminal is cold or the line voltage is low.

If the terminal beeps repeatedly and no cursor appears, the terminal has encountered a hard failure in its Self Test. The tests may be aborted, and the terminal forced to display, by holding down any key. The terminal may or may not operate correctly, depending on where and how bad the error is.

A CHECKSUM ERROR message indicates that the contents of NVM have changed since they were last saved. If you wish to disregard the error and continue operating, go into Setup mode, review and correct the terminal settings, and type the letter, S, to re-save them.

Repeated Checksum errors indicate a problem with the battery or the CMOS Ram, and should be checked by a qualified service person. The battery is a 3V lithium type (Panasonic #BR2325mm LI (wPCmt) 3V or equivalent) located in a plug-in socket on the Logic daughter board. It should last for 2-3 years of normal operation.

A MEMORY ERROR message indicates that one or more errors were detected during Self Test. If you wish to disregard the error and try operating anyway, depress CTRL-SHIFT-RESET. This re-initializes the terminal without performing the Self Test.

NOTE: The memory tests are run with worst-case timing, ie, worse than in normal operation. It is quite possible to get an error message, and find that the terminal works normally. The error message is an 'alert' that something is becoming marginal.

SETUP Y. For the Genie+ and Ambassador, if an intermittent error is suspected, the memory tests may be run repeatedly (eg, overnight) by typing the 2-key sequence, SETUP Y. The bell will beep each time an error is detected. The tests will end when the RESET key is held down.

CONTENTS	Page
ANSI X3.64 STANDARD .....	1
CODE NOTATION .....	1
CONTROL CODES .....	1
ESCAPE SEQUENCES .....	2
CONTROL SEQUENCES .....	2
PARAMETERS .....	2
NUMERIC PARAMETERS .....	3
SELECTIVE PARAMETERS .....	3
DEFAULT VALUES .....	3
PRIVATE-USE PROVISIONS .....	3

## ANSI X3.64 STANDARD

The XL Series uses the control-function code space defined by ANSI X3.64 "Additional Controls For Use With American National Standard Code For Information Interchange". The corresponding International Standard is ISO DP 6429. These Standards define a set of encoded control functions to facilitate data interchange with two-dimensional imaging devices, including display and printer terminals.

ANSI X3.64 augments the set of control functions defined in ANSI X3.4 (ASCII) and ANSI X3.41. The types of controls defined by X3.64 include cursor functions, editing functions, formatting functions, specification/control of input areas, setting/interrogation of status, mode selections, etc. The code structure defined is similar to escape sequences, but permits the inclusion of numeric and selective parameters. The paragraphs below summarize aspects of the Standard that are important to understanding the XL Series; the reader is referred to the Standard itself for more detail. Copies of the Standard may be ordered from the American National Standards Institute, 1430 Broadway, New York, New York 10018, (212)354-3300.

## CODE NOTATION

For consistency with the Standard, codes are represented by column/row of the ASCII table; see Appendix, page A-2. For example, the code for the character "K" is written as 4/11 (which is also 4B hex).

## CONTROL CODES

The terminals recognize and respond to only those ASCII control codes (0/0 to 1/15 inclusive) shown in the Appendix, Control Codes table.

## ESCAPE SEQUENCES

One of these control codes, ESC (1/11) tells the terminal to look at the next code(s) and assemble an escape sequence. Valid escape sequences are of the form:

ESC I...I F

ESC is the introducer code 1/11.

I...I are intermediate codes in the range 2/0 (Sp) to 2/15 (/) inclusive.

F is the final code in the range 3/0 (0) to 7/14 (~) inclusive.

The terminal assembles the sequence and, if it is one of the sequences recognized by the terminal (see Appendix, Escape Sequence tables), executes the control function called for by the sequence. Otherwise it ignores the entire sequence.

## CONTROL SEQUENCES

If the code immediately following the ESC (1/11) code is a final code 5/11 ([), the terminal is told to assemble a control sequence. Valid control sequences are of the form:

CSI P...P I...I F

CSI is the two-code sequence 1/11 5/11 (ESC [)

P...P are parameter codes in the range 3/0 (0) to 3/15 (?) inclusive.

I...I are intermediate codes in the range 2/0 (Sp) to 2/15 (/) inclusive.

F is the final code in the range 4/0 (@) to 7/14 (~) inclusive.

The terminal assembles the sequence and, if it is one of the sequences recognized by the terminal (see Appendix, Control Sequence tables), executes the control function called for by the sequence. Otherwise it ignores the entire sequence.

## PARAMETERS

The parameter portion of a control sequence may represent one or more parameter values. Each parameter may consist of one or more codes in the range 3/0 (0) to 3/9 (9), inclusive (representing the parameter value in decimal). Parameters are separated from each other by a 3/11 (;) code. There are two types of parameters: numeric and selective. Numeric parameters are used to pass numeric values. Selective parameters are used to select particular entries from a specified list. The form of both types is the same.



## NUMERIC PARAMETERS

The value passed by a numeric parameter is its decimal value (using the codes 3/0 (0) through 3/9 (9)). The value relates as directly as possible to the function (eg, line 12, column 56, move 34 columns, etc.). In each parameter, leading 3/0 (0) codes are not significant, and may be included or omitted. The maximum parameter value recognized by the terminals is 255.

## SELECTIVE PARAMETERS

The value passed by a selective parameter is also a decimal value, but specifies which entry (in a table of functions applicable to the control) is being called for. For example, the EL (Erase in Line) control permits the following parameter values: 0 (= from cursor to end), 1 (= from start to cursor), or 2 (= all of line).

## DEFAULT VALUE

If a parameter is omitted, or consists only of 3/0 (0) codes, it is given a default value which is specified in the control description.

## PRIVATE-USE PROVISIONS

The Standard provides for the addition of non-standard control sequences (to add features not envisioned by the standard) in two ways:

- 1) If the first code in the parameter string (of a control sequence) is 3/12 (<) to 3/15 (?) inclusive, then the entire parameter string is subject to private interpretation.
- 2) If the final character (of a control sequence) is 7/0 (p) to 7/14 (~) inclusive, then the entire sequence is subject to private interpretation.

The XL Series has used these provisions to add certain features to its control set. For users who wish to adhere rigidly to the Standard, these features are not necessary to operation of the terminal.



CONTENTS	Page
INTRODUCTION .....	1
CODE TABLES .....	2
CONTROL AND MODE DESCRIPTIONS .....	5

## INTRODUCTION

The purpose of this Appendix is to catalog all of the XL controls and modes for reference.

The Code Table section catalogs the terminal's controls, by code, in three tables: Control codes, Escape sequences, and Control sequences. These tables define the total code space used, and recognized, by the terminal.

The Descriptions section catalogs the terminal's controls and modes alphabetically by their ANSI (or private-use) mnemonic. Ann Arbor private-use controls and modes are identified by a leading lower-case 'z'; see User Guide, Introduction. DEC private-use controls and modes are identified by a leading lower-case 'd'. We have chosen to ignore the leading lower-case letters in the alphabetic sort, eg, zDBM follows DAQ and precedes DC1.

The terminal's modes are also listed in two other places. In the Descriptions section, under the SM (Set Mode) control, they are listed by mode number. In the Setup section of the User Guide, they are listed by Setup line location.

### CONTROLS:

By mnemonic	Appendix, Descriptions
By code	Appendix, Control Tables

### MODES:

By mnemonic	Appendix, Descriptions
By mode number	Appendix, Descriptions, SM (Set Mode)
By Setup line	Setup section

The notation used in the Appendix is consistent with that used in the other sections; see User Guide, Introduction, Notational Conventions.

ASCII CHARACTER SET. The ASCII character set consists of single codes in the range 0/0 to 7/15 inclusive. The full character set is shown in the table.

Column	0	1	2	3	4	5	6	7
Row	-----							
0	NUL	DLE	SP	0	@	P	`	p
1	SOH	DC1	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3	#	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

CONTROL CHARACTERS. The ASCII control characters are single codes in the range 0/0 TO 1/15 inclusive. The terminal responds to the control characters shown in the table. It ignores all others.

Column	0	1	2	3	4	5	6	7
Row	-----							
0	.	.	.	.	.	.	.	.
1	.	DC1	.	.	.	.	.	.
2	.	.	.	.	.	.	.	.
3	ETX	DC3	.	.	.	.	.	.
4	.	.	.	.	.	.	.	.
5	ENQ	.	.	.	.	.	.	.
6	.	.	.	.	.	.	.	.
7	BEL	.	.	.	.	.	.	.
8	BS	.	.	.	.	.	.	.
9	HT	.	.	.	.	.	.	.
10	LF	.	.	.	.	.	.	.
11	VT	ESC	.	.	.	.	.	.
12	FF	.	.	.	.	.	.	.
13	CR	.	.	.	.	.	.	.
14	SO	.	.	.	.	.	.	.
15	SI	.	.	.	.	.	.	.

ESCAPE SEQUENCES. Valid escape sequences are of the form, ESC I..I F. The terminal responds to two-character escape sequences with the final character shown in the table below. In addition, it responds to the three-character escape sequences listed below the table. It ignores all others.

Column	0	1	2	3	4	5	6	7
Row	-----							
0	.	.	.	.	.	DCS	DMI	.
1	.	.	.	.	.	.	INT	.
2	.	.	.	.	.	.	EMI	.
3	.	.	.	.	.	STS	RIS	.
4	.	.	.	.	IND	.	.	.
5	.	.	.	zTFC	NEL	.	.	.
6	.	.	.	zTI	SSA	SPA	.	.
7	.	.	.	zSC	ESA	EPA	.	.
8	.	.	.	zRC	HTS	.	.	.
9	.	.	.	zCGR	HTJ	.	.	.
10	.	.	.	zGUA	.	.	.	.
11	.	.	.	.	.	CSI	.	.
12	.	.	.	.	.	ST	.	.
13	.	.	.	.	RI	.	.	.
14	.	.	.	.	.	.	.	.
15	.	.	.	.	.	.	.	.

ESC ( A -SCSG0 (UK set)  
 ESC ( B -SCSG0 (ASCII set)  
 ESC ( 0 -SCSG0 (Graphics set)  
 ESC ) A -SCSG1 (UK set)  
 ESC ) B -SCSG1 (ASCII set)  
 ESC ) 0 -SCSG1 (Graphics set)

CONTROL SEQUENCES. Valid XL control sequences are of the form, CSI P..P F. The terminal responds to control sequences with the final character shown in the table. It ignores all others.

Column	0	1	2	3	4	5	6	7
Row	-----							
0	.	.	.	.	ICH	DCH	HPA	zSDP
1	.	.	.	.	CUU	SEE	HPR	.
2	.	.	.	.	CUD	CPR	REP	.
3	.	.	.	.	CUF	SU	DA	zPSH
4	.	.	.	.	CUB	SD	VPA	zPOP
5	.	.	.	.	CNL	.	VPR	zSTP
6	.	.	.	.	CPL	.	HVP	zSRC
7	.	.	.	.	CHA	CTC	TBC	zSPF
8	.	.	.	.	CUP	ECH	SM	.
9	.	.	.	.	CHT	.	MC	.
10	.	.	.	.	ED	CBT		zSTE
11	.	.	.	.	EL	.		.
12	.	.	.	.	IL	.	RM	.
13	.	.	.	.	DL	.	SGR	.
14	.	.	.	.	EF	.	DSR	.
15	.	.	.	.	EA	.	DAQ	.

## DESCRIPTIONS

The XL controls and modes are described below, arranged alphabetically by mnemonic, with leading lower-case letters ignored. For mode descriptions, the Setup line letter and mode number are shown in parentheses following the mode name. A dash (-) indicates that the mode is not provided on a Setup line.

**z8RCM 8-BIT REMOTE COPY MODE (C,61)**

In the RESET state, the Remote Copy operation of the MC control passes 7-bit data to the printer port.

In the SET state, 8-bit data may be passed to the printer port. However, the controls DC1, DC3, INT, and RIS (which are normally detected at the input to the FIFO), and the modes zINM (Ignore NUL) and zMMM (Meta Monitor), and the Parity Detection setting on Line B are all inactive. All received data is put into the FIFO as 8 bits. During Remote Copy, the output of the FIFO is passed on to the printer port until the specified number of characters have been passed or, if no number is specified, until an ETX-code is detected. Between Remote Copy's, the high-order bit is stripped and 7-bit data passed to the Display Processor. Caution: If this mode is reset, while there are following characters in the FIFO, those characters may display as parity errors regardless of the setting of Parity Detection.

**zACM ALTERNATE CURSOR MODE (-,51)**

The terminal provides two separate and independent sets of registers for cursor position and graphic rendition. This mode selects which set is referenced and affected by the incoming data stream.

In the RESET state, the incoming data stream references and affects the Primary registers. Significant characteristics of the Primary cursor are: (1) Its position is confined to the Page. (2) Its position (in the Window) is noted with a visible cursor symbol. (3) Erase and edit operations with the Primary cursor affect only the contents of the Page.

In the SET state, the incoming data stream references and affects the Alternate registers. Significant characteristics of the Alternate cursor are: (1) It can access all of Display Memory, including the Host Areas and Inactive Memory. (2) Its position is not visible, even if writing into the Window. (3) Erase and edit operations affect all of Display Memory. (4) Controls that require a line number parameter may require a different parameter value depending on which cursor is used. Address 1 of the Primary cursor points to the top line of the Page. Address 1 of the Alternate cursor points to the top line of Display Memory. These may differ by the number of lines of Upper Host Area. (5) The zSDP control, or any editing control that would move data below the Page, may not be used with the Alternate cursor.

During Monitor SRM, the operator is restricted to the Primary cursor, regardless of the state of zACM. The host may use the Alternate cursor to write to the terminal without interrupting the operator. Host entries into the Page may be affected by operator actions (eg, erase,

edit, scroll); DMI may be used to prevent this interaction. Likewise, host actions in or above the Page may affect the contents of the Page.

zAKDM AUTO KEYBOARD DISABLE MODE (B,46)

In the RESET state, the keyboard is immediately enabled upon completion of transmissions from the terminal's Display Memory to the host.

In the SET state, an implicit DMI control is performed at the completion of transmission, causing the keyboard to remain disabled until purposefully released by the host with an EMI control.

dANM VT52/ANSI MODE (D,-)

This is a DEC private-use mode provided for compatibility with commonly used DEC software. It puts the terminal into VT52 mode. It is set with the control ESC [ ? 2 h and reset with the control ESC [ ? 2 l. It can be changed in Setup mode, but the reset (VT52) state cannot be 'Saved', ie, the terminal always comes up in ANSI mode.

In the RESET state, the terminal responds only to VT52-compatible codes, as follows:

CONTROL	CODE	CONTROL	CODE
Cursor Up	ESC A	Home	ESC H
Cursor Down	ESC B	Reverse Line Feed	ESC I
Cursor Right	ESC C	Erase to End of Page	ESC J
Cursor Left	ESC D	Erase to End of Line	ESC K
Select Graphics	ESC F	Direct Cursor Address	ESC Y l c
Select ASCII	ESC G	Exit VT52 Mode	ESC <

In the SET state, the terminal responds to the normal XL control set.

zAPM AUTO PAUSE MODE (A,38)

In the RESET state, typing the PAUSE key causes the terminal to suspend the processing of data from its received-data FIFO buffer. When the FIFO becomes nearly full, an XOFF control is sent to the host to suspend further transmission of data. When the PAUSE key is typed again, the terminal resumes the processing of data from the FIFO. When the FIFO becomes nearly empty, an XON control is sent to the host to resume the transmission of data.

In the SET state, the processing of data from the FIFO is normally suspended. Typing the PAUSE key causes the terminal to process and display a single line of data from the FIFO. Typing SHIFT-PAUSE causes the terminal to process and display (two lines less than) a window-full of data from the FIFO. As in the reset state, XOFF and XON controls are sent to the host to avoid FIFO overflow.

In either state, typing CTRL-PAUSE sets zAPM. Typing CTRL-SHIFT-PAUSE resets zAPM. If zAXM is in the reset state, or if the host continues to



send data despite receiving an XOFF, the terminal automatically resumes processing to avoid FIFO overflow.

#### zAXM AUTO XOFF/XON MODE (B,37)

In the RESET state, the terminal is not permitted to send XOFF and XON controls to the host.

In the SET state, the terminal is permitted to send an XOFF control to the host to indicate that its received-data FIFO buffer is nearly full. The host is expected to suspend further transmission to the terminal until the terminal sends an XON control (indicating that the FIFO is nearly empty). The FIFO is a 256-character buffer. An XOFF is sent when the FIFO fills to 128 characters. A second XOFF is sent (in case the first was garbled) if the FIFO continues to fill to 160 characters. If the FIFO continues to fill to 224 characters, the terminal takes action to avoid loss of data; specifically, FIFO processing is resumed, Local Copy is terminated, and zSSM is reset.

#### BEL BELL (0/7)

Causes an audible bell tone to sound.

#### zBKCM BLOCK CURSOR MODE (A,31)

In the RESET state, the Underline rendition of the character at the cursor is complemented. If the character is rendered without Underline, the cursor is displayed as an Underline. If the character is rendered with Underline, the cursor is displayed as the absence of Underline.

In the SET state, the Reverse rendition of the character at the cursor is complemented. If the character is rendered as a 'white' character on a black field, the cursor is displayed as a 'white' field with black character. If the character is rendered as a black character on a 'white' field, the cursor is displayed as a black field with 'white' character.

#### zBNCM BLINKING CURSOR MODE (A,32)

In the RESET state, the cursor is displayed Bold and non-Blinking.

In the SET state, the cursor is displayed Bold and Blinking. The character at the cursor is alternated, at the blink rate, between its rendition at the cursor position and its rendition when not at the cursor position.

#### BS BACKSPACE (0/8)

Moves the cursor left one column. If zDBM is set, it also erases the character at that position. The rendition erased to depends on the setting of the GR-register; see SGR. If zWBM is set, the cursor may wrap

to the preceding line. If zACM is reset, the cursor may not move past the beginning of the Page.

CBT CURSOR BACKWARD TABULATION (ESC [ Pn Z)

Moves the cursor left Pn tab stops. If zWBM is set, the cursor may wrap to preceding lines. If zACM is reset, the cursor may not move past the beginning of the Page.

zCGR CHANGE GRAPHIC RENDITION (ESC 9)

Changes the graphic rendition of the character at the cursor, and all following characters in the active Qualified Area (see DAQ), to that specified by the present setting of the GR-register; see SGR.

CHA CURSOR HORIZONTAL ABSOLUTE (ESC [ Pn G)

Positions the cursor to column Pn in the active line. The cursor may not move past the line margins. This control is equivalent to HPA in the XL implementation.

CHT CURSOR HORIZONTAL TABULATION (ESC [ Pn I)

Moves the cursor right Pn tab stops. If zWFM is set, the cursor may wrap to following lines. If zACM is reset, the cursor may not move past the end of the Page.

zCLIM CAPS LOCK INVERT MODE (A,54)

This mode is applicable only to the 26 alphabetic keys with the CAPS LOCK key depressed.

In the RESET state, the keys generate upper-case codes, both unshifted and shifted.

In the SET state, the keys generate upper-case codes, unshifted, and lower-case codes, shifted.

CNL CURSOR NEXT LINE (ESC [ Pn E)

Moves the cursor down Pn lines to beginning of line. If zACM is reset, the cursor may not move past the bottom of the Page.

zCNM CR NEW-LINE MODE (D,55).

In the RESET state, entry or receipt of a CR (0/13) code moves the cursor to the beginning of the active line.

In the SET state, entry or receipt of a CR code moves the cursor to the

beginning of the next line.

CPL CURSOR PRECEDING LINE (ESC [ Pn F)

Moves the cursor up Pn lines to beginning of line. If zACM is reset, the cursor may not move past the top of the Page.

CPR CURSOR POSITION REPORT (ESC [ Pn1 ; Pn2 R)

Format for the report of cursor position. The parameters are the same as used by the CUP and HVP controls, ie, line Pn1, column Pn2. The report is solicited with the DSR control, ESC [ 6 n. The position reported is that of the Primary cursor regardless of the state of zACM.

CR CARRIAGE RETURN (0/13)

Moves the cursor to the beginning of the active line. If zCNM is set, it also does an implicit LF. If zCNM is set, zSPM and zACM are reset, and the cursor is in the bottom line of the Page, it does a scroll up of all data in the Page (losing the contents of the top line).

CSI CONTROL SEQUENCE INTRODUCER (ESC [ )

Introduces a control sequence.

zCSTM COLUMN SEPARATOR TRANSFER MODE (B,43)

In the RESET state, columnar tab stops do not cause HT (0/9) codes to be inserted into transmissions from the terminal's Display Memory to the host.

In the SET state, an HT code is inserted at each columnar tab stop. All spaces (2/0) immediately preceding the tab stop are suppressed in the transmission.

CTC CURSOR TABULATION CONTROL (ESC [ Ps W)

Sets or clears tab stops as selected by Ps:

- 0 Sets a tab stop at the cursor (default)
- 2 Clears the tab stop at the cursor
- 4 Clears all tab stops in the active line
- 5 Clears all tab stops

If TSM is reset, only columnar tab stops are affected. If TSM is set, page tab stops are affected; see DAQ. This control is equivalent to HTS in setting tab stops. It is equivalent to TBC (in function, but not in parameter values) in clearing tab stops.

**CUB** CURSOR BACKWARD (ESC [ Pn D)

Moves the cursor left Pn columns. If zWBM is set, the cursor may wrap to preceding lines. If zACM is reset, the cursor may not move past the beginning of the Page.

**CUD** CURSOR DOWN (ESC [ Pn B)

Moves the cursor down Pn lines. If zACM is reset, the cursor may not move past the bottom of the Page.

**CUF** CURSOR FORWARD (ESC [ Pn C)

Moves the cursor right Pn columns. If zWFM is set, the cursor may wrap to following lines. If zACM is reset, the cursor may not move past the end of the Page.

**CUP** CURSOR POSITION (ESC [ Pn1 ; Pn2 H)

Positions the cursor to line Pn1, column Pn2. If zACM is reset, Pn1 references the beginning of Page and the cursor may not move outside the Page boundaries. If zACM is set, Pn1 references the beginning of Active Memory. This control is equivalent to HVP in the XL implementation.

**CUU** CURSOR UP (ESC [ Pn A)

Moves the cursor up Pn lines. If zACM is reset, the cursor may not move past the top of the Page.

**DA** DEVICE ATTRIBUTES (ESC [ Pn c)

Provides a means for communicating device identification. The host may request the terminal to identify itself by sending the default control (ESC [ c). The terminal responds with a DA control containing a device identification parameter. In the XL implementation, this is the model-revision-option number of the terminal's firmware in the form NN;NN;NN. The model numbers of the Ambassador, Genie, and Guru are 11, 15, and 17 respectively. An alternative string may be down-loaded from the host with the DCS control.

**DAQ** DEFINE AREA QUALIFICATION (ESC [ Ps..Ps o)

Sets an Area Qualifier at the present cursor position, starting a Qualified Area, and assigning control attributes to it as selected by each Ps:

UNPROTECTED (Ps= 0). Local Entry is permitted into the Area.

PROTECTED/GUARDED (Ps= 1). Local Entry is not permitted into the Area, AND the contents of the Area will not be included in transmis-

sions to the host (if GATM is reset) or in transfers to the printer (if zGAPM is reset).

NUMERIC ONLY (Ps= 3). Local Entry is permitted into the Area, but restricted to numerics and punctuation only (codes 2/0 (Sp) to 3/15 (?) inclusive).

RIGHT JUSTIFY (Ps= 5). Local Entry is permitted into the Area, but is right justified when leaving the Area with the TAB key (ie, the entry is shifted right until a non-Space character occupies the last character position in the Area).

PAGE TAB STOP (Ps= 7). Sets a page tab stop at the beginning of the Area. The tab stop is active in respect to entries from the host, as well as Local Entry.

PROTECTED/UNGUARDED (Ps= 8). Local Entry is not permitted into the Area, but the contents of the Area are included in transmissions to the host and transfers to the printer, regardless of the settings of GATM and zGAPM.

If the first character in the parameter string is a > (3/14), the string is interpreted as a private-use parameter string supporting the standard parameters above, plus the following:

CONCEALED (Ps= 18). Local Entry is permitted into the Area, but is displayed only as 'white' spaces (2/0). Note: This qualification is not supported by the Guru.

The Area Qualifiers are stored in a control table indexed by character position. If any Area Qualifier has been set, a Forms flag is set, and the Display Processor references the table as part of processing its input. If the input is a local keyboard input, (ie, from the Keystroke buffer), the Display Processor references the table for each input. If the input is received from the host (ie, from the FIFO buffer), the Display Processor references the table only when it receives one of the tab controls, HT, CHT, CBT, or HTJ.

The table can hold up to 380 Area Qualifiers. However, there must be at least one entry in the table for each of 60 display lines, ie, the table capacity is reduced by 60 minus the number of lines that contain no Qualifiers.

The control SPA is equivalent to DAQ with a parameter value of 1. The control EPA is equivalent to DAQ with a parameter value of 7.

If TSM is set, the controls HTS, TBC, and CTC set and clear the tab stop attribute in Area Qualifiers without changing its other qualifications. Setting a tab stop adds the attribute to an existing Area Qualifier, or sets a new Area Qualifier (having the same qualifications as the preceding Qualifier). Clearing tab stops removes the attribute from the Area Qualifiers affected. If adjacent Area Qualifiers are left with the same qualification, the second Qualifier is removed.

If ERM is set, the controls EA, EF, EL, ED, and ECH remove Area Qualifiers from the region affected.

The Forms flag comes up reset at power-on. It is set by the first occurrence of any of the controls DAQ, SPA, EPA, or HTS (with TSM set). It may be reset by RIS, by erasing the entire Page with ERM set, or by zSDP (which permits clearing the form without erasing the Page); see Forms section.

zDBM DESTRUCTIVE BACKSPACE MODE (D,30)

In the RESET state, the BS (0/8) control causes the cursor to move left one column.

In the SET state, the BS control causes the cursor to move left one column and to erase the character at that position.

DC1 DEVICE CONTROL 1 (1/1)

Sets the Communications Ready Flag (CRF) in the terminal's communications interface. If CRF is set, AND the Clear-to-Send (CTS) input signal at the communications connector is 'on' (or open), this control enables data to be transmitted from the terminal's Display Memory to the host; see STS.

DC3 DEVICE CONTROL 3 (1/3)

Resets the Communications Ready Flag (CRF) in the terminal's communications interface. This control may be used by the host to prohibit or suspend the transmission of data from the terminal's Display Memory to the host. CRF is set at power-on. The host may reset it by sending the DC3 control, or condition the terminal to reset CRF automatically after transmitting each line of data; see zLTM. The host may re-enable or re-start the transmission by sending the DC1 control.

DCH DELETE CHARACTER (ESC [ Pn P)

Deletes Pn characters, starting with the character at the cursor, by shifting the remaining characters in the editing region to the left. The position of the cursor is not changed. Character positions vacated at the end of the editing region are erased. The rendition erased to is determined by the setting of the GR-register; see SGR. The extent of the editing region is determined by the setting of the EE-register; see SEE. If zACM is set, the control may not be used with an EE setting of zero (ie, to end of Page).

DCS DEVICE CONTROL STRING (ESC P)

Indicates the beginning of a device control string. A device control string is a string of graphic codes in the range 2/0 (Sp) to 7/14 (~) inclusive, starting with the first character after the DCS control and terminating with the ST control. The DCS control is used by the terminal to pass programming information to the terminal; see Programmable Strings section.

**zDDM DEL-CHARACTER DISPLAY MODE (D,35)**

In the RESET state, the DEL (7/15) code is ignored by the terminal.

In the SET state, the DEL code is accepted by the terminal as a valid graphic character and displayed as a block symbol.

**DL DELETE LINE (ESC [ Pn M)**

Deletes the contents of Pn lines, starting with the active line, by shifting the following lines upward. The position of the cursor is not changed. Lines vacated at the end of the Page are erased. The rendition erased to is determined by the setting of the GR-register; see SGR. If zACM is set, the control may not be used.

**DMI DISABLE MANUAL INPUT (ESC `)**

Sets KAM, which causes the keyboard to be disabled, except for the RESET, SETUP, and BREAK keys. These keys may be disabled by programming them with a null string. When the keyboard is disabled, key entry is inhibited (attempted entries result in an error 'beep') and the cursor is not displayed.

**DSR DEVICE STATUS REPORT (ESC [ Ps n)**

Provides a means for communicating device status or cursor position. The host may request the terminal for status by sending the DSR control with a parameter value of 5. The terminal responds with the DSR control with a parameter value of 0 or 1, as follows:

- 0 Ready (default)
- 1 Busy - retry later
- 5 Please report status (using a DSR control)
- 6 Please report cursor position (using a CPR control)

The busy response (ESC [ 1 n) means that a Page Print or Local Copy to the printer (see MC) is in progress and the screen should not be cleared. If the host's request contains a > (3/14), preceding the 5 (3/5), the busy response is not returned. Rather, the terminal suspends further processing from its FIFO until the print has been completed, and then returns a ready response (ESC [ n).

**EA ERASE IN AREA (ESC [ Ps 0)**

Erases some or all of the characters in the active Qualified Area as selected by Ps:

- 0 From cursor to end inclusive (default)
- 1 From start to cursor inclusive
- 2 All of the Qualified Area

The rendition erased to is determined by the setting of the GR-register;

see SGR. If ERM is set, protected characters may be erased, a page tab stop may be cleared, and the Qualified Area may be removed; see DAQ.

#### ECH ERASE CHARACTER (ESC [ Pn X)

Erases the following Pn characters, starting with the character at the cursor. The rendition erased to is determined by the setting of the GR-register; see SGR. If ERM is set, protected characters may be erased, page tab stops may be cleared, and Qualified Areas may be removed; see DAQ.

#### ED ERASE IN DISPLAY (ESC [ Ps J)

Erases some or all of the characters in the Page as selected by Ps:

- 0 From cursor to end inclusive (default)
- 1 From start to cursor inclusive
- 2 All of the Page

The rendition erased to is determined by the setting of the GR-register; see SGR. If ERM is set, protected characters may be erased, page tab stops may be cleared, and Qualified Areas may be removed; see DAQ.

#### EF ERASE IN FIELD (ESC [ Ps N)

Erases some or all of the characters in the active field as selected by Ps:

- 0 From cursor to end inclusive (default)
- 1 From start to cursor inclusive
- 2 All of the field

The rendition erased to is determined by the setting of the GR-register; see SGR. If ERM is set, protected characters may be erased, page tab stops may be cleared, and Qualified Areas may be removed; see DAQ.

#### EL ERASE IN LINE (ESC [ Ps K)

Erases some or all of the characters in the active line as selected by Ps:

- 0 From cursor to end inclusive (default)
- 1 From start to cursor inclusive
- 2 All of the line

The rendition erased to is determined by the setting of the GR-register; see SGR. If ERM is set, protected characters may be erased, page tab stops may be cleared, and Qualified Areas may be removed; see DAQ.



**EMI** ENABLE MANUAL INPUT (ESC b)

Resets KAM, which causes the keyboard to become enabled, and repositions the operator's window if necessary to contain the cursor.

**ENQ** ENQUIRY (0/5)

Causes the terminal to transmit a character string previously downloaded from the host with the DCS control. If no string has been downloaded, the control is ignored.

**EPA** END OF PROTECTED AREA (ESC W)

Sets an Area Qualifier with unprotected and tab stop attributes. This control is equivalent to a DAQ control with a parameter value of 7.

**ERM** ERASURE MODE (D,6)

In the RESET state, the controls EA, EF, EL, ED and ECH do not erase characters in protected Qualified Areas. They cannot affect the position of tab stops or Qualified Areas.

In the SET state, these controls erase characters regardless of their protected status, and may clear page tab stops and remove Qualified Areas; see DAQ.

**ESA** END OF SELECTED AREA (ESC G)

Writes an ESA symbol (backward-pointing triangle) at the cursor position. The ESA symbol is the end of a string of consecutive characters selected for subsequent transmission to the host or printer (see SATM, MATM). The beginning of the string is identified by an SSA symbol. The ESA symbol (and its control function) may be erased by erase controls, moved by edit controls, and is included in transmissions. The symbol is transmitted to the host as its escape sequence. It is transmitted to the printer as a Space (2/0) code.

**ESC** ESCAPE (1/11)

Introduces an escape sequence.

**ETX** END OF TEXT (0/3)

If received in the data stream following a zSRC control with a zero or default parameter, this control indicates the end of the data to be printed and terminates the Remote Copy operation. It is ignored otherwise.

**FETM FORMAT EFFECTOR TRANSFER MODE (B,14).**

In the RESET state, line separators are inserted into all transmissions from the terminal's Display Memory to the host. A separator is inserted at the end of each line from which characters have been transmitted. All spaces (2/0) immediately preceding the line separators are suppressed in the transmission. LNM selects CRLF (LNM reset) or LF (LNM set) as the line separator.

In the SET state, line separators are not inserted into transmissions to the host.

**FF FORM FEED (0/12)**

This control is equivalent to LF (0/10) in the XL implementation except that, if it is received with Local Copy on (see MC), the line that it terminates is tagged both to be printed and as end of (printed) page. When the line has been transferred to the printer, the terminal issues the necessary number of new-line codes (or an FF-code, if the Form Feed setting on Setup line C has been selected) to advance the paper to the top of the next page.

**zFRM FAST REPEAT MODE (A,47)**

This mode is significant only if zKRM is set.

In the RESET state, the key repeat rate is approximately 22 char/sec.

In the SET state, the key repeat rate is approximately 30 char/sec.

**zFSTM FIELD SEPARATOR TRANSFER MODE (B,44)**

In the RESET state, page tab stops do not cause HT (0/9) codes to be inserted into transmissions from the terminal's Display Memory to the host.

In the SET state, an HT code is inserted at each page tab stop. All spaces (2/0) immediately preceding the tab stop are suppressed in the transmission.

**zGAPM GUARDED AREA PRINT MODE (C,49)**

In the RESET state, characters located in guarded Qualified Areas (see DAQ) are printed as space (2/0) codes.

In the SET state, characters are printed regardless of their guarded qualification.

**GATM GUARDED AREA TRANSFER MODE (B,1)**

In the RESET state, characters located in guarded Qualified Areas (see DAQ) are not included in transmissions to the host.

In the SET state, characters are transmitted regardless of their guarded qualification.

**zGRTM GRAPHIC RENDITION TRANSFER MODE (B,45)**

In the RESET state, no SGR controls are inserted into transmissions to the host.

In the SET state, an SGR control is inserted at the beginning of transmission, to convey the rendition of the initial position, and thereafter at each change in rendition.

**zGUA GUARD UNPROTECTED AREAS (ESC :)**

Causes all unprotected Qualified Areas in the Page (see SPA, EPA, DAQ) to be guarded against transmission. Thereafter, any operator entry into a Qualified Area causes that area to become unguarded. May be used in conjunction with field separators (see zFSTM), to provide for transmission of only areas actually modified.

**zHAM HOLD IN AREA MODE (A,48)**

This mode is applicable only to keyboard input during Monitor SRM.

In the RESET state, when the operator enters a graphic character into the last character position of an unprotected Qualified Area (see SPA, EPA, DAQ), or uses the RIGHT-ARROW key at that position, the cursor auto-skips forward to the beginning of the next unprotected Area.

In the SET state, the cursor remains at that position, requiring use of the TAB key to advance to the next page tab stop.

**zHDM HALF-DUPLEX MODE (B,40)**

This mode applies to all keyboard inputs in Simultaneous SRM. In Monitor SRM, it applies only to inputs from keys programmed as Transmit-only by the DCS control.

In the RESET state, keyboard inputs are sent only to the host.

In the SET state, keyboard inputs are sent to the host and are echoed back locally within the terminal's communication interface.

**HPA HORIZONTAL POSITION ABSOLUTE (ESC [ Pn `)**

Positions the cursor to column Pn in the active line. The cursor may not move past the line margins. This control is equivalent to CHA in the XL implementation.

**HPR HORIZONTAL POSITION RELATIVE (ESC [ Pn a)**

Moves the cursor right Pn columns. The cursor may not move past the right margin.

**HT HORIZONTAL TABULATION (O/9)**

Moves the cursor right to the next tab stop. If zWFM is set, the cursor may wrap to following lines. If zWFM is set, zSPM and zACM are reset, and the cursor is at the last tab stop in the Page, all data in the Page scrolls up one line (losing the contents of the top line).

**HTJ HORIZONTAL TAB WITH JUSTIFY (ESC I)**

Shifts the characters between the preceding tab stop and the cursor, but not including the character at the cursor, forward up to the next tab stop (or end of line, whichever is first). The character positions between the preceding tab stop and the new beginning of the shifted string are erased. The rendition erased to is determined by the setting of the GR-register; see SGR. The cursor moves to the next tab stop. If zWFM is set, the cursor may wrap to following lines. If zWFM is set, zSPM and zACM are reset, and the cursor is at the last tab stop in the Page, all data in the Page scrolls up one line (losing the contents of the top line).

**HTS HORIZONTAL TAB SET (ESC H)**

Sets a tab stop at the cursor. If TSM is reset, a columnar tab stop is set. If TSM is set, a page tab stop is set. Setting a page tab stop may start a Qualified Area; see DAQ. This control is equivalent to the CTC control, ESC [ W.

**HVP HORIZONTAL AND VERTICAL POSITION (ESC [ Pn1 ; Pn2 f)**

Positions the cursor to line Pn1, column Pn2. If zACM is reset, Pn1 references the beginning of Page, and the cursor may not move outside the Page boundaries. If zACM is set, Pn1 references the beginning of Active Memory. This control is equivalent to CUP in the XL implementation.

**ICH INSERT CHARACTER (ESC [ Pn @)**

Inserts Pn spaces (2/0) at the cursor by shifting the character at the cursor, and all following characters in the editing region, to the

right. Characters shifted past the end of the editing region are lost. The position of the cursor is not changed. The rendition of the inserted characters is determined by the setting of the GR-register; see SGR. The extent of the editing region is determined by the setting of the EE-register; see SEE. If zACM is set, the control may not be used with an EE-setting of zero (ie, to end of Page).

#### zICM INVISIBLE CURSOR MODE (D,56)

In the RESET state, the cursor symbol is displayed in accordance with the state of zBKCM and zBNCM.

In the SET state, the cursor symbol is not displayed.

#### IL INSERT LINE (ESC [ Pn L)

Inserts Pn erased lines by shifting the contents of the active line, and all following lines, downward. The position of the cursor is not changed. Lines shifted past the end of Page are lost. The rendition of the inserted lines is determined by the setting of the GR-register; see SGR. If zACM is set, the control may not be used.

#### IND INDEX (ESC D)

Moves the cursor down one line. If zSPM and zACM are reset and the cursor is in the bottom line of the Page, all data in the Page scrolls up on line (losing the contents of the top line).

#### zINM IGNORE NUL MODE (B,54)

In the RESET state, NUL (0/0) codes received in the incoming data stream are put into the FIFO (so they may be passed on to the printer in Remote Copy).

In the SET state, NUL codes received in the incoming data stream are ignored (ie, not put into the FIFO) so that they may be used as pad characters on systems that cannot accommodate Xon/Xoff flow control; see zAXM. The following is the worst-case execution time of long controls. Typical is much less.

Insert/delete line	3 ms.
Insert/delete char (in line)	4
Push/pop line	5
Erase line	5
Erase 60 lines	156
Change graphic rendition	283
Insert/delete char (in display)	304
Erase 60 lines with max DAQ's	2368

**INT INTERRUPT (ESC a)**

This control is detected at the input to the terminal's FIFO, and causes the terminal to terminate a print or send in process.

**IRM INSERTION-REPLACEMENT MODE (-,4)**

In the RESET state, the entry or receipt of a graphic character causes the appropriate graphic symbol to replace the symbol currently imaged at the cursor, and then moves the cursor forward one character position.

In the SET state, the entry or receipt of a graphic character causes the appropriate graphic symbol to be inserted at the cursor, after shifting the character at the cursor, and all following characters in the editing region, forward one character position, and then moves the cursor forward one character position. The extent of the editing region is determined by the setting of the EE-register; see SEE. If ZACM is set, the mode may not be used in this state with an EE-setting of zero (ie, to end of Page).

**zIVM INVERSE VIDEO MODE (A,59)**

This mode is supported by the Guru only.

In the RESET state, characters rendered Normal are displayed as 'white' characters on a black background. Characters rendered Reverse are displayed as black characters on a 'white' background.

In the SET state, characters rendered Normal are displayed as black characters on a 'white' background. Characters rendered Reverse are displayed as 'white' characters on a black background.

**KAM KEYBOARD ACTION MODE (B,2)**

In the RESET state, keyboard input is enabled.

In the SET state, keyboard input is disabled, except for the RESET, SETUP, and BREAK keys. These keys may be disabled by programming them with a null string. When the keyboard is disabled, key entry is inhibited (attempted entries result in an error 'beep') and the cursor is not displayed.

**zKCM KEY CLICK MODE (A,26)**

In the RESET state, no audible feedback of key entry is generated.

In the SET state, an audible key click is sounded as each character is entered.

**zKPCM KEY PAD CONTROL MODE (A,27)**

In the RESET state, the control keypad generates the numeric set of the graphic characters and requires depression of the SHIFT key to generate terminal control functions.

In the SET state, the control keypad generates the terminal control functions and requires depression of the SHIFT key to generate the numeric set of graphic characters.

**zKRM KEY REPEAT MODE (A,28)**

Selects the keys that automatically repeat when held down for more than 1/2 second. (The repeat rate is determined by zFRM.)

In the RESET state, the four cursor ARROW keys, plus TAB, INSERT, and DELETE, repeat. (These keys may be re-programmed to not repeat.)

In the SET state, the number keys on the Control keypad and all keys on the Main keypad (except ESC, PAUSE, BREAK, and RETURN) also repeat.

**LF LINE FEED (0/10)**

Moves the cursor down one line. If LNM is set, it also does an implicit CR (0/13). If zSPM and zACM are reset and the cursor is in the bottom line of the Page, all data in the Page scrolls up one line (losing the contents of the top line).

**LNM LF NEW LINE MODE (D,20)**

In the RESET state, the LF (0/10) code moves the cursor down one line.

In the SET state, LF moves the cursor to the beginning of the next line, effecting the NL (New Line) option of ANSI X3.4. This mode does not affect the IND or NEL controls.

**zLTM LINE TRANSFER MODE (B,42)**

In the RESET state, the transmission of data from the terminal's Display Memory to the host is not automatically suspended at end of line.

In the SET state, the transmission of data from the terminal's Display Memory to the host is automatically suspended, by resetting the Communications Ready Flag (CRF) after each LF code. The host is required to send the DC1 control (to set CRF) after receiving each line, to complete the transmission.

**MATM MULTIPLE AREA TRANSFER MODE (B,15)**

This mode is applicable only if SATM is reset.

In the RESET state, only the single Selected Area containing the cursor is transmitted to the host or printer. Transmission starts with the first preceding SSA symbol. If there is no preceding SSA symbol, an SSA symbol is implied at the beginning of the Page. If an ESA symbol is encountered before an SSA symbol, the operation is terminated and the control is ignored. If TTM is reset, an ESA symbol is implied at the cursor and transmission terminates with the cursor position. If TTM is set, transmission terminates with the first occurrence of an ESA symbol. If there is no terminating ESA symbol, one is implied at the end of the Page.

In the SET state, all Selected Areas in the Page are transmitted. The transfer pointer starts scanning from the beginning of Page (unless otherwise set; see zSTP). Transmission is enabled by SSA symbols and disabled by ESA symbols. If the first symbol encountered is an ESA symbol, an SSA symbol is implied at the initial transfer-pointer position. If TTM is reset, scanning terminates at the cursor. If TTM is set, scanning continues to end of Page (unless otherwise set; see zSTE). In either case, an ESA symbol is implied at that point, and transmission terminates.

Actual SSA and ESA symbols are included in the transmission; implied symbols are not. The symbols are transmitted to the host as their escape sequence. They are transferred to the printer as space (2/0) codes.

#### zMBM MARGIN BELL MODE (A,25)

In the RESET state, no audible feedback is generated as the cursor approaches the right margin.

In the SET state, an audible bell is sounded whenever the operator types a graphic character with the cursor in column 72.

#### MC MEDIA COPY (ESC [ Ps i)

Controls the transfer of data to the printer as selected by Ps:

- 0 Print Page (default)
- 4 Turn off Local Copy
- 5 Turn on Local Copy

A parameter value of 0 initiates transfer from the terminal's Display Memory to the printer. All transfers start at the beginning of Page, unless otherwise specified; see zSTP. All transfers end at the cursor (TTM reset) or the end of Page (TTM set), unless otherwise specified; see zSTE. The form and content of the transfer are affected by the following modes:

- MATM Multiple Area Transfer Mode
- SATM Single Area Transfer Mode
- zGAPM Guarded Area Print Mode

A parameter value of 5 enables Local Copy. All data received by the



terminal are also copied to the printer. The data are copied from Display Memory (not from the data stream) after each display line has been completed with an LF or NEL control.

A parameter value of 4 disables Local Copy.

With either method of print, the format of the printout (margin and page size) may be specified with the zSPF control. Any transfer in progress to the printer may be aborted with the INT or RIS controls.

**zMKM META KEY MODE (B,52).**

In the RESET state, the S1 key performs non function.

In the SET state, the S1 key acts as a META SHIFT key, to facilitate use of the terminal with software that uses the parity bit for 8-bit data transmission. If held down while any other key is struck, the key code is transmitted with its parity bit equal to 'one'. Otherwise, the parity bit is transmitted equal to 'zero' (regardless of the Setup parity settings).

The META SHIFT function may be reprogrammed to another key(s). Likewise, the S1 key may be reprogrammed to perform a different function(s). See Programmable Strings section.

**zMMM META MONITOR MODE (B,57)**

In the RESET state, the terminal accepts 7-bit data and parity in the incoming data stream. If the Parity Detection bit (on Setup line B) is set, parity errors are displayed as an error symbol (checkerboard) on the screen, and as the character rendered in Bold Underline on the Data Monitor line.

In the SET state, the terminal accepts 8-bit data (in lieu of parity) in the incoming data stream. Characters received with the 8th bit equal to 'one' are displayed in the Data Monitor line rendered in Bold Underline. The 7-bit portion of the data (after stripping the 8th bit) is displayed on the screen.

**NEL NEXT LINE (ESC E)**

Moves the cursor to the beginning of the next line. If zSPM and zACM are reset and the cursor is in the bottom line of the Page, all data in the Page scrolls up one line (losing the contents of the top line).

**dOM ORIGIN MODE (D,-)**

This is a DEC private-use mode provided for compatibility with commonly used DEC software. It is set with the control ESC [ ? 6 h and reset with the control ESC [ ? 6 l.

In the RESET state, the origin is the first character position within the display Page. Line numbers are independent of the current top/bottom margin settings.

In the SET state, the origin is the first character position of the line designated to be the top margin; see dSTBM. Line numbers are relative to the top margin setting.

zPOP POP LINE (ESC [ Pn t)

Moves the contents of Pn lines at the bottom of the Page to (start at) the active line, by shifting the active line, and all following lines, downward. The cursor position is unchanged. If zACM is set, the control may not be used.

zPSH PUSH LINE (ESC [ Pn s)

Moves the contents of Pn lines, starting with the active line, to the bottom of Page, by shifting the bottom lines of the Page upward. The cursor position is unchanged. If zACM is set, the control may not be used.

zRC RESTORE CURSOR (ESC 8)

Restores the cursor position (and rendition setting) to its 'saved' values; see zSC. If zACM is reset, it restores the values to the Primary cursor. If zACM is set, it restores the values to the Alternate cursor.

REP REPEAT (ESC [ Pn b)

Causes the single graphic character immediately preceding the control to be repeated Pn times.

RI REVERSE INDEX (ESC M)

Moves the cursor up one line. If zSPM and zACM are reset and the cursor is in the top line of the Page, all data in the Page scrolls down one line (losing the bottom line).

RIS RESET TO INITIAL STATE (ESC c)

This control is detected at the input to the terminal's FIFO and causes the terminal to reset to its power-on condition, except that the string table (see DCS) is not re-initialized. No data, including pads, should be sent to the terminal for at least 200 ms. following an RIS control, to allow the terminal time to re-initialize its I/O. If the USART comes out of initialization in the middle of a received character, that character and probably following characters will be garbled, with indeterminate effect.

**zRLM** RETURN KEY CRLF MODE (A,29)

In the RESET state, the RETURN key generates the CR code (0/13).

In the SET state, the RETURN key generates the two-code sequence CR LF (0/13 0/10).

**RM** RESET MODE (ESC [ Ps..Ps 1)

Resets modes of the terminal as selected by each Ps. Multiple Ps must be separated by a ; (3/11). The control has both a standard and private-use form; see SM.

**SATM** SELECTED AREA TRANSFER MODE (B,17)

In the RESET state, the contents of Selected Areas (see SSA, ESA) are transmitted to the host or printer. Characters outside of Selected Areas are not transmitted. If MATM is reset, only the active Selected Area is transmitted. If MATM is set, all Selected Areas are transmitted.

In the SET state, characters are transmitted regardless of their Selected status.

**zSC** SAVE CURSOR (ESC 7)

Saves the cursor position and rendition setting in a temporary buffer for later recall; see zRC. If zACM is reset, the Primary cursor position and associated rendition are saved. If zACM is set, the Alternate cursor position and associated rendition are saved.

**SCSG0** SELECT CHARACTER SET GO (ESC ( Fc)

Designates the GO character set from one of the available sets depending on the final character, Fc:

- A (4/1) United Kingdom set
- B (4/2) ASCII set
- 0 (3/0) Graphics set

**SCSG1** SELECT CHARACTER SET G1 (ESC ) Fc)

Designates the G1 character set from one of the available sets depending on the final character, Fc:

- A (4/1) United Kingdom set
- B (4/2) ASCII set
- 0 (3/0) Graphics set

**SD** SCROLL DOWN (ESC [ Pn T)

Moves the contents of the Window down Pn lines, permitting the display of preceding lines. The position of the cursor (in the Page) is not changed.

**zSDP** SET DISPLAY PARAMETERS (ESC [ Pn1 ; Pn2 ; Pn3 ; Pn4 ; Pn5; Pn6 p)

Partitions the terminal's Display Memory and Screen, according to the parameters. In addition, it resets the Forms flag (see DAQ), and moves the cursor to home.

**ACTIVE MEMORY LINES (Pn1).** Specifies the number of lines of Display Memory to be used. The power-on value is 60 lines. If the parameter is omitted or zero, the present values of Active Memory, and Upper and Lower Host Area, height remain unchanged. If the value is changed, Display Memory is re-partitioned but its contents are not changed.

**UPPER HOST AREA LINES (Pn2).** Specifies the number of lines to be used for an Upper Host Area. These lines are always displayed at the top of the Screen and may be written into only by the host. The power-on and default value is 0.

**LOWER HOST AREA LINES (Pn3).** Specifies the number of lines to be used for a Lower Host Area. These lines are always displayed at the bottom of the Screen and may be written into only by the host. The power-on and default value is 0.

**SCREEN LINES (Pn4).** Specifies the number of lines to be displayed on the Screen, including Host Areas if any. Power-on value is 30 lines. If the parameter is omitted or zero, the present value is not changed.

**SCREEN COLUMNS (Pn5).** This parameter is active for the Guru only. It specifies the number of columns to be displayed on the Screen. The power-on value is 80 columns. If the parameter is omitted or zero, the present value is not changed.

**ACTIVE MEMORY COLUMNS (Pn6).** This parameter is active for the Guru only. It specifies the number of columns of Display Memory to be used. The power-on value is 80 columns. If a value is specified which, when multiplied by the present or specified Active Memory Lines (Pn1), exceeds the maximum capacity of Display Memory (approximately 28K characters), the Active Memory Lines are reduced to accommodate the specified width. If the parameter is omitted or zero, the present value is not changed. If the value is changed, Display Memory is re-partitioned AND its contents are erased.

The value of Pn1 should be greater than Pn2 + Pn3. The value of Pn4 should be greater than Pn2 + Pn3 + 1. The value of Pn6 should be greater than Pn5.

## SEE SELECT EDITING EXTENT (ESC [ Ps Q)

Sets an EE-register that determines the extent of the Page affected by the DCH and ICH controls, and by data entered or received with IRM set, as selected by Ps:

- 0 Edit in Page (default)
- 1 Edit in Line
- 2 Edit in Field (between tab stops)
- 3 Edit in Qualified Area (see DAQ, SPA, EPA)

## SGR SELECT GRAPHIC RENDITION (ESC [ Ps..Ps m)

Sets a GR-register that determines the graphic rendition with which characters subsequently entered or received are displayed as selected by each Ps. Multiple Ps must be separated by a ; (3/11).

- 0 Normal (default)
- 1 Bold or increased intensity
- 4 Underscore
- 5 Blink
- 7 Reverse (black character on 'white' field)
- 8 Concealed (Genie+ and Ambassador only)

The contents of the GR-register are appended to all characters written into the Page, including spaces (2/0) written as a result of erase and edit controls. The rendition is a part of the displayed character. When the character is erased or moved, its rendition is erased or moved with it. If zGRTM is set, the rendition information is included in transmissions to the host; see STS.

## SI SHIFT IN (0/15)

Causes characters subsequently entered or received to be displayed from the G0 character set; see SCSG0.

## SM SET MODE (ESC [ Ps..Ps h)

Sets modes within the terminal as selected by each Ps. Multiple Ps must be separated by a ; (3/11).

- 1 GATM Guarded Area Transfer Mode.
- 2 KAM Keyboard Action Mode.
- 4 IRM Insertion-Replacement Mode.
- 6 ERM Erasure Mode.
- 12 SRM Send-Receive Mode.
- 14 FETM Format Effector Transfer Mode.
- 15 MATM Multiple Area Transfer Mode.
- 16 TTM Transfer Termination Mode.
- 17 SATM Selected Area Transfer Mode.
- 18 TSM Tabulation Stop Mode.
- 20 LNM Line Feed New Line Mode.

If the first character in the parameter string is a > (3/14), the string is interpreted as a private-use parameter string supporting the standard parameters above, plus the following:

25	zMBM	Margin Bell Mode
26	zKCM	Key Click Mode
27	zKPCM	Key Pad Control Mode
28	zKRM	Key Repeat Mode
29	zRLM	Return Line Feed Mode
30	zDBM	Destructive Backspace Mode
31	zBKCM	Block Cursor Mode
32	zBNCM	Blinking Cursor Mode
33	zWFM	Wrap Forward Mode
34	zWBM	Wrap Backward Mode
35	zDDM	DEL-character Display Mode
36	zSPM	Scroll-Page Mode
37	zAXM	Auto Xoff/Xon Mode
38	zAPM	Auto Pause Mode
39	zSSM	Slow Scroll Mode
40	zHDM	Half-Duplex Mode
41	zTPDM	Transfer Pointer Display Mode
42	zLTM	Line Transfer Mode
43	zCSTM	Column Separator Transfer Mode
44	zFSTM	Field Separator Transfer Mode
45	zGRTM	Graphic Rendition Transfer Mode
46	zAKDM	Auto Keyboard Disable Mode
47	zFRM	Fast Repeat Mode
48	zHAM	Hold in Area Mode
49	zGAPM	Guarded Area Print Mode
51	zACM	Alternate Cursor Mode
52	zMKM	Meta Key Mode
53	zCLIM	Caps Lock Invert Mode
54	zINM	Ignore NUL Mode
55	zCNM	CR New-Line Mode
56	zICM	Invisible Cursor Mode
57	zMMM	Meta Monitor Mode
59	zIVM	Inverse Video Mode (Guru only)
61	z8RCM	8-Bit Remote Copy Mode (Genie+/Ambassador only)

#### SO SHIFT OUT (0/14)

Causes characters subsequently entered or received to be displayed from the G1 character set; see SCSG1.

#### SPA START OF PROTECTED AREA (ESC V)

Sets an Area Qualifier with protected attribute. This control is equivalent to a DAQ control with a parameter value of 1.

#### zSPF SET PRINT FORMAT (ESC [ Pn1 ; Pn2 ; Pn3 w)

Sets the format of printed copy resulting from use of the MC control according to the parameters:

PRINTED LINES (Pn1). Valid values are 0 to 126 inclusive. A non-zero value specifies the number of lines to be printed per page and takes the present position of the printer as the top of page. If the parameter is omitted or zero, the printer is caused to feed forward to the top of the next page, and the present value of Printed Lines, Total Lines, and Left Margin to remain unchanged.

TOTAL LINES (Pn2). Specifies the total number of lines per page. Valid values are 0 to 126. If the parameter is omitted or zero, the present value of Total Lines remains unchanged.

LEFT MARGIN (Pn3). Specifies the number of space (2/0) codes to be sent to the printer at the beginning of each line to set the Left Margin of the printout. Valid values are 0 to 254 inclusive. Default value is 0.

#### zSPM SCROLL-PAGE MODE (D,36)

In the RESET state, the terminal is in Scroll mode. Attempts to move the cursor past the bottom of Page with the IND, LF, or NEL controls, or past the end of Page with the HT or HTJ controls or graphic character entry (if zWFM is reset), cause the cursor to remain in the bottom line, and all characters in the Page to scroll up one line. The top line is removed and the bottom line is erased. Similarly, attempts to move the cursor past the top of Page with the RI control cause the cursor to remain in the top line, and all characters in the Page to scroll down one line. The bottom line is removed and the top line is erased. In either case, the rendition erased to is determined by the setting of the GR-register; see SGR.

In the SET state, the terminal is in Page mode. Attempts to move the cursor past the bottom of Page with the IND, LF, or NEL controls are ignored. Attempts to move the cursor past the end of Page with the HT or HTJ controls or graphic character entry cause the cursor to move to (or remain at) the end of the Page. Similarly, attempts to move the cursor past the top of Page with the RI control are ignored.

#### zSRC START REMOTE COPY (ESC [ Pn v)

Causes the characters and controls that follow to be copied directly to the printer interface, and not displayed or executed by the terminal. The parameter specifies the number of bytes to be copied. If the parameter is omitted, copy continues until the first occurrence of an ETX (0/3) code in the data stream.

#### SRM SEND-RECEIVE MODE (B,12)

In the RESET state, the terminal is in the Monitor SRM. Keyboard inputs are not sent to the host as they are keyed. They are, instead, displayed and executed locally, and the resultant display subsequently sent to the host as a block; see STS. The inputs do not use the terminal's received-data FIFO. The FIFO may be used by the host to buffer data to the printer (see zSRC) and/or to Display Memory.

In the SET state, the terminal is in Simultaneous SRM. The keyboard is logically disconnected from the display. Keyboard inputs are passed to the terminal's communications interface and sent to the host as they are keyed. The host may echo the inputs back to the display or the inputs may be echoed back locally (zHDM set). In either case, the inputs become part of the incoming data stream and are buffered through the terminal's FIFO (see zAXM).

SSA START OF SELECTED AREA (ESC F)

Writes an SSA symbol (forward pointing triangle) at the cursor position. The SSA symbol is the first of a string of consecutive characters selected for subsequent transmission to the host or printer; see SATM, MATM. The end of the string is identified by an ESA symbol. The SSA symbol (and its subsequent control function) may be erased by erase controls, moved by edit controls, and is included in transmissions. The symbol is transmitted to the host as its escape sequence. It is transmitted to the printer as a space (2/0) code.

zSSM SLOW SCROLL MODE (A,39)

In the RESET state, characters received in the incoming data stream are processed and displayed at the incoming character rate.

In the SET state, characters received in the incoming data stream are displayed at a maximum rate of 4 lines per second. The terminal suspends processing of data from the received-data FIFO buffer as required to maintain this display rate. If the FIFO becomes nearly full, an XOFF control is sent to the host to suspend further transmission of data. When the FIFO becomes nearly empty, an XON control is sent to the host to resume the transmission of data. If zAXM is reset (prohibiting the transmission of XOFF), the terminal automatically resets zSSM and resumes processing if the FIFO becomes full.

ST STRING TERMINATOR (ESC \)

When received in the data stream following a DCS control, indicates the end of the character string to be transferred and terminates the transfer. It is ignored otherwise.

dSTBM SET TOP AND BOTTOM MARGINS (ESC [ Pn1 ; Pn2 r)

This is a DEC private-use control provided for compatibility with commonly used DEC software. It sets top and bottom margins according to the parameters to define a scrolling region within the display page. Pn1 is the line number of the top line of the scrolling region. Pn2 is the line number of the bottom line of the scrolling region. Default is the entire page (ie, no margins).



**zSTE SET TRANSFER END (ESC [ Pn1 ; Pn2 z)**

Sets the ending address of the transfer pointer to line Pn1, column Pn2. The transfer includes the character at this position. A parameter value of zero (or omission of the parameter) for either parameter sets the ending address to the present cursor position for that parameter. At the completion of each transfer, the pointer is restored to the end of the Page. If TTM is reset, the transfer may terminate at the cursor.

**zSTP SET TRANSFER POINTER (ESC [ Pn1 ; Pn2 u)**

Sets the starting address of the transfer pointer to line Pn1, column Pn2. The transfer includes the character at this position. A parameter value of zero (or omission of the parameter) for either parameter sets the starting address to the present cursor position for that parameter. At the completion of each transfer, the pointer is restored to the beginning of the Page.

**STS SET TRANSMIT STATE (ESC S)**

Sets a Transmit Ready Flag (TRF) notifying the terminal's communications interface that the displayed data are ready to be transmitted to the host. This control initiates transmission if the Communications Ready Flag (CRF) is set (see DC1, DC3) and the Clear-to-Send (CTS) control input signal at the communications connector is 'on'.

All transmissions start at the beginning of Page, unless otherwise specified; see zSTP. All transmissions end at the cursor (TTM reset) or the end of Page (TTM set), unless otherwise specified; see zSTE. The form and content of the transmission are affected by the following modes:

- FETM Format Effector Transfer Mode
- GATM Guarded Area Transfer Mode
- MATM Multiple Area Transfer Mode
- SATM Single Area Transfer Mode
- zCSTM Column Separator Transfer Mode
- zFSTM Field Separator Transfer Mode
- zGRTM Graphic Rendition Transfer Mode
- zLTM Line Transfer Mode

During an operator-initiated transmission, the keyboard is disabled and the cursor symbol is not displayed. A transfer pointer symbol may be displayed; see zTPDM. At completion of the transmission, TRF is reset and the keyboard is enabled, unless otherwise specified; see zAKDM. Receipt of an STS control from the host does not remove the operator's cursor or disable the keyboard. The host is permitted to read all or any part of Display Memory without interfering with the operator's entry. A transmission in progress may be aborted with the INT or RIS controls.

**SU** SCROLL UP (ESC [ Pn S)

Moves the contents of the Window up Pn lines, permitting the display of following lines. The position of the cursor (in the Page) is not changed.

**TBC** TABULATION CLEAR (ESC [ Ps g)

Clears one or more tab stops as selected by Ps:

- 0 Clears the tab stop at the cursor (default).
- 2 Clears all tab stops in the active line.
- 3 Clears all tab stops.

If TSM is reset, it clears columnar tab stops. If TSM is set, it clears page tab stops, and may remove Qualified Areas (see DAQ). This control is equivalent to CTC (in function but not in parameter values).

**zTFC** TOGGLE FAST-BLINK CURSOR (ESC 5)

Toggles the cursor symbol between its 'normal' display (as selected by zBKCM and zBKNM) and a fast-blinking block, for operator attention or warning.

**zTI** TOGGLE IRM (ESC 6)

Toggles the state of IRM, ie, resets it if set, and sets it if reset.

**zTPDM** TRANSFER POINTER DISPLAY MODE (A,41)

During operator-initiated transmission of data from the terminal's Display Memory to the host or printer, the cursor symbol is not displayed (because operator entry is disabled); see STS, MC.

In the RESET state, no other marker is displayed. Reappearance of the cursor indicates completion of the operation.

In the SET state, a marker (block symbol) is displayed at the present transfer pointer position.

**TSM** TABULATION STOP MODE (D,18)

In the RESET state, the controls HTS, TBC, and CTC set and clear columnar tab stops. Columnar tab stops apply to all lines in the page. They are preset at power-on to columns 1, 9, 17, 25, 33, 41, 49, 57, 65, and 73. If all columnar tab stops are cleared, a columnar tab stop is implied in column 1.

In the SET state, these controls set and clear page tab stops. Page tab stops apply to individual character positions in the page. None are set at power-on.

**TTM TRANSFER TERMINATION MODE (B,16)**

In the RESET state, transfer of characters from Display Memory terminates with the cursor position.

In the SET state, transfer of characters from Display Memory terminates with the last character position in the Page.

The character located at the terminating position is sent, if eligible to be sent; see GATM, SATM, MATM.

**VPA VERTICAL POSITION ABSOLUTE (ESC [ Pn d)**

Positions the cursor to line Pn in the active column. If zACM is reset, Pn references the beginning of the Page, and the cursor may not move outside the Page. If zACM is set, Pn references the beginning of Active Memory.

**VPR VERTICAL POSITION RELATIVE (ESC [ Pn e)**

Moves the cursor down Pn lines. This control is equivalent to CUD in the XL implementation.

**VT VERTICAL TABULATION (O/11)**

This control is equivalent to LF (O/10) in the XL implementation.

**zWBM WRAP BACKWARD MODE (D,34)**

In the RESET state, attempts to move the cursor past the left margin with the controls BS, CBT, or CUB cause the cursor to remain at the left margin.

In the SET state, attempts to move the cursor past the left margin with these controls cause the cursor to wrap to the preceding line.

**zWFM WRAP FORWARD MODE (D,33)**

In the RESET state, attempts to move the cursor past the right margin with the controls CHT, CUF, HT, or HTJ, or by the entry of graphic characters, cause the cursor to remain at the right margin.

In the SET state, attempts to move the cursor past the right margin with these controls, or by the entry of graphic characters, cause the cursor to wrap to the next line.

**XON DEVICE ON (B,-)**

This control may be sent from the terminal to the host to indicate a FIFO 'nearly empty' condition; see zAXM. The code to be used is select-

ed (in decimal) on Setup line B. Default is DC1 (1/1).

XOFF DEVICE OFF (B,-)

This control may be sent from the terminal to the host to indicate a FIFO 'nearly full' condition; see ZAXM. The code to be used is selected (in decimal) on Setup line B. Default is DC3 (1/3).

ANN ARBOR XL SERIES  
USER'S GUIDE  
READER'S COMMENT FORM

Ann Arbor has developed this User's Guide to assist you in understanding and implementing the many features contained in the Ann Arbor XL Series terminals.

Please complete and return this form if you would like to contribute suggestions that would help us in our continuing effort to improve the quality and usefulness of this document.

What was your initial reaction to this User's Guide? Would you consider it complete and accurate? Would you consider it easy to use?

What sections did you find most helpful?

What sections did you find least helpful?

What errors or omissions have you found in it?

Are you generally satisfied with the contents?

How would you rate this manual as compared to other similar documents you have used?

Other comments:

Name	Street
Title	City/State/Zip
Company	Telephone

---

Fold Here

---

Do Not Tear - Fold Here and Staple

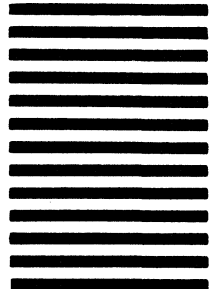


NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

**BUSINESS REPLY MAIL**  
First Class Permit No. 1682 Ann Arbor, Michigan

POSTAGE WILL BE PAID BY ADDRESSEE

**ANN ARBOR TERMINALS**  
6175 Jackson Road  
Ann Arbor, Michigan 48103



CONTENTS	Page
INTRODUCTION .....	1
KEY PROGRAMMING LOCK .....	1
SCREEN SAVER .....	1
ERASE TO NORMAL .....	1
KEY RATE CONTROL .....	1

INTRODUCTION.

The XL has been augmented with the following features.

KEY PROGRAMMING LOCK

A Lock feature has been added to the DCS control for security. The values permitted for the global parameter are now 0-3. A value of 0 or 1 restores the key programming to default before executing the programming instructions that follow. A value of 2 puts the terminal into a 'locked' state at the end of the DCS control. A value of 3 does both. In the locked state, the terminal will accept no DCS sequences until the terminal is unlocked by the operator by doing a power-on or a SETUP Z. Caution: If a DCS control with the Lock parameter is put into the power-on string, the terminal will stay always locked. The only way to clear this condition is to hold the RESET key down during power-on.

SCREEN SAVER

A screen saver has been added that blanks the screen after a period of keyboard and communications line inactivity. The period may be set from 1-120 minutes (approx) at the end of Setup Line A. A setting of zero disables the screen saver.

ERASE TO NORMAL

A private-use mode has been added to facilitate operation of the terminal with standard DEC software. DEC terminals erase to the 'normal' rendition only. AAT terminals erase to the current rendition, ie, the present setting of the SGR register. This mode provides for either.

zETN ERASE TO NORMAL (D,69)

In the reset state, the screen erases to the current graphic rendition.

In the set state, the screen erases to the 'normal' rendition.

## KEY RATE CONTROL

A private-use mode has been added to facilitate use of the terminal's programmable keys with hosts that are character input rate limited.

## zKXD KEY XMIT DELAY (B,70)

In the reset state, when a key containing a multiple-character string is depressed, the string is transmitted with no delay between characters.

In the set state, a delay is inserted between the characters reducing the transmission rate to approx. 60 chars/sec.



---

**Ann**

---

**Arbor XL**

---

**Controls**

---

**Summary**

---

**Sheet**

---

**ANN ARBOR**

# Ann Arbor XL Controls Summary

ANN ARBOR

## Cursor Controls

CR	Carriage Return	CR
LF	Line Feed	LF
BS	Backspace	BS
CUU	Cursor Up P <sub>n</sub> = # Lines	ESC [ P <sub>n</sub> A
CUD	Cursor Down P <sub>n</sub> = # Lines	ESC [ P <sub>n</sub> B
CUF	Cursor Forward P <sub>n</sub> = # Columns	ESC [ P <sub>n</sub> C
CUB	Cursor Backward P <sub>n</sub> = # Columns	ESC [ P <sub>n</sub> D
HPR	Horizontal Position Relative P <sub>n</sub> = # Columns	ESC [ P <sub>n</sub> a
VPR	Vertical Position Relative P <sub>n</sub> = # Lines	ESC [ P <sub>n</sub> e
IND	Index	ESC D
RI	Reverse Index	ESC M
NEL	Next Line	ESC E
CNL	Cursor Next Line P <sub>n</sub> = # Lines	ESC [ P <sub>n</sub> E
CPL	Cursor Preceding Line P <sub>n</sub> = # Lines	ESC [ P <sub>n</sub> F
CHA	Cursor Horizontal Absolute P <sub>n</sub> = Column #	ESC [ P <sub>n</sub> G
HPA	Horizontal Position Absolute P <sub>n</sub> = Column #	ESC [ P <sub>n</sub>
VPA	Vertical Position Absolute P <sub>n</sub> = Line #	ESC [ P <sub>n</sub> d
CUP	Cursor Position P <sub>n1</sub> = Line # P <sub>n2</sub> = Column #	ESC [ P <sub>n1</sub> ..P <sub>n2</sub> H
HVP	Horiz & Vert Position P <sub>n1</sub> = Line # P <sub>n2</sub> = Column #	ESC [ P <sub>n1</sub> ..P <sub>n2</sub> f
CPR	Cursor Position Report P <sub>n1</sub> = Line # P <sub>n2</sub> = Column #	ESC [ P <sub>n1</sub> ..P <sub>n2</sub> R
zSC	Save Cursor Position	ESC 7
zRC	Restore Cursor Position	ESC 8

## Tab Controls

HT	Horizontal Tabulation	HT
HTJ	Horizontal Tab with Justify	ESC I
CHT	Cursor Horizontal Tabulation P <sub>n</sub> = # Tab stops	ESC [ P <sub>n</sub> I
CBT	Cursor Backward Tabulation P <sub>n</sub> = # Tab stops	ESC [ P <sub>n</sub> Z
HTS	Horizontal Tabulation Set	ESC H
TBC	Tabulation Clear 0 = Clear tab stop at cursor 2 = Clear all tab stops in line 3 = Clear all tab stops	ESC [ Ps g
CTC	Cursor Tabulation Control 0 = Set tab stop at cursor 2 = Clear tab stop at cursor 4 = Clear all tab stops in line 5 = Clear all tab stops	ESC [ Ps W

## Erase Controls

ED	Erase in Display 0 = From cursor to end 1 = From start to cursor 2 = All of display	ESC [ Ps J
EL	Erase in Line 0 = From cursor to end 1 = From start to cursor 2 = All of line	ESC [ Ps K
EF	Erase in Field 0 = From cursor to end 1 = From start to cursor 2 = All of field	ESC [ Ps N
EA	Erase in Area 0 = From cursor to end 1 = From start to cursor 2 = All of area	ESC [ Ps O
ECH	Erase Character P <sub>n</sub> = # Characters	ESC [ P <sub>n</sub> X

## Edit Controls

ICH	Insert Character P <sub>n</sub> = # Characters	ESC [ P <sub>n</sub> @
DCH	Delete Character P <sub>n</sub> = # Characters	ESC [ P <sub>n</sub> P
SEE	Select Editing Extent 0 = Edit in page 1 = Edit in line 2 = Edit in field 3 = Edit in area	ESC [ Ps Q
IL	Insert Line P <sub>n</sub> = # Lines	ESC [ P <sub>n</sub> L
DL	Delete Line P <sub>n</sub> = # Lines	ESC [ P <sub>n</sub> M
zPSH	Push Line P <sub>n</sub> = # Lines	ESC [ P <sub>n</sub> s
zPOP	Pop Line P <sub>n</sub> = # Lines	ESC [ P <sub>n</sub> t
zTI	Toggle IRM	ESC 6

## Display Controls

zSDP	Set Display Parameters P <sub>n1</sub> = Page lines P <sub>n2</sub> = Upper host area lines P <sub>n3</sub> = Lower host area lines	ESC [ P <sub>n1</sub> ..P <sub>n3</sub> p P <sub>n4</sub> = Screen lines P <sub>n5</sub> = Screen columns P <sub>n6</sub> = Page columns
SGR	Select Graphic Rendition 0 = Normal      5 = Blink 1 = Bold        7 = Reverse-Video 4 = Underscore 8 = Concealed	ESC [ Ps..Ps m
zCGR	Change Graphic Rendition	ESC 9
SU	Scroll Up P <sub>n</sub> = # Lines	ESC [ P <sub>n</sub> S
SD	Scroll Down P <sub>n</sub> = # Lines	ESC [ P <sub>n</sub> T
zTFC	Toggle Fast Blink Cursor	ESC 5

## Form-Filling Controls

DAQ	Define Area Qualification 0 = Unprotected 1 = Protected/guarded 3 = Numeric entry 5 = Right justified entry	ESC [ > Ps..Ps o 7 = Set page tab stop 8 = Protected/unguarded 18 = Concealed entry
SPA	Start of Protected Area	ESC V
EPA	End of Protected Area	ESC W
zGUA	Guard Unprotected Areas	ESC :

## Send-Print Controls

SSA	Start of Selected Area	ESC F
ESA	End of Selected Area	ESC G
STS	Set Transmit State	ESC S
INT	Interrupt	ESC a
MC	Media Copy 0 = Print displayed page 4 = Turn off local copy 5 = Turn on local copy	ESC [ Ps i
zSRC	Start Remote Copy P <sub>n</sub> = # Characters	ESC [ P <sub>n</sub> v
zSPF	Set Print Format P <sub>n1</sub> = Printed lines P <sub>n2</sub> = Total lines P <sub>n3</sub> = Left margin	ESC [ P <sub>n</sub> ..P <sub>n</sub> w
zSTP	Set Transfer Pointer P <sub>n1</sub> = Line # P <sub>n2</sub> = Column #	ESC [ P <sub>n</sub> ..P <sub>n</sub> u
zSTE	Set Transfer End P <sub>n1</sub> = Line # P <sub>n2</sub> = Column #	ESC [ P <sub>n</sub> ..P <sub>n</sub> z

## Mode Controls

SM	Set Mode	ESC [ > Ps..Ps h
RM	Reset Mode	ESC [ > Ps..Ps l

## Communications Modes

zAXM	Auto Xon/Xoff Mode	B, 37
zHDM	Half-Duplex Mode	B, 40
zINM	Ignore NUL Mode	B, 54
SRM	Send-Receive Mode	B, 12

## Keyboard Modes

zCLIM	Caps Lock Invert Mode	A, 53
zDBM	Destructive Backspace Mode	D, 30
zFRM	Fast Repeat Mode	A, 47
KAM	Keyboard Action Mode	B, 2
zKCM	Key Click Mode	A, 26
zKPCM	Key Pad Control Mode	A, 27
zKRM	Key Repeat Mode	A, 28
zMBM	Margin Bell Mode	A, 25
zMKM	Meta Key Mode	B, 52
zRLM	Return Key CRLF Mode	A, 29

## Display Modes

zACM	Alternate Cursor Mode	-, 51
zAPM	Auto Pause Mode	A, 38
zBKCM	Block Cursor Mode	A, 31
zBNCM	Blinking Cursor Mode	A, 32
zCNM	CR New-Line Mode	D, 55

zDDM	DEL-Character Display Mode	D, 35
zICM	Invisible Cursor Mode	D, 56
IRM	Insertion-Replacement Mode	-, 4
zIVM	Inverse Video Mode	A, 59
LNM	LF New-Line Mode	D, 20

zSPM	Scroll-Page Mode	D, 36
zSSM	Slow Scroll Mode	A, 39
zWFM	Wrap Forward Mode	D, 33
zWBM	Wrap Backward Mode	D, 34

## Transfer Modes

zAKDM	Auto Keyboard Disable Mode	B, 46
zCSTM	Column Separator Transfer Mode	B, 43
FETM	Format Effector Transfer Mode	B, 14
zFSTM	Field Separator Transfer Mode	B, 44

zGAPM	Guarded Area Print Mode	C, 49
GATM	Guarded Area Transfer Mode	B, 1
zGRTM	Graphic Rendition Transfer Mode	B, 45
zLTM	Line Transfer Mode	B, 42

MATM	Multiple Area Transfer Mode	B, 15
SATM	Selected Area Transfer Mode	B, 17
zTPDM	Transfer Pointer Display Mode	A, 41
TTM	Transfer Termination Mode	B, 16

## Form-Filling Modes

ERM	Erasure Mode	D, 6
zHAM	Hold in Area Mode	A, 48
TSM	Tabulation Stop Mode	D, 18

## Diagnostic Modes

zMMM	Meta Monitor Mode	B, 57
------	-------------------	-------

## Miscellaneous Controls

BEL	Bell	BEL
DMI	Disable Manual Input	ESC \
EMI	Enable Manual Input	ESC b
ENQ	Enquiry	ENQ

REP	Repeat P <sub>n</sub> = # Times	ESC [ P <sub>n</sub> b
RIS	Reset to Initial State	ESC c
DCS	Device Control String	ESC P
ST	String Terminator	ESC \

DSR	Device Status Report 0 = Ready 2 = Busy 5 = Report status 6 = Report cursor position	ESC [ Ps n
-----	--	------------

DA	Device Attributes 0 = Report device attributes	ESC [ P <sub>n</sub> c
----	---	------------------------

# Keyboard

Reset 120	SetUp 121	F1 122	F2 123	F3 124	F4 125	F5 126	F6 127	F7 128	F8 129	F9 130	F10 131	F11 132	F12 133	F13 134	F14 135
--------------	--------------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	------------	------------	------------	------------	------------

F15 136	F16 137	F17 138	F18 139	Send 140
Erase 141	Edit 142	Delete 143	Insert 144	Print 145

Esc 80	! 81	@ 82	# 83	\$ 84	% 85	^ 86	& 87	* 88	( 89	) 90	- 91	= 92	~ 93	 94	Break 95
Tab 60	Q 61	W 62	E 63	R 64	T 65	Y 66	U 67	I 68	O 69	P 70	{ 71	}	Line Feed 73	Del 74	
Caps Lock 40	Ctrl 41	A 42	S 43	D 44	F 45	G 46	H 47	J 48	K 49	L 50	: 51	" 52	Return 53	Back Space 54	
Pause 20	Shift 21	Z 22	X 23	C 24	V 25	B 26	N 27	M 28	< 29	> 30	? 31	Shift 32	Scroll 33	Zoom 34	
S1 0	S2 1									S3 2	S4 3	S5 4	S6 5		6

TAB 110	7 116	↑ 8 117	9 118	- 119
	← 4 111	Home 5 112	→ 6 113	+ 114
	' 105	↓ 1 106	2 107	3 108
	∅ 100	∅ 102	.103	Enter 104

OTHER STRINGS: Header (150) Trailer (151) ENQ (152) DA (153) Power-on (200)

## Programmable Strings

DOWNLOAD SEQUENCE: ESC P > 0 ! Key number ; Shift level ; Transmit ; Repeat ; Shift ! Text ESC \

### KEY NUMBER

From keyboard diagram

### SHIFT LEVEL

0-3 = Unshift SHIFT CTRL CTRL-SHIFT  
4-31 = User defined 64 = META shift

### TRANSMIT ATTRIBUTE

0 = Follow SRM 1 = Transmit only 2 = Local only

### REPEAT ATTRIBUTE

0 = Follow zKRM 1 = Never repeat 2 = Always repeat

### SHIFT ATTRIBUTE

0 = Code-generating key 1 = Shift key

### SPECIAL CHARACTERS

! = Control field delimiter ~ = Control shift  
∅ = Special character escape

## ASCII Character Set

Column Row	0	1	2	3	4	5	6	7
0	NUL	DLE	SP	0	@	P	\	p
1	SOH	DC1	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3	#	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	/	?	0	_	o	DEL

## Setup Exits

S = Save  
M = Data Monitor Line  
T = Local Test  
X = Transparent Mode  
Y = Diagnostic (Long)  
Z = Diagnostic (Short)

Ann Arbor Terminals, Inc.  
6175 Jackson Road  
Ann Arbor, Michigan 48103  
Phone: 313/663-8000  
Telex: 706905